CRYPTOGRAPHY: FROM AMAZON TO THE CIA

Romil A. Sirohi

Pacific Cascade Middle School

"*November 6, 2011 -- Tongan hackers divert 2M Pa'anga (1.2M USD) to Fiji bank accounts*"

– Daily Nuku'alofa Times, unknown date and time

Mathematics drives myriad aspects of everyday, modern life, and even the Tongan hackers recognize this. While the above newspaper headline is false, if well versed in mathematics, the Tongan hackers could potentially carry this out. Ever since I was eight, the career path of computer science has been most meaningful to me. Math is an integral part of a computer scientist's life and is interwoven into many aspect of computer science, one of which is cryptography.

Cryptography is at the heart of many bank functions, including online payment systems – systems such as PayPal or Amazon Payments that you and I use for online purchases. We have to ensure that transaction-related messages are confidential – otherwise, vital information such as credit card numbers, pins, etc. may be leaked—and cryptographic algorithms enable this.

In typical cryptography, some sort of authentication algorithm that takes predetermined keys is required. However, the high volume of consumer traffic means keeping millions of keys at a time is unfeasible. That's when public-key cryptography comes into use. Public-key cryptography is a branch of cryptography in which two keys are created immediately on request: one public (to encrypt) and one private (to decrypt).

An algorithm called RSA is often used to guarantee confidentiality (e.g. no credit card numbers leaked) in public-key cryptography. Like much of cryptography, RSA works in binary, so we assume the message, m, is converted into its binary ASCII representation. In RSA, after the user (i.e. person wishing to make an online payment) requests to send a message (i.e. details of the payment), the server (i.e. bank) chooses two distinct primes, p and q. The server then multiplies the two primes and sends this number, n, to the user, ensuring that n is larger than m for most messages expected. The server trivially computes the Euler totient function value of n – trivial because n is the product of two primes –$\varphi(\mathrm{n}) = (\mathrm{p}-1)(\mathrm{q}-1)$. This is because, for any prime p, $\varphi(\mathrm{p}) = p - 1$, and the totient function is multiplicative. The server chooses any integer e coprime to $\varphi(\mathrm{n})$ exclusively between 1 and $\varphi(\mathrm{n})$, sending e to the user. The server then computes $d = e^{-1} \bmod \varphi(\mathrm{n})$ (usually using the Extended Euclidean Algorithm), and the user then computes $c = m^e\ mod\ n$, sending that to the server. The server decrypts by computing $m = c^d \bmod \varphi(\mathrm{n})$. A user has successfully transmitted a message with confidential information safely to a server. An attacker could not find the message because he doesn't know d, as it has to be computed $\mathrm{mod}\ \varphi(\mathrm{n})$, and computing $\varphi(\mathrm{n})$ is as hard as factoring if the prime factors are not known.

The next application of cryptography is safely protecting passwords in a database. When users log-on to Facebook or email (or even the CIA), a server has to check a confidential file listing all users and their passwords. This file in the hands of an imposter would be disastrous. Only the owner of the file should be allowed access to the file through a password, converted to a binary key through ASCII. The file can be encrypted and decrypted with the same key, as no information is being transferred. This uses a block cipher-- DES for instance. A block cipher takes an input and outputs a pseudo-random string as a result. DES is computed via a 16-round Feistel Network. In a Feistel Network, the plaintext (e.g. 64 bits file contents) to be encrypted is broken into $L_0\ and\ R_0$. For each round, n, from 0 to p (p = 16, for DES), compute $L_{n+1} = R_n$ and $R_{n+1} = L_n\ \oplus f(R_n,\ K_n)$. Finally L_{p+1} and R_{p+1} are concatenated. In DES, the function *f(a, b)* is calculated first by expanding *a* to 32-bits to 48-bits and XORing that with a subkey, as determined by a key schedule, a pre-determined algorithm for distributing keys. This XORed quantity is split into 8 strings of length 6-bits. Each of these strings is fed into many different S-Boxes, a tool like a hash-map that matches 6-bit strings to 4-bit strings by matching the outer and inner digits of the 6-bit string to a predetermined value. The results from the S-Box are concatenated together to create a 32-bit string, which is then permuted by a predetermined algorithm. This output is returned as the "encryption" of the file. To decrypt, compute $R_n = L_{n+1}, L_n = R_{n+1} \oplus f(L_{n+1}, K_n). L_0 \; and \; R_0,$ concatenated, is the plaintext. Notice f does not have to be invertible. The file contents (where the passwords are stored) are the plaintext, which is fed into DES and the result is the ciphertext, which is incomprehensible to anybody trying to find passwords.

Learning math allows for a more intuitive understanding of these algorithms, which would enable cryptologists to find attacks on existing algorithms before attackers find these attacks and use them malevolently. Furthermore, after attacks have been found, cryptologists with a background in math have the potential to create better algorithms, still based on mathematical concepts. We have seen Euler's totient function, the problem of factoring, modular arithmetic, the Extended Euclidean Algorithm, the Feistel Network function construction, the XOR operation, evaluation of problem complexity to ensure hardness, and much more. All of this is mathematics, and these are the building blocks of cryptography – and all new algorithms will have this as a base. Therefore understanding cryptography—and perhaps more importantly being an innovator in the field—requires a solid knowledge of math. Naturally, the Tongans learned their math well.

References

OpenCourseOnline. (2012, April 16). *3 - 2 - The Data Encryption Standard -Cryptography-Professor Dan Boneh* [Video file]. Retrieved from http://www.youtube.com/watch?v=UgFoqxKY7cY

OpenCourseOnline. (2012, May 9). *11 3 The RSA trapdoor permutation-Cryptography-Professor Dan Boneh* [Video file]. Retrieved from http://www.youtube.com/watch?v=n9cNR9B-MV8