Volume 1, Issue 3 | 5 November, 2018

Three Things you should know about: Cryptographic Algorithms

In the previous issue, we looked at cryptographic hash functions, a special class of cryptographic algorithms designed for one-way encryption. Reversible algorithms can perform both encryption and decryption. These algorithms employ a known value, or key, to convert a message to a cipher and then back to the original source.

Symmetric Key Algorithms use the same key (or a transformed version of the key) for encryption and decryption. This approach depends on both the sender and the recipient having the secret encryption key. The Advanced Encryption Standard, or AES, is the most commonly used symmetric key algorithm. AES uses either a 128-, 192-, or 256-bit key, which is used for encryption and decryption. It's considered susceptible only to brute-force attacks, where all possible 128-, 192-, and 256-bit combinations are tried. That's 10^77 different combinations; for comparison, it's estimated that there are 10^67 atoms in the universe!

Asymmetric Key Algorithms, also known as **public key algorithms,** use two unrelated keys, one public and one private. The public and private keys work together, and as the name implies, only the public key is visible to other users. To send an encrypted message, you encrypt your message with the recipient's public key and they decrypt it with their private key. To send a digital signature, you first hash your message using a cryptographic hashing algorithm, then encrypt the hash with your private key, and the recipient can decrypt with your public key to confirm that the message has not been altered. The most commonly used asymmetric key algorithm is RSA (Rivest–Shamir–Adleman, named for the creators). RSA is widely used for secure communications, and is the basis for transport layer security (TLS). Since RSA algorithm is slower than symmetric encryption, TLS uses a combination of these two methods: your browser and the server use RSA (or another asymmetric algorithm, such as elliptic curve) to negotiate a secret symmetric key, which is then used to encrypt all of the data for that session.

Post-quantum cryptography is an area of mathematical and cryptographical research that focuses on developing new cryptographic algorithms that can withstand attacks by future quantum computers. The security of RSA depends on the practical difficulty of factoring numbers that are the product of large prime numbers; because quantum computing promises a 10^8-fold increase in computing speed, RSA will not remain indefinitely secure. It's widely disputed as to how long it will be before quantum computers capable of breaking RSA will be available. Expert predictions may range from years to decades, but the consensus is that it is inevitable. Current research in areas such as lattice-based algorithms shows promise in developing public key algorithms that can be considered secure even against attacks by quantum computers. Numerous theoretical algorithms already exist that had previously not been practical for encryption because of the computing power required for encryption/decryption. Fortunately, it's not just bad actors who now have more computing power, so algorithms that were previously impractical will likely be part of new standards.