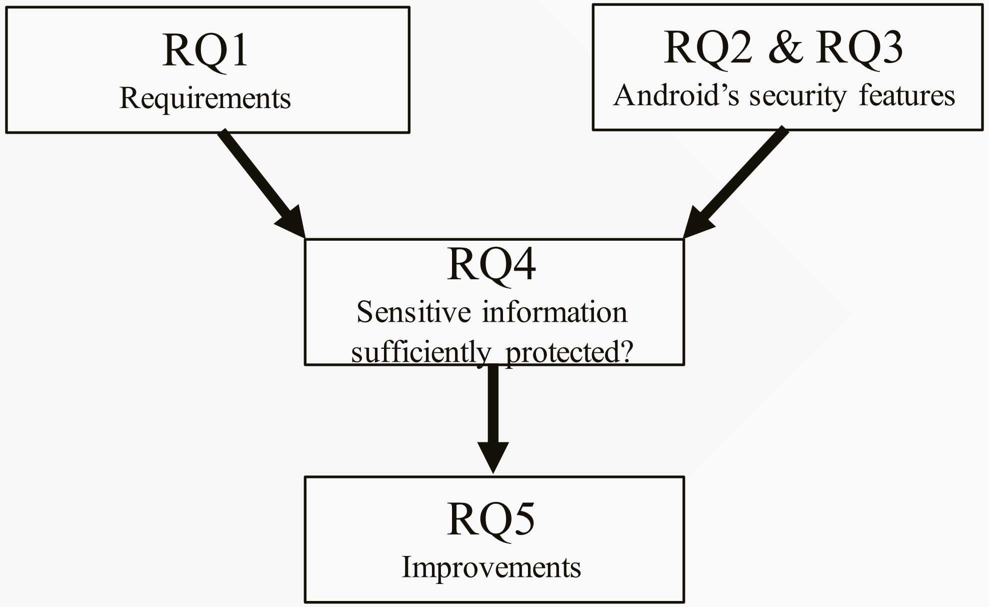
Using Sensitive Information on Android Based Smartphone

Romke van Dijk

UNIVERSITY OF AMSTERDAM

Android 6:
To what extent is sensitive information protected?

Methodology


RQ1
Requirements
RQ2 & RQ3
Android’s security features
RQ4
Sensitive information sufficiently protected?
RQ5
Improvements

Related work

- Guidelines generic (NIST)
- Platform specific guidelines (CESG)
- Android project

Contribution

- Why?
- How?
- (Individual researcher)

“Sensitive information refers to the majority of information processed (or created) by large enterprises or public services that are used in routine business operations and services and could have damaging consequences if lost, stolen or published in the media”

Source: Government Security Classifications by CESG (2011)

Protect against attackers with bounded capabilities and resources.

investigative journalist

competent individual hacker

the majority of criminal

Attack landscape

Stolen Device

Malicious apps

Exploits

Eavesdropping

Source: Cyber Threats to Mobile Phones by US-Cert

Data protection

- Data at-rest
- Data in-transit
- Authentication

Platform integrity

- Application segregation
- Secure boot sequence
- Malicious code execution (detection and prevention)
- Update policy

Based on:

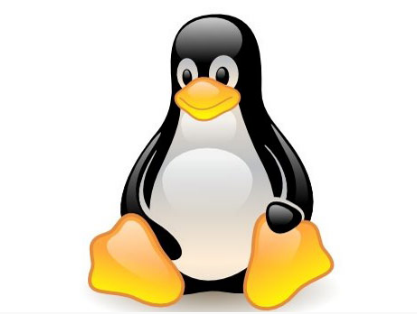
“End user device strategy: security framework and controls” by CESG (2013)
“Guidelines on cell phone and PDA security” by NIST (2011)

Data protection

- Data at-rest
- Data in-transit
- Authentication

Platform integrity

- Application segregation
- Secure boot sequence
- Malicious code execution (detection and prevention)
- Update policy



Based on:

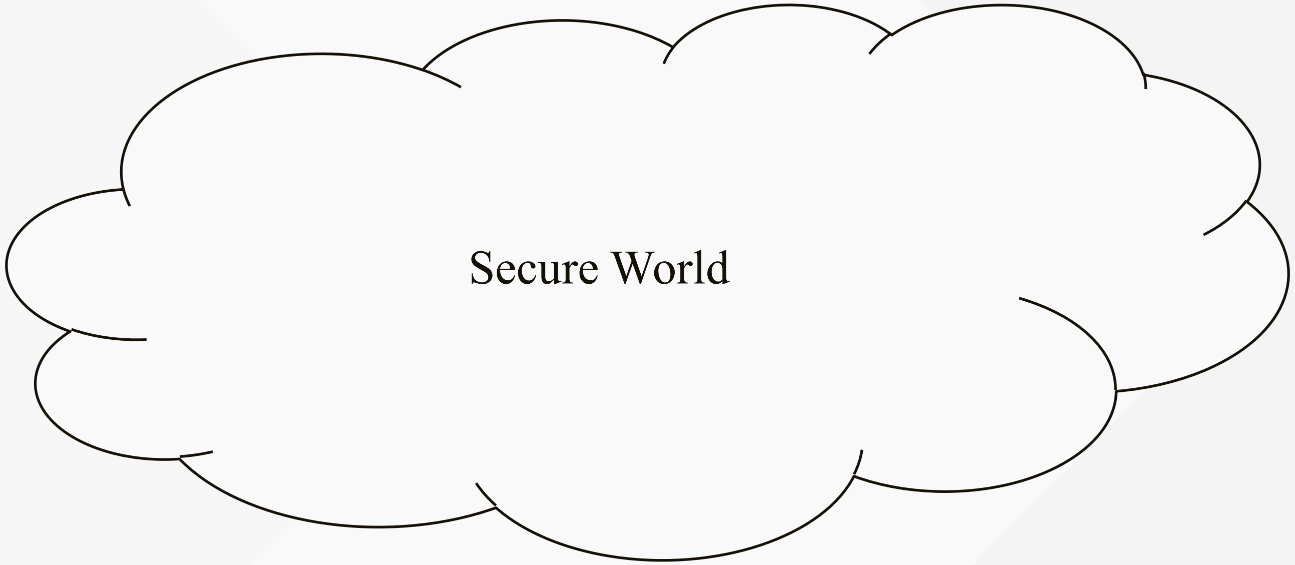
"End user device strategy: security framework and controls" by CESG (2013)
"Guidelines on cell phone and PDA security" by NIST (2011)

To what extent is sensitive information protected on an Android 6 based smartphone?

It depends…

Stolen device

- Trusted Execution Environment (TEE) must be implemented
- Strong authentication
- Up-to-date
- Locked bootloader
- Mobile Device Management (MDM)



Secure World

Data protection

- Data at-rest
- Data in-transit
- Authentication

Platform integrity

- Application segregation
- Secure boot sequence
- Malicious code execution (detection and prevention)
- Update policy

Based on:

"End user device strategy: security framework and controls" by CESG (2013)
"Guidelines on cell phone and PDA security" by NIST (2011)

“Encryption keys protecting sensitive data remain in device memory when the device is locked.”

Source: End User Devices Security Guidance: Android 6 by CESG (2016)

Stolen device

Up-to-date

CVE-2015-3860

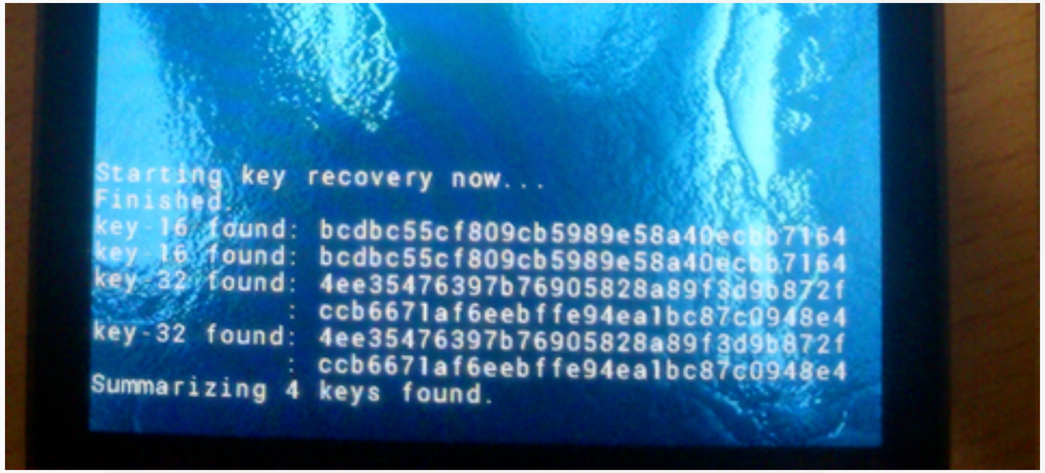
“Android 5 <= 5.1.1 does not restrict the number of characters in the passwordEntry input field, which allows physically proximate attackers to bypass intended access restrictions via a long password that triggers a SystemUI crash“

Source: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-3860

Stolen device

Locked bootloader

Muller et al. (2013) “FROST: Forensic Recovery Of Scrambled Telephones”



```
Starting key recovery now...
Finished.
key-16 found: bcdbc55cf809cb5989e58a40ecbb7164
key-16 found: bcdbc55cf809cb5989e58a40ecbb7164
key-32 found: 4ee35476397b76905828a89f3d9b872f
            : ccb6671af6eebffe94ea1bc87c0948e4
key-32 found: 4ee35476397b76905828a89f3d9b872f
            : ccb6671af6eebffe94ea1bc87c0948e4
Summarizing 4 keys found.
```

Authentication

- PIN
- Pattern
- Password
- Fingerprint

Max entropy $10^4 = 10000$

“The lock screen authentication MUST rate limit attempts and SHOULD have an exponential backoff algorithm as implemented in the Android Open Source Project.”

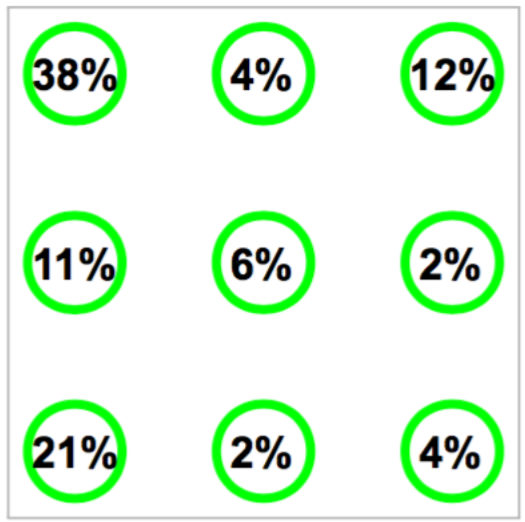
Source: http://source.android.com/compatibility/android-cdd.html

Solution: MDM, Wipe data after maximum failed attempts

Authentication

- PIN
- Pattern
- Password
- Fingerprint

What is stronger 4-digit random PINs or the practical entropy of patterns?



Entropy practically $2^{10.90} \approx 1910,85$

Source: “Quantifying the security of graphical passwords: The case of android unlock patterns” by Sebastian Uellenbeck et al.

Authentication

- PIN
- Pattern
- Password
- Fingerprint

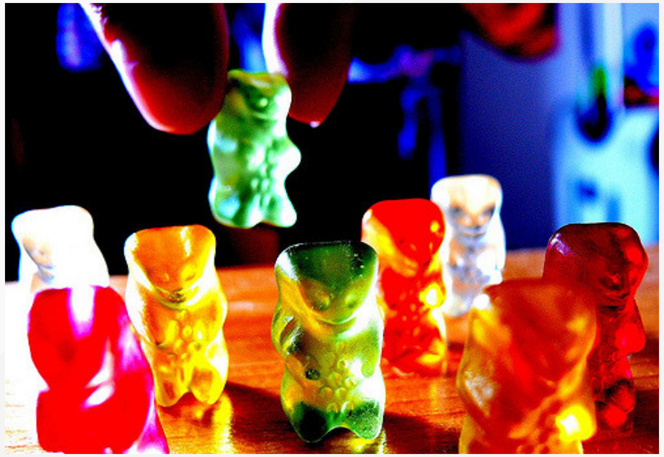
Enter complex password???

Authentication

- PIN
- Pattern
- Password
- Fingerprint

Use of lock screen authentication increased from 50% to 90% on Google Nexus devices.

Source: Google I/O 2016 Security Update



Artificial gummy fingers

Authentication

- PIN
- Pattern
- Password
- Fingerprint

What is stronger: fingerprint or 5 Digit PIN?

“MUST have a false acceptance rate not higher than 0.002%.”

Source: http://source.android.com/compatibility/android-cdd.html

$$k_b = \frac{1}{FMR} = \frac{1}{0{,}00002} = 50000$$

k_b = effective keyspace of biometric authentication

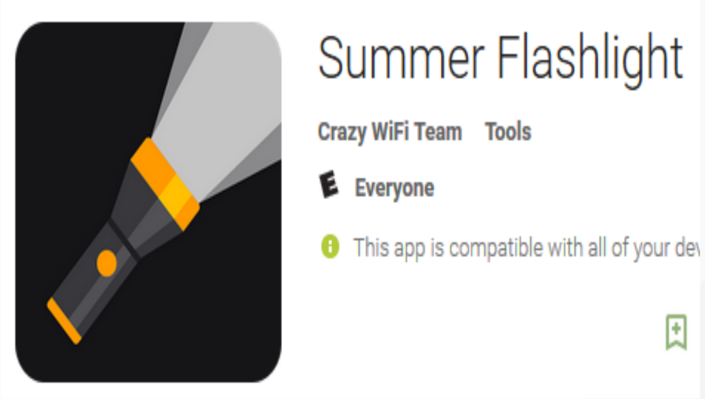
$10^5 = 100000$

Malicious Application

- Trusted Applications (White-listing)
- Up-to-date

ANDROIDOS_GODLESS.HRX aka Godless

- Targets Android <= 5.1



Summer Flashlight

Crazy WiFi Team Tools

Everyone

This app is compatible with all of your dev

Source: Trendmicro(2016) "'GODLESS' Mobile Malware Uses Multiple Exploits to Root Devices"

Android Security Issues

"LG will be providing security updates on a monthly basis which carriers will then be able to make available to customers immediately."

"Samsung Electronics will implement a new Android security update process that fast tracks the security patches over the air when security vulnerabilities are uncovered. These security updates will take place regularly about once per month."

Source: https://www.wired.com/2015/08/google-samsung-lg-roll-regular-android-security-updates/

Data protection

- Data at-rest
- Data in-transit
- Authentication

Platform integrity

- Application segregation
- Secure boot sequence
- Malicious code execution (detection and prevention)
- Update policy

Based on:

"End user device strategy: security framework and controls" by CESG (2013)
"Guidelines on cell phone and PDA security" by NIST (2011)

Exploit

- Locked bootloader
- Up-to-date

Eavesdropping

- Use a the native VPN in Always-On mode
- Educate users to not disable this

Data protection

- Data at-rest
- Data in-transit
- Authentication

Platform integrity

- Application segregation
- Secure boot sequence
- Malicious code execution (detection and prevention)
- Update policy

Based on:

"End user device strategy: security framework and controls" by CESG (2013)
"Guidelines on cell phone and PDA security" by NIST (2011)

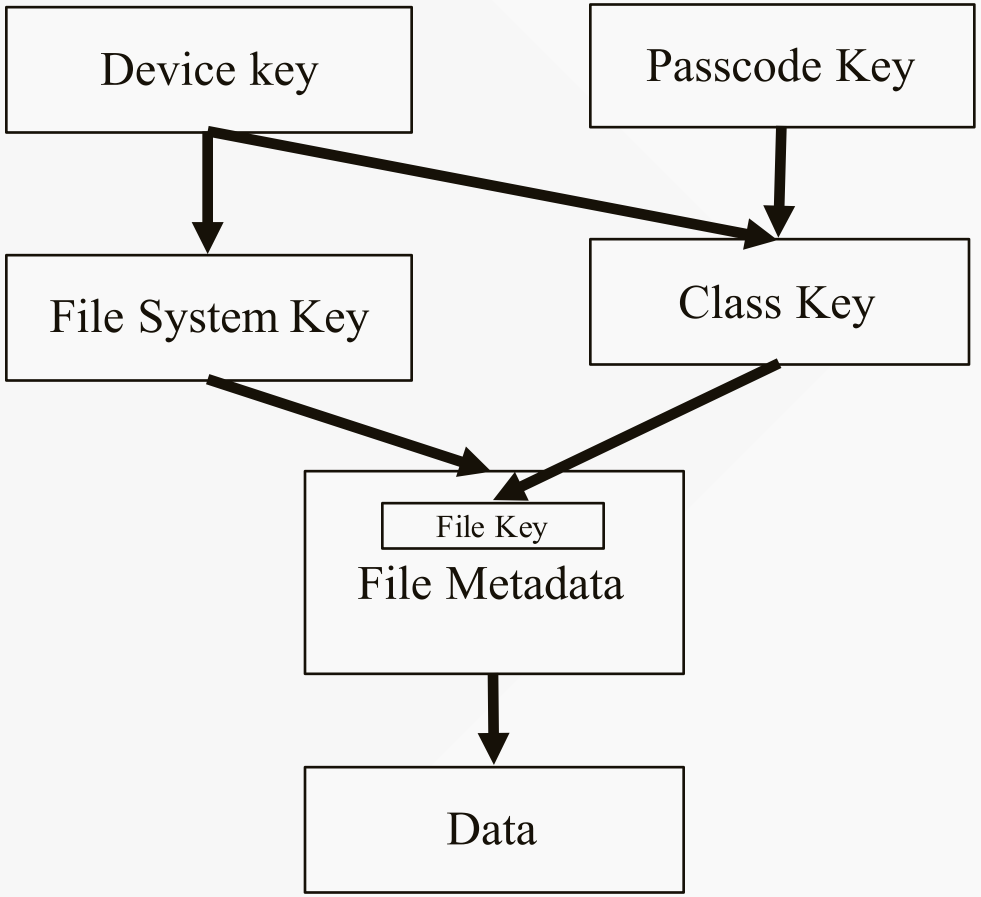
Conclusion

- TEE must be implemented
- Strong authentication
- Up-to-date
- Locked bootloader
- MDM
- Use a the native VPN in Always-On mode
- Trusted Applications (White-listing)

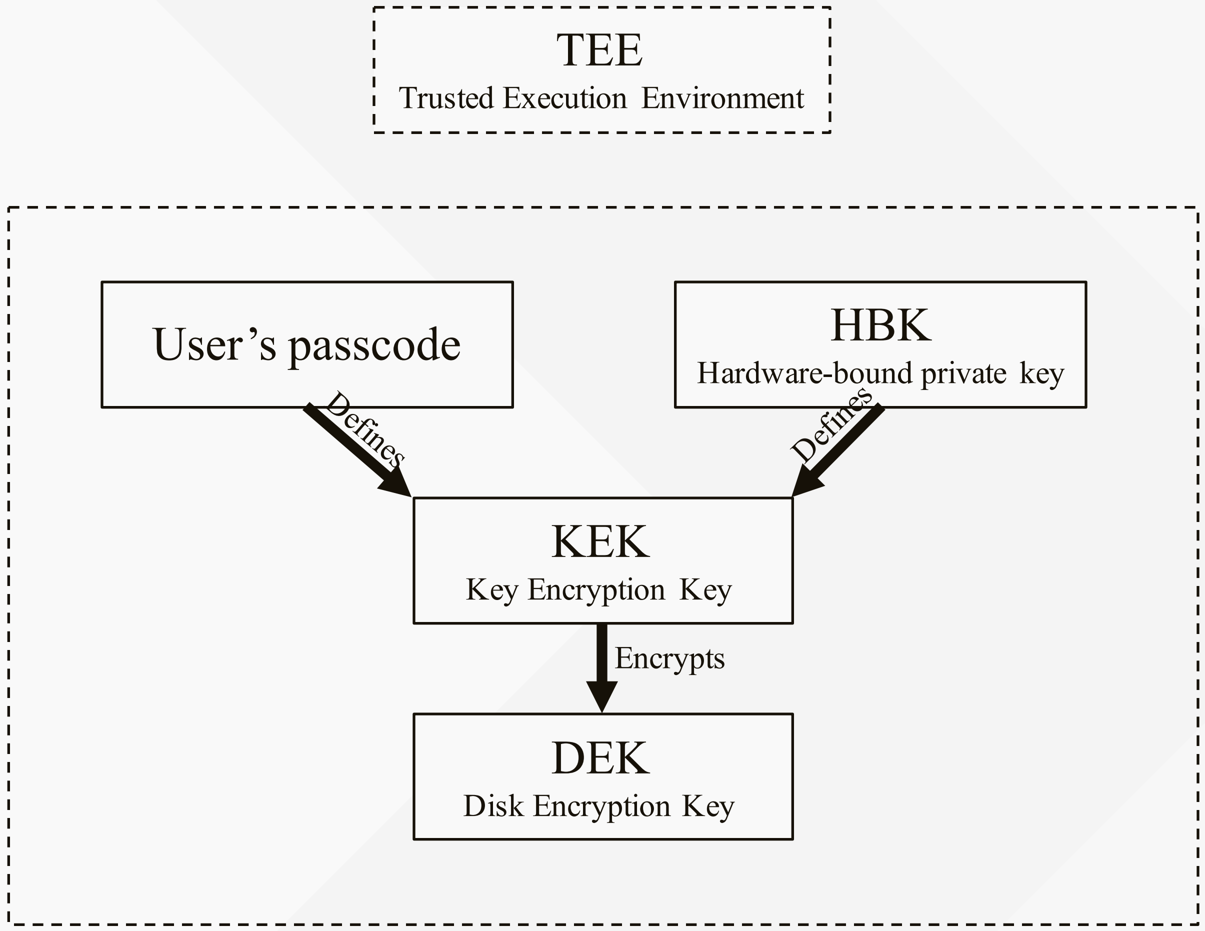
Sources

- Image slide 2: www.perspecsys.com
- Lego: hacker (https://www.flickr.com/photos/99717434@N04/), criminal (https://www.flickr.com/photos/sunface13/), cameraman (https://www.flickr.com/photos/gordon_mckinlay/)
- Pickpocket sign: https://www.flickr.com/photos/doctorow/ Bluescreen: https://www.flickr.com/photos/fsse-info/ App: https://www.flickr.com/photos/osde-info/ Eavesdropper: https://www.flickr.com/photos/smoovey/
- Yummy bears: https://www.flickr.com/photos/pocait/
- Linux: https://www.flickr.com/photos/doctorserone/, Selinux: https://www.flickr.com/photos/xmodulo/
- Android Malware: https://www.flickr.com/photos/cyberhades/, Stagefright: https://en.wikipedia.org/wiki/Stagefright_(bug)

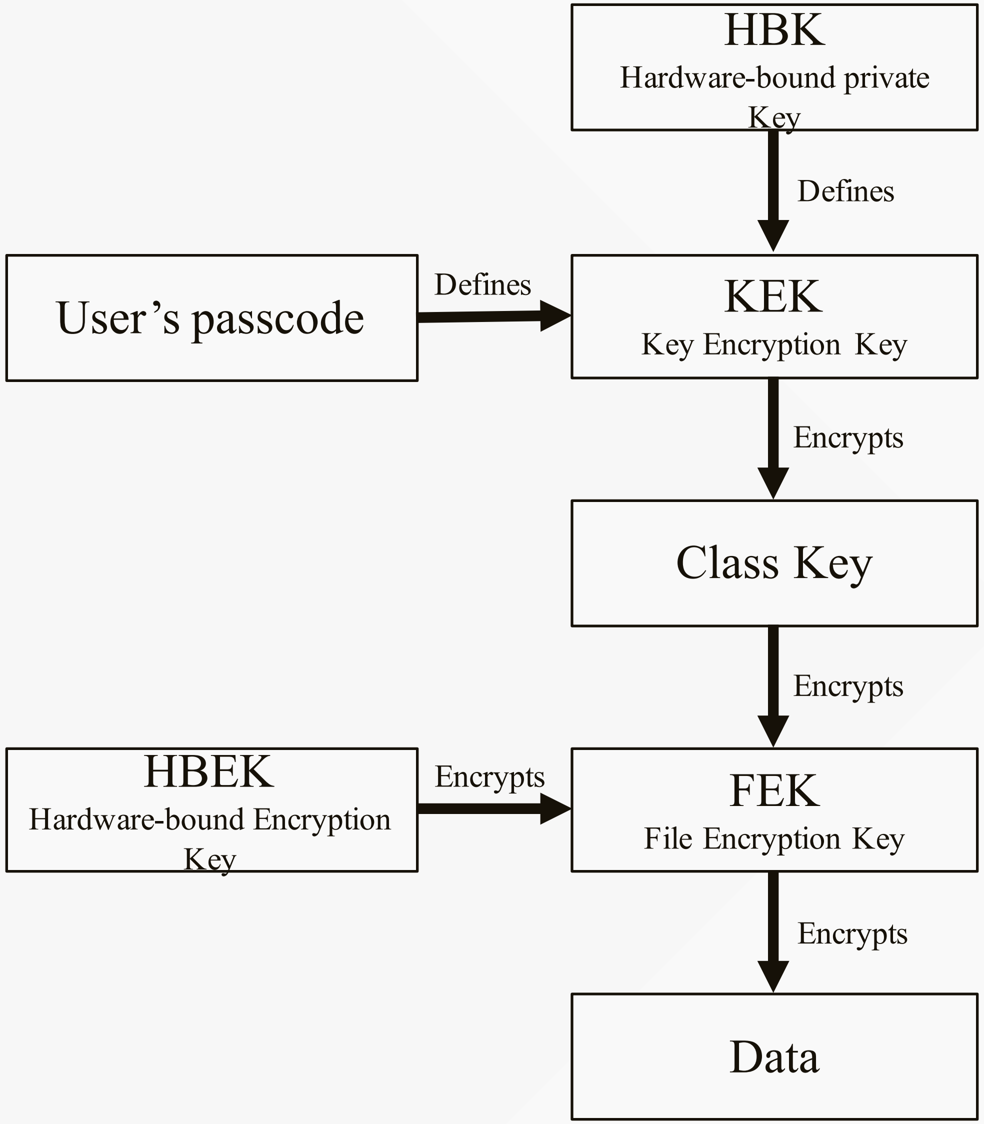
iOS Encryption



Full disk encryption



eCryptfs++



HBK
Hardware-bound private Key
Defines
User's passcode
Defines
KEK
Key Encryption Key
Encrypts
Class Key
Encrypts
HBEK
Hardware-bound Encryption Key
Encrypts
FEK
File Encryption Key
Encrypts
Data