W6 11-5 Is RSA a one-way function?

1、Is RSA a one-way permutation?

RSA真的是一个单向置换吗?

攻击者已知公钥(e, N)和x^e，期望找到x，则这个过程有多难？即计算模N下的e次根有多难?

如果说真的很棘手的话，RSA就是一个单向函数，如果很容易则RSA就算被破解了

事实上目前已知最好的算法是分解大整数N=pq，然后分别计算p和q的e次根，再通过中国剩余定理将结果恢复成模N的e次根，第二步很简单，棘手的点在于分解N

2、Shortcuts?

那问题又来了，一定要分解N吗？还是说有没有捷径可以不分解N就计算模N的e次根?

事实上可以将算法进行归约，如果存在一个高效的算法可以计算模N的e次方根，可以证明的是，这个算法可以被归约为一个因子分解的算法

这个证明意味着给定一个计算N模下的e次根的算法，同时也就有了一个因子分解的算法，也就是对N模下的e次根的计算不可能比对N进行因式分解更快，从而证明了破解RSA的难度与因子分解的难度是等价的

因子分解问题目前还尚未解决，事实上这是公钥加密算法中最古老的问题之一，看下面这个例子:

假设有一个算法可以计算模N下的立方根（即e=3），则能否证明利用该算法可以求出N的因子？结论未知

还是上面这个假设，考虑e=2的情况，即该算法可以计算模N下的平方根时，则该算法可以用于对N的因子分解，即计算模N下的平方根蕴含着因子分解的算法，二者难度相当

但是由于RSA的定义，$e \cdot d \equiv 1 \pmod{\varphi(N)}$，因此e一定与φ(N)是互素的，又由于p和q都是大素数，$\varphi(N)=(p-1)(q-1)$一定是个偶数，因此e和φ(N)一定不是互素的，从而即便是我们有这个优秀的算法，e=2也绝不可能作为RSA的密钥参数（事实上合法的最小RSA指数为3）

3、How not to improve RSA’s performance

就目前所知，RSA是一个单向函数，想要破解即通过计算模N下的e次方根，就需要分解N

人们也做了许多工作，试图改善RSA的加解密算法性能，但大多失败了

比如说优化RSA的解密算法，由于解密算法为计算$c^d \equiv m \pmod N$，并且该算法的复杂度与d的量级成正比，即复杂度为$O(\log(d))$，为了加速RSA解密为什么不用一个量级很小的d？如$d \approx 2^{128}$之类的

显然128 bits对于穷举搜索不太现实，但一般而言，解密指数d会和模数的大小差不多，即2000 bits左右，若使用128 bits的d可以将解密速度提高20倍左右，但是这个做法非常糟糕

Michael Wiener提出一种攻击方式，只要私钥指数$d<N^{1/4}$，则RSA完全不可靠（而且是最糟糕的那种不可靠），即如果使用这种不安全的指数d，利用公钥(e,N)可以很快找到私钥d

那上述攻击仅针对512 bits以下的d，那么要是把d设置得高一点呢？实际上仍然不可靠，有一种Wiener攻击的拓展形式，表明$d<N^{0.292}$，则RSA也是不可靠的，且针对该结论有人推测对于任何$d<N^{0.5}$，该结论都成立，意思就是即便是d接近$N^{0.4999}$也是不可靠的（但该推测仍然是个开放性话题，仍然具有争议性）

> RSA不可靠：在只给定公钥e和N的情况下可以恢复私钥d

为什么是0.292?

$$0.292 \approx 1 - \frac{1}{\sqrt{2}}$$

因此结论是：不应改变d的结构来优化RSA

也有不少实验结果表明，通过此类优化RSA的小伎俩最终都导致了灾难，因此这不是一个优化RSA的正确方式

4、Wiener's attack

我们期望恢复私钥d，且已知$d<N^{1/4}$，假设$d<N^{0.25}/3$（3不重要，1/4起决定作用），然后看看如何实施攻击

首先，我们有e·d ≡ 1 (mod φ(N))，意味着存在整数k，使得e·d = k·φ(N) + 1，等式两侧同除以d·φ(N)，移项之后得到如下等式

$$|\frac{e}{\varphi(N)} - \frac{k}{d}| = \frac{1}{d \cdot \varphi(N)} \leq \frac{1}{\sqrt{N}} \tag{1}$$

观察上述等式的左侧，e为已知，但φ(N)未知，因此不知道e/φ(N)，但是由于φ(N)与N非常接近，因此可以用N近似代替φ(N)，则e/φ(N)近似于e/N

回想一下欧拉φ函数

$$\varphi(N) = (p-1)(q-1) = N - p - q + 1$$

变形一下，去掉常数1，可以得到一个不等式

$$|N - \varphi(N)| \leq p + q$$

而p和q的大小都与√N差不多，所以上述不等式可以近似于如下不等式

$$|N - \varphi(N)| \leq p + q \leq 3\sqrt{N} \tag{2}$$

然后看回初始的假设

$$d \leq \sqrt[4]{N}/3$$

将该不等式两侧平方，然后取倒数，再乘1/2，得

$$\frac{1}{2d^2} - \frac{1}{\sqrt{N}} \geq \frac{3}{\sqrt{N}} \tag{3}$$

然后看我们需要的式子（下式不等号左侧），用三角不等式展开如下

$$|\frac{e}{N} - \frac{k}{d}| \leq |\frac{e}{N} - \frac{e}{\varphi(N)}| + |\frac{e}{\varphi(N)} - \frac{k}{d}| \tag{4}$$

对于不等号右侧的第二个绝对值上面（1）已经推出来了，第一个绝对值做通分，将（2）带入分子，化简得到

$$|\frac{e}{N} - \frac{e}{\varphi(N)}| = |\frac{e(\varphi(N) - N)}{N \cdot \varphi(N)}| \leq \frac{e \cdot 3\sqrt{N}}{N \cdot \varphi(N)} \leq \frac{3}{\sqrt{N}} \tag{5}$$

注意到（5）中第一个不等号去掉了分子的e和分母的φ(N)，由于e很小且φ(N)很大，实际上是将该表达式放大了，然后约去分子分母的√N，得到最右侧结果

然后再将（1）（3）（5）带入（4），得到最后的不等式

$$|\frac{e}{N}-\frac{k}{d}| \le |\frac{e}{N}-\frac{e}{\varphi(N)}|+|\frac{e}{\varphi(N)}-\frac{k}{d}| \le \frac{1}{2d^2}-\frac{1}{\sqrt{N}}+\frac{1}{\sqrt{N}} \le \frac{1}{2d^2}$$

观察这个不等式，我们知道e/N的值，我们期望知道k/d，由不等式可知这两者的差非常小，而出现这种情况的值非常少，只有少数的k和d才能出现这种情况

事实上这样的分式的个数近似于log(N)，因此存在一个连续性分式算法，可以从e/N推导出log(N)种可能的k/d，然后逐一尝试直到找到正确的k/d

由于e·d = k·φ(N) + 1，因此e·d ≡ 1 (mod k)，故d与k互素，如果使用有理分式来表示k/d的话，则其分母一定是d，即log(N)种可能的结果中肯定有一个分母是d

> 结论：如果私钥指数d非常小，小于1/$N^{0.25}$，那么可以很容易恢复d的值