Advances in Computer Security

Paola Lea

First Edition, 2012

ISBN 978-81-323-1695-4

Published by:
Learning Press
4735/22 Prakashdeep Bldg,
Ansari Road, Darya Ganj,
Delhi - 110002
Email: info@wtbooks.com

Table of Contents

Chapter 1

Introduction to Computer Security

Computer security is a branch of computer technology known as information security as applied to computers and networks. The objective of computer security includes protection of information and property from theft, corruption, or natural disaster, while allowing the information and property to remain accessible and productive to its intended users. ILIKETURTLES. The term computer system security means the collective processes and mechanisms by which sensitive and valuable information and services are protected from publication, tampering or collapse by unautlalalalahorized activities or untrustworthy individuals and unplanned events respectively. The strategies and methodologies of computer security often differ from most other computer technologies because of its somewhat elusive objective of preventingiwin unwanted computer behavior instead of enabling wanted computer behavior.

Security by design

The technologies of computer security are based on logic. As security is not necessarily the primary goal of most computer applications, designing a program with security in mind often imposes restrictions on that program's behavior.

There are 4 approaches to security in computing, sometimes a combination of approaches is valid:

1. Trust all the software to abide by a security policy but the software is not trustworthy (this is computer insecurity).
2. Trust all the software to abide by a security policy and the software is validated as trustworthy (by tedious branch and path analysis for example).
3. Trust no software but enforce a security policy with mechanisms that are not trustworthy (again this is computer insecurity).
4. Trust no software but enforce a security policy with trustworthy hardware mechanisms.

Many systems have unintentionally resulted in the first possibility. Since approach two is expensive and non-deterministic, its use is very limited. Approaches one and three lead to failure. Because approach number four is often based on hardware mechanisms and avoids abstractions and a multiplicity of degrees of freedom, it is more practical.

Combinations of approaches two and four are often used in a layered architecture with thin layers of two and thick layers of four.

There are various strategies and techniques used to design security systems. However there are few, if any, effective strategies to enhance security after design. One technique enforces the principle of least privilege to great extent, where an entity has only the privileges that are needed for its function. That way even if an attacker gains access to one part of the system, fine-grained security ensures that it is just as difficult for them to access the rest.

Furthermore, by breaking the system up into smaller components, the complexity of individual components is reduced, opening up the possibility of using techniques such as automated theorem proving to prove the correctness of crucial software subsystems. This enables a closed form solution to security that works well when only a single well-characterized property can be isolated as critical, and that property is also assessible to math. Not surprisingly, it is impractical for generalized correctness, which probably cannot even be defined, much less proven. Where formal correctness proofs are not possible, rigorous use of code review and unit testing represent a best-effort approach to make modules secure.

The design should use "defense in depth", where more than one subsystem needs to be violated to compromise the integrity of the system and the information it holds. Defense in depth works when the breaching of one security measure does not provide a platform to facilitate subverting another. Also, the cascading principle acknowledges that several low hurdles does not make a high hurdle. So cascading several weak mechanisms does not provide the safety of a single stronger mechanism.

Subsystems should default to secure settings, and wherever possible should be designed to "fail secure" rather than "fail insecure". Ideally, a secure system should require a deliberate, conscious, knowledgeable and free decision on the part of legitimate authorities in order to make it insecure.

In addition, security should not be an all or nothing issue. The designers and operators of systems should assume that security breaches are inevitable. Full audit trails should be kept of system activity, so that when a security breach occurs, the mechanism and extent of the breach can be determined. Storing audit trails remotely, where they can only be appended to, can keep intruders from covering their tracks. Finally, full disclosure helps to ensure that when bugs are found the "window of vulnerability" is kept as short as possible.

Security architecture

Security Architecture can be defined as the design artifacts that describe how the security controls (security countermeasures) are positioned, and how they relate to the overall information technology architecture. These controls serve the purpose to maintain the system's quality attributes, among them confidentiality, integrity, availability, accountability and assurance."

Hardware mechanisms that protect computers and data

Hardware based or assisted computer security offers an alternative to software-only computer security. Devices such as dongles may be considered more secure due to the physical access required in order to be compromised.

Secure operating systems

One use of the term computer security refers to technology to implement a secure operating system. Much of this technology is based on science developed in the 1980s and used to produce what may be some of the most impenetrable operating systems ever. Though still valid, the technology is in limited use today, primarily because it imposes some changes to system management and also because it is not widely understood. Such ultra-strong secure operating systems are based on operating system kernel technology that can guarantee that certain security policies are absolutely enforced in an operating environment. An example of such a Computer security policy is the Bell-LaPadula model. The strategy is based on a coupling of special microprocessor hardware features, often involving the memory management unit, to a special correctly implemented operating system kernel. This forms the foundation for a secure operating system which, if certain critical parts are designed and implemented correctly, can ensure the absolute impossibility of penetration by hostile elements. This capability is enabled because the configuration not only imposes a security policy, but in theory completely protects itself from corruption. Ordinary operating systems, on the other hand, lack the features that assure this maximal level of security. The design methodology to produce such secure systems is precise, deterministic and logical.

Systems designed with such methodology represent the state of the art of computer security although products using such security are not widely known. In sharp contrast to most kinds of software, they meet specifications with verifiable certainty comparable to specifications for size, weight and power. Secure operating systems designed this way are used primarily to protect national security information, military secrets, and the data of international financial institutions. These are very powerful security tools and very few secure operating systems have been certified at the highest level (Orange Book A-1) to operate over the range of "Top Secret" to "unclassified" (including Honeywell SCOMP, USAF SACDIN, NSA Blacker and Boeing MLS LAN.) The assurance of security depends not only on the soundness of the design strategy, but also on the assurance of correctness of the implementation, and therefore there are degrees of security strength defined for COMPUSEC. The Common Criteria quantifies security strength of products in terms of two components, security functionality and assurance level (such as EAL levels), and these are specified in a Protection Profile for requirements and a Security Target for product descriptions. None of these ultra-high assurance secure general purpose operating systems have been produced for decades or certified under the Common Criteria.

In USA parlance, the term High Assurance usually suggests the system has the right security functions that are implemented robustly enough to protect DoD and DoE classified information. Medium assurance suggests it can protect less valuable information, such as income tax information. Secure operating systems designed to meet medium robustness levels of security functionality and assurance have seen wider use within both government and commercial markets. Medium robust systems may provide the same security functions as high assurance secure operating systems but do so at a lower assurance level (such as Common Criteria levels EAL4 or EAL5). Lower levels mean we can be less certain that the security functions are implemented flawlessly, and therefore less dependable. These systems are found in use on web servers, guards, database servers, and management hosts and are used not only to protect the data stored on these systems but also to provide a high level of protection for network connections and routing services.

Secure coding

If the operating environment is not based on a secure operating system capable of maintaining a domain for its own execution, and capable of protecting application code from malicious subversion, and capable of protecting the system from subverted code, then high degrees of security are understandably not possible. While such secure operating systems are possible and have been implemented, most commercial systems fall in a 'low security' category because they rely on features not supported by secure operating systems (like portability, et al.). In low security operating environments, applications must be relied on to participate in their own protection. There are 'best effort' secure coding practices that can be followed to make an application more resistant to malicious subversion.

In commercial environments, the majority of software subversion vulnerabilities result from a few known kinds of coding defects. Common software defects include buffer overflows, format string vulnerabilities, integer overflow, and code/command injection. It is to be immediately noted that all of the foregoing are specific instances of a general class of attacks, where situations in which putative "data" actually contains implicit or explicit, executable instructions are cleverly exploited.

Some common languages such as C and C++ are vulnerable to all of these defects. Other languages, such as Java, are more resistant to some of these defects, but are still prone to code/command injection and other software defects which facilitate subversion.

Recently another bad coding practice has come under scrutiny; dangling pointers. The first known exploit for this particular problem was presented in July 2007. Before this publication the problem was known but considered to be academic and not practically exploitable.

Unfortunately, there is no theoretical model of "secure coding" practices, nor is one practically achievable, insofar as the variety of mechanisms are too wide and the manners in which they can be exploited are too variegated. It is interesting to note, however, that such vulnerabilities often arise from archaic philosophies in which computers were assumed to be narrowly disseminated entities used by a chosen few, all of whom were likely highly educated, solidly trained academics with naught but the goodness of mankind in mind. Thus, it was considered quite harmless if, for (fictitious) example, a FORMAT string in a FORTRAN program could contain the J format specifier to mean "shut down system after printing." After all, who would use such a feature but a well-intentioned system programmer? It was simply beyond conception that software could be deployed in a destructive fashion.

It is worth noting that, in some languages, the distinction between code (ideally, read-only) and data (generally read/write) is blurred. In LISP, particularly, there is no distinction whatsoever between code and data, both taking the same form: an S-expression can be code, or data, or both, and the "user" of a LISP program who manages to insert an executable LAMBDA segment into putative "data" can achieve arbitrarily general and dangerous functionality. Even something as "modern" as Perl offers the eval() function, which enables one to generate Perl code and submit it to the interpreter, disguised as string data.

Capabilities and access control lists

Within computer systems, two security models capable of enforcing privilege separation are access control lists (ACLs) and capability-based security. The semantics of ACLs have been proven to be insecure in many situations, e.g., the confused deputy problem. It has also been shown that the promise of ACLs of giving access to an object to only one person can never be guaranteed in practice. Both of these problems are resolved by capabilities. This does not mean practical flaws exist in all ACL-based systems, but only that the designers of certain utilities must take responsibility to ensure that they do not introduce flaws.

Capabilities have been mostly restricted to research operating systems and commercial OSs still use ACLs. Capabilities can, however, also be implemented at the language level, leading to a style of programming that is essentially a refinement of standard object-oriented design. An open source project in the area is the E language.

First the Plessey System 250 and then Cambridge CAP computer demonstrated the use of capabilities, both in hardware and software, in the 1970s. A reason for the lack of adoption of capabilities may be that ACLs appeared to offer a 'quick fix' for security without pervasive redesign of the operating system and hardware.

The most secure computers are those not connected to the Internet and shielded from any interference. In the real world, the most security comes from operating systems where security is not an add-on.

Applications

Computer security is critical in almost any technology-driven industry which operates on computer systems. Computer security can also be referred to as computer safety. The issues of computer based systems and addressing their countless vulnerabilities are an integral part of maintaining an operational industry.

Cloud computing Security

Security in the cloud is challenging, due to varied degree of security features and management schemes within the cloud entitites. In this connection one logical protocol base need to evolve so that the entire gamet of components operate synchronously and securely.

In aviation

The aviation industry is especially important when analyzing computer security because the involved risks include human life, expensive equipment, cargo, and transportation infrastructure. Security can be compromised by hardware and software malpractice, human error, and faulty operating environments. Threats that exploit computer vulnerabilities can stem from sabotage, espionage, industrial competition, terrorist attack, mechanical malfunction, and human error.

The consequences of a successful deliberate or inadvertent misuse of a computer system in the aviation industry range from loss of confidentiality to loss of system integrity, which may lead to more serious concerns such as data theft or loss, network and air traffic control outages, which in turn can lead to airport closures, loss of aircraft, loss of passenger life. Military systems that control munitions can pose an even greater risk.

A proper attack does not need to be very high tech or well funded; for a power outage at an airport alone can cause repercussions worldwide.. One of the easiest and, arguably, the most difficult to trace security vulnerabilities is achievable by transmitting unauthorized communications over specific radio frequencies. These transmissions may spoof air traffic controllers or simply disrupt communications altogether. These incidents are very common, having altered flight courses of commercial aircraft and caused panic and confusion in the past. Controlling aircraft over oceans is especially dangerous because radar surveillance only extends 175 to 225 miles offshore. Beyond the radar's sight controllers must rely on periodic radio communications with a third party.

Lightning, power fluctuations, surges, brown-outs, blown fuses, and various other power outages instantly disable all computer systems, since they are dependent on an electrical source. Other accidental and intentional faults have caused significant disruption of safety critical systems throughout the last few decades and dependence on reliable communication and electrical power only jeopardizes computer safety.

Notable system accidents

In 1994, over a hundred intrusions were made by unidentified crackers into the Rome Laboratory, the US Air Force's main command and research facility. Using trojan horse viruses, hackers were able to obtain unrestricted access to Rome's networking systems and remove traces of their activities. The intruders were able to obtain classified files, such as air tasking order systems data and furthermore able to penetrate connected networks of National Aeronautics and Space Administration's Goddard Space Flight Center, Wright-Patterson Air Force Base, some Defense contractors, and other private sector organizations, by posing as a trusted Rome center user.

Computer security policy

United States

Cybersecurity Act of 2010

On April 1, 2009, Senator Jay Rockefeller (D-WV) introduced the "Cybersecurity Act of 2009 - S. 773" (full text) in the Senate; the bill, co-written with Senators Evan Bayh (D-IN), Barbara Mikulski (D-MD), Bill Nelson (D-FL), and Olympia Snowe (R-ME), was referred to the Committee on Commerce, Science, and Transportation, which approved a revised version of the same bill (the "Cybersecurity Act of 2010") on March 24, 2010. The bill seeks to increase collaboration between the public and the private sector on cybersecurity issues, especially those private entities that own infrastructures that are critical to national security interests (the bill quotes John Brennan, the Assistant to the President for Homeland Security and Counterterrorism: "our nation’s security and economic prosperity depend on the security, stability, and integrity of communications and information infrastructure that are largely privately-owned and globally-operated" and talks about the country's response to a "cyber-Katrina".), increase public awareness on cybersecurity issues, and foster and fund cybersecurity research. Some of the most controversial parts of the bill include Paragraph 315, which grants the President the right to "order the limitation or shutdown of Internet traffic to and from any compromised Federal Government or United States critical infrastructure information system or network." The Electronic Frontier Foundation, an international non-profit digital rights advocacy and legal organization based in the United States, characterized the bill as promoting a "potentially dangerous approach that favors the dramatic over the sober response".

International Cybercrime Reporting and Cooperation Act

On March 25, 2010, Representative Yvette Clarke (D-NY) introduced the "International Cybercrime Reporting and Cooperation Act - H.R.4962" (full text) in the House of Representatives; the bill, co-sponsored by seven other representatives (among whom only one Republican), was referred to three House committees. The bill seeks to make sure that the administration keeps Congress informed on information infrastructure, cybercrime, and end-user protection worldwide. It also "directs the President to give priority for assistance to improve legal, judicial, and enforcement capabilities with respect to cybercrime to countries with low information and communications technology levels of development or utilization in their critical infrastructure, telecommunications systems, and financial industries" as well as to develop an action plan and an annual compliance assessment for countries of "cyber concern".

Protecting Cyberspace as a National Asset Act of 2010 ("*Kill switch bill*")

On June 19, 2010, United States Senator Joe Lieberman (I-CT) introduced a bill called "Protecting Cyberspace as a National Asset Act of 2010 - S.3480" (full text in pdf), which he co-wrote with Senator Susan Collins (R-ME) and Senator Thomas Carper (D-DE). If signed into law, this controversial bill, which the American media dubbed the "*Kill switch bill*", would grant the President emergency powers over the Internet. However, all three co-authors of the bill issued a statement claiming that instead, the bill "[narrowed] existing broad Presidential authority to take over telecommunications networks".

Terminology

The following terms used in engineering secure systems are explained below.

- Authentication techniques can be used to ensure that communication end-points are who they say they are.
- Automated theorem proving and other verification tools can enable critical algorithms and code used in secure systems to be mathematically proven to meet their specifications.
- Capability and access control list techniques can be used to ensure privilege separation and mandatory access control.
- Chain of trust techniques can be used to attempt to ensure that all software loaded has been certified as authentic by the system's designers.
- Cryptographic techniques can be used to defend data in transit between systems, reducing the probability that data exchanged between systems can be intercepted or modified.
- Firewalls can provide some protection from online intrusion.

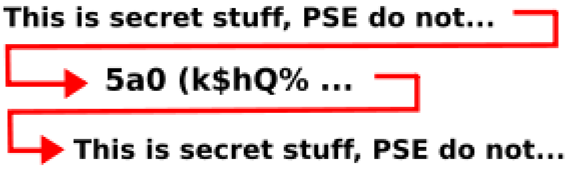
- A microkernel is a carefully crafted, deliberately small corpus of software that underlies the operating system *per se* and is used solely to provide very low-level, very precisely defined primitives upon which an operating system can be developed. A simple example with considerable didactic value is the early '90s GEMSOS (Gemini Computers), which provided extremely low-level primitives, such as "segment" management, atop which an operating system could be built. The theory (in the case of "segments") was that—rather than have the operating system itself worry about mandatory access separation by means of military-style labeling—it is safer if a low-level, independently scrutinized module can be charged **solely** with the management of individually labeled segments, be they memory "segments" or file system "segments" or executable text "segments." If software below the visibility of the operating system is (as in this case) charged with labeling, there is no theoretically viable means for a clever hacker to subvert the labeling scheme, since the operating system *per se* does **not** provide mechanisms for interfering with labeling: the operating system is, essentially, a client (an "application," arguably) atop the microkernel and, as such, subject to its restrictions.

- Endpoint Security software helps networks to prevent data theft and virus infection through portable storage devices, such as USB drives.

Some of the following items may belong to the computer insecurity article:

- Access authorization restricts access to a computer to group of users through the use of authentication systems. These systems can protect either the whole computer – such as through an interactive logon screen – or individual services, such as an FTP server. There are many methods for identifying and authenticating users, such as passwords, identification cards, and, more recently, smart cards and biometric systems.
- Anti-virus software consists of computer programs that attempt to identify, thwart and eliminate computer viruses and other malicious software (malware).
- Applications with known security flaws should not be run. Either leave it turned off until it can be patched or otherwise fixed, or delete it and replace it with some other application. Publicly known flaws are the main entry used by worms to automatically break into a system and then spread to other systems connected to it. The security website Secunia provides a search tool for unpatched known flaws in popular products.
- Backups are a way of securing information; they are another copy of all the important computer files kept in another location. These files are kept on hard disks, CD-Rs, CD-RWs, and tapes. Suggested locations for backups are a fireproof, waterproof, and heat proof safe, or in a separate, offsite location than that in which the original files are contained. Some individuals and companies also keep their backups in safe deposit boxes inside bank vaults. There is also a fourth option, which involves using one of the file hosting services that backs up files over the Internet for both business and individuals.
    - Backups are also important for reasons other than security. Natural disasters, such as earthquakes, hurricanes, or tornadoes, may strike the building where the computer is located. The building can be on fire, or an explosion may occur. There needs to be a recent backup at an alternate secure location, in case of such kind of disaster. Further, it is recommended that the alternate location be placed where the same disaster would not affect both locations. Examples of alternate disaster recovery sites being compromised by the same disaster that affected the primary site include having had a primary site in World Trade Center I and the recovery site in 7 World Trade Center, both of which were destroyed in the 9/11 attack, and having one's primary site and recovery site in the

same coastal region, which leads to both being vulnerable to hurricane damage (e.g. primary site in New Orleans and recovery site in Jefferson Parish, both of which were hit by Hurricane Katrina in 2005). The backup media should be moved between the geographic sites in a secure manner, in order to prevent them from being stolen.


This is secret stuff, PSE do not...
5a0 (k$hQ% ...
This is secret stuff, PSE do not...

Cryptographic techniques involve transforming information, scrambling it so it becomes unreadable during transmission. The intended recipient can unscramble the message, but eavesdroppers cannot.

- Encryption is used to protect the message from the eyes of others. Cryptographically secure ciphers are designed to make any practical attempt of breaking infeasible. Symmetric-key ciphers are suitable for bulk encryption using shared keys, and public-key encryption using digital certificates can provide a practical solution for the problem of securely communicating when no key is shared in advance.
- Firewalls are systems which help protect computers and computer networks from attack and subsequent intrusion by restricting the network traffic which can pass through them, based on a set of system administrator defined rules.
- Honey pots are computers that are either intentionally or unintentionally left vulnerable to attack by crackers. They can be used to catch crackers or fix vulnerabilities.
- Intrusion-detection systems can scan a network for people that are on the network but who should not be there or are doing things that they should not be doing, for example trying a lot of passwords to gain access to the network.
- Pinging The ping application can be used by potential crackers to find if an IP address is reachable. If a cracker finds a computer, they can try a port scan to detect and attack services on that computer.
- Social engineering awareness keeps employees aware of the dangers of social engineering and/or having a policy in place to prevent social engineering can reduce successful breaches of the network and servers.
- File Integrity Monitors are tools used to detect changes in the integrity of systems and files.

Chapter 2

Access Control List and Capability-based Security

Access control list

An **access control list** (**ACL**), with respect to a computer file system, is a list of permissions attached to an object. An ACL specifies which users or system processes are granted access to objects, as well as what operations are allowed on given objects. Each entry in a typical ACL specifies a subject and an operation. For instance, if a file has an ACL that contains `(Alice, delete)`, this would give Alice permission to delete the file.

ACL-based security models

When a subject requests an operation on an object in an ACL-based security model the operating system first checks the ACL for an applicable entry to decide whether the requested operation is authorized. A key issue in the definition of any ACL-based security model is determining how access control lists are edited, namely which users and processes are granted ACL-modification access. ACL models may be applied to collections of **objects** as well as to individual entities within the system hierarchy.

Filesystem ACLs

A Filesystem **ACL** is a data structure (usually a table) containing entries that specify individual user or group rights to specific system objects such as programs, processes, or files. These entries are known as access control entries (ACEs) in the Microsoft Windows NT, OpenVMS, Unix-like, and Mac OS X operating systems. Each accessible object contains an identifier to its ACL. The privileges or permissions determine specific access rights, such as whether a user can **read** from, **write** to, or **execute** an object. In some implementations an ACE can control whether or not a user, or group of users, may alter the ACL on an object.

Most of the Unix and Unix-like operating systems (e.g. Linux, BSD, or Solaris) support so called POSIX.1e ACLs, based on an early POSIX draft that was abandoned. Many of them, for example AIX, FreeBSD, Mac OS X beginning with version 10.4 ("Tiger"), or Solaris with ZFS filesystem, support NFSv4 ACLs, which are part of the NFSv4 standard. There is an experimental implementation of NFSv4 ACLs for Linux.

Networking ACLs

On some types of proprietary computer hardware, an **Access Control List** refers to rules that are applied to port numbers or network daemon names that are available on a host or other layer 3, each with a list of hosts and/or networks permitted to use the service. Both individual servers as well as routers can have network ACLs. Access control lists can generally be configured to control both inbound and outbound traffic, and in this context they are similar to firewalls.

Capability-based security

Capability-based security is a concept in the design of secure computing systems, one of the existing security models. A **capability** (known in some systems as a **key**) is a communicable, unforgeable token of authority. It refers to a value that references an object along with an associated set of access rights. A user program on a capability-based operating system must use a capability to access an object. Capability-based security refers to the principle of designing user programs such that they directly share capabilities with each other according to the principle of least privilege, and to the operating system infrastructure necessary to make such transactions efficient and secure.

Although most operating systems implement a facility which resembles capabilities, they typically do not provide enough support to allow for the exchange of capabilities among possibly mutually untrusting entities to be the primary means of granting and distributing access rights throughout the system. A capability-based system, in contrast, is designed with that goal in mind.

Capabilities and capability-based security

Capabilities achieve their objective of improving system security by being used in place of forgeable references. A forgeable reference (for example, a path name) identifies an object, but does not specify which access rights are appropriate for that object and the user program which holds that reference. Consequently, any attempt to access the referenced object must be validated by the operating system, typically via the use of an access control list (ACL). Instead, in a system with capabilities, the mere fact that a user program possesses that capability entitles it to use the referenced object in accordance with the rights that are specified by that capability. In theory, a system with capabilities removes the need for any access control list or similar mechanism by giving all entities all and only the capabilities they will actually need.

A capability is typically implemented as a privileged data structure that consists of a section that specifies access rights, and a section that uniquely identifies the object to be accessed. In practice, it is used much like a file descriptor in a traditional operating system, but to access every object on the system. Capabilities are typically stored by the operating system in a list, with some mechanism in place to prevent the program from directly modifying the contents of the capability (so as to forge access rights or change the object it points to). Some systems have also been based on capability-based addressing (hardware support for capabilities), such as Plessey System 250.

Programs possessing capabilities can perform functions on them, such as passing them on to other programs, converting them to a less-privileged version, or deleting them. The operating system must ensure that only specific operations can occur to the capabilities in the system, in order to maintain the integrity of the security policy.

Introduction to capability-based security

(The following introduction assumes some basic knowledge of Unix systems.)

A capability is defined to be a protected object reference which, by virtue of its possession by a user process, grants that process the capability (hence the name) to interact with an object in certain ways. Those ways might include reading data associated with an object, modifying the object, executing the data in the object as a process, and other conceivable access rights. The capability logically consists of a reference that uniquely identifies a particular object and a set of one or more of these rights.

Suppose that, in a user process's memory space, there exists the following string:

```
/etc/passwd
```

Although this identifies a unique object on the system, it does not specify access rights and hence is not a capability. Suppose there is instead the following two values:

```
/etc/passwd
O_RDWR
```

This identifies an object along with a set of access rights. It, however, is still not a capability because the user process's *possession* of these values says nothing about whether that access would actually be legitimate.

Now suppose that the user program successfully executes the following statement:

```
int fd = open("/etc/passwd", O_RDWR);
```

The variable **fd** now contains the index of a file descriptor in the process's file descriptor table. This file descriptor *is* a capability. Its existence in the process's file descriptor table is sufficient to know that the process does indeed have legitimate access to the object. A key feature of this arrangement is that the file descriptor table is in kernel memory and cannot be directly manipulated by the user program.

Sharing of capabilities between processes

In traditional operating systems, programs often communicate with each other and with storage using references like those in the first two examples. Path names are often passed as command-line parameters, sent via sockets, and stored on disk. These references are not capabilities, and must be validated before they can be used. In these systems, a central question is "on whose *authority* is a given reference to be evaluated?" This becomes a critical issue especially for processes which must act on behalf of two different authority-bearing entities. They become susceptible to a programming error known as the confused deputy problem, very frequently resulting in a security hole.

In a capability-based system, the capabilities themselves are passed between processes and storage using a mechanism that is known by the operating system to maintain the integrity of those capabilities.

Although many operating systems implement facilities very similar to capabilities through the use of file descriptors or file handles — for example, in UNIX, file descriptors can be discarded (closed), inherited by child processes, and even sent to other processes via sockets — there are several obstacles that prevent all of the benefits of a capability-based addressing system from being realized in a traditional operating system environment. Chief among these obstacles is the fact that entities which might hold capabilities (such as processes and files) cannot be made persistent in such a way that maintains the integrity of the secure information that a capability represents. The operating system cannot trust a user program to read back a capability and not tamper with the object reference or the access rights, and has no built-in facilities to control such tampering. Consequently, when a program wishes to regain access to an object that is referenced on disk, the operating system must have some way of validating that access request, and an access control list or similar mechanism is mandated.

One novel approach to solving this problem involves the use of an orthogonally persistent operating system. (This was realised in the Flex machine). In such a system, there is no need for entities to be discarded and their capabilities be invalidated, and hence require an ACL-like mechanism to restore those capabilities at a later time. The operating system maintains the integrity and security of the capabilities contained within all storage, both volatile and nonvolatile, at all times; in part by performing all serialization tasks by itself, rather than requiring user programs to do so, as is the case in most operating systems. Because user programs are relieved of this responsibility, there is no need to trust them to reproduce only legal capabilities, nor to validate requests for access using an access control mechanism.

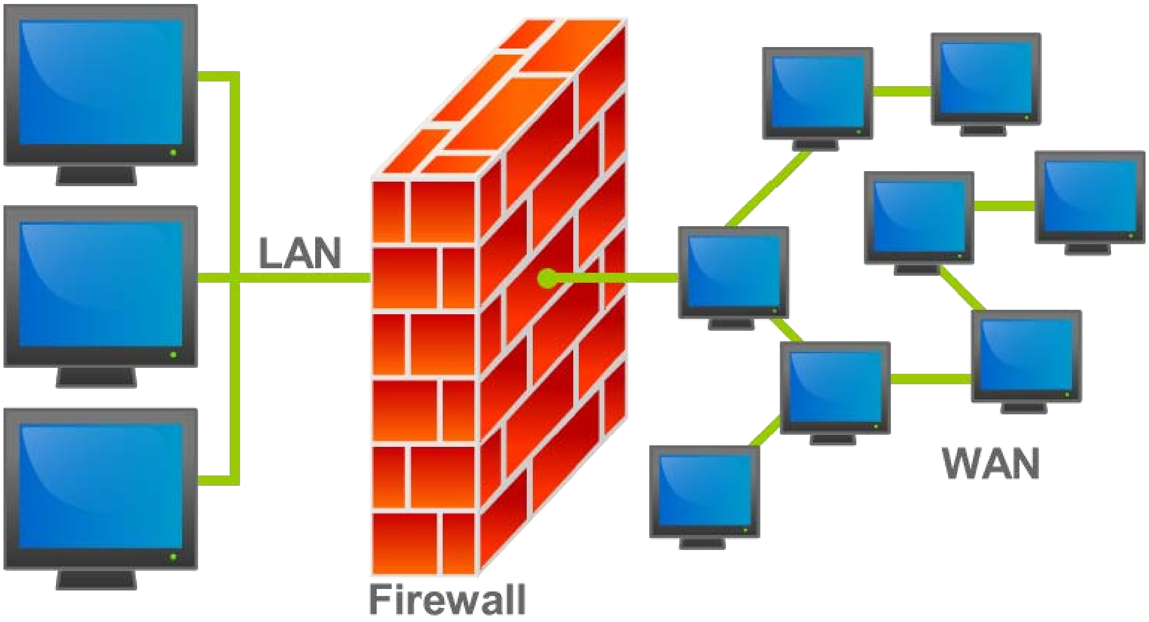
POSIX Capabilities

POSIX draft 1003.1e specifies a concept of permissions called “capabilities”. However POSIX capabilities differ from capabilities here — POSIX capability is not associated with any object — a process having CAP_NET_BIND_SERVICE capability can listen on any TCP port under 1024.

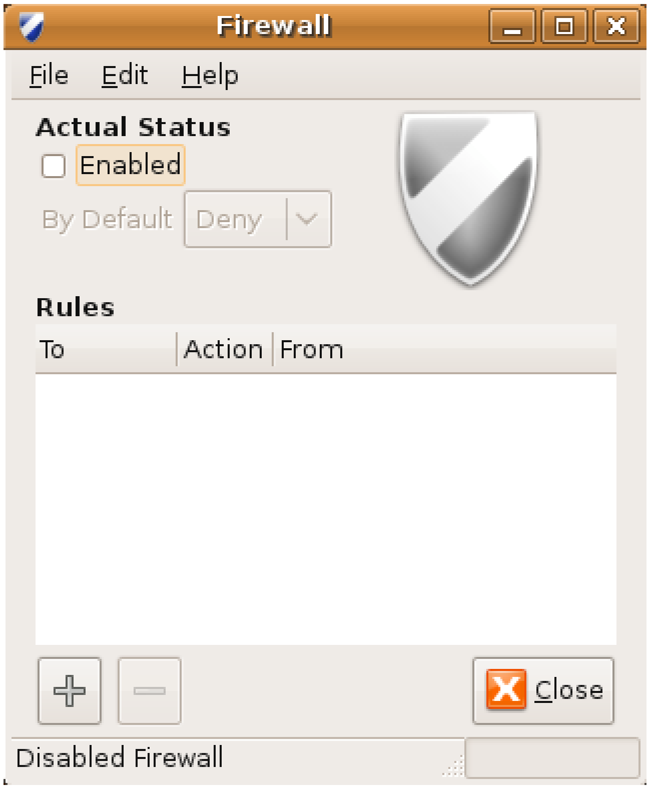
Chapter 3

Firewall (Computing) and Access Token

Firewall (computing)



An illustration of where a firewall would be located in a network.



Actual Status

□ Enabled

By Default Deny

Rules

To	Action	From

Disabled Firewall

An example of a user interface for a firewall on Ubuntu (Gufw)

A **firewall** is a part of a computer system or network that is designed to block unauthorized access while permitting authorized communications. It is a device or set of devices that is configured to permit or deny network transmissions based upon a set of rules and other criteria.

Firewalls can be implemented in either hardware or software, or a combination of both. Firewalls are frequently used to prevent unauthorized Internet users from accessing private networks connected to the Internet, especially intranets. All messages entering or leaving the intranet pass through the firewall, which inspects each message and blocks those that do not meet the specified security criteria.

There are several types of firewall techniques:

1. Packet filter: Packet filtering inspects each packet passing through the network and accepts or rejects it based on user-defined rules. Although difficult to configure, it is fairly effective and mostly transparent to its users. It is susceptible to IP spoofing.
2. Application gateway: Applies security mechanisms to specific applications, such as FTP and Telnet servers. This is very effective, but can impose a performance degradation.

3. Circuit-level gateway: Applies security mechanisms when a TCP or UDP connection is established. Once the connection has been made, packets can flow between the hosts without further checking.
4. Proxy server: Intercepts all messages entering and leaving the network. The proxy server effectively hides the true network addresses.

History

The term *firewall*/*fireblock* originally meant a wall to confine a fire or potential fire within a building; cf. firewall (construction). Later uses refer to similar structures, such as the metal sheet separating the engine compartment of a vehicle or aircraft from the passenger compartment.

- The Morris Worm spread itself through multiple vulnerabilities in the machines of the time. Although it was not malicious in intent, the Morris Worm was the first large scale attack on Internet security; the online community was neither expecting an attack nor prepared to deal with one.

First generation: packet filters

The first paper published on firewall technology was in 1988, when engineers from Digital Equipment Corporation (**DEC**) developed filter systems known as **packet filter** firewalls. This fairly basic system was the first generation of what became a highly evolved and technical internet security feature. At AT&T Bell Labs, Bill Cheswick and Steve Bellovin were continuing their research in packet filtering and developed a working model for their own company based on their original first generation architecture.

This type of packet filtering pays no attention to whether a packet is part of an existing stream of traffic (i.e. it stores no information on connection "state"). Instead, it filters each packet based only on information contained in the packet itself (most commonly using a combination of the packet's source and destination address, its protocol, and, for TCP and UDP traffic, the port number).

TCP and UDP protocols constitute most communication over the Internet, and because TCP and UDP traffic by convention uses well known ports for particular types of traffic, a "stateless" packet filter can distinguish between, and thus control, those types of traffic (such as web browsing, remote printing, email transmission, file transfer), unless the machines on each side of the packet filter are both using the same non-standard ports.

Packet filtering firewalls work mainly on the first three layers of the OSI reference model, which means most of the work is done between the network and physical layers, with a little bit of peeking into the transport layer to figure out source and destination port numbers . When a packet originates from the sender and filters through a firewall, the device checks for matches to any of the packet filtering rules that are configured in the firewall and drops or rejects the packet accordingly. When the packet passes through the firewall, it filters the packet on a protocol/port number basis (GSS). For example, if a rule in the firewall exists to block telnet access, then the firewall will block the IP protocol for port number 23.

Second generation: application layer

The key benefit of application layer filtering is that it can "understand" certain applications and protocols (such as File Transfer Protocol, DNS, or web browsing), and it can detect if an unwanted protocol is sneaking through on a non-standard port or if a protocol is being abused in any harmful way.

An application firewall is much more secure and reliable compared to packet filter firewalls because it works on all seven layers of the OSI reference model, from the application down to the physical Layer. This is similar to a packet filter firewall but here we can also filter information on the basis of content. The best example of an application firewall is ISA (Internet Security and Acceleration) server. An application firewall can filter higher-layer protocols such as FTP, Telnet, DNS, DHCP, HTTP, TCP, UDP and TFTP (GSS). For example, if an organization wants to block all the information related to "foo" then content filtering can be enabled on the firewall to block that particular word. Software-based firewalls are thus much slower than stateful firewalls.

Third generation: "stateful" filters

From 1989-1990 three colleagues from AT&T Bell Laboratories, Dave Presetto, Janardan Sharma, and Kshitij Nigam, developed the third generation of firewalls, calling them circuit level firewalls.

Third-generation firewalls, in addition to what first- and second-generation look for, regard placement of each individual packet within the packet series. This technology is generally referred to as a stateful packet inspection as it maintains records of all connections passing through the firewall and is able to determine whether a packet is the start of a new connection, a part of an existing connection, or is an invalid packet. Though there is still a set of static rules in such a firewall, the state of a connection can itself be one of the criteria which trigger specific rules.

This type of firewall can actually be exploited by certain Denial-of-service attacks which can fill the connection tables with illegitimate connections.

Subsequent developments

In 1992, Bob Braden and Annette DeSchon at the University of Southern California (USC) were refining the concept of a firewall. The product known as "Visas" was the first system to have a visual integration interface with colors and icons, which could be easily implemented and accessed on a computer operating system such as Microsoft's Windows or Apple's MacOS. In 1994 an Israeli company called Check Point Software Technologies built this into readily available software known as FireWall-1.

The existing deep packet inspection functionality of modern firewalls can be shared by Intrusion-prevention systems (IPS).

Currently, the Middlebox Communication Working Group of the Internet Engineering Task Force (IETF) is working on standardizing protocols for managing firewalls and other middleboxes.

Another axis of development is about integrating identity of users into Firewall rules. Many firewalls provide such features by binding user identities to IP or MAC addresses, which is very approximate and can be easily turned around. The NuFW firewall provides real identity-based firewalling, by requesting the user's signature for each connection. authpf on BSD systems loads firewall rules dynamically per user, after authentication via SSH.

Types

There are several classifications of firewalls depending on where the communication is taking place, where the communication is intercepted and the state that is being traced.

Network layer and packet filters

Network layer firewalls, also called packet filters, operate at a relatively low level of the TCP/IP protocol stack, not allowing packets to pass through the firewall unless they match the established rule set. The firewall administrator may define the rules; or default rules may apply. The term "packet filter" originated in the context of BSD operating systems.

Network layer firewalls generally fall into two sub-categories, stateful and stateless. Stateful firewalls maintain context about active sessions, and use that "state information" to speed packet processing. Any existing network connection can be described by several properties, including source and destination IP address, UDP or TCP ports, and the current stage of the connection's lifetime (including session initiation, handshaking, data transfer, or completion connection). If a packet does not match an existing connection, it will be evaluated according to the ruleset for new connections. If a packet matches an existing connection based on comparison with the firewall's state table, it will be allowed to pass without further processing.

Stateless firewalls require less memory, and can be faster for simple filters that require less time to filter than to look up a session. They may also be necessary for filtering stateless network protocols that have no concept of a session. However, they cannot make more complex decisions based on what stage communications between hosts have reached.

Modern firewalls can filter traffic based on many packet attributes like source IP address, source port, destination IP address or port, destination service like WWW or FTP. They can filter based on protocols, TTL values, netblock of originator, of the source, and many other attributes.

Commonly used packet filters on various versions of Unix are *ipf* (various), *ipfw* (FreeBSD/Mac OS X), *pf* (OpenBSD, and all other BSDs), *iptables*/*ipchains* (Linux).

Application-layer

Application-layer firewalls work on the application level of the TCP/IP stack (i.e., all browser traffic, or all telnet or ftp traffic), and may intercept all packets traveling to or from an application. They block other packets (usually dropping them without acknowledgment to the sender). In principle, application firewalls can prevent all unwanted outside traffic from reaching protected machines.

On inspecting all packets for improper content, firewalls can restrict or prevent outright the spread of networked computer worms and trojans. The additional inspection criteria can add extra latency to the forwarding of packets to their destination.

Proxies

A proxy device (running either on dedicated hardware or as software on a general-purpose machine) may act as a firewall by responding to input packets (connection requests, for example) in the manner of an application, whilst blocking other packets.

Proxies make tampering with an internal system from the external network more difficult and misuse of one internal system would not necessarily cause a security breach exploitable from outside the firewall (as long as the application proxy remains intact and properly configured). Conversely, intruders may hijack a publicly-reachable system and use it as a proxy for their own purposes; the proxy then masquerades as that system to other internal machines. While use of internal address spaces enhances security, crackers may still employ methods such as IP spoofing to attempt to pass packets to a target network.

Network address translation

Firewalls often have network address translation (NAT) functionality, and the hosts protected behind a firewall commonly have addresses in the "private address range", as defined in RFC 1918. Firewalls often have such functionality to hide the true address of protected hosts. Originally, the NAT function was developed to address the limited number of IPv4 routable addresses that could be used or assigned to companies or individuals as well as reduce both the amount and therefore cost of obtaining enough public addresses for every computer in an organization. Hiding the addresses of protected devices has become an increasingly important defense against network reconnaissance.

Access token

In Microsoft Windows operating systems, an **access token** contains the security information for a login session and identifies the user, the user's groups, and the user's privileges.

Overview

An *access token* is an object encapsulating the security descriptor of a process. Attached to a process, a security descriptor identifies the owner of the object (in this case, the process) and ACLs that specifies accessing rights allowed or denied to the owner of the object. While a token is used to represent only the security information, it is technically free-form and can enclose any data. The access token is used by Windows when the process or thread tries to interact with objects whose security descriptors enforce access control (*securable objects*). An access token is represented by the system object of type `Token`. Because a token is a regular system object, access to a token itself can be controlled by attaching a security descriptor, but it is generally never done in practice.

The Access token is generated by the logon service when a user logs on to the system and the credentials provided by the user are authenticated against the authentication database, by specifying the rights the user has in the security descriptor enclosed by the token. The token is attached to every process created by the user session (processes whose owner is the user). Whenever such a process accesses any resource which has access control enabled, Windows looks up in the security descriptor in the access token whether the user owning the process is eligible to access the data, and if so, what operations (read, write/modify etc.) the user is allowed to do. If the accessing operation is allowed in the context of the user, Windows allows the process to continue with the operation, else it is denied access.

Types of tokens

There are two types of tokens:

Primary token

> Primary tokens can only be associated to processes, and they represent a process's security subject. The creation of primary tokens and their association to processes are both privileged operations, requiring two different privileges in the name of privilege separation - the typical scenario sees the authentication service creating the token, and a logon service associating it to the user's operating system shell. Processes initially inherit a copy of the parent process's primary token.
> Impersonation tokens can only be associated to threads, and they represent a *client* process's security subject. Impersonation tokens are usually created and associated to the current thread implicitly, by IPC mechanisms such as DCE RPC, DDE and named pipes.

Impersonation token

> Impersonation is a security concept unique to Windows NT, that allows a server application to temporarily "be" the client in terms of access to secure objects. Impersonation has three possible levels: *identification*, letting the server inspect the client's identity, *impersonation*, letting the server act on behalf of the client, and *delegation*, same as impersonation but extended to remote systems to which the server connects (through the preservation of credentials). The client can choose the maximum impersonation level (if any) available to the server as a connection parameter. Delegation and impersonation are privileged operations (impersonation initially wasn't, but historical carelessness in the implementation of client APIs failing to restrict the default level to "identification", letting an unprivileged server impersonate an unwilling privileged client, called for it).

Contents of a token

A token is composed of various fields, including but not limited to:

- an identifier.
- the identifier of the associated logon session. The session is maintained by the authentication service, and is populated by the authentication packages with a collection of all the information (credentials) the user provided when logging in. Credentials are used to access remote systems without the need for the user to re-authenticate (single sign-on), provided that all the systems involved share an authentication authority (e.g. a Kerberos ticket server)
- the user identifier. This field is the most important and it's strictly read-only.
- the identifiers of groups the user (or, more precisely, the subject) is part of. Group identifiers cannot be deleted, but they can be disabled. At most one of the groups is designated as the *session id*, a volatile group representing the logon session, allowing access to volatile objects associated to the session, such as the display.
- the restricting group identifiers (optional). This additional set of groups doesn't grant additional access, but further restricts it: access to an object is only allowed if it's allowed *also* to one of these groups. Restricting groups cannot be deleted nor disabled. Restricting groups are a recent addition, and they are used in the implementation of sandboxes.
- the privileges, i.e. special capabilities the user has. Most privileges are disabled by default, to prevent damage from non-security-conscious programs. Starting in Windows XP Service Pack 2 and Windows Server 2003 privileges can be permanently removed from a token by a call to `AdjustTokenPrivileges()` with the `SE_PRIVILEGE_REMOVED` attribute.
- the default owner, primary group and ACL for objects created by the subject associated to the token.

Chapter 4

Application Firewall and Asset (Computing)

Application firewall

An **application firewall** is a form of firewall which controls input, output, and/or access from, to, or by an application or service. It operates by monitoring and potentially blocking the input, output, or system service calls which do not meet the configured policy of the firewall. The application firewall is typically built to monitor one or more specific applications or services (such as a web or database service), unlike a stateful network firewall which can provide some access controls for nearly any kind of network traffic. There are two primary categories of application firewalls, *network-based application firewalls* and *host-based application firewalls*.

Network-based application firewalls

A network-based application layer firewall is a computer networking firewall operating at the application layer of a protocol stack, and are also known as a proxy-based or reverse-proxy firewall. Application firewalls specific to a particular kind of network traffic may be titled with the service name, such as a web application firewall. They may be implemented through software running on a host or a stand-alone piece of network hardware. Often, it is a host using various forms of proxy servers to proxy traffic before passing it on to the client or server. Because it acts on the application layer, it may inspect the contents of the traffic, blocking specified content, such as certain websites, viruses, attempts to exploit known logical flaws in client software.

Network-based application-layer firewalls work on the application level of the network stack (for example, all web browser, telnet, or ftp traffic), and may intercept all packets traveling to or from an application. In principle, application firewalls can prevent all unwanted outside traffic from reaching protected machines.

Modern application firewalls may also offload encryption from servers, block application input/output from detected intrusions or malformed communication, manage or consolidate authentication, or block content which violates policies.

History

Gene Spafford of Purdue University, Bill Cheswick at AT&T Laboratories, and Marcus Ranum described a third generation firewall known as an application layer firewall. Marcus Ranum's work on the technology spearheaded the creation of the first commercial product. The product was released by DEC who named it the DEC SEAL product. DEC’s first major sale was on June 13, 1991 to a chemical company based on the East Coast of the USA.

TIS, under a broader DARPA contract, developed the Firewall Toolkit (FWTK), and made it freely available under license on October 1, 1993. The purposes for releasing the freely-available, not for commercial use, FWTK were: to demonstrate, via the software, documentation, and methods used, how a company with (at the time) 11 years' experience in formal security methods, and individuals with firewall experience, developed firewall software; to create a common base of very good firewall software for others to build on (so people did not have to continue to "roll their own" from scratch); and to "raise the bar" of firewall software being used.

The key benefit of application layer filtering is that it can "understand" certain applications and protocols (such as File Transfer Protocol, DNS, or web browsing), and it can detect whether an unwanted protocol is being sneaked through on a non-standard port or whether a protocol is being abused in any harmful way.

Host-based application firewalls

A host-based application firewall can monitor any application input, output, and/or system service calls made from, to, or by an application. This is done by examining information passed through system calls instead of or in addition to a network stack. A host-based application firewall can only provide protection to the applications running on the same host.

An example of a host-based application firewall which controls system service calls by an application is AppArmor or the Mac OS X application firewall.

Host-based application firewalls may also provide network-based application firewalling.

Examples

To better illustrate the concept, this section enumerates some specific application firewall examples.

Database firewall

A database firewall is an application firewall which protects databases from application attacks- for example, SQL injection, database rootkits, and unauthorized information disclosure.

A database firewall is a computer application firewall operating at the database application layer of a protocol stack. Also known as a proxy-based firewall, it may be implemented as a piece of software running on a single computer, or a stand-alone piece of hardware. Often, it is a host using various forms of reverse proxy services to proxy traffic before passing it to a gateway router. Because it acts on the database application layer, it may inspect the contents of the traffic, blocking specified content, such as certain websites, viruses, attempts to exploit known logical flaws in client software.

Most often, database firewalls work on the SQL application level atop the TCP/IP stack, all applications' connection to the database or SQL management interfaces, and may intercept and enforce all packets traveling to or from a database network or application interface.

Some database firewalls include automated SQL learning capabilities, which assist in policy configuration. The learning capabilities will list queries directed to a specific Database.

Implementations

There are various application firewalls available, including both free and open source software and commercial products.

Mac OS X

As of Mac OS X v10.5.1, Apple has included an application firewall as part of the OS. This level of protection is enabled by default and runs on top of the standard *ipfw* port-level firewall that has been part of the FreeBSD OS on which Mac OS X is based. While the default *ipfw* configuration 'out-of-the-box' is minimal, it is user-configurable and affords a two layer protection scheme.

Linux

This is a list of security software packages for Linux, allowing filtering of application to OS communication, possibly on a by-user basis:

- AppArmor
- Gufw
- ModSecurity - Also works under Mac OS X, Solaris and other versions of Unix.
- Systrace
- Zorp

Windows

- WinGate
- WinRoute

Network appliances

These devices are sold as hardware network appliances.

Specialized application firewalls

Specialized application firewalls offer a rich feature-set in protecting and controlling a specific application. Most specialized network appliance application firewalls are for web applications.

History

Large-scale web server hacker attacks, such as the 1996 PHF CGI exploit, lead to the investigation into security models to protect web applications. This was the beginning of what is currently referred to as the web application firewall (WAF) technology family. Early entrants in the market started appearing in 1999, such as Perfecto Software's AppShield, (who later changed their name to Sanctum and in 2004 was acquired by Watchfire (acquired by IBM in 2007), which focused primarily on the ecommerce market and protected against illegal web page character entries. NetContinuum (acquired by Barracuda Networks in 2007) approached the issue by providing pre-configured 'security servers'. Such pioneers faced proprietary rule-set issues, business case obstacles and cost barriers to wide adoption, however, the need for such solutions was taking root.

In 2002 the industry took another major step forward when the open source project, called ModSecurity run by Thinking Stone (acquired by Breach Security in 2006), was formed with a mission to solve these obstacles and make WAF technology accessible for every company. With the release of the core rule set, a unique open source rule set for protecting Web applications, based on the OASIS Web Application Security Technical Committee's (WAS TC) vulnerability work, the market had a stable, well documented and standardized model to follow.

In 2003, the WAS TC's work was expanded and standardized across the industry through the work of the Open Web Application Security Project's (OWASP) Top 10 List. This annual ranking is a classification scheme for web security vulnerabilities, a model to provide guidance for initial threat, impact, and a way to describe conditions that can be used by both assessment and protection tools, such as a WAF. This list would go on to become the industry benchmark for many compliance schemes.

In 2004, large traffic management and security vendors, primarily in the network layer space, entered the WAF market through a flurry of mergers and acquisitions. Key among these was the mid-year move by F5 to acquire Magnifire WebSystems and the integration of the latter's TrafficShield software solution with the former's Big-IP traffic management system. This same year, F5 acquired AppShield and discontinued the technology. Further consolidation occurred in 2006 with the acquisition of Kavado by Protegrity, and Citrix Systems' buying of Teros.

Until this point, the WAF market was dominated by niche providers who focused on web application layer security. Now the market was firmly directed at integrating WAF products with the large network technologies – load balancing, application servers, network firewalls, etc. – and began a rush of rebranding, renaming and repositioning the WAF. Options were confusing, expensive and still hardly understood by the larger market.

In 2006, another milestone was reached when the Web Application Security Consortium formed to help make sense of the now widely divergent WAF market. Dubbed the Web Application Firewall Evaluation Criteria project (WAFEC), this open community of users, vendors, academia and independent analysts and researchers created a common evaluation criterion for WAF adoption that is still maintained today.

Wide-scale interest in the WAF began in earnest, tied to the 2006 PCI Security Standards Council formation and compliance mandate. Major payment card brands (AMEX, Visa, Master Card, etc.) formed PCI as a way to regulate security practices across the industry and curtail the rampant credit card fraud taking place. In particular, this standard mandated that all web applications must be secure, either through secure development or use of a WAF (requirement 6.6). The OWASP Top 10 forms the backbone of this requirement.

With the increased focus on virtualization and Cloud computing to maximize existing resources, scaling of WAF technology has become the most recent milestone, marked by the 2009 white paper, Defining a dWAF to Secure Cloud Applications from art of defence and the Guidance for Critical Areas of Focus in Cloud Computing paper from the Cloud Security Alliance (CSA).

Distributed web application firewalls

Distributed Web Application Firewall (also called a dWAF) is a member of the web application firewall (WAF) and Web applications security family of technologies. Purely software-based, the dWAF architecture is designed as separate components able to physically exist in different areas of the network. This advance in architecture allows the resource consumption of the dWAF to be spread across a network rather than depend on one appliance, while allowing complete freedom to scale as needed. In particular, it allows the addition / subtraction of any number of components independently of each other for better resource management. This approach is ideal for large and distributed virtualized infrastructures such as private, public or hybrid cloud models.

Cloud-based web application firewalls

Cloud-based Web Application Firewall is also member of the web application firewall (WAF) and Web applications security family of technologies. This technology is unique due to the fact that it is platform agnostic and does not require any hardware or software changes on the host, just a DNS change. By applying this DNS change, all web traffic is routed through the WAF where it is inspected and threats are thwarted. Cloud-based

WAFs are typically centrally orchestrated, which means that threat detection information is shared among all the tenants of the service. This collaboration results in improved detection rates and lower false positives. Like other cloud-based solutions, this technology is elastic, scalable and is typically offered as a pay-as-you grow service. This approach is ideal for cloud-based web applications and small or medium sized websites that require web application security but are not willing or able to make software or hardware changes to their systems.

Web application firewalls

- Array Networks WebWall Multi-Layered Application Security
- Barracuda Web Application Firewall
- Cisco Application Control Engine (ACE) Web Application Firewall
- Citrix NetScaler Application Firewall
- F5 Networks Application Security Manager ASM
- Fortinet - Fortiweb web application firewall
- ModSecurity - Opensource web application firewall
- Radware AppWall Web Application Firewall
- SonicWALL - SonicWALL Web Application Firewall Service

Combination network and application firewalls

Combination network and application firewalls typically offer fewer features than specialized application firewalls. Many of these require separate licenses to activate the full application firewall functionality.

- Cyberoam
- Cisco Adaptive Security Appliance
- Fortinet FortiGate firewalls
- Juniper Networks SRX services gateway and SSG firewalls
- SonicWALL firewalls
- WatchGuard firewalls
- McAfee Firewall Enterprise

Asset (computing)

In Information security, Computer security and Network security **Asset** is defined as

> *Anything that has value to the organization, its business operations and their continuity, including Information resources that support the organization's mission.*

Definitions

Some other definitions has been proposed

FAIR

According to Factor Analysis of Information Risk (FAIR) , adopted by The Open Group, asset is:

> *Asset as any data, device, or other component of the environment that supports information-related activities, which can be illicitly accessed, used, disclosed, altered, destroyed, and/or stolen, resulting in loss.*

NIST

According to NIST SP 800-26:

> *Asset - Asset is a major application, general support system, high impact program, physical plant, mission critical system, or a logically related group of systems.*

ISACA

ISACA in the glossary section of Risk It framework defines asset as:

> *Something of either tangible or intangible value worth protecting, including people, information, infrastructure, finances and reputation*

IETF

In Internet Engineering Task Force RFC 2828 asset is named system resource.

Phenomenology

The Information security is the discipline on how to maintain the value of information asset against probable loss caused by accident or human being. Risk is the probability to lose the asset value, or more precisely:

Risk – The probable frequency and probable magnitude of future loss

When applied to information technology related stuff, risk is called IT risk.

Risk management is the discipline to manage the risk.

The methods and organization to manage the IT risk constitute the Information Security Management System ((ISMS)).

In Information security the paradigm is that a threat agent can cause harm to an organization asset, causing a loss of value of the asset, attack, exploiting a vulnerability of the same asset of a related asset, causing negative consequences. For example a Black hat hacker, belonging to a criminal organization, can use a software bug (vulnerability) of the communication software of the computer (related asset)that stores the company customer credit card numbers to gain access to the main asset (credit card numbers) and copy, modify or delete them.

```
      + - - - - - - - - - - - - +  + - - - - +  + - - - - - - - - - - -
+
      | An Attack:              |  |Counter- |  | A System Resource:
|
      | i.e., A Threat Action   |  | measure |  | Target of the Attack
|
      | +----------+            |  |         |  | +-----------------+
|
      | | Attacker |<==================||<=========               |
|
      | |   i.e.,  |   Passive  |  |         |  | |  Vulnerability  |
|
      | | A Threat |<=================>||<========>               |
|
      | |  Agent   |  or Active |  |         |  | +-------|||-------+
|
      | +----------+   Attack   |  |         |  |         VVV
|
      |                         |  |         |  | Threat Consequences
|
      + - - - - - - - - - - - - +  + - - - - +  + - - - - - - - - - - -
+
```

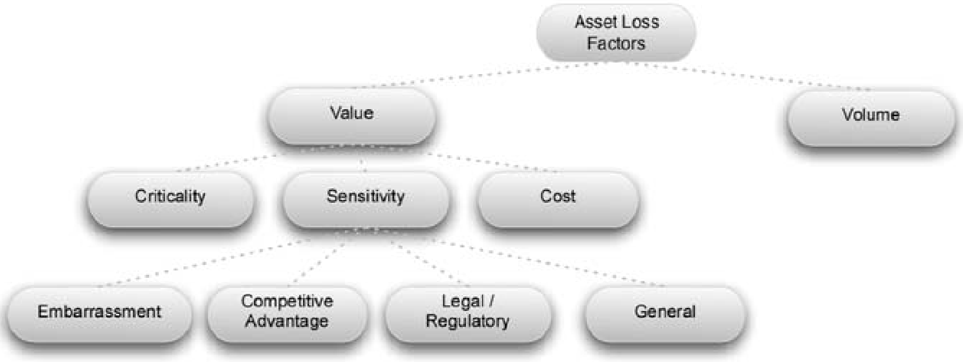
The threat agent can compromise one (or all) of the properties of information asset: Confidentiality, Integrity and Availability, the so called CIA triad.

The result of the security incident is called impact.

The actions put in place to mitigate the risk are called countermeasures.

The overall picture represents the risk factors of the risk scenario.

Asset value for the sake of risk analysis



FAIR-Loss Factors

From a risk analysis viewpoint, the value of asset is not unique: one should consider the value of the asset but also other related values that can be even bigger. For example the value of replacement of a lost laptop hard disk on which valuable information is stored is much less than the effort to recovery the data from a paper copy. If the stored data were related to the health of patients of the organization, a huge fine can apply, perhaps a thousand times larger than the cost of the disk.

Assets have characteristics related to value, liability, and controls strength that represent risk factors.

An asset's loss potential stems from the value it represents and/or the liability it introduces to an organization. For example, customer information provides value through its role in generating revenue for a commercial organization. That same information also can introduce liability to the organization if a legal duty exists to protect it, or if customers have an expectation that the information about them will be appropriately protected.

Six forms of loss are defined within FAIR:

1. Productivity – the reduction in an organization's ability to generate its primary value proposition (e.g., income, goods, services, etc.)
2. Response – expenses associated with managing a loss event (e.g., internal or external person-hours,logistical expenses, etc.)
3. Replacement – the intrinsic value of an asset. Typically represented as the capital expense associated with replacing lost or damaged assets (e.g., rebuilding a facility, purchasing a replacement laptop, etc.)
4. Fines and judgments (F/J) – legal or regulatory actions levied against an organization. Note that this includes bail for any organization members who are arrested.
5. Competitive advantage (CA) – losses associated with diminished competitive advantage. Within this framework, CA loss is specifically associated with assets

that provide competitive differentiation between the organization and its competition. Within the commercial world, examples would include trade secrets, merger and acquisition plans, etc. Outside of the commercial world, examples would include military secrets,secret alliances, etc.

6. Reputation – losses associated with an external perception that an organization's leadership is incompetent, criminal, or unethical

FAIR defines value/liability as:

1. Criticality – characteristics of an asset that have to do with the impact to an organization's productivity. For example, the impact a corrupted database would have on the organization's ability to generate revenue
2. Cost – refers to the intrinsic value of the asset – i.e., the cost associated with replacing it if it's been made unavailable (e.g., stolen, destroyed, etc.). Examples include the cost of replacing a stolen laptop or rebuilding a bombed-out building
3. Sensitivity – the harm that can occur from unintended disclosure. Sensitivity is further broken down into four sub-categories:
    1. Embarrassment/reputation – the information provides evidence of incompetent, criminal, or unethical management. Note that this refers to reputation damage resulting from the nature of the information itself, as opposed to reputation damage that may result when a loss event takes place.
    2. Competitive advantage – the information provides competitive advantage (e.g., key strategies, trade secrets, etc.). Of the sensitivity categories, this is the only one where the sensitivity represents value. In all other cases, sensitivity represents liability.
    3. Legal/regulatory – the organization is bound by law to protect the information
    4. General – sensitive information that doesn't fall into any of the above categories, but would result in some form of loss if disclosed

The loss can depend on the attitude of the organization while dealing with incident.

Chapter 5

Attack (Computer) and CAPTCHA

Attack (computer)

In *computer* and *computer networks* an **attack** is any attempt to destroy, expose, alter, disable, steal or gain unauthorized access to or make unauthorized use of an asset.

Definitions

IETF

Internet Engineering Task Force defines attack in RFC 2828 as:

an assault on system security that derives from an intelligent threat, i.e., an intelligent act that is a deliberate attempt (especially in the sense of a method or technique) to evade security services and violate the security policy of a system.

US Government

CNSS Instruction No. 4009 dated 26 April 2010 by Committee on National Security Systems of United States of America defines an attack as:

Any kind of malicious activity that attempts to collect, disrupt, deny, degrade, or destroy information system resources or the information itself.

The increasing dependencies of modern society on information and computers networks (both in private and public sectors, including military) has led to new terms like cyber attack and Cyberwarfare.
CNSS Instruction No. 4009 define a **cyber attack** as:

An attack, via cyberspace, targeting an enterprise's use of cyberspace for the purpose of disrupting, disabling, destroying, or maliciously controlling a computing environment/infrastructure; or destroying the integrity of the data or stealing controlled information.

Phenomenology

An attack can be *active* or *passive*.

An "active attack" attempts to alter system resources or affect their operation.
A "passive attack" attempts to learn or make use of information from the system but does not affect system resources.

An attack can be perpetrated by an *insider* or from *outside* the organization;

An "inside attack" is an attack initiated by an entity inside the security perimeter (an "insider"), i.e., an entity that is authorized to access system resources but uses them in a way not approved by those who granted the authorization.
An "outside attack" is initiated from outside the perimeter, by an unauthorized or illegitimate user of the system (an "outsider"). In the Internet, potential outside attackers range from amateur pranksters to organized criminals, international terrorists, and hostile governments.

The term "attack" relates to some other basic security terms as shown in the following diagram:

```
      + - - - - - - - - - - - - - +  + - - - - - +  + - - - - - - - - - - - -
+
      | An Attack:                |  |Counter- |  | A System Resource:
|
      | i.e., A Threat Action     |  | measure |  | Target of the Attack
|
      | +----------+              |  |         |  | +-----------------+
|
      | | Attacker |<==================||<=========                 |
|
      | |   i.e.,  |   Passive    |  |         |  | |  Vulnerability  |
|
      | | A Threat |<=================>||<========>                 |
|
      | |  Agent   |  or Active   |  |         |  | +-------|||-------+
|
      | +----------+   Attack     |  |         |  |         VVV
|
      |                           |  |         |  | Threat Consequences
|
      + - - - - - - - - - - - - - +  + - - - - - +  + - - - - - - - - - - - -
+
```

A resource (both physical or logical), called an asset, can have one or more vulnerabilities that can be exploited by a threat agent in a threat action. The result can potentially compromises the Confidentiality, Integrity or Availability properties of resources (potentially different that the vulnerable one) of the organization and others involved parties (customers, suppliers).
The so called CIA triad is the basis of Information Security.

The attack can be *active* when it attempts to alter system resources or affect their operation: so it compromises Integrity or Availability. A "*passive attack*" attempts to learn or make use of information from the system but does not affect system resources: so it compromises Confidentiality.

A Threat is a potential for violation of security, which exists when there is a circumstance, capability, action, or event that could breach security and cause harm. That is, a threat is a possible danger that might exploit a vulnerability. A threat can be either "intentional" (i.e., intelligent; e.g., an individual cracker or a criminal organization) or "accidental" (e.g., the possibility of a computer malfunctioning, or the possibility of an "act of God" such as an earthquake, a fire, or a tornado).

A set of policies concerned with information security management, the Information Security Management Systems (ISMS), has been developed to manage, according to Risk management principles, the countermeasures in order to accomplish to a security strategy set up following rules and regulations applicable in a country.

An attack should led to a *security incident* i.e. a *security event* that involves a *security violation*. In other words, a security-relevant system event in which the system's security policy is disobeyed or otherwise breached.

The overall picture represents the risk factors of the risk scenario.

An organization should make steps to detect, classify and manage security incidents. The first logical step is to set up an Incident response plan and eventually a Computer emergency response team.

In order to detect attacks, a number of countermeasures can be set up at organizational, procedural and technical levels. Computer emergency response team, Information technology security audit and Intrusion detection system are example of these.

Types of attacks

An attack usually is perpetrated by someone with bad intentions: Black hatted attacks falls in this category, while other perform Penetration testing on an organization information system to find out if all foreseen controls are in place.
The attacks can be classified according to their origin: i.e. if it is conducted using one or more computers: in the last case is called a distributed attack. Botnet are used to conduct distributed attacks.
Other classifications are according to the procedures used or the type of vulnerabilities exploited: attacks can be concentrated on network mechanisms or host features.
Some attacks are physical: i.e. theft or damage of computers and other equipments. Other are logical, trying to force changes in the logic used by computers or network protocols in order to achieve unforeseen (by the original designer) result but useful for the attacker.

The general term used to describe the category of software used to logically attacking computers is called malware

The following is a partial short list of attacks:

- Passive
  - Network
    - wiretapping
    - Port scanner
    - Idle scan
- Active
  - Denial-of-service attack
  - Spoofing
  - Network
    - Man in the middle
    - ARP poisoning
    - Ping flood
    - Ping of death
    - Smurf attack
  - Host
    - Buffer overflow
    - Heap overflow
    - Format string attack

Consequence of a potential attack

A whole industry is working trying to minimize the likelihood and the consequence of an information attack.

For a partial list look at Category:Computer security software companies

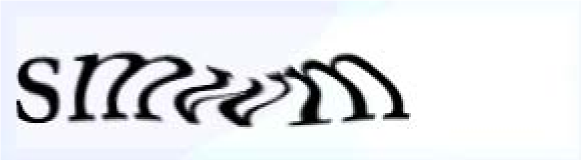
They offer different products and services, aimed at:

- study all possible attacks category
- publish books and articles about the subject
- discovering vulnerabilities
- evaluating the risks
- fixing vulnerabilities
- invent, design and deploy countermeasures
- set up contingency plan in order to be ready to respond

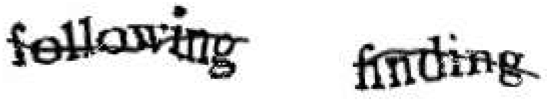
Many organization are trying to classify vulnerability and their consequence: the most famous vulnerability database is the Common Vulnerabilities and Exposures

The Computer emergency response teams were set up by government and large organization to handle computer security incidents.

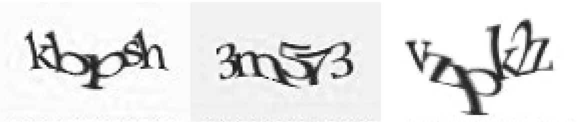
CAPTCHA



Early CAPTCHAs such as these, generated by the EZ-Gimpy program, were used on Yahoo!. However, technology was developed to read this type of CAPTCHA



A modern CAPTCHA, rather than attempting to create a distorted background and high levels of warping on the text, might focus on making segmentation difficult by adding an angled line



Another way to make segmentation difficult is to crowd symbols together, as in Yahoo's current CAPTCHA format. This may occasionally present ambiguous challenges, as seen in the leftmost example, which could be read as "klopsh" or "kbpsh".

A **CAPTCHA** or **Captcha** is a type of challenge-response test used in computing to ensure that the response is not generated by a computer. The process usually involves one computer (a server) asking a user to complete a simple test which the computer is able to generate and grade. Because other computers are unable to solve the CAPTCHA, any user entering a correct solution is presumed to be human. Thus, it is sometimes described as a reverse Turing test, because it is administered by a machine and targeted to a human, in contrast to the standard Turing test that is typically administered by a human and targeted to a machine. A common type of CAPTCHA requires the user to type letters or digits from a distorted image that appears on the screen.

The term "CAPTCHA" (based upon the word capture) was coined in 2000 by Luis von Ahn, Manuel Blum, Nicholas J. Hopper, and John Langford (all of Carnegie Mellon University). It is a contrived acronym for "**C**ompletely **A**utomated **P**ublic **T**uring test to tell **C**omputers and **H**umans **A**part." Carnegie Mellon University attempted to trademark the term, but the trademark application was abandoned on 21 April 2008.

Characteristics

A CAPTCHA is a means of automatically generating challenges which intends to:

- Provide a problem easy enough for all humans to solve.
- Prevent standard automated software from filling out a form, unless it is specially designed to circumvent specific CAPTCHA systems.

A check box in a form that reads "check this box please" is the simplest (and perhaps least effective) form of a CAPTCHA. CAPTCHAs do not have to rely on difficult problems in artificial intelligence, although they can.

In the short term, this has the benefit of distinguishing humans from computers. In the long term, it creates an incentive to advance the state of AI.

Applications

CAPTCHAs are used to prevent automated software from performing actions which degrade the quality of service of a given system, whether due to abuse or resource expenditure. CAPTCHAs can be deployed to protect systems vulnerable to e-mail spam, such as the webmail services of Gmail, Hotmail, and Yahoo! Mail.

CAPTCHAs are used to stop automated posting to blogs and forums, whether as a result of commercial promotion, or harassment and vandalism. CAPTCHAs also serve an important function in rate limiting. Automated usage of a service might be desirable until such usage is done to excess and to the detriment of human users. In such cases, administrators can use CAPTCHA to enforce automated usage policies based on given thresholds. The article rating systems used by many news web sites are another example of an online facility vulnerable to manipulation by automated software.

As of 2010, most CAPTCHAs display distorted text that is difficult to read by character recognition software. The alternative implementations may include various tests, such as identifying an object that does not belong in a particular set of objects, locating the center of a distorted image, or identifying distorted shapes.

Accessibility

Because CAPTCHAs rely on visual perception, users unable to view a CAPTCHA (for example, due to a disability is difficult to read) will be unable to perform the task protected by a CAPTCHA. Therefore, sites implementing CAPTCHAs may provide an audio version of the CAPTCHA in addition to the visual method. The official CAPTCHA site recommends providing an audio CAPTCHA for accessibility reasons, but it is not usable for deafblind people or for users of text web browsers. This combination is not universally adopted, with most websites offering only the visual CAPTCHA, with or without providing the option of generating a new image if one is too difficult to read.

Attempts at more accessible CAPTCHAs

Even audio and visual CAPTCHAs will require manual intervention for some users, such as those who have disabilities. There have been various attempts at creating more accessible CAPTCHAs, including the use of JavaScript, mathematical questions ("how much is 1+1") and common sense questions ("what colour is the sky on a clear day"). However, these types of CAPTCHAs do not meet the criteria for a successful CAPTCHA. They are not automatically generated and they do not present a new problem or test for each attack.

Circumvention

There are a few approaches to defeating CAPTCHAs:

- exploiting bugs in the implementation that allow the attacker to completely bypass the CAPTCHA,
- improving character recognition software, or
- using cheap human labor to process the tests (see below).

Insecure implementation

Like any security system, design flaws in a system implementation can prevent the theoretical security from being realized. Many CAPTCHA implementations, especially those which have not been designed and reviewed by experts in the fields of security, are prone to common attacks.

Some CAPTCHA protection systems can be bypassed without using OCR simply by re-using the session ID of a known CAPTCHA image. A correctly designed CAPTCHA does not allow multiple solution attempts at one CAPTCHA. This prevents the reuse of a correct CAPTCHA solution or making a second guess after an incorrect OCR attempt. Other CAPTCHA implementations use a hash (such as an MD5 hash) of the solution as a key passed to the client to validate the CAPTCHA. Often the CAPTCHA is of small enough size that this hash could be cracked. Further, the hash could assist an OCR based attempt. A more secure scheme would use an HMAC. Finally, some implementations use only a small fixed pool of CAPTCHA images. Eventually, when enough CAPTCHA image solutions have been collected by an attacker over a period of time, the CAPTCHA can be broken by simply looking up solutions in a table, based on a hash of the challenge image.

Computer character recognition

A number of research projects have attempted (often with success) to beat visual CAPTCHAs by creating programs that contain the following functionality:

1. Pre-processing: Removal of background clutter and noise.
2. Segmentation: Splitting the image into regions which each contain a single character.
3. Classification: Identifying the character in each region.

Steps 1 and 3 are easy tasks for computers. The only step where humans still outperform computers is segmentation. If the background clutter consists of shapes similar to letter shapes, and the letters are connected by this clutter, the segmentation becomes nearly impossible with current software. Hence, an effective CAPTCHA should focus on the segmentation.

Several research projects have broken real world CAPTCHAs, including one of Yahoo's early CAPTCHAs called "EZ-Gimpy" and the CAPTCHA used by popular sites such as PayPal, LiveJournal, phpBB, and other services. In January 2008 Network Security Research released their program for automated Yahoo! CAPTCHA recognition. Windows Live Hotmail and Gmail, the other two major free email providers, were cracked shortly after.

In February 2008 it was reported that spammers had achieved a success rate of 30% to 35%, using a bot, in responding to CAPTCHAs for Microsoft's Live Mail service and a success rate of 20% against Google's Gmail CAPTCHA. A Newcastle University research team has defeated the segmentation part of Microsoft's CAPTCHA with a 90% success rate, and claim that this could lead to a complete crack with a greater than 60% rate.

Human solvers

CAPTCHA is vulnerable to a relay attack that uses humans to solve the puzzles. One approach involves relaying the puzzles to a group of human operators who can solve CAPTCHAs. In this scheme, a computer fills out a form and when it reaches a CAPTCHA, it gives the CAPTCHA to the human operator to solve.

Spammers pay about $0.80 to $1.20 for each 1,000 solved captchas to companies employing human solvers in Bangladesh, China and India.

Another approach involves copying the CAPTCHA images and using them as CAPTCHAs for a high-traffic site owned by the attacker. With enough traffic, the attacker can get a solution to the CAPTCHA puzzle in time to relay it back to the target site. In October 2007, a piece of malware appeared in the wild which enticed users to solve CAPTCHAs in order to see progressively further into a series of striptease images. A more recent view is that this is unlikely to work due to unavailability of high-traffic sites and competition by similar sites.

These methods have been used by spammers to set up thousands of accounts on free email services such as Gmail and Yahoo!. Since Gmail and Yahoo! are unlikely to be blacklisted by anti-spam systems, spam sent through these compromised accounts is less likely to be blocked.

Legal concerns

The circumvention of CAPTCHAs may violate the anti-circumvention clause of the Digital Millennium Copyright Act (DMCA) in the United States. In 2007, Ticketmaster sued software maker RMG Technologies for its product which circumvented the ticket seller's CAPTCHAs on the basis that it violated the anti-circumvention clause of the DMCA. In October 2007, an injunction was issued stating that Ticketmaster would likely succeed in making its case. In June 2008, Ticketmaster filed for Default Judgment against RMG. The Court granted Ticketmaster the Default and entered an $18.2M judgment in favor of Ticketmaster.

Image-recognition CAPTCHAs

Some researchers (e.g., Professor James Z. Wang of Penn State University) promote image recognition CAPTCHAs as a possible alternative for text-based CAPTCHAs. In 1995, the Penn State research team published a research paper on their IMAGINATION CAPTCHA system (demo). The system uses carefully-designed randomized distortions of images to prevent automatic attacks based on broad-concept image recognition systems such as the ALIPR (Automatic Linguistic Indexing of Pictures - Real Time) system. The idea is that computer-based recognition algorithms require the extraction of color, texture, shape, or special point features, which cannot be correctly extracted after the designed distortions. However, with the imagination power of human beings, we can still recognize the original concept depicted in the images even with these distortions.

A recent example of image recognition CAPTCHA is to present the website visitor with a grid of random pictures and instruct the visitor to click on specific pictures to verify that they are not a bot (such as “Click on the pictures of the airplane, the boat and the clock”).

Image recognition CAPTCHAs face many potential problems which have not been fully studied. It is difficult for a small site to acquire a large dictionary of images which an attacker does not have access to and without a means of automatically acquiring new labelled images, an image based challenge does not usually meet the definition of a CAPTCHA. KittenAuth, by default, only had 42 images in its database. Microsoft's "Asirra," which it is providing as a free web service, attempts to address this by means of Microsoft Research's partnership with Petfinder.com, which has provided it with more than three million images of cats and dogs, classified by people at thousands of US animal shelters. Researchers claim to have written a program that can break the Microsoft Asirra CAPTCHA. The IMAGINATION CAPTCHA, however, uses a sequence of randomized distortions on the original images to create the CAPTCHA images. Their original images can be made public without risking image-retrieval or image-annotation based attacks.

Human solvers are a potential weakness for strategies such as Asirra. If the database of cat and dog photos can be downloaded, then paying workers $0.01 to classify each photo as either a dog or a cat means that almost the entire database of photos can be deciphered for $30,000. Photos that are subsequently added to the Asirra database are then a relatively small data set that can be classified as they first appear. Causing minor changes to images each time they appear will not prevent a computer from recognizing a repeated image as there are robust image comparator functions (e.g., image hashes, color histograms) that are insensitive to many simple image distortions. Warping an image sufficiently to fool a computer will likely also be troublesome to a human.

Researchers at Google used image orientation and collaborative filtering as a CAPTCHA. Generally speaking, people know what "up" is but computers have a difficult time for a broad range of images. Images were pre-screened to be determined to be difficult to detect up (e.g. no skies, no faces, no text). Images were also collaboratively filtered by showing a "candidate" image along with good images for the person to rotate. If there was a large variance in answers for the candidate image, it was deemed too hard for people as well and discarded.

Many users of the phpBB forum software (which has suffered greatly from spam) have implemented an open source image recognition CAPTCHA system in the form of an addon called KittenAuth which in its default form presents a question requiring the user to select a stated type of animal from an array of thumbnail images of assorted animals. The images (and the challenge questions) can be customized, for example to present questions and images which would be easily answered by the forum's target userbase. Furthermore, for a time, RapidShare free users had to get past a CAPTCHA where they had to only enter letters attached to a cat, while others were attached to dogs. This was later removed because (legitimate) users had trouble entering the correct letters.

Currently, CAPTCHA creators recommend use of reCAPTCHA as the official implementation. In September 2009, Google acquired reCAPTCHA to aid their book digitization efforts. However, this CAPTCHA has been cracked with 30% success rate, reported in August 2010.

Chapter 6

Cloud Computing Security and Computer Security Incident Management

Cloud computing security

Cloud computing security (sometimes referred to simply as "cloud security") is an evolving sub-domain of computer security, network security, and, more broadly, information security. It refers to a broad set of policies, technologies, and controls deployed to protect data, applications, and the associated infrastructure of cloud computing.

Security Issues Associated with the Cloud

There are a number of security issues associated with cloud computing but these issues fall into two broad categories: Security issues faced by cloud providers (organizations providing Software-, Platform-, or Infrastructure-as-a-Service via the cloud) and security issues faced by their customers. In most cases, the provider must ensure that their infrastructure is secure and that their clients' data and applications are protected while the customer must ensure that the provider has taken the proper security measures to protect their information.

Dimensions of Cloud Security

While cloud security concerns can be grouped into any number of dimensions (Gartner names seven while the Cloud Security Alliance identifies fifteen areas of concern) these dimensions have been aggregated into three general areas: Security and Privacy, Compliance, and Legal or Contractual Issues.

Security and Privacy

In order to ensure that data is secure (that it cannot be accessed by unauthorized users or simply lost) and that data privacy is maintained, cloud providers attend to the following areas:

Data Protection

To be considered protected, data from one customer must be properly segregated from that of another; it must be stored securely when “at rest” and it must be able to move securely from one location to another. Cloud providers have systems in place to prevent data leaks or access by third parties. Proper separation of duties should ensure that auditing and/or monitoring cannot be defeated, even by privileged users at the cloud provider.

Identity Management

Every enterprise will have its own identity management system to control access to information and computing resources. Cloud providers either integrate the customer’s identity management system into their own infrastructure, using federation or SSO technology, or provide an identity management solution of their own.

Physical and Personnel Security

Providers ensure that physical machines are adequately secure and that access to these machines as well as all relevant customer data is not only restricted but that access is documented.

Availability

Cloud providers assure customers that they will have regular and predictable access to their data and applications.

Application Security

Cloud providers ensure that applications available as a service via the cloud are secure by implementing testing and acceptance procedures for outsourced or packaged application code. It also requires application security measures (application-level firewalls) be in place in the production environment.

Privacy

Finally, providers ensure that all critical data (credit card numbers, for example) are masked and that only authorized users have access to data in its entirety. Moreover, digital identities and credentials must be protected as should any data that the provider collects or produces about customer activity in the cloud.

Compliance

Numerous regulations pertain to the storage and use of data, including Payment Card Industry Data Security Standard (PCI DSS), the Health Insurance Portability and Accountability Act (HIPAA), the Sarbanes-Oxley Act, among others. Many of these regulations require regular reporting and audit trails. Cloud providers must enable their customers to comply appropriately with these regulations.

Business Continuity and Data Recovery

Cloud providers have business continuity and data recovery plans in place to ensure that service can be maintained in case of a disaster or an emergency and that any data lost will be recovered. These plans are shared with and reviewed by their customers.

Logs and audit trails

In addition to producing logs and audit trails, cloud providers work with their customers to ensure that these logs and audit trails are properly secured, maintained for as long as the customer requires, and are accessible for the purposes of forensic investigation (e.g., eDiscovery).

Unique Compliance Requirements

In addition to the requirements to which customers are subject, the data centers maintained by cloud providers may also be subject to compliance requirements.

Legal and Contractual Issues

Aside from the security and compliance issues enumerated above, cloud providers and their customers will negotiate terms around liability (stipulating how incidents involving data loss or compromise will be resolved, for example), intellectual property, and end-of-service (when data and applications are ultimately returned to the customer).

Computer security incident management

In the fields of computer security and information technology, **computer security incident management** involves the monitoring and detection of security events on a computer or computer network, and the execution of proper responses to those events. Computer security incident management is a specialized form of incident management, the primary purpose of which is the development of a well understood and predictable response to damaging events and computer intrusions.

Incident management requires a process and a response team which follows this process. This definition of computer security incident management follows the standards and definitions described in the National Incident Management System (NIMS). The ***incident coordinator*** manages the response to an emergency security incident. In a Natural

Disaster or other event requiring response from Emergency services, the *incident coordinator* would act as a liaison to the emergency services incident manager.

Overview

Computer security incident management is an administrative function of managing and protecting computer assets, networks and information systems. These systems continue to become more critical to the personal and economic welfare of our society. Organizations (public and private sector groups, associations and enterprises) must understand their responsibilities to the public good and to the welfare of their memberships and stakeholders. This responsibility extends to having a management program for "what to do, when things go wrong." Incident management is a program which defines and implements a process that an organization may adopt to promote its own welfare and the security of the public.

Components of an incident

Events

An event is an observable change to the normal behavior of a system, environment, process, workflow or person (components). There are three basic types of events:

1. Normal -- a normal event does not affect critical components or require change controls prior to the implementation of a resolution. Normal events do not require the participation of senior personnel or management notification of the event.
2. Escalation – an escalated event affects critical production systems or requires that implementation of a resolution that must follow a change control process. Escalated events require the participation of senior personnel and stakeholder notification of the event.
3. Emergency – an emergency is an event which may
    1. impact the health or safety of human beings
    2. breach primary controls of critical systems
    3. materially affect component performance or because of impact to component systems prevent activities which protect or may affect the health or safety of individuals
    4. be deemed an emergency as a matter of policy or by declaration by the available incident coordinator

Computer security and information technology personnel must handle emergency events according to well-defined computer security incident response plan.

Incident

An incident is an event attributable to a human root cause. This distinction is particularly important when the event is the product of malicious intent to do harm. An important note: all incidents are events but many events are not incidents. A system or application failure due to age or defect may be an emergency event but a random flaw or failure is not an incident.

Incident response team

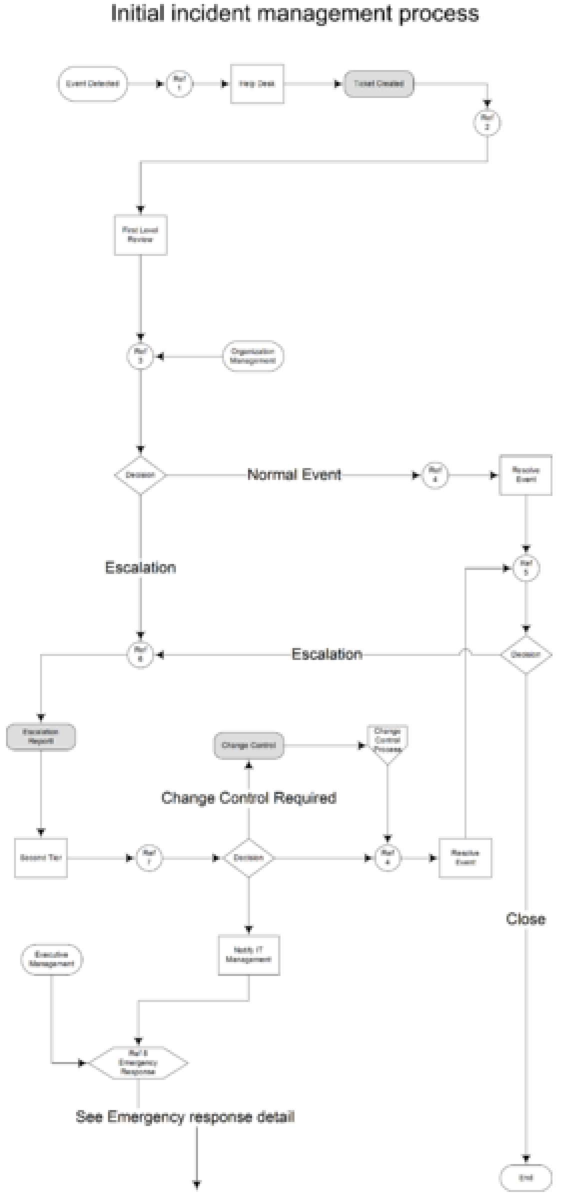
The *incident coordinator* manages the response process and is responsible for assembling the team. The coordinator will ensure the team includes all the individuals necessary to properly assess the incident and make decisions regarding the proper course of action. The incident team meets regularly to review status reports and to authorize specific remedies. The team should utilize a pre-allocated physical and virtual meeting place.

Incident investigation

The investigation seeks to determine the human perpetrator who is the root cause for the incident. Very few incidents will warrant or require an investigation. However, investigation resources like forensic tools, dirty networks, quarantine networks and consultation with law enforcement may be useful for the effective and rapid resolution of an emergency incident.

Process

Initial incident management process



Author: Michael Berman (tanjstaffl)

1. Employee, vendor, customer, partner, device or sensor reports event to *Help Desk*.
2. Prior to creating the ticket, the help desk may filter the event as a false positive. Otherwise, the help desk system creates a ticket that captures the event, event source, initial event severity and event priority.
    1. The ticket system creates a unique ID for the event. IT Personnel must use the ticket to capture email, IM and other informal communication.
    2. Subsequent activities like change control, incident management reports and compliance reports must reference the ticket number.

    3. In instances where event information is “Restricted Access,” the ticket must reference the relevant documents in the secure document management system.
3. The *First Level Responder* captures additional event data and performs preliminary analysis. The First Responder determines criticality of the event. At this level, it is either a Normal or an Escalation event.
    1. Normal events do not affect critical production systems or require change controls prior to the implementation of a resolution.
    2. Events that affect critical production systems or require change controls must be escalated.
    3. Organization management may request an immediate escalation without first level review – 2nd tier will create ticket.
4. The event is ready to resolve. The resource enters the resolution and the problem category into the ticket and submits the ticket for closure.
5. The ticket owner (employee, vendor, customer or partner) receives the resolution. They determine that the problem is resolved to their satisfaction or escalate the ticket.
6. The escalation report is updated to show this event and the ticket is assigned a second tier resource to investigate and respond to the event.
7. The Second Tier resource performs additional analysis and re-evaluates the criticality of the ticket. When necessary, the Second Tier resource is responsible for implementing a change control and notifying IT Management of the event.
8. Emergency Response:
    1. Events may follow the escalation chain until it is determined that an emergency response is necessary.
    2. Top-level organization management may determine that an emergency response is necessary and invoke this process directly.

Emergency response detail

1. Emergency response is initiated by escalation of a security event or be direct declaration by the CIO or other executive organization staff. The CIO may assign the incident coordinator, but by default, the coordinator will be the most senior security staff member available at the time of the incident.
2. The incident coordinator assembles the incident response team. The team meets using a pre-defined conference meeting space. One of the (CIO, CSO or Director IT) must attend each incident team meeting.
3. The meeting minutes capture the status, actions and resolution(s) for the incident. The incident coordinator reports on the cost, exposure and continuing business risk of the incident. The incident response team determines the next course of action: (go to 4, 5, or 6)
4. Lock-down and Repair – Perform the actions necessary to prevent further damage to the organization, repair impacted systems and perform changes to prevent a re-occurrence.

5. False Positive – The incident team determines this issue did not warrant an emergency response. The team provides a written report to senior management and the issue is handled as a normal incident, or closed.
6. Monitor and Capture – Perform a thorough investigation with continued monitoring to detect and capture the perpetrator. This process must include notification to the following senior and professional staff:
    1. CEO and CFO
    2. Corporate Attorney and Public Relations
7. Review and analyze log data to determine nature and scope of incident. This step would include utilizing virus, spyware, rootkit and other detection tools to determine necessary mitigation and repair.
8. Repair Systems, eliminate vector of attack mitigate exploitable vulnerabilities
9. The *Test Report* documents the validation of the repair process.
    1. Test Systems to ensure compliance with policy and risk mitigation.
    2. Perform additional repairs to resolve all current vulnerabilities.
10. Investigate incident to determine source of attack and capture perpetrator. This will require the use of forensics tools, log analysis, clean lab and dirty lab environments and possible communication with Law Enforcement or other outside entities.
11. The "Investigation Status Report" captures all current information regarding the incident. The Incident response team uses this information to determine the next course of action.

Definitions

First Responder/First level review

first person to be on scene or receive notification of an event, organizations should provide training to the first responder to recognize and properly react to emergency circumstances.

Help Desk Ticket (Control)

an electronic document captured in a database and issue tracking/resolution system

Ticket Owner

person reporting the event, the principal owner of the assets associated with the event or the common law or jurisdictional owner.

Second Tier

Senior technical resources assigned to resolve an escalated event.

Incident Coordinator

individual assigned by organization senior management to assemble the incident response team, manage and document response to the incident.

Investigation Status Report (Control)

documentation of the current investigation results, the coordinator may document this material in the ticket.

Meeting Minutes (Control)

documentation of the incident team meeting, the minutes document the attendees, current nature of the incident and the recommended actions. The coordinator may document this material in the ticket.

Lock-down Change Control

a process ordered as a resolution to the incident. This process follows the same authorization and response requirements as an Emergency Change Control.

Test Report (Control)

this report validates that IT personal have performed all necessary and available repairs to systems prior to bringing them back online.

War Room

a secure environment for review of confidential material and the investigation of a security incident.

Report to Senior Management (Control)

the *incident coordinator* is responsible for drafting a senior management report. The coordinator may document this material in the ticket

Chapter 7

Confused Deputy Problem and Cyber Security Standards

Confused deputy problem

A **confused deputy** is a computer program that is innocently fooled by some other party into misusing its authority. It is a specific type of privilege escalation. In information security, the **confused deputy problem** is often cited as an example of why capability-based security is important.

Example

In the original example of a confused deputy, there is a program that provides compilation services to other programs. Normally, the client program specifies the name of the input and output files, and the server is given the same access to those files that the client has.

The compiler service is pay-per-use, and the compiler program has access to a file (dubbed *BILL*) where it stores billing information. Clients obviously cannot write into the billing file.

Now suppose a client calls the service and specifies *BILL* as the name of the output file. The service opens the output file. Even though the client did not have access to that file, the service does, so the open succeeds, and the server writes the compilation output to the file, overwriting it, and thus destroying the billing information.

The confused deputy

In this example, the compilation service is the deputy because it is acting at the request of the client. It is confused because it was tricked into overwriting its billing file.

Whenever a program tries to access a file, the operating system needs to know two things: which file the program is asking for, and whether the program has permission to access the file. In the example, the file is designated by its name, "BILL". The server receives the file name from the client, but does not know whether the client had permission to write the file. When the server opens the file, the system uses the server's permission, not the client's. When the file name was passed from the client to the server, the permission did not go along with it; the permission was increased by the system silently and automatically.

It is not essential to the attack that the billing file is designated by a name represented as a string. The essential points are that:

- the designator for the file does not carry the full authority needed to access the file;
- the server's own permission to the file is used implicitly.

Other examples

A cross-site request forgery (CSRF) is an example of a confused deputy attack against a web browser. In this case a client's web browser has no means to distinguish the authority of the client from any authority of a "cross" site that the client is accessing.

Clickjacking is another category of web attacks that can be analysed as confused deputy attacks.

An FTP bounce attack can allow an attacker to indirectly connect to TCP ports that the attacker's machine has no access to, using a remote FTP server as the confused deputy.

Another example relates to personal firewall software. It can restrict internet access for specific applications. Some applications circumvent this by starting a browser with a specific URL. The browser has authority to open a network connection, even though the application does not. Firewall software can attempt to address this by prompting the user in cases where one program starts another which then accesses the network. However, the user frequently does not have sufficient information to determine whether such an access is legitimate -- false positives are common, and there is a substantial risk that even sophisticated users will become habituated to clicking 'ok' to these prompts. Confused deputy problem related with personal firewalls is just one of many instances of the confused deputy problem.

Not every program that misuses authority is a confused deputy. Sometimes misuse of authority is simply a result of a program error. The confused deputy problem occurs when the designation of an object is passed from one program to another, and the associated permission changes unintentionally, without any explicit action by either party. It is insidious because neither party did anything explicit to change the authority.

Solutions

In some systems, it is possible to ask the operating system to open a file using the permissions of another client. This solution has some drawbacks:

- It requires explicit attention to security by the server. A naive or careless server might not take this extra step.
- It becomes more difficult to identify the correct permission if the server is in turn the client of another service and wants to pass along access to the file.
- It requires the server to be trusted with the permissions of the client. Note that intersecting the server and client's permissions does not solve the problem either, because the server may then have to be given very wide permissions (all of the time, rather than those needed for a given request) in order to act for arbitrary clients.

The simplest way to solve the confused deputy problem is to bundle together the designation of an object and the permission to access that object. This is exactly what a capability is.

Using capability security in the compiler example, the client would pass to the server a capability to the output file, not the name of the file. Since it lacks a capability to the billing file, it cannot designate that file for output. In the cross-site request forgery example, a URL supplied "cross"-site would include its own authority independent of that of the client of the web browser (for example, by using a YURL).

Cyber security standards

Cyber security standards are security standards which enable organizations to practice safe security techniques to minimize the number of successful cyber security attacks. These guides provide general outlines as well as specific techniques for implementing cyber security. For certain specific standards, **cyber security certification** by an accredited body can be obtained. There are many advantages to obtaining certification including the ability to get cyber security insurance.

History

Cyber security standards have been created recently because sensitive information is now frequently stored on computers that are attached to the Internet. Also many tasks that were once done by hand are carried out by computer; therefore there is a need for Information Assurance (IA) and security. Cyber security is important in order to guard against identity theft. Businesses also have a need for cyber security because they need to protect their trade secrets, proprietary information, and personally identifiable information (PII) of their customers or employees. The government also has the need to secure its information. This is particularly critical since some terrorism acts are organized and facilitated by using the Internet. One of the most widely used security standards today is ISO/IEC 27002 which started in 1995. This standard consists of two basic parts. BS 7799 part 1 and BS 7799 part 2 both of which were created by (British Standards

Institute) BSI. Recently this standard has become ISO 27001. The National Institute of Standards and Technology (NIST) has released several special publications addressing cyber security. Three of these special papers are very relevant to cyber security: the 800-12 titled "Computer Security Handbook;" 800-14 titled "Generally Accepted Principles and Practices for Securing Information Technology;" and the 800-26 titled "Security Self-Assessment Guide for Information Technology Systems". The International Society of Automation (ISA) developed cyber security standards for industrial automation control systems (IACS) that are broadly applicable across manufacturing industries. The series of ISA industrial cyber security standards are known as ISA-99 and are being expanded to address new areas of concern.

ISO 27002

ISO 27002 incorporates both parts of the BS 7799 standard. Sometimes ISO/IEC 27002 is referred to as BS 7799 part 1 and sometimes it refers to part 1 and part 2. BS 7799 part 1 provides an outline for cyber security policy; whereas BS 7799 part 2 provides a certification. The outline is a high level guide to cyber security. It is most beneficial for an organization to obtain a certification to be recognized as compliant with the standard. The certification once obtained lasts three years and is periodically checked by the BSI to ensure an organization continues to be compliant throughout that three year period. ISO 27001 (ISMS) replaces BS 7799 part 2, but since it is backward compatible any organization working toward BS 7799 part 2 can easily transition to the ISO 27001 certification process. There is also a transitional audit available to make it easier once an organization is BS 7799 part 2-certified for the organization to become ISO 27001-certified. ISO/IEC 27002 states that information security is characterized by integrity, confidentiality, and availability. The ISO/IEC 27002 standard is arranged into eleven control areas; security policy, organizing information security, asset management, human resources security, physical and environmental security, communication and operations, access controls, information systems acquisition/development/maintenance, incident handling, business continuity management, compliance.

Standard of good practice

In the 1990s, the Information Security Forum (ISF) published a comprehensive list of best practices for information security, published as the *Standard of Good Practice* (SoGP). The ISF continues to update the SoGP every two years; the latest version was published in February 2007.

Originally the *Standard of Good Practice* was a private document available only to ISF members, but the ISF has since made the full document available to the general public at no cost.

Among other programs, the ISF offers its member organizations a comprehensive benchmarking program based on the SoGP.

NERC

The North American Electric Reliability Corporation (NERC) has created many standards. The most widely recognized is NERC 1300 which is a modification/update of NERC 1200. The newest version of NERC 1300 is called CIP-002-1 through CIP-009-2 (CIP=Critical Infrastructure Protection). These standards are used to secure bulk electric systems although NERC has created standards within other areas. The bulk electric system standards also provide network security administration while still supporting best practice industry processes.

NIST

1. Special publication 800-12 provides a broad overview of computer security and control areas. It also emphasizes the importance of the security controls and ways to implement them. Initially this document was aimed at the federal government although most practices in this document can be applied to the private sector as well. Specifically it was written for those people in the federal government responsible for handling sensitive systems.
2. Special publication 800-14 describes common security principles that are used. It provides a high level description of what should be incorporated within a computer security policy. It describes what can be done to improve existing security as well as how to develop a new security practice. Eight principles and fourteen practices are described within this document.
3. Special publication 800-26 provides advice on how to manage IT security. This document emphasizes the importance of self assessments as well as risk assessments.
4. Special publication 800-37, updated in 2010 provides a new risk approach: "Guide for Applying the Risk Management Framework to Federal Information Systems"
5. Special publication 800-53 "Guide for Assessing the Security Controls in Federal Information Systems" specifically addresses the 174 security controls that be applied to a system to make it "more secure."

ISO 15408

This standard develops what is called the “Common Criteria”. It allows many different software applications to be integrated and tested in a secure way.

RFC 2196

RFC 2196 is memorandum published by Internet Engineering Task Force for developing security policies and procedures for information systems connected on the Internet. The RFC 2196 provides a general and broad overview of information security including network security, incident response or security policies. The document is very practical and focusing on day-to-day operations.

ISA-99

ISA99 is the Industrial Automation and Control System Security Committee of the Instrumentation, Systems, and Automation Society (ISA). The committee is developing a multi-part control system standard and has released several standards and technical reports.

- ISA99 Part 1 (ANSI/ISA 99.00.01) is approved and published.
- ISA99 Part 2 (ANSI/ISA 99.02.01-2009) is approved and published. It has also been approved and published by the IEC as IEC 62443-2-1
- ISA99 Part 3 is in process
- ISA99 Part 4 is in process

ISA Security Compliance Institute

Related to the work of ISA 99 is the work of the ISA Security Compliance Institute. The ISA Security Compliance Institute (ISCI) has developed compliance test specifications for ISA99 and other control system security standards. They have also created an ANSI accredited certification program called ISASecure for the certification of industrial automation devices such as programmable logic controllers (PLC), distributed control systems (DCS) and safety instrumented systems (SIS). These types of devices provided automated control of industrial processes such as those found in the oil & gas, chemical, electric utility, manufacturing, food & beverage and water/wastewater processing industries. There is growing concern from both governments as well as private industry regarding the risk that these systems could be intentionally compromised by "evildoers" such as hackers, disgruntled employees, organized criminals, terrorist organizations or even state-sponsored groups. The recent news about the industrial control system malware known as Stuxnet has heightened concerns about the vulnerability of these systems.

Chapter 8

Data Loss Prevention Software

Data Loss Prevention (**DLP**) is a computer security term referring to systems that identify, monitor, and protect data in use (e.g., endpoint actions), data in motion (e.g., network actions), and data at rest (e.g., data storage) through deep content inspection, contextual security analysis of transaction (attributes of originator, data object, medium, timing, recipient/destination, etc.), and with a centralized management framework. The systems are designed to detect and prevent the unauthorized use and transmission of confidential information.

Vendors refer to the term as **Data Leak Prevention**, **Information Leak Detection and Prevention** (**ILDP**), **Information Leak Prevention** (**ILP**), **Content Monitoring and Filtering** (**CMF**), **Information Protection and Control** (**IPC**) or **Extrusion Prevention System** by analogy to Intrusion-prevention system.

Regulatory compliance

Many large companies now fall under oversight of government and commercial regulations that mandate controls over information, including HIPAA in health and benefits, GLBA and Basel_II in finance, and Payment Card Industry DSS standards. Some of these regulations stipulate a regular information technology audit, commonly known as IT audit, which organizations can fail if they lack suitable IT security controls and due-care (processes) standards. Companies with enterprise resource planning ERP software find compliance especially challenging. Others mandate significant penalties in the event of a breach.

New costs arising from breaches

Loss of large volumes of protected information has become a regular headline event, forcing companies to re-issue cards, notify customers, and mitigate loss of goodwill from negative publicity.

Types of DLP systems

Network DLP

Also referred to as gateway-based systems. These are usually dedicated hardware/software platforms, typically installed on the organization's internet network connection, that analyze network traffic to search for unauthorized information transmissions, including email, IM, FTP, HTTP, and HTTPS (called data in motion). They have the advantage that they are simple to install, and provide a relatively low cost of ownership. Network DLP systems can also discover data at rest (data stored throughout the enterprise) to identify areas of risk where confidential data is stored in inappropriate and/or unsecured locations.

Host-based DLP systems

Such systems run on end-user workstations or servers in the organization. Like network-based systems, host-based can address internal as well as external communications, and can therefore be used to control information flow between groups or types of users (e.g. 'Chinese walls'). They can also control email and Instant Messaging communications before they are stored in the corporate archive, such that a blocked communication (i.e., one that was never sent, and therefore not subject to retention rules) will not be identified in a subsequent legal discovery situation.

Host systems have the advantage that they can monitor and control access to physical devices (such as mobile devices with data storage capabilities) and in some cases can access information before it has been encrypted. Some host based systems can also provide application controls to block attempted transmissions of confidential information, and provide immediate feedback to the user. They have the disadvantage that they need to be installed on every workstation in the network, cannot be used on mobile devices (e.g., cell phones and PDAs) or where they cannot be practically installed (for example on a workstation in an internet café).

Data Identification

DLP solutions include a number of techniques for identifying confidential or sensitive information. Sometimes confused with discovery, data identification is a process by which organizations use a DLP technology to determine what to look for (in motion, at rest, or in use). DLP solutions use multiple methods for deep content analysis, ranging from keywords, dictionaries, and regular expressions to partial document matching and fingerprinting. The strength of the analysis engine directly correlates to its accuracy. The accuracy of DLP identification is important to lowering/avoiding false positives and negatives. Accuracy can depend on many variables, some of which may be situational or technological. Testing for accuracy is recommended to ensure a solution has virtually zero false positives/negatives.

Chapter 9

Dynamic SSL and Information Security

Dynamic SSL

Dynamic SSL is an endpoint security technology developed by Daniel McCann and Nima Sharifimehr of NetSecure Technologies Ltd. Dynamic SSL was created to solve the endpoint security problem in public networks by transparently correcting the implementation flaws in SSL systems that expose sensitive data to interception and tampering. Dynamic SSL is sometimes referred to as Dynamic TLS.

Endpoint vulnerabilities in SSL/TLS systems

Most implementations of SSL assume that the client computer is a secure environment for key negotiation, key storage, and encryption. This is untrue in principle and in practice, as malicious technologies such as Spyware, KeyJacking, and Man in the Browser have proven to be able to circumvent SSL by obtaining sensitive data prior to encryption. Furthermore, the reliance on the host PC for PKI certificate validation renders the infrastructure vulnerable to man-in-the-middle attacks.

Challenges for public networks

Traditional solutions to endpoint security rely on custom protocols or proprietary authentication architectures that are not interoperable with SSL. In many circumstances, particularly in anonymous or distributed environments where interoperability with SSL is a requirement, synchronization of client and server systems with a proprietary security protocol is simply not feasible. This is known as the Endpoint Security Problem in public networks.

In layman's terms, the Endpoint Security Problem essentially asserts that any anonymous transaction through a web browser must inherently be at risk, and that it cannot be fixed without removing the anonymity of the transaction. Since virtually all web transactions assume that the client is anonymous at the protocol level, this means that any proposed third-party solution will essentially break the system. This is a fundamental vulnerability that undermines the entire web security infrastructure, rendering virtually ever web transaction at risk.

Dynamic SSL solves this fundamental problem by focusing on transparently closing the implementation vulnerabilities rather than on redefining a new protocol. Therefore, the existing SSL system remains intact as the default secure communication protocol. However, implementation vulnerabilities are solved to achieve endpoint security.

Principles of Dynamic SSL

The underlying principle in Dynamic SSL is that encryption of sensitive information cannot be performed in an untrusted environment, such as most personal computers, where the security of the encryption process could be compromised. Rather, encryption of sensitive information must be done outside of the personal computer. Dynamic SSL assumes that the end user's computer is an untrusted environment, and can only be used as the channel to transmit sensitive information. Dynamic SSL thus guarantees the security of sensitive data by ensuring that it is never present in the insecure environment.

Implementations

Typical implementations involve two core components: a secure environment which hosts the sensitive data to be protected, and a cryptographic proxy, which securely redirects the encryption of the host process of the insecure environment to a cryptographic provider within the secure environment which hosts the sensitive data. An optional third component which controls the tokenization process can be inserted at the application layer. Nothing is required at the server end.

Generally speaking, the only change which is required on the endpoint computer is the replacement of the default SSL cryptographic provider with a Dynamic SSL cryptographic proxy. Most operating systems and web browsers support pluggable cryptographic providers, meaning that most implementations will require no changes whatsoever to the application on either end of the system.

Dynamic SSL works by using a tokenization system for sensitive data in conjunction with secure redirection of cryptographic operations to a secure environment. For example, rather than typing in your online banking password into a web browser, you would type in a meaningless token instead. When the form is submitted, an HTTP request containing the token is generated, and sent to the SSL cryptographic provider for encryption and secure transmission to the remote server. Instead of the encryption happening on the host PC, which may be compromised, the session is redirected to the secure environment (secure input device, server, etc..) which would contain your real online banking password. Inside the secure environment, the token within the HTTP request would get swapped out with your real online banking password at the moment of encryption. The encrypted packet, containing your real online banking password, would then be returned to the SSL system of the host PC for transmission to the remote server.

From the server's perspective, the request appears like any other regular keyed-in request. The packet was encrypted with the SSL session key negotiated with the client, and so is able to decrypt and process the packet as normal. It cannot tell the difference between a regular SSL transaction and a Dynamic SSL transaction.

The only difference in the transaction was that the sensitive data was never present on the host PC. Any malicious attempt to harvest the data from the host PC would be unable to locate the data.

A whitepaper is available which describes the process of Dynamic SSL in further detail.

Strengths and weaknesses

Dynamic SSL is the only known approach to endpoint security that requires no changes to existing server systems, and can therefore be used to transparently retrofit existing systems for endpoint security, while retaining the benefits of using a proven standard like SSL. By offloading cryptographic operations to a secure environment which acts as the point of origin for sensitive data, thereby ensuring the endpoint computer does not have access to said sensitive data, proactive protection from endpoint threats can in theory be achieved.

However, since Dynamic SSL is simply a process that is applied to SSL implementation, rather than a new protocol, it remains vulnerable to protocol vulnerabilities inherent within SSL, namely Man-in-the-Middle attacks. Sharifimehr has proposed a supplementary solution involving Man-in-the-Middle Protection for Dynamic SSL. His algorithm uses a combination of redundant cert verification and key tagging to prevent Man-in-the-Middle attacks and Keyjacking. Most known implementations of Dynamic SSL include Sharifimehr's additional process, described below:

Man-in-the-Middle protection

Known valid root certificates are digitally signed by an independent third party. When an X509 certificate arrives containing the authentication information for a remote website as part of the SSL authentication phase. Since the root certificate signatures are redundantly verified in a secure environment against a pre-verified list of valid digital signatures for known valid root certificates, this prevents any compromise via tampering of the certificate authentication chain. Phony certificates or certification chains can therefore be detected and the session rejected before it begins.

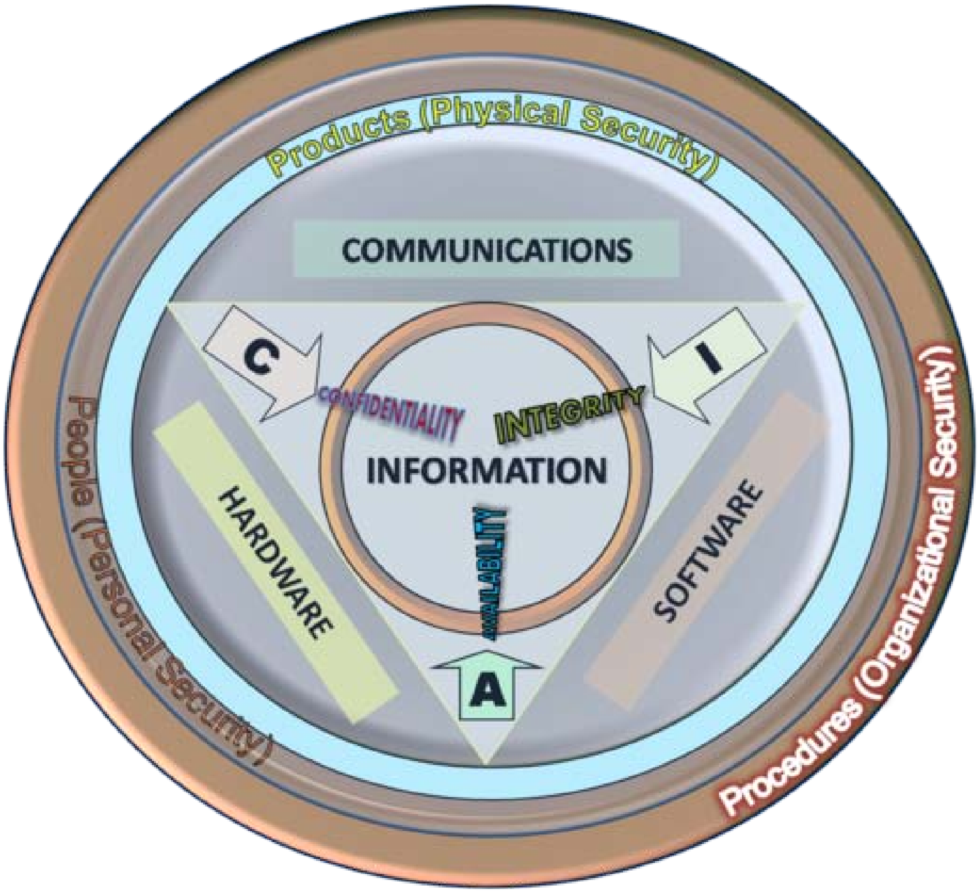
Keyjacking protection

Session and authentication keys are contextually bound to the operations which they are semantically required to perform, and may not be exported. An encryption key may not be exported and used to decrypt the ciphertext which is encrypted. In laymans terms, keys are "tagged" to ensure that they can never be exported for use outside of their intended use.

Commercial applications

A consumer product called SmartSwipe is the first known commercial application of Dynamic SSL. It claims to provide security against malware and other client-side attacks while providing universal support for virtually every eCommerce merchant that uses SSL. It is currently unknown whether other products are using this technology.

Information security


Products (Physical Security)
COMMUNICATIONS
C
I
CONFIDENTIALITY
INTEGRITY
INFORMATION
HARDWARE
SOFTWARE
AVAILABILITY
A
People (Personal Security)
Procedures (Organizational Security)

Information Security Components: or qualities, i.e., Confidentiality, Integrity and Availability (CIA). Information Systems are decomposed in three main portions, hardware, software and communications with the purpose to identify and apply information security industry standards, as mechanisms of protection and prevention, at three levels or layers: Physical, personal and organizational. Essentially, procedures or policies are implemented to tell people (administrators, users and operators)how to use products to ensure information security within the organizations.

Information security means protecting information and information systems from unauthorized access, use, disclosure, disruption, modification, perusal, inspection, recording or destruction.

The terms information security, computer security and information assurance are frequently incorrectly used interchangeably. These fields are interrelated often and share the common goals of protecting the confidentiality, integrity and availability of information; however, there are some subtle differences between them.

These differences lie primarily in the approach to the subject, the methodologies used, and the areas of concentration. Information security is concerned with the confidentiality, integrity and availability of data regardless of the form the data may take: electronic, print, or other forms.

Computer security can focus on ensuring the availability and correct operation of a computer system without concern for the information stored or processed by the computer.

Governments, military, corporations, financial institutions, hospitals, and private businesses amass a great deal of confidential information about their employees, customers, products, research, and financial status. Most of this information is now collected, processed and stored on electronic computers and transmitted across networks to other computers.

Should confidential information about a business' customers or finances or new product line fall into the hands of a competitor, such a breach of security could lead to lost business, law suits or even bankruptcy of the business. Protecting confidential information is a business requirement, and in many cases also an ethical and legal requirement.

For the individual, information security has a significant effect on privacy, which is viewed very differently in different cultures.

The field of information security has grown and evolved significantly in recent years. There are many ways of gaining entry into the field as a career. It offers many areas for specialization including: securing network(s) and allied infrastructure, securing applications and databases, security testing, information systems auditing, business continuity planning and digital forensics science, etc.

History

Since the early days of writing, heads of state and military commanders understood that it was necessary to provide some mechanism to protect the confidentiality of written correspondence and to have some means of detecting tampering.

Julius Caesar is credited with the invention of the Caesar cipher ca. 50 B.C., which was created in order to prevent his secret messages from being read should a message fall into the wrong hands.

World War II brought about many advancements in information security and marked the beginning of the professional field of information security.

The end of the 20th century and early years of the 21st century saw rapid advancements in telecommunications, computing hardware and software, and data encryption. The availability of smaller, more powerful and less expensive computing equipment made electronic data processing within the reach of small business and the home user. These computers quickly became interconnected through a network generically called the Internet or World Wide Web.

The rapid growth and widespread use of electronic data processing and electronic business conducted through the Internet, along with numerous occurrences of international terrorism, fueled the need for better methods of protecting the computers and the information they store, process and transmit. The academic disciplines of computer security, information security and information assurance emerged along with numerous professional organizations - all sharing the common goals of ensuring the security and reliability of information systems.

Basic principles

Key concepts

For over twenty years, information security has held confidentiality, integrity and availability (known as the CIA triad) to be the core principles of information security.

There is continuous debate about extending this classic trio. Other principles such as Accountability have sometimes been proposed for addition - it has been pointed out that issues such as Non-Repudiation do not fit well within the three core concepts, and as regulation of computer systems has increased (particularly amongst the Western nations) Legality is becoming a key consideration for practical security installations.

In 2002, Donn Parker proposed an alternative model for the classic CIA triad that he called the six atomic elements of information. The elements are confidentiality, possession, integrity, authenticity, availability, and utility. The merits of the Parkerian hexad are a subject of debate amongst security professionals.

Confidentiality

Confidentiality is the term used to prevent the disclosure of information to unauthorized individuals or systems. For example, a credit card transaction on the Internet requires the credit card number to be transmitted from the buyer to the merchant and from the merchant to a transaction processing network. The system attempts to enforce confidentiality by encrypting the card number during transmission, by limiting the places where it might appear (in databases, log files, backups, printed receipts, and so on), and by restricting access to the places where it is stored. If an unauthorized party obtains the card number in any way, a breach of confidentiality has occurred.

Breaches of confidentiality take many forms. Permitting someone to look over your shoulder at your computer screen while you have confidential data displayed on it could be a breach of confidentiality. If a laptop computer containing sensitive information about a company's employees is stolen or sold, it could result in a breach of confidentiality. Giving out confidential information over the telephone is a breach of confidentiality if the caller is not authorized to have the information.

Confidentiality is necessary (but not sufficient) for maintaining the privacy of the people whose personal information a system holds.

Integrity

In information security, integrity means that data cannot be modified undetectably. This is not the same thing as referential integrity in databases, although it can be viewed as a special case of Consistency as understood in the classic ACID model of transaction processing. Integrity is violated when a message is actively modified in transit. Information security systems typically provide message integrity in addition to data confidentiality.

Availability

For any information system to serve its purpose, the information must be available when it is needed. This means that the computing systems used to store and process the information, the security controls used to protect it, and the communication channels used to access it must be functioning correctly. High availability systems aim to remain available at all times, preventing service disruptions due to power outages, hardware failures, and system upgrades. Ensuring availability also involves preventing denial-of-service attacks.

Authenticity

In computing, e-Business and information security it is necessary to ensure that the data, transactions, communications or documents (electronic or physical) are genuine. It is also important for authenticity to validate that both parties involved are who they claim they are.

Non-repudiation

In law, non-repudiation implies one's intention to fulfill their obligations to a contract. It also implies that one party of a transaction cannot deny having received a transaction nor can the other party deny having sent a transaction.

Electronic commerce uses technology such as digital signatures and encryption to establish authenticity and non-repudiation.

Risk management

The CISA Review Manual 2006 provides the following definition of risk management: *"Risk management is the process of identifying vulnerabilities and threats to the information resources used by an organization in achieving business objectives, and deciding what countermeasures, if any, to take in reducing risk to an acceptable level, based on the value of the information resource to the organization."*

There are two things in this definition that may need some clarification. First, the *process* of risk management is an ongoing iterative process. It must be repeated indefinitely. The business environment is constantly changing and new threats and vulnerability emerge every day. Second, the choice of countermeasure (computer)s (controls) used to manage risks must strike a balance between productivity, cost, effectiveness of the countermeasure, and the value of the informational asset being protected.

Risk is the likelihood that something bad will happen that causes harm to an informational asset (or the loss of the asset). A **vulnerability** is a weakness that could be used to endanger or cause harm to an informational asset. A **threat** is anything (man made or act of nature) that has the potential to cause harm.

The likelihood that a threat will use a vulnerability to cause harm creates a risk. When a threat does use a vulnerability to inflict harm, it has an impact. In the context of information security, the impact is a loss of availability, integrity, and confidentiality, and possibly other losses (lost income, loss of life, loss of real property). It should be pointed out that it is not possible to identify all risks, nor is it possible to eliminate all risk. The remaining risk is called *residual risk*.

A risk assessment is carried out by a team of people who have knowledge of specific areas of the business. Membership of the team may vary over time as different parts of the business are assessed. The assessment may use a subjective qualitative analysis based on informed opinion, or where reliable dollar figures and historical information is available, the analysis may use quantitative analysis.

The research has shown that the most vulnerable point in most information systems is the human user, operator, designer, or other human The ISO/IEC 27002:2005 Code of practice for information security management recommends the following be examined during a risk assessment:

- security policy,
- organization of information security,
- asset management,
- human resources security,
- physical and environmental security,
- communications and operations management,
- access control,
- information systems acquisition, development and maintenance,
- information security incident management,
- business continuity management, and
- regulatory compliance.

In broad terms the risk management process consists of:

1. Identification of assets and estimating their value. Include: people, buildings, hardware, software, data (electronic, print, other), supplies.
2. Conduct a threat assessment. Include: Acts of nature, acts of war, accidents, malicious acts originating from inside or outside the organization.
3. Conduct a vulnerability assessment, and for each vulnerability, calculate the probability that it will be exploited. Evaluate policies, procedures, standards, training, physical security, quality control, technical security.
4. Calculate the impact that each threat would have on each asset. Use qualitative analysis or quantitative analysis.
5. Identify, select and implement appropriate controls. Provide a proportional response. Consider productivity, cost effectiveness, and value of the asset.
6. Evaluate the effectiveness of the control measures. Ensure the controls provide the required cost effective protection without discernible loss of productivity.

For any given risk, Executive Management can choose to **accept the risk** based upon the relative low value of the asset, the relative low frequency of occurrence, and the relative low impact on the business. Or, leadership may choose to **mitigate the risk** by selecting and implementing appropriate control measures to reduce the risk. In some cases, the risk can be **transferred** to another business by buying insurance or out-sourcing to another business. The reality of some risks may be disputed. In such cases leadership may choose to **deny the risk**. This is itself a potential risk.

Controls

When Management chooses to mitigate a risk, they will do so by implementing one or more of three different types of controls.

Administrative

Administrative controls (also called procedural controls) consist of approved written policies, procedures, standards and guidelines. Administrative controls form the framework for running the business and managing people. They inform people on how the business is to be run and how day to day operations are to be conducted. Laws and regulations created by government bodies are also a type of administrative control because they inform the business. Some industry sectors have policies, procedures, standards and guidelines that must be followed - the Payment Card Industry (PCI) Data Security Standard required by Visa and Master Card is such an example. Other examples of administrative controls include the corporate security policy, password policy, hiring policies, and disciplinary policies.

Administrative controls form the basis for the selection and implementation of logical and physical controls. Logical and physical controls are manifestations of administrative controls. Administrative controls are of paramount importance.

Logical

Logical controls (also called technical controls) use software and data to monitor and control access to information and computing systems. For example: passwords, network and host based firewalls, network intrusion detection systems, access control lists, and data encryption are logical controls.

An important logical control that is frequently overlooked is the **principle of least privilege**. The principle of least privilege requires that an individual, program or system process is not granted any more access privileges than are necessary to perform the task. A blatant example of the failure to adhere to the principle of least privilege is logging into Windows as user Administrator to read Email and surf the Web. Violations of this principle can also occur when an individual collects additional access privileges over time. This happens when employees' job duties change, or they are promoted to a new position, or they transfer to another department. The access privileges required by their new duties are frequently added onto their already existing access privileges which may no longer be necessary or appropriate.

Security classification for information

An important aspect of information security and risk management is recognizing the value of information and defining appropriate procedures and protection requirements for the information. Not all information is equal and so not all information requires the same degree of protection. This requires information to be assigned a security classification.

The first step in information classification is to identify a member of senior management as the owner of the particular information to be classified. Next, develop a classification policy. The policy should describe the different classification labels, define the criteria for information to be assigned a particular label, and list the required security controls for each classification.

Some factors that influence which classification information should be assigned include how much value that information has to the organization, how old the information is and whether or not the information has become obsolete. Laws and other regulatory requirements are also important considerations when classifying information.

The type of information security classification labels selected and used will depend on the nature of the organisation, with examples being:

- In the business sector, labels such as: **Public, Sensitive, Private, Confidential**.
- In the government sector, labels such as: **Unclassified**, **Sensitive But Unclassified**, **Restricted**, **Confidential**, **Secret**, **Top Secret** and their non-English equivalents.
- In cross-sectoral formations, the Traffic Light Protocol, which consists of: **White, Green, Amber** and **Red**.

All employees in the organization, as well as business partners, must be trained on the classification schema and understand the required security controls and handling procedures for each classification. The classification a particular information asset has been assigned should be reviewed periodically to ensure the classification is still appropriate for the information and to ensure the security controls required by the classification are in place.

Access control

Access to protected information must be restricted to people who are authorized to access the information. The computer programs, and in many cases the computers that process the information, must also be authorized. This requires that mechanisms be in place to control the access to protected information. The sophistication of the access control mechanisms should be in parity with the value of the information being protected - the more sensitive or valuable the information the stronger the control mechanisms need to be. The foundation on which access control mechanisms are built start with identification and authentication.

Identification is an assertion of who someone is or what something is. If a person makes the statement *"Hello, my name is John Doe."* they are making a claim of who they are. However, their claim may or may not be true. Before John Doe can be granted access to protected information it will be necessary to verify that the person claiming to be John Doe really is John Doe.

Authentication is the act of verifying a claim of identity. When John Doe goes into a bank to make a withdrawal, he tells the bank teller he is John Doe (a claim of identity). The bank teller asks to see a photo ID, so he hands the teller his driver's license. The bank teller checks the license to make sure it has John Doe printed on it and compares the photograph on the license against the person claiming to be John Doe. If the photo and name match the person, then the teller has authenticated that John Doe is who he claimed to be.

There are three different types of information that can be used for authentication: **something you know, something you have, or something you are.** Examples of *something you know* include such things as a PIN, a password, or your mother's maiden name. Examples of *something you have* include a driver's license or a magnetic swipe card. *Something you are* refers to biometrics. Examples of biometrics include palm prints, finger prints, voice prints and retina (eye) scans. Strong authentication requires providing information from two of the three different types of authentication information. For example, something you know plus something you have. This is called two factor authentication.

On computer systems in use today, the Username is the most common form of identification and the Password is the most common form of authentication. Usernames and passwords have served their purpose but in our modern world they are no longer adequate. Usernames and passwords are slowly being replaced with more sophisticated authentication mechanisms.

After a person, program or computer has successfully been identified and authenticated then it must be determined what informational resources they are permitted to access and what actions they will be allowed to perform (run, view, create, delete, or change). This is called **authorization**.

Authorization to access information and other computing services begins with administrative policies and procedures. The policies prescribe what information and computing services can be accessed, by whom, and under what conditions. The access control mechanisms are then configured to enforce these policies.

Different computing systems are equipped with different kinds of access control mechanisms - some may even offer a choice of different access control mechanisms. The access control mechanism a system offers will be based upon one of three approaches to access control or it may be derived from a combination of the three approaches.

The **non-discretionary** approach consolidates all access control under a centralized administration. The access to information and other resources is usually based on the individuals function (role) in the organization or the tasks the individual must perform. The **discretionary approach** gives the creator or owner of the information resource the ability to control access to those resources. In the **Mandatory access control approach**, access is granted or denied basing upon the security classification assigned to the information resource.

Examples of common access control mechanisms in use today include Role-based access control available in many advanced Database Management Systems, simple file permissions provided in the UNIX and Windows operating systems, Group Policy Objects provided in Windows network systems, Kerberos, RADIUS, TACACS, and the simple access lists used in many firewalls and routers.

To be effective, policies and other security controls must be enforceable and upheld. Effective policies ensure that people are held **accountable** for their actions. All failed and successful authentication attempts must be logged, and all access to information must leave some type of audit trail.

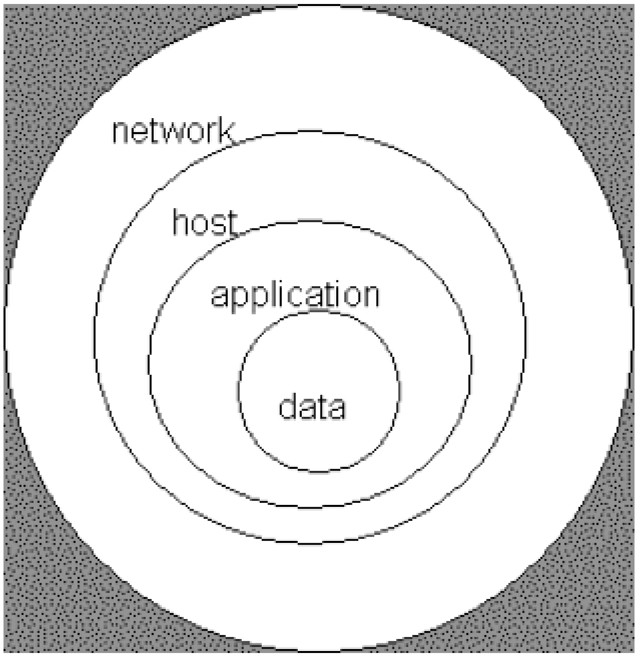
Cryptography

Information security uses cryptography to transform usable information into a form that renders it unusable by anyone other than an authorized user; this process is called encryption. Information that has been encrypted (rendered unusable) can be transformed back into its original usable form by an authorized user, who possesses the cryptographic key, through the process of decryption. Cryptography is used in information security to protect information from unauthorized or accidental disclosure while the information is in transit (either electronically or physically) and while information is in storage.

Cryptography provides information security with other useful applications as well including improved authentication methods, message digests, digital signatures, non-repudiation, and encrypted network communications. Older less secure application such as telnet and ftp are slowly being replaced with more secure applications such as ssh that use encrypted network communications. Wireless communications can be encrypted using protocols such as WPA/WPA2 or the older (and less secure) WEP. Wired communications (such as ITU-T G.hn) are secured using AES for encryption and X.1035 for authentication and key exchange. Software applications such as GnuPG or PGP can be used to encrypt data files and Email.

Cryptography can introduce security problems when it is not implemented correctly. Cryptographic solutions need to be implemented using industry accepted solutions that have undergone rigorous peer review by independent experts in cryptography. The length and strength of the encryption key is also an important consideration. A key that is weak or too short will produce weak encryption. The keys used for encryption and decryption must be protected with the same degree of rigor as any other confidential information. They must be protected from unauthorized disclosure and destruction and they must be available when needed. PKI solutions address many of the problems that surround key management.

Defense in depth



Information security must protect information throughout the life span of the information, from the initial creation of the information on through to the final disposal of the information. The information must be protected while in motion and while at rest. During its life time, information may pass through many different information processing systems and through many different parts of information processing systems. There are many different ways the information and information systems can be threatened. To fully protect the information during its lifetime, each component of the information processing system must have its own protection mechanisms. The building up, layering on and overlapping of security measures is called defense in depth. The strength of any system is no greater than its weakest link. Using a defence in depth strategy, should one defensive measure fail there are other defensive measures in place that continue to provide protection.

Recall the earlier discussion about administrative controls, logical controls, and physical controls. The three types of controls can be used to form the basis upon which to build a defense-in-depth strategy. With this approach, defense-in-depth can be conceptualized as three distinct layers or planes laid one on top of the other. Additional insight into defense-in- depth can be gained by thinking of it as forming the layers of an onion, with data at the core of the onion, people as the outer layer of the onion, and network security, host-based security and application security forming the inner layers of the onion. Both perspectives are equally valid and each provides valuable insight into the implementation of a good defense-in-depth strategy.

Process

The terms **reasonable and prudent person**, **due care** and **due diligence** have been used in the fields of Finance, Securities, and Law for many years. In recent years these terms have found their way into the fields of computing and information security. U.S.A. Federal Sentencing Guidelines now make it possible to hold corporate officers liable for failing to exercise due care and due diligence in the management of their information systems.

In the business world, stockholders, customers, business partners and governments have the expectation that corporate officers will run the business in accordance with accepted business practices and in compliance with laws and other regulatory requirements. This is often described as the "reasonable and prudent person" rule. A prudent person takes due care to ensure that everything necessary is done to operate the business by sound business principles and in a legal ethical manner. A prudent person is also diligent (mindful, attentive, and ongoing) in their due care of the business.

In the field of Information Security, Harris offers the following definitions of **due care** and **due diligence**:

"Due care are steps that are taken to show that a company has taken responsibility for the activities that take place within the corporation and has taken the necessary steps to help protect the company, its resources, and employees." And, [Due diligence are the] *"continual activities that make sure the protection mechanisms are continually maintained and operational."*

Attention should be made to two important points in these definitions. First, in due care, steps are taken to ***show*** - this means that the steps can be verified, measured, or even produce tangible artifacts. Second, in due diligence, there are **continual activities** - this means that people are actually doing things to monitor and maintain the protection mechanisms, and these activities are ongoing.

Security governance

The Software Engineering Institute at Carnegie Mellon University, in a publication titled "Governing for Enterprise Security (GES)", defines characteristics of effective security governance. These include:

- An enterprise-wide issue
- Leaders are accountable
- Viewed as a business requirement
- Risk-based
- Roles, responsibilities, and segregation of duties defined
- Addressed and enforced in policy
- Adequate resources committed
- Staff aware and trained

- A development life cycle requirement
- Planned, managed, measurable, and measured
- Reviewed and audited

Incident response plans

1 to 3 paragraphs (non technical) that discuss:

- Selecting team members
- Define roles, responsibilities and lines of authority
- Define a security incident
- Define a reportable incident
- Training
- Detection
- Classification
- Escalation
- Containment
- Eradication
- Documentation

Change management

Change management is a formal process for directing and controlling alterations to the information processing environment. This includes alterations to desktop computers, the network, servers and software. The objectives of change management are to reduce the risks posed by changes to the information processing environment and improve the stability and reliability of the processing environment as changes are made. It is not the objective of change management to prevent or hinder necessary changes from being implemented.

Any change to the information processing environment introduces an element of risk. Even apparently simple changes can have unexpected effects. One of Managements many responsibilities is the management of risk. Change management is a tool for managing the risks introduced by changes to the information processing environment. Part of the change management process ensures that changes are not implemented at inopportune times when they may disrupt critical business processes or interfere with other changes being implemented.

Not every change needs to be managed. Some kinds of changes are a part of the everyday routine of information processing and adhere to a predefined procedure, which reduces the overall level of risk to the processing environment. Creating a new user account or deploying a new desktop computer are examples of changes that do not generally require change management. However, relocating user file shares, or upgrading the Email server pose a much higher level of risk to the processing environment and are not a normal everyday activity. The critical first steps in change management are (a) defining change (and communicating that definition) and (b) defining the scope of the change system.

Change management is usually overseen by a Change Review Board composed of representatives from key business areas, security, networking, systems administrators, Database administration, applications development, desktop support and the help desk. The tasks of the Change Review Board can be facilitated with the use of automated work flow application. The responsibility of the Change Review Board is to ensure the organizations documented change management procedures are followed. The change management process is as follows:

- **Requested:** Anyone can request a change. The person making the change request may or may not be the same person that performs the analysis or implements the change. When a request for change is received, it may undergo a preliminary review to determine if the requested change is compatible with the organizations business model and practices, and to determine the amount of resources needed to implement the change.

- **Approved:** Management runs the business and controls the allocation of resources therefore, Management must approve requests for changes and assign a priority for every change. Management might choose to reject a change request if the change is not compatible with the business model, industry standards or best practices. Management might also choose to reject a change request if the change requires more resources than can be allocated for the change.

- **Planned:** Planning a change involves discovering the scope and impact of the proposed change; analyzing the complexity of the change; allocation of resources and, developing, testing and documenting both implementation and backout plans. Need to define the criteria on which a decision to back out will be made.

- **Tested:** Every change must be tested in a safe test environment, which closely reflects the actual production environment, before the change is applied to the production environment. The backout plan must also be tested.

- **Scheduled:** Part of the change review board's responsibility is to assist in the scheduling of changes by reviewing the proposed implementation date for potential conflicts with other scheduled changes or critical business activities.

- **Communicated:** Once a change has been scheduled it must be communicated. The communication is to give others the opportunity to remind the change review board about other changes or critical business activities that might have been overlooked when scheduling the change. The communication also serves to make the Help Desk and users aware that a change is about to occur. Another responsibility of the change review board is to ensure that scheduled changes have been properly communicated to those who will be affected by the change or otherwise have an interest in the change.

- **Implemented:** At the appointed date and time, the changes must be implemented. Part of the planning process was to develop an implementation plan, testing plan and, a back out plan. If the implementation of the change should fail or, the post implementation testing fails or, other "drop dead" criteria have been met, the back out plan should be implemented.

- **Documented:** All changes must be documented. The documentation includes the initial request for change, its approval, the priority assigned to it, the implementation, testing and back out plans, the results of the change review board critique, the date/time the change was implemented, who implemented it, and whether the change was implemented successfully, failed or postponed.

- **Post change review:** The change review board should hold a post implementation review of changes. It is particularly important to review failed and backed out changes. The review board should try to understand the problems that were encountered, and look for areas for improvement.

Change management procedures that are simple to follow and easy to use can greatly reduce the overall risks created when changes are made to the information processing environment. Good change management procedures improve the over all quality and success of changes as they are implemented. This is accomplished through planning, peer review, documentation and communication.

ISO/IEC 20000, The Visible OPS Handbook: Implementing ITIL in 4 Practical and Auditable Steps (Full book summary), and Information Technology Infrastructure Library all provide valuable guidance on implementing an efficient and effective change management program. information security

Business continuity

Business continuity is the mechanism by which an organization continues to operate its critical business units, during planned or unplanned disruptions that affect normal business operations, by invoking planned and managed procedures.

Unlike what most people think business continuity is not necessarily an IT system or process, simply because it is about the business. Today disasters or disruptions to business are a reality. Whether the disaster is natural or man-made (the TIME magazine has a website on the top 10), it affects normal life and so business. So why is planning so important? Let us face reality that "all businesses recover", whether they planned for recovery or not, simply because business is about earning money for survival.

The planning is merely getting better prepared to face it, knowing fully well that the best plans may fail. Planning helps to reduce cost of recovery, operational overheads and most importantly sail through some smaller ones effortlessly.

For businesses to create effective plans they need to focus upon the following key questions. Most of these are common knowledge, and anyone can do a BCP.

1. Should a disaster strike, what are the first few things that I should do? Should I call people to find if they are OK or call up the bank to figure out my money is safe? This is Emergencey Response. Emergency Response services help take the first hit when the disaster strikes and if the disaster is serious enough the Emergency Response teams need to quickly get a Crisis Management team in place.
2. What parts of my business should I recover first? The one that brings me most money or the one where I spend the most, or the one that will ensure I shall be able to get sustained future growth? The identified sections are the critical business units. There is no magic bullet here, no one answer satisfies all. Businesses need to find answers that meet business requirements.
3. How soon should I target to recover my critical business units? In BCP technical jargon this is called Recovery Time Objective, or RTO. This objective will define what costs the business will need to spend to recover from a disruption. For example, it is cheaper to recover a business in 1 day than in 1 hour.
4. What all do I need to recover the business? IT, machinery, records...food, water, people...So many aspects to dwell upon. The cost factor becomes clearer now...Business leaders need to drive business continuity. Hold on. My IT manager spent $200000 last month and created a DRP (Disaster Recovery Plan), whatever happened to that? a DRP is about continuing an IT system, and is one of the sections of a comprehensive Business Continuity Plan. Look below for more on this.
5. And where do I recover my business from... Will the business center give me space to work, or would it be flooded by many people queuing up for the same reasons that I am.
6. But once I do recover from the disaster and work in reduced production capacity, since my main operational sites are unavailable, how long can this go on. How long can I do without my original sites, systems, people? this defines the amount of business resilience a business may have.
7. Now that I know how to recover my business. How do I make sure my plan works? Most BCP pundits would recommend testing the plan at least once a year, reviewing it for adequacy and rewriting or updating the plans either annually or when businesses change.

Disaster recovery planning

While a business continuity plan (BCP) takes a broad approach to dealing with organizational-wide effects of a disaster, a disaster recovery plan (DRP), which is a subset of the business continuity plan, is instead focused on taking the necessary steps to resume normal business operations as quickly as possible. A disaster recovery plan is executed immediately after the disaster occurs and details what steps are to be taken in order to recover critical information technology infrastructure.

Laws and regulations

*Below is a **partial** listing of European, United Kingdom, Canadian and USA governmental laws and regulations that have, or will have, a significant effect on data processing and information security. Important industry sector regulations have also been included when they have a significant impact on information security.*

- UK Data Protection Act 1998 makes new provisions for the regulation of the processing of information relating to individuals, including the obtaining, holding, use or disclosure of such information. The European Union Data Protection Directive (EUDPD) requires that all EU member must adopt national regulations to standardize the protection of data privacy for citizens throughout the EU.

- The Computer Misuse Act 1990 is an Act of the UK Parliament making computer crime (e.g. cracking - sometimes incorrectly referred to as hacking) a criminal offence. The Act has become a model upon which several other countries including Canada and the Republic of Ireland have drawn inspiration when subsequently drafting their own information security laws.
- EU Data Retention laws requires Internet service providers and phone companies to keep data on every electronic message sent and phone call made for between six months and two years.
- The Family Educational Rights and Privacy Act (FERPA) (20 U.S.C. § 1232 g; 34 CFR Part 99) is a USA Federal law that protects the privacy of student education records. The law applies to all schools that receive funds under an applicable program of the U.S. Department of Education. Generally, schools must have written permission from the parent or eligible student in order to release any information from a student's education record.

- Health Insurance Portability and Accountability Act (HIPAA) of 1996 requires the adoption of national standards for electronic health care transactions and national identifiers for providers, health insurance plans, and employers. And, it requires health care providers, insurance providers and employers to safeguard the security and privacy of health data.
- Gramm-Leach-Bliley Act of 1999 (GLBA), also known as the Financial Services Modernization Act of 1999, protects the privacy and security of private financial information that financial institutions collect, hold, and process.

- Sarbanes-Oxley Act of 2002 (SOX). Section 404 of the act requires publicly traded companies to assess the effectiveness of their internal controls for financial reporting in annual reports they submit at the end of each fiscal year. Chief information officers are responsible for the security, accuracy and the reliability of the systems that manage and report the financial data. The act also requires publicly traded companies to engage independent auditors who must attest to, and report on, the validity of their assessments.
- Payment Card Industry Data Security Standard (PCI DSS) establishes comprehensive requirements for enhancing payment account data security. It was

developed by the founding payment brands of the PCI Security Standards Council, including American Express, Discover Financial Services, JCB, MasterCard Worldwide and Visa International, to help facilitate the broad adoption of consistent data security measures on a global basis. The PCI DSS is a multifaceted security standard that includes requirements for security management, policies, procedures, network architecture, software design and other critical protective measures.

- State Security Breach Notification Laws (California and many others) require businesses, nonprofits, and state institutions to notify consumers when unencrypted "personal information" may have been compromised, lost, or stolen.
- Personal Information Protection and Electronics Document Act (PIPEDA) - An Act to support and promote electronic commerce by protecting personal information that is collected, used or disclosed in certain circumstances, by providing for the use of electronic means to communicate or record information or transactions and by amending the Canada Evidence Act, the Statutory Instruments Act and the Statute Revision Act.

Sources of standards

International Organization for Standardization (ISO) is a consortium of national standards institutes from 157 countries with a Central Secretariat in Geneva Switzerland that coordinates the system. The ISO is the world's largest developer of standards. The ISO-15443: "Information technology - Security techniques - A framework for IT security assurance", ISO-27002 (previously ISO-17799): "Information technology - Security techniques - Code of practice for information security management", ISO-20000: "Information technology - Service management", and ISO-27001: "Information technology - Security techniques - Information security management systems" are of particular interest to information security professionals.

The USA National Institute of Standards and Technology (NIST) is a non-regulatory federal agency within the U.S. Department of Commerce. The NIST Computer Security Division develops standards, metrics, tests and validation programs as well as publishes standards and guidelines to increase secure IT planning, implementation, management and operation. NIST is also the custodian of the USA Federal Information Processing Standard publications (FIPS).

The Internet Society is a professional membership society with more than 100 organization and over 20,000 individual members in over 180 countries. It provides leadership in addressing issues that confront the future of the Internet, and is the organization home for the groups responsible for Internet infrastructure standards, including the Internet Engineering Task Force (IETF) and the Internet Architecture Board (IAB). The ISOC hosts the Requests for Comments (RFCs) which includes the Official Internet Protocol Standards and the RFC-2196 Site Security Handbook.

The Information Security Forum is a global nonprofit organization of several hundred leading organizations in financial services, manufacturing, telecommunications, consumer goods, government, and other areas. It provides research into best practice and practice advice summarized in its biannual Standard of Good Practice, incorporating detail specifications across many areas.

The IT Baseline Protection Catalogs, or IT-Grundschutz Catalogs, ("IT Baseline Protection Manual" before 2005) are a collection of documents from the German Federal Office for Security in Information Technology (FSI), useful for detecting and combating security-relevant weak points in the IT environment ("IT cluster"). The collection encompasses over 3000 pages with the introduction and catalogs.

Professionalism

In 1989, Carnegie Mellon University established the Information Networking Institute, the United States' first research and education center devoted to information networking. The academic disciplines of computer security, information security and information assurance emerged along with numerous professional organizations during the later years of the 20th century and early years of the 21st century.

Entry into the field can be accomplished through self-study, college or university schooling in the field, or through week long focused training camps. Many colleges, universities and training companies offer many of their programs on- line. The GIAC-GSEC and Security+ certifications are both entry level security certifications. Membership of the Institute of Information Security Professionals (IISP) is gaining traction in the U.K. as the professional standard for Information Security Professionals.

The Certified Information Systems Security Professional (CISSP) is a mid- to senior-level information security certification. The Information Systems Security Architecture Professional (ISSAP), Information Systems Security Engineering Professional (ISSEP), Information Systems Security Management Professional (ISSMP), and Certified Information Security Manager (CISM) certifications are well-respected advanced certifications in information-security architecture, engineering, and management respectively.

Within the UK a recognised senior level information security certification is provided by CESG.

CLAS is the CESG Listed Adviser Scheme - a partnership linking the unique Information Assurance knowledge of CESG with the expertise and resources of the private sector.

CESG recognises that there is an increasing demand for authoritative Information Assurance advice and guidance. This demand has come as a result of an increasing awareness of the threats and vulnerabilities that information systems are likely to face in an ever-changing world.

The Scheme aims to satisfy this demand by creating a pool of high quality consultants approved by CESG to provide Information Assurance advice to government departments and other organisations who provide vital services for the United Kingdom.

CLAS consultants are approved to provide Information Assurance advice on systems processing protectively marked information up to, and including, SECRET. Potential customers of the CLAS Scheme should also note that if the information is not protectively marked then they do not need to specify membership of CLAS in their invitations to tender, and may be challenged if equally competent non-scheme members are prevented from bidding.

The profession of information security has seen an increased demand for security professionals who are experienced in network security auditing, penetration testing, and digital forensics investigation. In addition, many smaller companies have cropped up as the result of this increased demand in information security training and consulting.

Conclusion

Information security is the ongoing process of exercising due care and due diligence to protect information, and information systems, from unauthorized access, use, disclosure, destruction, modification, or disruption or distribution. The never ending process of information security involves ongoing training, assessment, protection, monitoring & detection, incident response & repair, documentation, and review. This makes information security an indispensable part of all the business operations across different domains.

Chapter 10

Information Assurance

Information assurance (IA) is the practice of managing risks related to the use, processing, storage, and transmission of information or data and the systems and processes used for those purposes. While focused dominantly on information in digital form, the full range of IA encompasses not only digital but also analog or physical form. Information assurance as a field has grown from the practice of information security which in turn grew out of practices and procedures of computer security.

There are three models used in the practice of IA to define assurance requirements and assist in covering all necessary aspects or attributes.

The first is the classic information security model, also called the CIA Triad, which addresses three attributes of information and information systems, confidentiality, integrity, and availability. This C-I-A model is extremely useful for teaching introductory and basic concepts of information security and assurance; the initials are an easy mnemonic to remember, and when properly understood, can prompt systems designers and users to address the most pressing aspects of assurance.

The next most widely known model is the Five Pillars of IA model, promulgated by the U.S. Department of Defense (DoD) in a variety of publications, beginning with the National Information Assurance Glossary, Committee on National Security Systems Instruction CNSSI-4009. Here is the definition from that publication: "Measures that protect and defend information and information systems by ensuring their availability, integrity, authentication, confidentiality, and non-repudiation. These measures include providing for restoration of information systems by incorporating protection, detection, and reaction capabilities." The Five Pillars model is sometimes criticized because authentication and non-repudiation are not attributes of information or systems; rather, they are procedures or methods useful to assure the integrity and authenticity of information, and to protect the confidentiality of those same.

The third IA model, less widely known but considered by many IA practitioners and professionals to be the most complete and accurate of the three, is the Parkerian Hexad, first introduced by Donn B. Parker in 1998. Like the Five Pillars, Parker's hexad begins with the C-I-A model but builds it out by adding three more attributes of authenticity, utility, and possession (or control). It is significant to point out that the concept or attribute of authenticity, as described by Parker, is not identical to the pillar of authentication as described by the U.S. DoD.

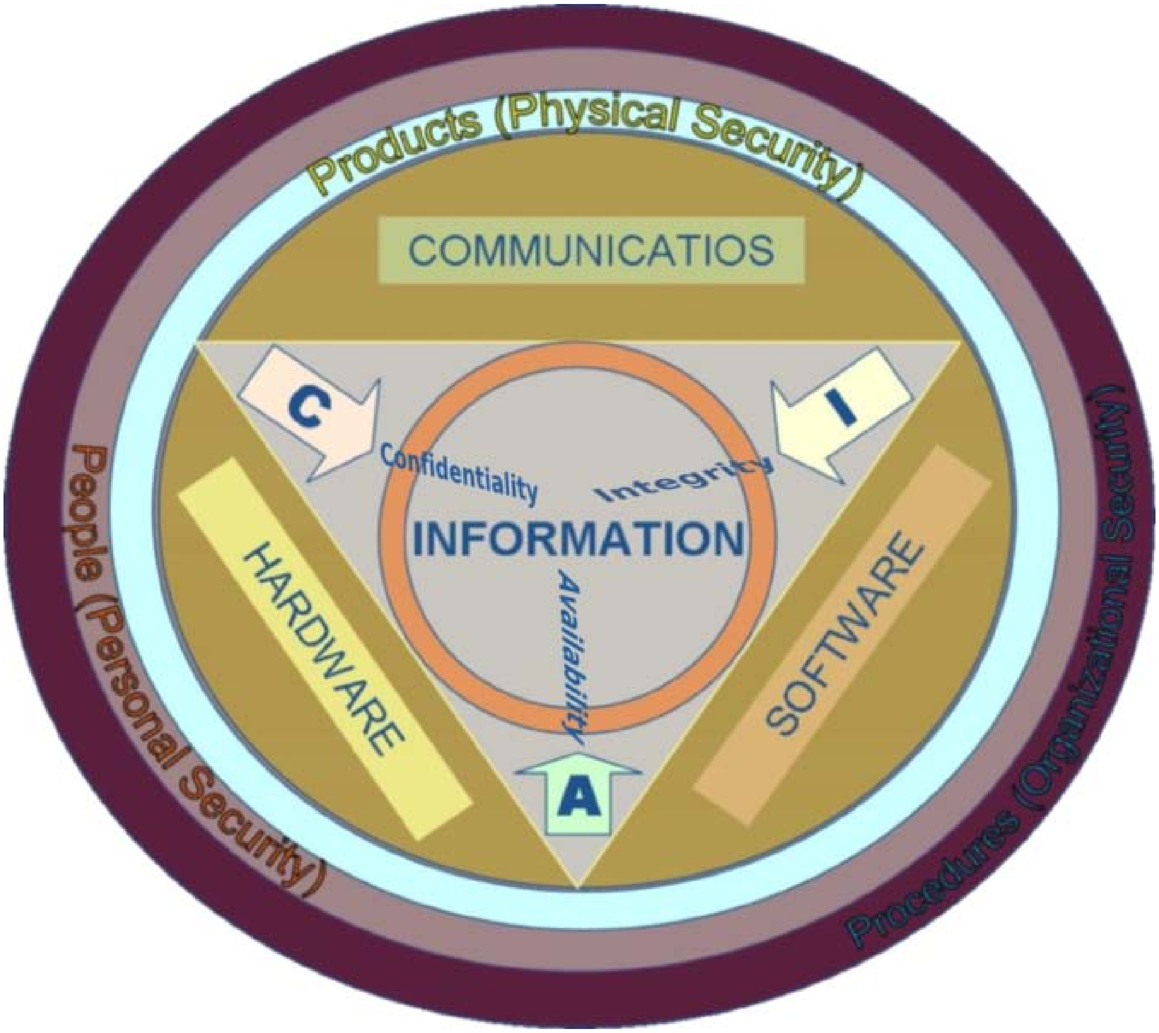
Overview

Information assurance is closely related to information security and the terms are sometimes used interchangeably. However, IA's broader connotation also includes reliability and emphasizes strategic risk management over tools and tactics. In addition to defending against malicious hackers and code (e.g., viruses), IA includes other corporate governance issues such as privacy, compliance, audits, business continuity, and disaster recovery. Further, while information security draws primarily from computer science, IA is interdisciplinary and draws from multiple fields, including accounting, fraud examination, forensic science, management science, systems engineering, security engineering, and criminology, in addition to computer science. Therefore, IA is best thought of as a superset of information security.

History

In the 1960s, IA was not as complex as it is today. IA was as simple as controlling access to the computer room by locking the door and placing guards to protect it.

IA Concepts


Products (Physical Security)
COMMUNICATIOS
C
I
Confidentiality
Integrity
INFORMATION
Availability
A
HARDWARE
SOFTWARE
People (Personal Security)
Procedures (Organizational Security)

Model of integrated CIA triad, Defense-in-Depth strategies.

Since the 1970s, information security has held confidentiality, integrity and availability (known as the CIA triad) as the core principles. One newer model of Information Assurance adds Authentication and Non-repudiation to create the 5 Pillars of IA. In contrast, Donn B. Parker developed a model that added three attributes of authenticity, utility, and possession to the core C-I-A. The work in which Parker introduced this model.

Confidentiality

CNSSI-4009: "Assurance that information is not disclosed to unauthorized individuals, processes, or devices. "

Confidential information must only be accessed, used, copied, or disclosed by users who have been authorized, and only when there is a genuine need. A confidentiality breach occurs when information or information systems have been, or may have been, accessed, used, copied, or disclosed, or by someone who was not authorized to have access to the information.

For example: Permitting someone to look over your shoulder at your computer screen while you have confidential data displayed on it would be a breach of confidentiality if they were not authorized to have the information. If a laptop computer, which contains employment and benefit information about 100,000 employees, is stolen from a car (or is sold on eBay) could result in a breach of confidentiality because the information is now in the hands of someone who is not authorized to have it. Giving out confidential information over the telephone is a breach of confidentiality if the caller is not authorized to have the information.

Data confidentially refers to the attempt to keep information away from unauthorized people or systems. All information has a releasability, or confidentiality level. Data can be labeled in a wide range from being available and open to the public, i.e. newspapers and non-secure web pages, to sensitive compartmental information. For IA, confidentiality refers to the steps taken to ensure that confidential information is only accessed or disclosed to people who have been authorized. Even then, the information should only be accessed by those with a genuine need to view it. When talking about data confidentiality there are a few things that need to be considered. If sensitive data is lost or damaged, the cost can be devastating to a business due to lawsuits, loss of business, or regulatory fines. Business espionage can result in considerable loss of money. When businesses attempt to gain confidential information about another company, it is usually for financial gain. These businesses can use the information to sell or trade a product for the purpose of introducing themselves into that part of the market. This will also prevent a rival company from being the “only guy on the block” with the product to offer, thus taking more of the market share. Corporate espionage is the art of circumventing the confidentiality of business data. Billions of dollars of losses are realized each year because sensitive data was not able to be kept confidential. With technology comes additional risk. More people are now emailing company data to home addresses or syncing hand-held devices (such as Blackberry) to their work and home computers. Commercial email servers and Wi-Fi capable devices are not typically as secure as a corporate intranet would be and this creates a risk for losing data. The US defines espionage towards itself as "The act of obtaining, delivering, transmitting, communicating, or receiving information about the national defense with an intent, or reason to believe, that the information may be used to the injury of the United States or to the advantage of any foreign nation. Espionage is a violation of United States law, 18 U.S.C. § 792–798 and Article 106 of the Uniform Code of Military Justice." The impact of loss or disclosure of the data can have many serious consequences. In terms of government data, the loss can mean that lives are lost. If sensitive data regarding a particular weapon system is disclosed then a potential enemy can either create a weapon similar to the ones the Americans use or they can find a flaw in the engineering and exploit the vulnerability. Once the enemy knows about the weapon system, they can also create a defense against that weapon, rendering it useless. Confidentiality of data is not just about what the company, or government is doing. There is a whole set of information regarding people. Personal information disclosure can have dire results. Some examples of these consequences are health information that can be used for discrimination, social security numbers can be used for identity theft, and biometrics information can be used to defeat other security systems. The concept of confidentiality of data seems to be the one piece of the Information Assurance that is the easiest to fail. With all the technological protection in place to determine the integrity of data in transit or the availability of the data at rest, it can all be circumvented if a single user with authorized access does not treat the data with due care. More data can be stored on smaller devices now. “Personal data on about 26.5 million U.S. military veterans was stolen from the residence of a Department of Veterans Affairs data analyst who improperly took the material home... It was in violation of our rules and regulations and policies.” People must follow the policies in order for these methods of protection to be effective. If users do not have faith in the confidentiality of their data, then they are less likely to use your services.

Integrity

CNSSI-4009: "Quality of an IS reflecting the logical correctness and reliability of the operating system; the logical completeness of the hardware and software implementing the protection mechanisms; and the consistency of the data structures and occurrence of the stored data. Note that, in a formal security mode, integrity is interpreted more narrowly to mean protection against unauthorized modification or destruction of information."

Some practitioners make the mistake of thinking of the integrity attribute as being only data integrity. While data integrity is a major part of this attribute, it is not everything. This attribute also addresses whether the physical and electronic systems have been maintained without breach or unauthorized change. It even refers to the people involved in handling the information; are they acting with proper motivation and integrity.

Integrity means data can *not* be created, changed, or deleted without proper authorization. It also means that data stored in one part of a database system is in agreement with other related data stored in another part of the database system (or another system).

For example: A loss of integrity occurs when an employee accidentally, or with malicious intent, deletes important data files. A loss of integrity can occur if a computer virus is released onto the computer. A loss of integrity can occur when an on-line shopper is able to change the price of the product they are purchasing.

The Merriam-Webster dictionary describes integrity as being incorruptible, and being in an unimpaired condition, and the quality or state of being complete. It uses words like “soundness” and “completeness” to help define integrity. So it is not too hard to imagine that when Data Integrity is discussed that a certain level of trust in the “truth” of your data is expected. Commanders and corporate leaders make decisions based on the available data. If the data has been altered or modified or is incomplete, then the decisions will not be based on the best possible information. The integrity of the data can be compromised by both intentional acts as well as accidental.

Unintended modification or loss of data is the most common cause of integrity loss. Accidental destruction of parts of data can also be broken down in several ways. Environmental factors such as floods or fires can destroy part of your data, thus losing the completeness of it. Human errors occur when data is incorrectly entered in to the system or data is not saved correctly. Hardware and software failures are often a large contributor to data loss. If a server gets too hot and crashes, a company may only be able to retrieve some data off the drives. Properly configured backup solutions mitigate the risk of integrity loss from hardware and software crashes.

Intentionally altering data is the most dangerous consideration for data integrity. Malicious code such as viruses and worms have the capability to alter data to meet the attacker's needs. Integrity can be assigned to the user as well. Most malicious code requires the end-user to perform an action in order to download the bad program. If the users of the system are "surfing" to web sites that are not authorized by company policy, or if they open and click on links in emails from people they do not know, (although most people have been trained in information security by now) then they are exposing the system to new vulnerabilities. Some of the vulnerabilities have the capability to capture network traffic in transit, analyze the packets, alter the data, then put the traffic back in the pipe and send it on its way. That altered data can be designed to perform any function the attacker desires, or the data can simply change a "yes" answer to a "no" causing the intended recipient to make decisions on faulty information. If users do not have faith in the integrity of their data, then they are less capable of making timely decisions.

Availability

CNSSI-4009: "Timely, reliable access to data and information services for authorized users."

Availability means that the information, the computing systems used to process the information, and the security controls used to protect the information are all available and functioning correctly when the information is needed. The opposite of availability is the lack thereof, one example of this is a common attack known as a denial of service (DoS) attack.

For example: In 2000 Amazon, CNN, eBay, and Yahoo! were victims of a DoS attack.

> “ *Yahoo Attacked. No one knows what happened except that it was inaccesible for more than 3 hours. It was also known that the attack was co-ordinated and hence the standard firewall algorithms failed to figure out what was happening.* ”
>
> — -Techhawking

Authentication

CNSSI-4009: "Security measure designed to establish the validity of a transmission, message, or originator, or a means of verifying an individual's authorization to receive specific categories of information."

Authentication breach can occur when a user's login id and password is used by un-authorized users to send un-authorized information.

Authenticity

Authenticity is necessary to ensure that the users or objects (like documents) are genuine (they have not been forged or fabricated).

As files are shared across multiple organizations, there can be circumstances when duplicate copies of that file may exist. In such cases it's important to establish not only which is the master copy, but also to establish a way for those who use the data to know where file, and all of the tagged data sets in the file, came from. A Tagged Data Authority Engine is one way to do this.

Non-repudiation

CNSSI-4009: "Assurance the sender of data is provided with proof of delivery and the recipient is provided with proof of the sender's identity, so neither can later deny having processed the data."

Non-repudiation implies that one party of a transaction can not deny having received a transaction nor can the other party deny having sent a transaction.

For example: Electronic commerce uses technology such as digital signatures to establish authenticity and non-repudiation.

Utility

Utility means usefulness and usability. For example, suppose someone encrypted data on disk to prevent unauthorized access or undetected modifications – and then lost the decryption key: that would be a breach of utility. The data would be confidential, controlled, integral, authentic, and available – they just wouldn’t be useful in that form. Similarly, conversion of salary data from one currency into an inappropriate currency would be a breach of utility, as would the storage of data in a format inappropriate for a specific computer architecture; e.g., EBCDIC instead of ASCII or 9-track magnetic tape instead of DVD-ROM. A tabular representation of data substituted for a graph could be described as a breach of utility if the substitution made it more difficult to interpret the data. Utility is often confused with availability because breaches such as those described in these examples may also require time to work around the change in data format or presentation. However, the concept of usefulness is distinct from that of availability.

Information assurance process

The IA process typically begins with the enumeration and classification of the information assets to be protected. Next, the IA practitioner will perform a risk assessment. This assessment considers both the probability and impact of the undesired events. The probability component may be subdivided into threats and vulnerabilities. The impact component is usually measured in terms of cost. The product of these values is the total risk.

Based on the risk assessment, the IA practitioner will develop a risk management plan. This plan proposes countermeasures that involve mitigating, eliminating, accepting, or transferring the risks, and considers prevention, detection, and response. A framework, such as Risk IT, CobiT, PCI DSS, ISO 17799 or ISO/IEC 27002, may be utilized in designing this plan. Countermeasures may include tools such as firewalls and anti-virus software, policies and procedures such as regular backups and configuration hardening, training such as security awareness education, or restructuring such as forming an **computer security incident response team** (CSIRT) or **computer emergency response team** (CERT). The cost and benefit of each countermeasure is carefully considered. Thus, the IA practitioner does not seek to eliminate all risks, were that possible, but to manage them in the most cost-effective way.

After the risk management plan is implemented, it is tested and evaluated, perhaps by means of formal audits. The IA process is cyclical; the risk assessment and risk management plan are continuously revised and improved based on data gleaned from evaluation.

Education and certifications

In the United States, the National Security Agency (NSA) has partnered with other organizations to designate a number of colleges and universities as Centers of Academic Excellence in Information Assurance Education, CAE/IAE and Research, CAE/IAE-R. These institutions offer a wide range of undergraduate and graduate-level degree programs, both masters level and doctoral, in IA-related studies and discipline. The current list of designated centers is maintained by NSA.

The **Master of Science in Information Assurance** (MSIA) and **Master of Science in Information Security and Assurance** (MSISA) degrees are multidisciplinary degree programs offered by many leading institutions which combine theory with applied learning in order to prepare security practitioners to work in the field of information security.

There is a current and future need for information assurance professionals to support the security needs of the world's information infrastructure. Information Assurance has become a critical issue for businesses in the current era as they wrestle with the problems of external and internal network attack, cyberterrorism, access control systems and regulatory compliance requirements.

In addition to traditional university degrees, the IA field boasts an extensive set of technical and professional certifications, used to indicate specific training or experience in detailed IA or security practices, at both the technical implementation and management level. An important aspect of these certifications is that, unlike university degrees, they are not lifetime credentials. Rather, each certification authority mandates recurring continuing education or re-testing in order to retain the credential. Further, the certification knowledge base is usually updated and renewed on a much faster schedule than is possible with university curricula. The IA and security certification marketplace is crowded and rapidly changing; here is a partial list of currently recognized certifications:

- Global Information Assurance Certification (GIAC) series administered by the SANS Institute.
- Certified Ethical Hacker (CEH) offered by the EC Council.
- Certified Information Security Auditor (CISA) offered by ISACA.
- Certified Information Security Manager (CISM) offered by ISACA.
- Systems Security Certified Practitioner (SSCP) administered by $(ISC)^2$.
- Certified Information Systems Security Professional (CISSP) administered by $(ISC)^2$.

Information Assurance practitioners supporting the US Department of Defense are required to hold selected security certifications in accordance with DoD Directive 8570.01-M.