Tales From the Cryptography

by Stephen Turbek

RAZORFISH REPORTS

a razorfish publication | 16 january 2001 | report 046

©2000 RAZORFISH, INC. RAZORFISH IS A REGISTERED TRADEMARK OF RAZORFISH, INC. AND THE RAZORFISH GEAR LOGO IS A SERVICE MARK OF RAZORFISH, INC. ALL RIGHTS RESERVED.

Humanity holds a long and deep relationship with secrets. Significant amounts of energy have gone into keeping messages secret. Significant effort has also gone into making those messages readable. This fascinating history of the 'arms race' between code makers and code breakers shows the development of ingenious methods of concealment.

Public Key encryption is as necessary as the browser for a useful Internet.

So far, using encryption has not affected the average person that much, but the emergence of the 'information economy' requires us to take control of our information. Be it the hundred dollar phone bill payment by the consumer, or the million dollar transfer of funds by a corporation; these equally important transactions –occurring more and more over the public Internet– require us to have secure communication.

A bit of history

Cryptography, or the study and use of codes, is several thousand years old. The earliest ciphers (the term for the encoding system) used in Caesar's time were shift ciphers, where all the letters were simply shifted a number of letters up or down. If "A" is shifted to "E" then "ACE" becomes "EGI". The biggest class of basic ciphers is the letter substitution cipher, where one substitutes one letter for another, using "A" for "T". These are quite difficult to crack.

Stephen Turbek (stephen@razorfish.com) is in the razorfish science department.

Frequency Analysis

Letters, it turns out have behaviors. With shift and substitution ciphers, one can often crack them by analyzing the frequency of letters in the message. In English, as well as most other languages, certain letters occur more frequently than others. The most common letters in English, in order, are E T A O I N S H R D L U.

If one counts the frequency of the letters in the encoded message, and substitutes them for their equivalent frequency in plain text, one can often get a good start deciphering. The letters "t" and "h" like to be close to each other, and "q" rarely exists without "u". A variety of ciphers were cracked this way.

Mechanical Cryptography

Before the 1900's, ciphers were encoded and decoded by hand, a laborious process that limited the complexity of the ciphers. Encryption machines made complex encryption feasible. The famous "Enigma", created in Germany before the Second World War, did a very complicated form of the substitution cipher, using multiple substitutions. It was finally cracked by Alan Turing and a group of British mathematicians and linguists, using specially constructed analysis devices (which led to the development of computers). Understanding of the Nazi communication was essential for the Allied forces to win the war.

An Enigma encoder / decoder

The number crunching power of the computer was immediately harnessed for code making and breaking. Multiple ciphers were developed. The United States finally settled on the "Data Encryption Standard" or DES, which it used for many years. It involved translating the message into binary form, then performing a number of mathematical functions on it to convert the sequence into seemingly random numbers. The key is also a number, and an understanding of the encoding algorithm. Cracking these codes is often as simple as finding the right number, unfortunately there are so many that this process takes centuries.

Key Exchange

Unfortunately, these systems all depend on both parties knowing the "key" to the code. If two parties can't trust the communication medium (mail, phone, Internet) they certainly can't send the key over it. Key exchange used to involve trusted couriers flying between cities with attaché cases handcuffed to them, obviously impractical for anyone except governments and arch-villains.

If two people who don't know each other are going to communicate, over a public medium, how do they exchange the code "keys"? E-commerce would never happen if we had to physically mail a code key to each company we bought from.

A One-Way Road

Many scientists struggled with the problem of key distribution and in 1975, Whitfield Diffie and Martin Hellman at Stanford University had an important insight. They realized that if there were a way to encode that was different than decoding it, one could just give out the algorithm of how to encode without anyone knowing the way to undo it. In 1977, Ron Rivest, Adi Shamir, and Leonard Adleman (the R, S, & A in RSA encryption), mathematicians at MIT, were the first to develop such a one-way algorithm. Known as "public key" encryption, it is used as the encryption standard for Web browsers, among many other things.

Rivest, Shamir, Adleman

Going Public

"Public key" encryption generates two keys, one "public" one for encoding the message, the other, "private", for decoding it. When two people want to communicate, they exchange their public keys and each encodes the message going to the other person. The genius of this system is that the message can't be decoded even by the people who encoded the message.

```
114 381 625 757 888 867 669
235 779 967 146 612 010 218
296 721 242 362 562 561 842
935 706 935 245 733 897 830
597 123 563 958 705 058 989
075 147 599 290 026 879 543
541 381 625 757 888 867 667
```

A short snippet of a Public Key

How it Works

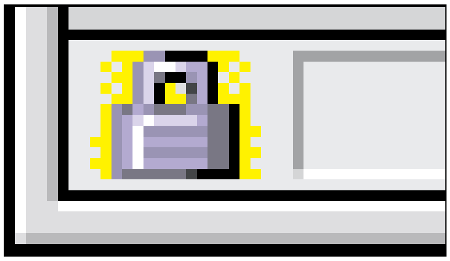
The math of public key encryption is pretty complicated, but it is based upon the fact that it is difficult to take a number and figure out which two numbers were multiplied to get it, called factoring. For example, it is easy to multiply 31 and 23 to get 713, but time consuming to work backward to figuring out the factors of 713. Factoring turns out to be a very computationally intensive process, especially with numbers that are 308 digits long.

The Math

The user's software picks two very large prime numbers (p and q), multiplies them together to get (n) and publishes this, along with a secondary number (s), as their public key. A message (m) written in binary form is encoded (c) using the formula $c = ms$ (mod n). A second formula is used to decode the message, $m = cx$ (mod n), where x is the private key, also created with the p and q. Modular equations are very difficult to reverse, so without p or q, a code breaker is forced to try a fantastically huge number of numbers to find it.

As this encoding and decoding is pretty mathematically intensive and therefore slow, Phil Zimmerman's Pretty Good Privacy took the step of using a traditional encryption, like DES to encode the message and just using public key encryption to encode the DES key. The result is so powerful that the United States tried to limit the export of this technology to prevent terrorists and criminals from using it to foil their wiretaps.

Public key encryption is not merely an interesting technology for spies and mathematicians, but a fundamental requirement for the success of the Internet. Integrated into browsers, online shoppers are unknowingly using the most secure encryption ever designed every time that little lock icon is locked.



Public Key encryption in action

for more information

The Code Book by Simon Singh

Cryptography FAQ
http://www.faqs.org/faqs/cryptography-faq/part05

Fun code breaking game
http://raphael.math.uic.edu/~jeremy/crypt/cgi-bin/magic_update.cgi

Cryptography Resources
http://www.geocities.com/SiliconValley/Bay/9648/pgut-links.html

Pretty Good Privacy
http://www.pgpi.org

RSA Security
http://rsa.com

razorfish reports are published regularly by razorfish for our colleagues and the public.