Criptografia Autenticada - Uma breve análise dos concorrentes do CAESAR

Luan Cardoso dos Santos, *Universidade Estadual de Campinas - UNICAMP*

***Resumo*—Este survey busca introduzir brevemente o ramo da criptografia autenticada bem como os conceitos da "lightweight cryptography" ou criptografia de peso leve, apresentando uma breve descrição dos algoritmos participantes do CAESAR, assim como introduzindo os conceitos e características dos cifradores autenticados.**

***Resumo* — Criptografia; Criptografia autenticada; criptografia leve; concurso CAESAR.**

***Abstract*--This survey searches to introduce briefly the study area of authenticated cryptography as well concepts of lightweight cryptography, presenting a short description of the algorithms participating in the CAESAR competition, plus their concepts and characteristics.**

***Index Terms* — Criptography; Authenticated encryption; Lightweight crypto; CAESAR competition.**

I. Introdução

Desde de tempos mais antigos a humanidade tem preocupação com o armazenamento e proteção de dados sensíveis. Na Roma antiga, legiões utilizavam a cifra de César para registrar movimento de tropas e coordenar ataques. Séculos mais tarde os Alemães durante a segunda guerra mundial fizeram uso da máquina criptográfica Enigma para o mesmo propósito, nos dias de hoje a criptografia é uma necessidade básica em um mundo conectado, devido ao advento da internet e o surgimento dos smartphones. Esses dispositivos, cada vez menores e mais pessoais, possuem uma quantidade limitada de recursos energéticos, de processamento, hardware e até mesmo software, o que dificulta a implementação de boa parte dos algoritmos criptográficos modernos.

Devido aos recursos limitados presentes em tais dispositivos e as dificuldades que estas proporcionam, o surgimento de um ramo dedicado a criação de primitivas criptográficas - cifras de bloco, fluxo, funções hash, criptografia de chave pública e assinatura - mais ”leves”era inevitável, sendo essas relativamente mais simples, tendo como foco um bom desempenho em sistemas com recursos limitados e que ainda sejam capazes de proteger as informações, garantido o nível de segurança necessário. Esse novo ramo da criptografia é chamado de lightweight cryptography. Sendo esses conceitos relativos de simplicidade ou leveza, por exemplo, com relação a hardware, um algoritmo seria considerado leve caso possua uma baixa latência, um baixo consumo energético, baixo uso de memória de execução e de armazenamento, entre outros. Com relação a software, um algoritmo é dito leve quando tem uma baixa latência, um baixo consumo energético e um bom desempenho com uma baixa quantidade de memória RAM disponível.

Cifradores autenticados possuem a característica de combinar em um único processo confidencialidade e integridade. Eles funcionam por meio da geração de uma tag de autenticação, que, combinada com o ciphertext e com dados adicionais, funciona de forma análoga a uma função de hash. Dessa forma, caso algum bit seja trocado na mensagem, ou nos dados associados, o processo de decifração é capaz de identificar essa modificação. Além de proteger a integridade e a confidencialidade de uma mensagem, criptografia autenticada provê segurança contra chosen plaintext attack. Nesse ataque, um adversário tenta extrair informações sobre a chave por meio de ciphertexts com características especiais, que são submetidos a um ”oraculo de decifração”, e analisando os resultados da decifração deste. Um cifrador autenticado não permite que um atacante utilize essa tática, a não ser que ele seja capaz de gerar uma tag de autenticação válida para o ciphertext que ele deseja utilizar.

Atualmente, a maioria dos cifradores autenticados são feitos por meio da combinação de um cifrador de bloco e um algoritmo de MAC[1] em um modo de operação autenticado (essa construção também é chamada de construção padrão). Exemplos de modos de operação autenticados são o GCM e o CCM, ambos padronizados e especificados por entidades como o NIST[2] e ENISA[3] .

II. Concursos criptográficos

Em 1997, os Estados Unidos, por meio do NIST, anunciaram uma competição aberta para a escolha do padrão avançado de critpografia. Essa competição reuniu 15 cifras de bloco, projetadas por 50 criptólogos de vários lugares do mundo. O resultado foi o AES, cifrador de bloco utilizado em vários locais nos dias de hoje[8]. A vantagem de um concurso aberto para a escolha de um algoritmo criptográfico é que a comunidade acadêmica pode analisar de forma independente o algoritmo, o que mitiga a possibilidade de haver backdoors no design da cifra, como foi o caso do DES. Hoje, o AES é um dos algoritmos de criptografia mais pesquisados no mundo, trazendo um amplo entendimento do design de cifradores de bloco para a comunidade criptográfica.

Devido ao sucesso do concurso AES, outros concursos foram feitos para a escolha de outros algoritmos para padronização: Em 2004, a ECRYPT[4] anunciou o eSTREAM, um concurso criptográfico seguindo os mesmos moldes do

---

[1] *Message authentication code* - Do inglês, código de autenticação de mensagem, é equivalente a uma função de hash calculada sobre um determinado dado, com a diferença de que normalmente possui como entradas o dado a ser processado e uma chave simétrica. Possui as mesmas propriedades de segurança que uma função de hash criptográfica

[2] *National Instute of Standarts and Technology* - Orgão norte americano responsável por padronizações e referencias técnicas

[3] European Network and Information Security Agency - Orgão Europeu responsável por padronização de segurança e padronizações.

[4] Entidade de excelência fundada pela União Europeia

concurso AES. Ao final do concurso, o comité do eSTREAM publicou um portfólio de cifradores de fluxo. Posteriormente, em 2007, o NIST anunciou uma competição para escolher o sucessor do SHA-2, SHA-3. Como resultado, o algoritmo Keccak[2] foi escolhido para ser o SHA-3, o que abriu portas e contribuiu para aumentar a popularidade de construções de esponja para algoritmos criptográficos.

Atualmente, o concurso CAESAR[3] –Competition for Authenticated Encryption: Security, Applicability, and Robustness – procura produzir um portfólio de cifradores autenticados, assim como o eSTREAM. O CAESAR se encontra na segunda fase, com 29 cifradores autenticados participando da competição. A previsão é que os algoritmos vencedores sejam escolhidos dentre estes até dezembro de 2017.

III. USOS DE CRIPTOGRAFIA AUTENTICADA

Um cifrador autenticado provê ambos confidencialidade e integridade a uma mensagem, utilizando uma única chave e uma única interface entre o cifrador e programa. Em adição a proteção da confidêncialidade e da integridade da mensagem, um cifrador autenticado é capaz de mitigar ataques do tipo Chosen Plaintext. Isso se deve ao fato de que, em um ataque desse tipo, o atacante se utiliza de ciphertexts especialmente escolhidos para tentar extrair alguma informação sobre a chave do texto decifrado. Em uma cifra autenticada, idealmente, o atacante não é capaz de gerar um MAC para um ciphertext sem conhecer o plaintext e chave. Com isso, o atacante não pode executar queries em um oráculo de decifração.

Com relação ao uso de cifradores autenticados, o primeio local onde as caracteristicas de autenti car dados cifrados foram em transações bancarias: Não apenas é necessário garantir a privacidade de uma mensagem, mas também garantir sua autenticidade, para que um atacante ativo no canal de comunicação não seja capaz de alterar o ciphertext para resultar em mudanças no plaintext. Além disso, um cifrador autenticado provê uma solução mais elegante e robusta do que simplesmente adicionar redundância no plaintext.

IV. O CONCURSO CAESAR

O concurso CAESAR é um similar a competição promovida pelo NIST em 1997, que teve como consequência a padronização do AES. O CAESAR tem como objetivo selecionar algoritmos de cifração autenticada que possuam vantagens quando comparados com o AES-GCM e que possam ser adotados em masssa, é esperado que o concurso CAESAR traga o mesmo tipo de impacto na área de cifradores autenticados que as competições passadas similares.

A. As construções dos algoritmos do CAESAR

O Concurso CAESAR define como cifra autenticada sendo uma função que recebe como argumentos cinco strings de bytes, e produz como saída uma string de bytes, sendo essas entradas plaintext e associated data com comprimento variável, secret message number, public message number e key, que possuem comprimento definido, sendo que deve ser possível recuperar o plaintext e o secret message number utilizando os outros dados, sendo, ciphertext, associated data, public message number e key.Cifradores que possuem entradas e saídas com formatos diferentes, devem especificar relações dentre os objetos e as strings de bytes.

Cifradores de que participam do concurso CAESAR podem especificar um limite de comprimento para o ciphertext e associated data, contando que esse limite não seja menos que 216 bytes, sendo que tais cifradores não precisam necessariamente suportar public message numbers ou secret message number (definir o valor do comprimento de entrada como sendo 0 bits). No entanto é esperado que os cifradores mantenham as características de segurança não importando a escolha de message number ou de seus comprimentos, sendo que também não podem requerer que message numbers seja escolhido pelo usuário de acordo com qualquer regra. Esses cifradores podem perder todas as suas características de segurança caso um mesmo message number seja reutilizado, sendo que esse comportamento seja documentado.

É permitido que informações referentes ao comprimento do plaintext sejam recuperadas a partir do comprimento do ciphertext, contudo, nenhuma outra informação de importância sobre o plaintext e as entradas utilizadas podem ser obtidas a partir do ciphertext, o concurso CAESAR recomenda aos competidores que suportem/utilizem os tamanhos padrões de chaves (80, 128 e 256 bits), mensagem pública (92 ou 104 bits) e tags de autenticação (32, 64, 96, 128 ou 168 bits). Os cifradores não precisam necessariamente oferecer suporte ou limitarem-se a esses parâmetros sugeridos.

As principais construções encontradas dentre os algoritmos do segundo round do CAESAR são:

Cifrador de bloco — São baseados em permutações de n bits com chave na forma:

$$E : \{0,1\}k \times \{0,1\}n \rightarrow \{0,1\}n$$

AES-like — Vários dentre os cifradores submetidos ao CAESAR utilizam construções baseadas no AES ou em primitivas do AES. Tal fato se deve ao fato de que o AES tem resistido durante a sua vida útil a varias tentativas de quebra, e é um dos mais bem estudados algoritmos de bloco. Muitos dos cifradores do CAESAR também utilizam o AES como o cifrador de bloco base para um modo de operação.

Cifras de fluxo — Um cifrador de fluxo é um gerador de bits-pseudoaleatórios simétrico, que recebe como parâmetro uma chave e retorna um fluxo de bits de comprimento igual ao da mensagem a ser cifrada. Acorn e TriviA-ck são exemplos de algoritmos do CAESAR que utilizam cifradores de fluxo como base em sua construção.

Permutações sem chave — O exemplo mais conhecido de keyless permutation são as funções de esponja, que "absorvem"dados para produzir um fluxo de dados cifrados. Esse tipo de construção tem se tornado popular dentre os trabalhos de criptografia desde a escolha do Keccak como algoritmo vencedor de SHA-3

Funções de Hash — Funções de hash ou funções de resumo, mapeiam uma mensagem de comprimento arbitrário para uma saída de comprimento definido. Em uma função de hash criptográfica, é computacionalmente difícil de se calcular a função inversa, ou seja, a partir de um valor de hash, computar uma mensagem que resulte nesse hash.

B. Os candidatos do segundo round

A seguir serão apresentados os atuais competidores do CAESAR, com uma breve descrição de cada cifrador. Uma lista dos algoritmos e das primitivas está disponível na tabela I.

Tabela I. Cifras do round 2 do CAESAR. Fonte: https://aezoo.compute.dtu.dk/doku.php

Nome	Tipo	Primitiva	ED paralela	Online	Inverse-free	Prova de Segurança	Nonce-MR
ACORN	SC	LFSR	+/+	+	+	-	NONE
AEGIS	BC	AES	+/-	+	+	-	NONE
AES-COPA	BC	AES	+/+	+	+	+	
AES-JAMBU	BC	AES	-/-	+	+	-	
AES-OTR	BC	AES	+/+	+	+	+	NONE
AEZ	BC	AES, AES	+/+	-	+	+	OFF-MAX
Ascon	Sponge	SPN	-/-	+	+	+	
CLOC	BC	AES,TWINE	-/-	+	+	+	NONE
Deoxys	BC	AES	+/+	+	-	+	
ELmD	BC	AES,AES	+/+	+	-	+	
HS1-SIV	SC	ChaCha/Poly1305	-/-	-	+	+	OFF-MAX
ICEPOLE	Sponge	Keccak-like	+/-	+	+	+	
Joltik	BC	AES	+/+	+	-	+	
Ketje	Sponge	Keccak-f	-/-	+	+	+	NONE
Keyak	Sponge	Keccak-f	+/-	+	+	+	NONE
Minalpher	P	SPN	+/+	+		+	
MORUS	SC	LRX	-/-	+	+	-	NONE
NORX	Sponge	LRX	+/+	+	+	+	NONE
OCB	BC	AES	+/+	+	-	+	NONE
OMD	CF	SHA2	-/-	+	+	+	NONE
PAEQ	P	AESQ	+/+	+	+	+	
π-Cipher	Sponge	ARX	+/+	+	+	-	NONE
PRIMATEs	Sponge	SPN	-/-	+	+	+	
SCREAM	BC	SPN	+/+	+	+	-	NONE
SHELL	BC	AES, AES	+/+	+	-	+	
SILC	BC	AES, PRESENT, LED	-/+	+	+	+	NONE
STRIBOB	Sponge	Streebog	-/-	+	+	+	NONE
Tiaoxin	BC	AES	+/+	+	+	-	NONE
TriviA-ck	SC	Trivium	+/+	-	+	+	NONE

- **ACORN** — Acorn é um cifrador autenticado que opera de forma *bitwise*, capaz de executar o processamento do dado em paralelo. Foi projetado para ser mais rápido que o AES-GCM. Uma característica importante do ACORN é que não necessita de informação sobre o comprimento do dado para operar.
- **AEGIS** — AEGIS utiliza como base de sua construção a função de round do AES, com aproximadamente metade do custo computacional do AES em modo CBC.
- **AES-COPA** — AES-COPA é projetada para manter segurança mesmo em um cenário de reutilização de *nonces*. Existe um ataque de falsificação de *ciphertext* com complexidade de 2^{62} [7].
- **AES-JAMBU** — Jambu é um modo de operação lightweight, que, quando usado com o AES, produz o cifrador autenticado AES-JAMBU. A característica de ser lightweight se deve ao cifrador utilizar apenas um registrador extra de n-bits para um bloco de $2n$-bits.
- **AES-OTR** — OTR –*Offset Two-Round*– é um modo de operação proposto por Minematsu para criptografia autenticada. Uma importante característica desse cifrador é que o processo de cifração e decifração podem ser feitos com a mesma primitiva de cifração do AES, e para uma mensagem de tamanho m a ser autenticada, é necessário apenas m+2 chamadas a primitiva de cifração.
- **AEZ** — AEZ opera por meio da concatenação de um bloco de autenticação no *plaintext* e então cifrando o resultado. AEX é paralelizável e tem complexidade próximo ao do AES-CTR.
- **Ascon** — Ascon é uma família de cifradores autenticados, online e *lightweight*. Possui um ataque de recuperação de chave com complexidade de 2^{35}[4].
- **CLOC** — Uma variação compacta e com baixo overhead do CFB. O design do CLOC tem como objetivo atingir segurança provável, e é capaz de manipular dados de pequeno comprimento, sendo ideal para processadores embarcados.
- **Deoxys** — É um modo de operação baseado na cifra de bloco Deoxys-BC, que por sua vez é uma cifra tweakable baseada no AES.
- **ELmD** — Sigla para *Encrypt-Linear mix-Decrypt*, é um modo de operação resistente a mal uso, totalmente paralelizável e resistente a adversarios adaptativos.
- **HS1-SIV** — HS1 é um cifrador autenticado projetado para ter altas velocidades em implementações de software rodando em processadores de 32 bits com instruções vetoriais, tais como Intel SSE e ARM Neon.
- **ICEPOLE** — É um cifrador autenticado que possui como base uma permutação sem chave. Possui um ataque[5] com complexidade de 2^{46}.
- **Joltik** — Possui construção similar a Deoxys, sendo baseada na cifra de bloco Joltik-BC, essa por sua vez uma cifra do framework TWEAKEY baseada no AES. Joltik, por sua vez, se baseia no modos de operação OCB e COPA
- **Ketje e Keyak** — Ketje e Keyak são dois cifradores similares, que se baseia na construção MonkeyDuplex e na função de permutação do algoritmo de hash Keccak.
- **Minalpher** — MInalper é um modo de operação, com um cifrador de bloco interno baseado na construção Tweakable Even-Mansour.
- **MORUS** — Morus é uma familia de cifradores autenticados com estado interno de 640 e 1280 bits, projetado para ser rapido em hardware, já que apenas shifts, AND e XOR são utilizados em sua construção. Segundo os autores, é capaz de atingir velocidades de 0.69 ciclos por byte em um processador Intel i7 4770 Haswell.
- **NORX** — Norx é um cifrador autencidado, com suporte a paralelismo arbitrário e baseado em primitivas ARX. É baseado na construção Monkey duplex e utiliza como nucleo uma esponja com permutação baseada no ChaCha.

[5] Esse ataque ainda não foi publicado, tendo sido escrito na lista de e-mail do CAESAR

- **OCB** — OCB é um modo de operação projetado para ser tão rapido quando o modo CTR, sendo paralelo e possuindo overhead de minimo para a autenticação.
- **OMD** — Offset Merkle-Damngard é um cifrador autenticado baseado em algoritmos de hash, capaz de tirar proveito de aceleradores utilizado nos algoritmos SHA-1 e SHA-256
- **PAEQ** — PAEQ –*Parallelizable Authenticated Encryption based on Quadrupled AES*– é um cifrador autenticado com design baseado em uma permutação simples, paralelizavel e com tamanho variavel de chave, até 160 bits de segurança.
- **π-Cipher** — π-Cipher é um cifrador autenticado do tipo *Encrypt then MAC*, paralelizavel e baseado em esponja. Possui um ataque de pré imagem com complexidades de 2^{11} para dados e memoria, e de 2^{22} para tempo [6].
- **POET** — POET –*pipelinable on-line encryption with authentication tag* é uma familia de cifradores autenticados robustos e flexiveis. Possui um ataque de falsificação com apenas uma query ao oraculo [5]
- **PRIMATEs** — Primates é uma familia de cifrados autenticados, projetados para serem executados em hardwares limitados. Seguem a construção SPN de algoritmos como o Rijndael.
- **SCREAM** — Scream é uma familia de algoritmos baseado na construção TAE de Liskov et al. iSCREAM, um dos algoritmos da familia, não passou para a segunda fase do CAESAR. Ataques de falsificação são descritos em [10].
- **SHELL** — Shell é um cifrador autenticado principalmente projetado para implementações em software. Shell é similar ao modo de operação GCM.
- **SILC** — SILC é construida dobre o CLOC, que por sua vez é baseado em CFB. SILC tem como objetico melhorar a optmização em hardware da CLOC.
- **STRIBOB** — É um cifrador paseado em permutações de esponja, baseado no AES e no padrão Russo de hash, Streebog
- **Tiaoxin** — Tiaoxin pe um cifrador autenticado baseado em nonce e orientada a software. Aceita mensagens longs, de comprimento máximo de $2^{128} - 1$ bits.
- **TriviA-ck** — TriviA-ck é um cifrador autenticado baseado no cifrador de fluxo Trivia-SC, e utiliza aritmética em campos binários. Possui um ataque de distinguibilidade de chave relacionada, com duas *queries*, devido a um problema na regra de padding[6] .

[6] Ataque publicado na lista de discussão do CAESAR apenas: https://groups.google.com/forum/\#!topic/crypto-competitions/Uzgt-2t3knM

V. AGRADECIMENTOS

Os autores gostariam de agradecer ao Professor Doutor Leonardo Botega pelo convite para submeter um artigo ao Journal, e ao Professor Doutor Fábio Pereira, pela orientação durante a graduação.

VI. CONCLUSÃO

A proposta do concurso CAESAR é de escolher e padronizar algoritmos de criptografia autenticada. Assim como os concursos SHA-3, AES, e eSTREAM, o CAESAR trará um maior entendimento sobre criptografia autenticada, assim como, finalmente, irá prover primitivas criptográficas que adicionarão mais uma faceta à segurança. Espera-se ainda que os resultados de criptoanalise dos cifradores submetidos crie um *know how* que possa ser aplicado a outros algoritmos. Por fim, a criptografia autenticada, principalmente aplicada à dispositivos limitados, se mostra uma necessidade nos dispositivos da Internet das Coisas, que precisam tanto das características de confidencialidade de dados, quanto da autenticidade destes.

REFERÊNCIAS

[1] Farzaneh Abed, Christian Forler, and Stefan Lucks. General overview of the first-round caesar candidates for authenticated ecryption. Technical report, Cryptology ePrint report 2014/792, 2014.

[2] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Keccak sponge function family main document. *Submission to NIST (Round 2)*, 3:30, 2009.

[3] CAESAR committee. Competition for authenticated encryption: Security, applicability, and robustness. http://competitions.cr.yp.to, April 2013.

[4] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Cryptanalysis of ascon. Cryptology ePrint Archive, Report 2015/030, 2015. http://eprint.iacr.org/.

[5] Jian Guo, Jérémy Jean, Thomas Peyrin, and Wang Lei. Breaking poet authentication with a single query. Cryptology ePrint Archive, Report 2014/197, 2014. http://eprint.iacr.org/.

[6] Gaëtan Leurent. Tag Second-preimage Attack against π-cipher. March 2014.

[7] Jiqiang Lu. On the security of the copa and marble authenticated encryption algorithms against (almost) universal forgery attack. Technical report, Cryptology ePrint Archive, Report 2015/079, 2015. http://eprint. iacr. org.

[8] National Institute of Standards and Technology. Aes: the advanced encryption standard. http://competitions.cr.yp.to/aes.html, January 2014.

[9] Phillip Rogaway. Evaluation of some blockcipher modes of operation. *Cryptography Research and Evaluation Committees (CRYPTREC) for the Government of Japan*, 2011.

[10] Siang Meng Sim and Lei Wang. Practical forgery attacks on scream and iscream. In *Posted on the crypto competitions mailing list at https://groups. google. com/d/forum/crypto-competitions*.