Concern over application vulnerabilities is growing, climbing to become the number one security concern in a recent Global Information Security Workforce study. About 70 per cent of the information security professionals participating in the survey identified software as their top security concern. One of the areas driving this concern is the proliferation of new mobile applications, with cloud–based services and consumer–owned devices (BYOD) infiltrating business systems.

Paco Hope, a member of the (ISC)² Application Security Advisory Board, considers five principles of secure mobile development.

In such a rapidly evolving ecosystem, it is challenging to ensure that security is prioritised, particularly given the development community’s focus on innovation rather than control. Further, we must work with unfamiliar tools and technologies as they continue to change at a rapid rate. Fortunately, many well–known security principles are readily adapted to mobile development, making our job a bit easier. Once we separate the new mobile–specific security issues from familiar security issues, we are better prepared to secure our mobile apps. The five principles here are a mixture of good news and bad news.

1. We already know a lot of useful principles
Much of the mobile world is underpinned by familiar web–based technologies. It often uses web technologies like AJAX, XML and JSON. Our servers are often running familiar operating systems, middleware and application code. The very latest mobile application running on the very latest mobile hardware is probably talking to a very familiar and well–understood server layer that runs in a well–understood data centre.

Concepts like session management, authentication, entitlement, logging and input validation are already well understood. If there is a truly different mobile slant, it is that the mobile platform is not a trusted platform and mobile–specific inputs like accelerometers and GPS location and near-field communication (NFC) data can be malicious or spoofed.

The last two decades of software security have produced standards, texts, principles and certifications that help us apply security throughout the software development lifecycle. While we may be adapting to the mobile environment, we are not starting from scratch.

2. We have a good threat model for mobile
When we think about the threats our mobile applications face, we can start with a well–known model, and then augment it to cover truly new issues that relate to mobile. Our starting point for modelling a mobile device can be a desktop PC infested with viruses and malware, where there is no antivirus and there is nothing we can do about that. While this sounds extreme and pessimistic, it’s not far off the mark.

Ultimately, our apps must be ‘trusted on busted.’ The application must work even if every security control on the mobile device has been subverted. There is no way to know with certainty whether or which
security controls have been compromised. It’s hard to classify malware even. From the perspective of our apps, malware can be truly malicious software that is unknown to the device’s owner or it can be intentionally installed by the user and used against us (for example, debuggers and such for reverse engineering). No malicious software or operating system subversion can be reliably detected and prevented all the time.

There are a few notable additions to the threat model that are related to the mobile platform. Data we put on the device (for example, a cache for better performance or offline use) is at significant risk of disclosure if the device is stolen. Most mobile devices have a few different ways in which their contents can be dumped if they are stolen.

A high proportion of mobile devices have connectivity that non-mobile devices do not, like LTE, GSM and CDMA. They create new threat vectors like rogue femtocells that masquerade as legitimate mobile towers. A GSM SIM card, for example, also has a small amount of computing capability, encryption keys, some unique identifiers, and some relatively permanent storage. Our threat model must take into account the malicious communications channels, malicious inputs from SIMs and radios. We also have to consider the impact to our application from the modification of information assets related to communications, like carrier settings.

3. Privacy is complex in the mobile world

The surge in adoption of mobile devices has thrown a spotlight on two key privacy concerns in the mobile world: correlation to the real world and the mix of personal and private data.

Mobile phones and tablets are tightly correlated to their owners’ location and they record their location frequently—potentially allowing us to trace the owner’s movements yesterday if we steal the device today. Metadata about the owner’s communications (email, SMS, voice) are there and can correlate to a very specific time and geography. The laws related to protecting data are both national and supranational (for example, European, UN and so on). This means that software that complies with the laws of one nationality (for example, the United States) might be operating illegally when operating in another country (for example, Switzerland). Personal and private data mix on mobile devices like never before. Personal contacts, like the owner’s spouse, will be listed in the same contact list as professional contacts, like the owner’s boss. Business data can be trivially emailed, sent via SMS or uploaded to sharing sites. Our mobile devices become one of the weakest links in the defences of personal and professional data.

When it comes to privacy, there are three really important principles. First, know what you collect, where it is and how long you have it. Next, avoid collecting or using any data that isn’t absolutely necessary. Data that does not exist cannot create privacy risk. Finally, discard or anonymise data as soon as possible and store it in the confines of the phone as little as possible. It is often easier to shrink the amount of time you hold something sensitive than to completely remove the need to hold it.

Privacy is another area where the general security principles have been well known and understood for some time. They have even been formalised into parts of training and certifications. The situations we encounter in mobile are changing, but underlying principles are still available to guide us.

4. Our intellectual property is exposed

Mobile applications cannot realistically be hidden from prying eyes. We have to accept that our applications can be downloaded from an application store, run in a simulator or a modified phone, and reverse-engineered. This is a different slant on the ‘trusted on busted’ principle. We must, to the extent that we can, keep any intellectual property or proprietary technology in our server-side systems. We must assume the actions of our applications will be revealed to competitors, hackers and fraudsters, who want to misuse and abuse them.

The server side of our application must account for the uncertainty of whether the application is really running unmolested on unmodified hardware. We must always accept the possibility that someone has reverse-engineered the application to find bugs we don’t yet know we have. Despite an increasing array of products and tools to obfuscate, harden and protect the integrity of applications, none can ever be perfectly complete. We must identify the residual level of risk and design our server side applications to address it.

5. We must chart totally new territory

Mobile devices are replete with sensors—accelerometers, compass, GPS, near field communication (NFC), magnetic closures, touch screens and so on. Whereas some are secured through the operating system (for example, the touch screen and GPS), others are not (accelerometers). We must continually evaluate whether the data that is present in the device is sensitive or not and whether or not the operating system’s efforts to secure it are sufficient. The same non-secured accelerometers that allow your phone to be a makeshift pedometer also recently allowed researchers to measure motion and infer the unlock PIN for some phones.

What aspects of the device should be considered private and which ones should not (and indeed how they should be protected) is still a work in progress on the various platforms. Every physical device has unique identifiers like IMEI, serial numbers and MAC addresses. Those unique identifiers create privacy concerns if they can be correlated with real-world identity data such as that provided when shopping online. It is tempting to leverage these identifiers in various ways, but doing so can make us subject to privacy laws. Applications can also create new data that can combine in unexpected ways with existing data.

Conclusion

We must remember ‘trusted on busted.’ The device might be compromised, there might be active malware attacking our application, and our application’s code might be completely reverse-engineered. We must design with a completely untrusted client side in mind and leverage all the things we already know about building a secure server side. We should be very cautious about collecting, sharing and using unique identifiers or data about the owner. Given these mobile-specific principles, we can apply many security development techniques directly to building our mobile applications, and we will be better off for doing so. Much of our security education and principles from the last 20–30 years are still valid to serve as a foundation. The techniques we apply in our software development lifecycles still work, but we have to adapt them to the mobile ecosystem.

About the author

Paco Hope (paco@cigital.com) is a principal consultant with Cigital Ltd, a member of the (ISC)² Application Security Advisory Board.