ANDROID SANDBOXING
Lecture 10a

COMPSCI 702
Security for Smart-Devices

Muhammad **Rizwan** Asghar

March 23, 2021

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

SANDBOXING

- It specifies which system resources the app is allowed to access
- It limits malicious apps to perform actions only in the sandboxed environment

TWO LEVELS OF SANDBOXING

- At the process level
  - Each app is run in a dedicated process
  - Access to sensitive resources depends on permissions
- At the filesystem level
  - Each app has its private data directory
  - Only the app can access its own data directory

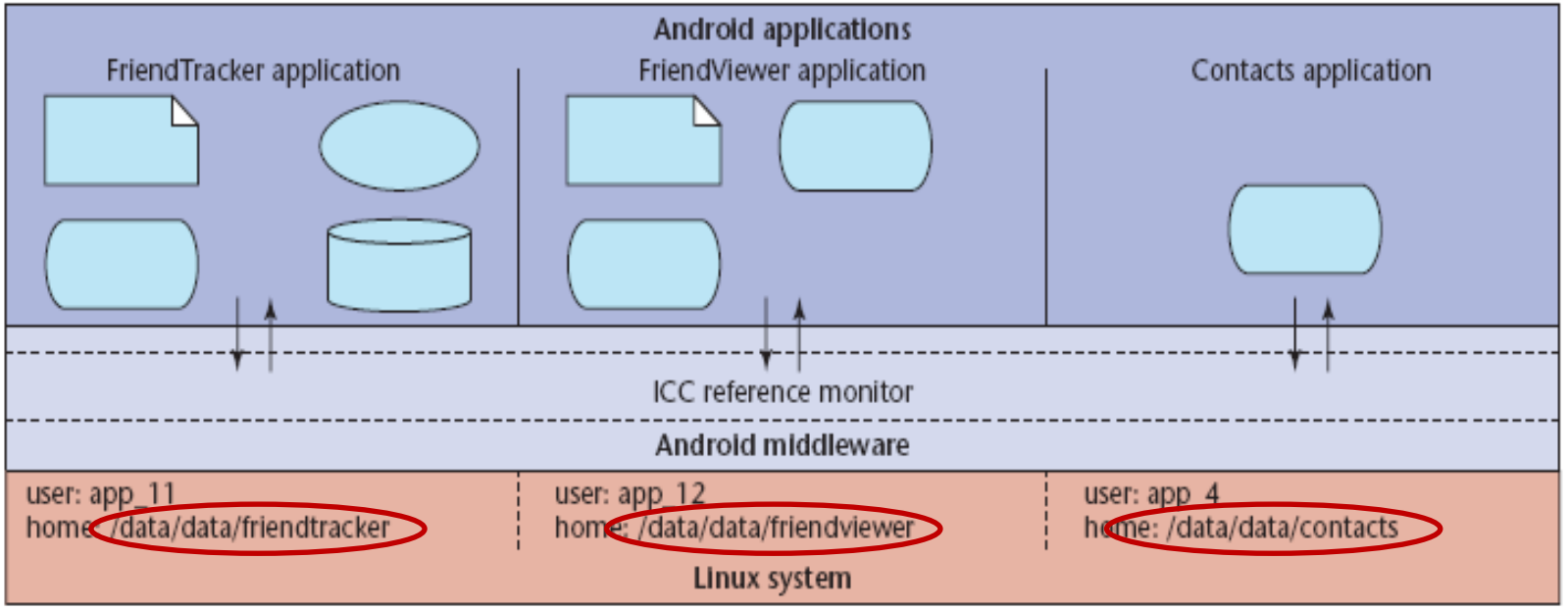
AT THE PROCESS LEVEL

- The Android system assigns a unique User ID (UID) to each Android app
- A UID is generated at install time
- A UID is often called an AppID
- It runs each app as a separate process with its UID
- Apps run within the sandboxing environment in the kernel

AT THE FILESYSTEM LEVEL

- Each app is assigned a dedicated data directory
- Only app has permission to read and write to its own data directory
- Sandboxing applies to all apps
  - Including native ones

EXAMPLE



- Each app has its own data directory

APP DATA DIRECTORY

- Implemented based on the Linux DAC
- Permissions are already set by the system
- Permissions include read, write, and execute
- rwx (user), rwx (group), rwx (others)
- These permissions can be changed

ANDROID UIDS

- System daemons and apps run under well-defined and constant UIDs
- The root uses UID 0
- System UIDs are statically defined in the *android_filesystem_config.h* header file
- UIDs for system services start from 1000
  - android.uid.phone (PHONE_UID, 1001)
  - android.uid.bluetooth (BLUETOOH_UID, 1002)
  - android.uid.log (LOG_UID, 1007)
  - android.uid.nfc (NFC_UID, 1027)

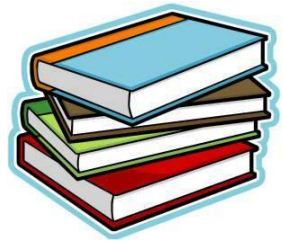
SHARED UIDS

- Apps can be installed using the same UID
- Apps can share files and even run in the same process
- Shared UIDs are used extensively by system apps, which use the same set of resources
- For example, in Android 4.4, the system UI and keyguard (lockscreen implementation) share UID 10012

THIRD PARTY APPS

- The shared UID facility is not recommended for non-system apps
- However, it is available to third party apps as well
- In order to share the same UID, apps need to be signed using the same signing key

RESOURCES



- **Chapter 1** of
  Android Security Internals: An In-Depth Guide to Android's Security Architecture
  Elenkov, Nikolay
  First Edition
  No Starch Press 2014
  ISBN:1593275811 9781593275815

Questions?

Thanks for your attention!