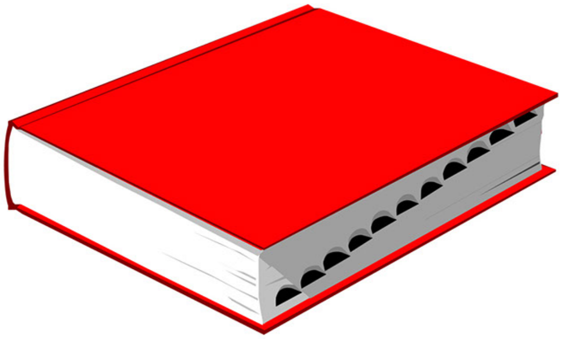
The Dictionary About (Almost) Everything

Piraveen Perinparajan

January 2017



Preface

This is a document that is meant to be understood by everyone. Ambitious? *indeed.* The idea is to provide definitions for common words and terms in software engineering, cryptography, mathematics and physics, in Layman's terms. Currently the main focus is software engineering and cryptography.

Especially for beginners, the different terms can be overwhelming. This document seeks to solve this by providing real world examples whenever it is possible.

This project started in January 2017.
Feel free to contribute.

Happy reading!

Contents

Acronyms

BIOS Basic Input Output System. 4

DHCP Dyanmic Host Configuration Protocol. 5

DNS Domain Name System. 5

ROM Read Only Memory. 4

SSH Secure Shell. 5

Chapter 1: Computer Science

1.1 Basic

Firmware
Firmware is software that is semi-permanently placed in hardware. Firmware does not disappear when hardware is turned off (typically stored in Flash or ROM). Firmware is typically involved with very basic low-level operations, which without a device, would be non-functional.

BIOS
Basic Input Output System (BIOS) is a software that is saved on the computer's motherboard and is turned on whenever the computer boots up. Its primary task is to prepare the components of the machine, so that other software (like the operating system) can boot up, run and take over the control of the machine.

Modular
Refers to the design of any system composed of separate components that can be connected together. The beauty of modular architecture is that you can replace or add any one component (module) without affecting the rest of the system. The opposite of a modular architecture is an integrated architecture, in which no clear divisions exist between components. The term modular can apply to both hardware and software. Modular software design, for example, refers to a design strategy in which a system is composed of relatively small and autonomous routines that fit together.

Standalone application
A standalone application is a application that is downloaded on your local computer and is self contained. Meaning it's not dependent on another service for it to run, unlike an web application that requires a web browser to work. (i.e. you need Chrome/Safari/Firefox to run Facebook).

1.2 Cybersecurity

Botnet [2]
A botnet is different from a isolated machine with malware. A botnet is a collection of infected machines which are coordinated through a central server called the Command Contrtrol server, CC server. The users of the machines involved in a botnet isn't necessarily aware that they're a part of the botnet. A classic example for which a botnet can be used is a DDoS attack (hence the D in Distributed).

Steganography `https://www.youtube.com/watch?v=TWEXCYQKyDc`

1.3 Networking

.

1.3.1 Application layer

DHCP
Dyanmic Host Configuration Protocol (DHCP) is a network protocol responsible for assigning IP addresses to hosts on a network. The host sends out a broadcast and gets an answer from all the DHCP-servers nearby. It's then up to the host to choose one server and then inform all the other servers which one it chose. Considering the host is IP-less the packet is broadcasted with *UDP*. A DHCP server is often found integrated inside a *router*.

DNS
Domain Name System (DNS) translates an *IP-address* to a website.
Example: For instance, YouTube's IP address is 216.58.209.142. Without DNS you would have to type in the IP-address, instead of youtube.com.

SSH
Secure Shell (SSH) is a cryptographic network protocol used over unsecured networks. It's commonly used to perform remote login on a machine.

1.3.2 Transport layer

TCP
UDP

1.3.3 Network layer

IP address

ICMP

Routing table

1.3.4 Link layer

ARP

MAC address

PPP

Firewall

FTP

Tor

1.4 Software development

ASP.NET
.

Async/await *.*

Binary file
Binary means executable code that can be run directly by the machine without the need to be compiled.

Framework *hello*

Git Git is a version control system.

Native language
A native programming language is compiled to machine code. This is code that's unique to a particular operating system and can only be executed in the environment for which it was compiled. As a result, when you're dealing with a native language you have to have a different compiler for each operating system. You'll have one compiler application for Windows, another one for Mac, another one for a distinct flavor of Linux, and so on. Example of native languages: C, C++, Objective-C

Managed languages
In contrast, managed languages are compiled to an intermediate format that works across operating systems. Typically, these languages are compatible across operating systems, and include languages such as C# and Java. In addition, in manages languages, memory is allocated dynamically at runtime which means the programmer don't need to worry about allocationg and deallocating memory, which is periodically done by the garbage collector.

Wrapper
In the context of software engineering, a wrapper is defined as an entity that encapsulates and hides the underlying complexity of another entity by means of well-defined interfaces.

Wrapper application
A wrapper can be a piece of software that provides compatibility layer to another piece of software.

Wrapper function
A wrapper function is a function that exists just to call another function.

Driver
Source: `https://www.youtube.com/watch?v=t-aRlwLI-b0`

1.5 Hardware

Router

Modem

Hotspot (WiFi)

1.6 Web

API
API stands for Application Programming Interface. It's a way to let one application talk to another application through a middle-man - the API.

Babel

RESTAPI
API stands for Application Programming Interface. It's a way to let one application talk to another application through a middle-man - the API.

Transpile
In the context of web, transpiling means transforming one script into another script. A good example would be what Babel does when developing with react - it transpiles the JavaScript from ECMAScript 6 to ES5. One might be tempted to think that from for instance C# to IL is called transpiling. This would be wrong because the two languages have very different levels of abstraction (complexity).

TLS
Transport Layer Security is an extra layer between the TCP and HTTP layer, that together forms the HTTPS protocol. The TLS does an extra step of authentication before the browser starts to recieve data from the server. During this step the TLS will check if the connected server is actually the server we want to connect to. [4]

Onion web/onion browser
It's called the onion web because when a request is made on the Tor-network (using onion) every request that is sent out from your computer is wrapped up in several layers of encryption - kind of like an onion. So as that request travels through multiple computers, every layer of encryption slowly peeled off until it reaches its final destination. So if you requested google.com, it will finally arrive at its destination where the content of google.com, that you requested, is again wrapped up in several layers of encryption and is sent back to you.

Dark web
The dark web is the part of the web that is commonly known as the part of the web where bad things happen. However, it's not designed for that, but the fact that the dark web uses the onion web, and thus ensures anonymity, makes it attractive for felons. The dark web is a copy of the web that communicates in a different way to ensure anonymity. [3]

Deep web
Deep web is the part of the web that is not indexed and thus not searchable by popular search engines like Google. Pages that typically aren't indexed can be pages that is only meant for the

user, like your personal Facebook account, which is behind a password.

1.7 Other

ASP.NET
.

Security by obscurity
.

Porting
In software engineering, porting is the process of taking one software/library that's written for a specific environment (read: programming language) and make it runnable in another environment.
For instance taking a library that's written for Java and making it runnable in C#. This can be done either manually - which requires the developer to know both the source platform (Java) and the target platform (C#) - or it can be done by automating the process using tools like Sharpen.
Another example may be a web application that is ported to a mobile application.

Chapter 2: Encryption

2.1 General

Assymetric encryption
Assymetric encryption (also called public-key encryption) is typically used when there's a transaction involved. Assymetric encryption has a public and priavte key. The public key is used to encrypt a message, while the private key is used to decrypt it.
Example: RSA is an assymetric type of encryption.

Symmetric encryption
On the other hand, symmetric encryption only uses one key, the private key. It's used both for encryption and decryption. Examples on symmetric encryption includes AES, Twofish, Blowfish.

Certificate Authority (CA)
Certificate authorities are trusted third-party organizations who verify the identity of individuals or organizations and then issue digital certificates containing both identity information and a copy of the subject's public key.

Digital Certificate
A digital certificate is a license issued by the Central Authority. This is a proof for anyone visitng your site that you are actually who you are claiming to be. The digital certificate can be provided to anyone you wish to communicate to without having to worry about sending it securely, because it doesn't contain any sensitive information. The person receiving the certificate doesn't have to verify your identity directly. They simply verify that the certificate is valid, by verifying the CA's signature on the certificate. If that checks out, they know that the public key contained in the certificate does, in fact, belong to the individual or organization named on the certificate.
Digital certificates should not be confused with digital signatures, which are used to verify that a message has not been tampered with. A digital certificate on the other hand associates a person with a specific public key with the help of a CA.

Digital signature
In asymmetric encryption, a message is encrypted using a public key and decrypted using a private key. That's because we are trying to create messages that only someone with a private key could read. In the case of digital signatures, we reverse this and use the private key for encryption, and the public key for decryption. That's because our goal is different.
We don't want to create a secret message, but rather we want to create a message that could only have been created by a specific person who possesses the private key and can then be verified by anyone with the corresponding public key.

Example: Let's say that Alice wants to send a message to Bob that includes Alice's digital signature.

Alice's side:

1. Alice takes her plain-text message and runs it through a hash function outputting a hash 9kjasd3.
2. Alice takes the hash and encrypts it using her own private key, producing what is known as, a digital signature. The digital signature is just the hash encrypted with the senders private key.
3. Alice sends both the plain-text message and the digial signature to Bob.

Bob's side:
Bob now needs to verify that the message he received from Alice has not been tampered with.

1. Bob takes the plain-text message and uses the same hash function Alice used to produce a hash, 9kjasd3.
2. Bob then takes the digital signature he received, and decrypts it using Alice's public key.
3. He then verifies that the decrypted text is actually the hash that was produced in step 1. If not, he knows the message has been tampered with.

Hashing (message digest)
Turning a variable length input into a *unique* fixed length output (the hash) is called hashing. Typically, passwords are hashed (and salted) to avoid storing the password in clear-text. This is done by running the clear-text password through a hash function. Unlike encryption, hashing is a one-way street and can't be reversed to its original form. You can however be sure that, if you run that same input through the hash function, you will get the same result every time - which is the idea behind hashing passwords.
Example: MD5, SHA-1 and SHA-2 are popular hashing functions.

Hash collision
No two inputs ran through a hash function should produce the same hash. If so, we have a hash collision. A hash collision is sign of a poor hash function.
Example: Researchers have been able to break (read: provoke hash collision in) MD5, and recently also SHA-1 [source].

Key derivation
Key derivation is a method used to create uniform keys from a non uniform source key, which the attacker may have some knowledge of. The purpose is to prevent unauthorized parties from accessing the original source key. A key is derived using a Key Derivation Function (KDF), a special algorithm designed for this purpose. In a KDF it is important that the source key contains sufficient amount of randomness preventing a potential attacker from "brute-forcing" the derived key using information about the source key.
Different KDFs have different uses and are suitable for different tasks. KDFs are typically used to derive keys to perform a cryptographic operation or to store passwords.

Public key
A public key is a key that can be distributed publicly. However, only the people with the private key can decrypt the message. Public keys are typically used in assymetric encryption.

Salting
In cryptography, a salt is a additional parameter that is used when performing a hash so that $Hash(password, salt)$. The purpose of a salt is to add an extra layer of 'randomness' to the hash.
Example: You have two different users that want to hash their password using the hash function $h(x)$. Unfortunately, both users have the same password, *password123*. This means that $hash(password123)$, would produce the same hash for both users. This is unfortunate when storing the hash in a server that might be breached and exposed to a *rainbow table attack*.

Public Key Infrastructure (PKI)
Public Key Infrastructure is a set of roles, policies, and procedures needed to create, manage, distribute, use, store, and revoke *Digital Certificates* and manage public-key encryption.

2.2 Symmetric encryption

Modes in AES
AES has 6 modes: ECB, CBC, CTR, OFB, CFB and GCM.

- **CBC**
  CBC (Cipher Block Chaining) is one of them and is commonly used in databases. CBC uses something called an Initialization Vector (IV). This ensures that even with the same key and the same block of plaintext, you end up with encrypted ciphertext that isn't the same. This gives a stronger security.
  As the name implies, Cipher Block Chaining utilizes block chaining - which, in this context, means that it uses output from one cryptographic operation as input to the next one - creating a dependency between each block. As a result, it creates the drawback that each block has to be 16 bytes. Which means that in cases where there's not a multiple of 16, padding is required.

2.3 Key exchange

Diffie-Hellman key exchange

Diffie-Hellman is an algorithm used to establish a shared secret between two parties. It is primarily used as a method of exchanging cryptography keys for use in symmetric encryption algorithms like AES [1].

Example 1. A Diffie-Hellman exhange:

1. Alice and Bob agree on a prime number, p, and a base, g, in advance. For our example, let's assume that $p = 23$ and $g = 5$.
2. Alice chooses a secret integer a whose value is 6 and computes $\mathrm{A} = g^a \; mod \; p$. In this example, A has the value of 8.
3. Bob chooses a secret integer b whose value is 15 and computes $\mathrm{B} = g^{\,b} \; mod \; p$. In this example, B has the value of 19.
4. Alice sends A to Bob and Bob sends B to Alice.
5. To obtain the shared secret, Alice computes $s = B^{\,a} \; mod \; p$. In this example, Alice obtains the value of $s = 2$.
6. To obtain the shared secret, Bob computes $s = A^{\,b} \; mod \; p$. In this example, Bob obtains the value of $s = 2$.

Chapter 3: Mathematics

3.1 Calculus

Integrand (n)
The function we're integrating. Which means the expression inside the integration symbol.

$$\int_a^b 3x^2 + 2x$$

In this case, $3x^2 + 2x$ is the integrand.

3.2 Linear algebra

Flux

Fourier
A fourier series is a function that is replaced with an infinite series.

Matrix In mathematics, a matrix is a rectangular array of numbers, symbols or expressions arranged in rows and columns.

$$M = \begin{bmatrix} a & 7 & c \\ 5 & 0 & v \\ 0 & 9 & 10 \end{bmatrix}$$

3.3 Statistics

Statistics
Statistics is a branch of mathematics dealing with the collection, analysis, interpretation, presentation, and organization of data.

Chapter 4: Physics

4.1 Calculus

- Moment of intertia / Treghetsmoment (n)
  Empty.

Index

Bibliography

[1] Wikipedia. *Diffie–Hellman key exchange*. [Online; accessed 9-May-2017]. 2017. URL: https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange#Cryptographic_explanation.

[2] YouTube. *Botnets*. [Online; accessed 24-May-2017]. 2016. URL: https://www.youtube.com/watch?v=UVFmC178_Vs.

[3] YouTube. *Secrets of the Deep Dark Web*. [Online; accessed 24-May-2017]. 2016. URL: https://www.youtube.com/watch?v=joxQ_XbsPVw.

[4] YouTube. *Secure Web Browsing*. [Online; accessed 24-May-2017]. 2016. URL: https://youtu.be/E_wX40fQwEA?t=6m15s.