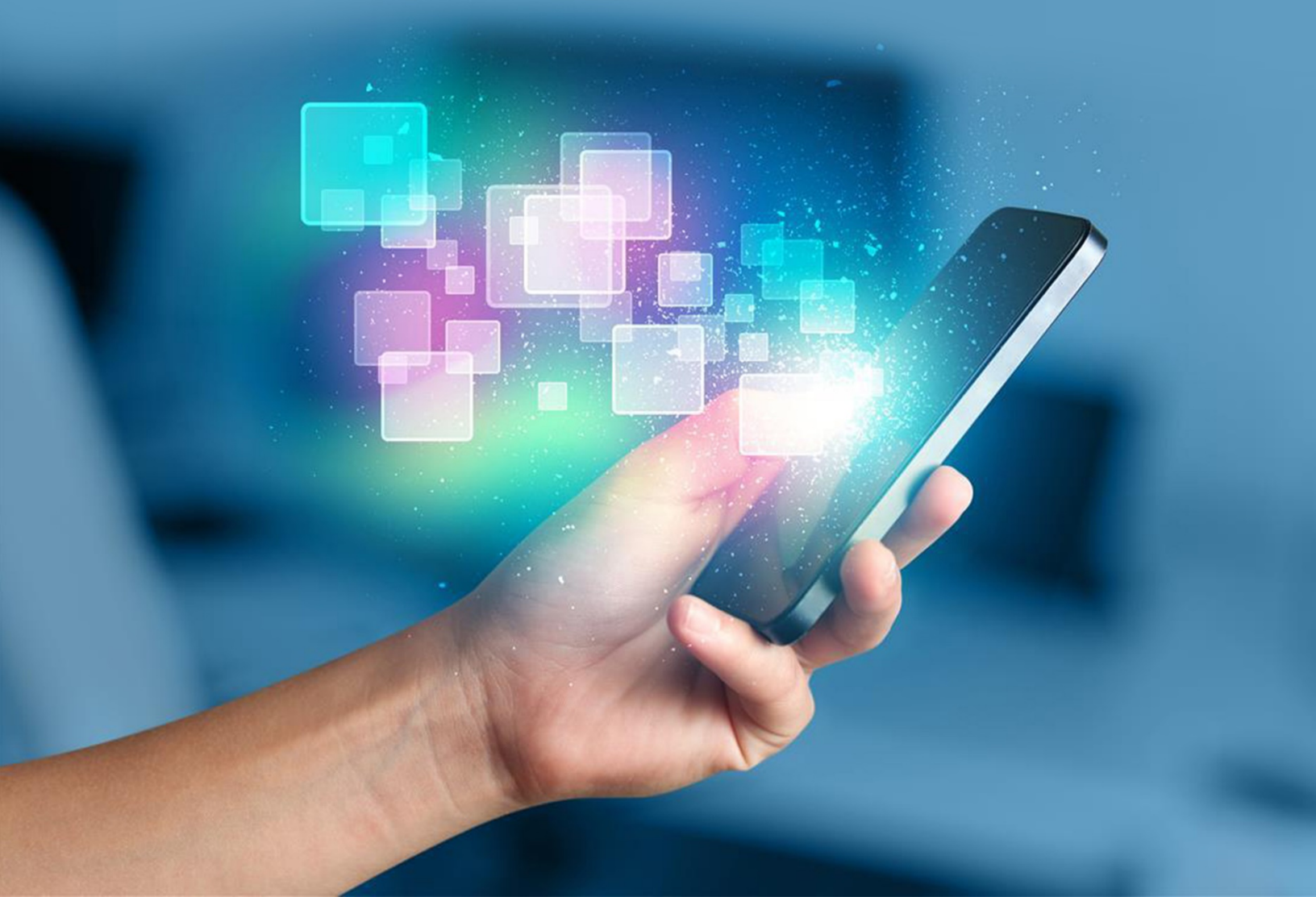
SECURE MOBILE APPLICATION DEVELOPMENT GUIDELINES

LEAN SECURITY

www.leansecurity.com.au



LEAN SECURITY
Web and Mobile Application Security Assurance

CONTENTS

INTRODUCTION:

The modern world has been marked by the popularity of mobile application development. These developments have made the scope of mobile phones widened, which have been used to make voice calls only. Smartphones' popularity was led to the development of new applications like games, internet browsing, and email. With the rapid development and growth of PDAs and smartphones, mobile technology has resulted in the increase of the requirement of advanced applications. The newest technologies that are used these days include J2EE, C++, and Dot Net, with lots of companies providing secured applications for various platforms.

Mobile companies have as well encountered with the challenge of meeting the increasing expectations for secured and innovative mobile applications. Along with the increasing expectations of customers, the ever-changing mobile technology put an additional pressure on the developers. In order to stay ahead of the competitors, different companies have been trying on creating more portable and innovative applications. It has resulted in a trend to outsource the development of mobile application development specialized companies for wireless devices.

The list of applications that can be outsourced include image and video sharing, integrated billing solutions, wireless internet security, content management, gaming, and location-based services. Companies have been engaged to develop applications through leading-edge tools and technologies for providing unfailing and consistent levels. Secure mobile application development involve a variety of dynamic and innovative approaches. It can become possible with the help of newest mobile gadgets. In developing mobile devices, the greatest concern is the security on the wireless devices. Mobile application security is developed with a target of securing phones in different threats like malware, viruses, OS exploits, and more.

There are 2 kinds of risks in mobile security. One is the vulnerabilities, which are the errors in design that is exposing the data to interception by attackers. The other is the malicious functionality, which can be defined as a list of unwanted behaviors in the mobile code. In order to get this issue solved, the developers have implemented a step-by-step approach to make sure having secure mobile applications.

There are several resources available online, which are featuring code examples, case studies, and best practices, in order to provide security in mobile applications. One will be able to find many information about how they can protect against vulnerabilities in the newest PDA and smartphone platforms.

With the help of guidelines, individuals will be able to provide security to wireless and mobile devices. The modern world is very competitive and dynamic, and developers need more than theoretical knowledge so as to get themselves ready at the IT industry. Here are the top 10 mobile risks that one needs to be careful of for the success of mobile application development.

MAJOR MOBILE RISKS THAT ARE REALLY A PAIN IN THE NECK

The changes in the advanced technology that we have now in the 21st century have brought not only advantages but also disadvantages. One of the best things that the 21st century has provided us is the ease of communication, online business and other exciting applications. The mobile industry is really highly volatile and whirling and a lot of users have 100% satisfaction with it, most especially with the mobile application developments which is now a very vital factor for small and large businesses alike. A mobile device will be just nothing and useless with an application, and when an application is already completed and installed, users are guaranteed that this program will run smoothly and with unique functionalities.

As the emergence of mobile devices has made it easier for us to communicate around the world anytime, so are the mobile security risks that just keep on increasing each time. Malware, scams, and hackers are just some of the bad elements online and in the mobile industry which can really be such a big pain in the neck. The evolution of the mobile phone communications has now already reached a much different level and manufacturers of such devices are working really hard both night and day thinking, developing, creating and commercializing possible new models.

There are some providers who make very sophisticated mobile devices that can be integrated with mobile applications and aside from that feature, these modern mobile phones can also be a simple and small pocket-computer devices because it is already possible for us to place computer chips on it. In short, these devices are designed not only for communication purposes but also to facilitate services which include research, entertainment, photography jobs and other possible tasks. Mobile phones like androids, smartphones and window phones are also designed to provide the users with an extra comfort. A lot of people love such devices because they are not only fashionable and attractive but they are also very multi-tasked.

With today's highly socialized society and modern world where the advanced technology is already a basic part of our daily lives, having a mobile phone is a necessity if you don't want to be left behind and be out of place in a generation where almost everybody that always has a mobile phone with them, wherever and whenever.

The mobile apps that we enjoy today truly offer a great level of convenience that we have never ever known before. From the office, while on the road, at the comfort of our comfort zone which is our home, and even from the hotel room in other countries while you are on vacation- you can certainly login right to your voicemail at work, in viewing your bank balance, in checking your credit card balance, in buying new clothes, in book travel and a whole lot more.

What most of us don't know is that all of these modern modifications of mobile phones are badly prone to render a number of security risks. The extreme levels of convenience that these mobile apps that we enjoy so much have brought with it also a very extreme number of risks that can really be such a hard pain in the neck as user's credit card information and details, passwords, bank logins and a lot more are flying right between the devices and the backend databases as well as systems all across the net. These are the hardest thing about the modern technology, specifically the mobile apps that we have today. It is very important that we try to understand these possible mobile risks because it can help us to prepare our app and to protect ourselves as well, our data and our users.

Top mobile risks that you must be fully aware of includes the following:

WEAK SERVER SIDE CONTROLS

This is the considered as the second most important and common mobile security threat that you must know about. In some ways, this is more familiar than the other existing mobile security risks or threats. As to what the experts had explained, most mobile enterprise applications out there rely on back-end services and with this fact, these applications is just similar somehow to the traditional client-server applications. The only difference here is that the mobile developers do not always take the traditional server-side security matters or considerations right into account.

The possible technical effects or impacts of this mobile vulnerability directly correspond to the technical impacts of a certain associated vulnerability as well that the adversary can be exploited through the device. The impact also of this vulnerability in terms of business corresponds to the overall business impact of a certain associated vulnerability which the adversary also exploits via the mobile phone device. A lot of people wonder if they are vulnerable to this particular kind of mobile risks, the truth is that, we do.

Weak server side controls are so prevalent in the world of mobile applications and the factors or reasons that have led to the proliferation of the weak server side include the rush to market, easy and fast access to frameworks that don't really prioritize security, lack of security knowledge due to the newness of the language, higher than the average outsourced development, lower security budgets intended for mobile applications, weakness or vulnerability due to the cross-platform development and compilations, and assumptions that mobile OS 100% takes full responsibility for security.

HOW CAN YOU PREVENT IT FROM HAPPENING?

This risk is quite plain simple: the possible servers that your app is accessing in the meantime must have an effective security measures right in place in order to prevent the unauthorized users from getting or having an access in your data. This includes the server that you are using, and as well as the 3rd party systems that your app may be possible accessing. Secure coding and as well as configuration services should be used on the server-side of any mobile application.

INSECURE DATA STORAGE

This is another serious mobile threat that you must also be aware of. The common pieces of information or data that are stored in your mobile phone and may possibly be at risk are the usernames, passwords, authentication tokens, cookies, location data, Network Connection name, EMEI/UDID, device name, and personal information which include the address, social, credit data and DoB. Application data such as transaction histories, stored application logs, cached application messages and debug information are also at risk here.

Insecure data storage can lead or result to data loss for a certain user when, for example, he loses his phone or if there are multiple users of the device, if that is the case then the app is secured improperly which leaves the user highly at risk.

This particular threat can mean a big problem once it happen to you, especially if you own a business because insecure data storage vulnerabilities can typically lead to a number of business risks right for the company or organization that own the risk app. These business risks include fraud, identity theft, material los, reputation damage and PCI or External Policy Violation.

ARE YOU A RISK OR VULNERABLE OF THIS PARTICULAR MOBILE RISK?

It is very important for you to threat-model your certain mobile app to fully understand that data and necessary information assets it processes and as well as how the underlying number of APIs which they assets. These certain APIs must strictly store all the sensitive information and data securely. The data that are stored insecurely most of the time include the log files, SQLite databases, Plist flies, Binary data stores, XML data stores, cookie stores, cloud synced and SD card.

When you are applying an encryption or a decryption to a certain sensitive information asset, problems like malware may possibly perform a binary attack right on your app to steal decryption or encryption keys. Once it is already done with its scheme to steal the keys, it will then decrypt your local data and get the sensitive information.

POSSIBLE WAYS TO PREVENT MOBILE THREATS LIKE INSECURE DATA STORAGE

The number one rule of mobile apps is to never store any data or information unless it is absolutely necessary or needed. In today's risky advanced technology world, we have to assume that the data is already forfeited once it touches the phone and we also have to consider the possible implications of losing our user's data to a root exploit or a silent jailbreak. If its usability vs. security trade-off is already too much for you, it is highly recommended that you scrutinize your platform data security APIs and make sure that you are calling them appropriately. The major lesson here is that you must know what information and data are being stored and as well as to protect it immediately and appropriately.

INSUFFICIENT TRANSPORT LAYER PROTECTION

When designing a certain mobile phone application, commonly, most data is being exchanged in a certain client-server fashion. In time that this data is finally exchanged, it will travel across the internet and the carrier network. You can expect that if the application is not secured and coded poorly, some of the online threat agents will use some techniques that will view the sensitive data while it is still traveling all across the wires. These threat agents include the users that are local to your network and may be compromised or in monitored wifi, network devices or carrier which include cell towers, routers, proxies and others, and the last is malware pre-existing right on the user's phone.

This flaw poses or exposes a certain user's data that can possibly lead to a big problem like, account theft. The entire site can be exposed also if the adversary will intercept a certain admin account and a poor SSL setup can possibly facilitate phishing and as well as MITM attacks. In terms of business impacts, the interception of the sensitive data through the communication channel can also result in privacy violation which may result in fraud, identity theft, and reputation damage.

Ways to prevent insufficient transport layer protection from happening to you and the effective general practices:

- Assume that your network layer is not secured and is 100% susceptible to things like eavesdropping.
- Apply TLS/SSL to transport channels which the mobile app will use in transmitting session tokens, sensitive information, and other sensitive data to a certain backend API or kind of web service.
- Account for a certain outside entity like a 3rd party analytics company, social network and other by the use of the SSL version when a certain application smoothly runs through the webkit or browser. Always avoid the mixed SSL sessions because they may possibly expose the session ID of the user.
- Use only a strong and industry standard cipher suites with an appropriate ley lengths and as well as certificates which are signed by a certain trusted CA provider.

- Don't ever allow self-signed certificates and always consider a strict certificate pinning for some security conscious apps.
- Make it a habit to always require an SSL chain verification.
- Establish only a secure connection right after verifying the identity of a certain endpoint server with the use of the trusted certificates in a key chain.
- Alert the users via UI if a mobile app detects a certain invalid certificate.
- Don't send sensitive data over the alternate channels such as MMS, SMS, or notifications.
- As much as possible, apply a separate layers of encryption to any sensitive data right before it is given to an SSL channel. In time that some vulnerabilities are detected or discovered in an SSL implementation, your encrypted data will then provide a certain secondary defense against the confidentiality violation.

IOSBEST PRACTICES OF PREVENTION

- Always ensure that the certificates are all valid and fail immediately closed.
- When using the CFNetwork, always consider to use a Secure Transport API in order to designate the trusted client certificates. Most of the time, NSSTreamSocketSecurityLevelITLSv1 must be used for a much higher standard cipher strength.
- After the development, ensure all of the NSURL calls and never allow the self-signed or the invalid certificates.
- You have to consider also the use of certificate pinning by exporting your certificate, including your app bundle, and then anchor it right to your certain trust object.

ANDROID BEST PRACTICES OF PREVENTION

- Removing all the code right after the development cycle will allow the app to accept all the possible certificates.
- If you are using a class which truly extends the SSLSocketFactory, always make sure that the checkServerTrusted method is implemented properly so that your server certificates will be checked correctly.

UNINTENDED DATA LEAKAGE

This happens when vulnerabilities such as mobile malware, modified versions of some legitimate apps are existent. Another reason is when a certain adversary which has a physical access right to the mobile device of the victim. When this vulnerability happens, it may result to some technical impacts like extraction of the application's sensitive information through the mobile malware, modified applications and some forensic tools. A sensitive information theft can also result to a number of business impacts which may really be a big problem such as privacy violations, fraud, PCI Violations and reputation damage. Watching how, where and when data moves, the attackers can find as well as exploit security holes.

WHO ARE VULNERABLE TO THIS KIND OF MOBILE RISK?

This mobile risk which is formerly called side-channel data leakage include vulnerabilities from frameworks, OS, new hardware, compiler movement, and others without the developer's knowledge. In terms of the mobile development, this problem is most seen in some undocumented or rather under-documented internal processes like:

- The way how OS caches images, data, key-presses, buffers and logging.
- The way how the development framework caches images, data, logging, key-presses and buffers.
- The way and amount of data ad, social, analytic, social or enablement frameworks cache images, data, logging, key-presses and buffers.

HOW TO PREVENT IT?

It is very important to threat-model your platforms, OS, and frameworks in order for you to see how they handle different types of features which are the following:

- URL caching, both response and request
- Keyboard press caching
- Logging
- Copy and paste buffer caching
- Application backgrounding
- HTML5 data storage
- Browser cookie objects
- Analytics data sent to third parties

Aside from that, it is also very important that you discern what a certain given framework or OS does by default.

POOR AUTHORIZATION AND AUTHENTICATION

Reputation al damage, fraud and information theft are just some of the problems that may be caused by this kind of mobile risk. Systems and apps connect and they should be protected by effective authorization and authentication properly. This will ensure that the device, the user and the system are fully authorized to transfer information and data in a certain app's workflow and ensure as well that the unauthorized devices, scripts and users are blocked and identified. The developer must instrument some local integrity checks right within their codes in order to detect any kind of unauthorized code changes.

BROKEN CRYPTOGRAPHY

Cryptography goes hand in hand with data security. It is one of the best and oldest understood ways in order to securely transmit sensitive information. Unfortunately, software data encryption isn't so completely understood by most of the mobile application developer available out there. Some of the developers are using the data encryption forms that just don't really succeed.

VULNERABILITY TO BROKEN CRYPTOGRAPHY

Insecure utilization of cryptography is common in most of the mobile apps that use encryption. There are 2 fundamental ways in which the broken cryptography is involved in mobile applications. First, the mobile application may use the behind the decryption / encryption that's fundamentally flawed and can possibly be exploited by adversary in order to decrypt sensitive information/data. Second, the mobile application may leverage or implement the decryption / encryption algorithm that's naturally weak and can directly be decrypted by adversary.

DEPENDENCE UPON THE BUILT-IN ENCRYPTION PROCESS CODE

By default, the iOS applications are completely protected from the reverse engineering through code encryption. The security model of iOS requires that apps to be signed and encrypted by the trustworthy sources in order for them to execute in non-jail break" environments. Upon the start-up, the app loader of iOS will decrypt the app in the memory and will process in order to execute the code after the iOS verification signature. In theory, this feature prevents the attacker from doing binary attacks against the mobile app of iOS.

Using the free tools such as GBD or CluthMod, the adversary will download onto their jailbroken device the encrypted apps and take a decrypted app snapshot once it was already loaded into the memory and decrypts it. After taking the snapshot and being stored on a disk, the adversary will use certain tools like Hopper or IDA Pro in order to easily perform a dynamic / static analysis of app and much further binary attacks.

Bypassing the built-in encryption code algorithms is very trivial at best. It's always needed to assume that the adversary is able to bypass any kind of built-in encryption code that is offered by the involved mobile OS.

POOR KEY MANAGEMENT PROCESSES

It's understood that best algorithms will be useless if keys are mishandled. Many people make mistake of using correct encryption algorithm but utilizing their own protocol in employing it. Some of the problems in poo key management process include:

- Using the keys in similar attacker-readable directory with encrypted content;
- Making the keys easily available to attackers;
- Avoid the utilization of hardcoded keys inside the binary; and
- Keys are intercepted through binary attacks.

CREATION AND UTILIZATION OF CUSTOM ENCRYPTION PROTOCOLS

There's not much easier way to mishandle the encryption than to try creating and using your own developed encryption protocols and algorithms. It's best to always use the modern algorithms that are proven strong by security community and as much as possible, leverage the most advanced encryption APIs in your mobile platform. The binary attacks can lead to adversary, identifying the usual libraries that you're using along with your binary hardcoded keys. In times there are very high security requirements around the encryption, it's best to consider the utilization of whitebox cryptography.

Many of the cryptographic protocols and algorithms shouldn't be used due to their significant weaknesses or insufficiency for the modern requirements in security. These cryptographic protocols and algorithms include the RC2, MD4, MD5, and SHA1.

CLIENT SIDE INJECTION

Client-side attacks or the client side injection might be the very first thing a normal person thinks s of when hearing about the threats in mobile security. In mobile application, client-side injection works in the same way to some server side security risks. The client side injection roots in the mobile software that treats the inputted data inappropriately by user as a code. Essentially, this is the same thing as the cross-site scripting or SQL, however the code is sent as data to client instead of server.

The best way in order to find out if thee mobile app is vulnerable to injection is to determine the input sources and validate that application/user supplied data is subjected to validation of input, preventing the code injection. The checking of code is one accurate and fast way in order to see if the mobile app is correctly handling the data. Tools for code analysis can help a security analyst to find the use of the interpreters and trace the flow of data through the mobile application.

Since the data can originate from numerous sources in the mobile apps, it's very important to list them based on what they're trying to achieve. Generally, the injection attacks on the mobile devices can target the following:

DATA ON DEVICE:

- The SQL Injection – SQLite and many of the phones' default mechanism for data storing can be subjected to in client side injection. The threat of leaving the data visible is very risky when the application accommodates several different users.

- Local File Inclusion – Mobile devices file handling has similar risks as mentioned above except if it concerns to reading the files that might be yours to view within the application directory.

THE MOBILE USERS SESSION:

- JavaScript Injection – JavaScript injection is also possible in mobile browsers. Normally, the mobile browsers have access to the cookies of mobile applications, which can possibly lead to the session theft.

SECURITY DECISIONS VIA UNTRUSTED INPUTS

Generally, the developers use values and fields or any kind of hidden functionality in order to distinguish the low lever users from higher level users. The attackers can intercept calls and temper such sensitive parameters. The weak implementation of hidden functionalities can lead to an improper behaviour of the app and granting the attacker higher permission levels. This can easily be exploited through hooking.

Mobile application can allow data from different kinds of sources. In most of the cases, this will be considered as IPC (Inter Process Communication) mechanism. ICP is the term used to define the activity of sharing the data across several and usually specialized processes through the use of communication protocols. Usually, the applications that are using ICP are being categorized as servers and clients, whereas the data is requested by clients and the server is the one responds to the requests.

IMPROPER SESSION HANDLING

While the session handling is a well-known security concern in web applications, it can be a much bigger problem in mobile applications world. Improper session handling can lead to many vulnerabilities that are common nowadays, despite of the fact that stolen or lost device could have extremely serious consequences. So, what's an improper session handling? As it name suggests, this issue occurs when the session tokens are not handled in their best way. While some might be intentional, proper care should be considered in order to add validation for user.

Due to the different ways the mobile applications are used, many of the developers allow the non-expiring or long user sessions, or use the session tokens that are very predictable. Usually, this happens intentionally because the businesses want the user to have a fast access to check out and purchasing in order for their sales to be made before the user can start having second thoughts. Reducing the constant logging needs in the application can reduce the friction for users.

To facilitate a constant transaction between the mobile application's backend servers and the use, the mobile apps need to use the session tokens in order to maintain a constant state over the stateless protocols such as SOAP or HTTP. In order to maintain the constant state, the mobile application must authenticate first the user from side to side the backend. In response to the successful authentication, servers now issue session cookies to the mobile applications. The mobile application adds the cookies to the entire service transactions in the future between the server and mobile app. This makes the server to enforce the authorization and authentication conveniently for any kind of service requests that is issued by mobile app. The improper session handling can occur when session tokens are unintentionally shared with adversary during the subsequent transactions between the backend servers and the mobile app.

LACK OF BINARY OPTIONS

A lack of binary option in a mobile app can expose the application as well as its owner to a huge variety of business and technical risks if the involved application is unsecured or exposes a sensitive intellectual property. This can leave the mobile app to be reverse-engineered, analyzed, and be modified by the adversary in a rapid manner.

HOW BINARY OPTIONS IS EXPLOITED?

Usually, hackers use an automated tool in order to reverse-engineer the code and to modify it by using a malware and perform hidden functionalities. It's quite hard to detect if the app's code has been reverse engineered. The app owner usually knows about this when similar code shows up in Google Play, iTunes, or similar third party application store.

VULNERABILITY TO INSUFFICIENT BINARY OPTION PROTECTION

If you're hosting a code in an environment that's untrustworthy, then you're susceptible to this kind of risk. The untrustworthy environment includes the mobile clients, could spaces, firmware in appliances, or datacenters in particular countries.

PREVENTING THE LACK OF BINARY OPTIONS

First and foremost, the application must act in accordance with the secure coding techniques for:

- Debugger detection controls;
- Certificate pinning controls;
- Checksum controls;
- Jailbreak detection controls.

Next, the application must mitigate adequately with 2 different technical risks that the above mentioned controls are being exposed to:

1. The organization that builds the mobile app must effectively prevent the adversary from reverse engineering and analyzing the app through the use of dynamic or static analysis techniques;
2. The app must detect at a runtime that the code has been changed or added from its integrity at a compile time. The app must appropriately react at runtime to a integrity violation code.

BEST PRACTICES FOR SECURE MOBILE APPLICATION DEVELOPMENT

Before one hire a mobile development team and start the development process, it is essential that you do not overlook your future application's security. Exposure of confidential information, blackmail, and theft are only several of the unfortunate consequences that a poorly implemented and designed security system of mobile application can lead to. If the application being developed relies on the confidential data, it is recommended to use best practices on the development process.

1. Evaluation of risks – Thoughtful risk evaluation is the first step, which can help you in deciding if one needs to accept residual risks or take an active role in minimizing them. The 3 major factors are knowing the losses, the risks, and the application's vulnerabilities.
2. Continuity of implementing security – In terms of mobile app security, there is no magic spell from what a developer will be able to cast in order to make the application protected from hacks as soon as the development phase is over. Security is a process, which is why it needs to start during the planning phase, goes through the implementation stage and code reviewing, as well as having penetration tests performed before releasing.
3. Validation of input – While PCs have lots of antivirus software in isolating and detecting malicious components, in general, mobile devise do not have any. Make sure that the developers will implement appropriate input checks, and validates that the input is the expected thing, no more and no less, before the application would start to process it.
4. Principle of least privilege – Designing an app requires only those permissions that are definitely necessary for the purpose of the mobile app. It must not make the users wonder about why the compass application need access to the SMS messages. Attentive users are more likely to prevent applications requesting permissions, which exceed the functions of application.

5. Strong passwords – Other than having the chance of hacker's access reduced to the data of the user, passwords can affect the strength of encryption. The encryption's strength rests on the strength of the key, whereas the key itself is most commonly protected by the algorithms employing the input date of the users.
6. Secure authentication – In the authentication system design, you should pay certain attention to the two peculiarities of mobile devices, which are the inconvenient way of typing texts and episodic sessions. Taking this into account, an appropriate investigation of typing a challenging succession of symbols is able to reduce user satisfaction, as well as their loyalty to the company.
7. Protection of data – Select the data that are actually needed to store. The less you store, the less you would need to protect. For the storage of data as a necessity for the project, it needs to be insisted to be encrypted and the encryption key must be entered each and every time by the users, or at least not be stored on the device.
8. Password strength checkers – Users have a tendency of selecting weak but fast and convenient to type passwords, instead of tolerating the turmoil of entering secure passwords every time in order to unlock their device. Thus, if the security is among the major priorities for projects, users must not be trusted and consider having a password strength checker implemented.
9. Data encryption – The word "encryption" has a tendency of being used as a synonym to secure solutions, but the amount of deciphered and stolen credit card numbers are keep growing each year. Therefore, an important question to be taken into consideration is not about which encryption strategy must be used but instead, about how to get it implemented properly.
10. Remote wipe – A wipe is an effective and useful data protection or strategy but not a panacea. It may usually be unarmed by just turning off the internet connection or device.
11. Protection of encryption keys – In order to protect encryption keys, it is not enough to rely on the standard platform measures. The first things that you need to discuss with the developers is the possibility to store the key outside the device.

12. HTTPS – There is always interception of information risk when you transfer data through the internet, like for instance, through some sniffer device. Rather than an ordinary HTTP connection, using SSL security protocol will help in reducing the risk, for it is implying that data will be encrypted.

13. The risks of using cloud services – If cloud services will be chosen as an outside storage, you should not forget that they need to be used with great care. Although cloud services have many benefits, cloud is not always the best solution when it comes to security. Even if you are working with the leading service providers, their services are not always complying with high security demands. If security is one of your top priorities, it is strongly recommended to use own servers for company.

14. Use logging with caution – Before releasing the mobile application, it must be ensured that there is no confidential information in the logs. The greatest solution would be to enable separate logs to release and debug versions.

15. Secure Bluetooth connection – Bluetooth is providing security services only in device level and not user level. It is not able to limit the access of sensitive data to the authorized users. Thus, developers must provide proper security controls that are offering security features in the level of identity, such as user authorization and user authentication.

16. Testing – It is a mandatory and usual procedure, and hardly any mobile application can be considered done without appropriate testing. Nevertheless, if your application involves a high degree of risk, such as enabling customers to trade stocks that they own, penetration testing must be added to the list. As also known to be ethical hacking, during this testing, the testers will act as attackers, attempting to compromise the mobile application.

It is only a brisk overview of several issues. The scope of security issues is really broad. Other than covering basics about secure development, the best practices are focusing on the essentials of secure authentication, as well as data storage and data transferring. These practices are important to pay strict attention to, not only because failure in complying could mean huge fines if you are audited, but due to the guidance and information that they are offering in terms of managing secure mobile application.

Security professionals devised important best practices for everything, from how the data is processed, to entering the card date to the system, through secure payment applications. When setting these guidelines in places, one is not only ensuring to adhere to best practices, but also boosting business reputation.

GUIDES DURING DEVELOPMENT PROCESS

This guide will give particular recommendations and suggestions to be used during the process of secure mobile application development. The security recommendations and attack descriptions in this report are not perfect or exhaustive, but you will be getting practical advice that you will be able to use in making your applications more secure.

- Coding practices
    1. Test the 3rd party libraries
    2. Use obfuscation and increase the code complexity
    3. Avoid using simple logic
    4. Understand secure data deletion
    5. Store sensitive data in RAM securely
    6. Implement anti-tamper strategies
    7. Avoid having query string for sensitive data
- Practices for handling sensitive data
    1. Use secure setting for cookies
    2. Implement secure storage of data
    3. Protect against SSL Strip
    4. Fully validate TLS/SSL
    5. Carefully treat Geolocation data
    6. Limit the use of UUID
    7. Implement two-factor or enhanced authentication
    8. Implement secure network transmission of the sensitive data
    9. Institute local session timeout
    10. Protect application settings
    11. Validate input from the client

12. Protect application settings

- Logging and caching
    1. Avoid crash logs
    2. Avoid caching application data
    3. Manage debug logs carefully
    4. Limit username caching
    5. Be aware of copy and paste
    6. Be aware of the keyword cache
- iOS
    1. Avoid snapshots of cached application
    2. Carefully use keychain
    3. Use ARC or the Automatic Reference Counting
- Webviews
    1. Protect from CSRF with the form tokens
    2. Prevent clickjacking and framing
- Servers
    1. Configure server-side SSL appropriately
    2. Protect internal resources
    3. Protect and pen test web services
    4. Use appropriate session management
    5. Implement appropriate configuration of web server
- Android
    1. Check activities
    2. Implement intents strictly
    3. Implement permissions of files carefully

4. Implement pending intents with caution
5. Implement content providers cautiously
6. Sign android APKs
7. Protect the app services
8. Use broadcasts strictly
9. Follow the best practices
10. Avoid caching GUI objects
11. Avoid Cached camera images storing
12. Avoid intent sniffing

CONCLUSION:

Nowadays, the plight of mobile application developers is a challenging one. On the other hand, developing in this area is full of opportunities and vibrant. From smartphones to tablets, a large spectrum of new devices has been redrawing the boundaries of what users are able to do. This new landscape is also bringing new development factors, which include how to create simple and effective applications, the type of devices to target, and about how to secure the downloaded and uploaded data. Particularly, the trend of IT consuming heavily weighs on the developers of secure mobile application. This trend encompasses various facets. Corporate users are increasingly accessing enterprise data from mobile devices, which could be their own and could be deployed by the internal IT department. This means that developers may not know what is the target platform, and requires either a multi-platform or a cross-platform development effort.

Mobile industry is strongly whirling and volatile. As soon as these mobile applications have been completed and entered in the mobile market, these applications run with unique functionality on the mobile phones. End users need to be satisfied. Once these users become satisfied, you win the battle. Web applications have been focused on the tools and experiences, which are the highly successful applications in the mobile market. The mobile applications are open, so when the customers are spreading and using the word counted as the most reliable one.

There may be lots of tasks in the management, which should be repetitively and frequently done. It is strongly time consuming for the individual involved. These applications can help you automate such tasks, and free up your valuable time for your business and personal reasons. You probably have a plethora of ideas and possibilities that may lead towards a useful and secure mobile application development. The motto of all businesses is getting maximum profit. Lots of web applications that have gone to mobile proved that the social aspect of businesses can be translated easily into mobile marketplace. These marketplaces foster viral campaigns that have a tendency of spreading from user to user with a minimal amount of additional marketing involved.

Other than these, mobile app development has also been improved in security and piracy. It is the most rapidly growing business in the industry these days. Everyone in the world has mobile phone. These little devices are very comfortable to use and is customizing the features and settings. Lots of companies have developed software that provides feature to make their own programs to share with friends. There are templates, which are readily available on quizzes, gifts, and games. The companies are most commonly charging a portion of the revenue. Mobile application makes consumers well aware of the market in comparing the prices. They connect to the net, browsing through famous shopping websites. If they find a better price, they will be able to order the item on the internet or go to the other store. Mobile application development has become an important part of your life. Therefore, using guidelines has also become a necessity in creating effective and secure mobile applications.