Universidade da Beira Interior
Faculdade de Engenharia
Departamento de Informática

Segurança Informática

Aula 2

Licenciatura em Engenharia Informática
Licenciatura em Informática Web

Computer Security

Lecture 2

Degree in Computer Science and Engineering
Degree in Web Informatics

Sumário

Definição de criptografia e criptanálise, modelos de ataques a sistemas criptográficos, segurança por obscurantismo e princípio de Kerckhoffs. Referência a alguns momentos e cifras que definem a evolução da criptografia clássica para moderna. A melhor aproximação prática à *one time pad*: cifras de chave simétrica contínuas.

Summary

Definition of cryptography and cryptanalysis, attack models on cryptographic systems, security by obscurity and Kerckhoffs' Principle. Discussion of some moments and ciphers that define the evolution of classical to modern cryptography. The best practical approximation to the one time pad*: symmetric-key stream ciphers.*

1 Criptografia, Criptanálise, Definição de Cifra e Modelos de Ataque

Criptography, Criptanalysis, Definition of Cipher and Attack Models

1.1 Criptografia e Criptanálise

Cryptography and Cryptanalysis

Criptografia – do grego *kryptos* (oculto, segredo) + *graph*, raiz de *graphein* (escrever).
Historicamente, a Criptografia consiste no **conjunto de técnicas que procuram tornar possível a comunicação secreta entre dois agentes, sobre um canal aberto**. Este conjunto de técnicas, que formam no fundo o **núcleo da criptografia**, é constituído por **cifras**, **mecanismos de integridade** e de **troca de chaves ou segredos criptográficos**.

Contudo, a criptografia moderna procura dar resposta a muitos mais requisitos do processo comunicativo para além da confidencialidade, tal como a **autenticidade**, o **anonimato** e o **não repúdio**. Aqui, essas propriedades são genericamente designadas por propriedades de segurança. É a **criptografia moderna** que **fornece as ferramentas** para se **cifrarem ficheiros em disco ou comunicações remotas**, **assinarem documentos digitalmente**, construir moedas virtuais (e.g., BitCoin) ou efetuarem leilões ou votações online de forma anónima e segura.

Criptanálise – Do grego *kryptos* (oculto, segredo) + *analýein* (desvendar).
Em contrapartida, a Criptanálise é constituída pelo **estudo de técnicas que visam** gorar os objetivos da Criptografia, isto é, **quebrar a segurança da comunicação**.

Os objectivos da criptanálise são, em última instância, os seguintes:

- **Obtenção do texto-limpo original** relativo a um dado criptograma;
- **Obtenção da chave de cifra** (ou de outra equivalente) usada para decifrar um determinado conjunto de criptogramas;
- Embora menos comum, pode ser também a descoberta do algoritmo de cifra (ou de um equivalente) usado para produzir determinado conjunto de criptogramas.

Criptologia – do grego *kryptos* (oculto, segredo) + *logia*, (estudo de).
Conjuntamente, a Criptografia e a Criptanálise **formam uma disciplina a que podemos chamar Criptologia**; uma área com **profundas raízes na Matemática e nas Ciências da Computação**. Um criptólogo é alguém que se dedica a estudar problemas tanto de criptografia como de criptanálise. Em inglês, os termos *Criptografia* e *Criptologia* usam-se de forma indiferenciada.

A criptografia é uma **ciência rigorosa** que elabora em **3 passos** simples para responder aos desafios que se lhe colocam:

1. **Especificar**, de modo preciso, **o modelo de ataque** a que o sistema estará sujeito;
2. **Propor uma construção** / sistema que simultaneamente preencha os requisitos e tenha em atenção o modelo de ataque;
3. **Provar que o comprometimento** da construção / sistema proposto **é aquivalente a resolver um problema matemático reconhecidamente difícil e que lhe está subjacente**.

1.2 Definição de Cifras
Definition of Cipher

A cifra constitui **um dos mecanismos mais importantes da criptografia**. Como ficará claro mais adiante, é possível definir **cifras de chave-simétrica** e **cifras de chave-pública**, bem como dividir as primeiras em cifras de chave-simétrica **contínuas** ou **por blocos**.

Definição de Cifra de Chave-Simétrica

Considere que $\mathcal{K}$ denota o espaço finito de chaves possíveis, $\mathcal{M}$ e $\mathcal{C}$ os espaços finitos dos mensagens e criptogramas, respetivamente. Uma cifra de chave-simétrica, definida para $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ é **um par de algoritmos *eficientes*** (E, D), em que

$$E : \mathcal{K} \times \mathcal{M} \to \mathcal{C}, D : \mathcal{K} \times \mathcal{C} \to \mathcal{M},$$

tal que

$$\forall m \in \mathcal{M}, k \in \mathcal{K} : D(k, E(k, m)) = m.$$

Repare que a definição anterior apenas diz que uma cifra é um par de algoritmos parametrizáveis por uma chave e que revertem os efeitos da sua aplicação reciprocamente. O algoritmo (ou operação) de **cifra é normalmente dotada de aleatoriedade**, enquanto que **a decifra é determinística**.

Definição de Cifra de Chave-Pública

Uma cifra de chave-pública, definida para $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ é **um terno de algoritmos *eficientes*** (G, E, D), em que

$$E : \mathcal{K} \times \mathcal{M} \to \mathcal{C}, D : \mathcal{K} \times \mathcal{C} \to \mathcal{M},$$

tal que $\forall (p_k, s_k)$ gerados por G, temos que $\forall m \in \mathcal{M}$

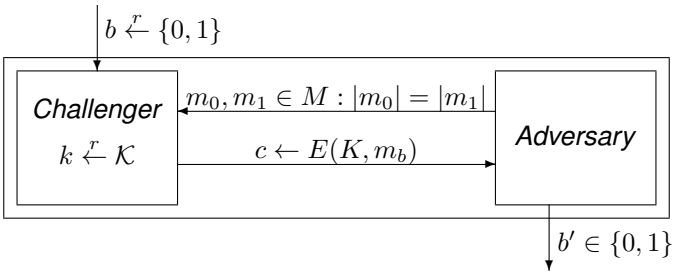
$$D(s_k, E(p_k, m)) = m.$$

Num tom mais coloquial, pode-se dizer que a definição determina a existência de três algoritmos, sendo que a função de um deles é debitar um par de chaves (p_k, s_k) associadas e que permitem que o algoritmo D inverta o efeito de $E(p_k, m)$ quando inicializado com a chave s_k associada.

1.3 Modelos de Ataque
Attack Models

Os modelos de ataque que se costumam definir no contexto da avaliação da segurança de uma cifra são os seguintes:

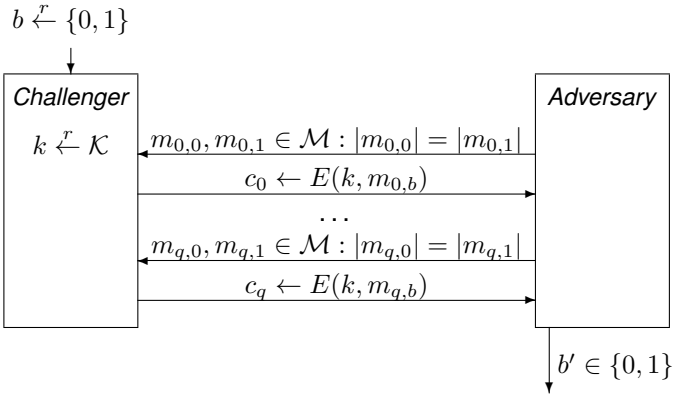
- Ataques em que **apenas o criptograma é conhecido** (*ciphertext-only attack* – COA), em que se procura obter o **texto-limpo** ou a cifra (**chave de cifra** ou algoritmo) que originaram determinado criptograma, partido só do conhecimento deste último.



  Este modelo é usado para definir a **segurança semântica** de **cifras de chaves simétricas contínuas**. Neste caso, considera-se **um jogo** em que um adversário envia duas mensagens com igual tamanho para a entidade que o desafia. A entidade escolhe uma chave aleatoriamente do espaço de chaves possíveis $\mathcal{K}$ e um b (ou 0 ou 1), cifra a mensagem m_b e envia-a ao adversário. A **cifra é semanticamente segura se**, para quaisquer duas mensagens escolhidas pelo adversário, **este não consegue diferenciar qual delas foi cifrada e enviada de volta**. Repare-se que este modelo de ataque é, na verdade, o mais fraco de todos.

- Ataques **com conhecimento de texto-limpo original** (*known plaintext attack* – KPA), em que o adversário (atacante) procura obter o texto-limpo ou a cifra (chave ou algoritmo) que originaram determinado criptograma, partido do conhecimento do criptograma e de partes do texto-limpo original ou de outros pares texto-limpo/criptograma.
  Este modelo de ataque foi usado durante a 2ª Grande Guerra. Os aliados intercetavam os criptogramas e sabiam parte dos mensagens que lhes correspondiam. Repare que, **ao contrário do modelo seguinte**, neste tipo de ataque, **o adversário não controla quais os pares texto-limpo/criptograma que conhece**.

- Ataques com **texto-limpo original escolhido** (*chosen-plaintext attack* – CPA), em que o adversário procura obter o texto-limpo ou a cifra (chave ou algoritmo) que originaram determinado criptograma, partindo do conhecimento deste último e **podendo pedir a um oráculo que cifre texto limpo à escolha numa fase anterior ao desafio**. Este ataque ilustra-se, normalmente, pelo diagrama incluido em baixo.



- Ataques com **texto-limpo escolhido de forma adaptativa** (*adaptative-chosen-plaintext attack* – CPA2), em que o adversário procura obter o texto-limpo ou a cifra (chave ou algoritmo) que originaram determinado criptograma, partido do conhecimento deste último e **podendo ir pedindo a um oráculo que cifre texto-limpo escolhido em função dos criptogramas obtidos** (dinâmico).
- Ataques com **criptogramas escolhidos** (*chosen-ciphertext attack* - CCA1), em que se procura obter o texto-limpo ou a cifra (chave ou algoritmo) que originaram determinado criptograma, partido do conhecimento deste útlimo e podendo pedir a um oráculo que **decifre vários criptogramas escolhidos**, mas diferentes daquele que queremos decifrar.
- Ataques com **criptogramas escolhidos de forma adaptativa** (*adaptive chosen-ciphertext attack* CCA2), em que o adversário **pode escolher criptogramas** e pedir ao oráculo que os decifre, podendo **cada pedido ser o resultado da análise dos pares texto-limpo/criptogramas anteriores**. Neste caso, o adversário pode pedir que o oráculo decifre criptogramas antes e depois do desafio, contrariamente ao anterior.

1.4 Ataque de Força Bruta
Brute Force Attack

Um **Ataque de Força Bruta** é aquele em que o adversário opta por **percorrer todo o espaço de chaves de cifra**, **inicializando e executando os algoritmos de cifra ou decifra** para encontrar o texto-limpo correspondente a um criptograma. Este ataque **pressupõe que o espaço de chaves é muito inferior ao espaço de mensagens, o que normalmente acontece**. É um ataque que **é sempre passível de ser aplicado a uma cifra**, mas c**uja viabilidade se encontra condicionada pelo tempo** que demora percorrer todo o espaço das chaves de cifra.

Pode, portanto, **ser evitado definindo espaços para chaves de cifra que são inviáveis de percorrer em tempo útil**. Repare-se que **este ataque não se inclui na criptanálise**, visto que esta disciplina (a criptanálise) se preocupa em fugir dos ataques de busca exaustiva o mais possível.

1.5 Segurança por Obscurantismo e o Princípio de Kerckhoffs
Obscurantism and the Kerckhoffs Principle

A criptografia existe desde a antiguidade, normalmente associada a **atividades militares e diplomáticas**. Por exemplo, **a segurança das cifras clássicas** dependia, quase exclusivamente, do **secretismo que rodeava as técnicas utilizadas** para cifrar e decifrar (o que, historicamente, se revelou *catastrófico*). Esta tendência fez-se notar ainda no Século XX, durante a 1ł e a 2ł Guerras Mundiais e prolongou-se durante as primeiras décadas da Guerra Fria. Contudo, em **1883**, **Auguste Kerckhoffs**, um linguista e criptógrafo alemão, escreveu **6 princípios básicos** para o desenho de cifras práticas, num artigo da revista das Ciências Militares chamado *La Cryptographie Militaire (Military Cryptography)*.

O segundo princípio dessa lista ficou conhecido como **o Princípio de Kerckhoffs**, que reza do seguinte teor:
[A cryptosystem] must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience;
Um sistema criptográfico deve ser seguro mesmo se tudo acerca desse sistema, à exceção da chave, é do conhecimento público.

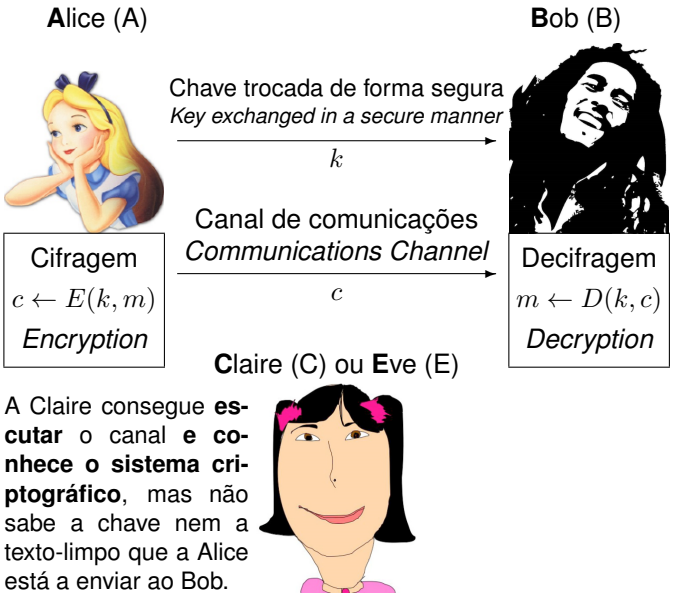
Este princípio é o aquele que ainda hoje define o trabalho de criptógrafos e criptanalistas. No fundo, para **avaliar a segurança** de uma técnica criptográfica, devemos **assumir que esta é do conhecimento de eventuais inimigos**.

Um **corolário** que deriva imediatamente do princípio enunciado é que **a segurança da cifra deve ser totalmente dependente de um parâmetro explícito: a chave de cifra**.

Uma metáfora material para o conceito de obscurantismo e para o princípio de Kerkoffs pode ser a seguinte:
*Ordenar fazer um cofre muito seguro para guardar segredos, e **nunca revelar os seus planos a ninguém corresponde a segurança por obscurantismo**. Logo que alguém descobrisse os planos do cofre ou conseguisse fazer engenharia reversa ao mesmo, a segurança poderia estar comprometida. O **princípio de Kerckhoffs** defende que um cofre verdadeiramente seguro seria aquele para o qual **os planos são conhecidos, e mesmo assim ninguém conseguia comprometer a sua segurança**.*

Um **modelo simples de comunicação** (e ataque) que se costuma colocar junto à discussão do princípio de Kerckhoffs é o que se inclui em baixo. Nele, estão representadas as partes interessadas em comunicar de forma segura (Alice e Bob), bem como uma entidade com índole maliciosa. Os termos *Alice* (A), *Bob* (B) e *Claire* (C) são usados para representar as partes em comuni-

cação, sobretudo por ser mais simples ler o texto com nomes, do que com *Parte A* e *Parte B*. A Alice, o Bob e a Claire são arquétipos introduzidos por Bruce Schneier[1] no seu livro *Applied Cryptography*. Há muitos outros nomes com significado específico nesta área (ver http://en.wikipedia.org/wiki/Alice_and_Bob).



A Claire consegue **escutar** o canal **e conhece o sistema criptográfico**, mas não sabe a chave nem a texto-limpo que a Alice está a enviar ao Bob.

2 Da Criptografia Clássica à Moderna

From Classical to Modern Criptography

2.1 Criptografia Clássica

Classical Criptography

Uma das cifras clássicas **mais antigas e *mais elaborada*** foi inventada pelos **Espartanos** em 400 A.C., que **usavam um bastão com diâmetro específico** para cifrar e decifrar mensagens. Os pergaminhos eram enrolados no bastão e a mensagem era escrita por cima. Somente enrolando novamente os pergaminhos num bastão de igual diâmetro se conseguia obter a mensagem.

Júlio César (44 A.C.) usava uma **cifra de substituição** aditiva das letras do alfabeto, em que a função de cifragem corresponde apenas a substituir cada letra do texto-limpo por aquela que lhe sucede ou antecede um número fixo de letras no alfabeto. **Este número fixo é a chave de cifra**. Matematicamente e para um alfabeto de 26 letras, a cifra pode ser definida por:

$$E(k, m_i) = m_i + k \bmod 26,$$

$$D(k, c_i) = c_i - k \bmod 26,$$

em que m_i e c_i representam as letras da mensagem e do criptograma, respetivamente e $k = \{1, ..., 25\}$.

[1] Um criptógrafo americano muito conhecido atualmente.

Repare que esta cifra é uma **cifra de substituição monoalfabética** (define apenas um alfabeto de substituição). Alguns detalhes desta cifra e de outras incluídas a seguir são explorados na aula prática mas, a titulo de exemplo, pode observar-se a cifragem da palavra `OLA` com esta cifra (chave $K = 20$):

```
Texto-limpo:  OLA (14 11  0)
Criptograma:  IFU (8  5  20)
```

Porque

$$E(20, O) = 14 + 20 \bmod 26 = 8,$$

$$E(20, L) = 11 + 20 \bmod 26 = 5, \text{ e}$$

$$E(20, A) = 0 + 20 \bmod 26 = 20.$$

No **século XIX**, **Vigenère** reinventou a cifra de Bellaso. As **grandes guerras fomentaram o desenvolvimento de máquinas para cifrar e decifrar**, sendo a **Enigma** o maior reflexo desse desenvolvimento (pelo menos a mais conhecida). Claro que este desenvolvimento **também fomentou o desenvolvimento da criptanálise e de super computadores** (para a altura) para análise da Enigma.

Ao contrário da cifra referida anteriormente, a cifra de Vigenère e a Enigma são cifras de substituição **polialfabéticas**, porque **definem substituições diferentes para cada posição do texto-limpo**.

A cifra de Vigenère é, de resto, **uma das cifras mais conhecidas** da criptografia clássica. **A chave de cifra é normalmente dada como uma sequência de caracteres de tamanho fixo** (e.g., `LEMON`) que é repetida e somada ou subtraída módulo 26 com cada letra do texto-limpo ou criptograma, respetivamente, para obter a cifragem ou a decifragem dos mesmos. Se a mensagem a cifrar for constituída por n letras do alfabeto, $m = m_1 m_2 ... m_n$, tal como o criptograma, $c = c_1 c_2 ... c_n$, e a chave for constituída por r letras, $k = k_1 k_2 ... k_r$, a cifra é definida pelas funções:

$$E(k, m_i) = m_i + k_{i \bmod r} \bmod 26,$$

$$E(k, c_i) = c_i - k_{i \bmod r} \bmod 26.$$

A titulo de exemplo, mostra-se a cifragem de `ATTACKATDAWN` com a chave-de-cifra `LEMON`:

```
Texto-limpo:  ATTACKATDAWN
Chave:        LEMONLEMNONL
Criptograma:  LXFOPVEFRNHR
```

É comum recorrer-se a uma tabela de subsituição como a que se inclui em baixo quando se lida com a cifra de Vigenère.

2.2 História da Criptografia Moderna

History of Modern Cryptography

A **era moderna** da criptografia **começou com a Teoria da Informação de Claude Shannon**, onde é definido o conceito da **entropia da informação** (ver *A Mathematical Theory of Communication by Claude E. Shannon*).

A criptografia é uma das áreas que mais tem vindo a evoluir nos últimos 60 anos, motivada sobretudo pelo enorme desenvolvimento que as telecomunicações, em geral, e a Internet, em particular, sofreram. Alguns dos (muitos) momentos que definem a história da criptografia moderna são:

1948-1949	Claude Shannon desenvolve a Teoria da Informação e enquadra o estudo da Criptografia nessa teoria.
1970-1977	Desenvolvimento e normalização do Data Encryption Standard (DES).
1976	Primeiro artigo científico de Diffie e Hellmann definindo os princípios da criptografia de chave-pública.
1978	Rivest, Shamir e Adleman descobrem (no sentido de inventar) a primeira cifra assimétrica: o RSA.
1985	Descoberta (no sentido de inventar) da cifra assimétrica El Gamal.
1995	Normalização do Digital Signature Algorithm (DSA).
1998	Normalização do Triple DES (3DES) como meio de reforçar a resistência a ataques de força bruta.
2001	Escolha do substituto do DES: o Advanced Encryption Standard (AES).
2007	Início dos esforços para encontrar o substituto do Secure Hash Algorithm 1 (SHA1.)
2012	Escolha do vencedor do concurso para encontrar o substituto do SHA1, tendo sido escolhido o algoritmo Keccak 2 para ser o SHA3.

2.3 Confusão e Difusão

Confusion and Diffusion

No seu artigo de 1949, intitulado Communication Theory of Secrecy Systems, Claude Shannon definiu dois métodos básicos que permitem frustrar a análise estatística de criptogramas devolvidos por uma boa cifra. Chamou a esses métodos **confusão** e **difusão**:

Confusão que consiste em **obscurecer a relação que existe entre o texto-limpo, o criptograma e a chave**. A ideia subjacente consiste em tornar difícil a obtenção do texto-limpo ou da chave a partir da análise do criptograma e de partes do texto-limpo. Note-se que a relação entre o critptograma e o texto-limpo é salvaguardado pela chave, que é a única que se considera verdadeiramente secreta (partes do texto-limpo podem ser conhecidas por análise estatística). Por isso, Shannon defende que esta relação (com a chave) deve ser o mais complexa possível, para não ser possível ao criptanalista recuperar a chave através da resolução de um simples sistema de equações. Na sua forma mais simples, implementa-se através de uma substituição.

Difusão que consiste e **espalhar as redundâncias do texto-limpo por todo o criptograma**, de forma a que seja necessária a recolha de uma grande quantidade de dados para reconstruir essa estrutura que foi difundida. Na sua forma mais simples, tal é conseguido através de uma permutação.

Como veremos adiante, **estes dois métodos são usados como pedras basilares na construção de cifras simétricas modernas**, nomeadamente o AES.

2.4 *One Time Pad*

One Time Pad

A *One Time Pad*, também conhecida como a **cifra de Vernam**, foi o **primeiro exemplo de uma cifra *segura*** (contra COA). Esta cifra, desenvolvida em **1917**, define-se da seguinte forma. Seja $\mathcal{M} = \mathcal{C} = \{0,1\}^n$ e $\mathcal{K} = \{0,1\}^n$, em que n **define o tamanho da mensagem a cifrar em bits**. As funções de cifragem e decifragem são:

$$C = E(k, m) = k \oplus m$$

e

$$D(k, c) = k \oplus c,$$

em que $\oplus$ denota a **operação de XOR aplicada bit a bit (i.e., adição módulo 2) aos seus argumentos**. A cifra determina ainda que **cada chave deve ser usada apenas uma vez** (*one time*) e **escolhida aleatoriamente para cada texto-limpo** a cifrar.

A operação XOR (adição módulo 2) está representada na tabela seguinte:

X	Y	$X \oplus Y$	
0	0	0	$0 + 0 \bmod 2 = 0$
0	1	1	$0 + 1 \bmod 2 = 1$
1	0	1	$1 + 0 \bmod 2 = 1$
1	1	0	$1 + 1 \bmod 2 = 0$

Note-se que a definição da cifra acima incluída está correta (i.e., vai de encontro à definição dada no início desta aula), já que

$$D(k, E(k, m)) = k \oplus E(k, m) = k \oplus k \oplus m = m.$$

O uso desta cifra pressupõe que se gera uma chave aleatória do mesmo tamanho que o texto-limpo, antes de o cifrar. E.g., se a mensagem a cifrar for `0 1 1 0 1 1 1` e a chave gerada for `1 0 1 1 0 1 0`, o criptograma é `1 1 0 1 1 0 1`:

`0 1 1 0 1 1 1` (texto-limpo)
`1 0 1 1 0 1 0` $\oplus$ (chave)
`1 1 0 1 1 0 1` (criptograma)

Se a chave for novamente gerada, o criptograma daria diferente. Dito de outra forma, **existe um criptograma diferente para cada chave possível.** É precisamente neste facto que o conceito de **segurança teórica da informação** assenta.

Segundo Shannon, uma cifra (E, D) definida sobre $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ apresenta **secretismo perfeito** se

$$\forall m_0, m_1 \in \mathcal{M}, |m_0| = |m_1|, \text{ e } \forall c \in \mathcal{C}$$

$$P[E(k, m_0) = c] = P[E(k, m_1) = c],$$

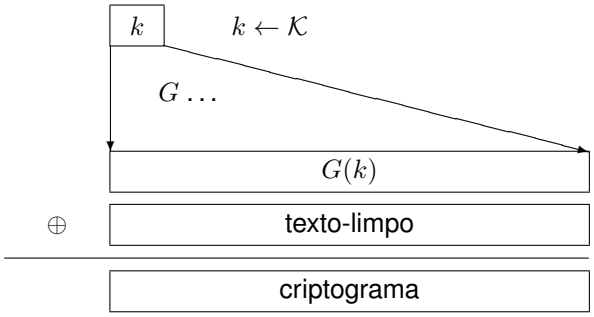
em que k é uma variável uniforme em $\mathcal{K}$ (i.e. $k \leftarrow \mathcal{K}$).
Repare que, basicamente, o que a definição de secretismo perfeito diz é que, para quaisquer duas mensagens, a probabilidade de obter o mesmo criptograma depois de cifradas é igual. Isto acontece para a *one time pad* porque é possível arranjar sempre uma chave que, depois de XORed com uma ou outra mensagem, dê o criptograma pretendido.

Repare-se também que **o conhecimento de um criptograma desta cifra não revela absolutamente nada acerca do texto-limpo ou da chave** (a não ser que se conheçam factos do texto-limpo ou da chave), já que qualquer texto-limpo pode ser cifrado para determinado criptograma, dependendo apenas da chave gerada. A cifra é também **extremamente eficiente**, já que a **cifragem e a decifragem consistem apenas no XOR da mensagem com a chave**. Apesar das suas grandes vantagens, **esta cifra não é usada na prática** porque **as chaves de cifra** (que também têm de ser trocadas entre as entidades que comunicam) **têm de ser geradas para cada mensagem**, e porque **têm de ter o mesmo tamanho da mensagem**. Para além disso, a cifra é maleável, pelo que não pode ser usada nalguns casos. Prova-se que as cifras que apresentem *secretismo perfeito* **necessitam de chaves com tamanho igual ou superior ao tamanho do texto-limpo, pelo que não práticas de serem utilizadas**.

3 Cifras de Chave Simétrica Contínuas

Stream Ciphers

Em cima foi dito que a cifra *one time pad* oferece secretismo perfeito e uma eficiência muito elevada, mas que não pode ser usada na prática devido ao tamanho das chaves. As **cifras de chave simétrica continuas** refletem o esforço de a tornar praticável, elaborando na ideia de **substituir a chave aleatória** de tamanho n igual ao do texto limpo **por uma chave pseudo-aleatória que é gerada, a partir de uma semente, por um gerador *seguro***.



3.1 Definição

Definition

De um modo geral, as cifras de chave simétrica contínuas são definidas da seguinte forma. Seja $\mathcal{M} = \mathcal{C} = \{0,1\}^n$ e $\mathcal{K} = \{0,1\}^s$, em que n **define o tamanho da mensagem a cifrar em bits** e s denota o tamanho da chave de cifra, também em bits. As funções de cifragem e decifragem são:

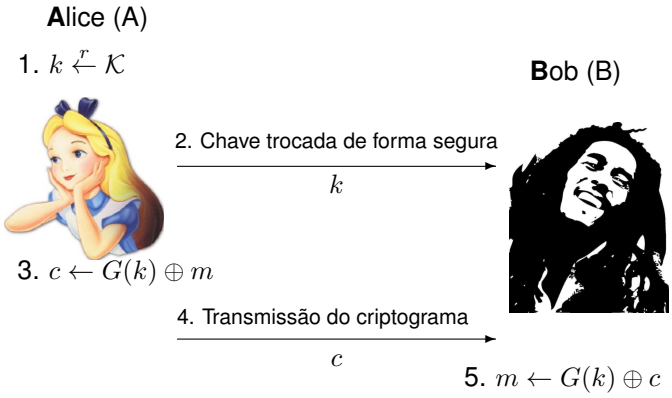
$$C = E(k, m) = G(k) \oplus m$$

e

$$D(k, c) = G(k) \oplus c,$$

em que $\oplus$ denota a **operação de XOR aplicada bit a bit (i.e., adição módulo 2) aos seus argumentos**, e $G(k)$ **simboliza um gerador de sequencias pseudo-aleatórias que expande** k **para o tamanho do texto-limpo a cifrar**.

A figura seguinte ilustra o protocolo seguido pela Alice e pelo Bob para comunicarem de forma segura usando uma cifra de chave simétrica contínua.



Os passos representados na figura são:

1. A Alice gera uma chave aleatória k ($k \xleftarrow{r} \mathcal{K}$).
2. A Alice e o Bob trocam a chave de forma segura (e.g., encontram-se num café para conversar).
3. Mais tarde, a Alice liga-se à Internet, inicializa um gerador de sequências pseudo-aleatórias seguro com a chave k, e faz o XOR da sua expansão com a mensagem que quer transmitir.

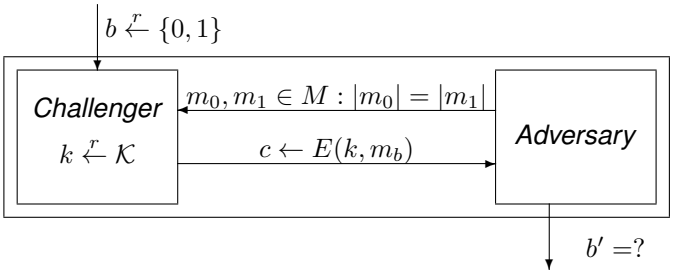
4. A Alice transmite o criptograma c sobre o canal potencialmente inseguro.
5. Bob recebe o criptograma, e começa por inicializar o mesmo gerador de sequências pseudo-aleatórias que a Alice, com a mesma semente (chave k) que ela usou.
6. Finalmente, o Bob decifra a mensagem, fazendo o XOR do criptograma com a expansão obtida.

3.2 Segurança Semântica sob COA
Semantic Security under COA

Dado as cifras de chave simétrica contínuas (e outras que veremos mais à frente) relaxarem o tamanho da chave para tornar a cifra *one time pad* prática, **estas não podem exibir secretismo perfeito**. É necessário uma nova definição de segurança, que depende, para estas cifras, do gerador de sequências pseudo-aleatórias.

Para estas cifras em específico, considera-se sempre que as chaves são apenas usadas uma única vez, e que **o atacante terá à sua disposição apenas um criptograma cifrado para cada chave**. Assim, o ataque que lhe é permitido montar a este tipo de cifra é apenas o COA. Adiante, colocar-se-ão outros cenários mais elaborados, e definir-se-à segurança semântica em cada contexto.

A **definição de segurança semântica para cifras simétricas contínuas é assim feita em relação ao jogo** apresentado na figura que acompanha o **modelo COA**, transcrita para esta secção por comodidade.



Considere que se definiam dois tipos de experiência diferentes no jogo acima ilustrado: na experiência 0 ($b = 0$), a mensagem cifrada pelo *Challenger* era sempre m_0, enquanto que na experiência 1 ($b = 1$), a mensagem cifrada era sempre m_1. Neste jogo, o *Challenger* repetia várias vezes a experiência 0 e a 1 (aleatoriamente), escolhendo uma chave de cifra, também aleatoriamente, para cada tentativa. No final de cada experiência, enviava a cifra resultante ao adversário e este tenta adivinhar qual (das duas que enviou) foi a mensagem que foi cifrado nessa vez. **Se o adversário errar aproximadamente tantas vezes como acerta, então a cifra é semanticamente segura.**

Matematicamente, a ideia anterior formaliza-se através do **conceito de vantagem**. Considere que W_0 e W_1 denotam o evento em que o adversário diz que $b = 1$ na experiência 0 e na experiência 1. A vantagem $\mathsf{Adv}_{SS}(A, E)$ é definida por:

$$\mathsf{Adv}_{SS}(A, E) = |P(W_0) - P(W_1)| \in [0, 1].$$

A vantagem $\mathsf{Adv}_{SS}(A, E)$ é 1 se o adversário consegue distinguir perfeitamente as experiências (i.e., neste caso, $P(W_0) = 0$ e $P(W_1) = 1$, logo $|P(W_0) - P(W_1)| = 1$), e é próximo de 0 caso a cifra o *confunda quase* sempre (i.e., $P(W_0) = 0.5$ e $P(W_1) = 0.5$, logo $|P(W_0) - P(W_1)| = 0$).

Formalmente, diz-se que a cifra é semanticamente segura sobre o COA se $\mathsf{Adv}_{SS}(A, E)$ for negligenciável, i.e., muito próxima de 0:

$$\mathsf{Adv}_{SS}(A, E) < \epsilon \leq \frac{1}{2^{80}}$$

Note que, **atualmente**, considera-se que **a cifra é semanticamente segura se a vantagem for menor do que** $1/2^{80}$.

Para as cifras de chave simétrica contínuas, um prérequisito que necessita ser satisfeito para que sejam semanticamente seguras é que **o gerador de sequências pseudo-aleatórias seja imprevisível**, i.e., não é conhecido nenhum algoritmo eficiente que seja capaz de prever a evolução de $G(k)$ a partir de nenhum número finito de bits.

3.3 *Two Time Pad*
Two Time Pad

Em cima foi dito que **a chave de cifra não deve ser usada mais do que uma vez** para cifrar mensagens diferentes. Considere que havia cifrado duas mensagens m_0 e m_1 com a mesma chave k, dando origem a $c_0 = m_0 \oplus G(k)$ e $c_1 = m_1 \oplus G(k)$. Isoladamente, c_0 e c_1 não dão informação nenhuma sobre as mensagens que escondem, mas o XOR dos criptogramas revela uma vulnerabilidade:

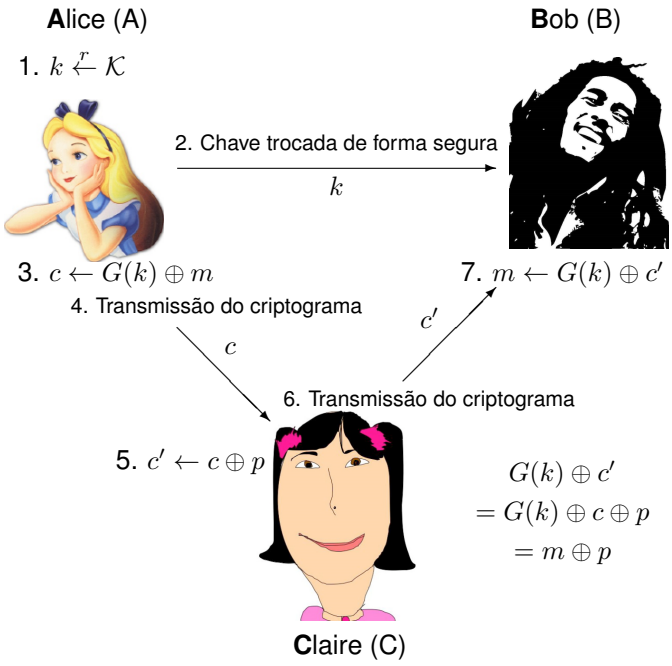
$$c_0 \oplus c_1 = m_0 \oplus G(k) \oplus m_1 \oplus G(k) = m_0 \oplus m_1$$

Neste caso, **pode haver redundância suficiente na linguagem e codificação utilizada (e.g., ASCII) para permitir reaver as mensagens originais a partir do seu XOR** ($m_0 \oplus m_1$).

3.4 Maneabilidade
Maleability

As cifras de chave simétrica continuas **podem ser usadas** em situações em que o **canal de comunicação pode ser escutado mas não manipulado** ou sujeito a perdas de bits (não confundir *perdas* com *erros*). O problema é que **estas cifras são maneáveis**, o que significa que **o criptograma pode ser alterado sem que tal facto seja notado após decifragem**. Mais, **a alteração tem impacto previsível no texto limpo**.

Por exemplo, considere observar a figura incluída em baixo, onde se ilustra **um ataque de homem no meio ativo** (ou mulher no meio, neste caso). A Claire tem a **capacidade de escutar** o canal de comunicações, **intercetar**, **manipular**, e **voltar a injetar mensagens** na comunicação (daí a designação de *ativo*).



Se a Alice quiser enviar a mensagem `100010` cifrada com a chave `101011`, calcula:

```
  100010  (mensagem original enviada pela Alice)
⊕ 101011  (chave)
  001001  (criptograma),
```

e envia `001001`. Se a Claire quiser alterar o primeiro bit da mensagem original, apenas tem de intercetar o criptograma e calcular o seu XOR com `100000`. Repare-se que **a Claire nunca chega a decifrar ou a atacar a cifra de qualquer outra forma**:

```
  001001  (criptograma),
⊕ 100000  (alteração feita pela Claire)
  101001  (resultado).
```

A Claire envia `101001` ao Bob, que o decifra com a chave `101011`:

```
  101001  (resultado)
⊕ 101011  (chave)
  100010  (mensagem recebida pelo Bob).
```

Note que o texto-limpo obtido é diferente do original no primeiro bit e **o Bob não tem como detetar esta alteração**.

3.5 Exemplos de Cifras de Chave Simétrica Contínuas

Examples of Stream Ciphers

Um dos **melhores exemplos** de cifras de chave simétrica continua é a **Rivest Cipher 4 (RC4)**, que aceita uma **chave de 128 bits** e foi desenvolvida para a empresa Rivest, Shamir e Adleman (RSA) por Ron Rivest, em **1987**. **Esta foi uma das cifras mais usadas até à poucos anos atrás**, já que faz parte dos protocolos *Transport Layer Security* (TLS) e *Wired Equivalent Privacy* (WEP).

Esta cifra foi inicialmente mantida secreta através de acordos de sigilo mas, em 1994, *alguém* alguém publicou os detalhes da cifra num *billboard* (supostamente o próprio Ron Rivest, por acreditar no princípio de Kerckhoffs). Uma utilização pouco cuidada desta cifra levou à vulnerabilidade descoberta no *standard* WEP.

Exemplos de cifras de chave simétrica contínuas **mais modernas** podem ser encontrados no projeto *eSTREAM: the ECRYPT Stream Cipher Project*, **composto por 7 cifras deste tipo**, algumas delas vocacionadas para implementações em harware, outras para implementações em software. Por exemplo, **a Salsa20/r é uma cifra orientada para implementações em software** proposta por Daniel J. Bernstein, **que aceita chaves de 128 e 256 bits**.

3.6 Eficiência Computacional

Computational Performance

De modo a obter uma ideia da *performance* deste tipo de algoritmos, podem-se mencionar alguns valores relativos à velocidade de geração de valores dos dois geradores mencionados em cima. Estes valores dizem respeito à implementação em C++ de Wei Dai na biblioteca Crypto++ 5.6.0, e foram obtidos numa máquina AMD Opteron com um processador a 2.2 GHz com sistema operativo Linux:

	Gerador	Velocidade (MB/sec)
	RC4	126
eStream	Salsa20/12	643
eStream	Sosemanuk	727

Nota: o conteúdo exposto na aula e aqui contido não é (nem deve ser considerado) suficiente para total entendimento do conteúdo programático desta unidade curricular e deve ser complementado com algum empenho e investigação pessoal.