教科书式 RSA 方案面临的攻击及防御措施

关于 RSA

加密过程

- 选择一对不相等且足够大的质数，例如：选 z1 和 z2 为较大的两个质数
- 计算 z_1 和 z_2 的乘积，$n = z_1 \times z_2$
- 计算 n 的欧拉函数 $\phi(n) = (z_1 - 1) * (z_2 - 1)$
- 选一个与 $\phi(n)$ 互质的整数 e，且 $e < \phi(n)$
- 算出 e 对于 $\phi(n)$ 的模反元素 d
- 公钥：$K_U = (e, n)$，e 和 n 是成对的，共同组成公钥
- 私钥：$K_R = (d, n)$，同上

RSA 的使用方法

明文M加密：$M^e \mod n = C$

密文C解密：$C^d \mod n = M$

一、直接分解攻击

介绍

这种方式类似于暴力攻击，因为 RSA 的安全性就是来自大素数乘积分解的困难性，如果 n 比较小的话，我们可以通过暴力穷举的方式进行攻击，也就是把小于 n 的这些素数都试一遍，但如果 n 比较大的话这种方法就十分困难了。推荐一个网站可以在线分解 http://factordb.com/

这就与破解 md5 的方式是一样的，其实就是把之前破解好的一些密码给储存起来放到一个数据库中，逆向时直接进行查询。

防御方法

只需要把 n 给的特别大就可以了，一般来说 2048 bit 的 n 就会被认为是特别安全的，现在一般公钥都是用 4096 bit 的，这种情况下其实很难进行分解。

二、利用公约数分解 n

介绍

如果我们得到了 n1 和 n2，而且 n1 和 n2 拥有两个相同的因子的话，我们就可以使用这种攻击方式来进行一个计算。

计算方法很简单，我们可以使用欧几里得 gcd 算法来直接计算 n1 和 n2 的最大公约数。欧几里得算法的复杂度是 log(n)，复杂度是比较低的，因而计算较快，是一种比较高效的计算方法。

防御方法

这种情况其实是比较罕见的，一般很少会出现拥有公约数的情况，事实上避免这种攻击的方法也十分的简单，选的 n 较大，然后使用程序自动生成一个 n，这种情况下出现有公约数的概率微乎其微。

三、共模攻击

介绍

如果使用了相同的模数对一段相同的密文进行了加密，就能通过共模攻击还原出明文 m 的值，当然这种情况是不需要分解 m 的。基本的原理如下。

首先通过相同的 n 对相同的 m 进行加密：

$c_1 \equiv m^{e_1} \mod n$

$c_2 \equiv m^{e_2} \mod n$

首先，两个加密的指数是互质的，那么就是 $gcd(e_1, e_2) = 1$，根据欧几里得算法可以得到一组一正一负的解(s_1, s_2)，使得 $e_1 s_1 + e_2 s_2 = 1$，然后带入化简就可以得到：$c_1^{s_1} c_2^{s_2} \equiv m \mod n$

防御方法

只要避免多个人使用同一个模数即可，这样攻击者是无法推导明文的。

四、低指数攻击

介绍

在 RSA 中 e 也被称为加密指数，e 是可以随便选取的，一般来说选小一点的 e 可以使加密更加迅速，但是如果太小的话就可能造成比较严重的安全问题。

防御方法

把指数设置的大一点，比如一个5位数或者4位数，这种情况下黑客使用逆运算将十分困难。

五、Hastad 攻击

介绍

如果使用的是一个比较小的指数进行加密，并且还把加密的消息 mod 不同的 n 后发给了其他的多个人，这种情况下我们就可以通过低加密指数广播攻击来进行攻击。

假设 e 比较小，而且 n_1, n_2, n_3 是互素的，我们可以使用中国剩余定理来求得 m^e。

防御方法

将 e 的取值范围设的更广一点, 这样黑客很难猜到 e。

六、Fermat攻击

介绍

针对大整数分解的算法之一，Fermat 方法适用于 p, q 相差不大的时候。

防御方法

可以将 p, q 的大小设的不要太接近，这样可以使分解 n 的难度加大。

七、Pollard rho 方法攻击

介绍

针对大整数分解的算法之一，Pollard-Rho 方法适用于 p, q 相差比较大的时候。

防御方法

可以将 p, q 的大小差别不要太大，这样可以使分解 n 花费的时间更长。