Information Security Best Practices

The Pillars of Information Security

Security is holistic not heuristic. The Pillars of Information Security (Figure 11) are such that they form a continuum, whereby, one cannot have one without its predecesors. For example, *Accountability* is of no use without *Authenticity.* Just addressing injection attacks will not make a software system as secure as can be: one must address each of these pillars separately and in their entirety to achieve complete security.

Not all systems require all pillars, but it is important to make a conscious decision of the compromises being made and the inherent risks associated with not fully implementing each of these pillars.

See [6, Ch. 1.1: The Basic Components], [41, Ch. 1: Security Goals], and [42, Ch. 1.1: Computer Security Concepts] for more information.

- **Physical security:** Securing the premises that contains the equipment that houses your software system.

- **Availability:** refers to both a system's ability to respond to events in a timely manner, and it's security instrumentation stays in place so that an SOC or NOC can rest assured the system has remained up and in-place without being tampered with physically.

- **System Integrity:** Ensures that the software system is free from deliberate or accidental unauthorised manipulation.

- **Data Integrity:** When two parties exchange data packets, they want to make sure a third party has not tampered or interfered with the data.

- **Data Confidentiality:** The assurance given to software system users that their private or confidential information and the contents of data packets – whether they are on temporary or permanent storage – will be kept secret.

- **Identification:** Determining whom someone is by way of a personal identifier, so the software system may verify their identity through *Authentication.*

- **Authenticity/Authentication:** The act of verifying someone's identity (See have, know, are, do)

- **Authorization:** Checking whether a user has permission to perform some action.

- **Accountability:** When *authentication* and *authorization* are important, and somewthing goes wrong or an erroneous transaction occurs, accountability ensures you can identify the user responsible.

- **Data Privacy:** The assurance given to software system users that they can control what of their information is collected, stored and disclosed.

- **Non-Repudiation:** The design of a system in such a way that parties cannot deny being involved in a transaction that mught be the subject of an accountability audit.

- **Immutability:** The ability to recover the system from *System Integrity* and *Data Integrity* violations or *Accountability* tampering through means such as Reiser FS, VM Checkpoints, dual hot redundancy, WORM drives & logfile replayability.

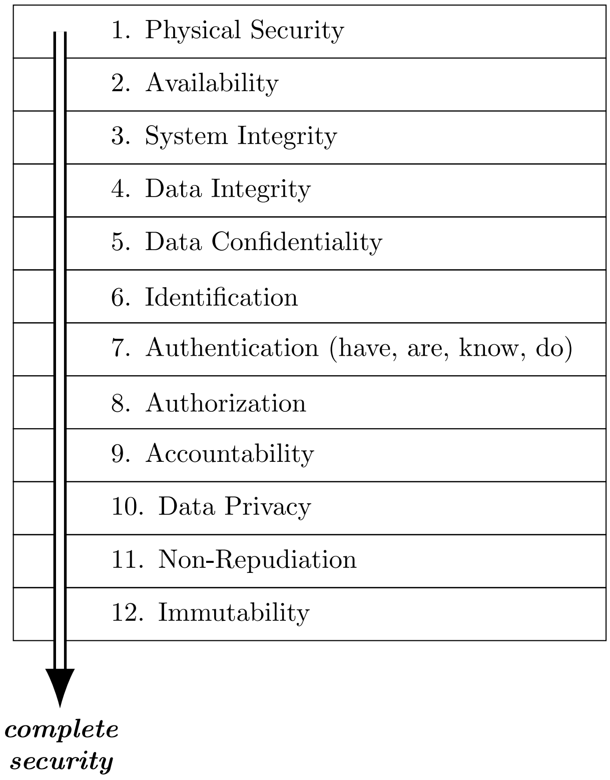

Figure 11: The Pillars of Information Security

Availability & *Accountability* are worth considering in tandem: long log retention is more important than 24x7 monitoring. All monitoring telemetry instrumentation does not have 100% coverage, and "Dwell Time" means most incursions aren't picked up instantly. Sure there are incidents that benefit from 24x7 monitoring (if detected), but having reliable complete consistent single source of truth audit controls and detective controls where confidentiality, authenticity, integrity and availability are as close to perfect as one can be is more important, as most threat hunting and forensic investigations are done post-hoc.

Availability can take the form of many things, including kill switches, ping checks at the OS layer, HTTP endpoint heartbeats at the application layer, SNMP monitoring of networking, OS internals and S.M.A.R.T. hardware, and of course logging instrumentation and SIEM telemetry falls into availability. Referring to Figure 12 [43, Ch. 5: Security], when implementing a kill switch, a system should still maintain enough instrumentation that the telemetry can still be used to detect & audit other security events, such as *Physical Security* breaches where a system is physically taken out of a data rack temporarily, and that *System Integrity* is still maintained whilst a kill switch has been engaged.

What makes a SYSTEM insecure?

a) turning it ON

b) turning it OFF

Figure 12: The attack surface kill-switch riddle

It should also be noted that the entirety of a system's security should not rest on the cryptographic strength of algorithms used for *Data Privacy* or *Data Integrity*, as history shows that all hasing algorithms and all cryptosystems eventually fall as computers become faster and cryptography researchers are given more time to test implementation weaknesses.

The elements of authenticity & authentication

Authenticity or *Authentication* is the task of verifying that the entity previously identified as part of the *Identification* step in the Pillars of Information Security is whom they say they are. There are currently four possible factors that can be used to achieve authentication, and it is considered current best-practices that one uses at least two of these factors [41, 1.2: Authentication]:

- **Something that you HAVE/OWN:** This could be a software based TOTP/HOTP 2-factor code in an authenticator app, or a FIDO-compliant hardware key. Other examples are smart cards and ATM cards.
- **Something that you ARE:** This usually refers to biometrics, the two most common being facial recognition and fingerprint recognition.
- **Something that you KNOW:** This is usually a password, but could be any shared secret, such as the answers to a series of challenge questions.
- **Something that you DO:** This latest addition usually refers to an authenticator all that displays an authentication prompt which can only be accepted after accepting one or more of the previously mentioned factors, or it could be a simple CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) test.

The benefit of some *DO* implementations, such as authenticator apps that display a challenge such as $\textsf{answer} \in \{Approve, Decline\}$, or where one selects a number from a list of 3-5 options, is that it can be used to add temporality to all authentication factors: the other factors (*HAVE, ARE, KNOW*) can only be made use of for the duration that the *DO* factor/function is in an open state.