# SecuritySentinel v1 - Android & iOS

This is an application that run in background and send some information to an API to identify security weaknesses on the mobile.

This app must be able to **intercept application installation** to send the application to the API and analyze it.

It must also gather installed applications information and system information (like **Wifi details, and detect if the phone is rooted**), this information is sent to another API that will analyze this data.

This app does not need user control, only some view to show some information about the data:

* **No user login/auth**
* **No ads**
* **No app submission**
* **No API to develop, only integrate the API described in the doc (api.txt)**

## Details of the features

**The backed API is provided, only the mobile side must be developed**, no user login required. There are 2 API endpoints, JSON format for request and reply:

* Analysis API, packages are sent, a reply contains an analysis status (pending, processing, done), and an action (allowed or blocked)
* Heartbeat API, all other information described below. The heartbeat is sent periodically, information is check before sending the heartbeat.
* Package analysis:

When an app in installed, the package is sent to an API (I provide the backend API) to analyze it.

A notification is shown when a malicious package is installed.

The analysis results are shown on a dedicated page in the app as a list:

* Package name
* Package provider (if possible)
* Date of analysis
* Analysis result
* Vulnerability analysis:

Periodically, the version of the OS, the libs and installed packages are sent to an API (I provide the backend API) to analyze these information and check if there are some known vulnerabilities.

A notification is shown when a new vulnerability is detected.

Vulnerabilities are shown on a dedicated page in the app as a list:

* Impacted program with version
* Vulnerability name
* Vulnerability release date
* Vulnerability score

(don’t keep the history of vulnerability, each API reply should overwrite the current vulnerability list)

* Environment analysis:

Some extra information are shown on a dedicated page called “Environment”, v1 features :

* Rooted device:

Periodically, check if the device is rooted. The information is sent to an API

A notification is shown when a new vulnerability is detected.

* Unsafe Wifi:

Periodically check if the Wifi is not secured, add the information in the heartbeat message (Wifi name, unsafe details)

* + - A captive portal is used
    - Encryption algorithm is weak

The app must be authenticated on the server API to communicate, this is done with the API heartbeat call.

To send a heartbeat, the server URL , port and key (like a password) must be setup in the App. In addition, a uniq identifier (generated based on phone hardware) must be sent to identify the device.

A settings page must show a form to setup the server and the key.

In order to facilitate the setup process, the url:port and key can be completed automatically by scanning a QR code (in the doc design.pptx)