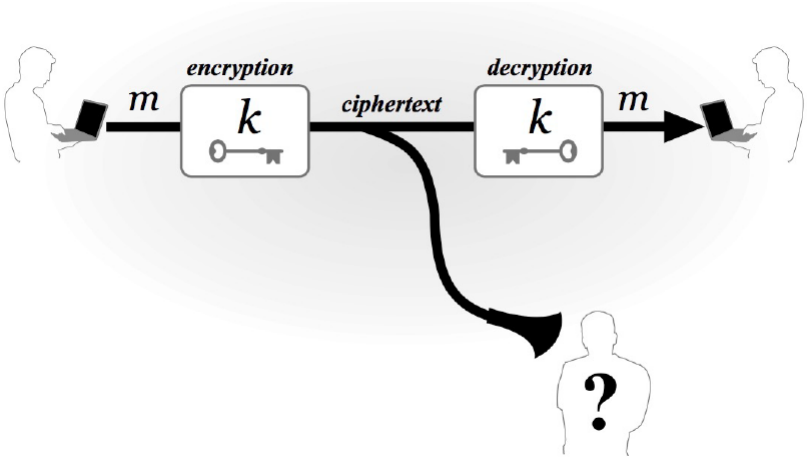
CAN304 W2

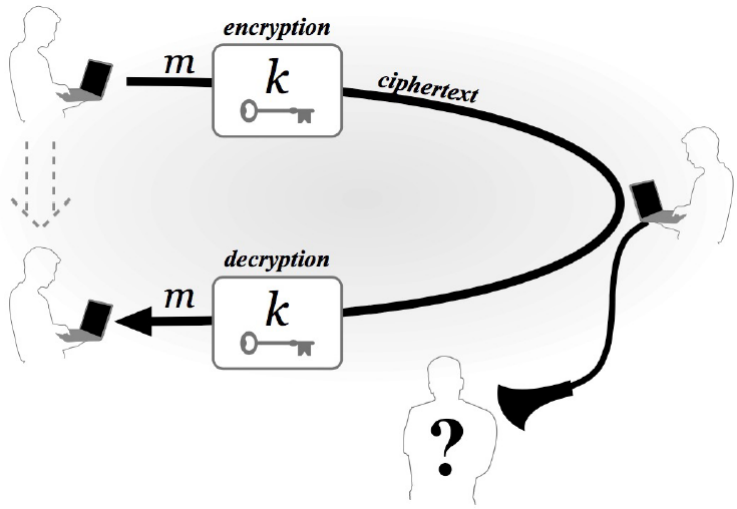
Classical and modern cryptography

Classical cryptography 完全依赖于通信双方之间共享的秘密信息(Private-key cryptography, 又叫 secret-key / shared-key / symmetric-key cryptography)。

Secure communication：双方共享一个密钥，用于安全通信。



Secure storage：单个用户可以在一段时间内安全地存储数据。



Private-key encryption

A private-key encryption scheme is defined by a message space M and algorithms (Gen, Enc, Dec):

- Gen (key-generation algorithm): generates k
- Enc (encryption algorithm): takes key k and message $m \in M$ as input; outputs ciphertext c

$$c \leftarrow Enc_k(m)$$

- Dec (decryption algorithm): takes key k and ciphertext c as input; outputs m or "error"

$$Dec_k(c) = m$$

  - For all $m \in M$ and k output by Gen,

$$Dec_k(Enc_k(m)) = m$$

The shift cipher

对字母进行加密，如果密钥 k = 1，那么字母 A 加密后变成字母 B ...。解密的时候逆向进行该操作。

- M = {strings over lowercase English alphabet}

- Gen: choose uniform $k \in \{0, \ldots, 25\}$

- $Enc_k(m_1 \ldots m_t)$: output $c_1 \ldots c_t$, where

$$c_i := [m_i + k \bmod 26]$$

- $Dec_k(c_1 \ldots c_t)$: output $m_1 \ldots m_t$, where

$$m_i := [c_i - k \bmod 26]$$

然而 shift cipher 并不安全，因为它的密钥空间 (key space) 较小，仅为 26。

密钥空间应足够大，以防止"暴力破解 (brute-force)"详尽搜索攻击 (exhaustive-search attacks)。

The Vigenère cipher

现在密钥是一个字符串。

要进行加密，将内容依据密钥指示向后挪动相应的位。解密即反转该过程。

例如：k=‘cafe’ (a 代表 0，b 代表 1 ...)

```
tellhimaboutme
cafecafecafeca
veqpjiredozxoe
```

key space：如果 key 是一个长度为 14 的字符串，那么 key space 的大小为 $26^{14} \approx 2^{66}$。

因此 Vigenère cipher 很安全，使用 brute-force search 几乎不可能。

不过如果被人知道了 key 的长度，那么破解起来就很容易：假如 key 长度为 14，那么位置在第 1，15，... 的元素都是用同一种方式加密，这样就可以破解密码。

Symmetric encryption

Definition

- A pair of “efficient” algorithms (E, D)
- defined over (K, M, C)
- where $E: K \times M \rightarrow C, D: K \times C \rightarrow M$
- such that $\forall m \in M, k \in K: D\big(k, E(k, m)\big) = m$

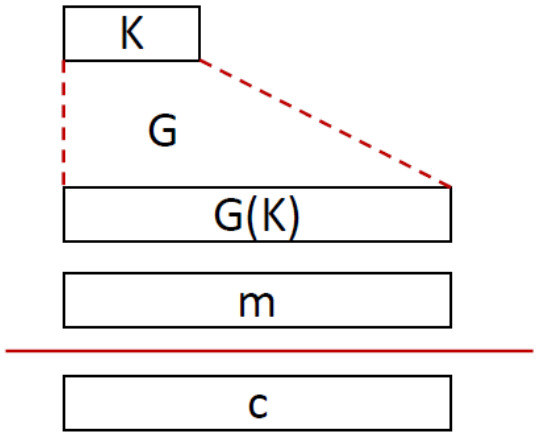
注：E 是加密算法，D 是解密算法。K 是密钥，M 是明文信息，C 是密文信息。

Stream ciphers vs. Block ciphers

Stream ciphers：对于一个密钥 k (k 一般比明文 m 短)，使用算法 G 将 k 变得和 m 长度一致 (k 是种子密钥，G 是随机数发生器，生成的结果为 key stream)，然后对明文一个一个 (一 bit 一 bit) 进行加密。解密为其逆过程。

Stream ciphers

- $c := E(K, m) = G(K) \oplus m$
- $m := D(K, c) = G(K) \oplus c$



Block ciphers：将 m 分成指定长度的 block，一块一块进行加密。

两者的主要区别是按 bit 加密，和按块加密。

Advantages of block ciphers:

- Good diffusion
  - 更容易使一组加密字符相互依赖，因为 block 中有多个 bit，这些 bit 一起被加密，之间会产生联系
- Immunity to insertions
  - 加密文本是已知长度，因为我们知道 block 的长度，如果给密文中插入，那么长度就会不一致
- Most common Internet crypto are done with block ciphers

Disadvantages of block ciphers:

- Slower
  - 在加密/解密开始之前需要等待数据块
- Worse error propagation
  - 错误会影响整个块 (要是出错，整个块都会出错)

Block ciphers (AES)

The Data Encryption Standard：DES，Block ciphers 的一种。使用 ***substitutions*** **(**替换**)**, ***permutations*** **(**排列，比如改变顺序**)**, ***table lookups*** **(**表查找**)** 进行加密；加密会进行很多轮，每轮重复上述操作。但 DES 并不安全，因为其 key 不够长 (short key)。

The Advanced Encryption Standard：AES，可以作为 DES 的替代品，使用 combination of permutation and substitution 加密。

AES Internals（下面不考）

一个 128 bit 的 process block 可以选择长度为 128, 192, 和 256 bit 的 key。

我们可以将 16 byte (16 byte = 128 bit) 的明文看成由 byte 组成的二维 array s，这个 array 被称为 internal state。

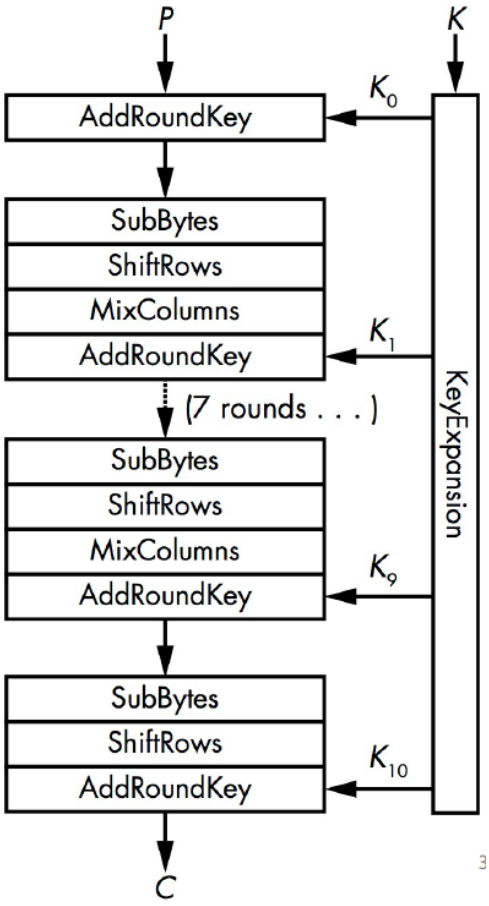
s_{00}	s_{01}	s_{02}	s_{03}
s_{10}	s_{11}	s_{12}	s_{13}
s_{20}	s_{21}	s_{22}	s_{23}
s_{30}	s_{31}	s_{32}	s_{33}

The internal state of AES viewed as a 4 × 4 array of 16 bytes.

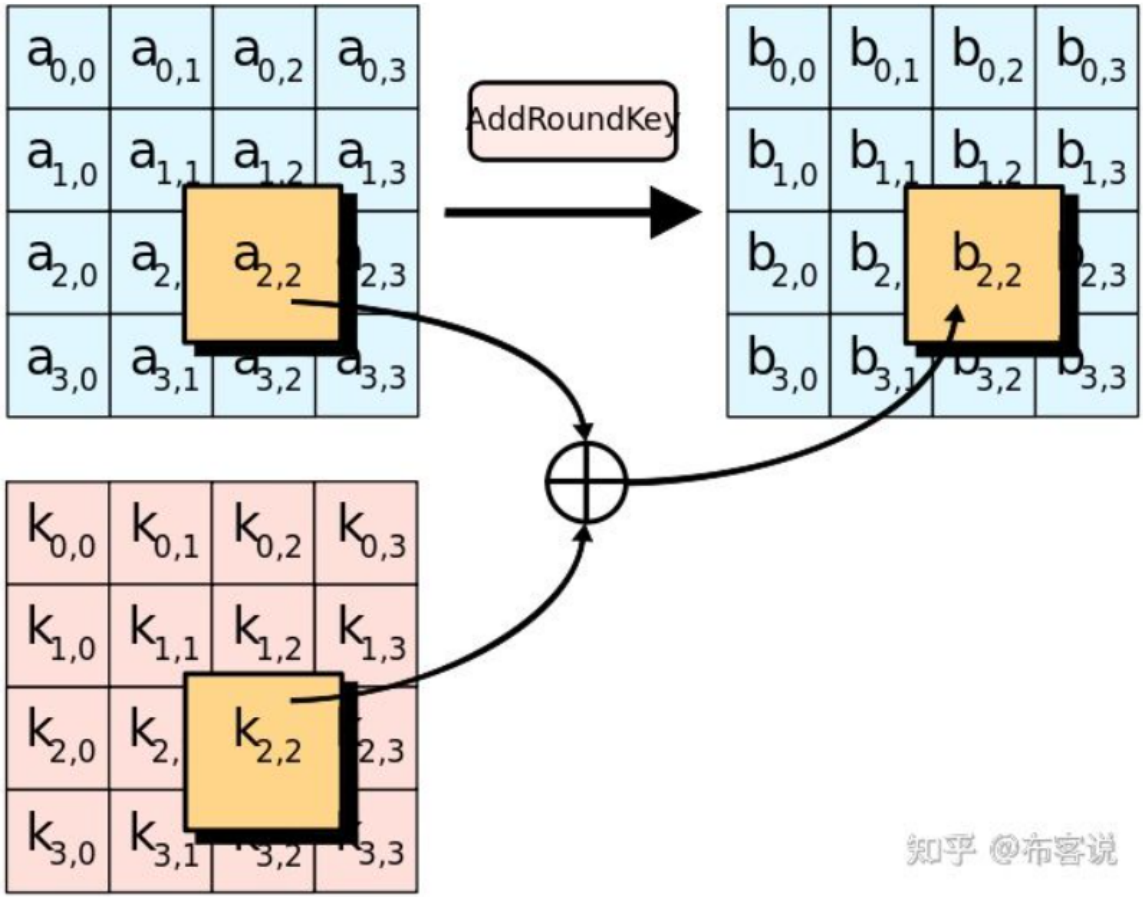
AES 使用 SPN structure 转换 array 的 byte、列和行，以生成密文。

Substitution–permutation network (SPN)

- In order to transform its state, AES uses an SPN structure, with 10 rounds for 128-bit keys, 12 for 192-bit keys, and 14 for 256-bit keys.
- Four building blocks
  - AddRoundKey
  - SubBytes
  - ShiftRows
  - MixColumns



- AddRoundKey
  - SPN 每轮会产生一个 round key，round key 的长度为 4，即每个元素 32 bit。将 128 bit 的 round key 和 internal state 中的数据进行逐位异或操作



- SubBytes
  - 根据 S-box 将每个 byte (s_{00}, s_{01} ,... , s_{33}) 替换为其他 byte (假如 s_{01} 的值为 2e，那么它将被替换为 31)

		y														
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
x	0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
	1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
	2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
	3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
	4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
	5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
	6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
	7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
	8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
	9	60	81	4F	DC	22	2A	90	88	46	EE	B9	14	DE	5E	0B	DB
	A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
	B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
	C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
	D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
	E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
	F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

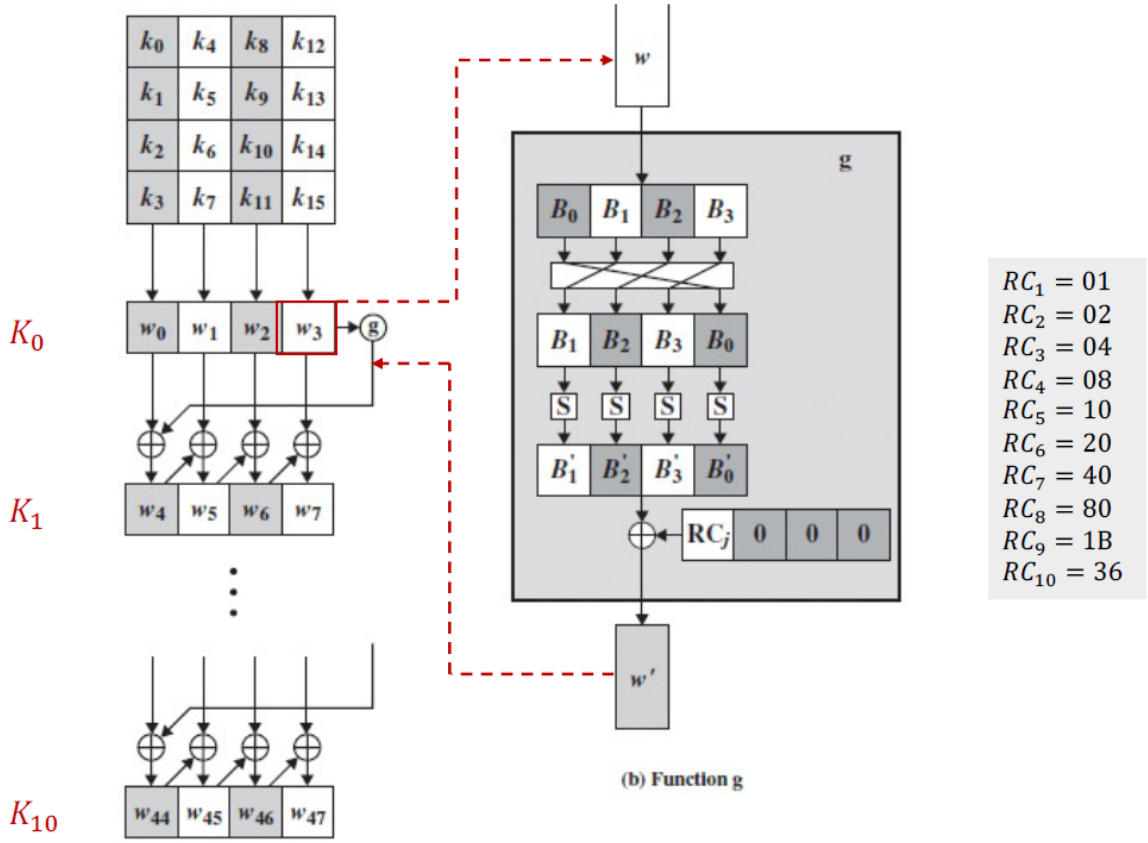
- ShiftRows
  - 将第 i 行的第 i 个位置的元素移动到最左边 (整行一起移动)

$$\begin{bmatrix} S_{00} & S_{01} & S_{02} & S_{03} \\ S_{10} & S_{11} & S_{12} & S_{13} \\ S_{20} & S_{21} & S_{22} & S_{23} \\ S_{30} & S_{31} & S_{32} & S_{33} \end{bmatrix} \xrightarrow{\text{shiftRows( )}} \begin{bmatrix} S_{00} & S_{01} & S_{02} & S_{03} \\ S_{11} & S_{12} & S_{13} & S_{10} \\ S_{22} & S_{23} & S_{20} & S_{21} \\ S_{33} & S_{30} & S_{31} & S_{32} \end{bmatrix}$$

- MixColumns
  - 将每列的 4 个 byte 作为输入，和一个给定的 matrix 相乘，从而将它们转换为新的 byte

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} S_{00} & S_{01} & S_{02} & S_{03} \\ S_{10} & S_{11} & S_{12} & S_{13} \\ S_{20} & S_{21} & S_{22} & S_{23} \\ S_{30} & S_{31} & S_{32} & S_{33} \end{bmatrix} = \begin{bmatrix} S'_{00} & S'_{01} & S'_{02} & S'_{03} \\ S_{10} & S'_{11} & S_{12} & S'_{13} \\ S'_{20} & S'_{21} & S'_{22} & S'_{23} \\ S'_{30} & S'_{31} & S'_{32} & S'_{33} \end{bmatrix}$$

- KeyExpansion
  - 从 16 byte 的 initial key 创建 11 个 16 byte 的 round key (k0-k10)，其中使用和 SubBytes 中一样的 S-box 和异或操作



注：上面是 Key schedule function，k 是 initial key，w 是 round key (也是 16 byte)。通过上述机制生成 round key。

AES 与 block cipher 一样安全：所有输出 bit 都以某种复杂的伪随机方式依赖于所有输入 bit。但没有证据表明 AES 对所有可能的攻击免疫(例如，新的侧信道攻击，new side-channel attacks，核心思想是通过加密软件或硬件运行时产生的各种泄漏信息获取密文信息)。

Cryptographic modes

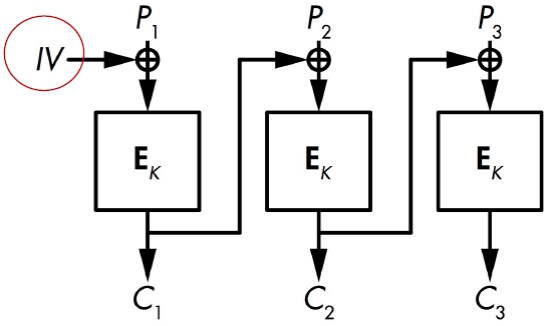
Block cipher 和 stream cipher 都是基本的加密方式，但现实中要求的加密可能更复杂 (比如 block cipher 只能加密固定长度的内容，但现实中要求加密任意长度的内容)。因此，应用 cipher 的方法就是 cryptographic mode。相同的加密方式可以用在不同的模式中。

Electronic Codebook (ECB) Mode：ECB 是 block cipher 的一种最基本的工作模式。在该模式下，待处理信息被分为大小合适的分组，然后分别对每一分组独立进行加密或解密处理。但 ECB 的所有分组的加密方式一致，这会导致相同的明文被加密后的密文也相同 - 不安全。

CBC mode

Cipher Block Chaining (CBC) mode 可以将一组相关的加密块绑定在一起，并隐藏相同内容的两个块。

CBC 使用先前块的密文来加密当前块：将先前块的密文和当前块的明文进行异或操作，再加密其结果。这样每个块的加密取决于所有先前块的内容。



不过 CBC 有一个缺陷，对于第一个 block 没有之前的块进行加密，这会导致和 ECB 相同的问题。因此我们使用 initialization vectors (IV) 来解决这个问题。

IV 会随机生成一个 string 来代替之前的块对第一个块进行加密。这确保了加密结果始终是唯一的。

First block of message

1	1	0	1	0	0	0	1

Second block of message

0	0	0	1	1	0	0	0

Initialization vector

0	1	0	0	1	1	0	0

Use previous msg for CBC

XOR IV and message

1	0	0	1	1	1	0	1

Apply CBC (XOR P2 with C1)

0	0	1	0	1	1	1	1

Encrypt message and send IV plus message

0	0	1	1	0	1	1	1

Encrypt and send second block of message

1	0	0	1	1	1	1	0

解密方会收到 IV 以及一系列密文 (C1, C2, ...)。

Uses of symmetric cryptography

Symmetric cryptography 可以被用于很多事：

- Secrecy (confidentiality)：只有知道密钥的人才能解密 - 保证机密性
- Authentication
  - 只有我和 A 知道密钥 k，如果我发消息给某人，那人解密了消息，那人必然是 A
  - 但这存在一些问题，比如 non-repudiation problem：只有我和 A 知道密钥，我加密了一个消息，但我否认是我加密的，这样第三方就不知道到底是谁加密的消息 (因为我和 A 都可以加密)。

    同样的，如果是三个人共享密钥，那就更不知道到底是谁加密的消息 (这种情况可以通过加密公钥来解决，即非对称加密)。

    如果 authentication 的时候不考虑泄密问题，可以不使用加密。
- Prevention of alteration (integrity)：更改加密消息会使解密后的明文变得完全混乱；如果 checksum 被用于检测，也可以检查出消息是否被更改。

Symmetric cryptography 还存在一个问题：在互联网上，每两个人之间需要共享一个密钥进行通信，那么需要极多的密钥才能让整个互联网正常工作。