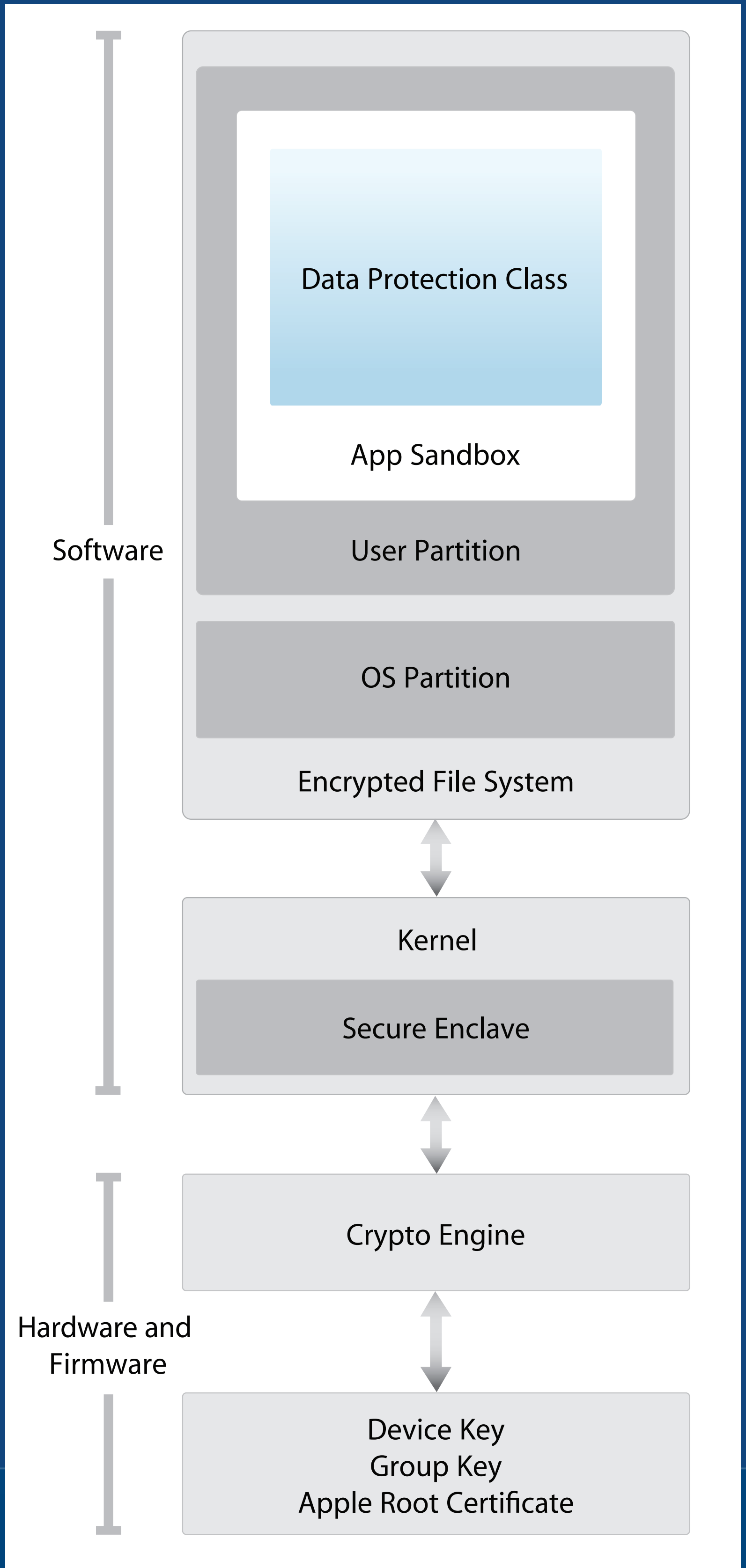
Case Study: iOS Security

Konstantin Beznosov

overall stack



source: “iOS Security” Apple, September 2014

secure boot chain

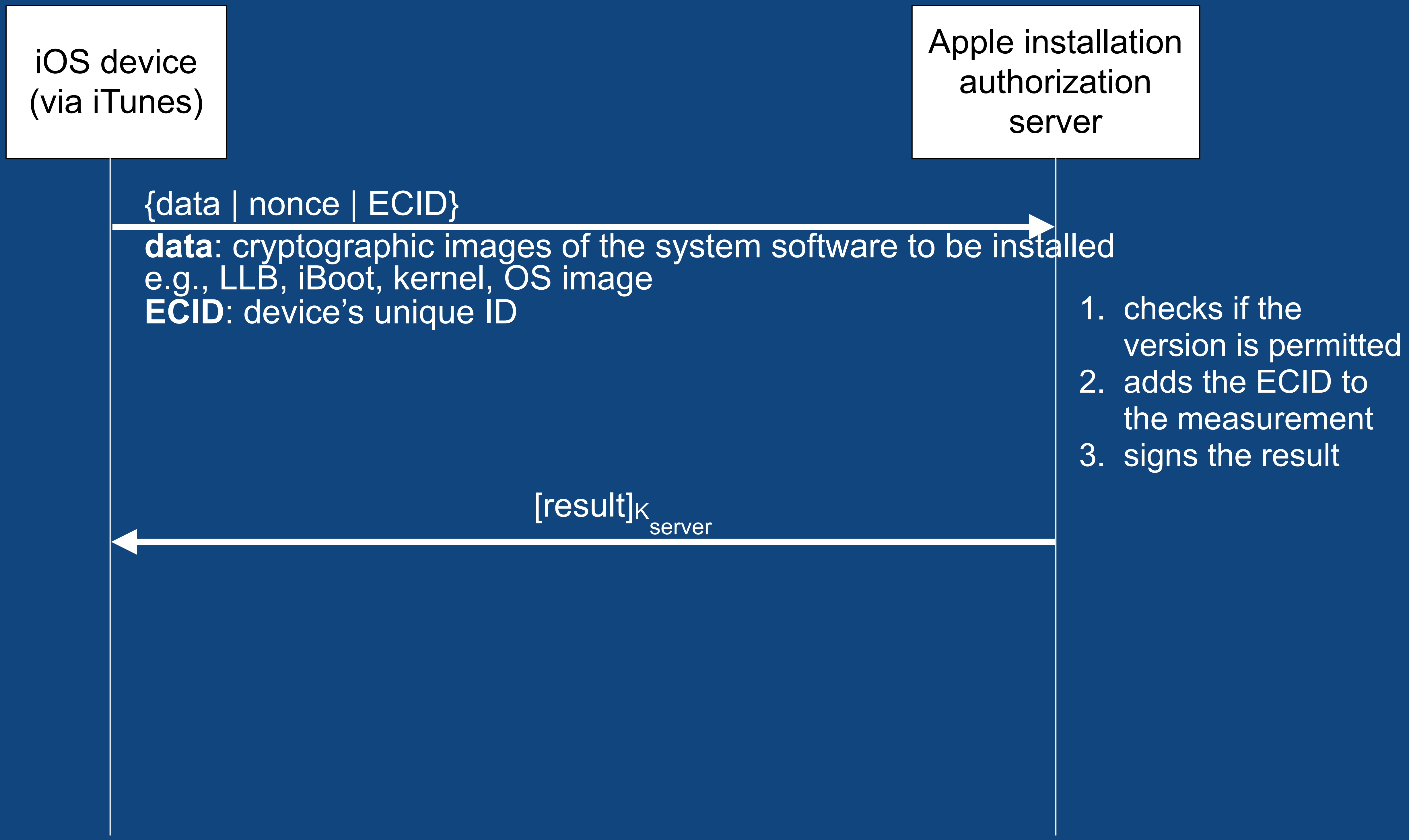
1. processor executes **Boot ROM**
   - immutable
   - contains Apple Root CA public key
   - hardware root of trust — implicitly trusted
2. Boot ROM verifies that **Lowe-Level Bootloader (LLB)** is signed by Apple
3. LLB verifies signature of and runs **iBoot**
4. iBoot verifies signature of and runs **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

$[\text{result}]_{K_{\text{server}}}$