现代密码学
Modern Cryptography

张方国

中山大学计算机学院

Office: Room 305, IM School Building
E-mail: isszhfg@mail.sysu.edu.cn
HomePage: https://cse.sysu.edu.cn/content/2460

第十三讲 Hash函数（三）

- 消息认证码MAC
- 安全消息认证码的构造
- 消息认证码的应用

消息认证码

消息认证码即满足某些安全性质的带密钥的Hash函数。

MAC所需要的安全性质与（不带密钥）Hash函数所需要的安全性质是截然不同的。

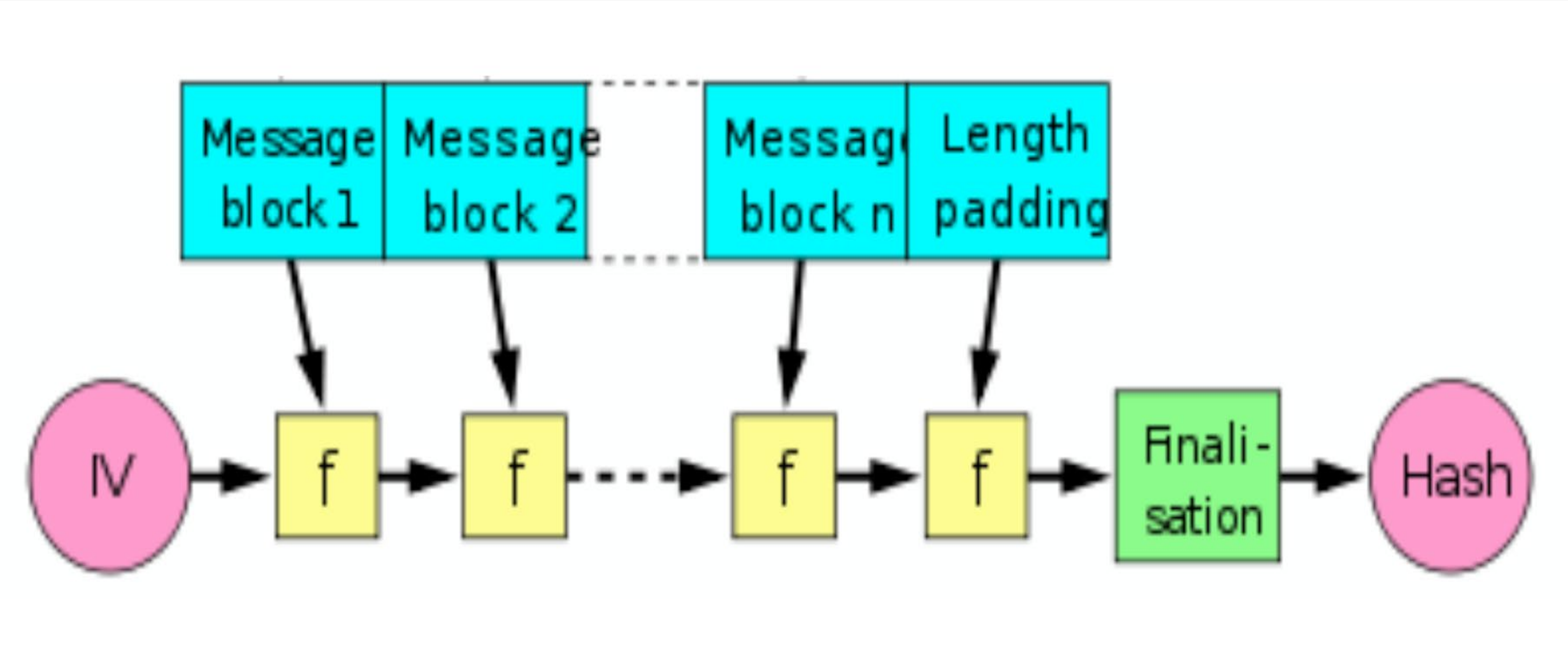
构造MAC的一个常用方法是通过一个密钥作为要被Hash的消息的一部分，从而在一个不带密钥的Hash函数中介入一个秘密密钥。

消息认证码MAC

- 认证码(MAC,也称密码检验和)
  - 对选定消息,使用一个密钥,产生一个短小的定长数据分组,称认证码,并将它附加在消息中,提供认证功能. (MAC = C_k(M) ,其中M是可变长的消息, K是共享密钥,C_k(M)是定长的认证码.)
- 应用认证码,如果只有收发方知道密钥,同时收到的MAC与计算得出的MAC匹配:
  - 确认消息未被更改;
  - 确信消息来自所谓的发送者;
  - 如果消息包含序号,可确信该序号的正确性;

- MAC函数应有如下性质(攻击者没有K):
  - 有M和$C_k(M)$,试图生成M′, 使得$C_k(M') = C_k(M)$,这在计算上不可行;
  - $C_k(M)$应能均匀分布;对于随机选取的消息M和M′, $C_k(M) = C_k(M')$的概率为2^{-n}其中n 为MAC的比特长度;(抗选择明文攻击)
  - 消息M′为M的某种已知代换,即M′=f(M),则$C_k(M) = C_k(M')$的概率为2^{-n}.

迭代Hash函数构造


Message block 1
Message block 2
Message block n
Length padding
IV
f
f
f
f
Finali-sation
Hash

消息认证码的构造

我们使用一个不带密钥的迭代Hash函数h来构造一个新的带密钥的Hash函数h_K，假设$IV = K$，并保密该值。

假定h没有预处理或输出变换。这样的Hash函数需要每个输入消息x的长度是t的倍数，其中$compress : \{0,1\}^{m+t} \to \{0,1\}^m$是用于建立$h$的压缩函数。而且，密钥$K$的长度是$m$比特。

攻击1：攻击者怎样对给定的任何消息x及相应的MAC，即$h_K(x)$，无须知道密钥K，就可以构造某个消息有效MAC。设x'为任意的长度为t的比特串，考虑消息$x||x'$。

这个消息的MAC，即$h_K(x||x')$可用下式计算

$$h_K(x||x') = compress(h_K(x)||x')$$

因为$h_K(x)$和x'都是已知的，对攻击者来说，即使K是保密的，计算$h_K(x||x')$也是一件简单的事情。

消息认证码的构造

攻击2： 即使消息被填充，攻击仍然成立。

假定在预处理步骤中$y = x||pad(x)$。注意到对某一整数r，有$|y| = rt$。设w是长度为t的任意比特串，定义$x' = x||pad(x)||w$。在预处理中，我们计算

$$y' = x'||pad(x') = x||pad(x)||w||pad(x')$$

其中对某一整数$r' > r$，有$|y'| = r't$。

显然有$z_r = h_K(x)$，因此攻击者可做如下计算：

$$z_{r+1} \leftarrow compress(h_K(x)||y_{r+1})$$

$$z_{r+2} \leftarrow compress(z_{r+1}||y_{r+2})$$

$$\cdots$$

$$z_{r'} \leftarrow compress(z_{r'-1}||y_{r'})$$

则$h_K(x') = z_{r'}$，因此攻击者即使不知道密钥K，也可以计算出$h_K(x')$。

消息认证码的构造

MAC安全性的含义：攻击者的目标是试图在一个未知但是固定的密钥K下，产生一对有效的(x,y)。

攻击者允许请求Q个自己选择的消息$x_1,x_2,\ldots$的有效MAC。（假设存在一个喻示器或黑盒子）

攻击者通过向喻示器提出消息$x_1,\ldots,x_Q$的请求，得到一系列有效对（在未知密钥K的情况下）：

$$(x_1,y_1),\ldots,(x_Q,y_Q)$$

后来，当攻击者输出对(x,y)时，要求$x\notin\{x_1,\ldots,x_Q\}$。

如果攻击者输出一个假冒的概率至少为ε（最差情况的概率），则攻击者对给定的MAC，被称为一个(ε,Q)假冒者。

安全消息认证码的构造：嵌套

嵌套MAC： 一个嵌套MAC是指合成两个（带密钥的）Hash族来建立一个MAC算法。

假定$(\mathbb{X},\mathbb{Y},\mathbb{K},\mathbb{G})$和$(\mathbb{Y},\mathbb{Z},\mathbb{L},\mathbb{H})$是Hash族。这些Hash族的复合是指Hash族$(\mathbb{X},\mathbb{Z},\mathbb{M},\mathbb{G}\circ\mathbb{H})$，其中$\mathbb{M}=\mathbb{K}\times\mathbb{L}$，并且

$$\mathbb{G}\circ\mathbb{H}=\{g\circ h: g\in\mathbb{G}, h\in\mathbb{H}\}$$

通常有$|\mathbb{X}|>|\mathbb{Y}|>|\mathbb{Z}|$。

嵌套MAC安全的条件： 粗略地讲，如果满足以下两个条件，则可证明嵌套MAC是安全的。

（1）给定一个固定的（未知的）密钥，作为MAC，$(\mathbb{Y},\mathbb{Z},\mathbb{L},\mathbb{H})$是安全的。

（2）给定一个固定的（未知的）密钥，$(\mathbb{X},\mathbb{Y},\mathbb{K},\mathbb{G})$是碰撞稳固的。

直观地说，就是通过一个安全的“小MAC”（即$(\mathbb{Y},\mathbb{Z},\mathbb{L},\mathbb{H})$）和一个碰撞稳固的带密钥的Hash族（即$(\mathbb{X},\mathbb{Y},\mathbb{K},\mathbb{G})$）的复合来构建一个安全的“大MAC”（即嵌套MAC）。

安全消息认证码的构造

安全性实际上是比较对三种Hash族的某种类型的攻击的相对困难性。我们将考虑以下三种攻击者：

- 对嵌套MAC的假冒者，即大MAC的攻击。
- 对小MAC的假冒者，即小MAC的攻击。
- 当密钥是保密的，对Hash函数族的碰撞-探测者，即未知密钥碰撞攻击。

大MAC攻击： 选择并保密一对密钥(K,L)。攻击者允许选择x的值，并查询大MAC喻示器关于$h_L(g_K(x))$的值。然后攻击者试图产生一对(x',z)使得$z=h_L(g_K(x'))$，其中x'从未进行过查询。

小MAC攻击： 选择并保密一对密钥L。攻击者允许选择y的值，并查询小MAC喻示器关于$h_L(y)$的值。然后攻击者试图产生一对(y',z)使得$z=h_L(y')$，其中y'从未进行过查询。

未知密钥碰撞攻击： 选择并保密一对密钥K。攻击者允许选择x的值，并查询Hash喻示器关于$g_K(x)$的值。然后攻击者试图产生一对(x',x'')使得$x'\neq x''$，并且$g_K(x')=g_K(x'')$。

消息认证码的构造

- 假定对随机选择的$g_K \in \mathbb{G}$不存在$(\varepsilon_1, q+1)$未知密钥碰撞攻击。
- 假定对随机选择的$h_L \in \mathbb{H}$不存在(ε_2, q)小MAC攻击，其中L是保密的。
- 假定对随机选择的$(g \circ h)_{(K,L)} \in \mathbb{G} \circ \mathbb{H}$，存在$(\varepsilon, q)$大MAC攻击，其中$(K, L)$是保密的。

由于概率至少为ε，大MAC攻击在向大MAC喻示器最多查询q次后，能输出有效对(x, z)。设$(x_1, ..., x_q)$表示攻击者的查询，又设$z_1, ..., z_q$是喻示器做出的相应回答。在攻击者执行完查询后，可以得到一些列有效对$(x_1, z_1), ..., (x_q, z_q)$以及可能的有效对$(x, z)$。

- 假定现在取出$x_1, .., x_q, x$，向Hash喻示器做$q+1$次查询。可以获得一系列值$y_1 = g_K(x_1), ..., y_q = g_K(x_q), y = g_K(x)$。假定恰巧$y \in \{y_1, ..., y_q\}$，比如说$y = y_i$。则可以得到一对碰撞$(x, x_i)$。这是一次成功的未知密钥碰撞攻击。
- 另一方面，如果$y \notin \{y_1, ..., y_q\}$，则可输出对$(y, z)$，这（可能）是小MAC的有效对。从$q$个小MAC查询中得到$q$个答案后，也就是$(y_1, z_1), ..., (y_q, z_q)$，可构成一个假冒。

消息认证码的构造

- 由假定，任何未知密钥的碰撞攻击最多有ε_1的成功率。而假定大MAC攻击至少有ε的成功率。因此，(x,z)是有效对并且$y \notin \{y_1,\ldots,y_q\}$的概率至少是$\varepsilon-\varepsilon_1$。
- 任何小MAC攻击的成功率最多是ε_2，而上述的小MAC攻击的成功率至少是$\varepsilon-\varepsilon_1$。因此，有$\varepsilon \le \varepsilon_1+\varepsilon_2$。

定理4.9 假定$(\mathbb{X},\mathbb{Z},\mathbb{M},\mathbb{G}\circ\mathbb{H})$是一个嵌套MAC。当密钥$K$是保密的，假定对随机选择的$g_K \in \mathbb{G}$不存在$(\varepsilon_1,q+1)$碰撞攻击。而且，假定对随机选择的$h_L \in \mathbb{H}$不存在$(\varepsilon_2,q)$假冒者，其中$L$是保密的。最后，假定随机选择的$g\circ h_{(K,L)} \in \mathbb{G}\circ\mathbb{H}$，对嵌套MAC存在$(\varepsilon,q)$假冒者。则$\varepsilon \le \varepsilon_1+\varepsilon_2$。

嵌套MAC的实例：HMAC

HMAC是一个于2002年3月被提议作为FIPS标准的嵌套MAC算法。它通过不带密钥的Hash函数来构造MAC。我们基于SHA-1来描述HMAC。

$$HMAC_K(x) = SHA-1((K \oplus opad)||SHA-1((K \oplus ipad)||x))$$

其中K是密钥，x是需要认证的消息，$ipad = 3636...36$，$opad = 5C5C...5C$。

安全消息认证码的构造：CBC

构造一个MAC的最常用的方法之一是基于一个固定的初始化向量的CBC模式。

我们由一个记为IV的初始化向量开始，定义$y_0 = IV$。然后用如下规则构造$y_1, y_2, \ldots$:

$$y_i = e_K(y_{i-1} \oplus x_i) \quad i \geq 1$$

假定$(\mathbb{P}, \mathbb{C}, \mathbb{K}, \mathbb{E}, \mathbb{D})$是一个内嵌式密码体制，其中$\mathbb{P} = \mathbb{C} = \{0,1\}^t$。设$IV$是由$t$个0组成的比特串，又设$K \in \mathbb{K}$为秘密密钥。令$x = x_1||\ldots||x_n$是长度为$tn$的比特串，其中每个$x_i$都是长度为$t$的比特串。计算$CBC - MAC(x, K)$的过程如下：

密码体制4.2 CBC-MAC(x, K)

令$x = x_1||\cdots||x_n$

$IV \leftarrow 00\cdots0$

$y_0 \leftarrow IV$

for $i \leftarrow 1$ to n

　　do $y_i \leftarrow e_K(y_{i-1} \circ x_i)$

return(y_n)

- 基于DES的MAC

描述如下:

被认证消息分成连续的64bit分组:$D_1, D_2, \ldots D_n$(必要时用0填充).使用DES算法E,密钥K,数据认证码计算如下(16<= M <=64):

$$C_1 = E_k(D_1)$$

$$C_2 = E_k(D_2 \oplus C_1)$$

$$\cdots$$

$$C_n = E_k(D_n \oplus C_{n-1})$$

CBC-MAC安全性

对CBC-MAC(x, K)的已知最好的通用攻击是生日攻击。

当基本加密算法满足适合的安全性质时，CBC-MAC是安全的。也就是说，如果某些似乎合理且未证明的关于加密方案随机性的假定是对的，那么CBC-MAC是安全的。

(生日攻击) 令$n \geq 3$是整数。令$x_3, \ldots, x_n$是长度为t的固定比特串。令$q \approx 1.17 \times 2^{t/2}$为整数，选择任意$q$个不同的、长度为$t$的比特串，记为$x_1^1, \ldots, x_1^q$。又设$x_2^1, \ldots, x_2^q$为随机选择的长度为$t$的比特串。
对$1 \leq i \leq q$和$3 \leq k \leq n$，定义$x_k^i = x_k$，并对$1 \leq i \leq q$，定义

$$x^i = x_1^i || \cdots || x_n^i$$

如果$i \neq j$，则$x^i \neq x^j$。

CBC-MAC安全性

- 攻击者请求得到$x^1, x^2, \ldots, x^q$的MAC。在使用密码体制4.2计算x^i的MAC的过程中，得到值$y_0^i, y_1^i, \ldots, y_n^i$，$y_n^i$是$x_i$的MAC。
- 假定x^i和x^j有相同的MAC，即$y_n^i = y_n^j$。注意到$y_n^i = y_n^j$当且仅当$y_2^i = y_2^j$，当且仅当$y_1^i \oplus x_2^i = y_1^j \oplus x_2^j$
- 设x_δ为长度为t的任意比特串。定义

$$v = x_1^i || (x_2^i \oplus x_\delta || \cdots || x_n^i$$

和

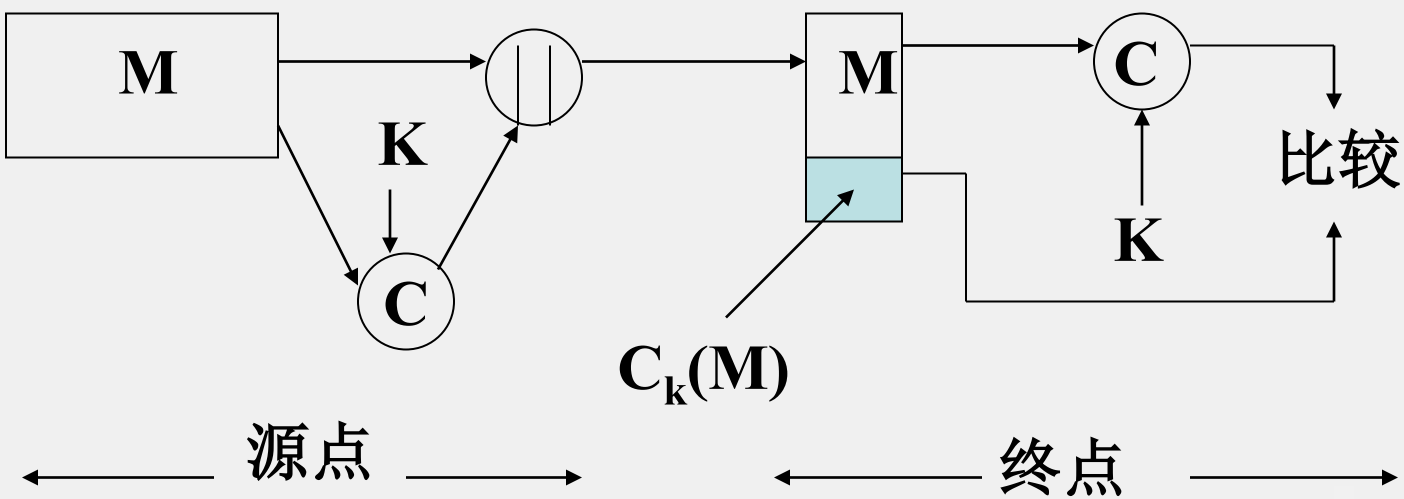
$$w = x_1^j || (x_2^j \oplus x_\delta) || \cdots || x_n^j$$

然后攻击者要求得到v的MAC。不难看出，v和w有相同的MAC。

消息认证码的应用

- MAC的基本用法1
  - A->B: M || C_k(M)

  提供认证, 因仅A和B共享K;

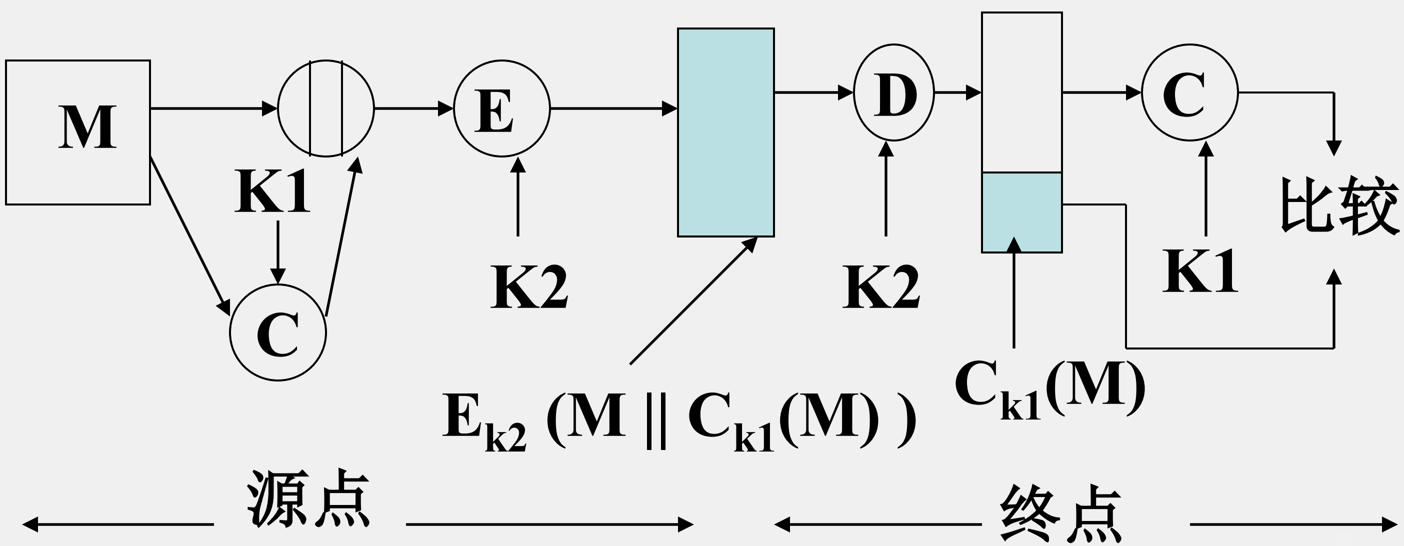


消息认证码的应用

- MAC的基本用法2
  - A->B: $E_{k2}(M \| C_{k1}(M))$

    提供认证, 因仅A和B共享K1;

    提供保密,因仅A和B共享K2;



消息认证码的应用

- MAC的基本用法3
  - A->B: $E_{k2}(M) \| C_{k1}(E_{k2}(M))$

    提供认证, 因仅A和B共享K_1;

    提供保密,因仅A和B共享K_2;