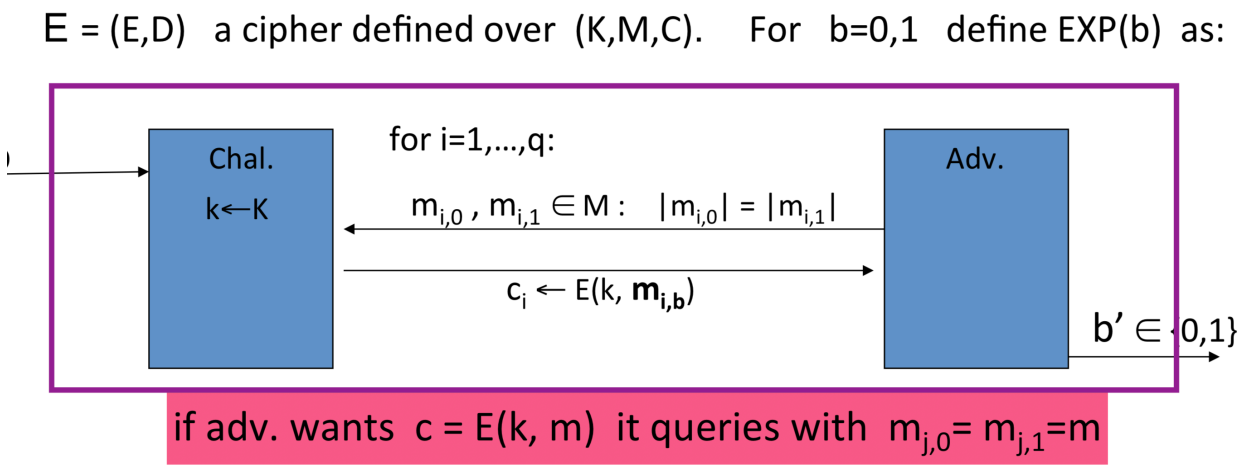
安全级别 (由上到下, 逐渐增强)

Semantic Security for one time key

- 不能有查询的步骤
- 只能直接给挑战者一个 m0, m1 , 然后立马猜实验为0, 1 的概率

Semantic Security for many time key CPA (Chosen - plaintext attack)

- Adv 可以给Chl 多次明文, 而获得返回的密文
- 攻击Game的定义 :
  - 攻击者可以先询问挑战者任意次数的加密密文 , 且每一次询问的格式为 (m0,m1) 返回 (c0, c1)
  - 如果攻击者想知道某个消息的加密密文, 则只需要提交的明文为(m0,m1) and m0 = m1
  - 开始攻击时, 最后一次提交一对(m0,m1)然后挑战者随机返回 C_i i属于0, 1
  - 则此时便有了安全的定义 : 即如果E为安全, 则对于返回的 C_i 挑战着能猜出为是明文m0, 或m1的概率应该相同, 即 P(guss(m0)) - P(guss(m1)) 是可以忽略的 .
- **如果相同密文下返回相同的明文, 则一定不是CCA安全** , 要做到CCA级别的安全, 必须要相同的密文返回不同的密文, 这就需要在加密的时候, 加入随机变量, 或者噪声等 .

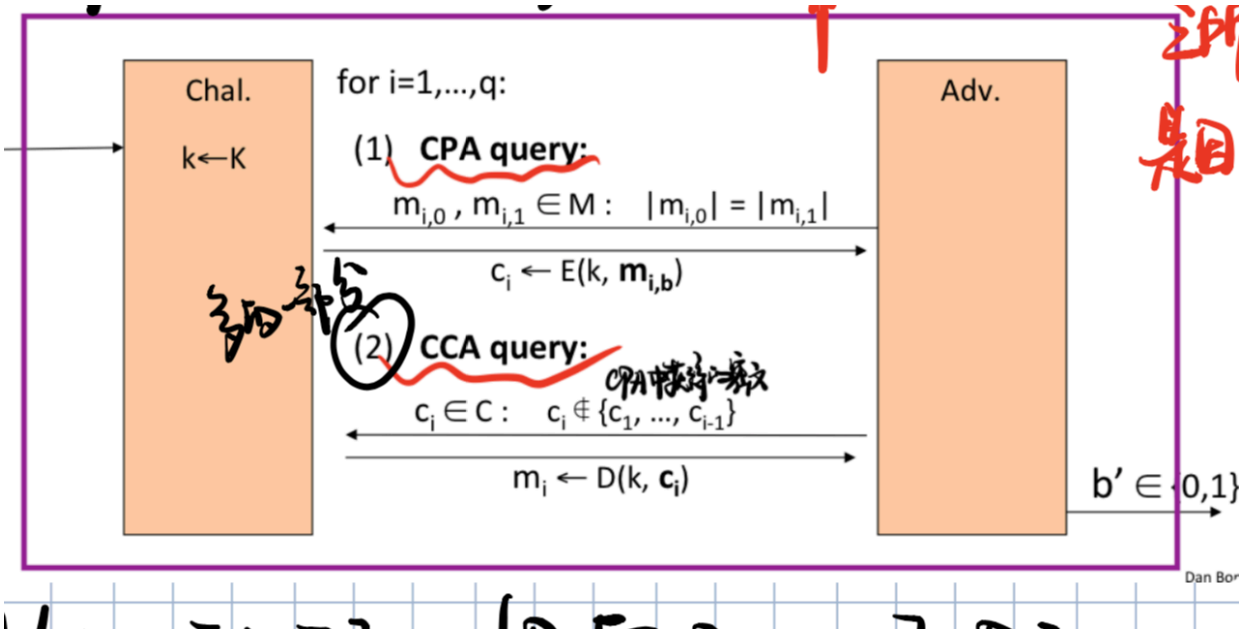


Semantic Security for many time key CCA (Chosen - CipherText attack)

- CCA (Chosen - CipherText attack) : 其中满足CCA的加密模式, 肯定是已经满足了CPA
  - Adv 可以给Chl 多次明文, 而获得返回的密文
  - Adv 可以给Chl 多次密文, 而获得返回的明文

- 因此 CCA 包含 CPA

- CCA1



- CCA2
  - 选择密文查询. 接着提交挑战
  - **在提交明文挑战之后, 还可以进行密文查询** 这是个CCA1的主要区别