Lyra2

Password Hashing Scheme with improved security against time-memory trade-offs (TMTO)

Ewerton Rodrigues Andrade

eandrade@larc.usp.br

Escola Politécnica da Universidade de São Paulo – EP/USP

Agências de fomento:
CAPES, FDTE e Erasmus Mundus

Defesa de Tese de Doutorado

07 de junho de 2016

Banca Examinadora:

- Prof Dr Marcos Antonio Simplicio Junior – EP/USP *(presidente)*
- Prof Dr Wilson Vicente Ruggiero – EP/USP
- Profª Drª Denise Hideko Goya – CMCC/UFABC
- Prof Dr Diego de Freitas Aranha – IC/UNICAMP
- Dr Rafael Misoczki – Intel Labs

Sumário

1. Introdução
   - Motivação
   - Objetivos
   - Metodologia
2. Lyra2
   - *The Bootstrapping phase*
   - *The Setup phase*
   - *The Wandering phase*
   - *The Wrap-up phase*
3. Lyra2 x scrypt x finalistas do PHC
   - Segurança
   - Desempenho
4. BlaMka
   - Resultados Parciais
5. Considerações Finais
   - Principais Resultados
   - Trabalhos Futuros

Sumário

1. Introdução
   - Motivação
   - Objetivos
   - Metodologia
2. Lyra2
   - *The Bootstrapping phase*
   - *The Setup phase*
   - *The Wandering phase*
   - *The Wrap-up phase*
3. Lyra2 x scrypt x finalistas do PHC
   - Segurança
   - Desempenho
4. BlaMka
   - Resultados Parciais
5. Considerações Finais
   - Principais Resultados
   - Trabalhos Futuros

Motivação

A **autenticação é vital** para a segurança dos sistemas computacionais modernos

Motivação

A **autenticação é vital** para a segurança dos sistemas computacionais modernos

KNOW	HAVE	ARE
		
Passwords ID Questions Secret Images	Token (Smart) Card Phone	Face Iris Hand/Finger

Motivação *(Cont.)*

A maioria dos usuários escolhe senhas com uma **entropia relativamente baixa** (aproximadamente 40.5 bits [FH07])

Facilitando a execução de ataques de "força-bruta":

- Dicionário
- Busca exaustiva
- Tabelas pré-calculadas (*Rainbow tables*, tabelas de hashs, ...)

Motivação *(Cont.)*

A maioria dos usuários escolhe senhas com uma **entropia relativamente baixa** (aproximadamente 40.5 bits [FH07])

Facilitando a execução de ataques de "força-bruta":

- Dicionário
- Busca exaustiva
- Tabelas pré-calculadas (*Rainbow tables*, tabelas de hashs, ...)

Como aumentar o custo destes ataques?

Empregando **Esquemas de Hash de Senhas** (PHS):

PBKDF2
bcrypt
scrypt
Lyra

PHS ao longo dos anos

- 60's

armazenar: $"password"$

- 70's

armazenar: $hash("password")$

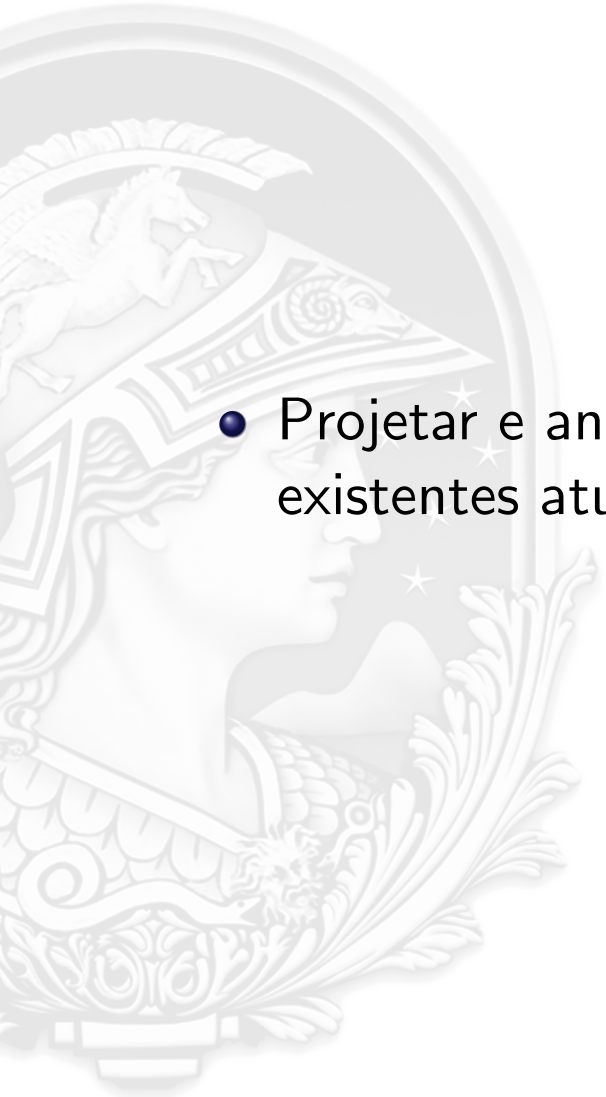
- final dos 70's, 80's e meados dos 90's

armazenar: $hash("password", salt)$

- 2000's ...

armazenar: $hash("password", salt, cost)$

Objetivo Geral



- Projetar e analisar uma **melhor alternativa** aos algoritmos existentes atualmente

Objetivos Específicos

Manutenção

Manter a eficiência e flexibilidade do Lyra, o que inclui:

- A capacidade de **configurar a quantidade de memória e o tempo** de processamento utilizados pelo algoritmo *(flexibilidade)*
- A capacidade de **utilizar mais memória** para um tempo de processamento similar ao do scrypt *(eficiência)*

Objetivos Específicos

Manutenção

Manter a eficiência e flexibilidade do Lyra, o que inclui:

- A capacidade de **configurar a quantidade de memória e o tempo** de processamento utilizados pelo algoritmo *(flexibilidade)*
- A capacidade de **utilizar mais memória** para um tempo de processamento similar ao do scrypt *(eficiência)*

Melhoria *(segurança)*

Em comparação ao seu predecessor, o Lyra2 adiciona:

- Melhorias no nível de segurança contra ataques que substituam memória por tempo de processamento (**TMTO** – *time-memory trade-offs*)
- Ajustes que aumentam o custo envolvido na **construção de um hardware** dedicado para atacar o algoritmo
- Equilíbrio entre ataques de **canal colateral** *(side-channel)* e ataques que se baseiam no uso de dispositivos de **memória mais barata** (e, consequentemente, mais lenta)

Metodologia

- Pesquisa aplicada, baseada em **hipótese-dedução**, utilizando referências científicas para definição do problema, especificação da hipótese de solução e sua avaliação

- **Principais etapas:**
  - Revisão da literatura
  - Projeto do algoritmo
  - Comparação com as soluções existentes
  - Escrita da tese e artigos

Sumário

1. Introdução
   - Motivação
   - Objetivos
   - Metodologia
2. Lyra2
   - *The Bootstrapping phase*
   - *The Setup phase*
   - *The Wandering phase*
   - *The Wrap-up phase*
3. Lyra2 x scrypt x finalistas do PHC
   - Segurança
   - Desempenho
4. BlaMka
   - Resultados Parciais
5. Considerações Finais
   - Principais Resultados
   - Trabalhos Futuros

Visão geral *(Esponjas criptográficas)*

- Construído sobre a estrutura de **Esponjas criptográficas**

Por que?

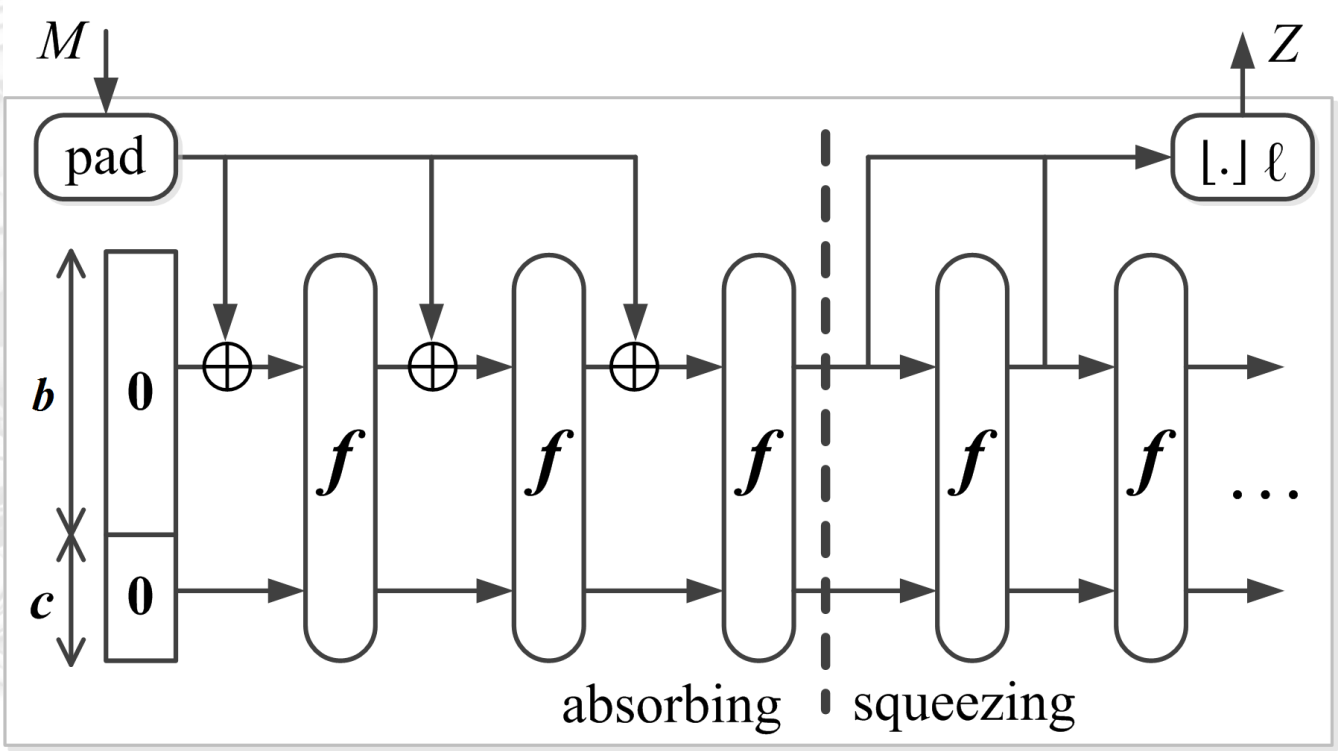
Elegância, Flexibilidade, **Segurança**

Visão geral *(Esponjas criptográficas)*

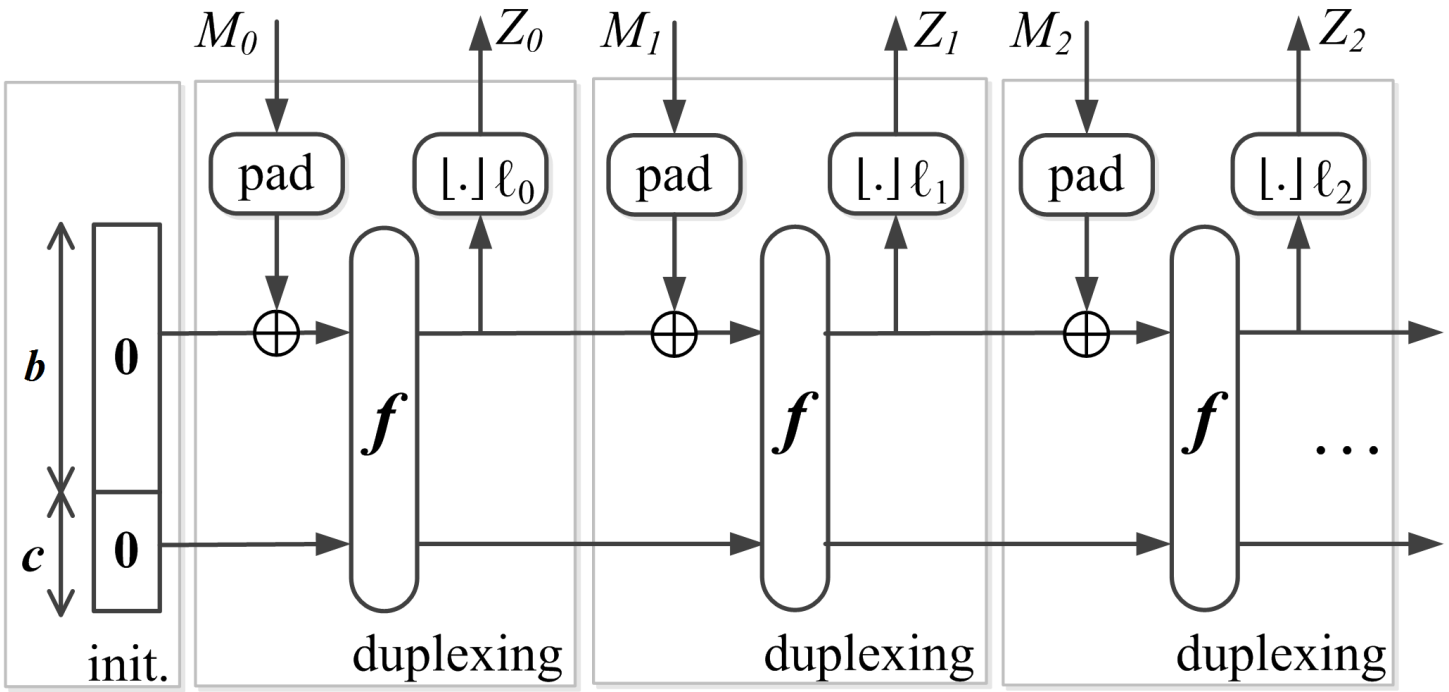
- Construído sobre a estrutura de **Esponjas criptográficas**

Por que?

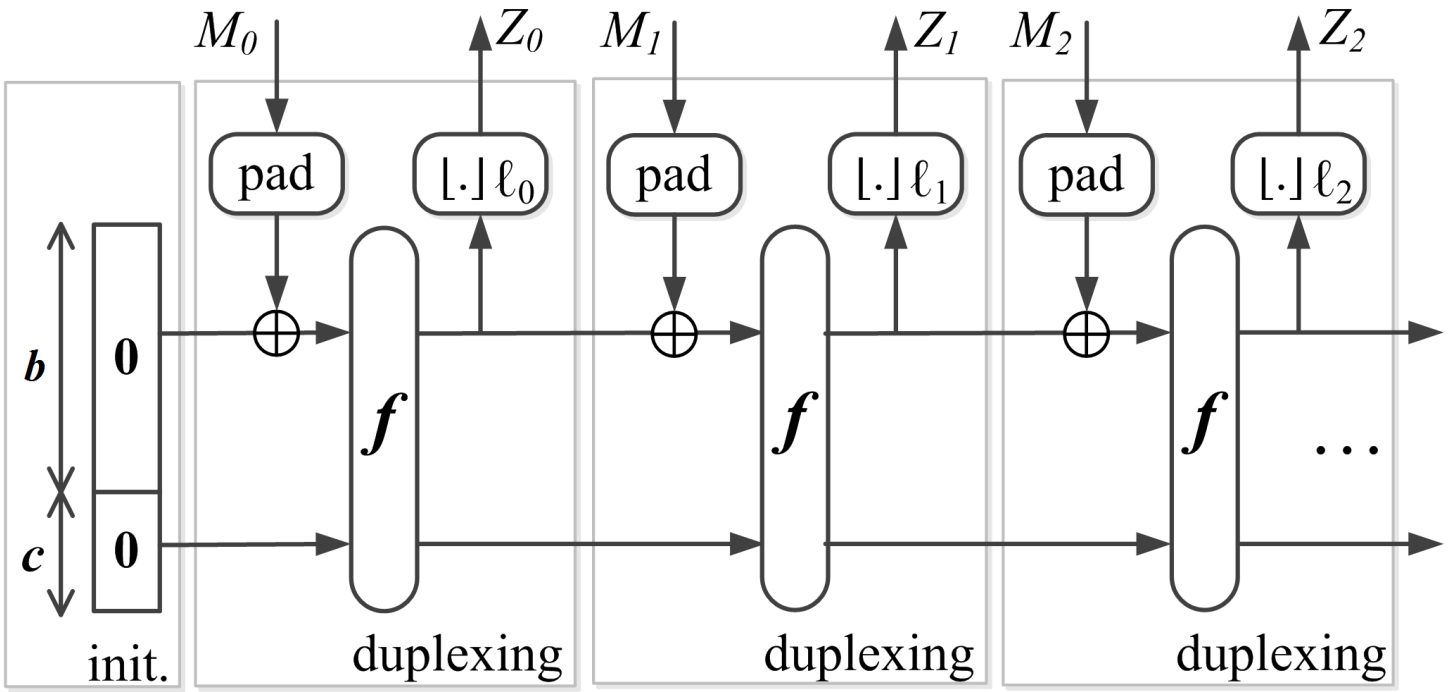
Elegância, Flexibilidade, **Segurança**



Visão geral *(Esponjas criptográficas)*



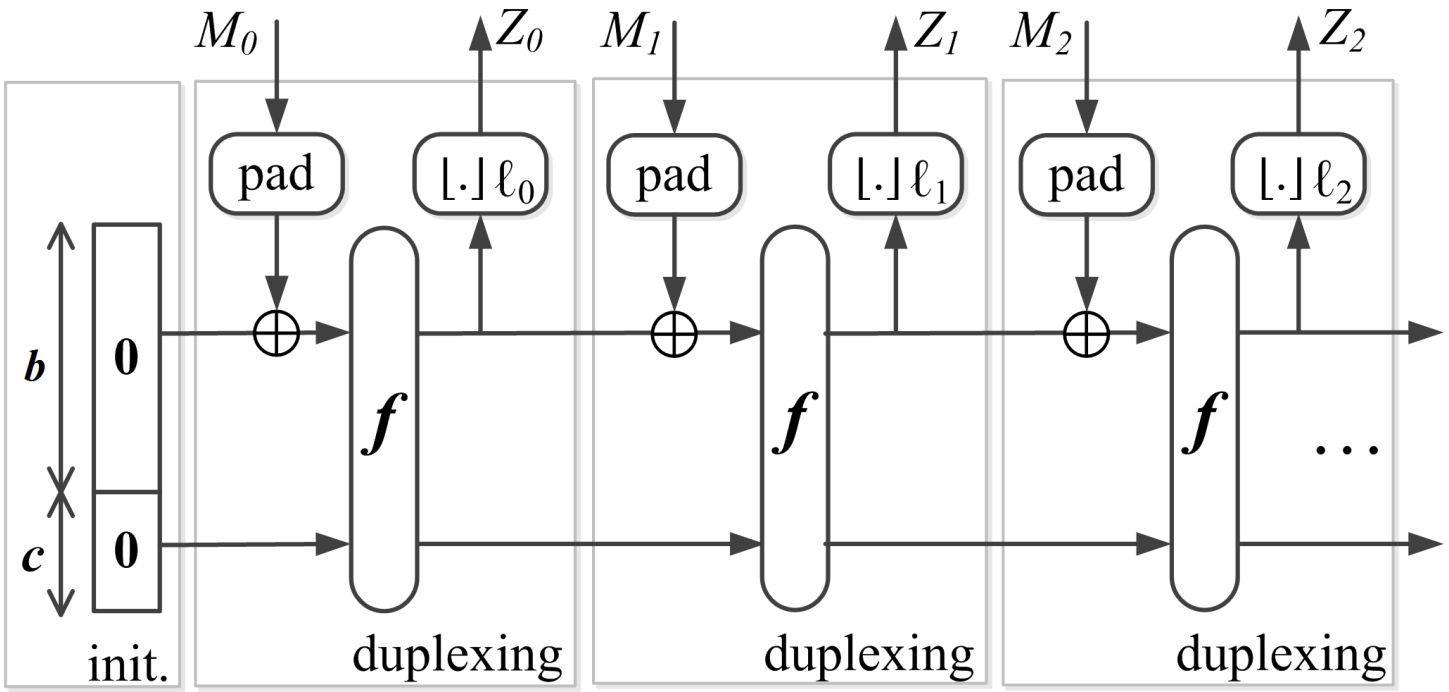
Visão geral *(Esponjas criptográficas)*



Instâncias

- Keccak (SHA-3), Quark, Photon, Spongent, Gluon ... [BDPA07]

Visão geral *(Esponjas criptográficas)*



Instâncias

- Keccak (SHA-3), Quark, Photon, Spongent, Gluon ... [BDPA07]

PHC special recognition

"pelo design elegante, baseado em esponjas criptográficas" [PHC15]

Visão geral *(Lyra2)*

- Baseado em quatro fases
  - **Bootstrapping**: Inicializa a esponja e as variáveis utilizadas pelo algoritmo
  - **Setup**: Inicializa a matriz de memória
  - **Wandering**: Visita e reescreve a matriz de memória iterativamente
  - **Wrap-up**: Provê a saída

The Bootstrapping phase

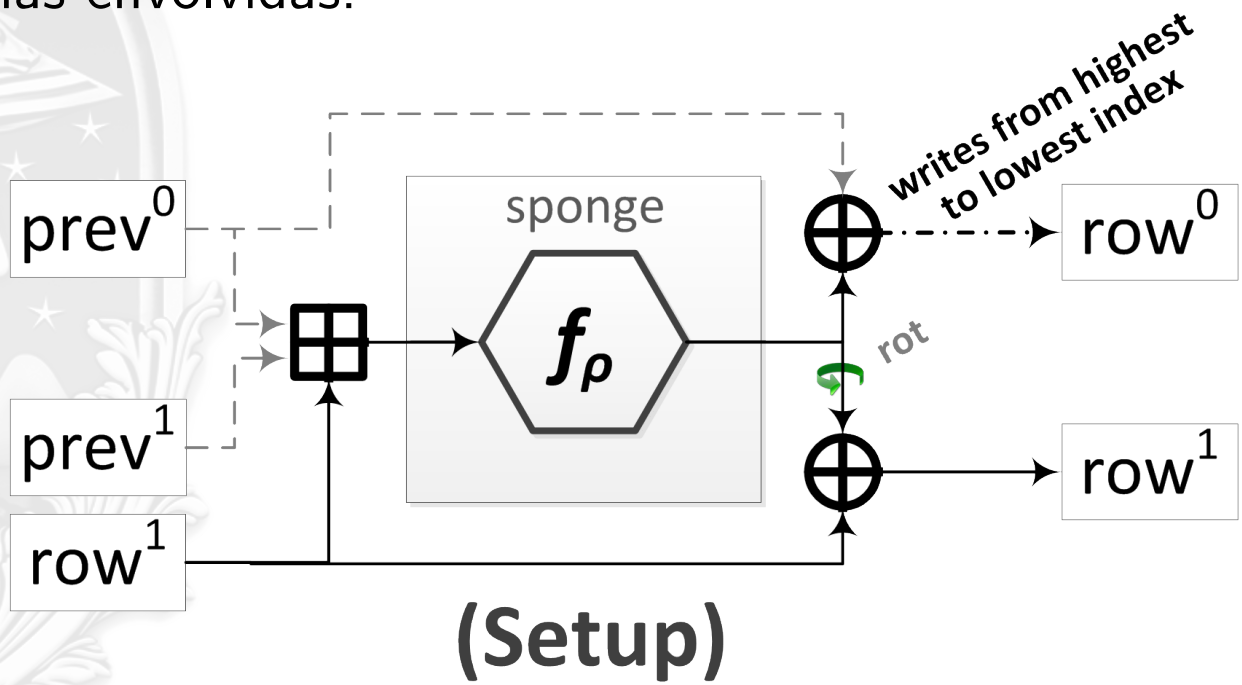
Inicializa a esponja e as variáveis utilizadas pelo algoritmo

- Absorve (operação *absorb*): *pwd*, *salt*, e *parameters*
- Inicializa demais variáveis *(contadores)*

The Setup phase

Inicializa a matriz de memória

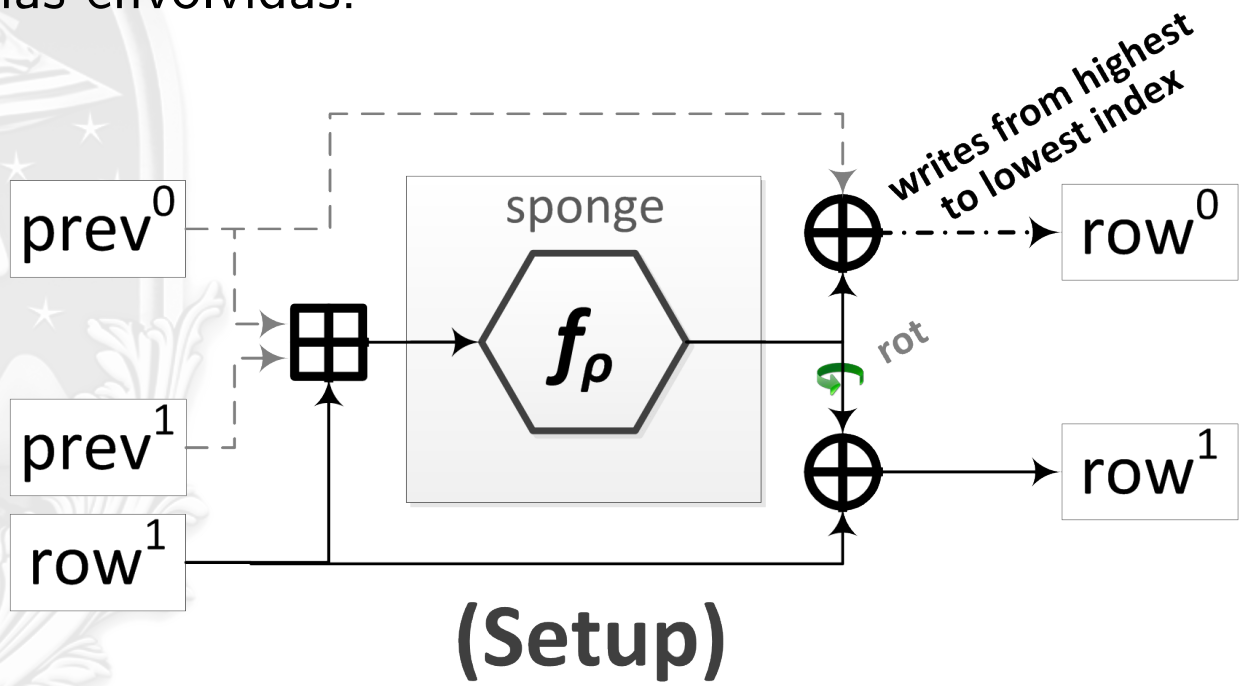
- Determinística *(i.e., protege de ataques por canal colateral)*
- Linhas envolvidas:



The Setup phase

Inicializa a matriz de memória

- Determinística *(i.e., protege de ataques por canal colateral)*
- Linhas envolvidas:

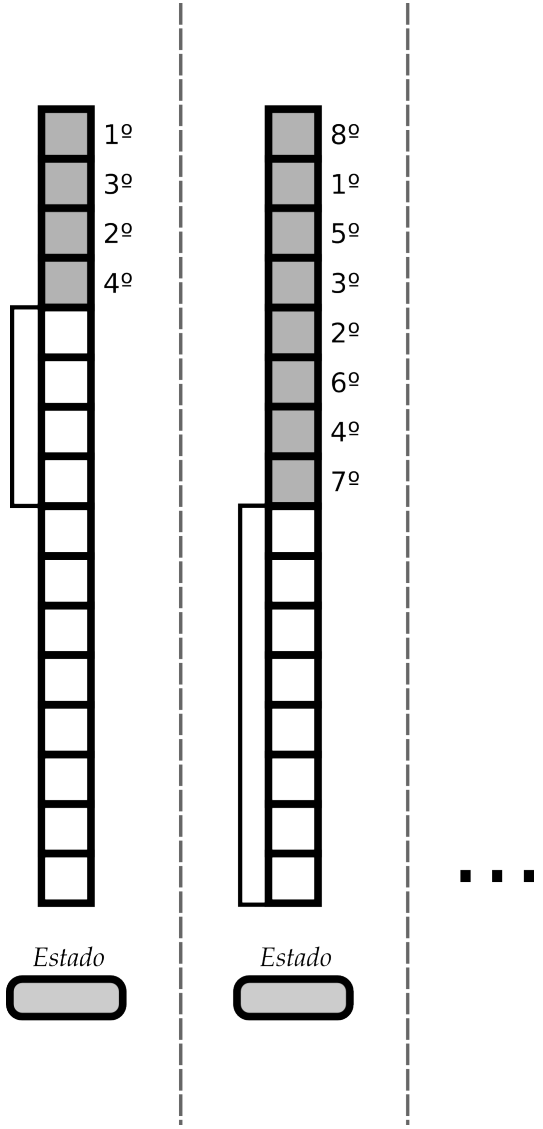


- Dificulta o **pipelining**, e aumenta a **latência em hardware** *(em ataques)*

The Setup phase (natureza determinística)

$$\begin{bmatrix} \mathbf{0} & 2 \\ 1 & 3 \end{bmatrix} \qquad \begin{bmatrix} \mathbf{0} & 4 \\ 1 & 5 \\ 2 & 6 \\ 3 & 7 \end{bmatrix}$$

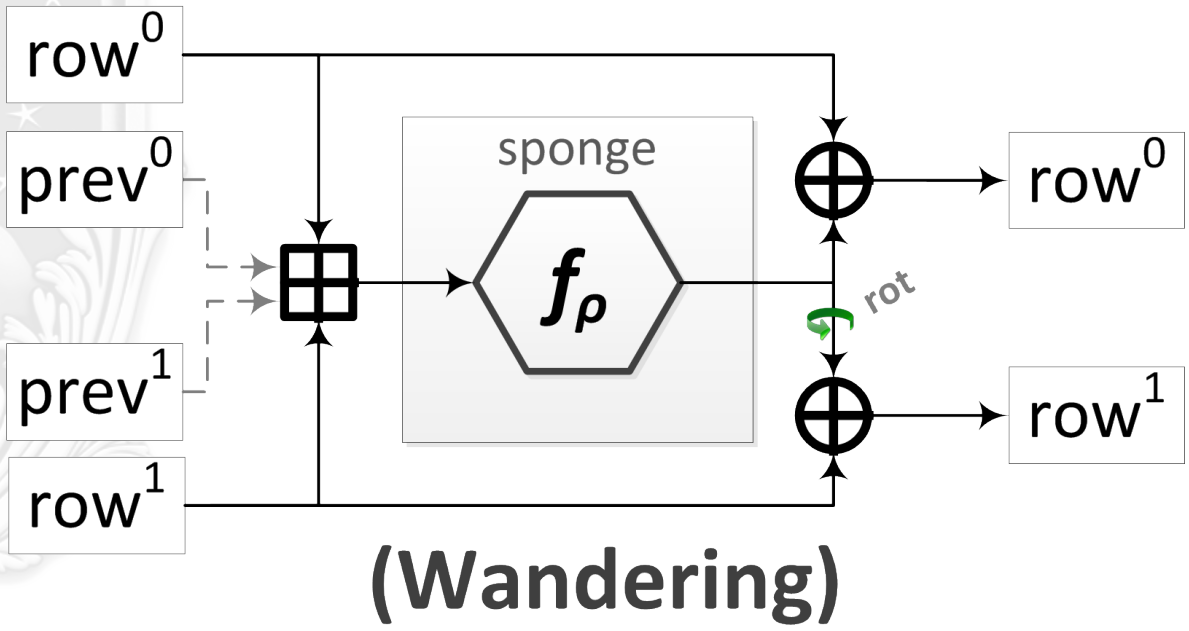
- Caso *wnd* possua **raiz quadrada**:
  - Diagonais cíclicas serão visitadas
  - $sqrt = \sqrt{wnd}$
- **Caso contrário**:
  - Anti-diagonais cíclicas serão visitadas
  - $sqrt = 2\sqrt{wnd/2}$



The Wandering phase

Visita e reescreve a matriz de memória iterativamente

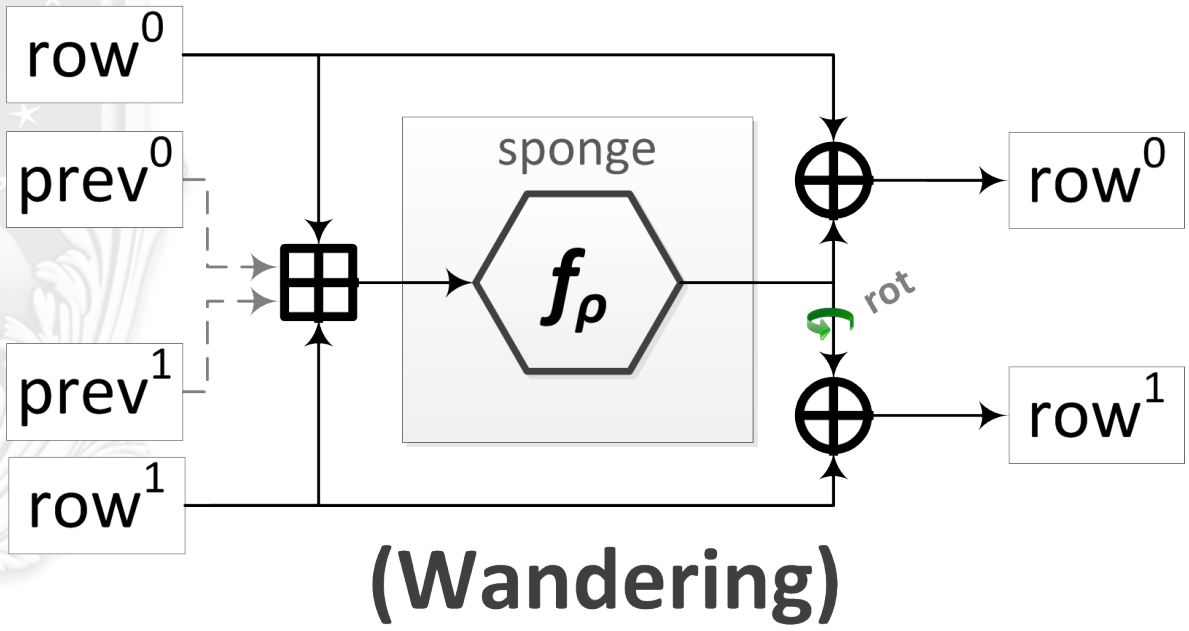
- Natureza pseudo-aleatória *(aumento o TMTO)*
- As colunas também são selecionadas pseudo-aleatoriamente *(diminui o desempenho: GPUs e plat. com cache pequeno)*
- Linhas envolvidas:



The Wandering phase

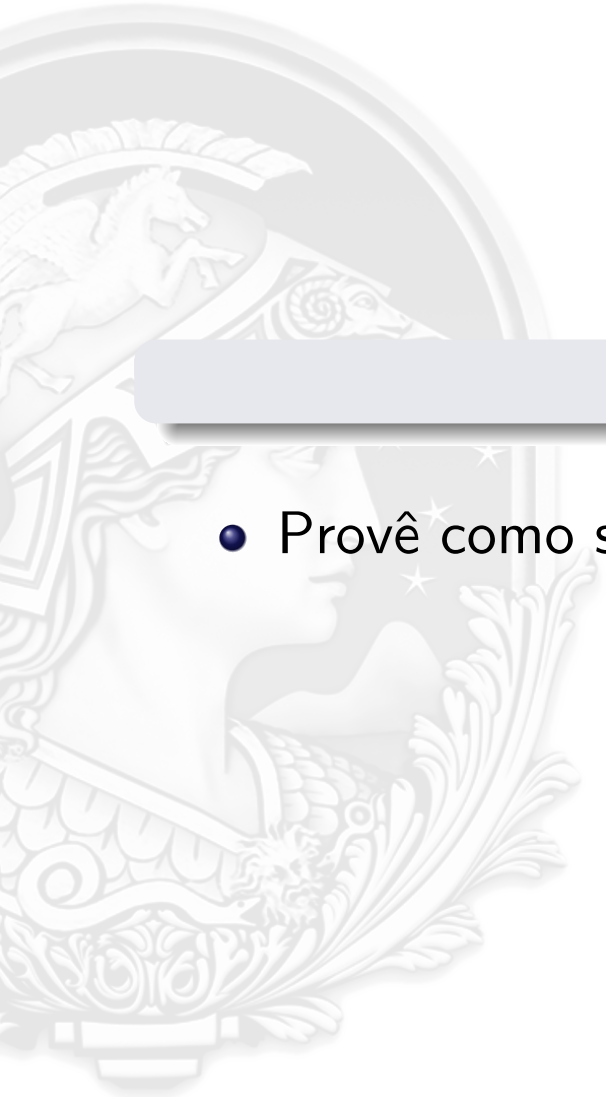
Visita e reescreve a matriz de memória iterativamente

- Natureza pseudo-aleatória *(aumento o TMTO)*
- As colunas também são selecionadas pseudo-aleatoriamente *(diminui o desempenho: GPUs e plat. com cache pequeno)*
- Linhas envolvidas:



- Prioriza **plat. legítimas**, e aumenta o **custo de hardwares dedicados**

The Wrap-up phase



Provê a saída

- Provê como saída uma cadeia de bits de tamanho k *(squeeze)*

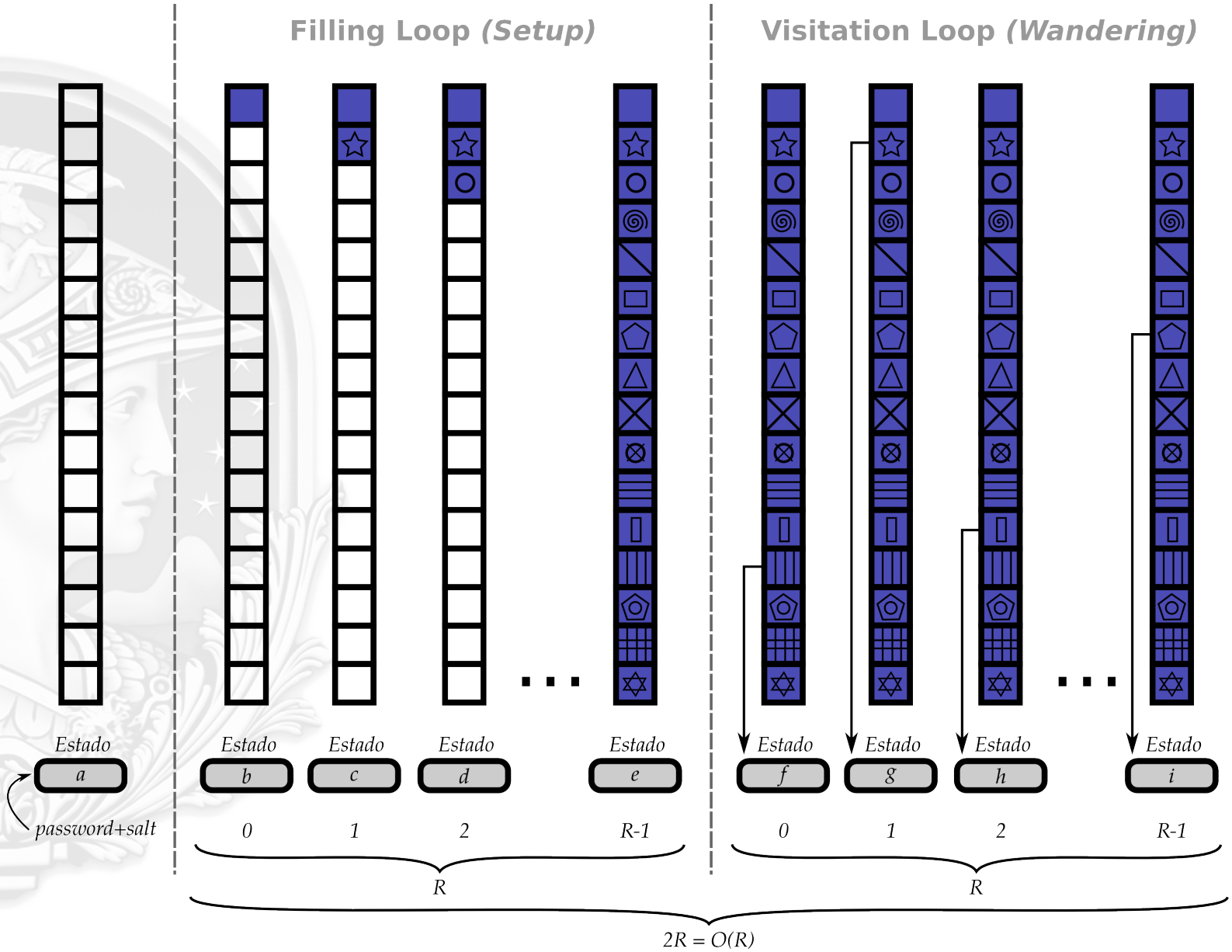
Sumário

1. Introdução
   - Motivação
   - Objetivos
   - Metodologia
2. Lyra2
   - *The Bootstrapping phase*
   - *The Setup phase*
   - *The Wandering phase*
   - *The Wrap-up phase*
3. Lyra2 x scrypt x finalistas do PHC
   - Segurança
   - Desempenho
4. BlaMka
   - Resultados Parciais
5. Considerações Finais
   - Principais Resultados
   - Trabalhos Futuros

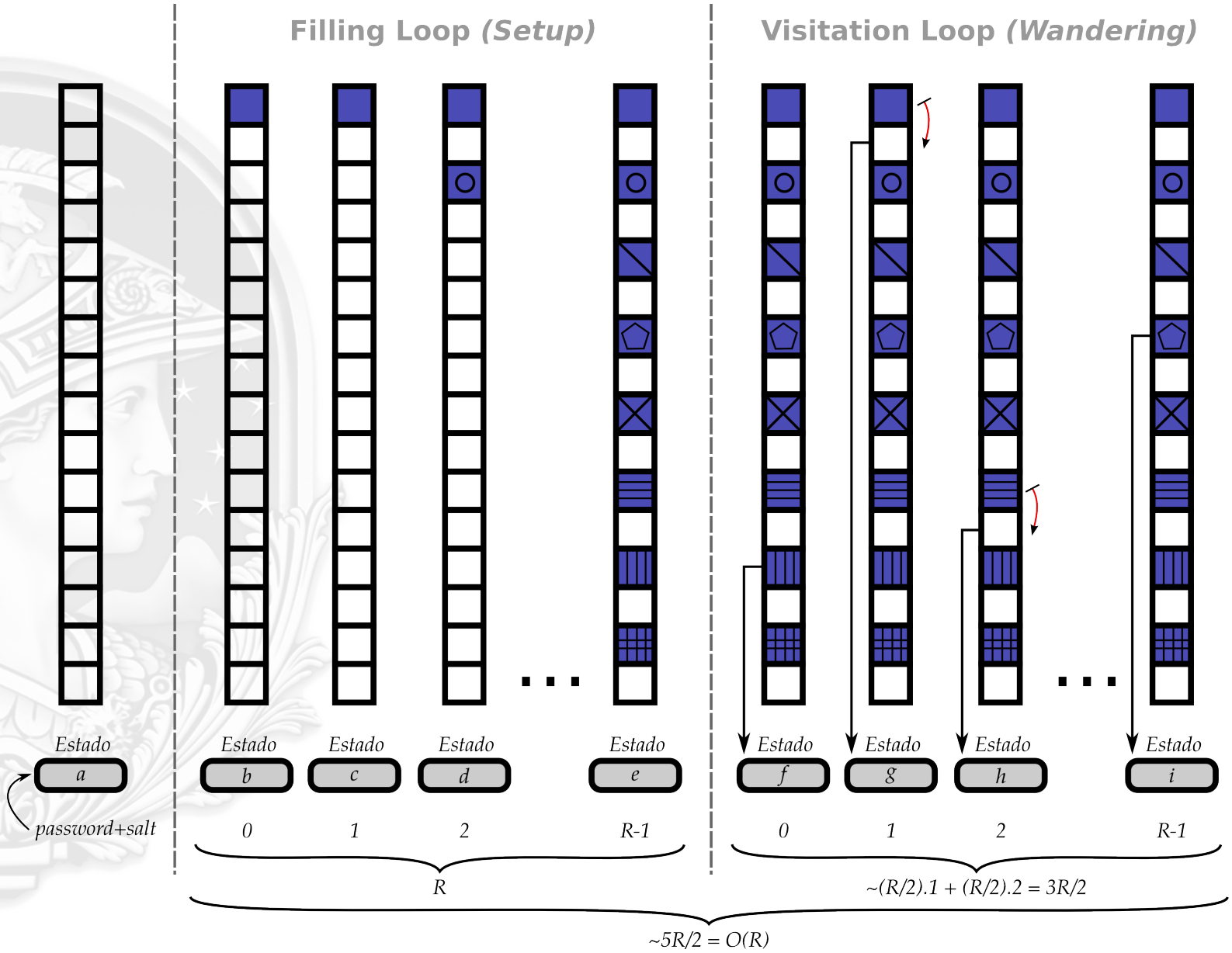
Low-Memory attacks: Estratégias

- *Consumer-producer strategy*
- *Sentinel-based strategy*

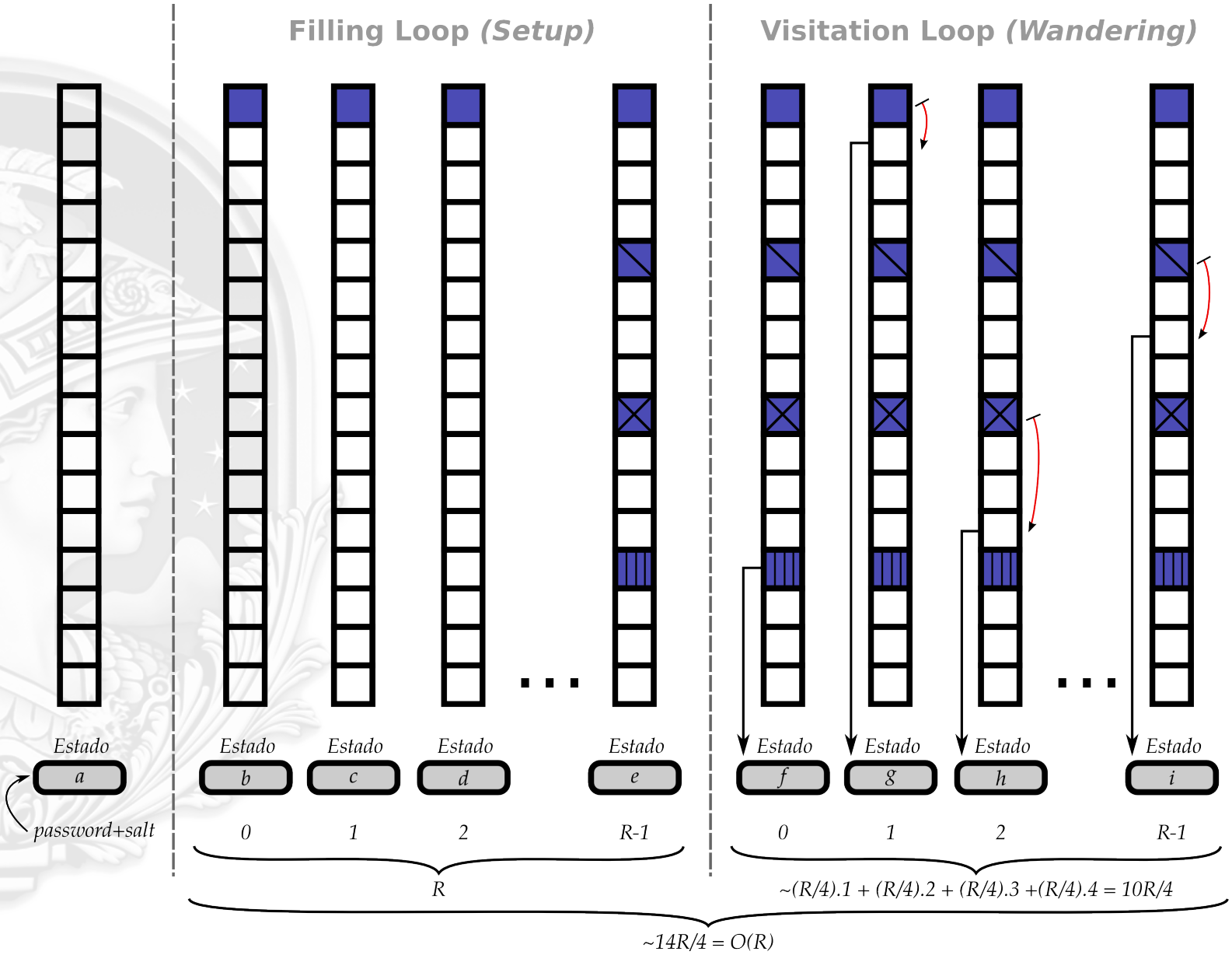
Low-Memory attack: scrypt



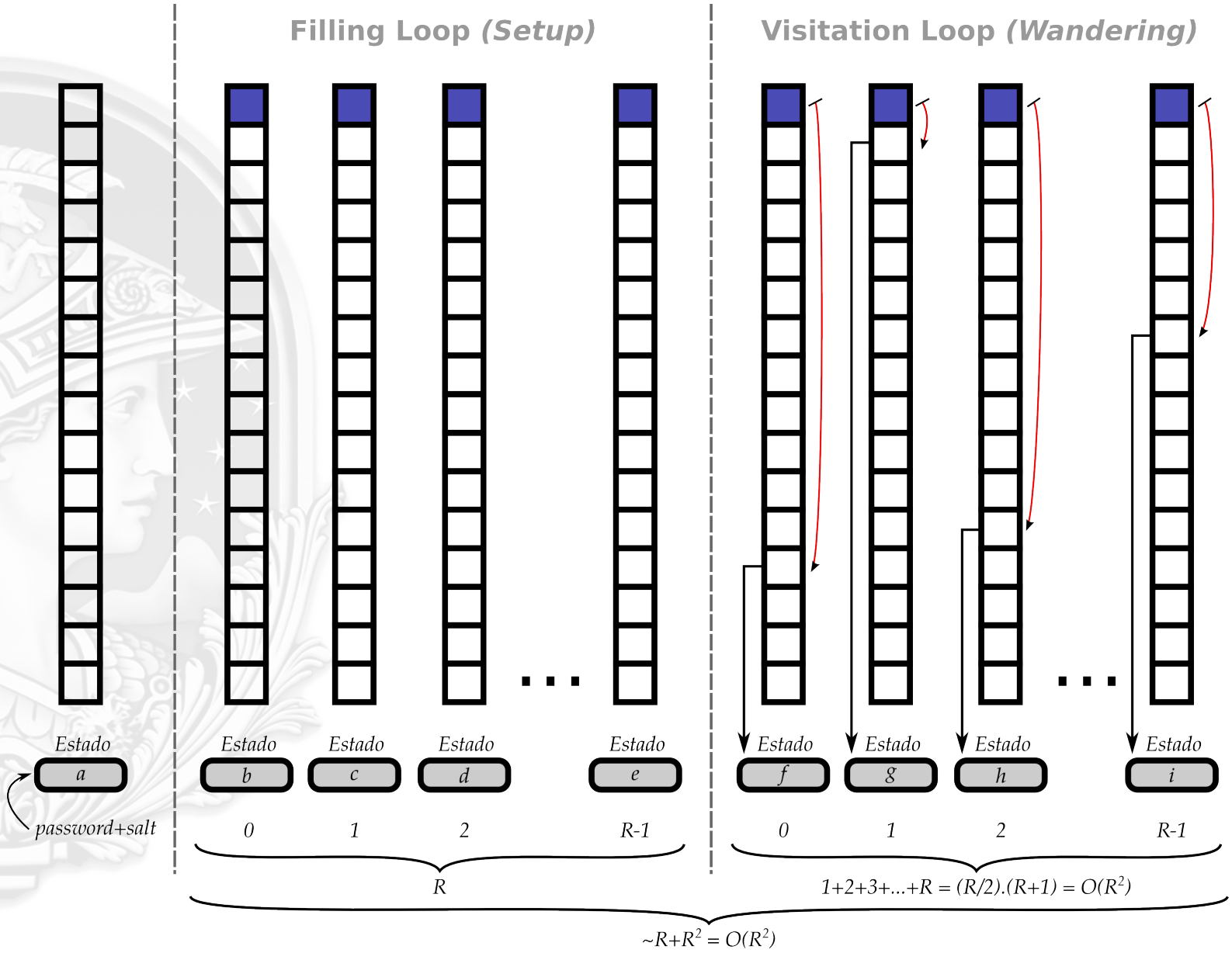
Low-Memory attack: scrypt



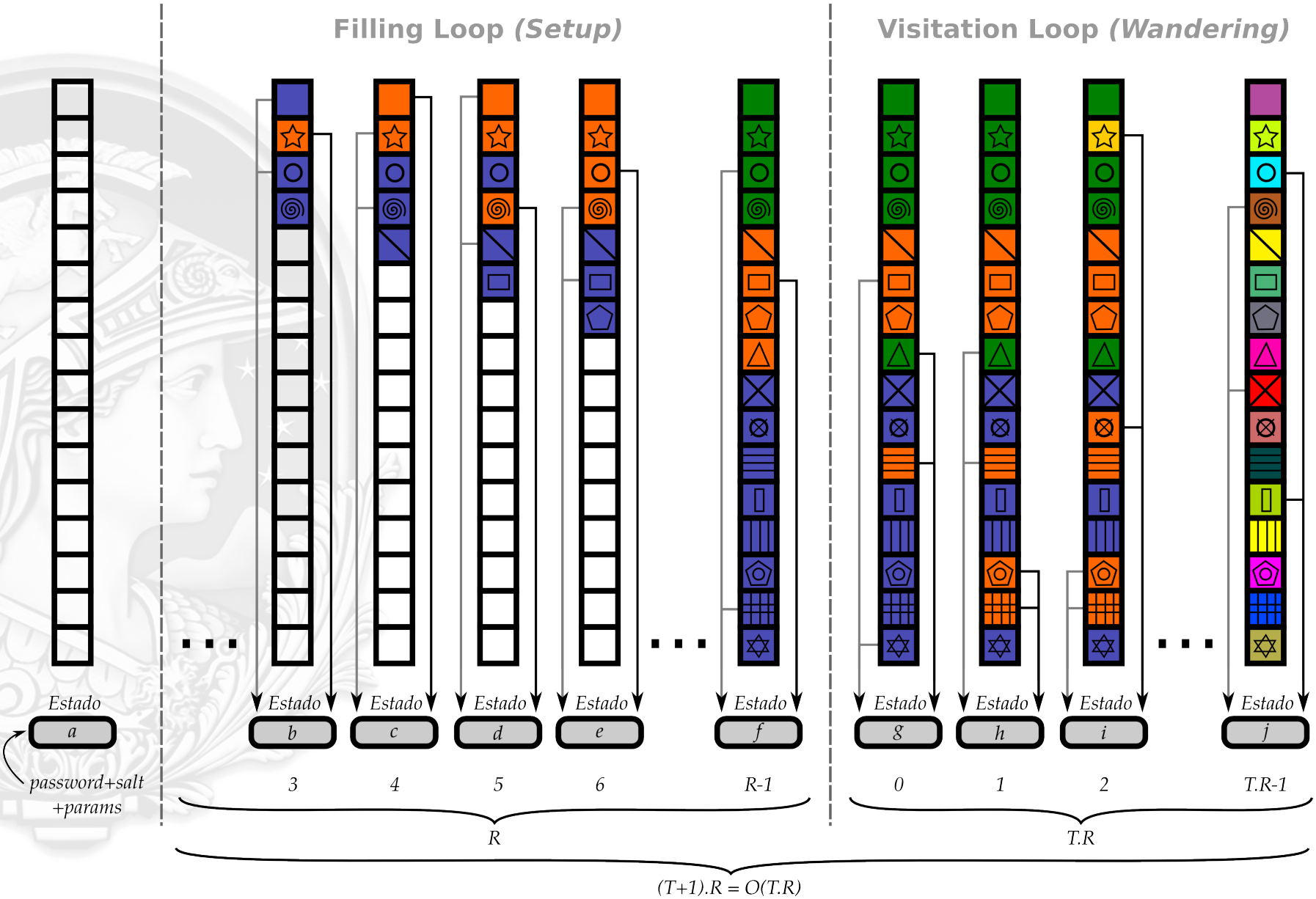
Low-Memory attack: scrypt



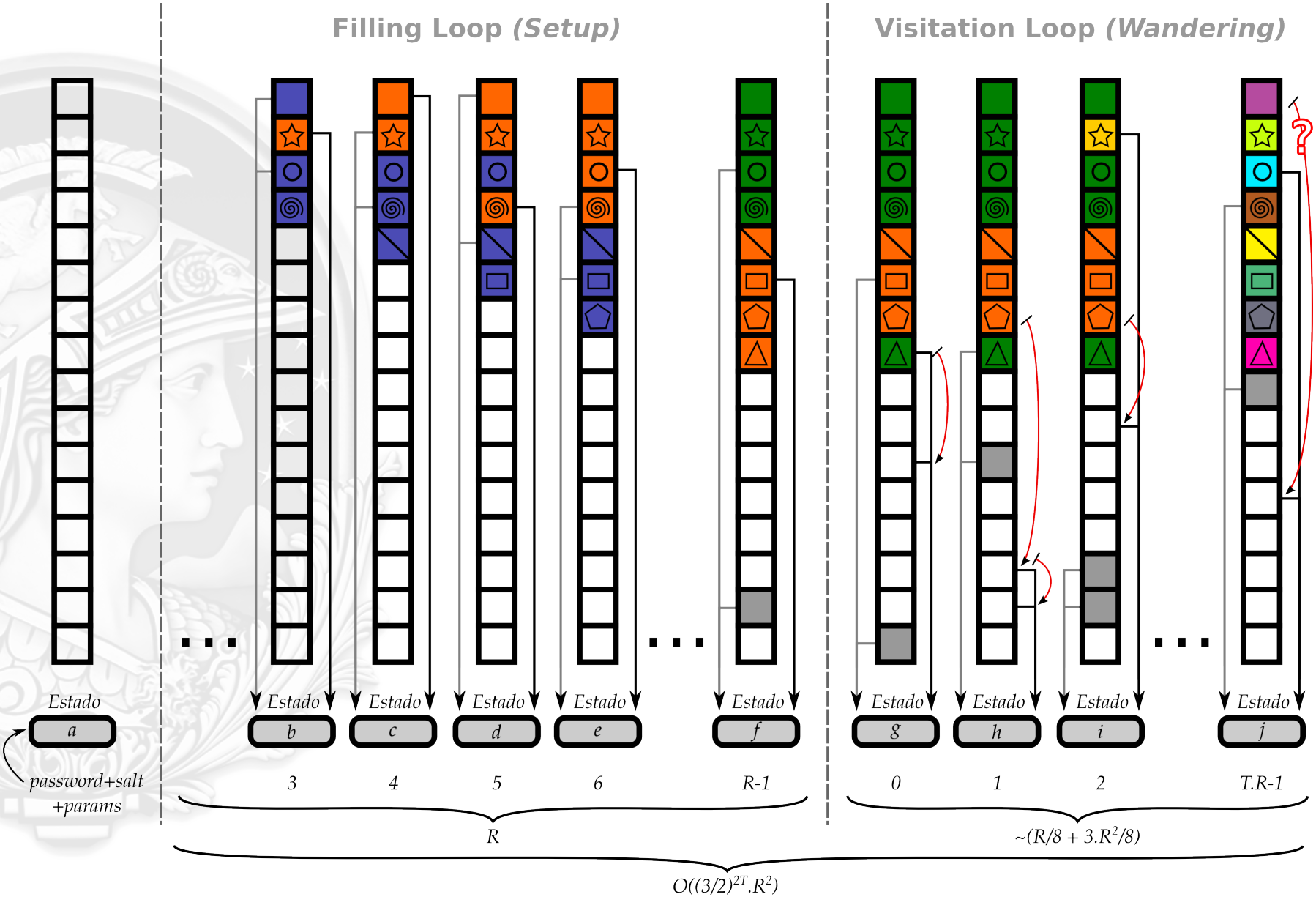
Low-Memory attack: scrypt



Low-Memory attack: Lyra2



Low-Memory attack: Lyra2

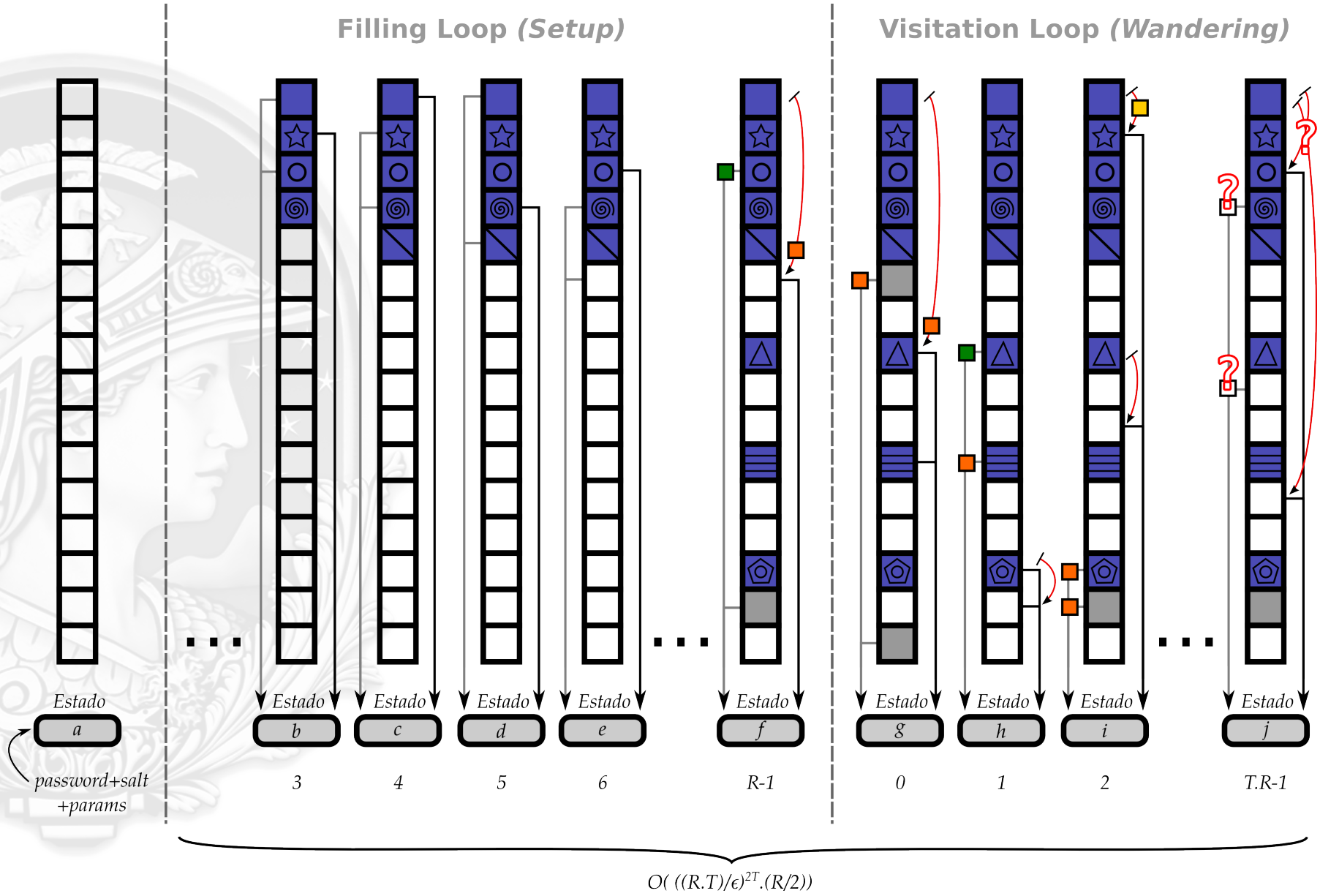


Low-Memory attack: Lyra2

- Quando a memória utilizada pelo atacante for **menor do que a metade** (i.e., $\frac{R}{2^{n+2}}$, onde $n \geq 0$)
- A "árvore de dependência" cresce significativamente, resultando em uma **complexidade aproximada de**:

$$\mathcal{O}(2^{2nT} R^{2+n/2}), \text{para } n \gg 1$$

Low-Memory attack: Lyra2 (*Sentinel-based*)



Low-Memory attack: *Sentinel-based strategy*

- Utiliza uma máquina que possui diversos núcleos que podem acessar a memória simultaneamente sem restrições de banda (irreal)
- **Complexidade** aproximada de:

$$\mathcal{O}\left(((R \cdot T)/\epsilon)^{2T}(R/2)\right)$$

- Dependendo da configuração é **menos vantajosa** do que a estratégia *Consumer-producer*
  - Vantajosa quando o Lyra2 esteja configurado com **parâmetros reduzidos** (e.g., $T = 1$)

Slow-Memory and Cache-timing attacks

Slow-Memory

X

Cache-timing
(side-channel)

Slow-Memory and Cache-timing attacks

Slow-Memory

X

Cache-timing
(side-channel)

PHC special recognition

"abordagem alternativa para resistir a ataques por canal colateral" [PHC15]

Segurança: Visão Geral (finalistas PHC)

Algoritmo	TMTO	SM	SC	Hw/GPUs	GC
Argon2i [BDK16]	for $R' = R/6$ $\approx 2^{15.5} \cdot T \cdot R$	—	✓	✓	✓
Argon2d [BDK16]	for $R' = R/6$ $\approx 2^{19.6} \cdot T \cdot R$	✓	✗	✓	✓
battcrypt [Tho14]	—	✓	✗	✓	✓
Catena [FLW13]	$\mathcal{O}(1)$ $\Theta(R^{T+1})$	—	✓	✓	✓
Lyra2 [nosso]	Para $R' = R/2^{n+2}$, onde $n \geq 0$ $\mathcal{O}(2^{2nT} R^{2+n/2})$, para $n \gg 1$	✓	!	✓	✓
POMELO [Wu15]	—	✓	!	✓	✓
yescrypt [Pes15]	$\mathcal{O}(1)$ $\mathcal{O}(R^{T+1})$	✓	✗	✓	✓*
scrypt [Per09]	$\mathcal{O}(1)$ $\mathcal{O}(R^2)$	✓	✗	!	✗

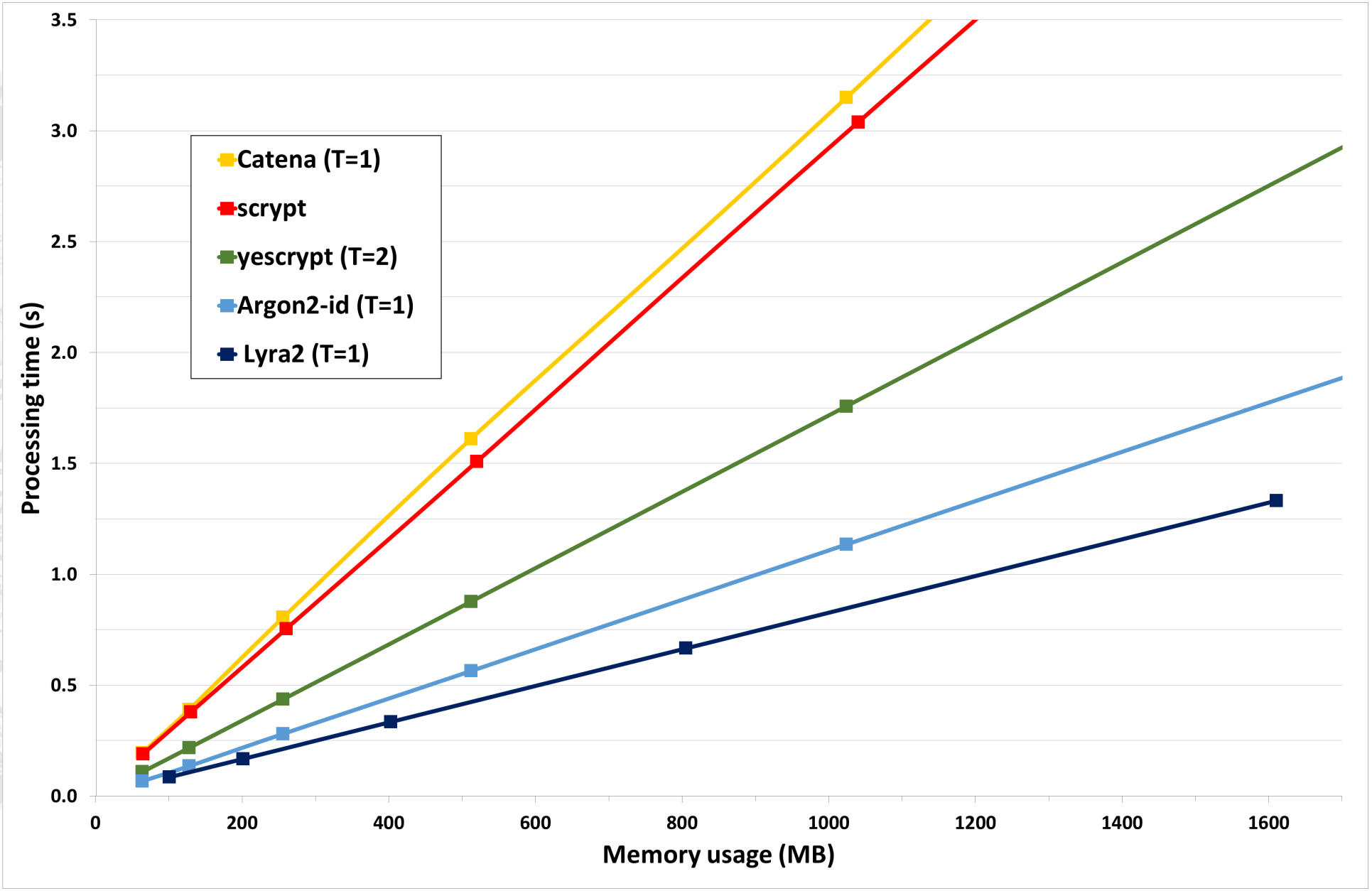
✓- Possui proteção; ✗- Não possui proteção; ! - Proteção parcial; — - Nada declarado.

Desempenho: chamadas à função subjacente

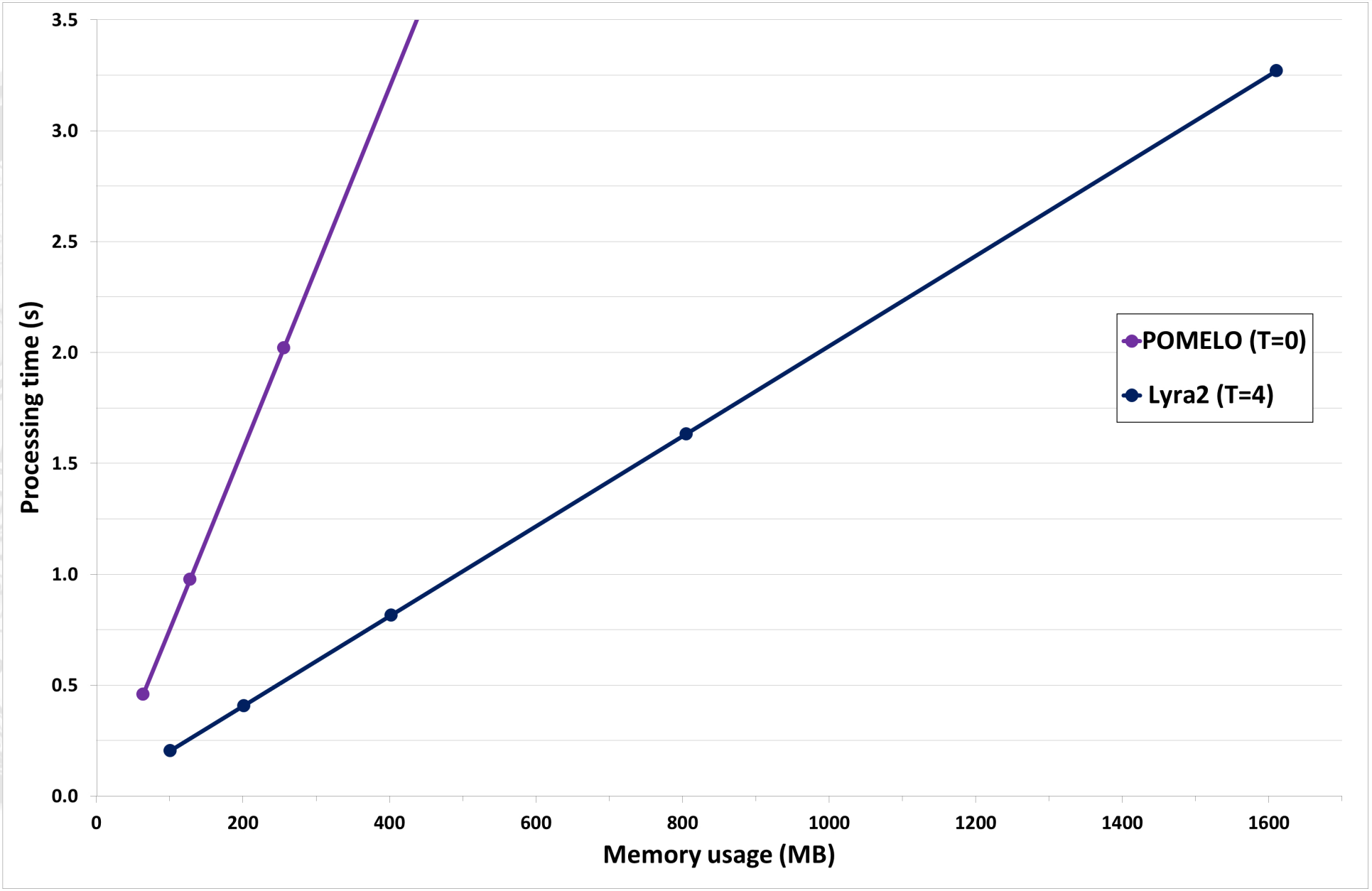
Algoritmo	Chamadas à função subjacente	Instruções SIMD
Argon2 [BDK16]	$2 \cdot T \cdot M$	Sim
battcrypt [Tho14]	$\{2^{\lfloor T/2 \rfloor} \cdot [(T \mod 2) + 2] + 1\} \cdot M$	Não
Catena[1] [FLW13]	$(T + 1) \cdot M$	Sim
Lyra2 [nosso]	$(T + 1) \cdot M$	Sim
POMELO [Wu15]	$\left(3 + 2^{2T}\right) \cdot M$	Não
yescrypt [Pes15]	$T \cdot M$	Sim
scrypt [Per09]	$2 \cdot M$	Sim

[1] No Catena, o número exato de chamadas a função subjacente é definido pela equação $(g - g_0 + 1) \cdot (T + 1) \cdot M$, onde g e g_0 são, respectivamente, o atual e o mínimo *"garlic"*. Contudo, como normalmente $g = g_0$, optamos por utilizar a simplificação desta equação: $(T + 1) \cdot M$.

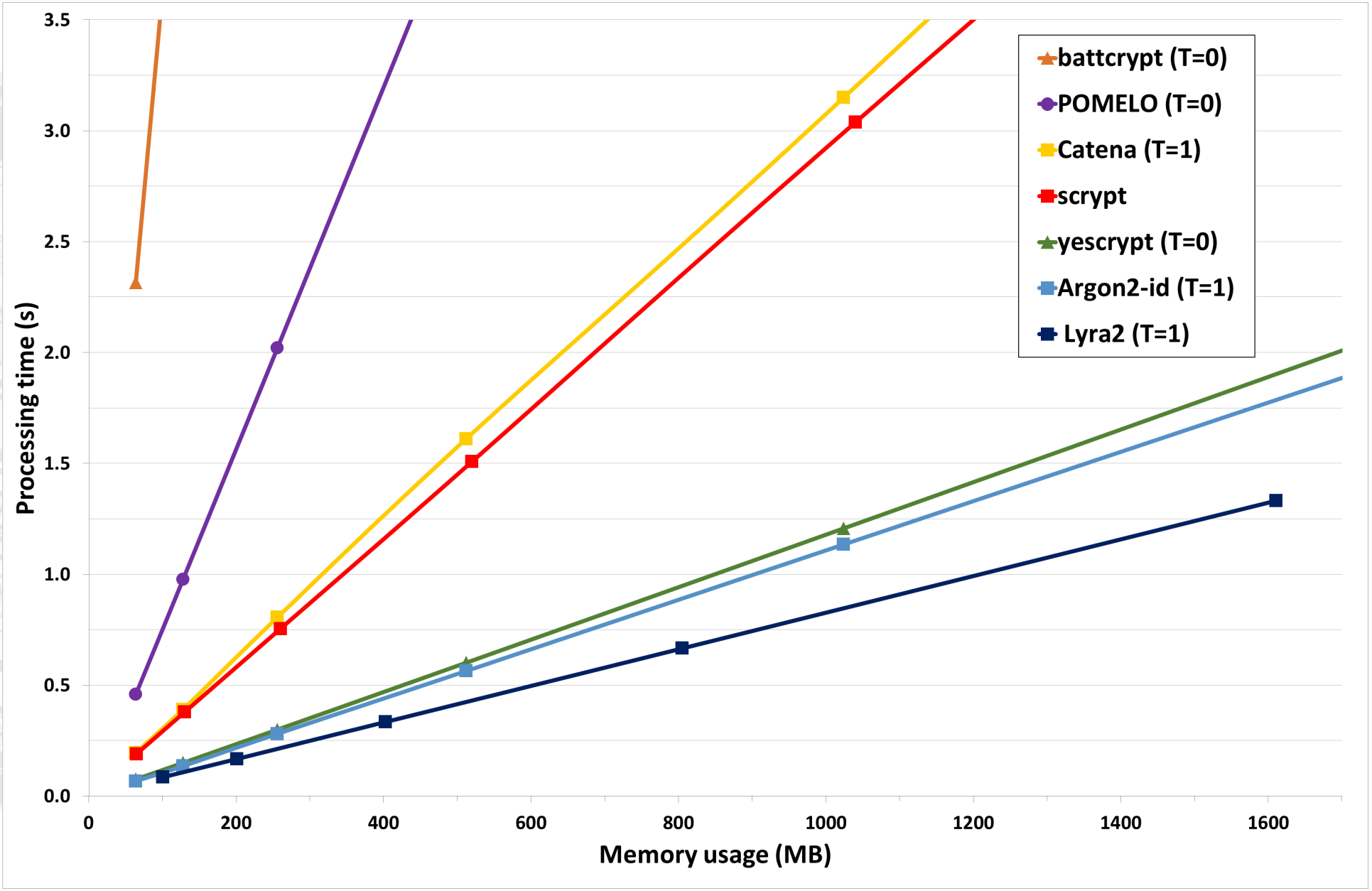
Desempenho *(chamadas comparáveis ao Lyra2 $T = 1$)*



Desempenho *(chamadas comparáveis ao Lyra2 $T = 4$)*



Desempenho *(parâmetros mínimos)*



Hardware dedicado / GPUs

- **Melhoram o desempenho em software / aumentam o custo do hardware**
  - Utilizar uma **quantidade significativa de memória** em plataformas legítimas *(bom desempenho em software)*
  - Tornar o algoritmo ajustável o suficiente para melhor utilizar a **hierarquia de memórias** *(registradores, cache, memória principal, ...)*

Hardware dedicado / GPUs

- **Dificulta o pipelining (TMTO)**
  - **Ler colunas** em ordem crescente e escrever em ordem decrescente *(Setup)*
- **Dificulta o prefetching / GPUs antigas**
  - **Ler colunas** de forma pseudo-aleatória e escrever em ordem crescente *(Wandering)*
- **Aumentam a latência em hardware**
  - Não utilizar somente operações de **XOR** *(adição normal)*
  - Utilizar funções de permutação que **priorizem implementações em software** *(Blake2,...)*
  - Disponibilizar funções de permutação que realizem **operações menos eficientes em hardware** *(BlaMka)*

Sumário

1. Introdução
   - Motivação
   - Objetivos
   - Metodologia
2. Lyra2
   - *The Bootstrapping phase*
   - *The Setup phase*
   - *The Wandering phase*
   - *The Wrap-up phase*
3. Lyra2 x scrypt x finalistas do PHC
   - Segurança
   - Desempenho
4. BlaMka
   - Resultados Parciais
5. Considerações Finais
   - Principais Resultados
   - Trabalhos Futuros

Visão Geral

- Busca por função de permutação que realize **operações menos eficientes em hardware**
- Mantenha a segurança do Blake2 (no mínimo)

$$
\begin{array}{lcl}
a & \leftarrow & a + b \\
d & \leftarrow & (d \oplus a) \ggg 32 \\
c & \leftarrow & c + d \\
b & \leftarrow & (b \oplus c) \ggg 24 \\
a & \leftarrow & a + b \\
d & \leftarrow & (d \oplus a) \ggg 16 \\
c & \leftarrow & c + d \\
b & \leftarrow & (b \oplus c) \ggg 63
\end{array}
$$

(a) Função G Blake2b, baseada em adições.

$$
\begin{array}{lcl}
a & \leftarrow & a + b + 2 \cdot \mathrm{lsw}(a) \cdot \mathrm{lsw}(b) \\
d & \leftarrow & (d \oplus a) \ggg 32 \\
c & \leftarrow & c + d + 2 \cdot \mathrm{lsw}(c) \cdot \mathrm{lsw}(d) \\
b & \leftarrow & (b \oplus c) \ggg 24 \\
a & \leftarrow & a + b + 2 \cdot \mathrm{lsw}(a) \cdot \mathrm{lsw}(b) \\
d & \leftarrow & (d \oplus a) \ggg 16 \\
c & \leftarrow & c + d + 2 \cdot \mathrm{lsw}(c) \cdot \mathrm{lsw}(d) \\
b & \leftarrow & (b \oplus c) \ggg 63
\end{array}
$$

(b) Função $G_{\texttt{tls}}$ BlaMka, baseada em quadrados latinos truncados (`tls`).

Visão Geral

- Busca por função de permutação que realize **operações menos eficientes em hardware**
- Mantenha a segurança do Blake2 (no mínimo)

$$
\begin{array}{lcl}
a & \leftarrow & a + b \\
d & \leftarrow & (d \oplus a) \ggg 32 \\
c & \leftarrow & c + d \\
b & \leftarrow & (b \oplus c) \ggg 24 \\
a & \leftarrow & a + b \\
d & \leftarrow & (d \oplus a) \ggg 16 \\
c & \leftarrow & c + d \\
b & \leftarrow & (b \oplus c) \ggg 63
\end{array}
$$

(a) Função G Blake2b, baseada em adições.

$$
\begin{array}{lcl}
a & \leftarrow & a + b + 2 \cdot \mathrm{lsw}(a) \cdot \mathrm{lsw}(b) \\
d & \leftarrow & (d \oplus a) \ggg 32 \\
c & \leftarrow & c + d + 2 \cdot \mathrm{lsw}(c) \cdot \mathrm{lsw}(d) \\
b & \leftarrow & (b \oplus c) \ggg 24 \\
a & \leftarrow & a + b + 2 \cdot \mathrm{lsw}(a) \cdot \mathrm{lsw}(b) \\
d & \leftarrow & (d \oplus a) \ggg 16 \\
c & \leftarrow & c + d + 2 \cdot \mathrm{lsw}(c) \cdot \mathrm{lsw}(d) \\
b & \leftarrow & (b \oplus c) \ggg 63
\end{array}
$$

(b) Função $G_{\texttt{tls}}$ BlaMka, baseada em quadrados latinos truncados (`tls`).

Adoção

O Argon2 adotou o BlaMka como padrão [BDK16].

Desempenho *(CPU)*

Microarquitetura	Família CPU	Modelo	*G*	Implementação	Throughput (Gbps)	Throughput Ratio
Sandy Bridge	Intel Xeon	E5-2430	Blake2b	Genérica	51.578	1
			BlaMka	Genérica	21.465	0.416

Tabela : Testes realizados em CPU, executando apenas um round da função *G* (i.e., 256 bits).

Desempenho *(Hardware dedicado)*

FPGA (ISE Design Suite)

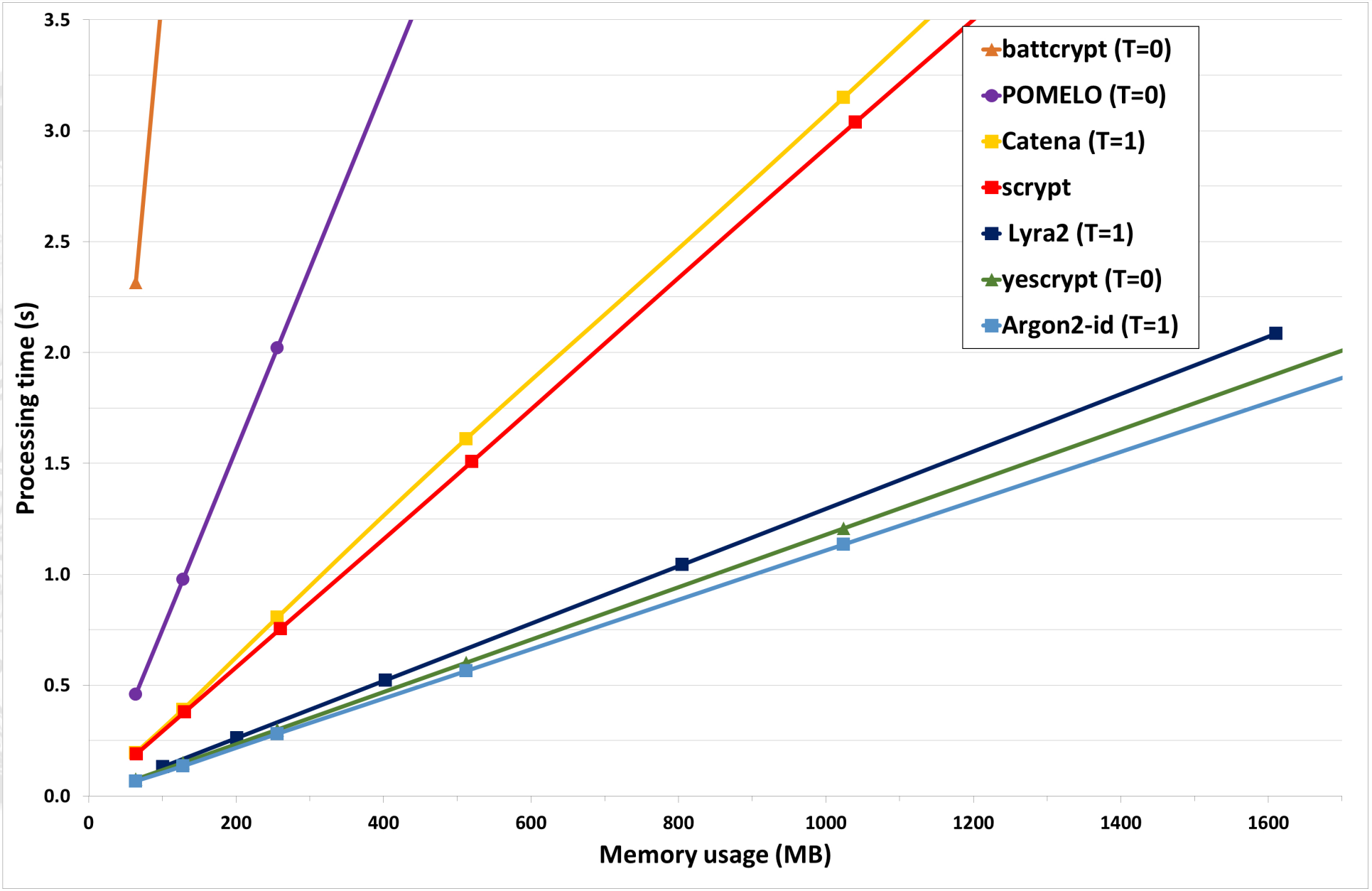
Tecnologia	Família FPGA	G	Implem.	Área		Throughput		Throughput Relativo
				Slices	Ratio	Gbps	Ratio	
90nm	Spartan 3E	Blake2b	Genérica	508	1	1.570	1	1
		BlaMka	Genérica	787	1.549	0.711	0.453	0.292
		BlaMka	Otimizada	3567	7.022	0.741	0.472	0.067

ASIC (Synopsys Design Compiler)

Tecnologia	Família	G	Implem.	Área		Throughput		Throughput Relativo
				μm^2	Ratio	Gbps	Ratio	
90nm	SAED EDK 90nm	Blake2b	Genérica	40191	1	3.837	1	1
		BlaMka	Genérica	107088	2.664	3.433	0.895	0.336
		BlaMka	Otimizada	142911	3.556	3.145	0.820	0.231

Tabela : Testes iniciais realizados em FPGA e ASIC, executando apenas um round da função G (i.e., 256 bits).

Desempenho *(Lyra2 utilizando BlaMka)*



Sumário

1. Introdução
   - Motivação
   - Objetivos
   - Metodologia
2. Lyra2
   - *The Bootstrapping phase*
   - *The Setup phase*
   - *The Wandering phase*
   - *The Wrap-up phase*
3. Lyra2 x scrypt x finalistas do PHC
   - Segurança
   - Desempenho
4. BlaMka
   - Resultados Parciais
5. Considerações Finais
   - Principais Resultados
   - Trabalhos Futuros

Resultados

- Neste trabalho apresentamos um **novo Esquema de Hash de Senhas** que:
  - É melhor do que as soluções pré-PHC
  - Contribuiu significativamente para o amadurecimento das soluções apresentadas no PHC
- **Mantêm a eficiência e a flexibilidade** do seu predecessor, e ainda **aumenta sua segurança** em termos de:
  - TMTO
  - Custos envolvidos na construção de um hardware dedicado
  - Equilíbrio entre ataques de canal colateral e ataque que se baseiam no uso de dispositivos de memória mais barata

Publicações, adoções e demais contribuições *(relacionados)*

PHC special recognition [PHC15]

Adoções

- Vertcoin migrou do scrypt para o Lyra2 [a4314, Day14]
- O Sgminer adicionou o Lyra2 em suas distribuições [Cry15]
- O Argon2 adotou o BlaMka como padrão [BDK16]

Publicações

- Lyra publicado no JCEN [AABS14]
- Lyra2 aceito no IEEE trans. on Computers [ASBS16]
- Resumo do Lyra2 apresentado no 3º WPG-EC, na LatinCrypt'14 e na ICISSP'16 [AS14b, AS14a, AS16]

Prêmio

- Melhor trabalho de doutorado [AS16, ICI16]

Publicações, adoções e demais contribuições *(PhD)*

Participação em projeto de pesquisa

- Segurança em Redes Virtuais para Computação em Nuvem (2013–2015)
- Padrão de defesa em sistema em nuvem (2016–Atual)

Publicações

Capítulo de livro

- BARROS, B. M.; SIMPLICIO JUNIOR, M. A.; CARVALHO, T. C.; ROJAS, M. A. T.; REDIGOLO, F. F.; **ANDRADE, E. R.**; MAGRI, D. R. C.. *Applying Software-defined Networks to Cloud Computing*. In: MARTINELLO, M.; ROBEIRO, M. R. N.; ROCHA, A. A. A.. (Org.). Minicursos do XXXIII SBRC. 1ed. Porto Alegre/RS: SBC, 2015, v. 1, p. 1-54.

Conferências

- ALMEIDA, T. R. M.; **ANDRADE, E. R.**; BARROS, B. M.; SIMPLICIO JUNIOR, M. A.. *Avaliação de Desempenho em Nuvens Computacionais utilizando IPsec em conjunto com SR-IOV*. In: Anais / XXXIV SBrT, Santarém, PA. Porto Alegre: SBC, 2015.
- ATENIESE, G.; MAGRI, B.; VENTURI, D.; **ANDRADE, E. R.**. *Redactable Blockchain – or – Rewriting History in Bitcoin*. In: Proceedings of 23nd ACM CCS, Vienna, Austria. **Submetido**.

Comissão de avaliação

- SIIC-USP 2014, SBSEG'15, FEBRACE'15, FEBRACE'16, EUROCRYPT'16.

Trabalhos Futuros

- Estender os estudos do BlaMka
- Lyra2 e GPUs *(friendly e unfriendly)*
- Lyra2 totalmente resistente a ataques por canal colateral
- O quão próximo ao melhor resultado possível é a escolha de índices da fase de *Setup* do Lyra2

Obrigado!

Referências I

[a4314] a432511. PoW Algorithm Upgrade: Lyra2 – Vertcoin. https://vertcoin.org/pow-algorithm-upgrade-lyra2/. Accessed: 2015-05-06., 2014.

[AABS14] L. C. Almeida, E. R. Andrade, P. S. L. M. Barreto e M. A. Simplicio Jr. Lyra: Password-Based Key Derivation with Tunable Memory and Processing Costs. Journal of Cryptographic Engineering, 4(2):75–89, 2014. See also http://eprint.iacr.org/2014/030.

[AS14a] E. R. Andrade e M. A. Simplicio Jr. Lyra2: a password hashing schemes with tunable memory and processing costs. Third International Conference on Cryptology and Information Security in Latin America, LATINCRYPT'14. Florianópolis, Brazil.http://latincrypt2014.labsec.ufsc.br/., 2014.

[AS14b] E. R. Andrade e M. A. Simplicio Jr. Lyra2: Um Esquema de Hash de Senhas com custos de memória e processamento ajustáveis, October 2014.

[AS16] E. R. Andrade e M. A. Simplicio Jr. Lyra2: Efficient Password Hashing with high security against Time-Memory Trade-Offs. Em Doctoral Consortium – Proceedings of 2nd International Conference on Information Systems Security and Privacy, ICISSP 2016, Rome, Italy, February 2016. Institute for Systems and Technologies of Information, INSTICC. http://www.icissp.org/?y=2016.

[ASBS16] E. R. Andrade, M. A. Simplicio Jr, P. S. L. M. Barreto e P. C. F. dos Santos. Lyra2: efficient password hashing with high security against time-memory trade-offs. IEEE Transactions on Computers, PP(99), 2016. See also http://eprint.iacr.org/2015/136.

[BDK16] A. Biryukov, D. Dinu e D. Khovratovich. Argon2: the memory-hard function for password hashing and other applications. Password Hashing Competition, Luxembourg, v1.3 of argon2 edição, Feb 2016. https://github.com/P-H-C/phc-winner-argon2/blob/master/argon2-specs.pdf.

[BDPA07] G. Bertoni, J. Daemen, M. Peeters e G. Van Assche. Sponge functions. (ECRYPT Hash Function Workshop 2007), 2007. http://sponge.noekeon.org/SpongeFunctions.pdf. Accessed: 2015-06-09.

[Cry15] Crypto Mining. Updated Windows Binary of sgminer 5.1.1 With Fixed Lyra2Re Support – Crypto Mining Blog. http://cryptomining-blog.com/4535-updated-windows-binary-of-sgminer-5-1-1-with-fixed-lyra2re-support/., 2015.

Referências II

[Day14] Timothy Day. Vertcoin (VTC) plans algorithm change to Lyra2 – coinbrief. http://coinbrief.net/vertcoin-algorithm-change-lyra2/. Accessed: 2015-05-06, 2014.

[FH07] D. Florencio e C. Herley. A Large-scale Study of Web Password Habits. Em Proceedings of the 16th International Conference on World Wide Web, páginas 657–666, New York, NY, USA, 2007. ACM.

[FLW13] C. Forler, S. Lucks e J. Wenzel. Catena: A Memory-Consuming Password Scrambler. Cryptology ePrint Archive, Report 2013/525, 2013. http://eprint.iacr.org/2013/525.Accessed:2014-03-03.

[ICI16] ICISSP. Previous awards. International Conference on Information Systems Security and Privacy – website, 2016. http://www.icissp.org/PreviousAwards.aspx.

[Per09] C. Percival. Stronger key derivation via sequential memory-hard functions. Em BSDCan 2009 – The Technical BSD Conference, Ottawa, Canada, 2009. University of Ottawa. See also: http://www.bsdcan.org/2009/schedule/attachments/87_scrypt.pdf. Accessed: 2013-12-09.

[Pes15] A. Peslyak. yescrypt - a Password Hashing Competition submission. Password Hashing Competition, Moscow, Russia, v1 edição, Jan 2015. https://password-hashing.net/submissions/specs/yescrypt-v0.pdf. Accessed: 2015-05-22.

[PHC15] PHC. Password Hashing Competition. https://password-hashing.net/#phc., 2015.

[SO12] D. Song e J. Oberheide. Modern Two-Factor Authentication: Defending Against User-Targeted Attacks. Duo Security, 2012. https://speakerdeck.com/duosec/modern-two-factor-authentication-defending-against-user-targeted-attacks.

[Tho14] S. Thomas. battcrypt (Blowfish All The Things). Password Hashing Competition, Lisle, IL, USA, v0 edição, Mar 2014. https://password-hashing.net/submissions/specs/battcrypt-v0.pdf.

[Wu15] H. Wu. POMELO – A Password Hashing Algorithm (Version 2). Password Hashing Competition, Nanyang Ave, Singapore, v3 edição, Apr 2015. https://password-hashing.net/submissions/specs/POMELO-v3.pdf.

Créditos

- A imagem utilizada como plano de fundo em todos os slides segue a licença de uso que consta em `http://www.poli.usp.br` – © Escola Politécnica da USP.
- A imagem utilizada no slide de Motivação foi retirada de [SO12] e segue a licença de uso © GitHub Inc.
- A imagem utilizada no slide de Motivação *(Cont.)* foi adaptada de imagens dos sites: `http://1aled.fotomaps.ru/` e `http://www.harvestsolutions.net/`, seguindo suas respectivas licenças de uso.
- A imagem utilizada no slide de Metodologia foi retirada do site `http://corneralliance.com/` e segue a licença de uso que consta no respectivo site.
- As imagens utilizadas no slide *Slow-Memory and Cache-timing attacks* foram retiradas dos sites `http://www.toshiba.com/`, `http://www.engadget.com/` e `https://wiki.teamfortress.com/`; e seguem as licenças de uso que constam nos respectivos sites.
- As demais imagens utilizadas ao longo desta apresentação foram confeccionadas pelos autores.