iOS Security

iOS provides layers of protection to ensure that apps are signed, verified, and sandboxed (a sandbox is a security mechanism for separating programs) to protect user data, thus offering protection from malware and other unwanted third party attacks. The iOS kernel controls which user processes and apps can be run. iOS requires that all executable code be signed using an Apple-issued certificate. This certificate ensues that all apps come from a known and approved source. Third-party apps must also be validated and signed using an Apple-issued certificate, thus preventing third-party apps from loading unsigned code resources or using self-modifying code.

To protect the system and other apps from loading third-party code inside of their address space, the system will perform a code signature validation of all the dynamic libraries that a process links against at launch time. This verification is accomplished through a 10-character alphanumeric string (i.e. 1A2B3C4D5F) called the team identifier, which is extracted from an Apple-issued certificate. A program may link against any platform library that ships with the system or any library with the same team identifier in its code signature as the main executable. Since the executables shipping as part of the system don't have a team identifier, they can only link against libraries that ship with the system itself.

iOS does not allow users to install potentially malicious unsigned apps from websites, or run untrusted code. At runtime, code signature checks of all executable memory pages are made as they are loaded to ensure that an app has not been modified since it was installed or last updated.

Even after an app is verified and approved, iOS enforces security measures designed to prevent it from compromising other apps or the rest of the system by "sandboxing" all third-party apps. This ensures that third-party apps are restricted from accessing files stored by other apps or from making changes to the device, thus preventing one app from obtaining or changing information stored by another app. Each app has a randomly assigned and unique home directory for its files given to it when the app is installed. A third-party app can only obtain outside information by using services explicitly provided by iOS[i].

XCodeGhost

Since Apple makes it difficult for hackers to infect iOS apps with malware, this has generally not been a problem. In September 2015, hackers were able to get malware past Apple's app store review by modifying some code in the XCode toolset and passing it off as Apple's official version of XCode. In order to prevent this type of attack, developers should make sure that they download XCode directly from Apple[ii].

[i] iOS Security Guide, September 2015, page 18-20

[ii] http://www.zdnet.com/article/how-malware-finally-infected-apple-ios-apps-xcodeghost/