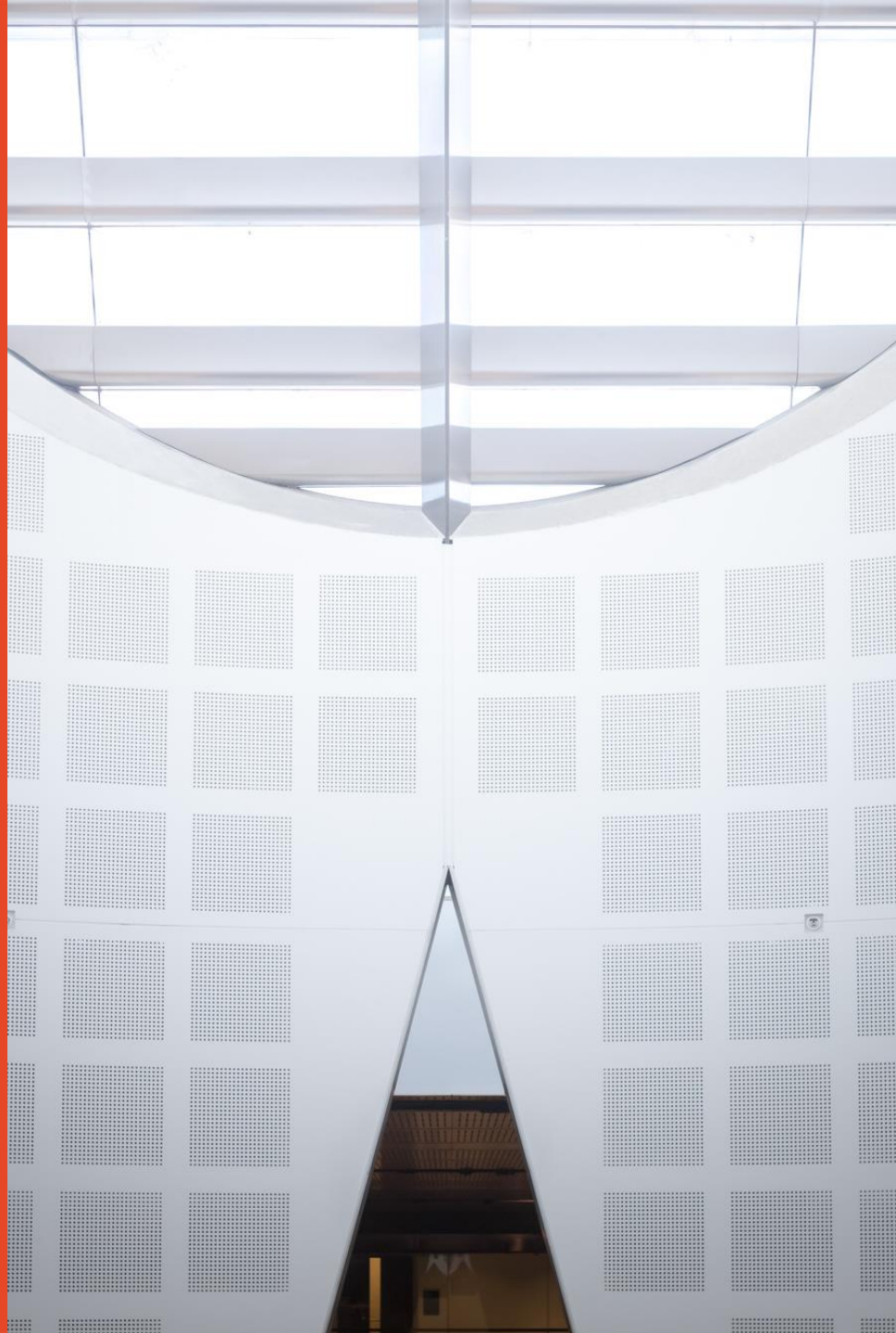
Mobile Computing
COMP5216/COMP4216

Week 09

Semester 2, 2022

Dr. Kanchana Thilakarathna
School of Computer Science

THE UNIVERSITY OF SYDNEY



Outline

- State of Mobile Security & Privacy
- What is Privacy ?
- Mobile Security threat models
- Security of Mobile Operating Systems
  - App sandboxing
  - Permissions
  - Releasing apps
- Best Mobile Security Practices

Security Challenge

- Exponential **growth of smart devices** and **third party apps**.
- Leads to security & privacy threats:

- Theft of personal information.

- Increased risks of malware.

Security concerns of smart devices

Healthcare challenges: Ransomware and the Internet of Things are the tip of the iceberg

BY LYSA MYERS POSTED 7 APR 2017 - 02:00PM

RANSOMWARE

Wearable fitness trackers in the workplace: surveillance by fitbit?

By Clare Gilroy-Scott on 26 Apr 2017 in Data protection, Employment law, Occupational Health, Staff monitoring, Wellbeing

13

Wearable fitness trackers such as fitbit are promoted as useful tools for employee wellbeing programmes. But employers that collect and monitor data from this technology risk breaching data protection law if their policies and procedures are not kept up to date. Clare Gilroy-Scott of law firm Goodman Derrick advises.

Zero-day mobile malware surged 92% in last six months

Networks Asia staff | August 31, 2018

Share G+ Like 0 Tweet share Print Email

In the last 6 months, Pradeo Lab has observed a massive 92% rise of zero-day malware on mobile devices, demonstrating that hackers are strongly focusing their attention on enterprise mobility and constantly innovating to overcome security fences.

"FITNESS AND MEDICAL DEVICES ARE OFTEN FULL OF SENSITIVE INFORMATION, YET SECURITY AND PRIVACY ARE OFTEN AN AFTERTHOUGHT."

More Devices Means More Targets

First, we had to worry about the physical security of our computers. More recently, we have learned to worry about mobile phones and tablet devices. Now, according to CIO, "we have to worry about protecting our car, our home appliances, our wearables and many other IoT devices."

If your mobile phone is running slowly or always losing battery, it might have been hacked to mine cryptocurrency – here's how to protect yourself

Ana Zarzalejos, Business Insider España 18h 14,090

Simple but extremely effective: Inside the world's most prolific mobile banking malware

Asacub trojan has quietly been going about its business for years, stealing funds from hundreds of thousands of victims - but it can also be easily avoided.

By Danny Palmer | August 29, 2018 -- 14:28 GMT (00:28 AEST) | Topic: Security

Security threats are expected to grow further…

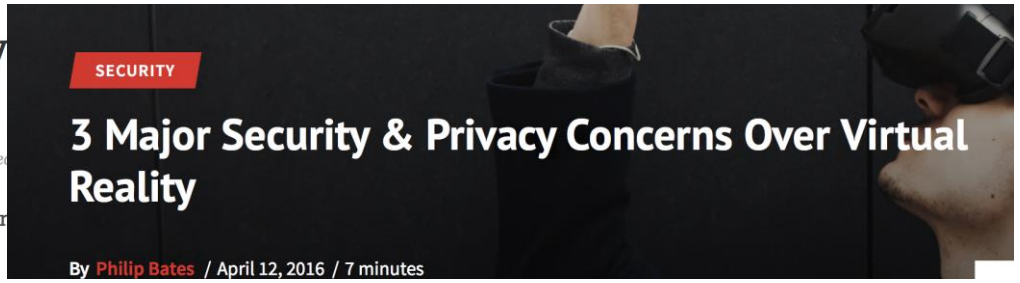
- Advanced sensing - 3D, IR cameras, HR, Brainwaves, etc.

The new privacy debate: ensuring privacy … reality' world

December 14, 2016 · by itu4u · in Cybersecurity/Trust, Emerging Trends, IoT, Uncategorize

"I'm taking everybody's privacy away!" Robert Scoble, Entrepreneur in Reside… declared during his Centre Stage debate at Web Summit 2016.

Wearing a pair of M… generation "mixed … debated whether we… our privacy in the n… advancement.



Culture | Ethics | Technology | Virtual Reality

Mixed Reality Comes With New Privacy Concerns

May 20 · Versability · 0 Comments · augmented reality privacy, mixed reality privacy, virtual reality privacy

At this point, we're all fully aware that everything you do can be put online, and whatever's online can be seen by a lot of people (unless, of course, it's on this blog).

Big Tech recasts 'wearables': Privacy concerns may draw regulatory glare

As companies both in the internet and consumer electronics space attempt … technology, the next port of call could seemingly be augmented reality and …

Written by **Pranav Mukul** | New Delhi | September 13, 2021 1:18:09 am

Explained: What Facebook's Ray-Ban Stories smart glasses do, and the concerns they raise

Facebook's smart glasses will let users record the world around them, and take pictures. This is exactly what Snap's Spectacles also let users do.

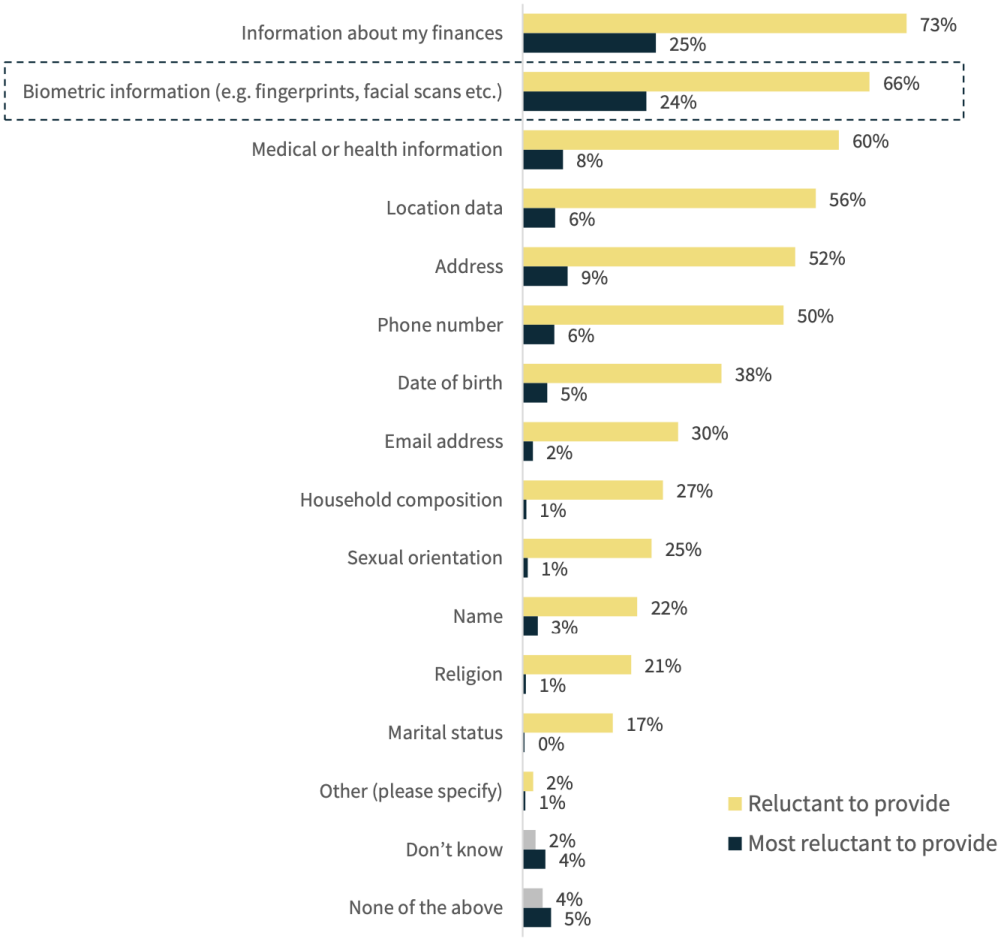
Written by **Shruti Dhapola** , Edited by Explained Desk | New Delhi | Updated: September 11, 2021 1:32:56 pm

ADVERTISEMENT

What is Privacy ?

- "Personal Information"
  - Any information that identifies you or could reasonably be used to identify you
    - E.g. name, address, financial details, opinions, memberships, ethnic origin, health information, criminal record, etc.
  - Not just demographics
    - E.g. photos, IP address, Device IDs, MAC address, Contact list, Call history, Location, Installed apps, etc.
- **Carefully treat and protect personal information collection, use, storage and sharing through your service**

What do people consider private information?



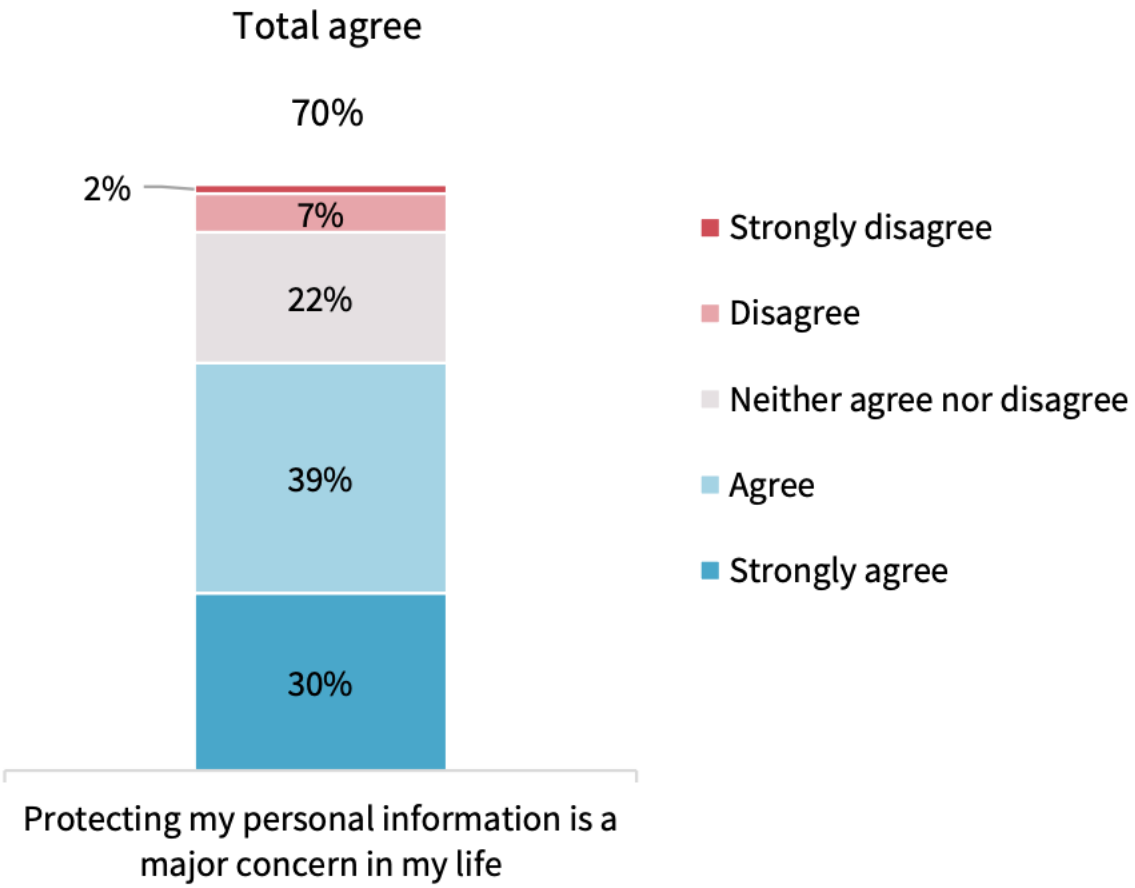
B3. Thinking now about providing your personal information to any business, organisation or government agency, in general, what types of information are you reluctant to provide? B4. And which one of these do you feel most reluctant to provide? Base: Australians 18+ (n=1,506)

What is Privacy?

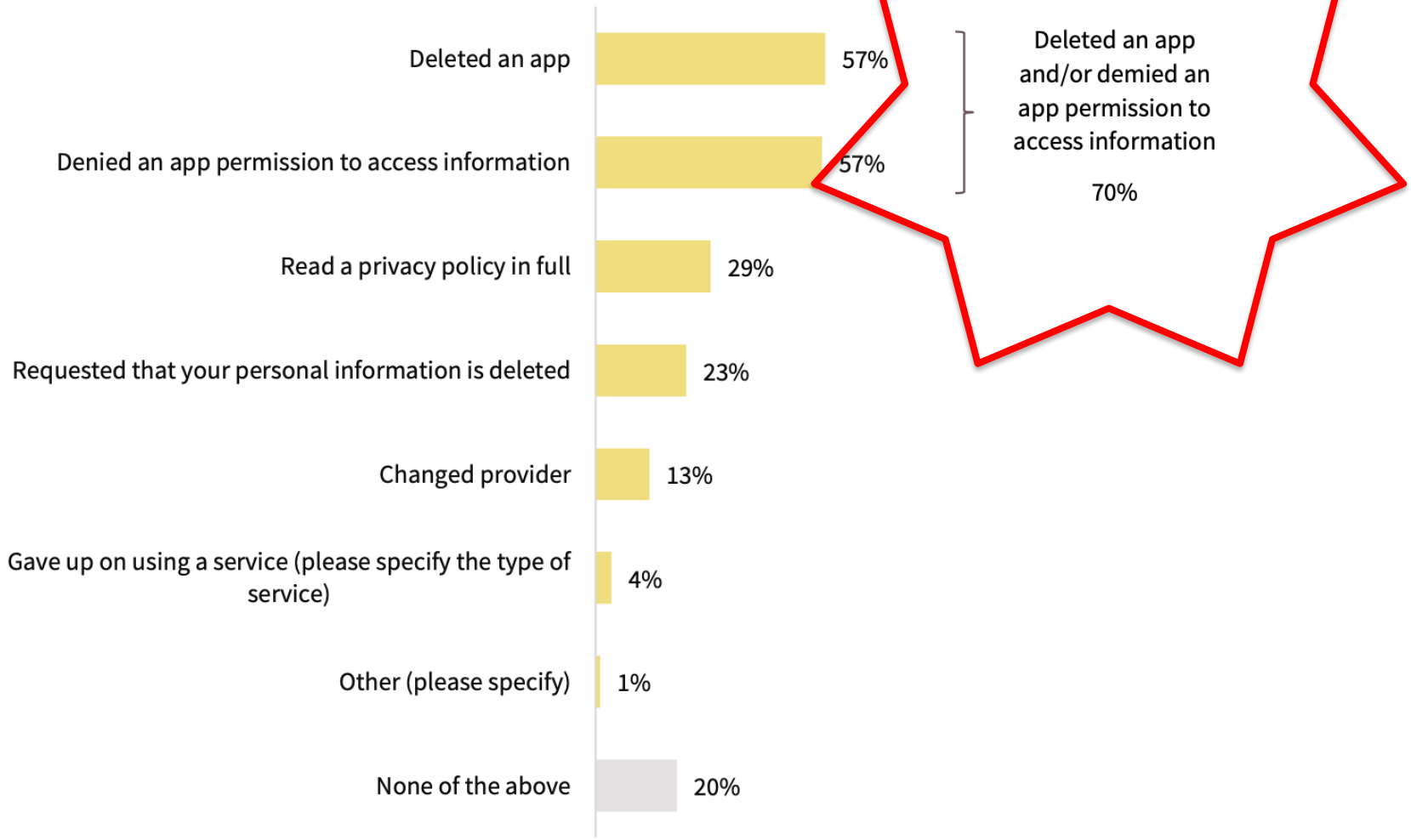
- What Privacy means to Australians? –
  - Australian Community Attitudes to Privacy Survey 2020

- the idea of keeping one's information private and confidential (41%)
- the idea of having control over one's information (27%)
- the concept of protection against harmful practices and security (19%)
- the idea of living free from interference and maintaining one's lawful right to be left alone (18%)
- the idea of not having one's information shared or sold without permission (11%), and
- the right to security and respect (11%).

Do people care of their privacy?



What people have done to protect their privacy?



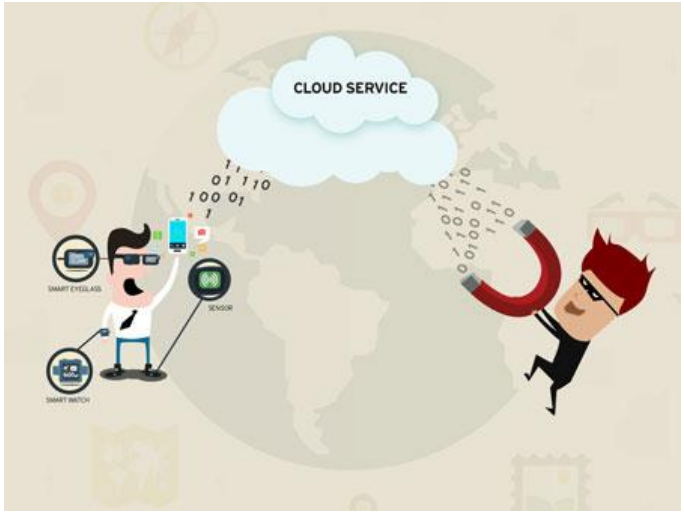
A11. Have you ever done any of the following out of concern for your data privacy? Base: Australians 18+ (n=1,510)

Outline

- State of Mobile Security & Privacy
- What is Privacy ?
- Mobile Security threat models
- Security of Mobile Operating Systems
  - App sandboxing
  - Permissions
  - Releasing apps
- Best Mobile Security Practices

Mobile Security Threat Models

- **Physical Attacks**
  - Circumvent authentication to unlock the device.
- **App Attacks**
  - Use malicious app to hijack the access to other apps, etc.
  - Code tampering
- **System Attacks**
  - Use mobile platform (Apple, Android, etc.) vulnerabilities which impacts all apps installed on the device.
- Server/Cloud Attacks
  - Data breaches
  - Common to all other web services
- Network Attacks
  - Use packet sniffing or spoofing
  - Man-In-the-Middle attacks
  - Common to all other web services



Physical Attacks

- Current device unlocking methods - Passwords, PINs, Patterns, Biometrics
  - Once unlocks all apps are accessible

- **What are the potential authentication attacks?**

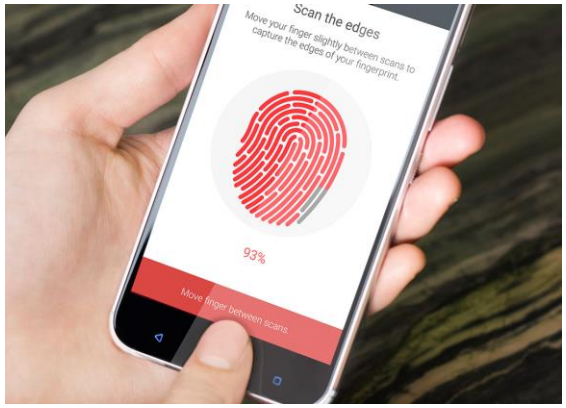
- Smudge attacks [Aviv et al. 2010]
  - Entering patterns leave smudge that can be detected with various lighting techniques
  - Aviv, A. J., Gibson, K. L., Mossop, E., Blaze, M.,&Smith, J. M. (2010). Smudge Attacks on Smartphone Touch Screens. Woot, 10, 1-7.

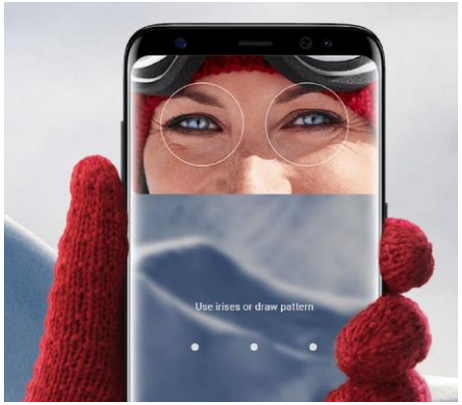
- Fingerprint extraction
  - Many demos on YouTube

Physical Attacks

- People choose common simple patterns
  - Low entropy – Faster brute force attacks
  - At most 1600 patterns with less than 5 strokes
- People often reuse passwords, PINs
- Security questions are often very standard, with predictable answers and limited possibilities
  - Mother's maiden name? – depending on culture, try Smith, Chang, Kim, Schmidt, ...
  - First car? – try Golf, Yaris, Corolla, ...
  - Social networks help collect additional information about a person

Physical Attacks

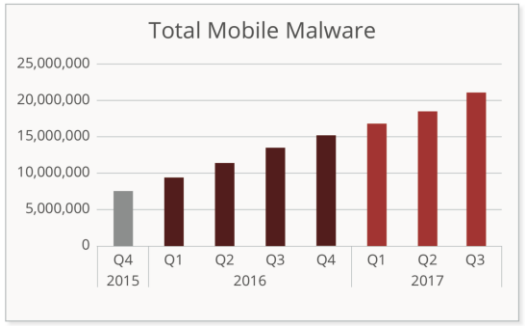




- Is our phones more secure than earlier with biometric authentication?
  - Most (if not all) biometric authentication falls back to PIN
  - No more secure than PIN
- Biometrics – if compromised, lost for ever
  - Can not be changed

App Attacks - Mobile Malware

– Capable of performing System Attacks and/or App Attacks



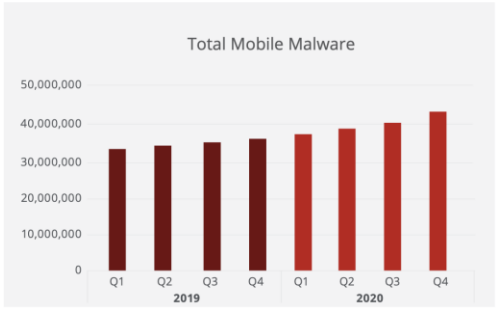


Figure 7. Total mobile malware detections by quarter.

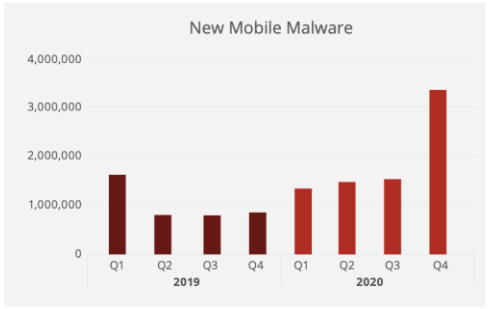
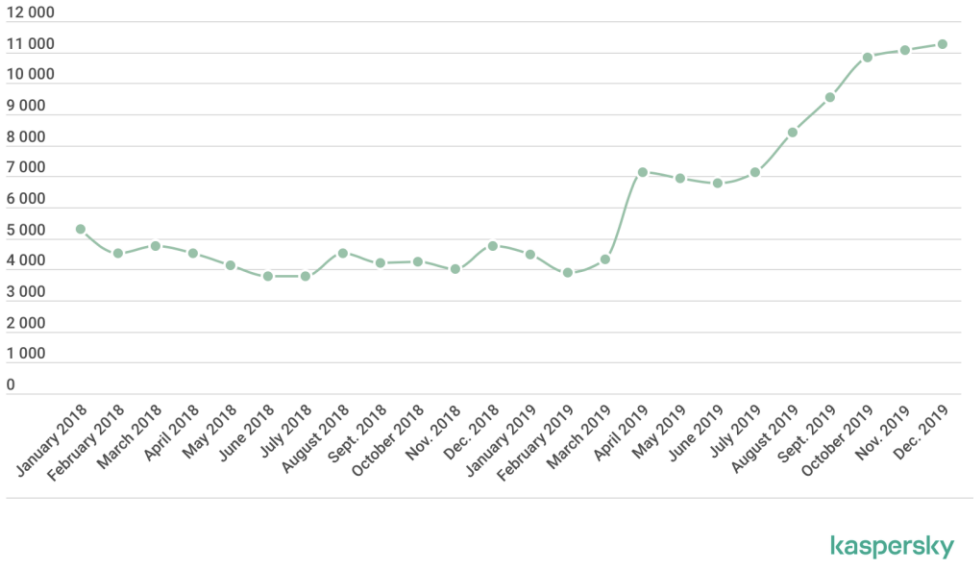
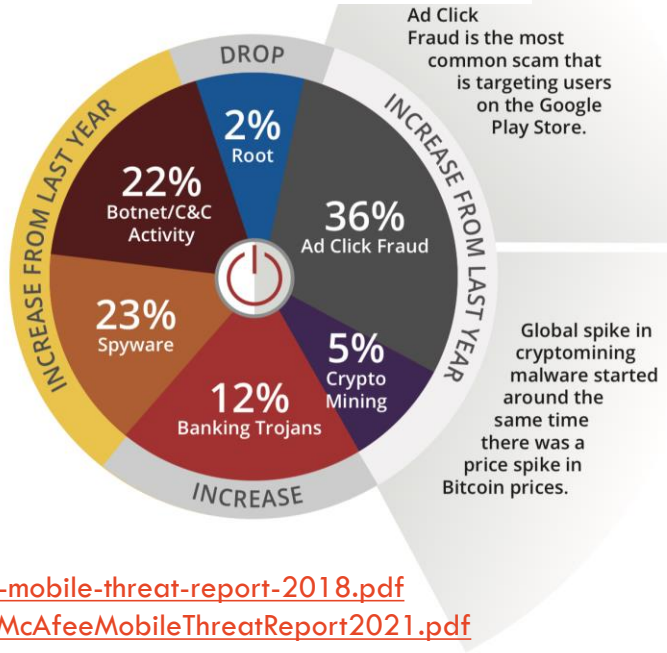


Figure 8. New mobile malware detections by quarter.



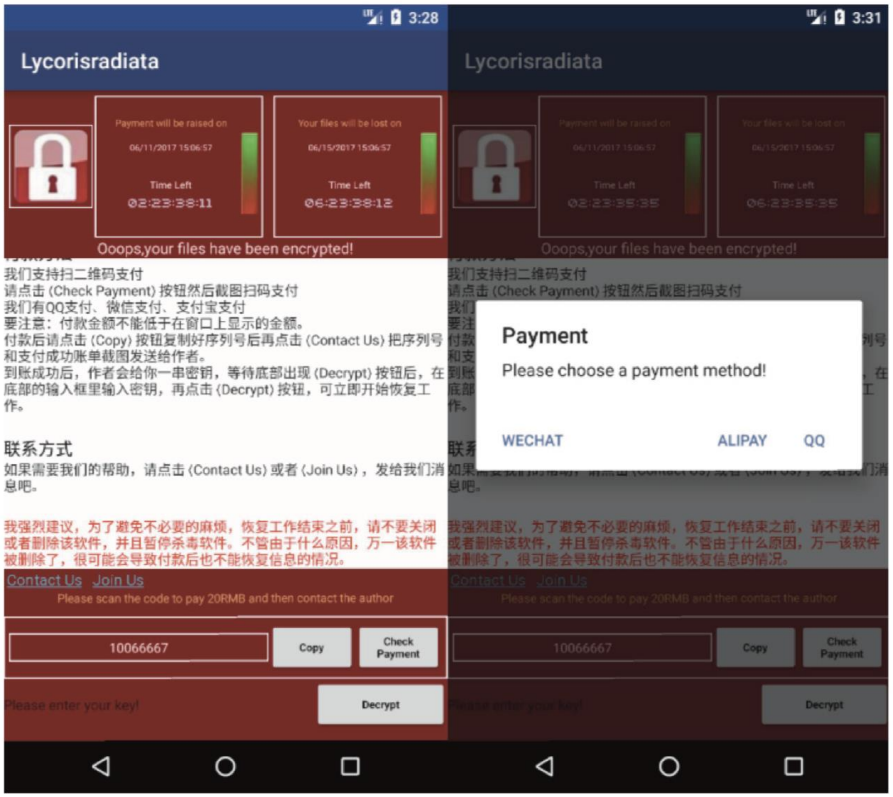
kaspersky



https://www.mcafee.com/enterprise/en-us/assets/reports/rp-mobile-threat-report-2018.pdf
https://www.mcafee.com/content/dam/global/infographics/McAfeeMobileThreatReport2021.pdf

App Attacks - Mobile Malware

- Ransomware example: Fake app for popular Chinese game King of Glory
  - Direct user to pay via WeChat, AliPay, QQ



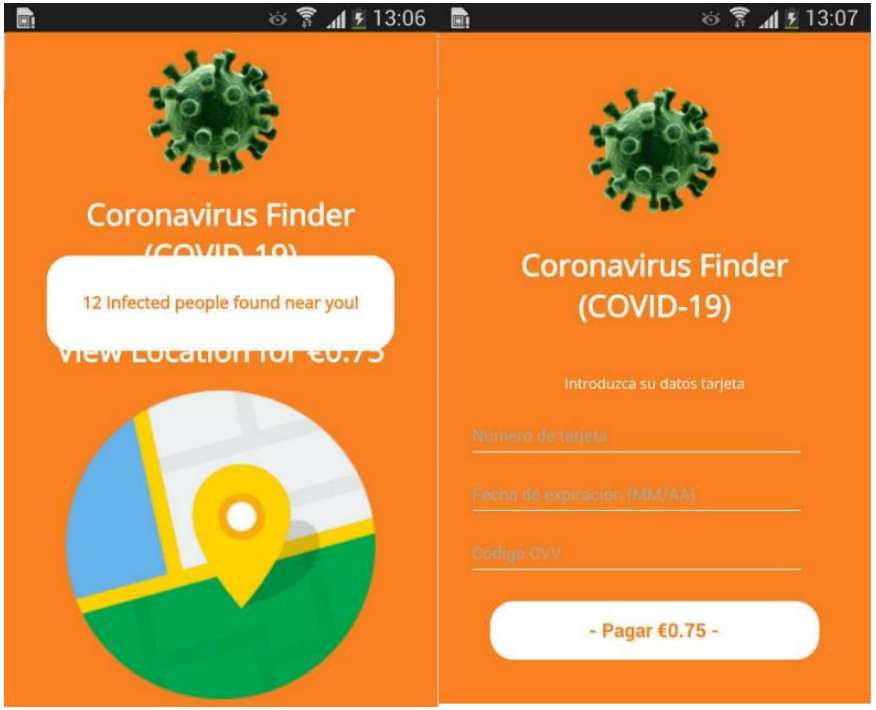
https://www.sophos.com/en-us/medialibrary/PDFs/technical-papers/malware-forecast-2018.pdf?la=en

App Attacks - Mobile Malware

- Types of Android Ransomware
  - Lock Screen Ransomware
  - Crypto
  - Send SMS
  - Steal sensitive information
  - Disable anti-virus software
- Advertisement Hijacking
  - Take a popular application & change the advertisement ID
  - Publish in a different app market
- For fun: Change scores in games/Skip levels

App Attacks - Mobile Malware

- Pandemic themed Malware
  - GINP banking Trojan pretended to be an app that searched for COVID-19-infected individuals: the victim was coaxed into providing their bank card details under the pretext of a €0.75 fee charge.



https://www.kaspersky.com.au/blog/ginp-trojan-coronavirus-finder/27096/

System Attacks – OS vulnerabilities

- Android exploits and vulnerabilities
  - Janus attack – 2017

https://medium.com/mobis3c/exploiting-apps-vulnerable-to-janus-cve-2017-13156-8d52c983b4e0

Modify the APK (add extra bytes) without changing the signature

- Exploited to update an already installed app without the knowledge of the developer

  - Stagefright attack - 2015
  - https://www.androidcentral.com/stagefright
    - A video sent via MMS could be used to attack libStageFright mechanism which process video files
    - Exploited to do remote code executions

System Attacks – OS vulnerabilities

- "**Rooting**" Android Devices
  - Enables "Root" access to the system
  - Allows to replace the existing OS with custom ROMs
- "**Jail Breaking**" iOS Devices
  - Allows to bypass the app signatures
  - Exploited to download & install apps, extensions, from outside Apple AppStore
- Popularity of jail breaking and rooting are going down
- Vendor are also keep making it difficult to highjack the OS

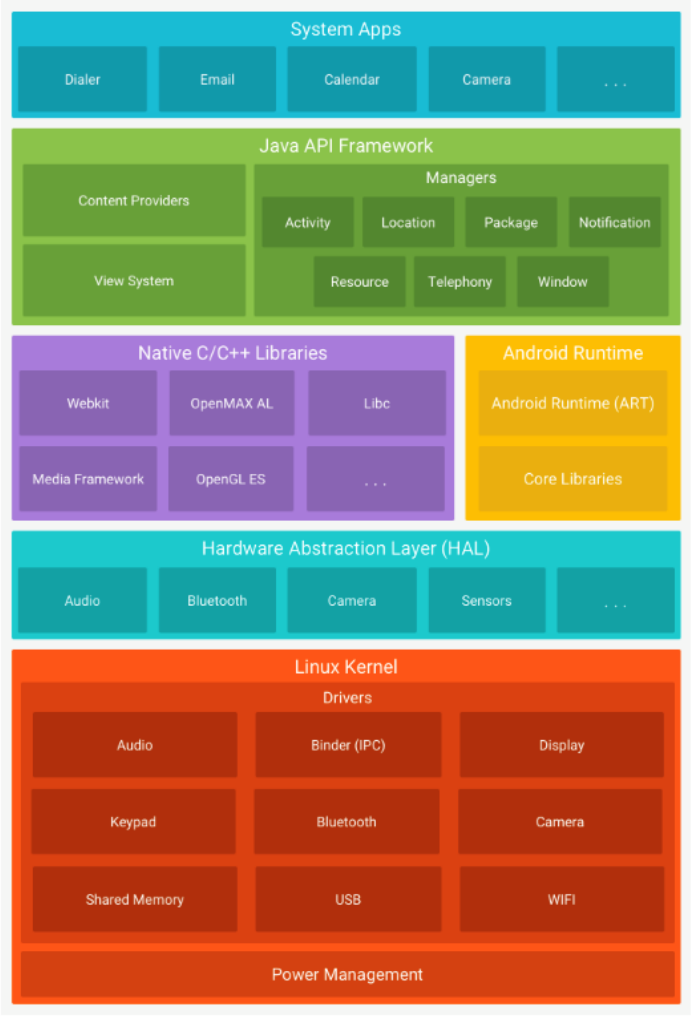
Security of Mobile Operating Systems

Operating Systems got you covered (mostly) …

- Closed-source operating system based on Unix (Darwin)
- Apps are developed in Swift
- Native development in Objective-C
- **App Sandbox**
- **User permission structure**
- Vendor (Apple) **singed app release**

- Open-source operating system based on Linux (by Google)
- Public review, no obscurity
- Native development in Java
- **App Sandbox**
- **User permission structure**
- Developer (self) **singed app release**

Android OS Architecture



Source: Android developer documentation

Applications: Users interact with the device via the apps. Can be either first party or third party.

Android Framework: Provides basic functions such as communication between apps, managing voice calls or managing app life cycles.

Native Libraries: C/C++ libraries that contain instructions to the device on handling different types of data. E.g. Webkit, SSL, SQLite, and OpenGL.

Android Runtime: Dalvik Virtual Machine and Core Libraries.

Hardware Abstraction Layer (HAL): Converts the Java API calls to system calls that is understood by the Linux kernel.

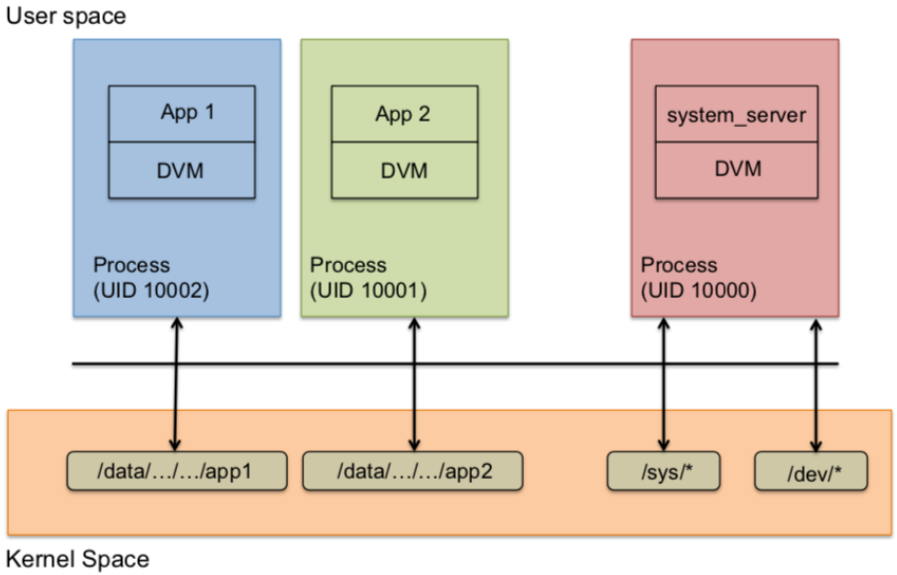
Linux Kernel: A kernel built on top of Linux kernel2. Additional modifications done by Google to make it suitable for smartphones (E.g. power management). Handles all conventional operating system functions such as process management and memory management.

1. Android App Sandbox

- Similar to user-based protection model in Linux

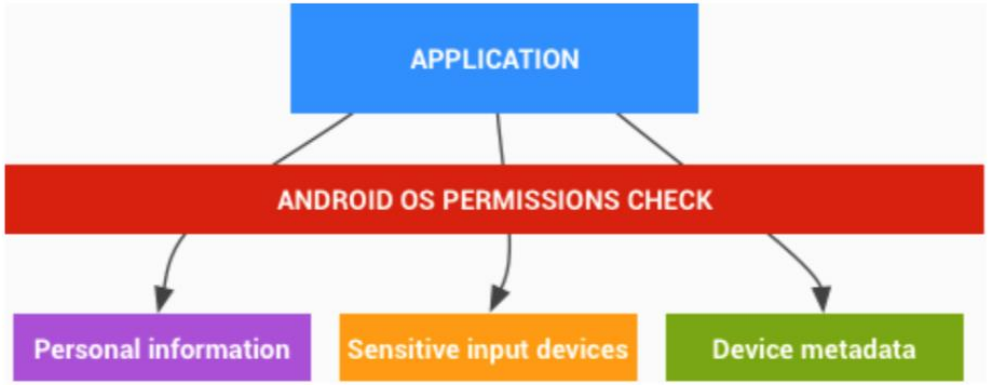
1. **Each app runs with its UID in its own Dalvik Virtual Machine**
2. **Apps are not allowed to talk to each other**
3. **Limited access to the OS (Kernel)**

- Apps must explicitly share resources and actions by declaring the required permissions for additional capabilities not provided by the basic sandbox



2. User Permission Structure

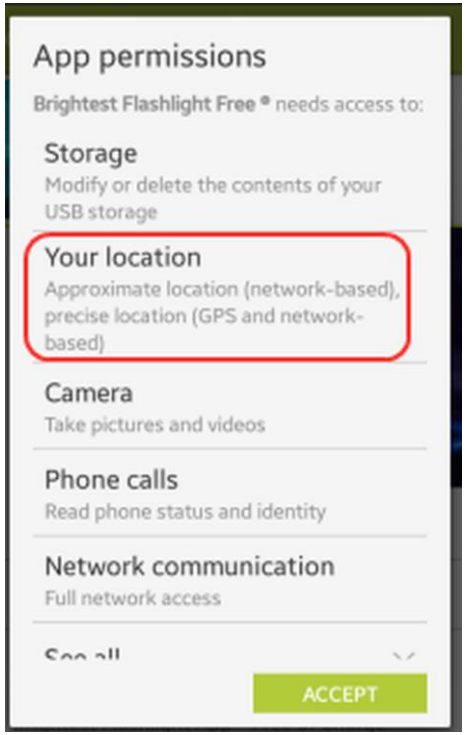
- App must get permission to do anything that
  - Uses data or resources that the app did not create
  - Uses network, hardware, features that do not belong to it
  - Affects the behaviour of the device
  - Affects the behaviour of other apps
- **If it isn't yours, get permission!**



2. User Permission Structure

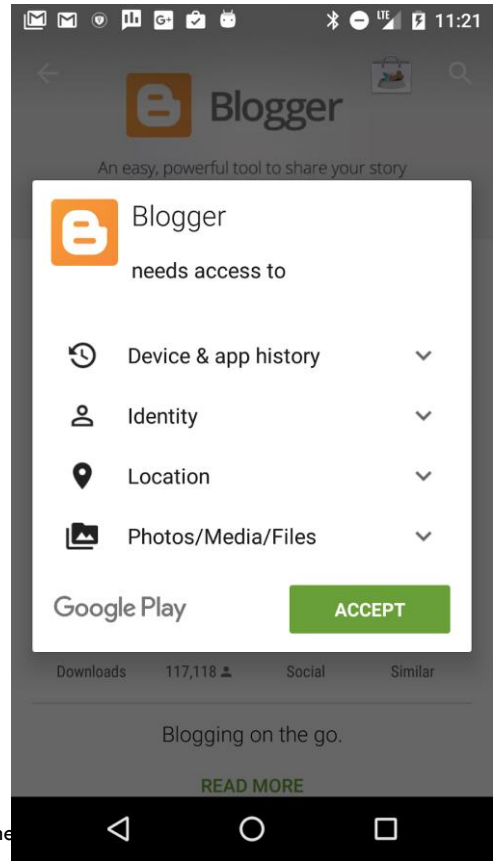
- **Normal** permissions do not directly risk the user's privacy
  - *Example*: Set the time zone
  - Android automatically grants normal permissions.

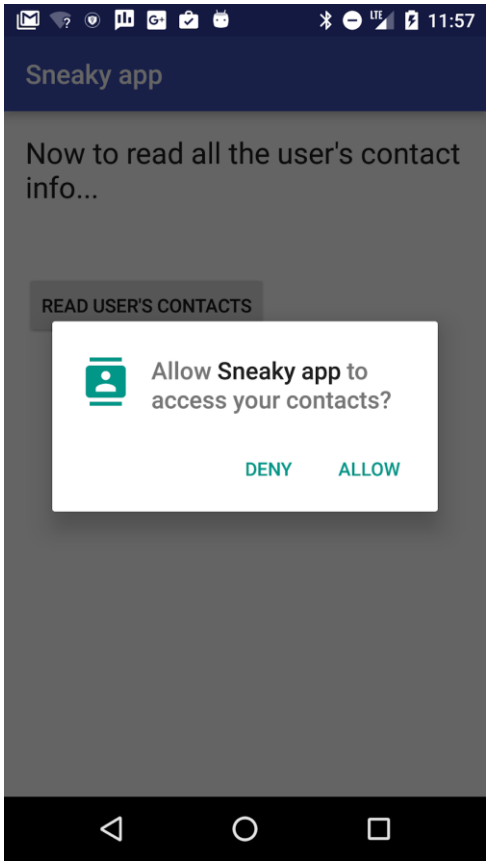
- **Dangerous** permissions give access to user's private data
  - *Example*: Read the user's contacts
  - Android asks user to explicitly grant dangerous permissions



2. User Permission Structure

- Before Marshmallow (API 23)
  - Grant permission before installing
- After Marshmallow (API 23)
  - App must get runtime permission

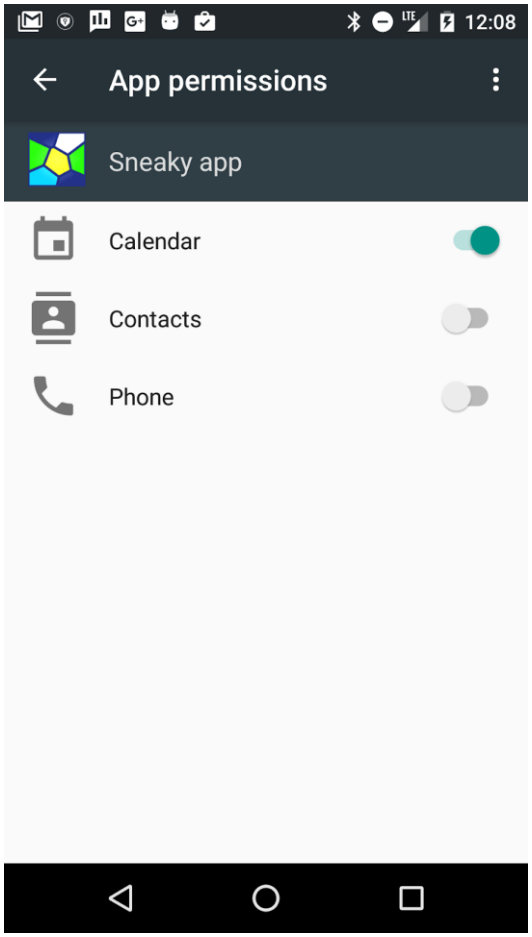




2. User Permission Structure

- Before API 23 → Uninstall app !
- After API 23
  - Can revoke each permission at any time **Settings > apps > permissions**

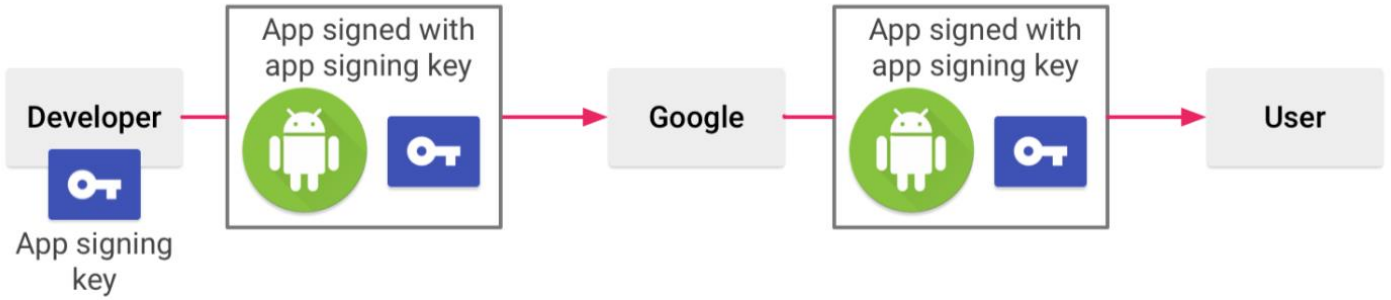
- Use Android Support Library to develop backward compatible permission structure



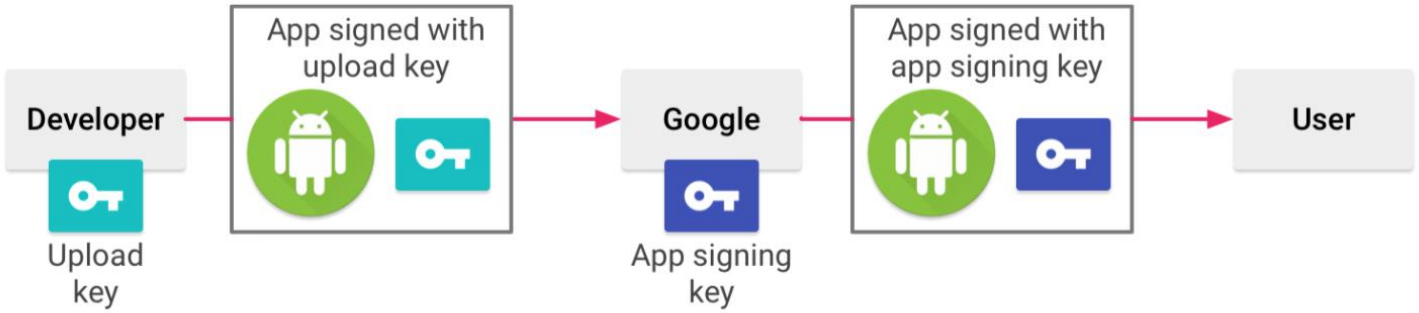
3. Android app signing process

- The code we write is built to an Android Application Package (APK)
- **Developer (self) signed app release**

Managing your own key



Google Play App signing



Must use Play App Signing if you want to distribute app with Google Playstore since August 2021

Outline

- State of Mobile Security & Privacy
- What is Privacy ?
- Mobile Security threat models
- Security of Mobile Operating Systems
  - App sandboxing
  - Permissions
  - Releasing apps
- Best Mobile Security Practices

Security Best Practices

Best Practices for Privacy Aware Apps

- Do not ask "personal information" if not necessary
- **Privacy by Design**
  - Building privacy and data protection up front, into the design specifications and architecture of information and communication systems and technologies, in order to facilitate compliance with privacy and data protection principles
- Make privacy your competitive advantage
- Why do so many people claim to be concerned about privacy… but at the end do nothing to protect it?
  - Will anyone buy privacy?
    - We buy curtains/blinds
- Draft a privacy policy (data management procedure) if you access sensitive information
- Beware of what you log. Android log can be read by other apps with `READ_LOGS` permission

Advanced privacy controls

- Anonymity (unlinkability) - Data is not linked to an identity
  - Location anonymity (Tor, mixes)
  - Data anonymity - "we anonymized the data before releasing it"

- Perturbation of data
  - E.g. Australian Census Data Release – TableBuilder
  - https://www.abs.gov.au/websitedbs/d3310114.nsf/home/about+tablebuilder
  - How TableBuilder works?
    - https://unece.org/fileadmin/DAM/stats/documents/ece/ces/ge.46/2013/Topic_1_ABS.pdf

- Adding noise to data
  - Differential Private data release and collection
    - **There is no extra risk for a particular individual being in the database to not being in the database.**

  - Local Differential Privacy exaples:
  - Google RAPPOR: https://github.com/google/rappor
  - Apple: https://www.apple.com/privacy/docs/Differential_Privacy_Overview.pdf

Legal support for privacy protection

- Failing to protect privacy could also result in a breach of the Privacy Act
  - https://www.oaic.gov.au/privacy-law/privacy-act/
- EU General Data Protection Regulation (GDPR)
  - https://www.eugdpr.org
- Australian Privacy Act 1988
  - https://www.oaic.gov.au/privacy/the-privacy-act/
- **Office of the Australian Information Commissioner**
  - **Tips for good privacy practice**
    - https://www.oaic.gov.au/privacy/privacy-for-organisations/tips-for-good-privacy-practice/
  - A better practice guide for mobile app developers Developed in 2014 – Old, but still provides useful guidelines
  - https://www.oaic.gov.au/resources/agencies-and-organisations/guides/guide-for-mobile-app-developers.pdf

User concerns for "Who is the data revealed to ?"

- The government ?
- Friends/Family ?
- The Internet ?
- Faceless company ?
- Local company ?
- International company ?
- Colleagues/employers ?

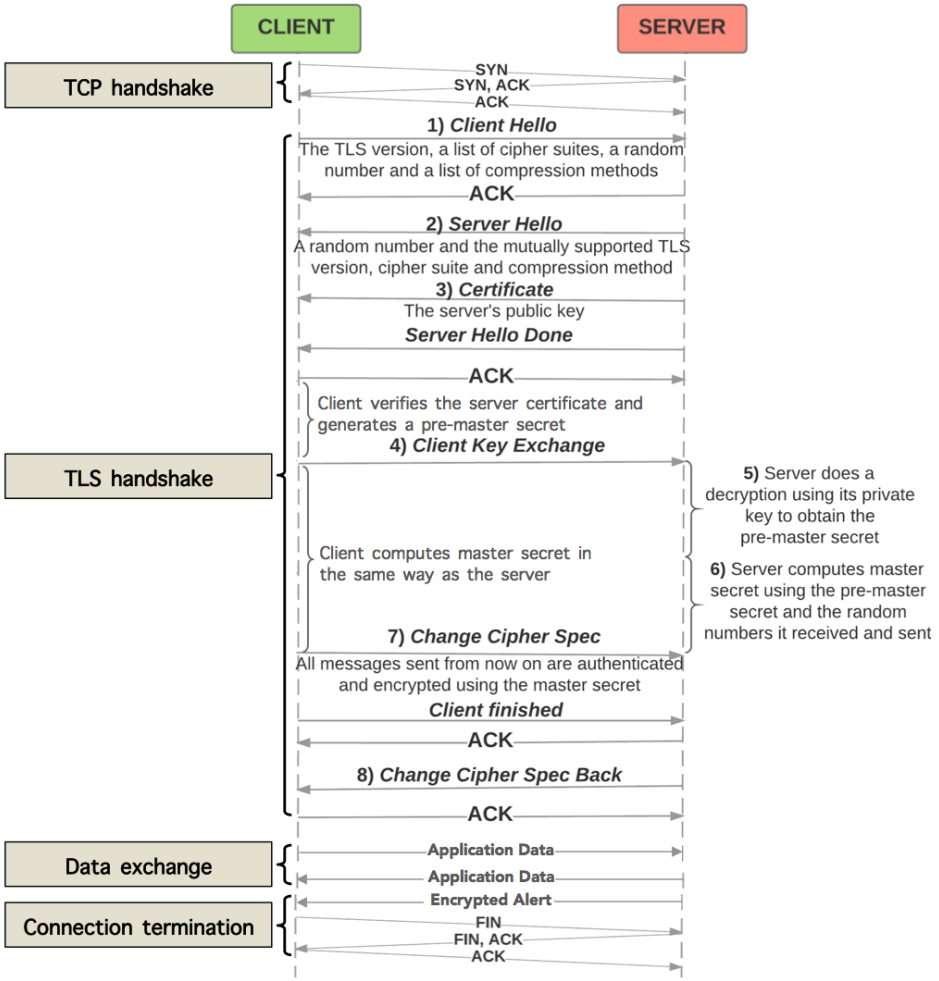
B8. Thinking now about trustworthiness. How trustworthy or untrustworthy would you say the following organisations are with regards to how they protect or use your personal information? Base: Australians 18+ (n=1,506)

Security Best Practices - Networking

- Minimize networking activities
- Authenticated, encrypted socket-level communication via SSLSocket class
- Avoid writing new protocols
- Never write new cryptographic algorithms
- Do not use SMS for sensitive information exchange
  - SMS are not encrypted
  - Not strongly authenticated
  - Can be read by any application with READ_SMS permission
- Use HTTPS over HTTP wherever, whenever possible
  - When is it not possible to use HTTPS ?

Security Best Practices - Why HTTPS (HTTP over TLS) ?

- If somebody can capture the network traffic generated by the previous app, he will be able to see what words you are looking for?
- Who potentially can capture the traffic generated by the smartphone?
- Solution: End to End Encryption → HTTPS

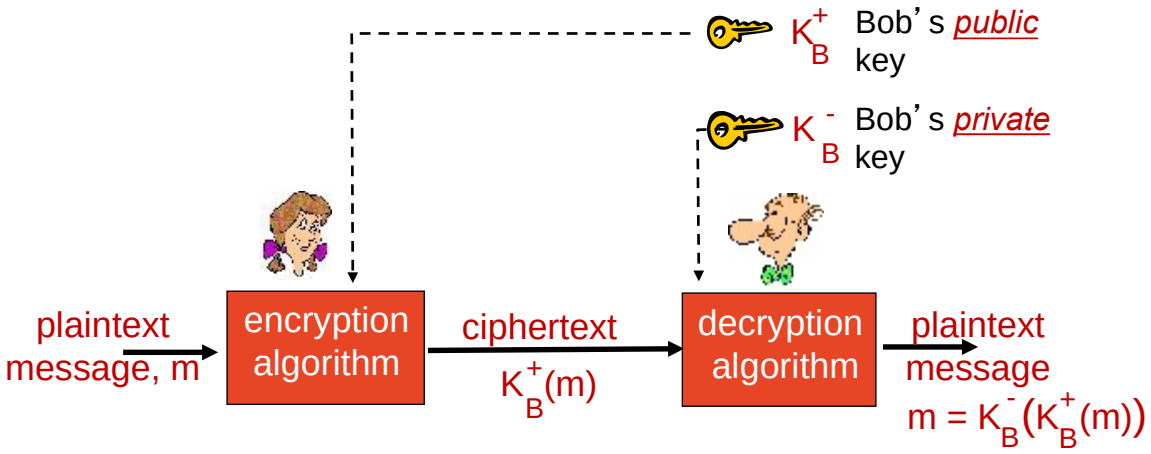


Security Best Practices - Encryption

Public key signature

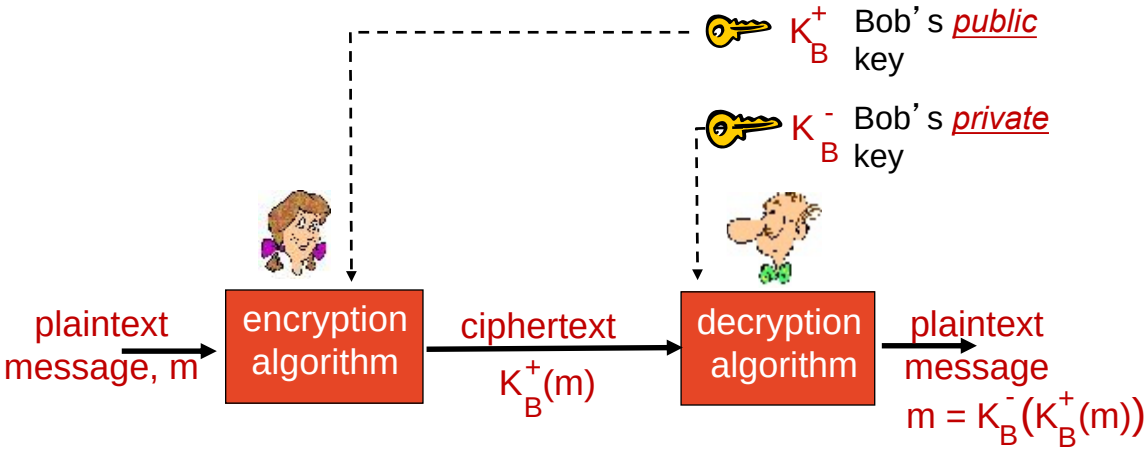
– Alice sends a message P to Bob

1. Alice encrypts it with her private key K_A^- and sends it off to Bob
2. She can use Bob's public key K_B^+ to keep the message secret and sends $K_B^+(P, K_A^-(P))$, combining P and the version she signed
3. Bob decrypts the signed version of the message with Alice's public key. If the message is the same as the non-signed one, then it has been sent by Alice.



Is this provide enough integrity?

Security Best Practices - Encryption



Issues with public key signatures

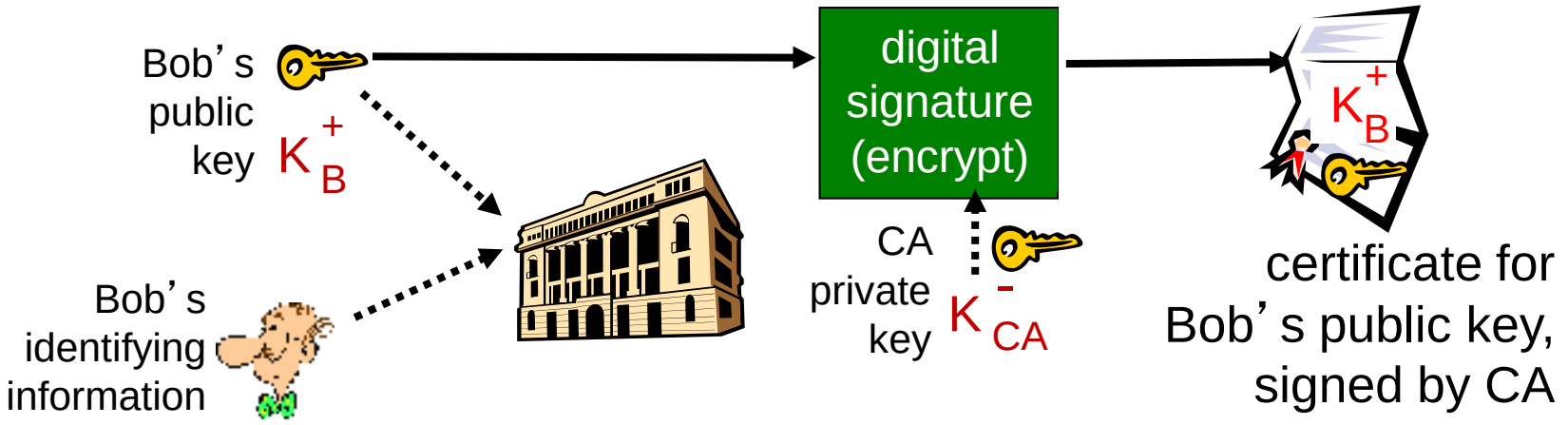
- Alice's signature is valid only until Alice's private key remains a secret
- If Alice wants to bail out, Alice could claim that her private key was stolen
- Alice can change her private key
- Central authority may be required keep track of keys

Security Best Practices - Validation of Certificates

- A certificate is a simple text file containing some information such as Company Name, the domain name, and a public key.
- Anybody can create such file and create a server pretending to be somebody else.
- Answer is Certificate Authorities.
  - Android comes with the set of CAs it trusts. Once you receive a certificate from a server & if it says it is issued by a trusted CA in the phones list, Android can verify the certificate.
- Example CAs are Comodo, Symantex, DigiCert, and Entrust.

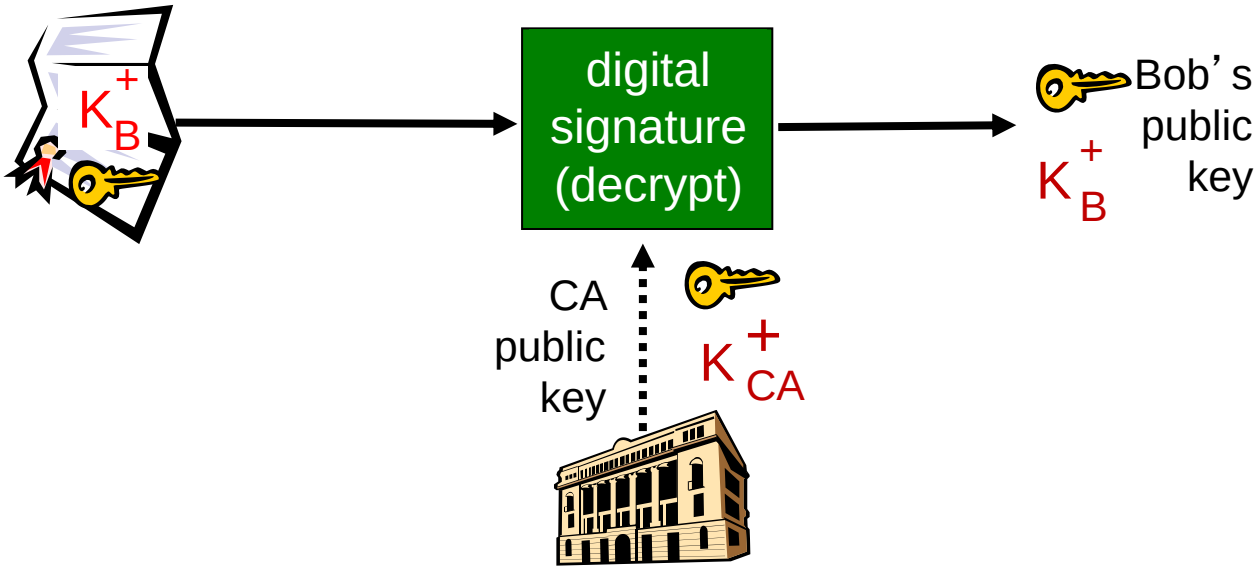
Certification authorities

- *certification authority (CA):* binds public key to particular entity, E.
- E (person, router) registers its public key with CA.
  - E provides "proof of identity" to CA.
  - CA creates certificate binding E to its public key.
  - certificate containing E's public key digitally signed by CA – CA says "this is E's public key"



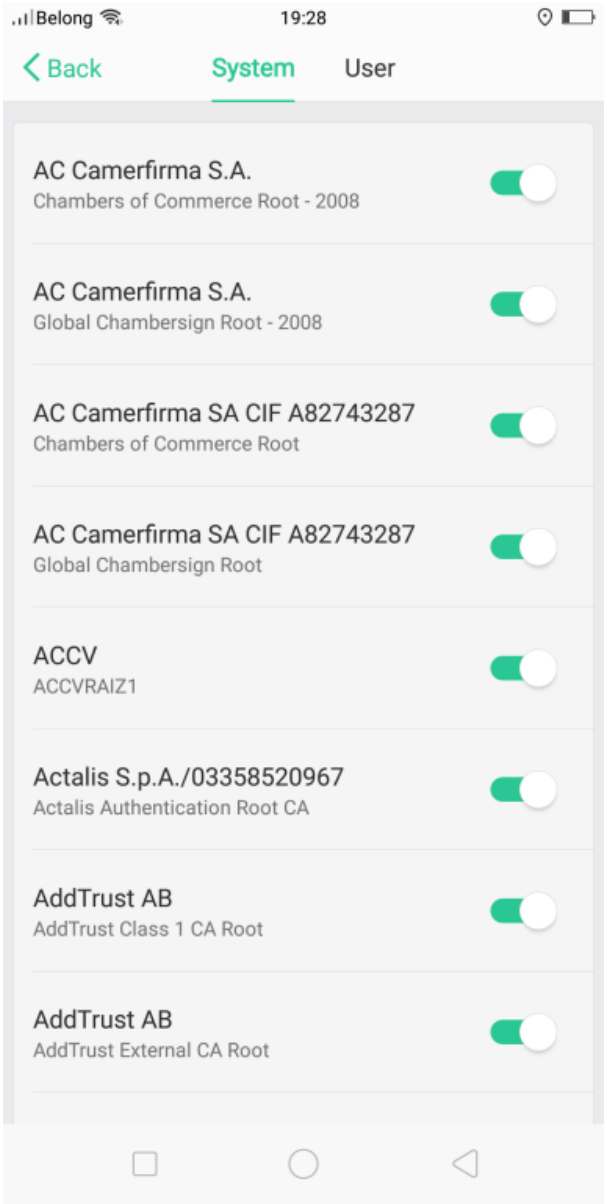
Certification authorities

- when Alice wants Bob's public key:
  - gets Bob's certificate (Bob or elsewhere).
  - apply CA's public key to Bob's certificate, get Bob's public key



Security Best Practices - Trusted CA in Android

- If you go to Settings → Additional Settings → Security & Privacy → Trusted credentials.



Security Best Practices – Digital signatures

cryptographic technique analogous to hand-written signatures:

- sender (Bob) digitally signs document, establishing he is document owner/creator.
- *verifiable, nonforgeable:* recipient (Alice) can prove to someone that Bob, and no one else (including Alice), must have signed document

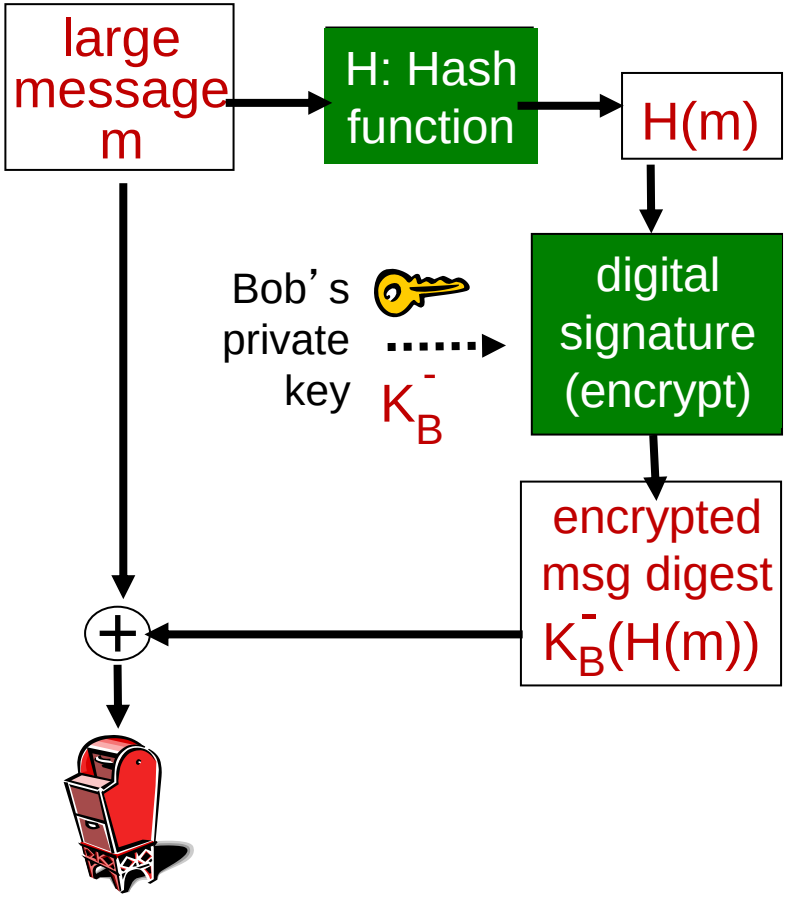
- Is it efficient to encrypt every item?
  - computationally expensive to public-key-encrypt long messages

goal: fixed-length, easy- to-compute digital "fingerprint"

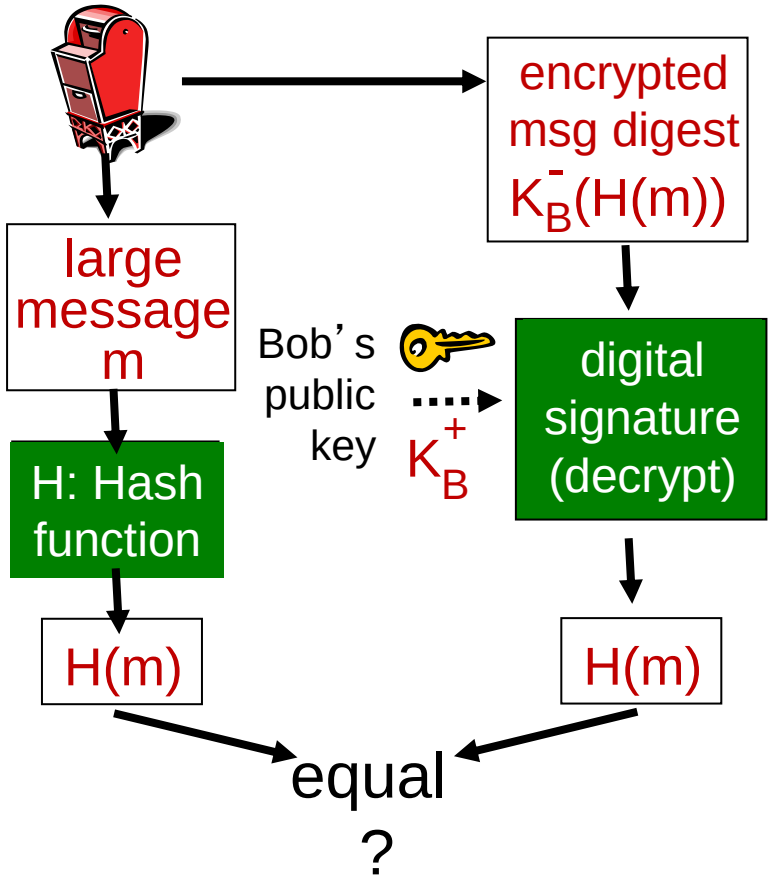
- apply hash function H to *m*, get fixed size message digest, *H(m)*.

Digital signature = signed message digest

Bob sends digitally signed message:



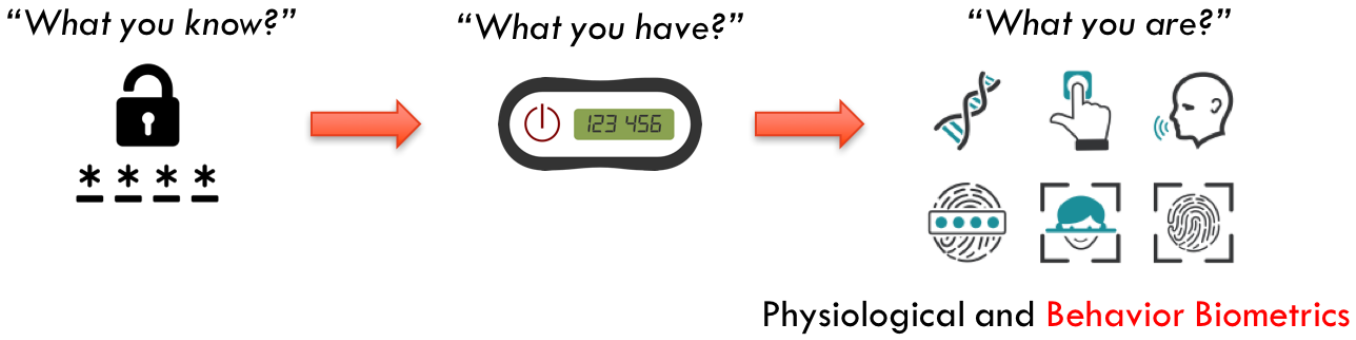
Alice verifies signature, integrity of digitally signed message:



Security Best Practices - Physical Attacks

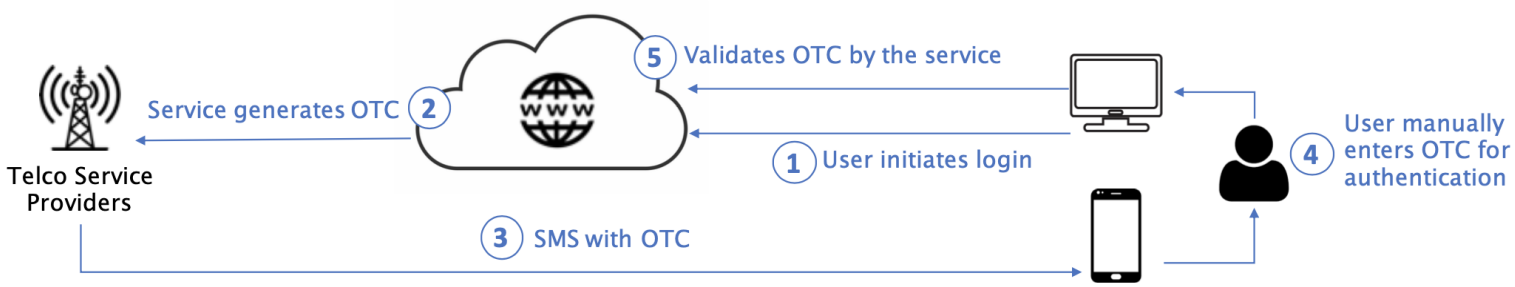
- PIN or a pattern for individual apps (second layer of defence)
  - E.g. Perfect AppLock https://play.google.com/store/apps/details?id=com.morrison.applocklite&hl=en

- Use **Multi-Factor Authentication**
  - Smartwatch, glasses, cloth, etc.



Security Best Practices - Physical Attacks

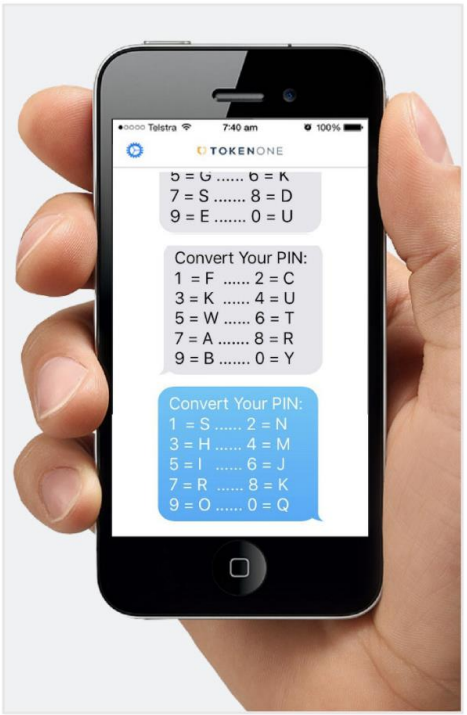
- Two-factor authentication with SMS messages and OTC (one-time-code)

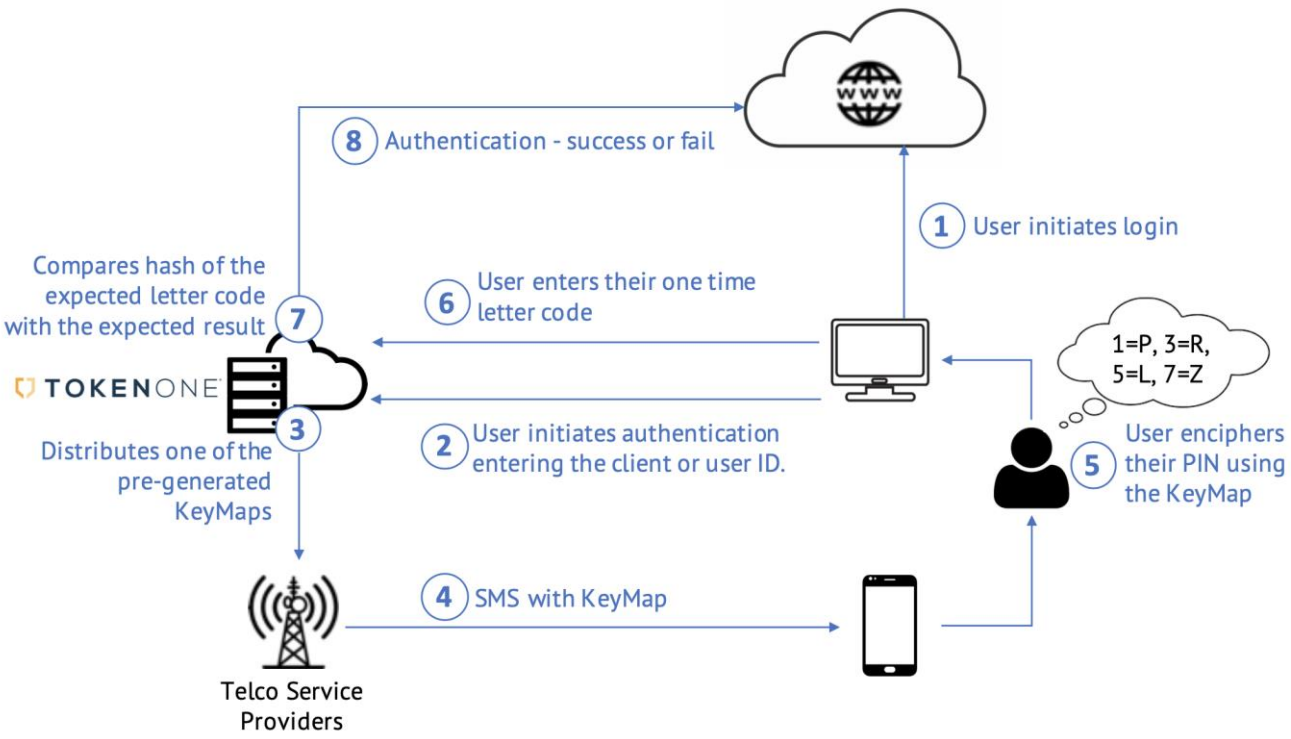


- Is this secure? What are the vulnerabilities?
  - **SMS interception/hijacking:** As a result of the less secure signalling protocols used in mobile networks
    - In 2017, attackers successfully intercepted the SMS authentication used by some German banks by creating a fake mobile network and sending messages to the O2-Telefonica mobile network
  - **SIM-swap**
  - **Mobile number port-out**
  - **Interception by malware and trojans**
    - Check Point Ltd. discovered a trojan named "EuroGrabber" which carried out similar attacks in Eastern Europe and swiped approximately $47 million from over 30,000 customers

Security Best Practices - Physical Attacks

- Advanced SMS-based two-factor authentication with KeyMaps
  - Merging with the ZKPP (Zero-Knowledge Password Proof)

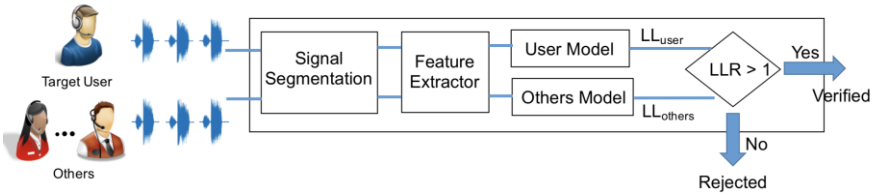




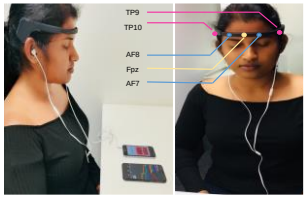
- E.g. https://www.tokenone.com

Security Best Practices - Physical Attacks

- Behavioural authentication for two-factor authentication
  - Nearly impossible to perfectly mimic behaviours, e.g. replay attacks.
- BreathPrint
  - Breathing acoustics for user authentication



- MusicID for smart headsets
  - Brainwave patterns for user authentication



- Sooriyaarachchi J, Seneviratne S, Thilakarathna K, Zomaya AY. MusicID: A Brainwave-Based User Authentication System for Internet of Things. IEEE Internet of Things Journal. 2020 Dec 15;8(10):8304-13.

Security Best Practices - IDs

- **Follow NIST Digital Identity Guidelines**
  - https://pages.nist.gov/800-63-3/sp800-63-3.html
- Don't store usernames and passwords on the device
- Use username and passwords for the initial authentication
- Use a hash or non-reversible form of data if you plan to transmit sensitive data
    - E.g. use hash of an email for the primary key, not the email address.
- Hash function H are used to produce a hash h of fixed length given a message m: $h = H(m)$
  - One-way function: computationally infeasible to find an input m that corresponds to an output h, whereas computing h from m is easy
  - Weak collision resistant: given an input m and an output h, it is infeasible to find another different input m' such that $H(m) = H(m')$

Security Best Practices - IDs

- User short-lived, service specific authorization tokens
  - Use the com.google.android.gms.iid InstanceID API.
  - Use randomUUID()
- For a unique identifier to track users across apps
  - Why ?
  - GUID (Globally Unique Identifier) is required, don't use IMEI or phone number
  - Create a large unique number
- For a unique identifier to track users across apps
  - For Advertising and Analytics
  - Use the Advertising Identifier available from the AdvertisingIdClient.Info class via the getId() method
  - https://developers.google.com/android/reference/com/google/android/gms/ads/identifier/AdvertisingIdClient

Security Best Practices - Permissions

- Only use permission that is necessary for the functionality of the app
- Beware of the permission requested by libraries
  - Users don't see the library, Users see your app.
  - Review libraries and pick the one with minimum permission
- Explain the reason for requesting a particular permission to the user
- Indicate when you access sensitive information to the user

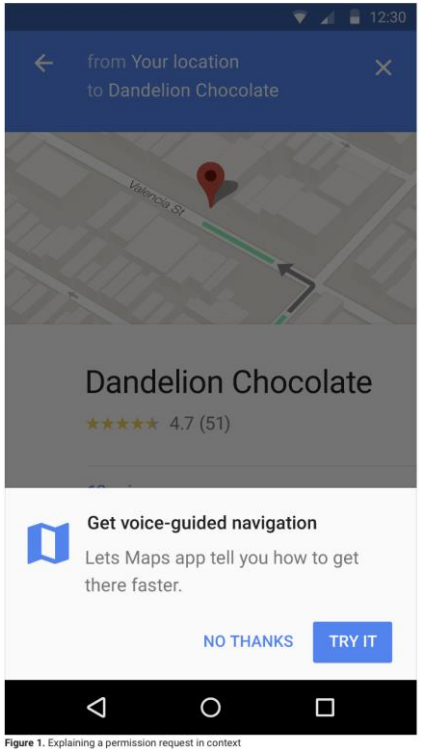


Figure 1. Explaining a permission request in context

Security Best Practices - Permissions

- Ask permission at the right time
  - E.g. Photo app (Camera Permission)
    - At the launch – Access to Camera
    - When user wants to share – Access to Contacts

- Ask the right (minimum) permission
  - E.g. Reducing the volume of audio playback when receiving a call
  - READ_PHONE_STATE permission allows you to detect receiving a call
    - It also allows you to read Phone Hardware IDs, SIM, Incoming phone number, etc. → **Over permission ?**
  - Instead, use `AudioFocus`
    - Don't need any permission
    - https://developer.android.com/guide/topics/media-apps/volume-and-earphones

Security Best Practices - Permissions

- Can you avoid using permission ?

Security Best Practices - Permissions

- Can you avoid using permission ?
  - **Use another app to perform the task you wanted… How ?**

Security Best Practices - Permissions

- Use another app to perform the task you wanted… How ?

- Example: Taking a Photo
  - With CAMERA Permission
    - Allows your app to access the Camera directly
    - You have to design the UI for taking a photo
    - Only prompt the permission request once

  - With Intent type MediaStore.ACTION_IMAGE_CAPTURE
    - You do not have to design the UI for taking a photo
    - User can pick the favorite app to take a photo
    - Your app will not have direct access to Camera
    - Selection prompt appears every time user invoke this action

Security Best Practices - Storage

- Three methods to save files
  - Internal Storage
  - External Storage
  - Content Providers

Internal Storage

- Only accessible to the app, good enough for most of the apps
- For more sensitive data, you can encrypt files
  - Do not make keys accessible to the app
  - Encrypt with KeyStore - https://developer.android.com/reference/java/security/KeyStore
- If you want to share data with another app…

Security Best Practices - Storage

- If you want to share data with another app…
  - Use Content Provider
  - Avoid the MODE_WORLD_WRITEABLE or MODE_WORLD_READABLE modes

External Storage

- Don't store sensitive information on the external storage
  - External storage can be readable and writable by every app
  - External storage can be removed by the user

- Perform input validation before receiving data from the external storage
  - https://developer.android.com/training/articles/security-tips#InputValidation

Security Best Practices – Web content access

- Carefully use WebView due to common exploits with HTML and JavaScript
  - E.g. Cross-Site Scripting
  - If you app do not use JavaScript, *do not* call setJavaScriptEnabled()
  - Carefully use addJavaScriptInterface() as it allows JavaScript to perform like another app
    - Only for web sites that can trust
  - If sensitive data was exchanged, use clearCache()

Security Best Practices – Releasing the App

- You can use Android Studio to sign your app
- Sign up as a developer (Need to pay a subscription fee).
  - https://play.google.com/apps/publish/signup/
- Go to the developer dashboard.
  - https://play.google.com/apps/publish/
- **Google App Security Improvement Program**
  - https://developer.android.com/google/play/asi
  - A good way to identify malicious third-party libraries
- **Launch Checklist**
  - https://developer.android.com/distribute/best-practices/launch/launch-checklist
  - Week 11 Tutorial