# What did you learn?

> Security Definitions and Proofs

## Number 27: What is the AEAD security definition for symmetric key encryption?

前面（#18）介绍的分组密码的各种模式，本质上是为了机密性（confidentiality）服务的，现实情况中，我们还需要保证完整性（integrity）和认证性（authenticity）。

> **AE** 指的就是 **Authenticated Encryption**，认证加密 **AD** 指的就是 **variable-length Associated Data**，例如 packet header
>
> > For more information on the impact of associated data, see here and here.

为了达到这些目的，引入一个叫做 message authentication code （MAC）的东西，经典的实现方法是利用哈希函数（HMAC）。但是加密和认证这两个原语不能随随便便地组合在一起。为了达到 IND-CCA 安全性，我们必须遵守"先加密后认证"原则（'Encrypt-then-MAC' paradigm）

> Encrypt-and-MAC, MAC-then-Encrypt 是不可以的，英文教材可以参照 **P132** (有证明)

We normally expect authenticity and integrity but not confidentiality from this optional component. For further reading and examples, see Adam Langley's blog on the topic

1. IND-CCA2 (下章)
2. IND-CCA3

   > Shrimpton 2004 年提出来的，解密语言机只能返回无效符号⊥，而不是错误的明文。这个模型更加接近现实，因为在现实中，解密器不会告诉你明文是错误的，而是会告诉你解密失败了。就跟在 AE 中一样，完整性和机密性是分开了的。

3. CCM

   > a combination of a blockcipher in **C**ounter mode with **C**BC-MAC with the **M**AC-then-Encrypt approach。由于需要提前知道信息长度，所以不适用于online，且需要调用两次分组密码，所以在效率上会逊色一些。

4. GCM

   > uses Encrypt-then-**M**AC with a blockcipher in **C**ounter mode and a polynomial-based hash function called **G**HASH. Saarinen 指出，这个方法存在弱密钥。

5. CAESAR

   > Competition for Authenticated Encryption: Security, Applicability, and Robustness AE Zoo

## Number 28: What is the IND-CCA security definition for public key encryption?

IND-CCA security : **Ind**istinguishable **C**hosen **C**iphertext **A**ttack

*model 1* : **find-then-guess** security game

> 敌手允许在第 3，4 步之前与之后访问加密与解密预言机，当然也有第 3,6 步

1. Generate the public and secret keys $(p_k, s_k)$. The adversary A has access to the public key $p_k$
2. Assign $b \leftarrow 0,1$ privately
3. A is allowed to query the decryption oracle $Dec_{sk}$ and the encryption oracle $Enc_{pk}$
4. A then outputs a pair of messages $(m_0, m_1)$
5. We output the encryption $c = Enc_{pk}(m_b)$
6. The adversary is allowed to enquire for more encryptions or decryptions, as in step 3, but he is not allowed to ask for the decryption of $c$
7. A outputs $b' \in 0,1$ . A wins if $b = b'$

我们说 the **advantage** of A is $Adv(A) = 2 \mid Pr[A\ wins] - 1/2 \mid$ . A scheme is said to be IND-CCA secure if the said advantage is negligible.

*model 2* : **real-or-random** security game

> 上一章的 IND-CCA2 就是这个模型

不同点就是在 Step 5, A 不会每次都乖乖加密 $m_b$，而是在 $b = 0$ 的时候返回一个随机消息 $m'$ 的加密, A 必须得区分他是在 “real” 还是 “random” 的世界中， advantage 和安全性的定义是相似的。

这两个定义其实是等价的，对于模型 1 的攻击者 A，我们可以构造一个模型 2 的攻击者 B：

$$Adv_{find-and-guess}(A) = 2 \cdot Adv_{real-and-random}(B)$$

## Number 29: What is the UF-CMA security definition for digital signatures?

在之前的文章中我们介绍了 DSA, Schnorr and RSA-FDH（#16）签名算法的细节 现在我们来看看签名系统及其安全性。

1. 签名系统的定义 签名方案用于证明消息的来源 A signature scheme $S$ is a tuple of algorithms $(KG, Sign, VRFY)$ :
   - $KG$ is a randomised algorithm which outputs a secret key $sk$ and a public key $pk$.
   - $Sign$ is a (possibly) randomised algorithm which on input $sk$ and a message $m$ it outputs a signature $\sigma$
   - $VRFY$ is a deterministic (non-stateful) algorithm which takes in the public key $pk$, a message $m$ and a signature $\sigma$ and returns 1 if $\sigma$ is a signature on $m$ and 0 otherwise
2. UF-CMA security 需要玩下面这个游戏：
   - The game runs $KG$ to get $(p_k, s_k)$
   - The adversary A is given $p_k$ and can then send messages $m_i$ to the game and get back signatures $\sigma_i$ under the secret key $s_k$

- A must output a pair $(m^*, \sigma^*)$ 如果 A $\sigma^*$ 是 $m^*$ 的签名，且 $m^*$ 不是 A 之前询问过的消息，那么我们就说 A 获胜。

UF-CMA security 就是说 A 获胜的概率是可以忽略的。

## Number 30: Roughly outline the BR security definition for key agreement

> 主要是关于 authenticated key exchange 的安全性定义

密钥交换一直是个非常难以解决的问题，光从定义角度看都比简单的加密要难好多，即使是著名的 Diffie-Hellman protocol, 都不能保证认证性。

> 例如存在中间人攻击

为了模型化这些攻击，于是有了两种方法来进行安全性定义：

1. symbolic model 采用形式化的技术来对协议进行建模和分析，例如 BAN logic 。 优点：对于已经存在的攻击，可以很容易地对其建模，可以很好的识别攻击；也可以使用 theorem provers 使证明半自动化 缺点：底层逻辑很难捕捉到所有类别的攻击，所以使用该模型分析出的安全性没那么靠谱
2. computational model 1993年，Bellare and Rogaway 为认证密钥交换提出了在计算模型中的基于游戏的安全性定义，即 BR security definition。非常类似于 IND-CPA 和 IND-CCA。

现在我们不需要考虑系统是不是单纯的牢不可破，而是量化攻击者成功的概率。 所以也不用具体化攻击者的能力，我们直接给一个概念能力： **所有的通信都在敌手的控制之下**

> read, modify, delay, replay, etc. paper原话：The adversary can deliver messages out of order and to unintended recipients, and she can concoct messages of her own choosing. What is more, the adversary can conduct as many sessions as she pleases amongst the players, and she can control, for each, who is attempting to authenticate to whom.

敌手还可以与其他方同时运行任意数量的协议实例。

希望达到的安全性用大白话讲就是：对手让一方接受商定密钥的唯一方法是转发来自真实协议运行的诚实消息，而这样他们不可能学到任何新东西。

> The intuition behind the AKA security game is that the only way an adversary can get a party to accept an agreed key is by forwarding honest messages from a genuine protocol run, in which case they cannot possibly learn anything new.

这个安全性游戏包含很多不同的预言机供敌手查询，三个最主要的：

- corruption oracle
  > allows the adversary to take control of a chosen party
- key registration oracle
  > registers a public key for any chosen user
- message oracle

> **main** oracle used for **passing messages**. Note that messages are not sent directly between the participants, instead the adversary does this using the message oracle (我感觉可以理解为控制通信)

message oracle 是主要的预言机，允许敌手与各方创建协议会话（session），目标是建立短期或临时共享密钥并发送消息。当查询预言机时，可以执行以下操作之一：

- 在两个用户之间启动新会话

  > Start a new session between two users

- 了解任意终止会话的密钥

  > Learn the secret key of any terminated session

- 在现有会话中发送消息并接收响应 （不太理解这有什么用？）

  > Send a message in an existing session and receive the response

这个安全性游戏遵循前面的 real-or-random paradigm

> 选一个 bit $b$, $b=0$ 的话给敌手一个随机的密钥，$b=1$ 的话给敌手一个真实的密钥，敌手的目标就是区分这两种情况

与预言机交互后，敌手选择一个已终止的会话，其中双方都没有被贿赂，并且不存在泄露过密钥的对话（不然就没意思了）。然后会这次会话的挑战密钥。猜对了 $b$ 意味着获胜。

> 泄露密钥的会话我理解的是：1. 这个会话没有使用同样的密钥；2. 这个会话中发的消息中不包含当前的密钥

## Number 31: Number 31: Game Hopping Proof

我们将编写一系列游戏，其中第一个游戏（Game 0）是最初的挑战者，然后每次对挑战者进行小的修改。**相邻的比赛应该没有区别**，但最后一次比赛敌手应该不可能获胜。

本章以 DDH （#11）与 ElGamal 为例:

> DDH: $g^a, g^b, g^{ab}$ 是不可区分的 ElGamal: 私钥 $x$，公钥 $X=g^x$，密文 $(c_1, c_2)=(g^y, MX^y)$；解密 $m=c_2c_1^{-x}$

### $\mathrm{Game}_0$

> (adversary $\mathcal{A}$ plays)

1. $x \xleftarrow{\$} \mathbb{Z}_q, X \leftarrow g^x$
2. $(M_0, M_1) \xleftarrow{\$} \mathcal{A}(X)$

   > $\mathcal{A}$ 拥有公钥并生成一对儿挑战消息 $M_0, M_1$

3. $b \xleftarrow{\$} 0,1$
4. $y \xleftarrow{\$} \mathbb{Z}_q, c_1 \leftarrow g^y, Z \leftarrow X^y, c_2 \leftarrow M_bZ$
5. $b' \leftarrow \mathcal{A}(c_1, c_2)$

6. if $b = b'$ output 1 else 0

如果游戏返回 1，那么我们就说 $\mathcal{A}$ 获胜，而 the **advantage** of $\mathcal{A}$ against the IND-CPA security of ElGamal is:

$$\mathrm{Adv}(\mathcal{A}) = 2|\Pr[\mathcal{A}\ wins\ \mathrm{Game}_0] - 1/2| \tag{1}$$

## $\mathrm{Game}_1$

区别仅在第四步，将 $Z \leftarrow X^y$ 替换为：

$$z \xleftarrow{\$} \mathbb{Z}_q, Z \leftarrow g^z$$

所以新的密文为 $(c_1, c_2) = (g^y, M_b g^z)$，而不是 $(g^y, M_b g^{xy})$

重申，如果游戏返回 1，那么我们就说 $\mathcal{A}$ 获胜。不同之处在于 $Z$ 现在是随机的（因为 $z$ 是随机的），所以 $c_2$ 也是随机的，与 $X, c_1, b$ 均无关。因此 $\mathcal{A}$ 无法从 $c_1, c_2$ 中学到任何关于 $b$ 的信息，所以其获胜的概率是 1/2（只能蒙），即：

$$\Pr[\mathcal{A}\ wins\ \mathrm{Game}_1] = 1/2 \tag{2}$$

## Link

发现前后仅仅是将 $Z$ 从 $g^{xy}$ 替换为 $g^z$，怎么把这个和 DDH问题 联系到一起就很显而易见了~ 为了将这种联系显示的更清楚一些，我们通过敌手 $\mathcal{A}$ 构建敌手 $\mathcal{B}$ 来解决 DDH 问题：

1. 对于输入 $X, Y, Z$，将 $X$ 作为输入运行 $\mathcal{A}$ 并一对儿挑战消息 $(M_0, M_1)$
2. 随机选择一个 bit $b \xleftarrow{\$} 0, 1$，并计算 $M_b Z$
3. 将 "密文" $(Y, M_b Z)$ 发送给 $\mathcal{A}$，并接受输出 $b'$
4. 如果 $b = b'$，则猜测 $Z = g^{xy}$并返回 1； 反之则猜测 $Z$ 为随机数并返回 0

如果给 $\mathcal{B}$ 的是一个真正的 DH 三元组 $(g^x, g^y, g^{xy})$，那么上述就是对 $\mathcal{A}$ 运行$\mathrm{Game}_0$的模拟 如果给 $\mathcal{B}$ 的是一个假的三元组 $(g^x, g^y, g^z)$，那么上述就是对 $\mathcal{A}$ 运行$\mathrm{Game}_1$的模拟

因此，$\mathcal{A}$ 运行 $\mathrm{Game}_0$ 获胜 和运行 $\mathrm{Game}_1$ 获胜的区别恰好就是给 $\mathcal{B}$ 输入 $(g^x, g^y, g^{xy})$ 时输出 1 和给 $\mathcal{B}$ 输入 $(g^x, g^y, g^z)$ 时输出 1 的区别。 这个区别也就是 the advantage of $\mathcal{B}$ against DDH.

结合 $(1)$ 和 $(2)$，我们就得到 **the advantage of $\mathcal{A}$ against the IND-CPA security of ElGamal is no greater than the advantage of $\mathcal{B}$ against DDH**.

如果 DDH 问题对于所有多项式时间对手来说都是困难的话（意味着他们的优势可以忽略不计），那么 ElGamal 也肯定是 IND-CPA 安全的。

> 很像规约证明啊（就是一个东西吧）

# Number 32: difference between game-based and simulation-based security definitions

## game-based

字面意思，通过游戏进行安全性定义

1. 游戏围绕着一些通用**原语**展开，通常是敌手和挑战者玩，其中挑战者向对手提出挑战，并牢记某个 “目标”。 敌手往往也有一些特殊的能力，例如可以访问若干预言机，如果其实现 “目标” 则 “获胜”。

   > “获胜”通常意味着敌手根据挑战者提出的挑战来提供了某些 “正确” 的输出

2. 敌手的 优势（advantage） 则被量化为一个数字，代表着敌手能比盲猜 “更好” 多少

   > 一个由该通用原语实例化的密码学系统，如果所有的 PPT 敌手对其的优势都是可以忽略不计的，那么我们就说这个密码学系统满足该安全性定义。

3. 一般情况下，可以将挑战者视为合法用户，将敌手视为想要违背合法用户意愿实现某些目标的 bad guy；挑战者有权访问所有秘密参数（比如私钥），敌手则只能访问一些预言机（比如公钥与哈希函数）与挑战者给的东西（比如公共参数和挑战本身）

4. 这种安全性证明包含两个重要概念：

   - **reduction**：将安全性与计算困难问题联系起来

     > 'if an adversary wins the game with non-negligible advantage, it is possible to construct an algorithm that uses the adversary as a subroutine to solve some hard problem efficiently.'

   - **game hopping** (#31)

5. 前面五篇有四篇是基于游戏的安全性定义，还有一个是基于一系列游戏的证明

### simulation-based

安全性是根据模拟器的存在和一些理想的“功能”来定义的。

考虑一个现实世界中的密码学方案，然后想象一下我们希望该方案在理想世界中如何表现？ 就拿投票系统举例： 我们希望有一个可信的第三方，可以给所有投票者提供一个安全信道，投票通过安全信道汇总然后公布结果，除此之外不暴露任何信息。

**如果对于现实世界中所有敌手，存在一个模拟器，在面对理想世界中的理想“功能”时，可以提供与这些现实敌手相同的输出，那么我们就说该密码学方案是安全的。**

> A cryptographic scheme is now secure if, for any adversary against this scheme in the real world, there exists a simulator that provides the same output as the adversary in the real world, while interacting with the ideal 'functionality' in the ideal world.

因为这意味着现实世界中任何可能的“攻击”都可以应用于理想世界中的理想功能。相反，如果理想功能能够抵御理想世界中的攻击，那么真实方案也能够抵御现实世界中的这些攻击。

> 这个概念最初出现于 1987年 一篇关于多方联合计算的文章中。它说可以玩任何游戏（我觉得因为和游戏本身没关系了，所以不是game-based），不管在游戏的哪一步，反正只要参与人数少于总人数的一半，那么就不可能学到任何新知识。**现在这个概念也常常用于 MPC。**

### difference

在基于游戏的方法中，每个安全概念都有自己的游戏。如果这个概念能够正确捕获或模拟了我们希望系统具有的现实世界的属性，那么这个概念就定义好了。如果我们的方案需要满足不同的安全概念，我们就需要为每一

个概念都进行对应的游戏。

> 感觉就像中药抓药一样，要哪个就抓哪个。当然，药材之间也存在上下级关系，例如 IND-CCA 隐含着 IND-CPA 。

相反，在基于模拟的方法中，安全性是通过理想功能（functionality）来建模的。我们的方案只要不被攻击破坏功能就可以了！这意味着该模型捕获了不同的安全概念。

> 就像看病一样，别管我怎么看的，求神拜佛还是周易算卦还是吃药手术，反正病好了就行！

p.s. : More and More