ITMC411

Security in mobile computing

LECTURE 3

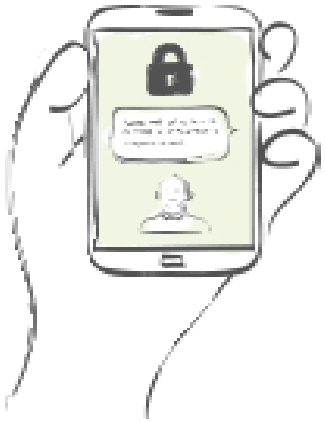
Mobile Computing Security Overview

Mobile Computing Security Overview

- **Confidentiality**, **Integrity**, and **Availability** Threats in Mobile Phones

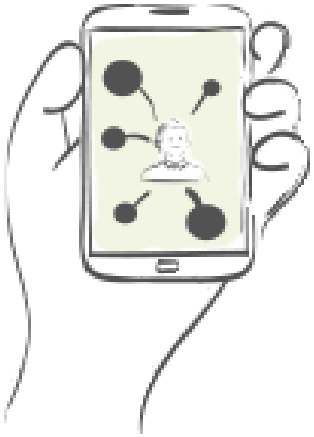
Confidentiality

sent/received/accessed data are **not read** by third parties



Integrity

detecting any intentional or unintentional changes in the **sent/received** data



Availability

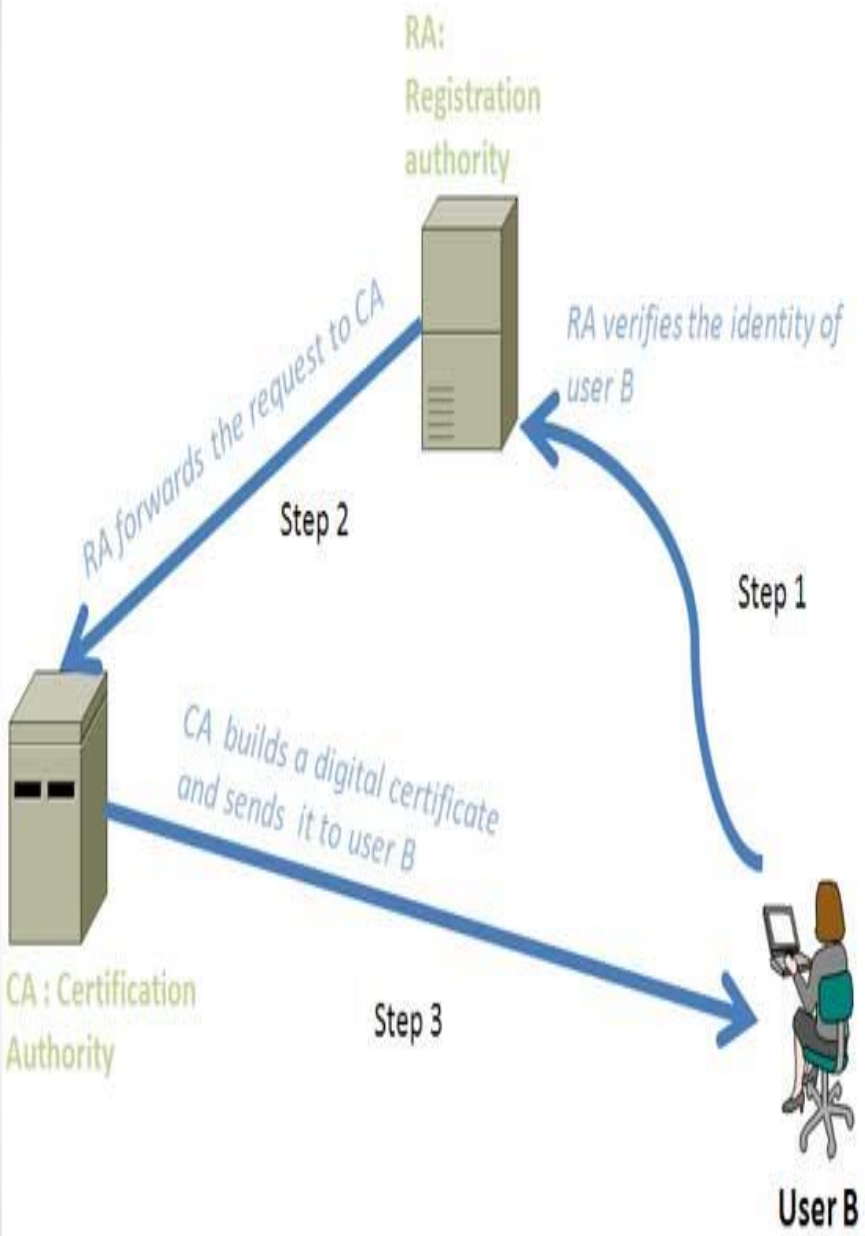
user can **access data** and resources whenever he/she needs them

Mobile Devices Threats

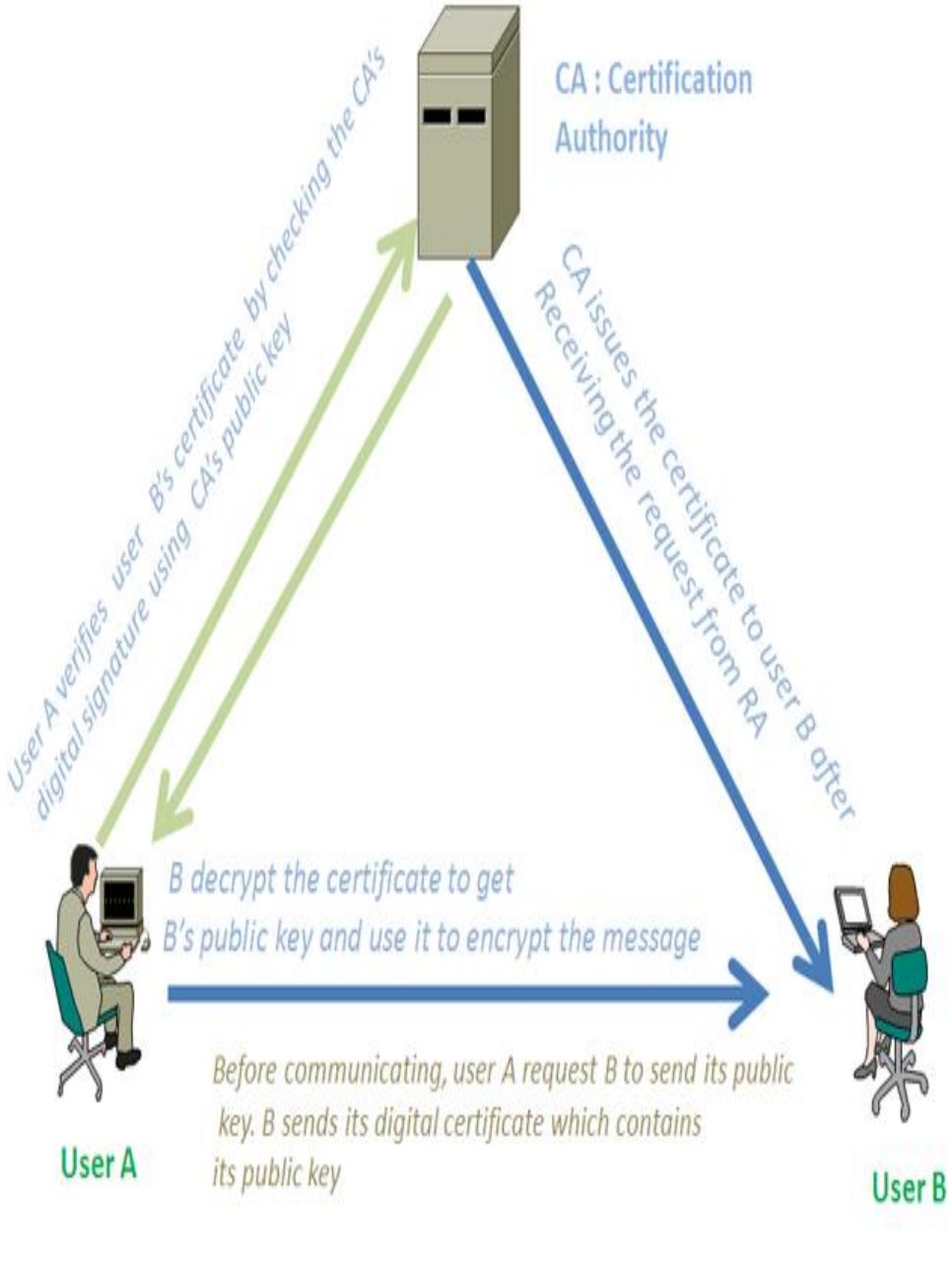
1. **Lack of control over physical security**

Protecting sensitive data by **encrypting** the data stored on the device or by **eliminating** the possibility to **store data** into the cell phone's memory.

use of **authentication** when accessing the mobile device and the resources. Most mobile devices use a simple **PIN** – but there can be implemented authentication methods based on digital certificates or domain authentication.



Digital Certificate Creation Process



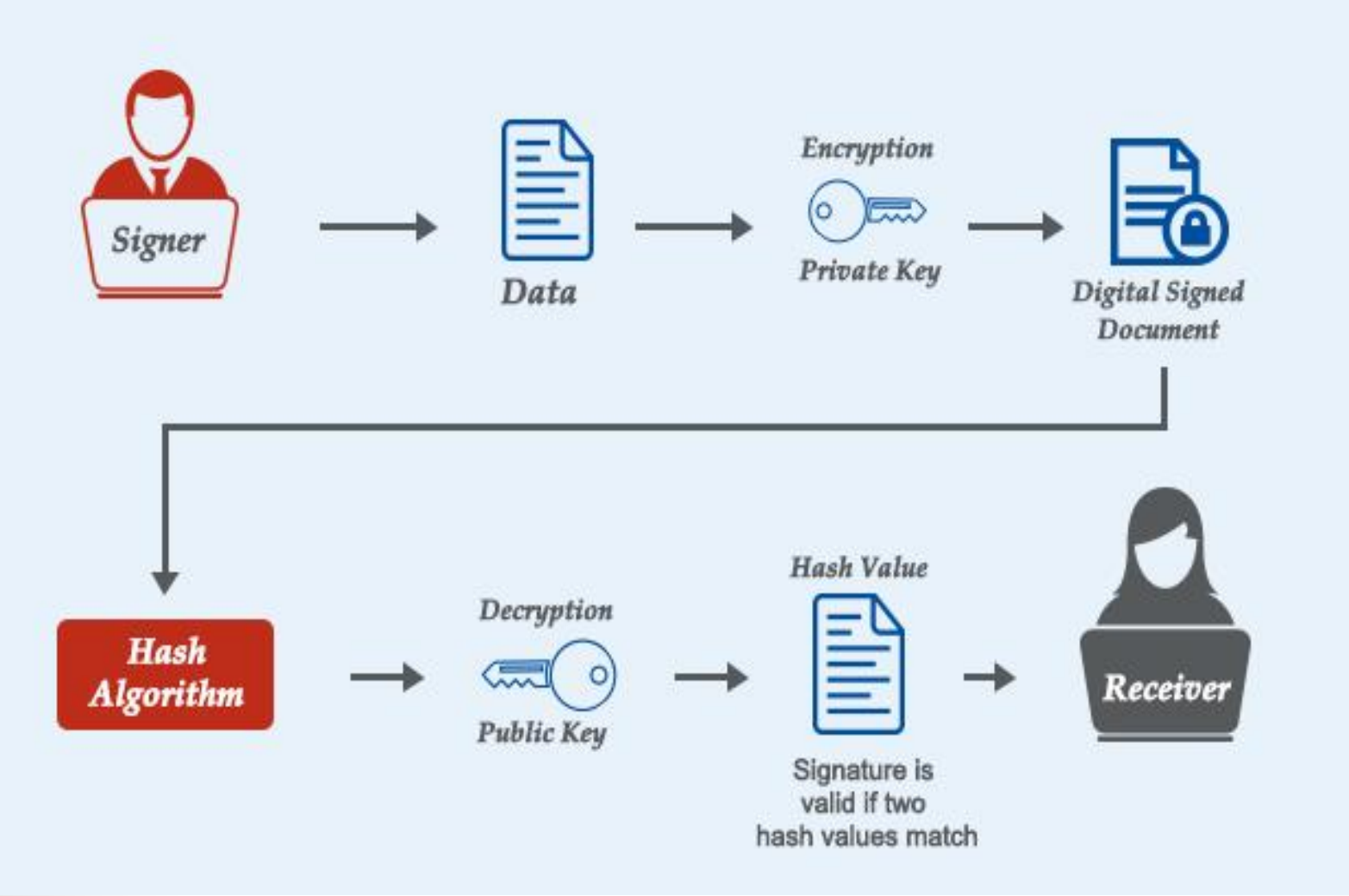
Simplified diagram: Secure communication with digital certificate

Mobile Devices Threats

2. Using unsecure networks

- These communications systems are prone to **man-in-the-middle** attacks that compromise both the communication's **integrity** and its **confidentiality**.
- The risks of using external networks are **eliminated** by **strong encryption** technologies and **digital signatures**

How does a Digital Signature Work?



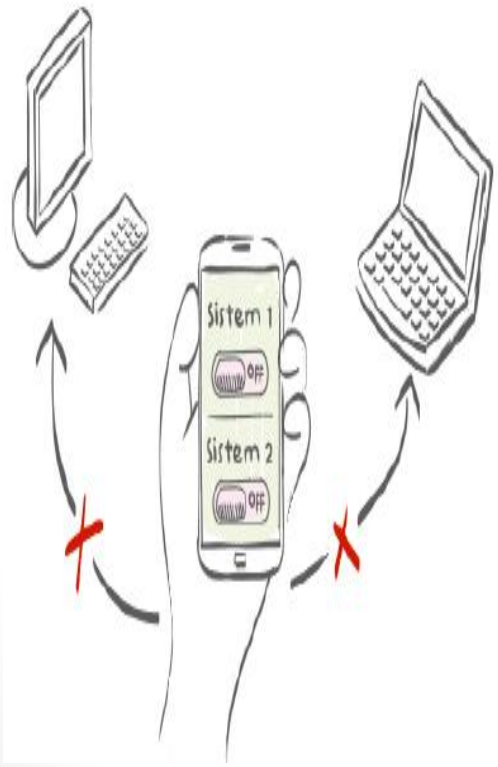
Mobile Devices Threats

3. **Using untrusted third party apps**

- These risks can be reduced in several ways, starting by **restricting access** to such **apps**.
- a different browser that includes a **secure sandbox** for accessing organizational resources can be installed, allowing the **standard browser** to be used freely.

Mobile Devices Threats

4. Integration with other systems



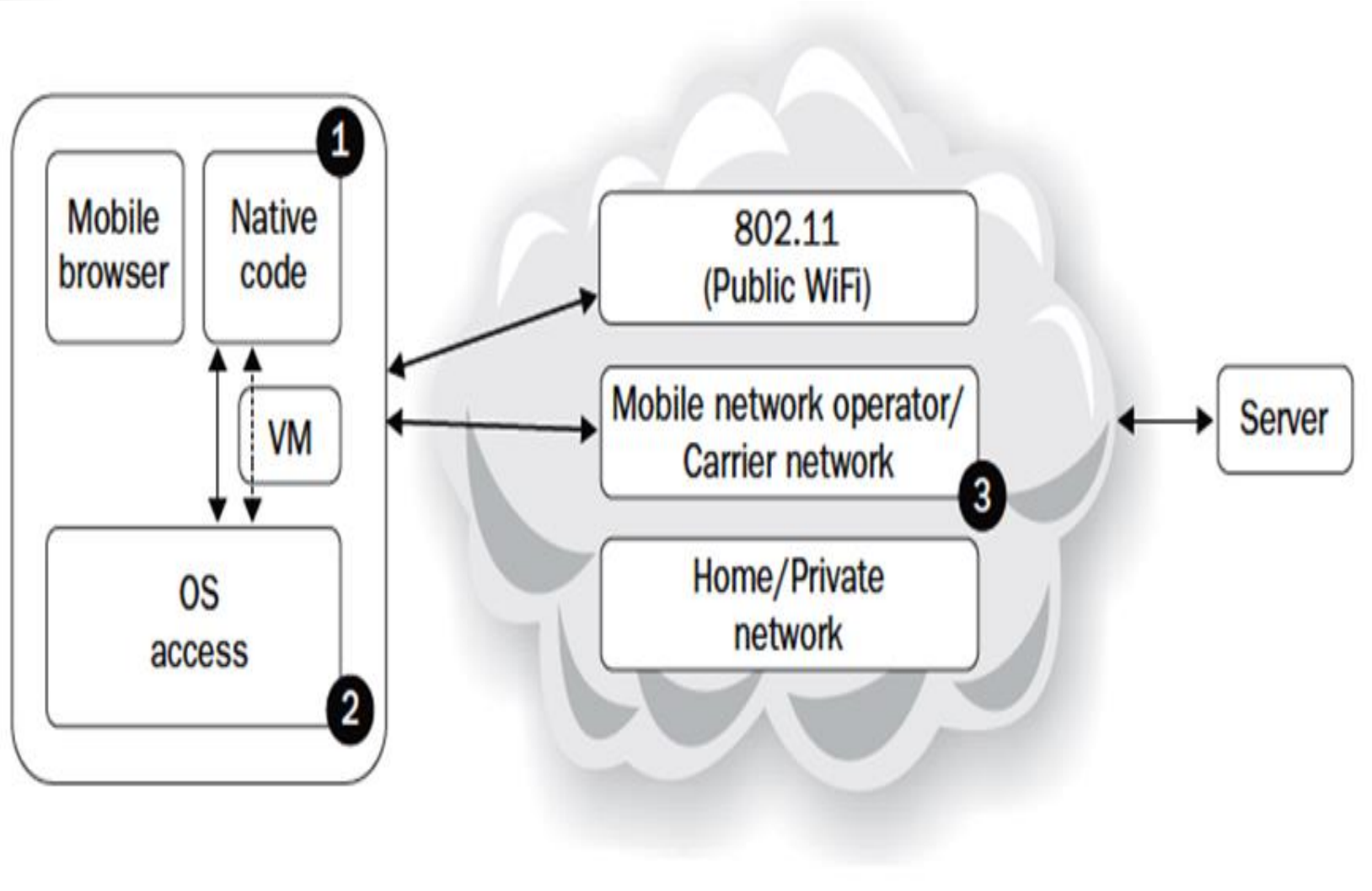
- Risks can be **avoided** simply by **disabling** the option that allows **synchronizing** a company owned mobile device with a **personal computer**.
- **eliminating** the use of **external back-up** services.

Mobile Network Architecture

The classic **3-tier architecture** that we used modified to be a mobile architecture.

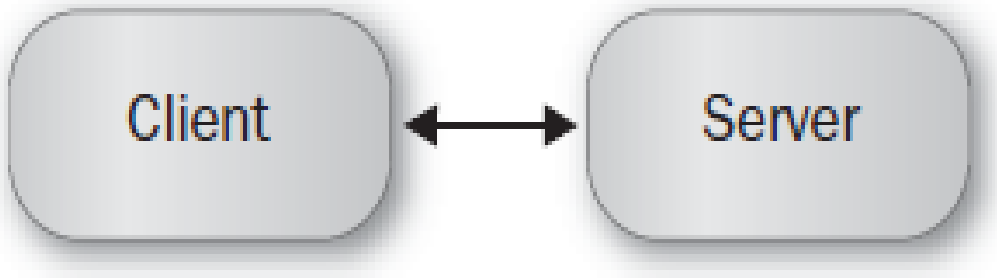
1. **Native code:** Native applications may be written in languages that execute **without the benefit of a virtual machine** or **rigorous sandbox**. These applications may be written in unsafe languages (for instance, **Objective-C**) and have increased access to other apps and resources as compared to browser based apps.
2. **OS access:** Software running in a browser has limited access to the underlying **OS**, **libraries**, **file system access**, **interprocess communication**, and **system calls**.
3. **Internet access:** mobile devices commonly use their mobile carrier's network and public **WiFi** to connect to the Internet. These means of access may provide increased opportunity for **man-in-the-middle** (**MiTM**) attacks.

Mobile Network Architecture



MOBILE RISK MODEL

- **Fundamentally** , we are still talking about a **client-server architecture**:



Risk Model

Understanding the **risk model** means first asking the **questions**:

- **Q1**: Who are the **stakeholders**? (**stakeholders**)
- **Q2**: What **items** are valuable to these stakeholders? (**assets**)
- **Q3**: What **risks** **are** **relevant** to these **assets** from each stakeholder's perspective? (**Attack Surfaces**)

Risk Model

Stakeholders

- **Mobile network operators MNO** (companies provide communication services to their clients)
- **Device manufacturers** (aka **OEMs**, hardware manufacturers, and so on)
- **Mobile operating system (OS)** vendors like **Apple** and **Google**
- **Application Store curators** (Apple, Google, Amazon, and so on)
- **Organizational IT** (corporate security's mobile device management software)
- **Mobile application developers**
- **End users**

Risk Model

Assets

- **OS manufacturer**
  - **Threats include:**
    - looks at all **applications** as a threat.
    - The **phone's user** is a threat to the OS as well; they may try to **jailbreak** the phone as soon as they get it home.
- **phone's user**
  - **Threats include**:
    - looks at the **OS** may be a threat, violating their **privacy** by capturing data and exporting it for "**statistical purposes**."
    - **Applications** preloaded by the **MNO** could be perceived similarly.

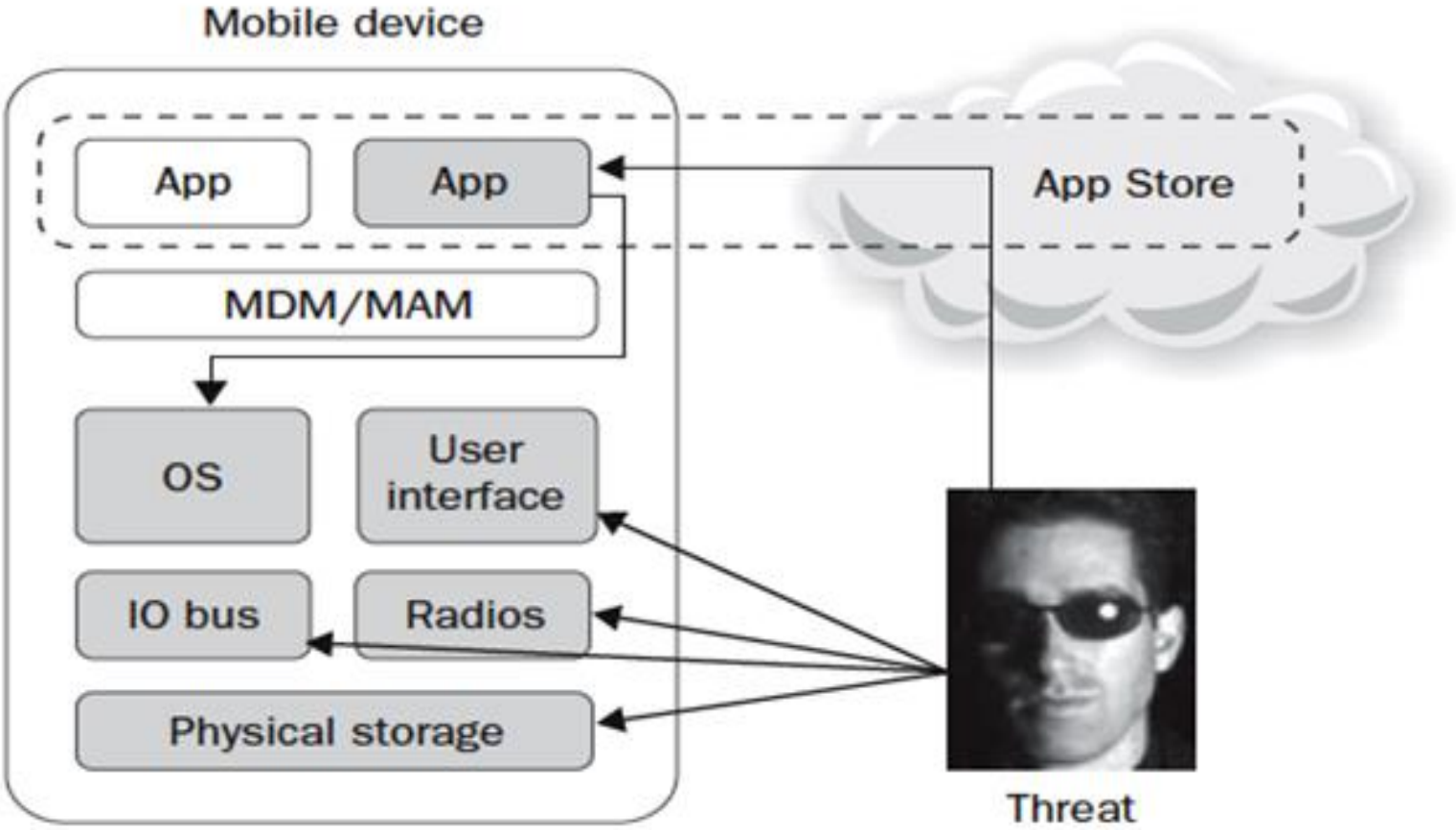
Risk Model

Attack Surfaces

- **Physical theft** allows access to the ***user interface, physical storage,*** the ***IO bus***, and the ***radios***.
- **App publication** allows the **threat** to distribute either :
  - **Trojan horse** application or other **malware**
  - The threat's app may have relaxed access to ***OS* resources**,
  - Interprocess communication.
  - **Unsandboxed environment** with which to attack its victim, depending on the state of the mobile platform (**jailbroken/rooted**),
  - **Weak** app **permission** configuration,
  - end-users' **over-permissive** settings.

Risk Model

Attack Surfaces



Security in Development Lifecycle

- Once you've **established** the **risk model**, you can design against it and more rationally adapt **downstream processes** (for example, **check implementation** using things like **code review** and **penetration testing**).