# Mobile Application Security

## Building security into the development process

Rajneesh Mishra
Senior Consultant - Secure Mobile

SDG
[ technology + passion ] - risk

Mobile devices have outnumbered PCs and laptops to become the primary medium for accessing content & services. Businesses are already in the process of developing mobile applications to attract new customers and to increase employee productivity by making corporate applications and information available on mobile devices. According to Gartner, by 2017 mobile applications will be downloaded more than 268 billion times and generate more than $77 billion dollar business.

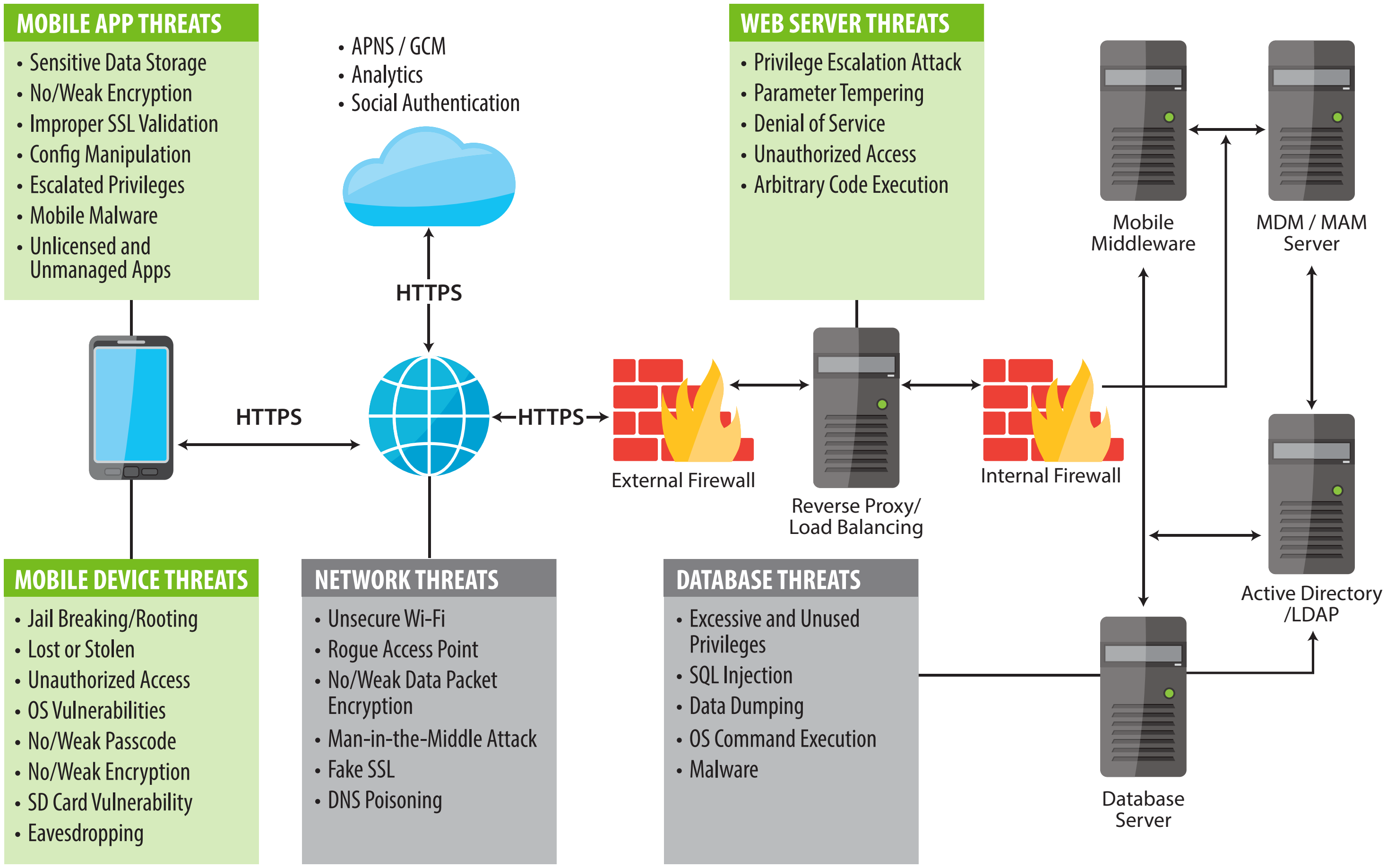
*...fast paced development of mobile applications has introduced a major security concern for businesses, especially businesses that have a 'bring your own device' (BYOD) policy for their employees.*

This fast paced development of mobile applications has introduced a major security concern for businesses, especially businesses that have a 'bring your own device' (BYOD) policy for their employees.  Cybercriminals can exploit mobile devices or applications to steal information and to harm customers. Recent cyber-attacks have alarmed businesses and forced them to rethink mobile security, as the traditional security models employed to protect information accessed by off-site/remote workers, do not seem to be effective any more.

## Security threats to the mobile enterprise

There are a variety of security threats that can affect mobile devices. These mobile threats are ever evolving and can be physical or software based and can target mobile applications, the mobile device, the network the data center or any combination of these. The diagram on the next page illustrates this point and demonstrates the need for a multi-layered approach to protect against security breaches.

This paper focuses specifically on mobile device and application security and some of the major vulnerabilities and commonly used methods for remediation. It then goes on to discuss the need for an organization to have a clearly defined security and risk posture that can be used to drive the development of secure applications by  integrating security very early into the SDLC.



**MOBILE APP THREATS**

- Sensitive Data Storage
- No/Weak Encryption
- Improper SSL Validation
- Config Manipulation
- Escalated Privileges
- Mobile Malware
- Unlicensed and Unmanaged Apps

- APNS / GCM
- Analytics
- Social Authentication

**WEB SERVER THREATS**

- Privilege Escalation Attack
- Parameter Tempering
- Denial of Service
- Unauthorized Access
- Arbitrary Code Execution

**MOBILE DEVICE THREATS**

- Jail Breaking/Rooting
- Lost or Stolen
- Unauthorized Access
- OS Vulnerabilities
- No/Weak Passcode
- No/Weak Encryption
- SD Card Vulnerability
- Eavesdropping

**NETWORK THREATS**

- Unsecure Wi-Fi
- Rogue Access Point
- No/Weak Data Packet Encryption
- Man-in-the-Middle Attack
- Fake SSL
- DNS Poisoning

**DATABASE THREATS**

- Excessive and Unused Privileges
- SQL Injection
- Data Dumping
- OS Command Execution
- Malware

## Mobile Device Security Vulnerabilities

Mobile devices face security threats that take advantage of vulnerabilities found in these devices. These vulnerabilities can be the result of poorly implemented technical controls or lack of awareness. The following is a list of mobile device vulnerabilities:

**Lost or Stolen Devices**: With a growing mobile workforce, there is a huge security risk to the enterprise associated with lost or stolen employee devices.

**Mobile Device Password**: Mobile devices often lack passwords to authenticate users and control access to data stored on the devices. Many devices have the technical capability to support passwords, personal identification numbers (PIN), or pattern screen locks for authentication. Some mobile devices also include a biometric reader to scan a fingerprint for authentication but users seldom employ these mechanisms

**Wi-Fi Transmission:** Wireless transmissions are not always encrypted. Information such as e-mails sent by a mobile device is usually not encrypted while in transit. In addition, many applications do not encrypt the data they transmit and receive over the network, making it easy for the data to be intercepted.

**Mobile devices Malware:** Users may download applications that contain malware. It is difficult for users to tell the difference between a legitimate application and one containing malware.

**OS Vulnerabilities:** In past few months, major security vulnerabilities have been discovered in popular operating systems like iOS, Android. Though Apple and Google fixed the issue and released the patch but still there are many devices having outdated OS. Security patches or fixes for mobile devices' operating systems are not always installed on mobile devices in a timely manner due to device or vendor restriction.

**Jail-breaking/Rooting Mobile Device:** Mobile devices may have unauthorized modifications. The process of modifying a mobile device to remove its limitations so consumers can add features changes how security for the device is managed and could increase security risks.

## Mobile Device Security Risk Mitigation

Security threats to the mobile device can be largely mitigated with the implementation of mobile device management tools (MDM), strong governance policies that leverage MDM as well as a comprehensive employee education and awareness program. The following is a sample of some of the controls provided by MDM.

The ability to:

- **Remotely lock a mobile device or erase data from a lost or stolen mobile device.**
- **Configure or disable Wi-Fi and VPN.**
- **Enforce folder, device or email encryption settings.**
- **Use geo-fencing or time-fencing rules to enforce location or time related compliances.**
- **Detect and restrict jail broken and rooted devices.**
- **Enforce password policies.**
- **Blacklist/ Whitelist applications.**
- **Disable risky interfaces on the device.**

## Mobile Application Security Vulnerabilities

Mobile applications offer a level of convenience that has never been known before. This extreme level of convenience has brought with it an extreme number of security risks as user's (or a client's) personal information such as credit card details, bank logins, passwords and more are flying between devices and backend databases and systems across the net.

The cause of these security risks can be largely classified into the following:

**Insecure data storage:** This can result in stolen user data from an application that is improperly secured.  Examples of data that are at risk are - usernames, passwords, authentication tokens, location data, personal information or application data.

**No encryption or weak encryption:** Encryption systems are constantly evolving because they are constantly being "solved" or broken. Apps that allow the transmission of unencrypted or weakly encrypted data are vulnerable to attack.

**Poor Authorization and authentication:** Apps and the systems they connect with should be properly protected with authorization and authentication best practices. This ensures that un-authorized devices, users and scripts are identified and blocked.

**Improper or insufficient transport layer protection:** Mobile applications are usually designed to exchange data in a client-server protocol. When this data is exchanged it travels across the carrier network and the internet. If the application is coded poorly, and not secured, "threat agents" can use techniques to view sensitive data while it's traveling across the network. These threat agents can be users or entities on a network; or they may be malware that pre-exist on the user's mobile device

**Client side injection:** If an app is improperly coded, attackers can mount simple text-based attacks that target almost any source of data including resource files or the application itself.

**Unintended permissions:** Misconfigured apps can sometimes open the door to attackers by granting unintended permissions.

**Escalated privileges:** A hacker could exploit a bug, design flaw or configuration oversight in an app to gain access to resources normally protected from an application or user.

## Mobile Application Security Risk Mitigation

Security threats can be mitigated with strong, well-understood solutions. The following list encompasses some of the most commonly used measures designed to protect data and systems from a variety of different attack methods.

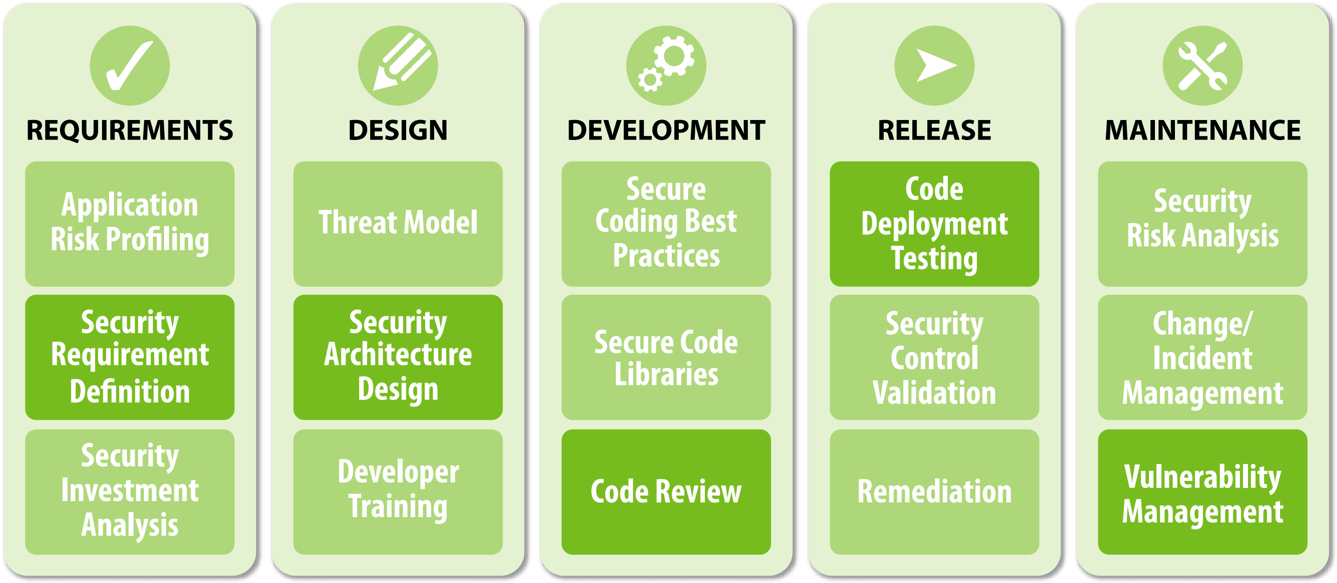
- **Storing sensitive data securely, or not at all.**
- **Handling authentication and sessions properly.**
- **The correct use of security and encryption tools.**
- **Avoiding unintended information leakage.**
- **Resisting runtime manipulation.**
- **Leveraging code obfuscation and anti-tampering to prevent reverse engineering.**
- **Validating the security/authenticity of 3rd party code/libraries.**
- **Using tools to perform security tests and security code scans.**
- **Following the OWASP mobile security project to keep you updated with evolving mobile security risks and ways to protect yourself.**

Sometimes it takes a hack attack to realize that an app is not secure and that proprietary information is vulnerable. In many software development efforts, security is implemented as an afterthought, or security issues are identified and remediated at the end of development, during the testing phase of the software development lifecycle (SDLC). This reactive mode can be dangerous and costly, wasting both time and money. To avoid this, an enterprise must, first, have a clearly defined security strategy based on their security and risk posture; and, second, this strategy must be understood and adopted by development teams while also being tightly integrated over the entire software development lifecycle (SDLC).

## Building Security into the Application Development Process

Once an organization has established a security and risk strategy, it can then model and calibrate the Secure Software Development Life Cycle (S-SDLC) according to their needs. Irrespective of the methodology being used, waterfall, agile or iterative, security related tasks and activities can be integrated into the phases of the S-SDLC process as shown in figure below. In each phase, specific security related activities take place to ensure that security is built into the application to ensure confidentiality, integrity, and availability. The goal of a good SDLC process is to capture, verify, and implement all the requirements, including security requirements needed to make the application useful to the organization. If security requirements are correctly identified and implemented, the result will be a secure application.

The sections below offer an example of a typical set of activities that can take place in each phase of an SDLC. This is by no means a comprehensive list of activities, but rather, it highlights some of the common activities integrated into the S-SDLC.



| REQUIREMENTS | DESIGN | DEVELOPMENT | RELEASE | MAINTENANCE |
|---|---|---|---|---|
| Application Risk Profiling | Threat Model | Secure Coding Best Practices | **Code Deployment Testing** | Security Risk Analysis |
| **Security Requirement Definition** | **Security Architecture Design** | Secure Code Libraries | Security Control Validation | Change/ Incident Management |
| Security Investment Analysis | Developer Training | **Code Review** | Remediation | **Vulnerability Management** |

The following is a brief description of activities that are highlighted in bold in the above diagram:

**Security Requirement Definition** is focused on specifying the behavior of an application with respect to security. Organizations must ensure that the requirements are specific, measurable and reasonable and that they conform to their security, risk and compliance strategy.

**Secure Architecture Design** is focused on proactive steps for an organization to design and build a secure application. By enhancing the design process with reusable secure services and components, the subsequently developed application will be more secure, whilst the time and effort will be dramatically reduced.

**Code Review** is focused on the inspection of the application at the source code level in order to find security vulnerabilities. Organizations should use lightweight checklists for common problem and also use automation technology to improve coverage and efficacy of code review activities.

**Security Testing** is focused on the inspection of the application in the runtime environment in order to find security problems. Organizations should specify security test cases based on known requirements and common vulnerabilities, and also perform application penetration testing before each major release.

**Vulnerability Management** is focused on the processes within an organization that handle vulnerability reports and operational incidents related to security. To effectively implement these processes, an organization should define a security point of contact for the application and also create an informal security response team to handle security incidents.

It is always advisable to follow a well-established secure development life cycle (Secure SDLC) process, mandated by company policy, audited by internal information security department and tested by external security teams. This process must contain at least one checkpoint at every relevant phase.

While at first glance it may seem that adding security activities to the SDLC will add more checkpoints and therefore require additional time to deliver the application, in reality, its quite the opposite. This is because much time has been saved from re-engineering the application to plug security leaks that would be discovered during testing, or worse after the release of the application. Development time can be further reduced by leveraging secure, reusable application components and through the use of automation tools for scan, test and delivery processes.

## Conclusion

Mobile application security is complicated, it is not just the code running on the devices, there are innumerable other factors like the device platform, web-services, cloud based 3rd party services etc., which play a very important role in mobile application security. Organizations should perform a detailed analysis of their risk posture against all possible known security threats to an application and use this to create a mobile security strategy. This strategy should then translate to the creation of a custom S-SDLC for the development of organization mobile applications. By putting an S-SDLC in place, mobile application vulnerabilities can be identified and eliminated well in advance of deploying the application, thereby resulting in considerable saving on investment.

## About SDG

SDG is a leading provider of technology, consulting and risk management solutions to strengthen enterprise businesses while managing IT risk. We focus on six practices: Risk and Security; Identity and Access Governance; Digital Collaboration; Quality Assurance; Mobility and Cloud. In addition we offer a GRC solution, called TruOps, to manage enterprise IT risk and compliance.

> *"SDG helps enterprises realize their dreams by helping them develop, manage and deploy solutions with acceptable risk."*

For over two decades, SDG has enabled enterprises to realize their dreams by helping them develop, manage and deploy solutions with acceptable risk. We combine technology, thought leadership and a relentless passion for customer success. SDG partners with enterprise brands, but we specifically focus on mitigating client IT risk. Our ultimate goal is to help enterprises realize the opportunity of technology, increase innovation, improve speed-to-market and maximize returns on investment.