Case Study: iOS Security

Konstantin Beznosov
Apple designed the iOS platform with security at its core. When we set out to create the best possible mobile platform, we drew from decades of experience to build an entirely new architecture. We thought about the security hazards of the desktop environment, and established a new approach to security in the design of iOS. We developed and incorporated innovative features that tighten mobile security and protect the entire system by default. As a result, iOS is a major leap forward in security for mobile devices.

Every iOS device combines software, hardware, and services designed to work together for maximum security and a transparent user experience. iOS protects not only the device and its data at rest, but the entire ecosystem, including everything users do locally, on networks, and with key Internet services.

iOS and iOS devices provide advanced security features, and yet they're also easy to use. Many of these features are enabled by default, so IT departments don't need to perform extensive configurations. And key security features like device encryption are not configurable, so users can't disable them by mistake. Other features, such as Touch ID, enhance the user experience by making it simpler and more intuitive to secure the device.

This document provides details about how security technology and features are implemented within the iOS platform. It will also help organizations combine iOS platform security technology and features with their own policies and procedures to meet their specific security needs.

This document is organized into the following topic areas:

- **System security**: The integrated and secure software and hardware that are the platform for iPhone, iPad, and iPod touch.
- **Encryption and data protection**: The architecture and design that protect user data if the device is lost or stolen, or if an unauthorized person attempts to use or modify it.
- **App security**: The systems that enable apps to run securely and without compromising platform integrity.
- **Network security**: Industry-standard networking protocols that provide secure authentication and encryption of data in transmission.
- **Internet services**: Apple's network-based infrastructure for messaging, syncing, and backup.
- **Device controls**: Methods that prevent unauthorized use of the device and enable it to be remotely wiped if lost or stolen.
- **Privacy controls**: Capabilities of iOS that can be used to control access to Location Services and user data.

source: “iOS Security” Apple, September 2014
secure boot chain

1. processor executes **Boot ROM**
   - immutable
   - contains Apple Root CA public key
   - hardware root of trust — implicitly trusted
2. Boot ROM verifies that **Lowe-Level Bootloader (LLB)** is signed by Apple
3. LLB verifies signature of and runs **iBoot**
4. iBoot verifies signature of and runs **iOS kernel**
   - on devices with cellular access
     - **baseband subsystem** boots similarly
   - on devices with A7 or later processor
   - **Secure Enclave co-processor** goes through similar boot process
system software authorization

iOS device (via iTunes)

Apple installation authorization server

{data | nonce | ECID}

- **data**: cryptographic images of the system software to be installed (e.g., LLB, iBoot, kernel, OS image)
- **ECID**: device’s unique ID

1. checks if the version is permitted
2. adds the ECID to the measurement
3. signs the result

[result]_server