**On Lightweight Mobile Phone Application Certification**

Shaked Gofin 316407576

Sean Peer 313581613

The feature set that the paper uses is a set of 1034 permissions such as:

ENABLE\_NOTIFICATION

ACCESS\_MOCK\_LOCATION

DEVICE\_ADMIN\_ENABLED

MICROPHONE

RECEIVE\_DEVICE\_ADMIN\_STATUS

ACCESS\_FIND\_LOCATION

READ\_LOGS

ANSWER\_PHONE

READ\_SMS

CAPTURE\_SECURE\_VIDEO\_OUTPUT

CAMERA

ACCESS\_GOOGLE\_PASSWORD

Running the classifier video

The type of classifier that we analyze is static

GitHub link: <https://github.com/SeanPeer/When-the-guard-failed-the-droid>

Research paper: [On Lightweight Mobile Phone Application Certification](https://www.enck.org/pubs/ccs09-enck.pdf)

**Second Assignment - Weak Spots**

After examination of our model, we have found that the Kirin model looks for malicious permissions from a predefined set of permissions.

This practice forces the model to always be up to date in order for it to have a list of all the malicious permissions that exist. as new technology is constantly being added to phones and with them new permissions of activation.

This is a great vulnerability as we can present the model with a data set with new malicious permissions that are unknown to the model.

Moreover, we can present a combination of malicious permissions that are not part of the pre-defined set of malicious features.

For example, with the permissions:

* ACCESS\_FIND\_LOCATION
* FILE\_ACCESS
* MANAGE\_EXTERNAL\_STORAGE (grants Read and write access to all files within shared storage)
* [NEARBY\_WIFI\_DEVICES](https://developer.android.com/reference/android/Manifest.permission#NEARBY_WIFI_DEVICES) (new permission on Android 13 - API level 33)

An attacker can access nearby Wi-Fi devices and transfer valuable user information as it has permission to access files on the device, and because those features are not part of the pre-defined malicious features they will not be flagged as such.

Another vulnerability in the Kirin model is having API's that don’t check for permissions as prescribed in the documentation and thus bypass the Kirin permission check altogether.

In the next report we will suggest an attack on the model using the MB (Manifest Based Attack) because we as the attackers have access to the training dataset and can manipulate the permissions to bypass the model defenses, since we know the set of malicious permissions that the model is looking for and can manipulate the model to misclassify the apps.