Softwares and Software Security

Lesson 9

KEY CONCEPTS

■ Defects ■ Bugs ■ Failure ■ Flaws ■ Vulnerabilities ■ Software ■ System Software ■ Application Software ■ Utility Software ■ Software Security

Learning Objectives

To understand:

- The concept of software
- The Differentiate between different types of software
- And appreciate the concept and need of software security
- The software security mechanisms
- And learn how software security is perceived by the courts through various case laws
- Appreciate the legal and statutory compliances regarding software security.
- The get familiar with recent trends in software security
- The best practices of software security

Lesson Outline

INTRODUCTION

Security concerns in terms of software has become a pertinent issue in recent times and rightly so, because a weak software or complete lack of software can be a potential breeding ground for common security issues and malicious attacks.

In today's tech savvy world, software security cannot be ignored as we regularly interact with a multitude of software in our everyday life, right from home to office. Any kind of external intervention in software systems can seriously threaten the security of data and make it prone to unauthorized access. Therefore, it is important to take timely decisions for software security at all levels. An organization has to make decisions regarding investment in appropriate software security solutions to ensure company software systems are adequately protected. Software developers must design and develop security software mechanisms to safeguard against malware and other attacks. Individuals must regularly check and update their software security mechanisms to identify and guard against possible breaches.

In this chapter, we will understand about software, potential threats to software security and solutions to software security.

SOFTWARE-OVERVIEW

A computer system is composed of hardware and software components. Hardware is the external physical components of the computer which can be touched by users. Examples of hardware include desktop, printer, mouse, etc. On the other hand, software refers to set of instructions which enable the hardware components to perform[1]. Examples include Windows, Linux, MS Word, PowerPoint, MS Excel, etc. Software, thus, can be referred to as a "set of instructions" which direct the hardware to do a particular task and specifies the manner in which the said task must be accomplished[2].

The concept of software rests on the basic principle of:

The input fed into the computer through the hardware device is programmed by the software to produce the desired result or output.

The difference between software and hardware has been discussed in detail below:

Hardware	***Software***
It is the physical parts of the computer which process the data1.	It is the collection of instructions which directs the computer as to 'which' task to perform and 'how' to perform it.
Hardware is unable to perform any task without the support of software.	Software can only be run and executed in computer hardware.
It understands only machine-level language.	It accepts inputs by users, converts it into machine-level language which is then sent to Hardware for processing.
It can be seen, felt and touched.	It can only be seen and cannot be touched.
Examples: Monitor, Hard disk, CPU, Keyboard, Printer, Mouse, etc.	Examples: Windows, Linux, MS PowerPoint, MS Excel, etc.

1. Difference between Hardware and Software, BYJU'S, Available at: https://byjus.com/free-ias-prep/difference-between-hardware-and-software/#:~:text=A%20computer%20system%20is%20divided,a%20specific%20set%20of%20tasks.

2. Study material on Computing Basics, Available at: https://ftms.edu.my/v2/wp-content/uploads/2019/02/csca0101_ch07.pdf.

CHARACTERISTICS OF GOOD SOFTWARE

The desirable characteristics of a good software have been outlined below:

a) ***Security:*** A good software must have adequate security measures in place to safeguard against attacks.

b) ***Functionality:*** A software must be able to perform its intended function and fulfil the purpose for which it was designed[3].

c) ***Reliability:*** A software must be able to operate, without failure, over a specified period of time, under certain conditions.

d) ***Versatility:*** It refers to quality of the software of being able to perform as per the requirements of the user in varied conditions and environments.

e) ***Stability:*** A software must be stable enough to withstand changes made in its design and

f) ***Modifiability:*** A software must be capable of accepting modifications in its structure and design, necessitated by needs which may arise in due course.

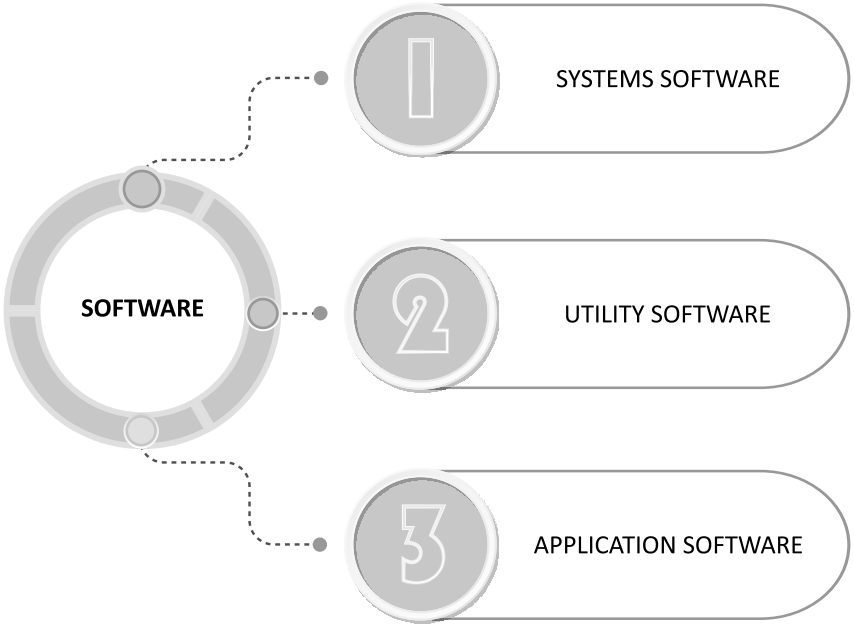
g) ***User-Friendliness:*** A software must be easy to understand and use.

h) ***Portability:*** It refers to the ease with which software can be ported from one platform to another without (or with minimal) changes, while obtaining similar results.

i) ***Maintainability:*** Software must be able to withstand changes made its code, so that the software can be altered or modified or new changes/features can be introduced as per requirements.

SOFTWARE CLASSIFICATION

Software can be classified into the following types:

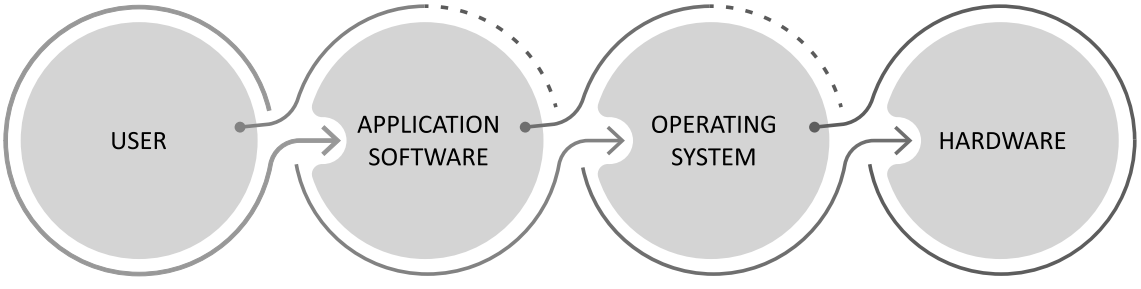


3. "Software Engineering", GEEKS FOR GEEKS, Available at: https://www.geeksforgeeks.org/software-engineering-characteristics-of-good-software/.

Systems Software: These software enables interaction of the user with hardware components of the computer and helps in running programs on the computer. System software serves as a point of interaction between the hardware components of computer and user applications. It is relevant to note that computer hardware only understands machine language (that runs into binary digits of 0s and 1s) whereas application function in human-readable language (for example: English). Therefore, a system software converts machine language to human readable language and vice versa to facilitate smooth functioning of the computer.

Types of system software

1. **Operating System:** It refers to the amalgamation of hardware configuration of a computer with a certain software package[4]. For example, Windows, DOS, MAC, Linux. Operating system is basic and essential to the functioning of a computer as all computer applications sit inside the operating system of a computer. The basic functions of storing data, retrieving files and scheduling of tasks are also managed by the operating system[5]. The software which is loaded just when the computer is turned on is the Operating System and this process is known as 'booting'.



Functions of an Operating System:

i. ***Performing Hardware Functions***: the operating system performs the important task of converting the simple instructions given by the user through application program into a set of elaborate and detailed instructions which are required by the hardware devices. It also informs the user about any error or malfunction which has occurred or if any of the input/output devices need attention[6].

ii. ***Interface***: The operating system provides an interface for the user to insert instructions and command the computer. Graphical User Interface, as the name suggests, provides graphics (in the forms of icons and menus on the screen) which are used by users to command the computer[7]. For example, Windows by Microsoft.

iii. ***File Management*:** Files are managed by the Operating System so that it can be made available to the CPU as and when required[8]. Operating System also ensures that the files are protected and kept secure from unauthorized access.

iv. ***Management of Processing Tasks*:** Operating System enables the user to run multiple programs at the same time or allow multi users to use the computer at the same time. This is made possible through the task processing feature of operating systems.

v. ***Memory Management*:** Operating systems manage the memory and storage of the computer so that data can be stored, retrieved and supplied to the application program so that instructions can be effectively executed.

4. Software Engineering, Available at: https://www.skylineuniversity.ac.ae/pdf/software-engineering/Software.pdf.

5. Basics of Computer Software, Available at: https://www.tutorialspoint.com/basics_of_computers/basics_of_computers_software_concepts.htm.

6. Study Material on Computer Software, Available at: https://egyankosh.ac.in/bitstream/123456789/7375/1/Unit-3.pdf.

7. Ibid.

8. Study Material on Computer Software, Available at: https://egyankosh.ac.in/bitstream/123456789/7375/1/Unit-3.pdf.

2. **Language Processor:** It performs the task of conversion of human readable language into machine language and vice versa[9]. The interactions between users and computer takes place in three languages namely:

   - **Machine Language:** This language consists of binary digits of O's and 1's, which is only understood by a machine. It can be called as a 'machine-friendly' language.
   - **Assembly Language:** This is the Low-Level Language (LLL) which uses "Mnemonics" or words and symbols in English language to represent long strings of binary digits (0's and 1's)[10]. For example: the mnemonic "READ" means that data has to be retrieved from the memory[11]. This language is dependent upon machines and varies according to the processor used[12].
   - **High Level Language:** This language consists of statements in English, which is readable and understandable to humans[13]. It can be called 'programmer friendly' language. For example: JAVA, C++, etc.

   Since the machine level language cannot be comprehended by users, HLL is commonly used for coding. The codes so developed (i.e. source codes) have to again be converted into machine language (i.e. machine/ object code) so that it can be understood by the computer to produce the desired output. This conversion is primarily the function of Language Processor.

   A language processor is typically made of three essential components:

   - ***Assembler:*** This component of language processor converts assembly language into high level language.
   - ***Compiler:*** This component of language processor converts high level language into machine language in one go.
   - ***Interpreter:*** This component of language processor is employed in line by line conversion of high-level language into machine language. The execution time of this processor is slow.

   The difference between Compiler and Interpreter has been outlined below:

COMPILER	***INTERPRETER***
It is concerned with simultaneous conversion of High Level Language into Machine Language, all at once.	It is concerned with line by line conversion of High Level Language into Machine Language.
Errors are reported after compilation.	Errors are reported line by line, as and when they occur.
Faster in comparison to an interpreter.	It is comparatively slower than a compiler.
It first scans the entire source code which is then converted to object/machine code, in one go.	It scans the source code line by line and converts one statement at a time.

9. Software and its Types, GEEKS FOR GEEKS, Available at: https://www.geeksforgeeks.org/software-and-its-types/.

10. Supra, Note 8.

11. Ibid.

12. Study Material on Types of Software, Available at: https://stlawrencehighschool.edu.in/uploads/onlineclass/364aecdbfc1a21ddc39606d5692ec3cf.pdf.

13. Ibid.

3. ***Device Driver:*** This software acts an interface between the user and the input-output devices of a computer[14].
4. ***BIOS:*** Basic Input Output System is responsible for controlling the input output devices of a computer. BIOS also initiates the booting process of a computer.

Application Software

These are software which are dedicated to the performance of a particular task or function.

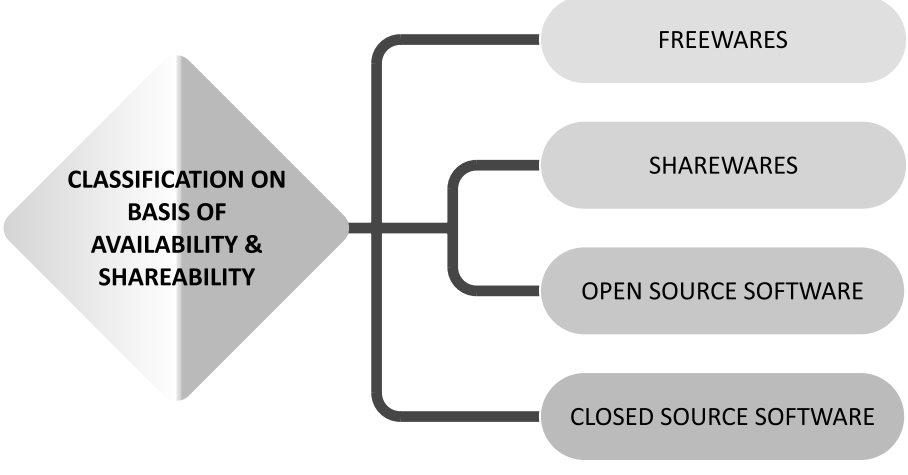
Application software are of the following two types:

- ***General Purpose Application Software:*** These are "ready to use" software, made for common use. For example, VLC Media Player is used to play audios and videos, MS PowerPoint is used to create presentations, etc.
- ***Specific Purpose Application Software:*** These types of software are customised for businesses or organisation. For example, a library may have a software that keeps a track of all the books, ticket reservation software, software used for management and allocation of rooms in a hotel, etc.

Utility Software

A utility software is a type of application which assists the system software in performing its work[15]. This software performs basic tasks, which are required daily. A computer may function without a utility software; however, it is found in most computers as it enhances the performance and output of a computer system. Examples of such software can be an anti-virus software which guards the computer against virus and malware attacks, Text Editors like Notepad which allows users to create text documents, Disk Defragmenter Tools which rearranges the files in proper order in a disk, etc.[16].

OTHER TYPES OF SOFTWARE ON THE BASIS OF AVAILABILITY AND SHAREABILITY



- **Freeware:** As the name suggests, freeware software are those which can be downloaded from the internet and used by the users free of cost. These are standardized software which cannot be tailored to suit individual needs. For example, Google Chrome, Mozilla Firefox, WPS, etc.

---

14. Study Material on Types of Software, Available at: https://stlawrencehighschool.edu.in/uploads/onlineclass/364aecdbfc1a21ddc39606d5692ec3cf.pdf.

15. Basics of Computer Software, TUTORIALSPOINT, Available at: https://www.tutorialspoint.com/basics_of_computers/basics_of_computers_software_concepts.htm#.

16. Supra, Note 14.

- **Sharewares:** These are made available to the users with a free trial for a limited time period during which the users can use the software and decide whether they want to purchase the software or not. Here, users are asked to pay in case they wish to use the software beyond the free trial period.
- **Open Source Software:** The peculiar feature of this software is that the source course is made available to the users along with the software and this source code can be modified by the users to suit their individual needs. This software is available for free or for charge on the internet. For example, Apache Web Server. Open Source Software offer varied benefits to the user like cost savings, customization and security.
- **Closed Source Software:** This type of software can be purchased by the users. However, the source code of such software is protected by intellectual property laws. Most of the commonly used software belong to this category. For example, Mac OS, Win RAR, etc.

SOFTWARE SECURITY

Software Security refers to the practice of developing and engineering the software in a manner which keeps it secure from external malicious attacks, while also ensuring that in case of any such attack, the software does not malfunction and continues to operate. It is essentially a preventive technique to manage and avoid security risks.

Since computers are heavily reliant upon software for their smooth functioning, it becomes important to secure software and protect them from undesirable viruses and attacks. Software attacks are increasingly common in mobile applications. Software developers need to be extra cautious while designing and developing a software as any loophole in the software can be exploited by hackers. The Developer should be aware about any possible vulnerability which may arise in respect of a particular software so that the source code can be written in a manner which guards against such a vulnerability.

Software Security: Overview and Significance

Systems are usually complex in nature and systems run by software are not only complex but also involve multiple risks. One such risk is that of protecting such complex systems as it is difficult to understand it, analyse and evaluate it to decide appropriate protection strategy. The complexity results in a situation where security risks are often identified at stages when most damage has already been done. Even simple systems might pose a challenge to secure when there are inherent flaws in the software design.

Hidden risks can affect any computing system. The software that is initially loaded on the device can be altered by malicious programmers[17].

Users may install a programme wrongly, introducing uninvited risk, or, worse, transmit a virus by installing new programmes or software updates. In a multiuser system, a malicious user may install a Trojan horse to steal the passwords of other users[18].

The significance of software security lies in the increased inter-connectivity of computers, mobiles and other electronic devices through the internet, which enables dissemination of information on the click of a button. Furthermore, people, organisations, and governments are becoming increasingly reliant on network-enabled communication tools like e-mail and websites offered by information systems. Unfortunately, because these systems are connected to the Internet, they are vulnerable to remote attacks. This interconnectivity not only increases possible areas of attacks but also enables attackers located in one corner of the world to target systems located in another part of the world. Resultantly, this means that an attacker does not need physical access to a system to trigger security issues.

17. Introduction to Software Security, Available at: https://img2.helpnetsecurity.com/dl/reviews/viegach1.pdf.

18. Supra, Note 20.

Software security also depends upon the degree of extensibility of systems. Systems are made "extensible" through updates/extensions which are made to improve the capabilities and functionality of the software, without re-writing the entire code. These updates must be designed with security of the software in mind.

These trends of extensive networking, increasing system complexity, and built-in extensibility, when combined, make software security more critical than ever[19].

SOFTWARE SECURITY: A PROACTIVE SECURITY

In the past, security issues in terms of systems, software, internet, etc. were only paid attention to when any breach or untoward attack came to light. However, one malicious attack has the potential to seriously disrupt any organization by leaking its data, hacking into its systems, accessing customer information, new product designs, etc., leading to loss of revenue, reputation and business of the organization. These kind of security solutions (which were conventionally used in the past) where actions are taken to remedy the breach only after the breach has occurred or damage has taken place, are called reactive security practices or solutions. The major task of reactive security is to identify the attacker, hacker or intruder, after discovery of breach.

Even though reactive security is not the suggested mode of security solution, it still has certain benefits which have been enumerated below:

- It scans and monitors any anomaly (for example, unusual traffic, authentication failures, etc.) through intrusion detection systems[20].
- It provides prompt incident response where every data breach incident is inquired and investigated to find out the root problem, which is then cured.
- Anti-malware and anti-spam applications are examples of reactive security. On detection of any spam or malware on computer, these applications immediately take steps to remove it.

Software security, on the other hand, is a type of proactive security. It is based on the principle "prevention is better than cure". Proactive security aims to prevent any data breach incident before any malware is able to access network server or before vulnerabilities in a software are exploited by hackers[21]. Proactive security typically necessitates additional software and hardware designed for spotting threats before they turn into serious incidents[22]. Giving administrators information on vulnerabilities so they may take the required steps to promptly fix them is another part of proactive security[23].

Some benefits of proactive security have been outlined below:

- It helps an organisation to save money and maintain its brand value by preventing attacks before they occur.
- It prevents breach of sensitive data and malicious attacks.
- It enables early identification of vulnerabilities before they are discovered by potential attackers.
- It enables organisations to stay complaint by constantly monitoring the data.
- It prevents potential crises and reduces the stress of facing any malicious attack so that management and employees can focus on growth and expansion.

19. Ibid.

20. Proactive v. Reactive Security, IMUNIFY SECURITY, Available at: https://blog.imunify360.com/proactive-vs.-reactive-security-5-tips-for-proactive-cyber-security.

21. Ibid.

22. Ibid.

23. Ibid.

Software Security Goals

Software security seeks to achieve the following goals:

- **Prevention of malicious attacks:** The main goal of software security is to prevent attacks in this age of Internet, where data breach spreads like wildfire.
- **Timely threat detection:** It aims to detect and prevent possible breach of security in time.
- **Effective Monitoring:** Real time monitoring is an effective tool to avoid attacks before they happen and prevent damage.
- **Privacy and Confidentiality**: Developers must address this issue effectively as software run on machine which has the ability to track or access information stored on the software. Therefore, a machine may be able to access information which the software may attempt to hide.
- **Anonymity:** This is an important concern which requires the developers to consider what may happen to any data which is collected by a program and whether the same has been adequately protected.
- **Authentication:** Most software require user to log into the system through Login ID and Password. It is essential for security as it is relevant to understand who can be trusted and who cannot be trusted[24].
- **Integrity:** Software should enable maintenance of integrity of data, i.e. the ability to remain the same, to ensure that data is not manipulated during transmission from sender to receiver.

SOFTWARE SECURITY: BEST PRACTICES

1. **Verification of Software:** Software Verification is the process of verifying whether a code corresponds to particular specifications. The security limitations are encoded and given as configuration to the verification process[25]. The process of verification ensures that a particular software conforms to the specifications, which in turn guarantees that security violations will not take place[26]. For example, CompCert is a formally verified compiler which ensures that conversions are always correct and no bug is introduced during compilation[27].
2. **Language Based Security:** A new area of research has emerged in the form of language-based security where security properties are implemented in the programming language[28]. This practice encourages development of programming languages in security specific manner and prevents the programmers from making mistakes[29]. For example, type safety and memory safety are enforced and implemented as part of programming language by Java (a widely used programming language).
3. **Software Testing:** It is the practice of finding flaws or possible vulnerabilities in software while executing a program[30]. However, it is important to note that security requirements cannot be tested easily as software testing only reveals and detects if the program is infused with bugs, but it cannot confirm absence of bugs[31].
4. **Training:** All users, be it employees, students, businessmen, should be trained to use the software in a proper manner and must be cautioned and informed about security risks and how to identify, avoid

24. Study Material, Available at: https://img2.helpnetsecurity.com/dl/reviews/viegach1.pdf.

25. Mathias Payer, "Software Security: Principles, Policies and Protection", July 2021, Available At: https://nebelwelt.net/SS3P/softsec.pdf.

26. Ibid.

27. Supra, Note 28.

28. Mathias Payer, "Software Security: Principles, Policies and Protection", July 2021, Available At: https://nebelwelt.net/SS3P/softsec.pdf.

29. Ibid.

30. Ibid.

31. Ibid.

and prevent them. Users must also be trained about remedial measures that must be taken as soon as a breach is detected.

5. **Data Encryption:** Data encryption converts data into an unreadable format which can only be deciphered with the help of a security key. It is a proven security technique which is widely used to protect against data breach.
6. **Patching Software:** It is always recommended to resolve and fix system vulnerabilities as and when they are detected. This is known as 'patching the software'.
7. **Use of Firewall:** Firewall act as a protective layer and barrier between the internal computer network and the internet. It allows or prohibits traffic from the internet as per the pre-established security rules and policies. Use of firewalls is thus recommended as a good security practice.
8. **Two factor authentication:** It is a security practice which allows access to a software or program only on providing credentials like login ID and password. Only a user who knows the login ID and password would be able to access the account.
9. **Penetration Testing:** This security practice involves hiring a testing team which attempts to penetrate into the system by using the same tools and in the same manner that a hacker would do, to identify and flag security issues in the software. It is recommended that penetration testing should be done periodically. Benefit of this practice is that existing issues or vulnerabilities in the software are quickly identified and resolved before they are discovered by hackers.
10. **Remain agile and proactive:** It is recommended that one must always be aware of current and emerging security trends and new security issues and constantly evolve and adapt to new, updated and emerging security practices, only after checking its viability.

Therefore, it can be concluded that software security is a continuous process which starts right at the design and developing stage and continues throughout as software is regularly patched and updated to resolve security issues.

WHAT TO AVOID IN SOFTWARE SECURITY?

Never misplace trust.

- Avoid storing sensitive client information except where it is necessary. If sensitive information has been stored, it must be ensured that it has been adequately protected at all levels, and cannot be compromised.
- Incoming data from untrusted sources should always be scrutinized and tested before allowing access to the system.
- Avoid sharing of passwords and resources like MAC address, etc.

Allow authorisation after authentication.

- Users should be allowed to access the system and perform tasks only after their identity has been verified and duly authenticated.
- Process of authentication may involve confirming identity of user through use of password or establishing identity through finger print, facial recognition sensors.
- Authorization may be granted to access the system, and depends on nature of request, time and location of request, etc.

- Sensitive transactions or high value transactions may be conducted only after double verification or by using higher level of authentication techniques.

Never mingle data and control functions.

- Lack of proper bifurcation and separation of data and control instructions can result in memory specific vulnerabilities.

Keep in mind the Users.

- Software must be developed keeping in mind the ultimate users and how they will operate the software. Keeping the background, biases, preferences of users in mind, security features must be built-in within the software.
- Moreover, security configurations must be easy to use and set up, so that users can make use of the same.
- Do not consider security as a feature, rather, built a foundation of the software on basis of security.

CASE STUDY AND CASE LAWS OF INDIAN LAW ON SOFTWARE SECURITY

CASE STUDY: MOBILE SECURITY

Smart phones, tablets, smart-watches, etc have become an indispensable part of our everyday lives. These devices run on mobile operating systems, which serves as a user interface and allows the user to download and run applications from the internet. Android is one such example of a mobile operating system which is widely used all over the world. One major issue for android is to accommodate devices made with different configurations, by different companies on one standard operating system framework.

Android is based on modified version of Linux kernel[32] Under android based system, applications are kept separate from each other and each application (be it downloaded from the internet or pre-installed) has its own individual user account, having its own login credentials. This means that multiple applications do not use the same user id. The interaction between the apps is minimised through an Application Programming Interface[33].

The applications used on smart phones and tablets can be downloaded from a central marketplace of apps like Google Play Store. Developers upload their apps on such marketplaces. Every developer has to pay a certain amount of entry fees to upload their apps to the marketplace. Moreover, automatic updates are instantly provided to existing users, who have already installed the application on their mobile phones and tablets.

Once downloaded, there are certain in-built security restrictions on the app. In order to function properly, every application would require some sort of permission access. For example, through Instagram, users upload their pictures and stories on social media. Therefore, in order for Instagram to properly function, it would require access to the phone's photos and photo gallery. Once this permission access is granted, applications perform their designated tasks. It must be noted that how much access and what areas of access to mobile device is provided to the application is in the hands of the user; who may decide to give or withhold such permission. In this manner, the ultimate security decisions lie in the hands of the user. However, to what extent the users are able to take informed decisions, depends upon the level of awareness and cautiousness exercised by them.

32. Android (Operating System), WIKIPEDIA, Available at: https://en.wikipedia.org/wiki/Android_(operating_system).

33. Mathias Payer, "Software Security: Principles, Policies and Protection", July 2021, Available At: https://nebelwelt.net/SS3P/softsec.pdf.

CASE STUDY: PUNE CITIBANK MPHASIS CALL CENTRE FRAUD (2005)

These were instances of source engineering in the United States of America. In 2005, $350,000 was illegally moved from four Citibank clients' bank accounts through the internet. By reassuring certain clients that they would find it simple to help them in difficult circumstances while they were far from the bank's branch, the bank personnel were able to gain their trust. The money was allegedly moved from the client's account with the bank to a phoney bank account in Pune, according to witnesses. There was no decoding of the encrypted software or breaking of firewalls; only the Mphasis system's weaknesses were found. The defendants in this case were former workers of the Mphasis call centre. Every time an employee entered or exited, they were scrutinized, and it was discovered that the employees were unable to write down the numbers and had to commit it to memory as this was the only option. Subsequently, money was transferred from a cybercafé.

The defendant was found guilty of utilizing unauthorized access to clients for criminal purposes, according to the court's observation. The defendant was found guilty of the crime by the court, and they were punished in accordance with sections 43(a) and 66 of the Information Technology Act as well as sections 420, 465.467, and 471 of the Indian Penal Code.

CASE LAWS ON SOFTWARE SECURITY

Syed Asifuddin and Ors. v. State of Andhra Pradesh and Anr. [Andhra High Court, reported as 2005 CriLJ 4314]

The present case concerned quashing of FIR registered against the Petitioners (who were staff members of TATA Indicom) under Sections 409, 420, 120B of the Indian Penal Code, 1860, Section 65 of the Information Technology Act, 2000 and Section 63 of the Copyright Act, 1957. In this case, a complaint was filed by Sales and Marketing Department of M/s Reliance Infocom Ltd. Hyderabad, who were offering Reliance India Mobile (RIM) services to telecom subscribers. Under the RIM scheme, a subscriber would get a mobile handset worth Rs. 10, 500/- along with service bundle for a period of three years. However, the handset offered was locked technologically, which meant that only Reliance services could be availed on the handset. One major drawback was that if a customer wanted to opt out of the service, it had to pay the original price of the handset along with others service charges.

The subscribers, after availing the said scheme, were not happy and were lured by better tariff plans offered by other telecom service providers, which also offered an option to switch the service provider. This scheme had a great influence on the market as the handset offered was a 3rd generation digital device and therefore, this scheme quickly garnered good market share. Resultantly, the other service providers in the market resorted to illegal tactics to divert and manipulate the customers. Employees of TATA Indicom started making calls to Reliance's customers, made a deal, where Reliance headsets were reprogrammed (by hacking the ESN i.e. Equipment Special Number) so that other service provider services could be used on it.

The Andhra High Court held that the definition of computer under S. 2(1)(i) of the Information Technology Act, 2000 includes within its ambit a mobile handset. The High Court while refusing to quash the offence under under Section 65 of the Information Technology Act, 2000, held that reprograming the handset by alteration of ESN by service providers like TATA Indicom, is an offence under the said section.

***Shreya Singhal* v. *Union of India* (AIR 2015 SC 1523)**

This case touched upon the subject of free expression on the internet and censorship and is often regarded as a milestone decision in the jurisprudence of information technology. This matter concerned two girls who were detained by police for posting remarks on Facebook criticizing the closure of Mumbai city following the death of a political leader. The matter was taken before the Supreme Court of India in 2013.

Any online communication that may be "offensive" or "menacing" was considered illegal under S. 66A of the Information Technology Act, 2000. The Supreme Court declared S.66A to be unconstitutional as it was ambiguous, overbroad, and unclear, thus infringing the fundamental right of free speech and expression guaranteed under Art. 19(1)(a) of the Constitution.

The case has received widespread praise as a victory for those who support free expression and has established a significant precedent for laws on free speech on the internet in India.

Avnish Bajaj v. State (NCT) of Delhi, (2008) 150 DLT 769

In the abovementioned case, the question of intermediaries' responsibility for user-generated content on their platforms was addressed. The conflict arose when the CEO of Baazee.com (now eBay India), an online marketplace, was detained for posting a listing of an offensive video.

According to the Information Technology Act of 2000, the court determined that Baazee.com was an intermediary and was not responsible for any information supplied by a user, provided that the intermediary was unaware of the content's illegality.

The court emphasized that while intermediaries are not obligated to actively track every piece of information posted on their platforms, they have a duty to promptly remove any illegal content after receiving notice from the authorities. The CEO was released from jail when the court ruled that his arrest was illegal. This case has clarified the issue of intermediaries' responsibility in India and established a significant precedent.

State of Tamil Nadu v. Suhas Kutti, CC No. 4680 of 2004

In this case, the question of cyberstalking and the application of the Indian Penal Code, 1860 (hereinafter referred to as 'IPC') to issue of online harassment were addressed.

The case included a man who had been texting, calling, and emailing a woman to harass her. The man was found to have engaged in cyberstalking, and the court ruled that these situations were within the provisions of IPC for criminal intimidation and stalking.

The decision made clear that harassment on the internet is a severe offence that may cause the victim great mental anguish and suffering, and that the law has to change to accommodate this new type of crime. The court also emphasized that the anonymity and seclusion offered by online communication might give offenders the confidence to act in ways they ordinarily wouldn't in face-to-face meetings.

The case has established a crucial precedent in India for the application of IPC provisions to cases of online harassment and has brought attention to the necessity for legislation and regulations to deal with the particular problems created by cybercrime.

MC Pneumatics (India) Pvt. Ltd. vs. Jogesh Kwatra, CM APPL. No. 33474 of 2016

This case addressed the subject of online libel and the responsibility of intermediaries for offensive material provided by their users.

The case concerned a complaint made by MC Pneumatics against Jogesh Kwatra, a former employee who was accused of posting false information about the business and its directors on social media channels. After concluding that the posts were defamatory and harmful to the company's reputation, the Court ordered Kwatra to delete the posts and stop publishing any further damaging claims.

The court further determined that the social media sites where the content was posted were intermediaries under the Information Technology Act of 2000 and weren't responsible for the material their users submitted, provided that they met the legal criteria for due diligence. The court emphasized that after obtaining notification from the person who is impacted or a court order, intermediaries must take immediate action to remove unlawful information.

The case has established a significant precedent for intermediaries' responsibility in cyberdefamation cases in India and has brought attention to the necessity for a balance between the protection of free expression and the protection of reputation and privacy.

CBI v. Arif Azim, (2008) 150 DLT 769

This case, also known as the Sony Sambandh case, is a well-known case in India, where the accused individual had stolen Barbara Campa's credit card information. He subsequently made purchases using the stolen card on a Sony India Private Limited website (sony-sambandh.com). Barbara protested the purchase, and the credit card company notified Sony of this. Sony, therefore, informed the CBI of the Internet fraud and cheating case.

In response, the CBI opened a probe under IPC sections 418, 419, and 420. The accused person was found guilty of the crime of cyber-fraud by the Delhi HighCourt under the aforementioned provisions of IPC. This case is specifically related to Section 66C of the IT Act, 2000, which addresses identity theft and the illegal and dishonest use of electronic passwords, signatures, and other personal identifying elements.

ANALYSIS OF INDIAN LAW ON SOFTWARE SECURITY

Various laws regulate cybercrime in India depending on the type of crime committed by the offenders, however the Indian Evidence Act of 1872, the Indian Penal Code of 1860 along with the Information Technology Act of 2000 are the most significant ones. The Indian Penal Code, 1860 (Hereinafter referred to as "IPC") and Information Technology Act, 2000 (Hereinafter referred to as "IT Act" both outline penalties and punishments related to numerous cybercrimes, and several provisions in both statutes have connections to one another. The IPC can be used to adequately prosecute cybercrimes that are not regulated under IT Act. However, it is important to note that both the IT Act and IPC penalize offences committed in the cyber space and internet.

I. The Information Technology Act (IT Act 2000)

In order to protect the leadership, finance, and retail sectors that are managed by people or organisations using the internet, the IT Act, 2000 was passed in India by the Parliament. The IT Act's purview has been expanded to include regulating internet-connected communication tools and offering remedies when a person's rights have been infringed. A number of cybercrimes are covered by the IT Act, and the act exhaustively makes an attempt to cover all cybercrimes. To draw attention to the crime and offer justice for the transgression committed, the IPC may have the last say in the situation. The IT Act has the following significant provisions that provide crimes and penalties for various types of cyber fraud.

1. **Section 43 of IT Act** – This section applies to anybody who intentionally damages another person's computer equipment without first letting the owner permit for the same. In such circumstances, the owner of the computer or electronic device is entitled to full recompense for the whole harm. This section empowers the victims with the ability to seek compensation for the violation of the rights guaranteed by the Act.
2. **Section 66 of IT Act** – This section is appropriate for injuries suffered by individuals as a result of dishonesty or deception on the part of the accused while committing any of the acts as described in Section 43. The offender will receive a punishment that may last up to three years in prison or a fine up to Rs. 5 lakhs.
3. **Section 66B of IT Act** – This section includes the penalties for obtaining a computer or other illegal communication equipment unlawfully. Depending on the seriousness of the violation, a three-year sentence can be coupled with one lakh rupees as fine.
4. **Section 66C of IT Act -** This section addresses crimes including impersonation in digital signatures, password hacking, and distinguishing identifying characteristics. If the accused is proven guilty, he/she might face up to 3 years in jail and an additional penalty of Rs. 1 lakh in addition to their sentence.

5. **Section 66 D of IT Act** – It was put into effect with an emphasis on punishing offenders who use computer resources to impersonate others. The penalty for the offence is a period of imprisonment that may last up to 3 years, as well as a fine that may amount to one lakh rupees.

II. The Indian Penal Code (IPC 1980)

The Information Technology Act of 2000 does not offer or include certain offences, which are covered under the IPC. Major crimes that can have a significant negative impact on society as a whole such as theft of identity and other related cyber scam offences can be penalized through IPC, in addition to the remedies provided by the Information Technology Act of 2000.

The key provisions of the IPC, 1860 that deal with crimes linked to cybercrimes and the associated penalties are as follows:

1. **Forgery (Section 464)**

   Forgery stipulates that an individual who commits the crime of forgery, i.e. creates a false document or a false electronic record with the intention to cause damage or injury to the public or any individual in general, will be penalized by imprisonment for a time that may not exceed two years, a fine, or a combination of the two.

2. **Forgery for purpose of Cheating (Section 468)**

   This section provides that an individual who forges a document or electronic record with the intention of using it to cheat others, is liable for this offence. If the suspect is proven to be at fault, he will be sentenced to a period of jail that may last up to seven years, along with payment of fine.

3. **Presenting a forged document as genuine (Section 471)**

   If any person, who knows that a document/electronic record is forged, uses it as genuine, he shall be punished in the same manner as if he had himself forged such document or electronic record.

4. **Forgery for harming reputation (Section 469)**

   This offence involves using a forged document or electronic record intending to use to harm person's reputation. The penalty prescribed will be 3 years in jail and a fine which will be imposed as punishment for the offender.

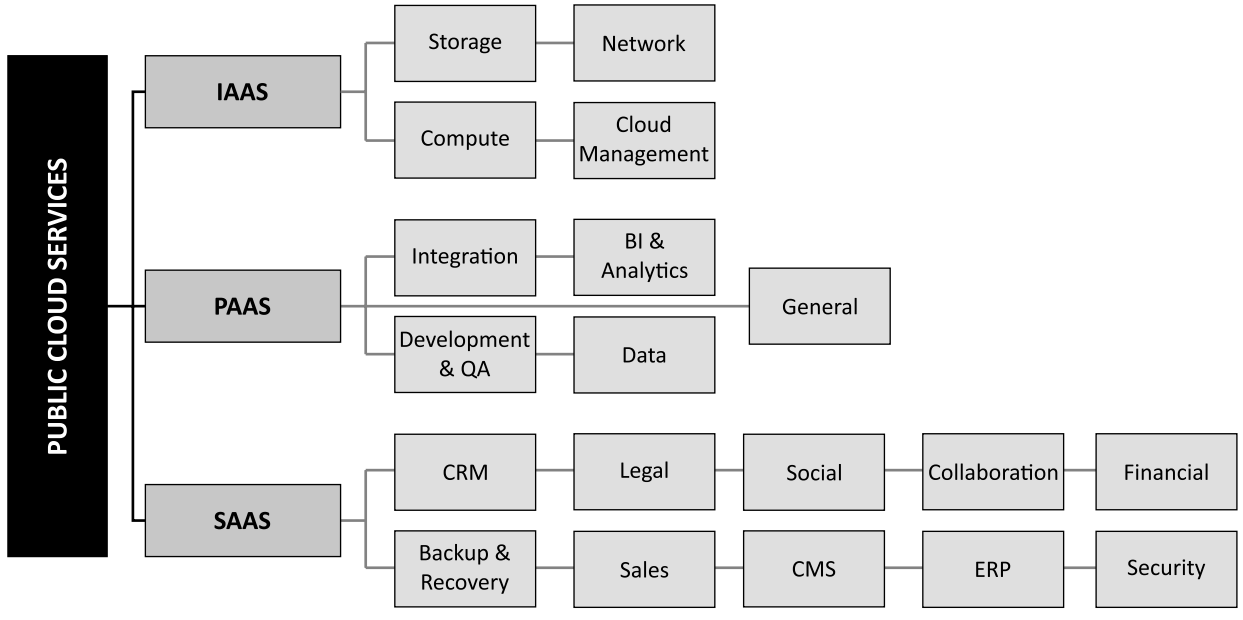
5. **Punishment for extortion (Section 384)**

   This offence involves putting the victim in a state of fear or inkjury so as to induce him to deliver to the offender any property, valuable security or anything that can be converted into a valuable security. The crime is punishable by a sentence of prison that may last up to 3 years, a fine, or both of them.

6. **Cheating and dishonestly inducing delivery of property (Section 420)**

   The perpetrator will get a penalty as well as a jail sentence that may last up to 7 years.

SAAS; PAAS, IAAS AND ON-PREMISE SOFTWARE: OVERVIEW AND RECENT TRENDS



IAAS Services

Infrastructure-as-a-Service, abbreviated as "IaaS," is a type of cloud computing where basic computing, networking, and storage resources are made available to users on demand through the internet and on a pay-as-you-go basis.

IaaS enables end users to increase and reduce resources as needed, lowering the need for expensive, upfront capital expenditures or needless "owned" infrastructure. IaaS offers the most fundamental level of control over cloud resources in contrast to PaaS and SaaS (even more modern computing paradigms like containers and serverless).

Early in the 2010s, IaaS began to gain popularity as a computing paradigm, and ever since then, it has evolved into the de facto abstraction model for many different kinds of workloads. IaaS remains fundamental but is more competitive than ever due to the emergence of new technologies like containers and serverless, as well as the introduction of the microservices application paradigm.

IAAS Platform and Architecture

IaaS is made up of a collection of physical and virtualized resources that provide consumers with the basic building blocks needed to run applications and workloads in the cloud.

- **Physical data centers:** IaaS providers manages large data centers that contain the physical machines required to power the various layers of abstraction on top of them and that are made available to end users over the web.
- **Compute:** IaaS is commonly understood as virtualized compute resources. Providers manage the hypervisors and end users can then programmatically provision virtual "instances" with desired amounts of compute and memory (and sometimes storage). Most providers offer both CPUs and GPUs for different types of workloads. Cloud compute also typically comes paired with supporting services like auto scaling and load balancing that provide the scale and performance characteristics that make cloud desirable in the first place.
- **Network:** Networking in the cloud is a form of Software Defined Networking in which traditional networking hardware, such as routers and switches, are made available programmatically. More

advanced networking use cases involve the construction of multi-zone regions and virtual private clouds.

- **Storage:** The three primary types of cloud storage are block storage, file storage and object storage. Block and file storage are common in traditional data centers but can often struggle with scale, performance and distributed characteristics of cloud. Thus, of the three, object storage has thus become the most common mode of storage in the cloud given that it is highly distributed, it leverages commodity hardware, data can be accessed easily over HTTP, and scale is not only essentially limitless but performance scales linearly as the cluster grows.

Pricing

IaaS prices are based on usage, so customers only pay for what they really use. There are now many different granularity levels included in the cloud infrastructure pricing models:

- **Subscriptions and reserved instances:** Many providers offer discounts off the sticker price for clients willing to commit to longer contract terms, typically around one to three years.
- **Monthly billing:** Monthly billing models are most common in the BMaaS market, where physical infrastructure typically implies steady state workloads without spiky characteristics.
- **By the hour/second:** The most common granularity for traditional cloud infrastructure, end users are charged only for what they use.
- **Transient/spot:** Some providers will offer up unused capacity at a discount via transient/spot instances, but those instances can be reclaimed if the capacity is needed.

Advantages

Several factors, when considered collectively, make cloud infrastructure seem like a good fit:

- **Pay-as-you-Go:** Unlike traditional IT, IaaS does not require any upfront, capital expenditures, and end users are only billed for what they use.
- **Speed:** With IaaS, users can provision small or vast amounts of resources in a matter of minutes, testing new ideas quickly or scaling proven ones even quicker.
- **Availability:** Through things like multizone regions, the availability and resiliency of cloud applications can exceed traditional approaches.
- **Scale:** With seemingly limitless capacity and the ability to scale resources either automatically or with some supervision, it's simple to go from one instance of an application or workload to many.
- **Latency and performance:** Given the broad geographic footprint of most IaaS providers, it's easy to put apps and services closers to your users, reducing latency and improving performance.

Typical use Cases

IaaS represents general purpose compute resources and is thus capable of supporting use cases of all types. IaaS is now mostly utilised for development and test environments, websites and web applications that are accessed by consumers, data storage, analytics, and data warehousing workloads, as well as backup and recovery, notably for on-premises workloads[34]. IaaS is also well suited for setting up and running popular commercial applications and software, such as SAP.

IaaS Services Infrastructure as a service helps companies to move their physical infrastructure to the cloud with

34. Available at: https://www.ibm.com/in-en/topics/iaas#:~:text=Infrastructure%2Das%2Da%2DService%2C%20commonly%20referred%20to%20as,as%2Dyou%2Dgo%20basis.

a level of control similar to what they would have in a traditional on-premise data center. As comparison to other service kinds, IaaS offers the most resemblance to the internal data centre. Storage, servers (or processing units), the network itself, and management tools for infrastructure upkeep and monitoring make up the core elements of a data center's infrastructure. Each of these elements has produced its own specific market niche.

IAAS: Storage

Companies can use storage services to store data on the storage equipment of third-party providers. Customers can access cloud storage online, which is displayed to them as a collection of storage pools or buckets, utilising sophisticated interfaces like command-line tools, web interfaces, or programming APIs[35]. Clients are unaware of the intricacy of the cloud storage architecture, although it is very sophisticated on the back end and often consists of distributed storage devices that are controlled by centralised software. Algorithms are used by sophisticated storage management software to manage data scattered across numerous storage devices.

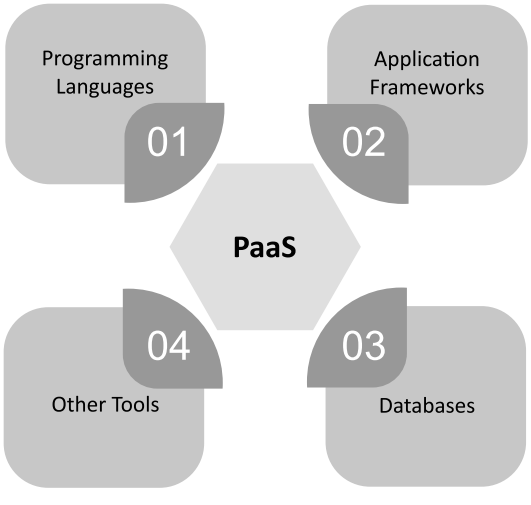
Network slowness, reliance on internet accessibility, security issues, and restricted control are all potential drawbacks. Due to the cloud provider's data center's distinct geographic location, network latency is greater than with internal storage. If a customer doesn't have a local copy of their data and instead saves it all in a public cloud, they are entirely dependent on internet connectivity. To prevent information loss or compromise, a cloud provider should provide high-level security, and data transit must be encrypted.

PAAS

Platform-as-a-Service, also known as PaaS, is a cloud computing model that offers customers a full cloud platform—including hardware, software, and infrastructure for creating, deploying, and managing applications without the expense, complexity, and rigidity that frequently accompany building and maintaining that platform on-premises. Without the hassles of updating the operating system, development tools, or hardware, PaaS offers everything developers require for application development. Instead, a third-party service provider uses the cloud to supply the whole PaaS environment or platform.

PaaS helps businesses avoid the hassle and cost of installing hardware or software to develop or host new custom applications. To build custom apps, development teams can easily obtain pay-as-you-go access to infrastructure, development tools, operating systems, and other resources. The result is simpler, faster, and secure app development that gives developers the freedom to focus on their application code.

PaaS includes infrastructure (servers, storage, and networking) and platform (middleware, development tools, database management systems, business intelligence, and more) to support the web application life cycle. **Example:** Google App Engine, Force.com, Joyent, Azure.



35. Available at: https://www.techtarget.com/searchcloudcomputing/definition/Infrastructure-as-a-Service-IaaS.

1. **Programming Languages**

   PaaS providers provide various programming languages for the developers to develop the applications. Some popular programming languages provided by PaaS providers are Java, PHP, Ruby, Perl, and Go.

2. **Application Frameworks**

   PaaS providers provide application frameworks to easily understand the application development. Some popular application frameworks provided by PaaS providers are Node.js, Drupal, Joomla, WordPress, Spring, Play, Rack, and Zend.

3. **Databases**

   PaaS providers provide various databases such as ClearDB, PostgreSQL, MongoDB, and Redis to communicate with the applications.

4. **Other Tools**

   PaaS providers provide various other tools that are required to develop, test, and deploy the applications.

Advantages of PAAS

The most commonly-cited benefits of PaaS, include**:**

1. **Simplified Development -** PaaS allows developers to focus on development and innovation without worrying about infrastructure management.
2. **Lower Risk -** No need for up-front investment in hardware and software. Developers only need a PC and an internet connection to start building applications.
3. **Prebuilt Business Functionality -** Some PaaS vendors also provide already defined business functionality so that users can avoid building everything from very scratch and hence can directly start the projects only.
4. **Instant Community** - PaaS vendors frequently provide online communities where the developer can get the ideas to share experiences and seek advice from others.
5. **Scalability** - Applications deployed can scale from one to thousands of users without any changes to the applications.

Disadvantages of PAAS

1. **Vendor Lock-in** - One has to write the applications according to the platform provided by the PaaS vendor, so the migration of an application to another PaaS vendor would be a problem.
2. **Data Privacy** - Corporate data, whether it can be critical or not, will be private, so if it is not located within the walls of the company, there can be a risk in terms of privacy of data.
3. **Integration with the rest of the systems applications** – It may happen that some applications are local, and some are in the cloud. Thus, there will be chances of increased complexity when we want to use data which in the cloud with the local data.

Use Cases for PAAS

By providing an integrated and ready-to-use platform and by enabling organizations to offload infrastructure management to the cloud provider and focus on building, deploying and managing applications, PaaS can ease or advance a number of IT initiatives, including:

- **API development and management:** Because of its built-in frameworks, PaaS makes it much simpler

for teams to develop, run, manage and secure APIs (application programming interfaces) for sharing data and functionality between applications.

- **Internet of Things (IoT):** Out of the box, PaaS can support a range of programming languages (Java, Python, Swift, etc.), tools and application environments used for IoT application development and real-time processing of data generated by IoT devices.
- **Agile development and DevOps:** PaaS can provide fully-configured environments for automating the software application lifecycle including integration, delivery, security, testing and deployment.
- **Cloud migration and cloud-native development:** With its ready-to-use tools and integration capabilities, PaaS can simplify migration of existing applications to the cloud particularly via *replatforming* (moving an application to the cloud with modifications that take better advantage of cloud scalability, load balancing and other capabilities) or *refactoring* (re-architecting some or all of an application using microservices, containers and cloud-native technologies).
- **Hybrid cloud strategy:** Hybrid cloud integrates public cloud services, private cloud services and on-premises infrastructure and provides orchestration, management and application portability across all three. The result is a unified and flexible distributed computing environment, where an organization can run and scale its traditional (legacy) or cloud-native workloads on the most appropriate computing model. The right PaaS solution allows developers to build once, then deploy and mange anywhere in a hybrid cloud environment.

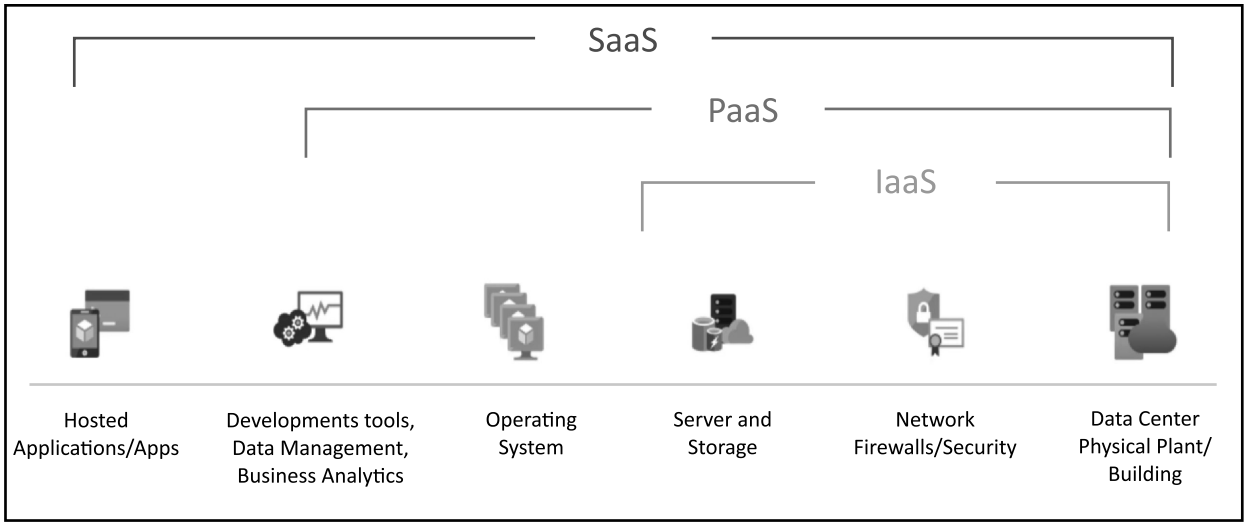
Purpose-Built PAAS Types

Many cloud, software and hardware vendors offer PaaS solutions for building specific types of applications, or applications that interacting with specific types of hardware, software or devices[36].

- **AIPaaS (PaaS for Artificial Intelligence)** lets development teams build artificial intelligence (AI) applications without the often prohibitive expense of purchasing, managing and maintaining the significant computing power, storage capabilities and networking capacity these applications require. AiPaaS typically includes pre-trained machine learning and deep learning models developers can use as-is or customize, and ready-made APIs for integrating specific AI capabilities, such as speech recognition or speech-to-text conversion, into existing or new applications.
- **iPaaS (integration platform as a service)** is a cloud-hosted solution for integrating applications. iPaaS provides organizations a standardized way to connect data, processes, and services across public cloud, private cloud and on-premises environments without having to purchase, install and manage their own backend integration hardware, middleware and software. (Note that Paas solutions often include some degree of integration capability API Management, for example – but iPaaS is more comprehensive.)
- **cPaaS (communications platform as a service)** is a PaaS that lets developers easily add voice (inbound and outbound calls), video (including teleconferencing) and messaging (text and social media) capabilities to applications, without investing in specialized communications hardware and software.
- **mPaaS (mobile platform as a service)** is a PaaS that simplifies application development for mobile devices. mPaaS typically provides low-code (even simple drag-and-drop) methods for accessing device-specific features including the phone's camera, microphone, motion sensor and geolocation (or GPS) capabilities.

36. Available at: https://www.techtarget.com/searchcloudcomputing/definition/Platform-as-a-Service-PaaS.

SAAS SERVICES



Software as a service (or SaaS) is a way of delivering applications over the Internet as a service. SaaS applications are also known as Web-based software, on-demand software, or hosted software. Common examples are email, calendaring, and office tools (such as Microsoft Office 365). It is a cloud-based software delivery model that allows SaaS applications to run on SaaS providers' servers instead of installing and maintaining software on-premises. The SaaS provider manages access to the application, including security, availability, and performance.

SaaS being a cloud based service where instead of downloading software your desktop PC or business network to run and update, you instead access an application via an internet browser. The software application could be anything from office software to unified communications among a wide range of other business apps that are available.

This offers a variety of advantages and disadvantages. Key advantages of SaaS includes accessibility, compatibility, and operational management. Additionally, SaaS models offer lower upfront costs than traditional software download and installation, making them more available to a wider range of businesses, making it easier for smaller companies to disrupt existing markets while empowering suppliers.

The major disadvantage of SaaS applications is that they ordinarily require an internet connection to function. However, the increasing wide availability of broadband deals and high-speed phone networks such as 5G makes this less of an issue. Additionally, some SaaS applications have an offline mode that allows basic functionality. Google Workspace, Trello, Zoom, DocuSign, Slack, Adobe Create Cloud, Mailchimp are some popular examples of SaaS products.

Characteristics of SAAS

1. **SaaS Multi-Tenant Architecture** - Multi-tenancy is an architecture where all SaaS vendor clients and applications share a single, common infrastructure and code base that is centrally maintained. This architecture allows vendors to innovate more quickly, saving development time previously spent on maintaining outdated code
2. **Easy Customisation with SaaS** - Users can easily customise applications to fit their business processes without affecting the shared infrastructure. A SaaS model supports each user and company's unique customisations changes and preserves them through regular upgrades. This means SaaS providers can make upgrades more often, with less customer risk and lower adoption costs.

3. **Better Access From Network Devices** - A SaaS model allows your business to remotely access data from any networked device, making it easy to manage privileges, monitor data use, and ensure many users can see the same information simultaneously.

4. **SaaS Harnesses the Consumer Web** – Anyone familiar with Amazon.com or My Yahoo! will be familiar with the Web interface of typical SaaS applications. With the SaaS model, you can customise with point-and-click ease, making the weeks or months it takes to update traditional business software seem hopelessly old-fashioned.

Advantages of SAAS

SaaS removes the need for organizations to install and run applications on their own computers or in their own data centers. This eliminates the expense of hardware acquisition, provisioning and maintenance, as well as software licensing, installation and support[37]. Due to the increased efficiency and cost-effectiveness of software as service applications, many businesses turn to cloud-based SaaS for solutions due to following reasons:

1. **Low Set Up and Infrastructure costs** - You only pay for what you need, so it is a very cost-effective solution for all-sized businesses.

2. **Scalability** - You can adapt your requirements to the number of people who need to use the system, the volume of data and the functionality required as your business grows.

3. **Accessible from Anywhere** - Just connect to the internet, and you can work from wherever you need to be via desktop, laptop, tablet or mobile or other networked devices.

4. **Automatic, frequent updates** - Providers offer timely improvements thanks to their scale and because they receive feedback about what their customers need. This frees up your IT department for other, more business-critical tasks.

5. **Security at the highest level required by any customer** - Because of the shared nature of the service, all users benefit from the security level set up for those with the highest need.

Future of SAAS

SaaS and cloud computing have come a long way in assisting businesses in creating comprehensive integrated solutions. Organizations are creating SaaS integration platforms (or SIPs) to build additional SaaS apps as knowledge and adoption of the model grow[38]. SaaS is one of many cloud computing remedies for enterprise IT problems. Other “as-a-Service” choices include:

- Infrastructure as a Service (IaaS) – the provider hosts hardware, software, storage and other infrastructure components.
- Platform as a Service (PaaS).
- Everything as a service (XaaS) – which is essentially all the “aaS” tools neatly packaged together.

The payment model for these kinds of services is typically a per-seat, per-month charge based on usage – so a business only has to pay for what they need, reducing upfront costs.

With companies adopting various “aaS” services, long-term relationships with service providers will grow, leading to innovation as customers’ evolving needs are understood and provided for. SaaS may one day help address critical business challenges, such as predicting which customers will churn or which cross-selling practices work best.

37. Available at: https://azure.microsoft.com/en-in/resources/cloud-computing-dictionary/what-is-saas/.

38. Available at: https://www.techradar.com/news/what-is-saas

On Premise Software

On-premises software is a type of software delivery model that is installed and operated from a customer's in-house server and computing infrastructure. Also meaning, thereby that subsequent maintenance, repairs, safety, and further updates are all handled on-site. It just needs a licenced or purchased copy of software from an independent software provider and makes use of the natural computer resources of the company. Software installed on-site is also referred to as shrink wrap. After buying software, the company often installs it on its servers, connecting relevant database software and configuring operating systems accordingly. Since there is no involvement of any third-party, the company has full ownership and responsibility.

On-premises software is the most prevalent, traditional method of using enterprise and consumer applications. On-premises software typically requires a software license for each server and/or end user. The customer is responsible for the security, availability and overall management of on-premises software. However, the vendor also provides after sales integration and support services. Because it requires onsite server gear, capital expenditures for software licences, onsite IT support employees, and longer integration times, onsite software is more expensive than on-demand or cloud software. On-premises software, however, is regarded as being more secure because the full instance of the software stays on the organization's grounds

In the past, on-premise software was the only solution available to companies. Today, that's changing, as more and more off-site solutions become popular, and cloud computing becomes the standard. There is now agreement amongst IT professionals that companies can't solely rely on on-premise applications. In any case, a mixture of off-premise and on-premise solutions, also known as a hybrid IT environment, will be the way forward.

Pros and Cons of On-Premise Software

Features	*Pros*	*Cons*
Cost	Overall costs in the long-term are lower	Substantial upfront investment required
Security	Companies can deploy their own security protocols	Technical IT support is required, increasing costs
Control	Full control to the user	Trained IT Staff is required to provide support
User Access	Internet connectivity is often not required for in-house solutions	This also means access isn't available on-the-go
Future-Proofing	Additional software can be purchased at extra costs	No updates are provided and new features are costly to add

LEGAL AND COMPLIANCE REQUIREMENTS OF SOFTWARE SECURITY

Under the Information Technology Act, 2000, anyone who "controls, processes and handles" the data must have a lawful basis to do the same and the same must be done within applicable data retention requirements[39].

All body corporates are required to adhere to The Information Technology (Reasonable Security Practices and Procedures and Sensitive Personal Data or Information) Rules, 2011.

39. Cybersecurity: Laws and Regulations, India, (November 14, 2022), ICGL, Available at: https://iclg.com/practice-areas/cybersecurity-laws-and-regulations/india.

The Information Technology (Reasonable Security Practices and Procedures and Sensitive Personal Data or Information) Rules, 2011 (hereinafter referred to as 'IT Rules, 2011') defines 'cyber incidents' under Rule 2(d) in the following words: *"Cyber incidents means any real or suspected adverse event in relation to cyber security that violates an explicitly or implicitly applicable security policy resulting in unauthorised access, denial of service or disruption, unauthorised use of a computer resource for processing or storage of information or changes to data, information without authorisation*[40]*"*

Rule 8 of the IT Rules 2011 talks about the Reasonable Security Practices and Procedures to be followed by body corporates and individuals. Rule 8[41] has been summarized herein below:

As per Rule 8(1) of IT Rules 2011, it is considered that Body Corporates and individuals have compiled with the reasonable security practices if they have such security practices and standards in place which corresponds with the information assets which are sought to be protected, keeping in mind the nature of the business[42]. It is obligatory to have a documented and elaborate information security programme and security policy containing *"managerial, technical, operational and physical security control measures"*[43].

It has been clarified in Rule 8(2) that an example of one such standard as referred to in sub-rule (1) of Rule 8 is the International Standard IS/ISO/IEC 27001 on "Information Technology - Security Techniques - Information Security Management System – Requirements"[44]. Furthermore, if any association or entity follows practices for data protection other than the standard practices prescribed by IS/ISO/IEC, then it is mandatory to get its code of best practices approved and notified by Central Government[45]. The Rules also make it mandatory to get an audit of reasonable security practices and procedures conducted by an auditor every year or as and when there is significant upgradation of the processes and computer resources of a body corporate[46].

Moreover, the Data Security Council of India (DSCI) regularly publishes standards and best practices in cyber security, which can be kept in mind while developing security practices.

Reporting of Incidents:

The Information Technology (The Indian Computer Emergency Response Team and Manner of Performing Functions and Duties) Rules, 2013 (for brevity, 'CERT-In Rules') prescribes the procedure of function of CERT-IN, which is the Computer Emergency Response Team, appointed by Central Government in accordance with Section 70B of the Information Technology Act, 2000. The CERT-In has been designated as the Nodal Agency for performing certain cyber security functions, some of which have been enumerated below:

- Handling cybersecurity incidents through emergency measures.
- Issuing guidelines, practices and advisories on cybersecurity and safe practices.
- Flagging, forecasting and alerting cybersecurity incidents, etc[47].

Rule 12 of the CERT-In Rules provides for a 24-hour Incident Response Helpdesk for reporting of cyber-security

40. The Information Technology (Reasonable Security Practices and Procedures and Sensitive Personal Data or Information) Rules, 2011, Available at: https://www.meity.gov.in/writereaddata/files/GSR313E_10511%281%29_0.pdf.

41. Ibid.

42. Ibid.

43. Ibid.

44. Supra, Note 43.

45. Rule 8(3), Ibid.

46. Rule 8, The Information Technology (Reasonable Security Practices and Procedures and Sensitive Personal Data or Information) Rules, 2011.

47. The Information Technology Act, 2000 and The Information Technology (The Indian Computer Emergency Response Team and Manner of Performing Functions and Duties) Rules, 2013.

incidents[48]. Certain cybersecurity incidents have been identified in the annexure to the CERT In Rules and the same shall be mandatorily reported to the Incident Response Helpdesk as soon as possible[49].

Rule 12(a) reads as follows[50]:

"Reporting of incidents: Any individual, organisation or corporate entity affected by cyber security incidents may report the incident to CERT-In. The type of cyber security incidents as identified in Annexure shall be mandatorily reported to CERT-In as early as possible to leave scope for action. Service providers, intermediaries, data centers and body corporate shall report the cyber security incidents to CERT-In within a reasonable time of occurrence or noticing the incident to have scope for timely action"[51].

The Annexure to CERT-In Rules provides for mandatory reporting of the following incidents:

a. "Targeted scanning/ probing of critical networks/ systems;
b. Compromise of critical systems/ information;
c. Unauthorised access of IT systems/ data;
d. Defacement of website or intrusion into a website and unauthorised changes such as inserting malicious code, links to external websites etc.;
e. Malicious code attacks such as spreading of virus/ worm/ Trojan/ Botnets/ Spyware;
f. Attacks on servers such as Database, Mail and DNS and network devices such as Routers;
g. Identity Theft, spoofing and phishing attacks;
h. Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks;
i. Attacks on Critical infrastructure, SCADA Systems and Wireless networks;
j. Attacks on Applications such as E-Governance, E-Commerce etc"[52].

The Ministry of Electronics and Information Technology has come out with directions regarding compliances and reporting obligations passed vide Notification No. 20(3)/2022-CERT-In dated 28.04.2022[53]. The notification has made it mandatory to report cyber incidents as listed above within a period of six hours of the incident coming into notice or being brought into the attention of the concerned person. This has been done to ensure prompt action to the cyber security incident and prevent delays in investigation and consequent action. The new 6 (six) hour deadline does not have retrospective effect and will apply only to cyber security incidents that take place on or after June 27, 2022[54].

48. Supra, Note 42.

49. Ibid.

50. CERT-IN'S SIX HOUR REPORTING RULE FOR CYBER SECURITY INCIDENTS- Statutory Interpretation and Analysis, Argus Partners, Available at: https://www.argus-p.com/uploads/blog_article/download/1664436637_Reporting_CyberSecurity_Incidents_in_India-_Statutory_Interpretation_and_Analysis.pdf.

51. The Information Technology (The Indian Computer Emergency Response Team and Manner of Performing Functions and Duties) Rules, 2013.

52. Ibid.

53. CERT-IN'S SIX HOUR REPORTING RULE FOR CYBER SECURITY INCIDENTS- Statutory Interpretation and Analysis, Argus Partners, Available at: https://www.argus-p.com/uploads/blog_article/download/1664436637_Reporting_CyberSecurity_Incidents_in_India-_Statutory_Interpretation_and_Analysis.pdf.

54. Supra, Note 56.

LESSON ROUND-UP

- Any kind of external intervention in software systems can seriously threaten the security of data and make it prone to unauthorized access.
- Software refers to set of instructions which enable the hardware components to perform.
- System software serves as a point of interaction between the hardware components of computer and user applications.
- Operating system is basic and essential to the functioning of a computer as all computer applications sit inside the operating system of a computer.
- Application Software are dedicated to the performance of a particular task or function.
- Utility Software assists the system software in performing its work.
- Software Security refers to the practice of developing and engineering the software in a manner which keeps it secure from external malicious attacks, while also ensuring that in case of any such attack, the software does not malfunction and continues to operate.
- Proactive security aims to prevent any data breach incident before any malware is able to access network server or before vulnerabilities in a software are exploited by hackers.
- Infrastructure-as-a-Service, abbreviated as "IaaS," is a type of cloud computing where basic computing, networking, and storage resources are made available to users on demand through the internet and on a pay-as-you-go basis.
- Platform-as-a-Service, also known as PaaS, is a cloud computing model that offers customers a full cloud platform—including hardware, software, and infrastructure for creating, deploying, and managing applications without the expense, complexity, and rigidity that frequently accompany building and maintaining that platform on-premises.
- Software as a Service (or SaaS) is a way of delivering applications over the Internet as a service. SaaS applications are also known as Web-based software, on-demand software, or hosted software. Common examples are email, calendaring, and office tools (such as Microsoft Office 365).
- On-premises software is a type of software delivery model that is installed and operated from a customer's in-house server and computing infrastructure.

GLOSSARY

Defects: A defect refers to a mistake made by developer while developing the software which leads to differences in intended results and actual results.

Bugs: A bug is an error, problem or defect in the design or development of a software which hampers its ability to produce the desired result or results into producing invalid results.

Failure: It refers to the inability of a software to perform its designated.

Flaws: A flaw refers to a problem in the software code which makes the software prone to security risks. Software flaws can be rectified by updates formulated by software developer.

Vulnerabilities: It refers to exploitable points within the software which can be targets for potential attack by hackers.

Software: It is the collection of instructions which directs the computer as to 'which' task to perform and 'how' to perform it.

System Software: These software enables interaction of the user with hardware components of the computer and helps in running programs on the computer.

Application Software: These are software which are dedicated to the performance of a particular task or function.

Utility Software: A utility software is a type of application which assists the system software in performing its work.

Software Security: It refers to the practice of developing and engineering the software in a manner which keeps it secure from external malicious attacks, while also ensuring that in case of any such attack, the software does not malfunction and continues to operate.

TEST YOURSELF

(These are meant for recapitulation only. Answer to these questions are not to be submitted for evaluation.)

1. Differentiate between hardware and software.
2. What are the functions of an operating system?
3. Discuss the classification of software on the basis of availability and shareability.
4. Differentiate between a Compiler and Interpreter.
5. Discuss best practices in software security.
6. Enumerate software security goals.
7. Discuss about SaaS, PaaS and IaaS.
8. Write a short note on on-premise software.
9. Write a short note on legal and compliance requirement of software security.

LIST OF FURTHER READINGS

- Mathias Payer, "Software Security: Principles, Policies and Protection", July 2021, Available At: https://nebelwelt.net/SS3P/softsec.pdf
- LeBlanc, John Viega, 19 deadly sins of software security. McGraw-Hill, 2005.
- Gary McGraw, Software Security: Building Security In. Addison-Wesley Professional, 2006.