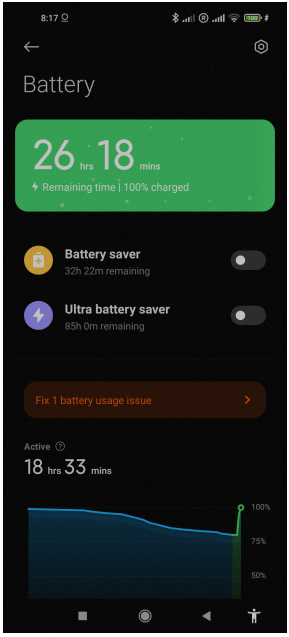
Bypassing Android isolation with fuel gauges: new risks with advanced power ICs

Vincent Giraud David Naccache
firstname.lastname@ens.fr

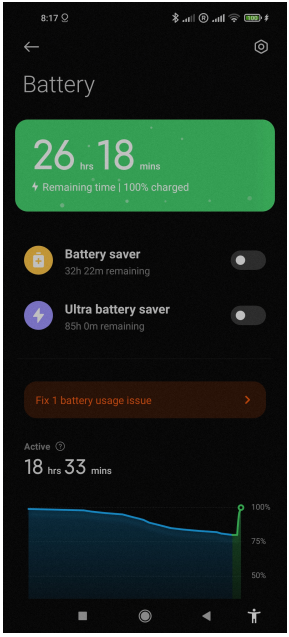
École Normale Supérieure, Université PSL, CNRS
Ingenico

These works are being conducted within the framework of a thesis supervised by David Naccache and Guillaume Bouffard.

Its objective is to investigate the feasability of sensitive and secure processes exploitation on uncontrolled environments.



Estimating the remaining battery lifetime of a device is a difficult task when only considering the **voltage** at the battery's terminals.



Estimating the remaining battery lifetime of a device is a difficult task when only considering the **voltage** at the battery's terminals.

However it is easier if we have the **operating temperature**, the **battery's age**, the **load extracted from it since it was last full**, the **lithium quality**...

Embedded power management is difficult. In particular, it requires to :

- preserve the battery health

Embedded power management is difficult. In particular, it requires to :

- preserve the battery health
- optimize the consumption

Embedded power management is difficult. In particular, it requires to :

- preserve the battery health
- optimize the consumption
- optimize the charging

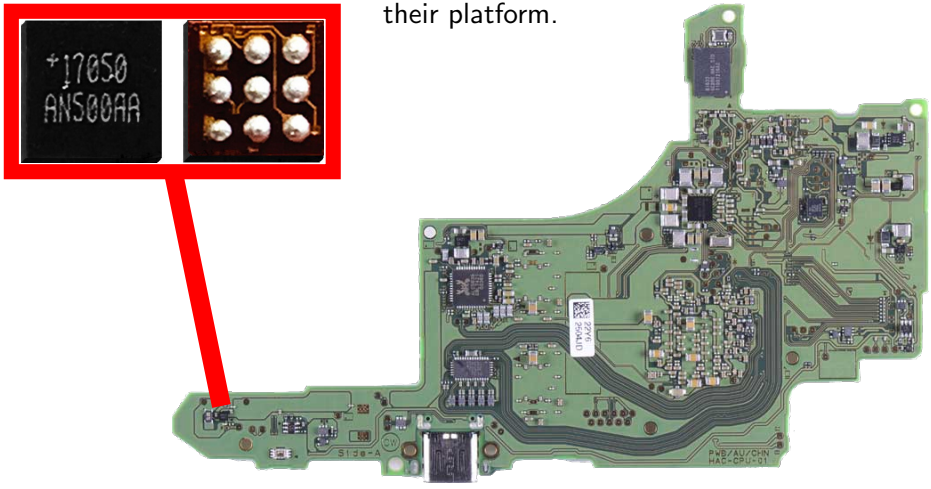
Embedded power management is difficult. In particular, it requires to :

- preserve the battery health
- optimize the consumption
- optimize the charging
- be able to operate on batteries with varying quality and ages

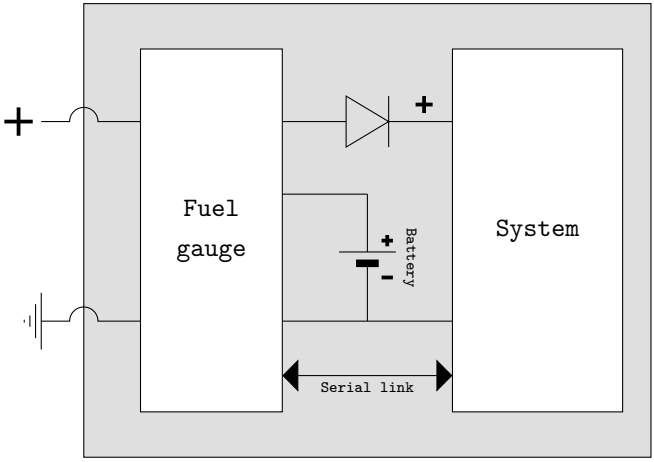
Embedded power management is difficult. In particular, it requires to :

- preserve the battery health
- optimize the consumption
- optimize the charging
- be able to operate on batteries with varying quality and ages
- manage the security on the whole circuit

To make this easier, system designers can integrate a component called “fuel gauge” in their platform.



This integrated circuit will make various measurements in real time, and will process different estimations based on them.



In most cases, the fuel gauge is placed between the main system and the power sources.

The presence of a fuel gauge in a system is never indicated on the devices' technical sheet.

After purchase, one can look for it by visually inspecting the circuit board.
On Android, one can detect them with software requests, without being *root* :

```
$ ls -a /sys/class/power_supply
battery
dc
gcpm
gcpm_pps
main-charger
maxfg
pca9468-mains
tcpm-source-psy-i2c-max77759tcpc
usb
wireless
```

Fuel gauges were only studied from a precision and performance point of view.

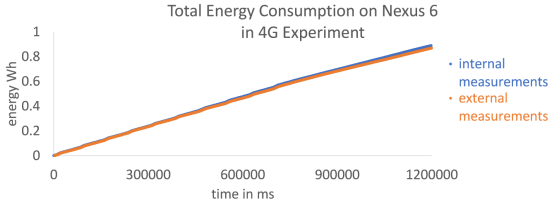


Fig. 1. 4G experiment on Nexus 6. Energy consumption is measured using internal and external energy meter.

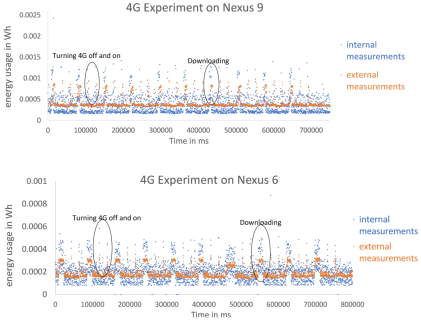


Fig. 2. Wi-Fi experiment on both devices. Energy consumption is measured using internal and external energy meter.

Bokhari M., Xia Y., Zhou B., Alexander B., and Wagner M.: *Validation of Internal Meters of Mobile Android Devices*. CoRR 2017.

Fuel gauges were only studied from a precision and performance point of view.

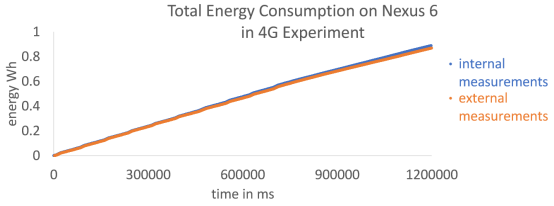


Fig. 1. 4G experiment on Nexus 6. Energy consumption is measured using internal and external energy meter.

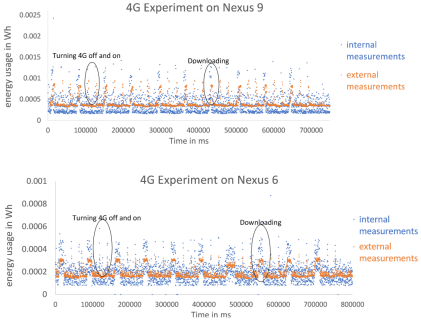
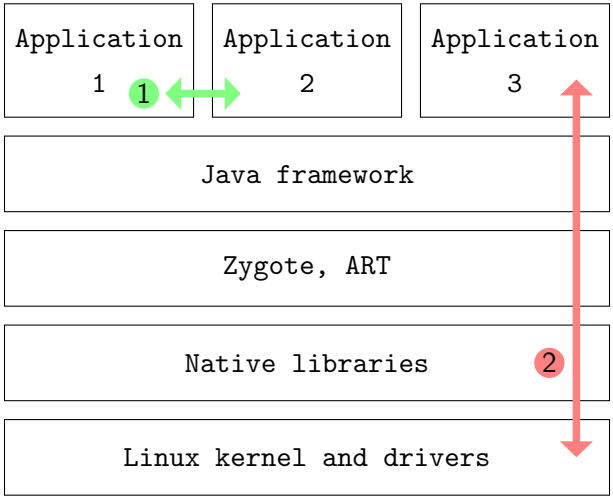


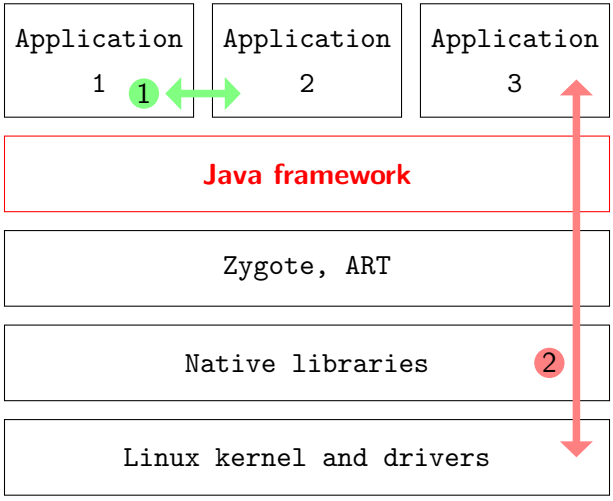
Fig. 2. Wi-Fi experiment on both devices. Energy consumption is measured using internal and external energy meter.

The security implications they have, however, were not looked into.

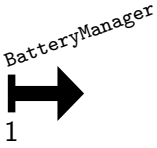
Bokhari M., Xia Y., Zhou B., Alexander B., and Wagner M.: *Validation of Internal Meters of Mobile Android Devices*. CoRR 2017.



While the Android security policy is explicit regarding horizontal interactions 1, it is less the case concerning vertical interactions 2.

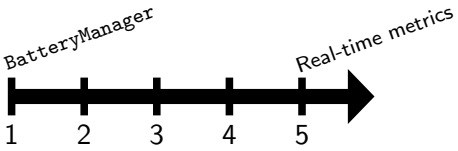


While the Android security policy is explicit regarding horizontal interactions 1, it is less the case concerning vertical interactions 2.



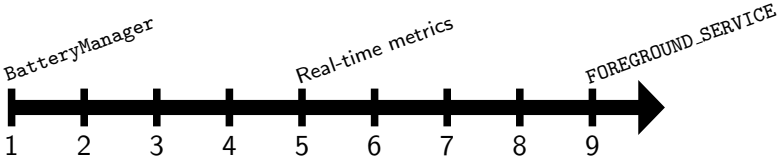
The `BatteryManager` system service exists since the beginnings of Android. At the time it only allowed to get very limited information about power, such as :

- the battery's health
- the battery's state of charge
- which external sources are present

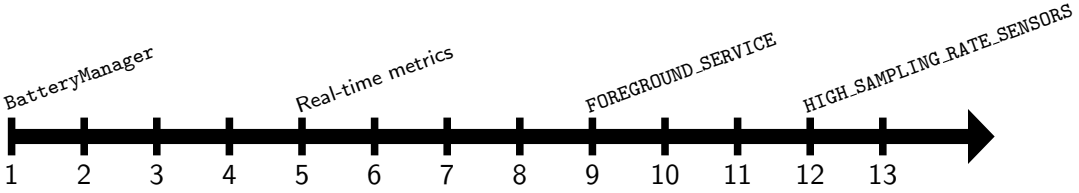


Starting with version 5, the service was able to provide real-time metrics potentially sourced from a fuel gauge, such as :

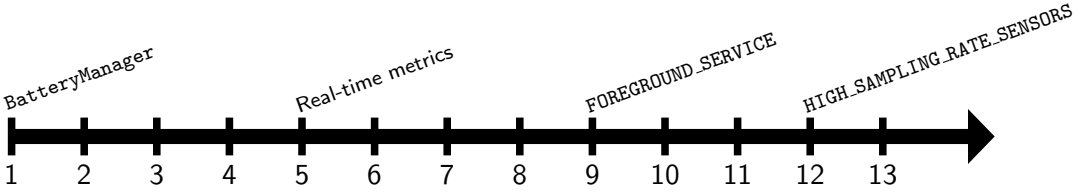
- the instantaneous current consumption in microamperes
- the remaining power in nanowatt-hours
- the fraction of remaining charge in percentage



With version 9 comes the necessity to declare a notification in order to be able to run a service in the background.



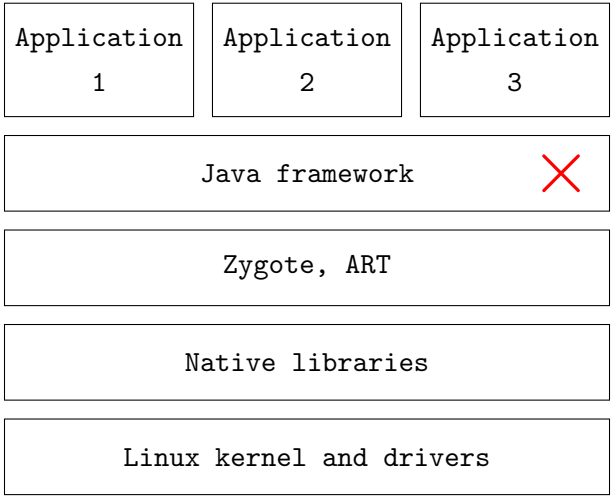
Android 12 introduces a frequency limit on the polling of embedded sensors...



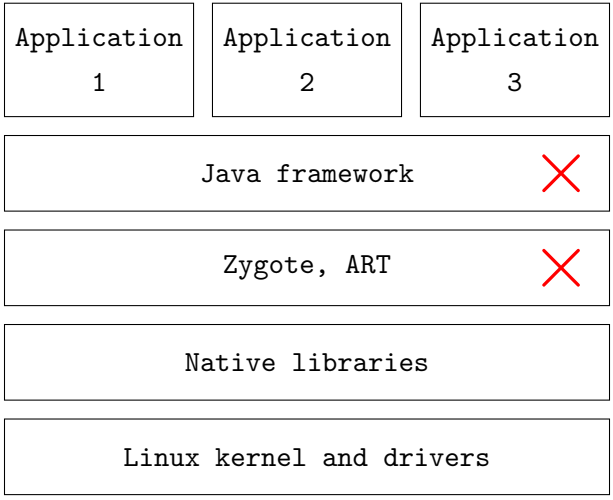
Android 12 introduces a frequency limit on the polling of embedded sensors... which is not applied to fuel gauges.

```
/**
 * Checks if a sensor should be capped according to HIGH_SAMPLING_RATE_SENSORS permission.
 * [...]
 */
private boolean isSensorInCappedSet(int sensorType) {
    return (sensorType == Sensor.TYPE_ACCELEROMETER
            || sensorType == Sensor.TYPE_ACCELEROMETER_UNCALIBRATED
            || sensorType == Sensor.TYPE_GYROSCOPE
            || sensorType == Sensor.TYPE_GYROSCOPE_UNCALIBRATED
            || sensorType == Sensor.TYPE_MAGNETIC_FIELD
            || sensorType == Sensor.TYPE_MAGNETIC_FIELD_UNCALIBRATED);
    }
```

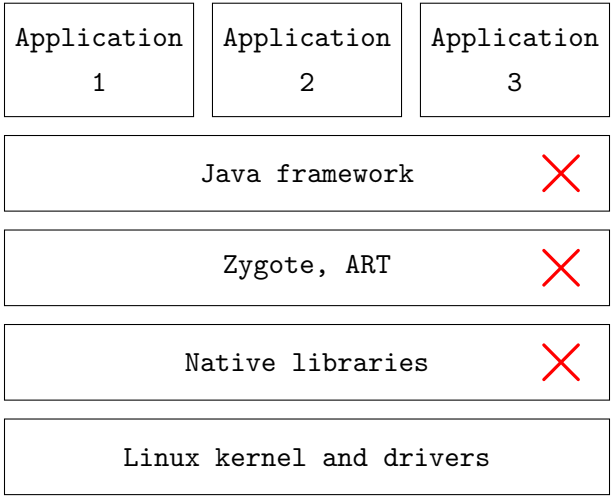
If accesses to fuel gauges are not regulated in the Java framework level, where are they ?



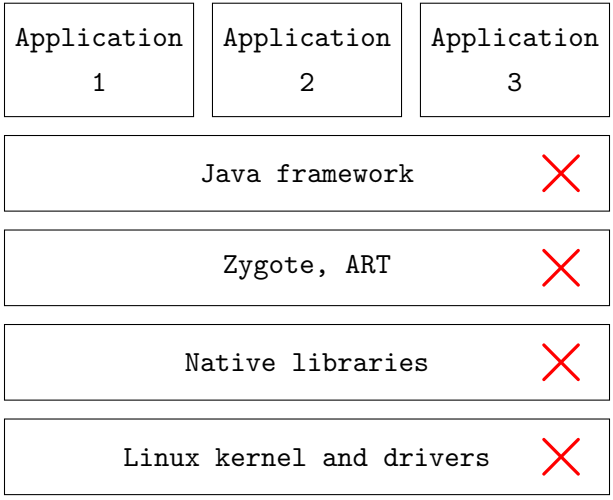
If accesses to fuel gauges are not regulated in the Java framework level, where are they ?



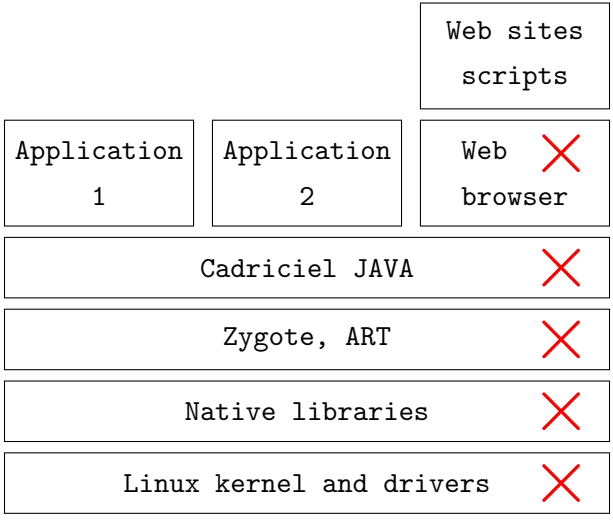
If accesses to fuel gauges are not regulated in the Java framework level, where are they ?



If accesses to fuel gauges are not regulated in the Java framework level, where are they ?



If accesses to fuel gauges are not regulated in the Java framework level, where are they ?



Thus we identify three risks :

1. the one of an activity logging done without the user's knowledge or consent

Thus we identify three risks :

1. the one of an activity logging done without the user's knowledge or consent
2. the one of a secret, unauthorized communication channel between applications

Thus we identify three risks :

1. the one of an activity logging done without the user's knowledge or consent
2. the one of a secret, unauthorized communication channel between applications
3. the one based on the spying of a sensitive process

Thus we identify three risks :

1. the one of an activity logging done without the user's knowledge or consent
2. the one of a secret, unauthorized communication channel between applications
3. the one based on the spying of a sensitive process

Research on PIN code spying is quite developed. It is often based on temporal analysis, mainly on side channels such as :

- electromagnetic emissions

Research on PIN code spying is quite developed. It is often based on temporal analysis, mainly on side channels such as :

- electromagnetic emissions
- sound

Research on PIN code spying is quite developed. It is often based on temporal analysis, mainly on side channels such as :

- electromagnetic emissions
- sound
- rotations and movements

Research on PIN code spying is quite developed. It is often based on temporal analysis, mainly on side channels such as :

- electromagnetic emissions
- sound
- rotations and movements
- the current going through a charging cable

Research on PIN code spying is quite developed. It is often based on temporal analysis, mainly on side channels such as :

- electromagnetic emissions
- sound
- rotations and movements
- the current going through a charging cable

By using fuel gauges, we intend to propose an attack vector which is not impacted by Android's security policy, is discreet, and do not requires training.

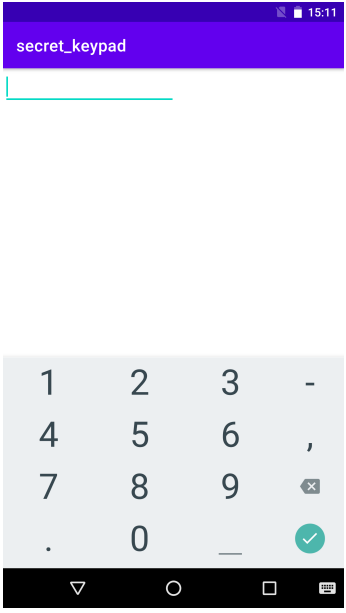
Research on PIN code spying is quite developed. It is often based on temporal analysis, mainly on side channels such as :

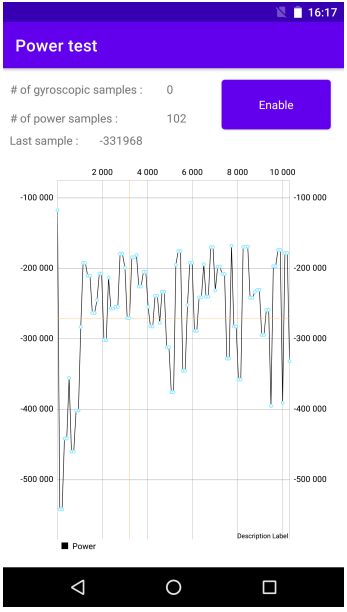
- electromagnetic emissions
- sound
- rotations and movements
- the current going through a charging cable

By using fuel gauges, we intend to propose an attack vector which is not impacted by Android's security policy, is discreet, and do not requires training.

We will consider only one metric : **the instantaneous current** going in or out of the battery.

For this proof of concept, we will consider a simple target application.



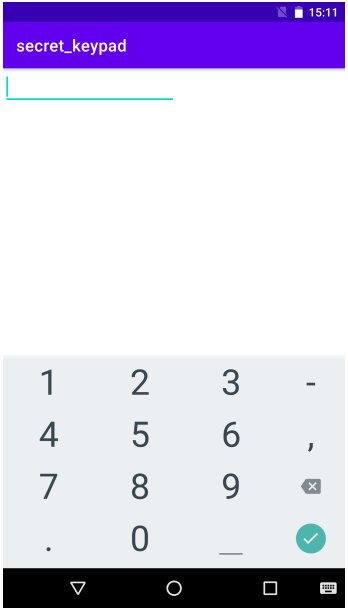


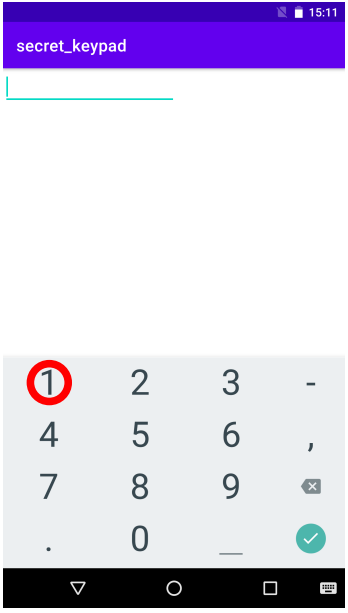
On the other end, we developed an application dedicated to the attack, which polls the consumption from the fuel gauge at the right frequency.

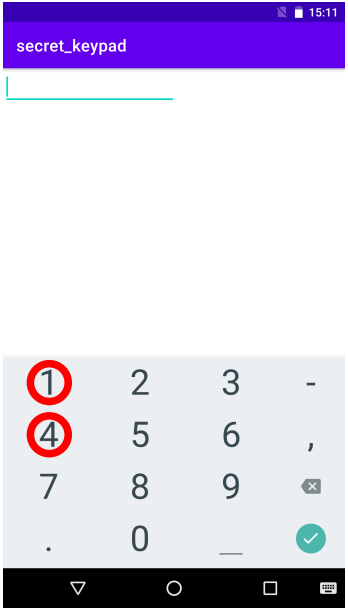
In order not to perturb the operations, we do not export the data in real-time, neither on a wired link, nor a wireless one.

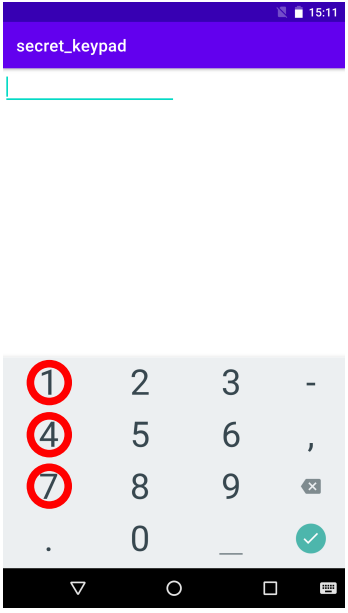
```
BatteryManager mBatteryManager =
          (BatteryManager) this.getSystemService(Context.BATTERY_SERVICE);
long courant =
          mBatteryManager.getLongProperty(BatteryManager.BATTERY_PROPERTY_CURRENT_NOW);
```

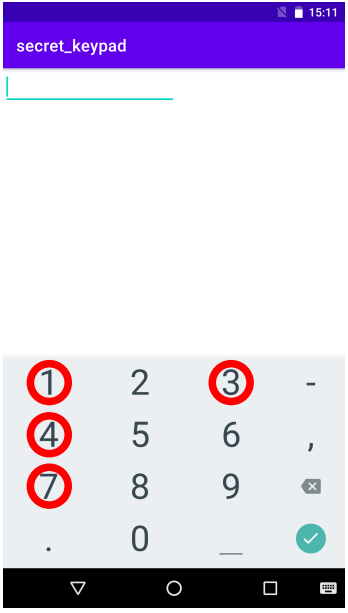
We place our polling process in an Android service so that it stays active.

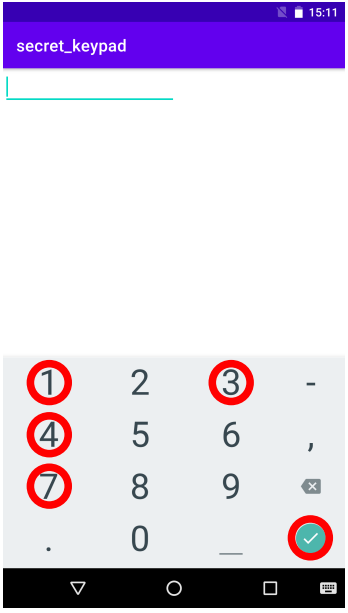

15:11
secret_keypad
1 2 3 -
4 5 6 ,
7 8 9
. 0

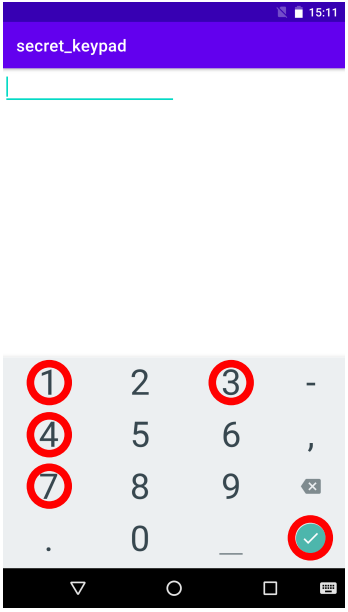

15:11
secret_keypad
1 2 3 -
4 5 6 ,
7 8 9
. 0 _

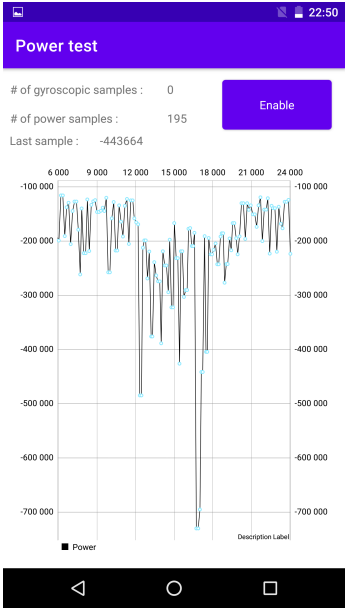

15:11
secret_keypad
1
2
3
-
4
5
6
,
7
8
9
.
0

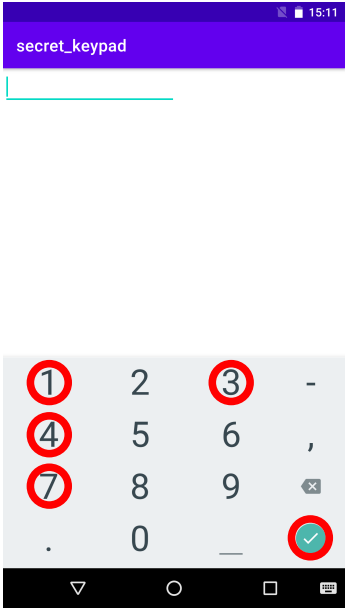

15:11
secret_keypad
1
2
3
-
4
5
6
,
7
8
9
.
0

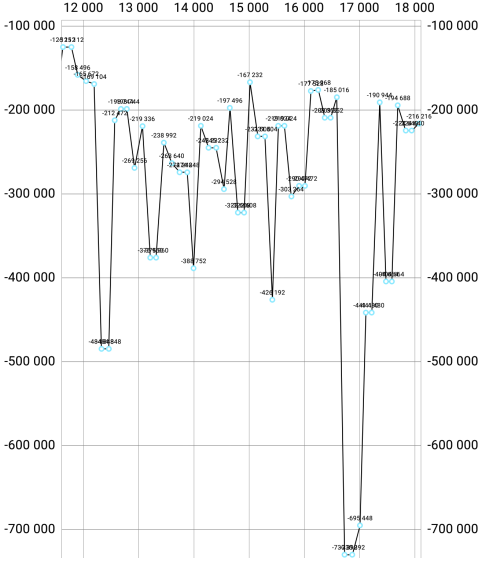

15:11
secret_keypad
1
2
3
-
4
5
6
,
7
8
9
.
0

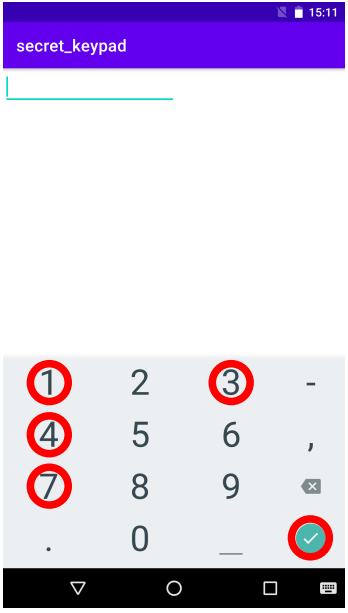

15:11
secret_keypad
1
2
3
-
4
5
6
,
7
8
9
.
0

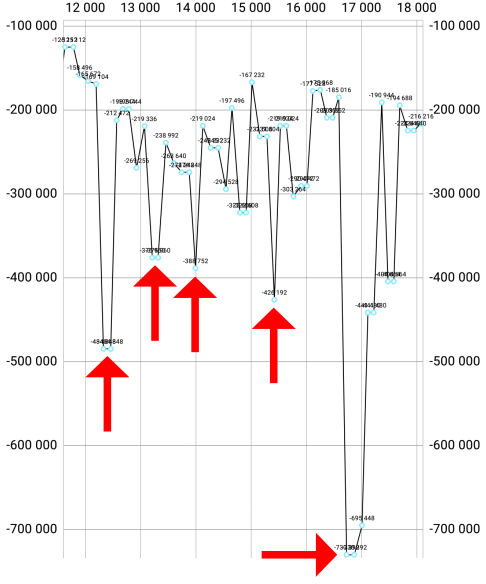

15:11
secret_keypad
1 2 3 -
4 5 6 ,
7 8 9
. 0 _

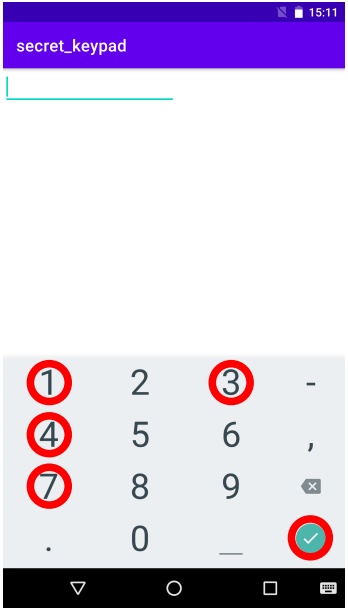

22:50
Power test
# of gyroscopic samples : 0
# of power samples : 195
Last sample : -443664
Enable
6 000 9 000 12 000 15 000 18 000 21 000 24 000
-100 000 -200 000 -300 000 -400 000 -500 000 -600 000 -700 000
Description Label
Power

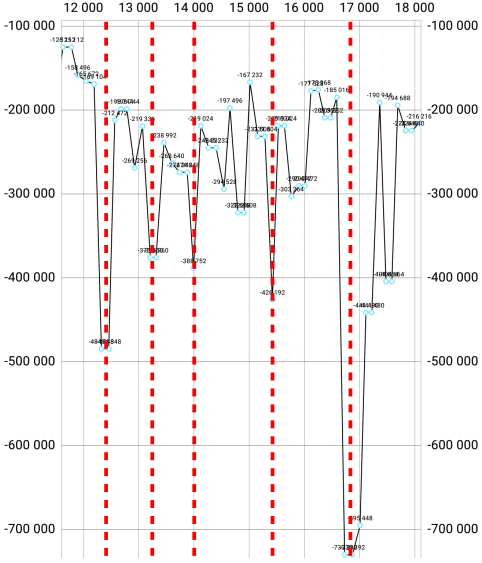

15:11
secret_keypad
1
2
3
-
4
5
6
,
7
8
9
.
0

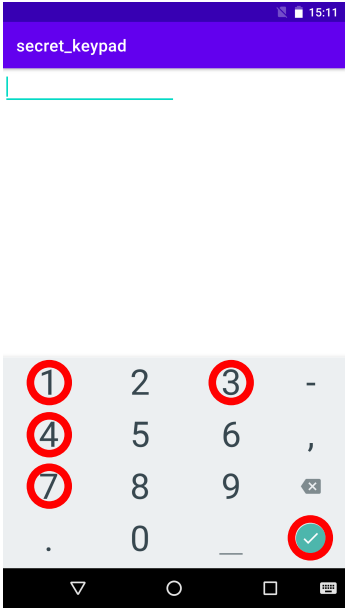

12 000
13 000
14 000
15 000
16 000
17 000
18 000
-100 000
-200 000
-300 000
-400 000
-500 000
-600 000
-700 000

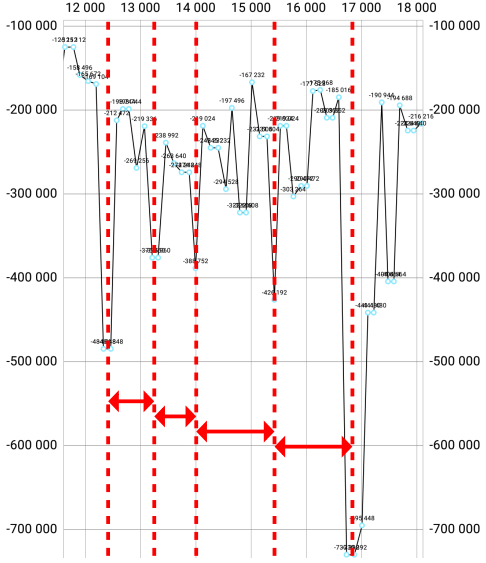

15:11
secret_keypad
1
2
3
-
4
5
6
,
7
8
9
.
0

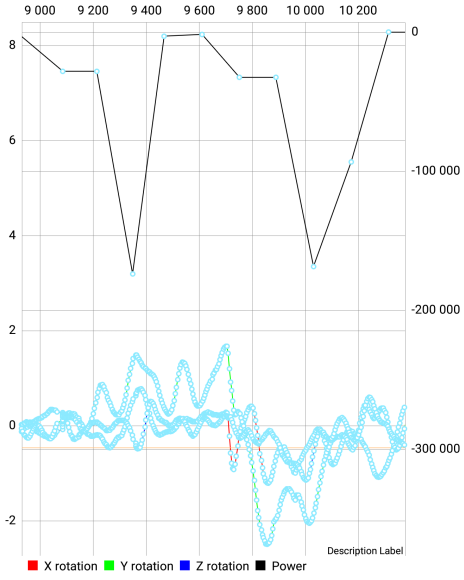

12 000
13 000
14 000
15 000
16 000
17 000
18 000
-100 000
-200 000
-300 000
-400 000
-500 000
-600 000
-700 000


15:11
secret_keypad
1
2
3
-
4
5
6
,
7
8
9
.
0

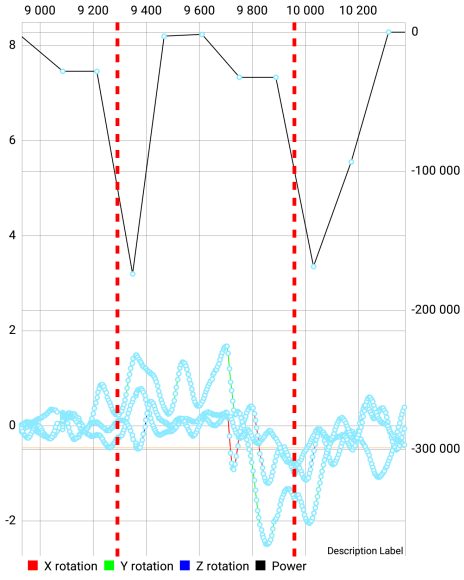

12 000
13 000
14 000
15 000
16 000
17 000
18 000
-100 000
-200 000
-300 000
-400 000
-500 000
-600 000
-700 000


15:11
secret_keypad
1
2
3
-
4
5
6
,
7
8
9
.
0


12 000
13 000
14 000
15 000
16 000
17 000
18 000
-100 000
-200 000
-300 000
-400 000
-500 000
-600 000
-700 000



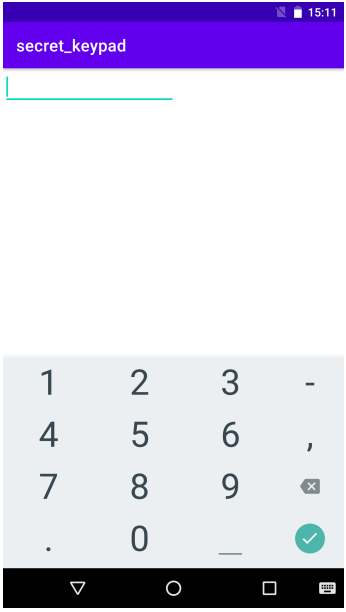
If needed, we can refine the temporal positioning by using data from the gyroscope.



If needed, we can refine the temporal positioning by using data from the gyroscope.

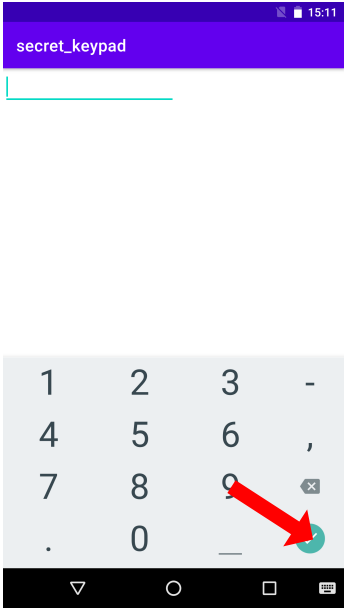
Once the data has been collected, one can apply state-of-the-art techniques.

Here, we will consider the deterministic developement of a tree containing the most possible codes.



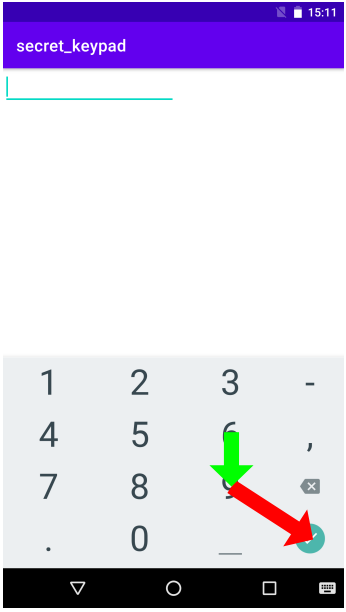
Once the data has been collected, one can apply state-of-the-art techniques.

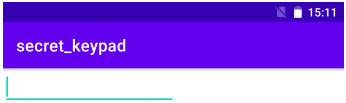
Here, we will consider the deterministic developement of a tree containing the most possible codes.



Once the data has been collected, one can apply state-of-the-art techniques.

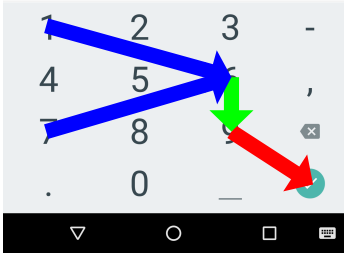
Here, we will consider the deterministic developement of a tree containing the most possible codes.

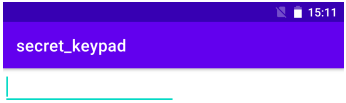




Once the data has been collected, one can apply state-of-the-art techniques.

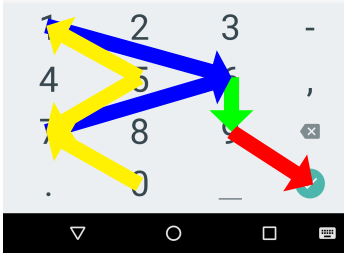
Here, we will consider the deterministic developement of a tree containing the most possible codes.





Once the data has been collected, one can apply state-of-the-art techniques.

Here, we will consider the deterministic developement of a tree containing the most possible codes.



Capture demonstration

Recursive script demonstration

The possibilities shown here compromise the confidentiality of Android platforms.

When developping sensitive solutions intended for these environments, **one should not rely on a perfect sofware isolation**, even on a device with root rights still locked.

Removing this attack vector is easy if we control the system.

One possible action is to patch the battery manager at the Java framework level.

Combatting this kind of attacks is more difficult when a developer only has access to the application layer.

Among the currently studied options, we are considering techniques based on jamming, or sensitive signals simulation.

In conclusion :

- while fuel gauges are useful for system designers, their integrations require caution

In conclusion :

- while fuel gauges are useful for system designers, their integrations require caution
- third-party developers should not completely rely on Android's isolation guarantees

In conclusion :

- while fuel gauges are useful for system designers, their integrations require caution
- third-party developers should not completely rely on Android's isolation guarantees
- the other aforementioned risks are under study

In conclusion :

- while fuel gauges are useful for system designers, their integrations require caution
- third-party developers should not completely rely on Android's isolation guarantees
- the other aforementioned risks are under study
- and protections at the application level only are, too

In conclusion :

- while fuel gauges are useful for system designers, their integrations require caution
- third-party developers should not completely rely on Android's isolation guarantees
- the other aforementioned risks are under study
- and protections at the application level only are, too

Thank you to Guillaume Bouffard from the National Cybersecurity Agency of France (ANSSI) for his support in these works.