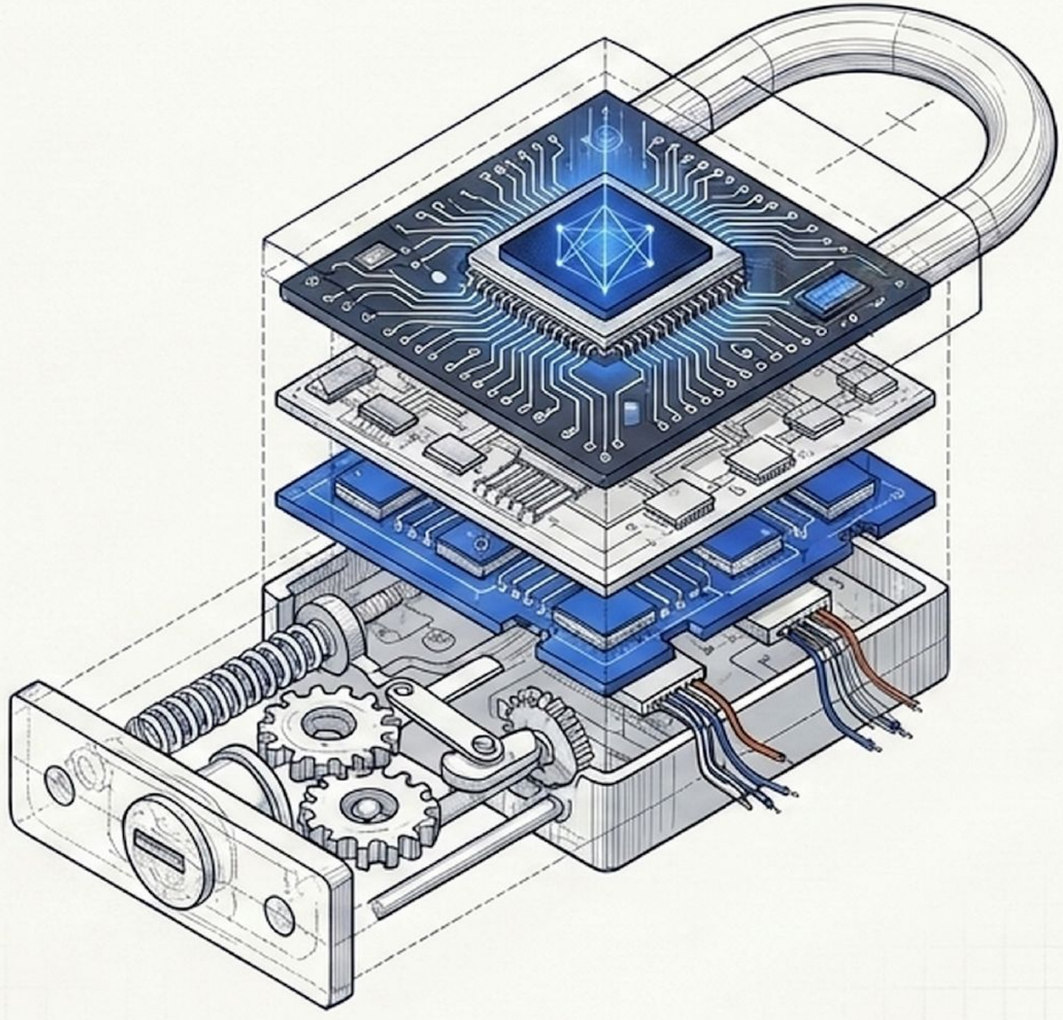
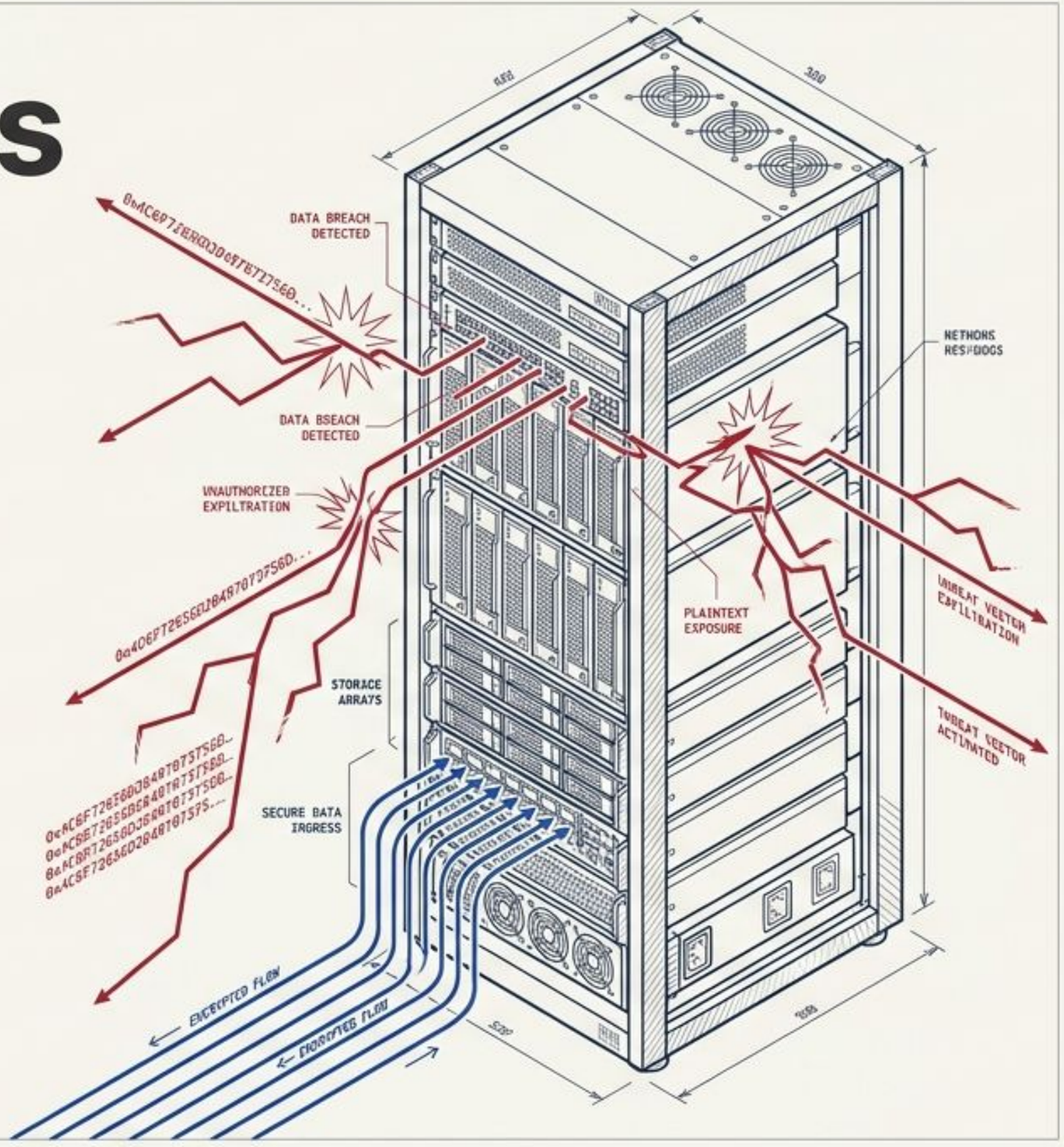
A Anatomia da Criptografia de Senhas



O Banco de Dados Vazou. E Agora?

Uma dissecação técnica sobre o **armazenamento de senhas**: do desastre do texto puro ao estado da arte criptográfico.



A mentalidade arquitetural para segurança de dados.

A Ilusão da Segurança e a Era do Plain Text

O Padrão dos Anos 90

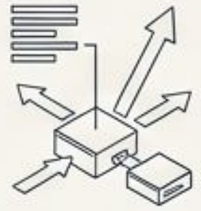
Armazenamento em texto puro (ex: `senha123`)

Vulnerabilidade primária: **SQL Injection**

Consequência: O vazamento expunha a senha exata, permitindo ataques em cascata a outros serviços do usuário.

O Fator RockYou

2009



32 milhões de senhas em texto puro expostas no vazamento da RockYou.

HOje

Listas consolidadas contêm mais de **29.6 bilhões** de senhas reais vazadas.

O Perigo: Esse arsenal permitiu a criação dos ataques de **Pré-computação** (Rainbow Tables).

A Ilusão do Texto Puro e o Efeito Cascata

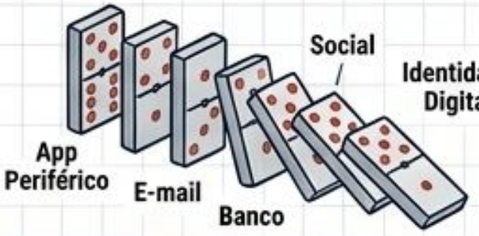
O Caso RockYou (2009)

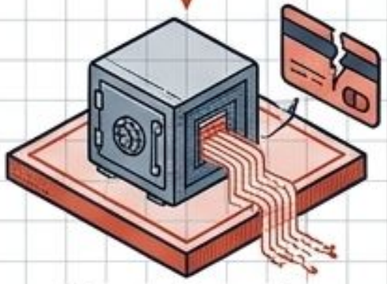
32 milhões de senhas vazadas em Plain Text (texto puro). O marco zero dos mega-vazamentos.

RockYou - 2009

O Efeito Cascata

O ser humano recicla senhas constantemente. O vazamento de um aplicativo periférico destrói a identidade digital inteira de um usuário em múltiplos serviços.





Bancos Pessoais

E-mails Corporativos

Redes Sociais

O Legado: The RockYou List

Atualizada anualmente por hackers, a base de dados de dicionário moderno contém:

317 GB
de Arquivo TXT

29.6 Bilhões
de Senhas Reais

A Primeira Defesa: O Efeito Avalanche dos Hashes (MD5 / SHA)

Determinístico: A mesma entrada sempre gera o mesmo hash.	**Unidirecional:** Impossível reverter matematicamente o hash para o texto original.

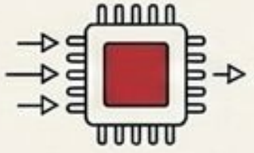
The Avalanche Visualizer

Apenas alterar um **s** minúsculo para maiúsculo embaralha 100% da string criptográfica resultante.

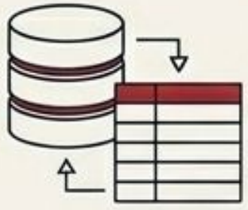
A Falha Fatal: Quando a Velocidade se Torna um Inimigo

O Problema: MD5 e SHA foram desenhados para velocidade extrema, não para proteção de senhas.

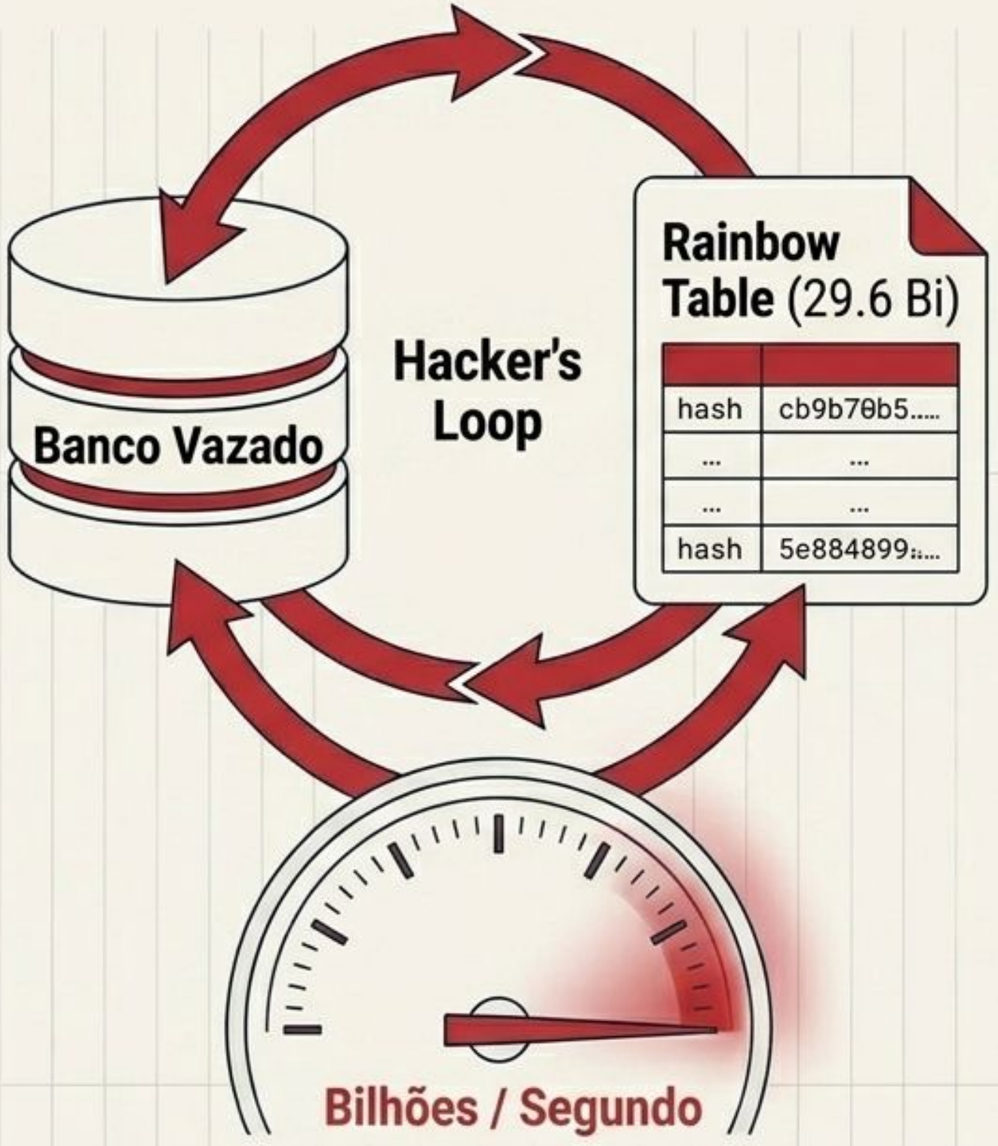
A Capacidade: Um hardware comum gera hashes na casa dos bilhões por segundo.



O Ataque: O hacker pré-computa hashes para 29.6 bilhões de senhas e apenas compara os resultados com o banco.



Se a Maria e a Juliana usam a mesma senha, ambas terão o mesmo hash. Quebrar uma expõe a outra.



Mitigação Intermediária: Adicionando Salt

O Conceito	O Resultado	Derrota da Rainbow Table
O sistema gera uma string aleatória (Salt) e a concatena com a senha.	Senhas idênticas agora geram hashes completamente diferentes.	Tabelas pré-computadas tornam-se inúteis. O atacante precisa refazer o cálculo para cada usuário específico.

A Ameaça Persiste: O algoritmo ainda é rápido demais. O hacker continua varrendo bilhões de combinações em segundos. O Salt atrasa, mas não impede a força bruta.

A Era CPU-Hard: Desacelerando o Ataque com Bcrypt

Bcrypt introduz o **Fator de Trabalho logarítmico**.
O objetivo não é a velocidade, mas ser propositalmente lento.

```
$2a$12$R9h/cIPz0gi.URNNX3rub.3...
```

Versão do algoritmo.

Fator de Trabalho / Custo. (Ex: Custo 12 = 4.096 iterações).
Força a CPU a refazer o cálculo, levando ~323 milissegundos por hash.

Salt gerado automaticamente (22 caracteres).

O Hash Final.

Impacto: Limita o atacante a testar apenas ~3 hashes por segundo por núcleo.

A Reviravolta do Hardware: O Poder do Paralelismo (GPU)

A Evolução

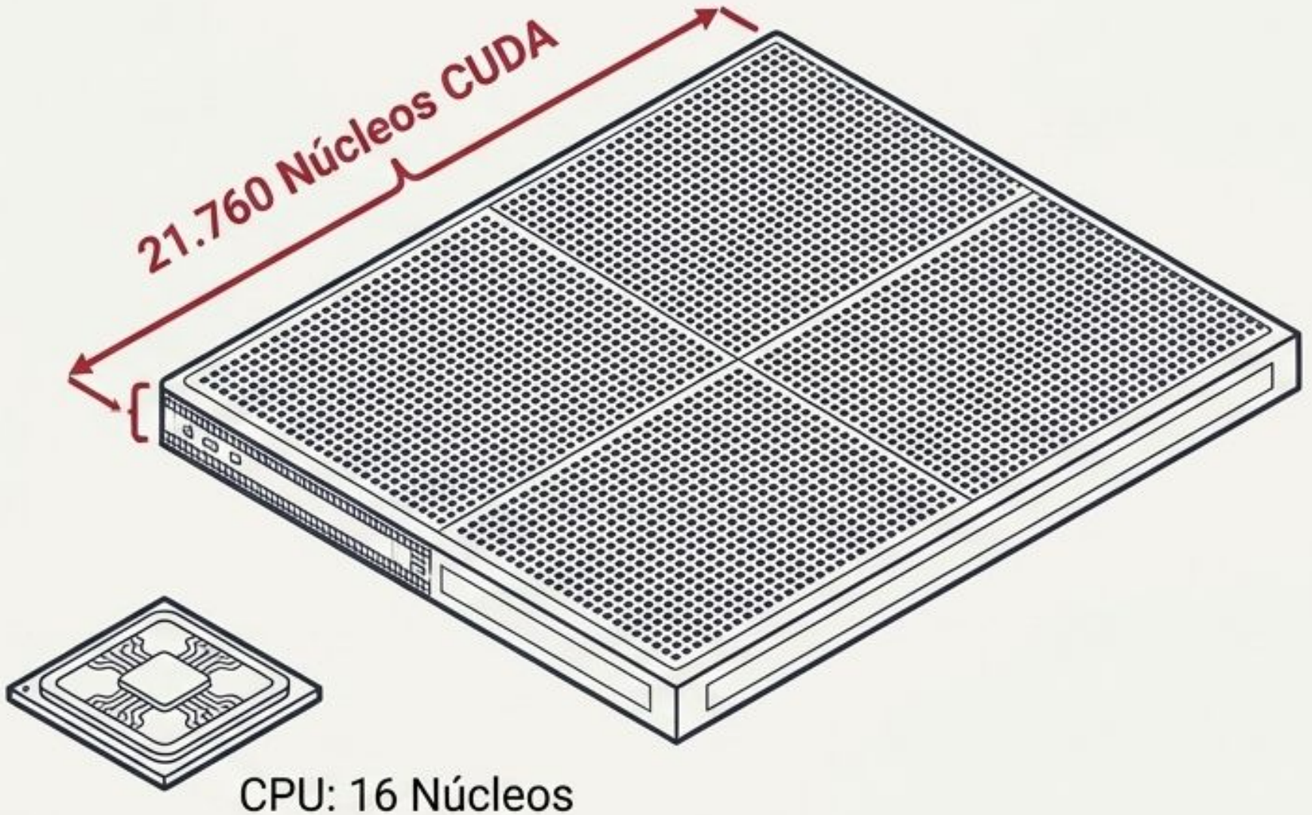
CPUs comuns possuem 16 ou 24 núcleos. Mas as GPUs modernas mudaram as regras do jogo.

A Falha do Bcrypt

Ocupa apenas 4 KB de memória. Pode ser facilmente alocado e paralelizado em milhares de núcleos simultaneamente.

O Veredito

Uma única rig de GPUs modernas varre bilhões de possibilidades por dia.
O Bcrypt já não é intransponível.



3 hashes/seg x 21.760 núcleos
= 65.280 hashes por segundo.

Diagnóstico de Ameaça: O Retorno Sobre o Investimento do Hacker

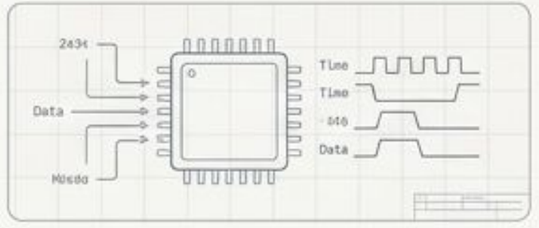
A capacidade computacional destruiu a barreira do tempo.

Algoritmo & Hardware	Taxa de Teste	Tentativas por Dia	Veredito
MD5 (CPU)	Bilhões / seg	Incalculável	Vulnerabilidade Crítica
Bcrypt (CPU 16 cores)	48 / seg	4.14 Milhões	Seguro contra CPUs
Bcrypt (GPU RTX 5090)	65.280 / seg	5.6 Bilhões	Vulnerável ao Paralelismo

O Desafio: Precisamos atacar a única fraqueza que resta nas GPUs para impedir o paralelismo.

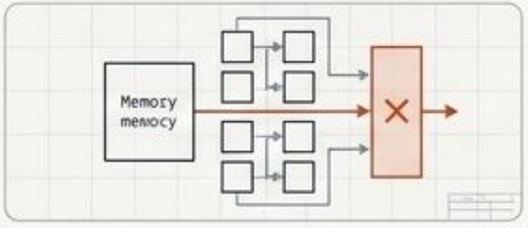
O Atual Estado da Arte: Argon2

Vencedor Invicto: Password Hashing Competition 2015



Argon2i

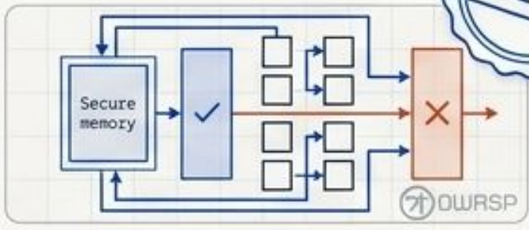
Otimizado contra ataques de canal lateral (Side-channel attacks). Ideal para instâncias onde o tempo de processamento é estritamente monitorado.



Argon2d

Focado especificamente em máxima resistência contra quebra bruta por GPU, dificultando o acesso sequencial.

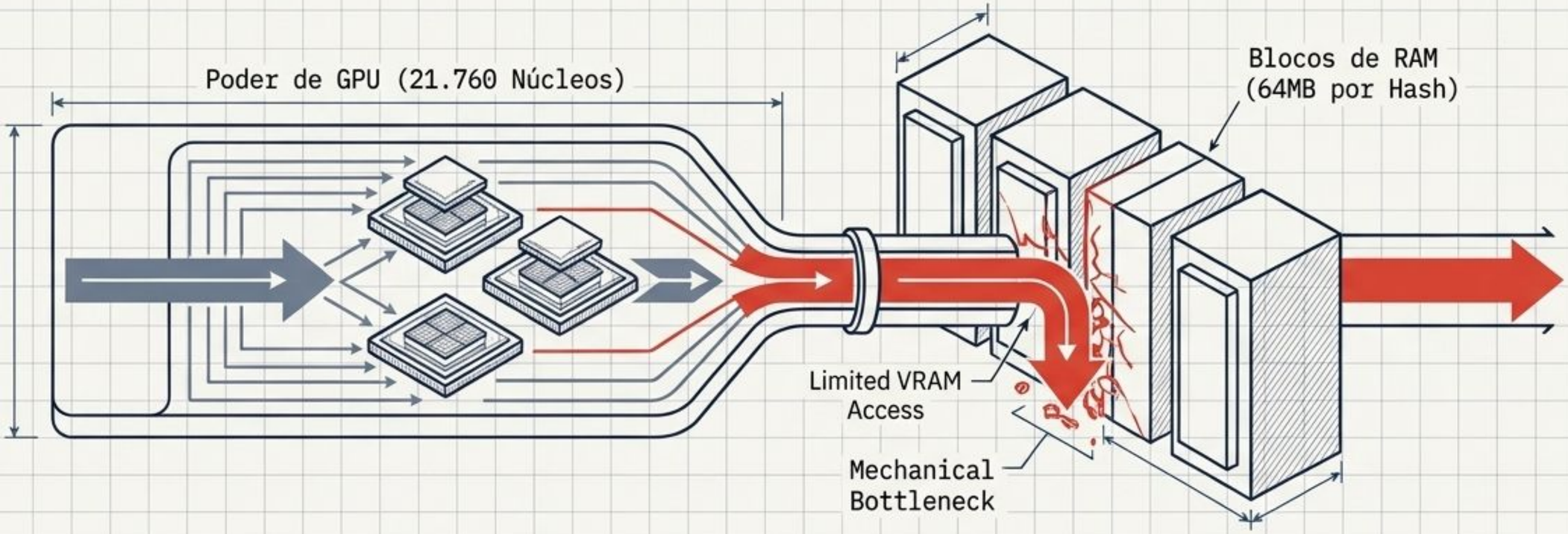
RECOMENDADO



Argon2id

O algoritmo híbrido definitivo. Une a proteção de memória do 'i' com a resistência massiva de GPU do 'd'. É o atual padrão ouro recomendado pela fundação OWASP.

Dissecando o Argon2: A Estratégia Memory-Hard



A Lógica do Gargalo

GPUs possuem um poder de processamento matemático colossal, mas a memória VRAM compartilhada é extremamente limitada (ex: 24GB ou 32GB).

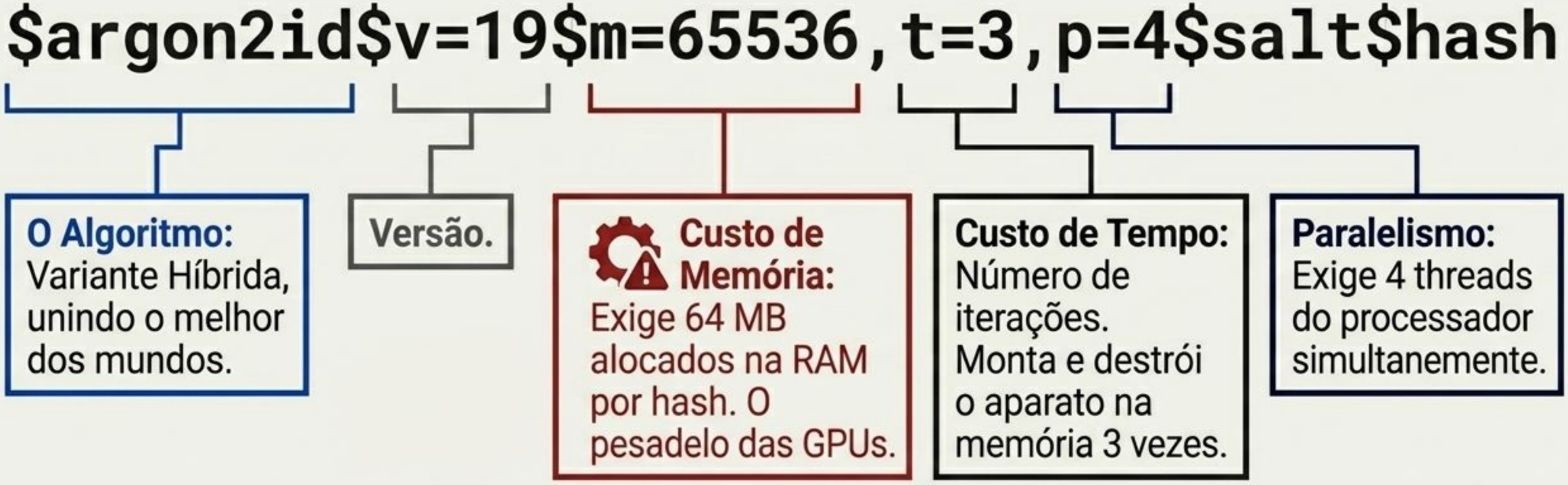
Os 3 Pilares Configuráveis

Ao contrário de algoritmos antigos, o Argon2 exige configuração de três restrições: **Time Cost** (Iterações), **Parallelism** (Bloqueio de threads) e **Memory Cost** (Preenchimento de RAM real).

A Mecânica da Defesa

Tentar paralelizar 1.000 ataques simultâneos de Argon2 configurado a 64MB exige 64GB de VRAM instantaneamente, esgotando a placa de vídeo e inviabilizando a força bruta física.

Anatomia da Defesa: Dissecando o Argon2id



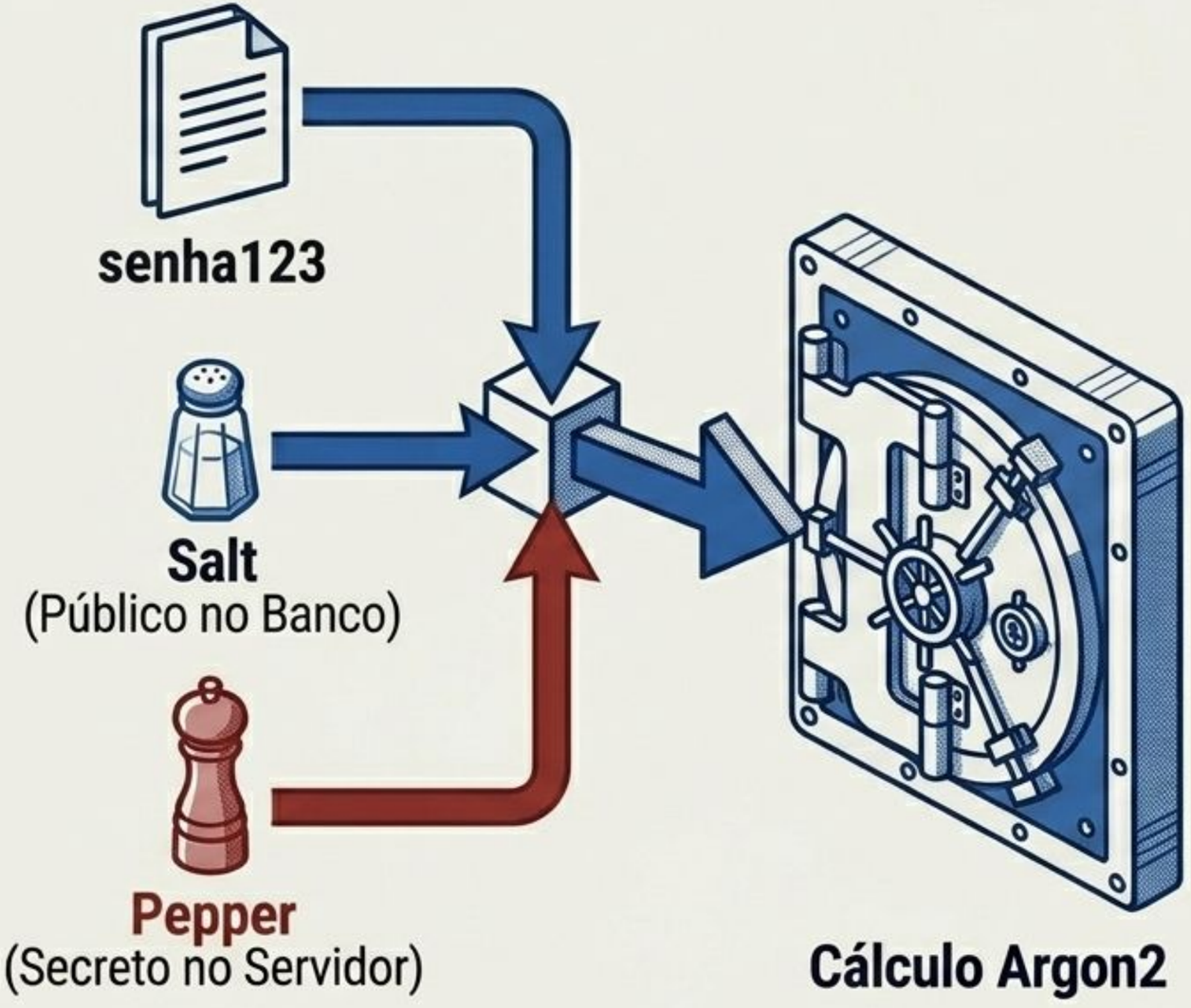
Um algoritmo propositalmente lento, volumoso e pesado.
A defesa perfeita contra paralelismo.

O Último Cadeado: A Pimenta (Pepper)

O Conceito:
Mesmo com Argon2, a matemática do hash pode ser atacada se o banco inteiro (incluindo o Salt) for exposto.

A Solução:
Uma string secreta, única e global armazenada exclusivamente nas variáveis de ambiente (.env) do servidor.

A Defesa:
A aplicação concatena a senha com o Pepper ANTES de gerar o hash. Sem acesso à infraestrutura, é matematicamente impossível iniciar o ataque.



Isolamento Arquitetural: Onde Vivem os Segredos

Zona Comprometida: Banco de Dados

Armazena: O Hash Final e o Salt.

Natureza do Salt: Único por usuário, não necessita ser oculto.

Status em Vazamento: Totalmente exposto ao Hacker.

Separação de Contexto

Zona Segura: Servidor de Aplicação

Armazena: O Pepper (Arquivo `.env`).

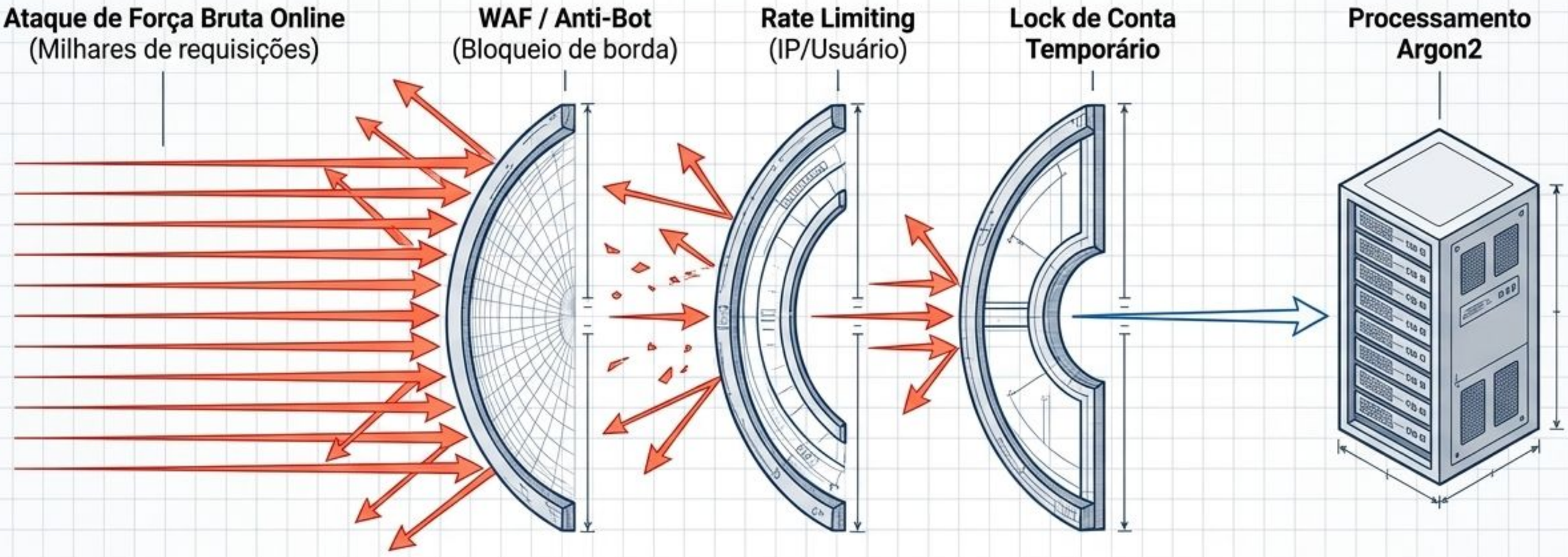
Natureza do Pepper: Global, super secreto, chave-mestra do cálculo.

Status em Vazamento: Totalmente inacessível via banco de dados.

A Corrida Armamentista Criptográfica

Algoritmo	Mecanismo Base	Vetor de Ataque	Veredito
Texto Puro & MD5	Rapidez Extrema	Rainbow Tables / Força Bruta	**Desastre Total**
MD5 + Salt	Quebra tabelas pré-computadas	Força bruta focada (~bilhões/s)	**Obsoleto**
Bcrypt (CPU-Hard)	Desacelera a CPU logaritmicamente	Rigs de GPU Paralelizadas	Inseguro contra hardware moderno
Argon2id + Pepper (Memory-Hard)	Esgota a RAM/VRAM; exige chave do servidor	Falha por falta de memória e falta do `.env`	**Estado da Arte**

Gerenciando o Alto Custo do Argon2 em Produção



O Medo da Infraestrutura

Se o Argon2 consome tanta RAM, uma rajada de logins não causará um auto-DDoS derrubando meu próprio servidor?

Separação de Responsabilidades

A Criptografia (Argon2) deve lidar estritamente com ataques OFFLINE (o banco foi roubado). Quem lida e barra os ataques ONLINE (rajadas de login) é a arquitetura e proteção de borda da sua aplicação.

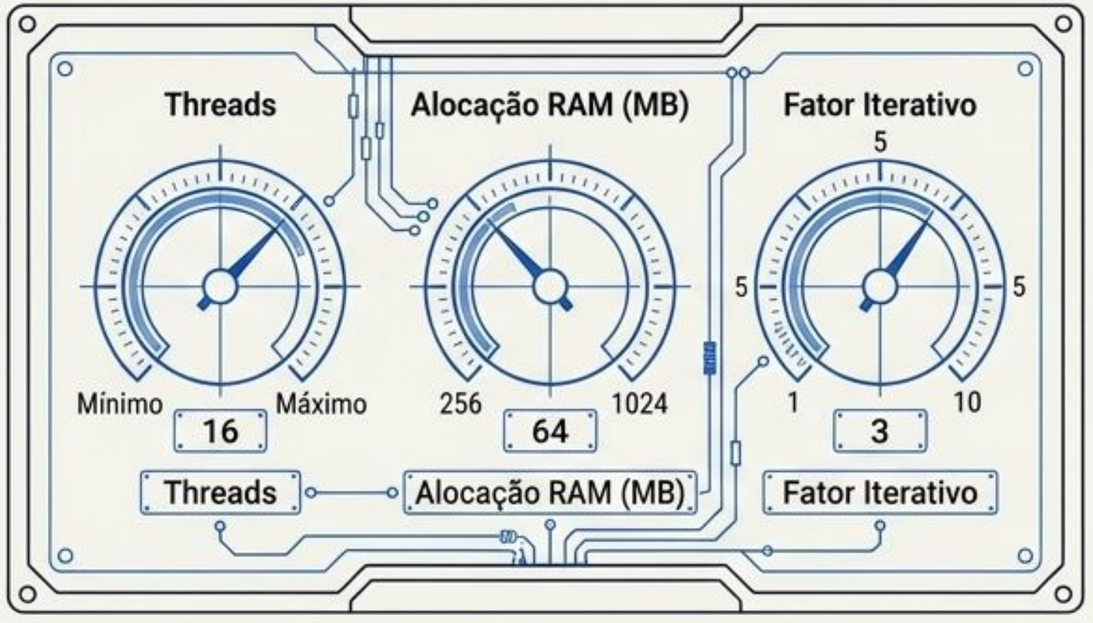
O Perigo do Modo Padrão (Default)

O Risco da Confiança Cega

Programadores comuns utilizam funções de hash de frameworks sem entender as engrenagens. No modo padrão, você não sabe quanto de RAM o Argon2 consome ou se o custo do Bcrypt resiste às GPUs atuais.

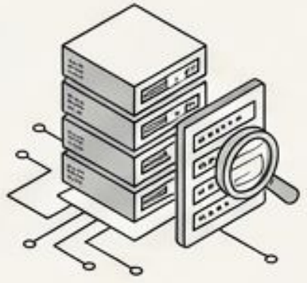
A Postura Arquitetural

Se você não entende os fundamentos de CPU, Paralelismo e Memória RAM, não saberá se defender no dia do vazamento. Utilizar a biblioteca não é arquitetura; é apenas esperança.

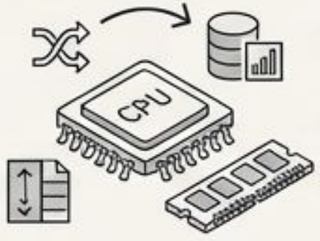


Adote a Mentalidade Arquitetural

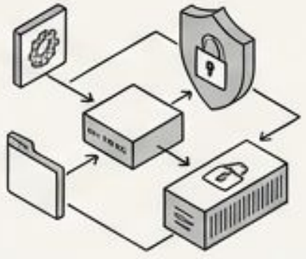
Segurança não é sobre instalar a biblioteca mais recente. É sobre entender o hardware, antecipar o paralelismo dos ataques e arquitetar os gargalos a seu favor.



Audite as senhas do seu sistema legado hoje.



Migre a infraestrutura do Bcrypt para o Argon2id.



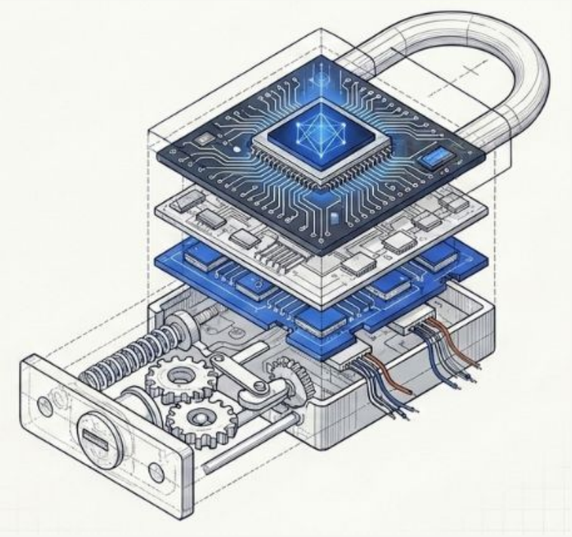
Isole a criptografia utilizando a técnica de Pepper.

A jornada exige domínio total das engrenagens internas. Assuma o controle da sua infraestrutura.

A Anatomia da Criptografia de Senhas

Autor: Luiz Fernando Zacarão

Fontes:



Sobre armazenar senhas no banco de dados: https://www.youtube.com/watch?v=VW2mywTTz80

What Is Argon2?: https://jumpcloud.com/it-index/what-is-argon2

Quantum Computing & Encryption: What It Means for Security?:
https://www.compassitc.com/blog/what-will-quantum-computing-mean-for-passwords-and-encryption