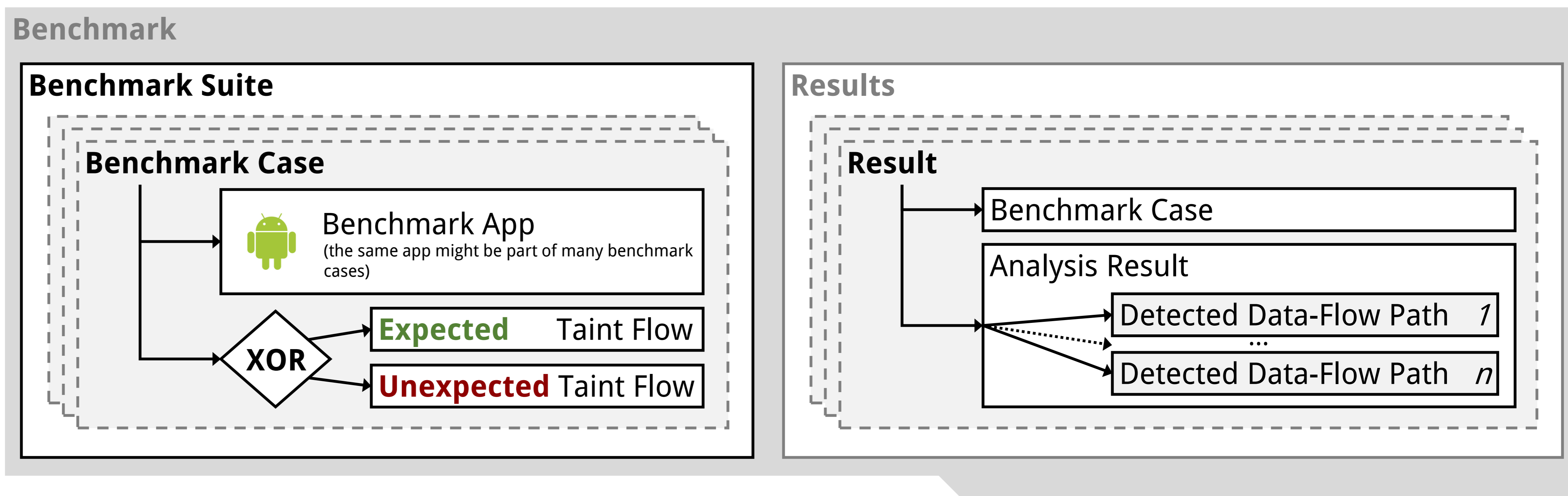


### Use it. Profit from it. Contribute to it. Now!

All artifacts available!

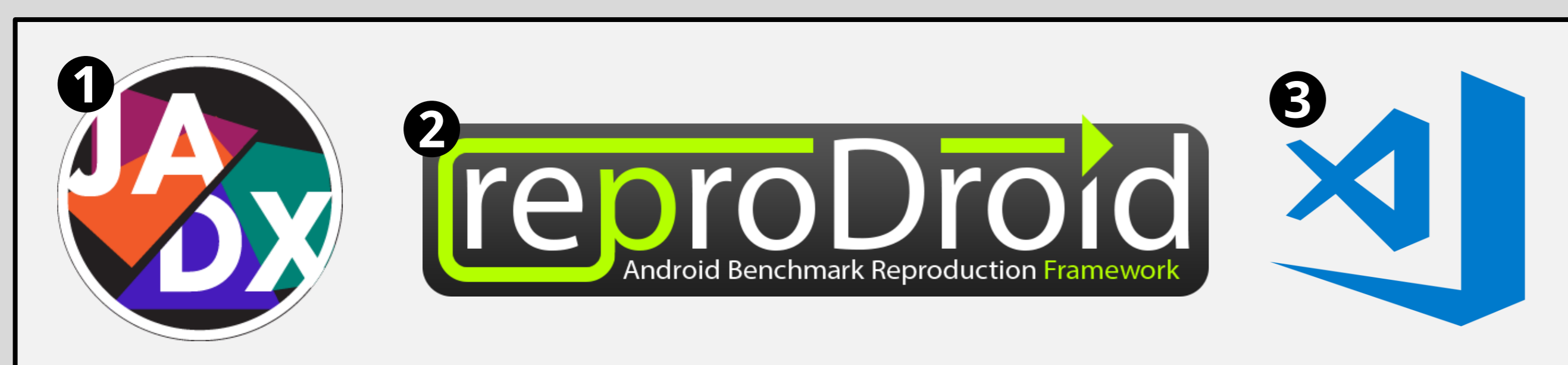
### 1 Benchmark Definition

The figure on the right depicts how benchmarks are structured. To evaluate a benchmark, we count true and false positives and negatives (TP/FP/TN/FN). These are then used to compute precision, recall and F-measure. Example: If an (un-)expected taint flow matches a detected data-flow path, it is a TP (FP).

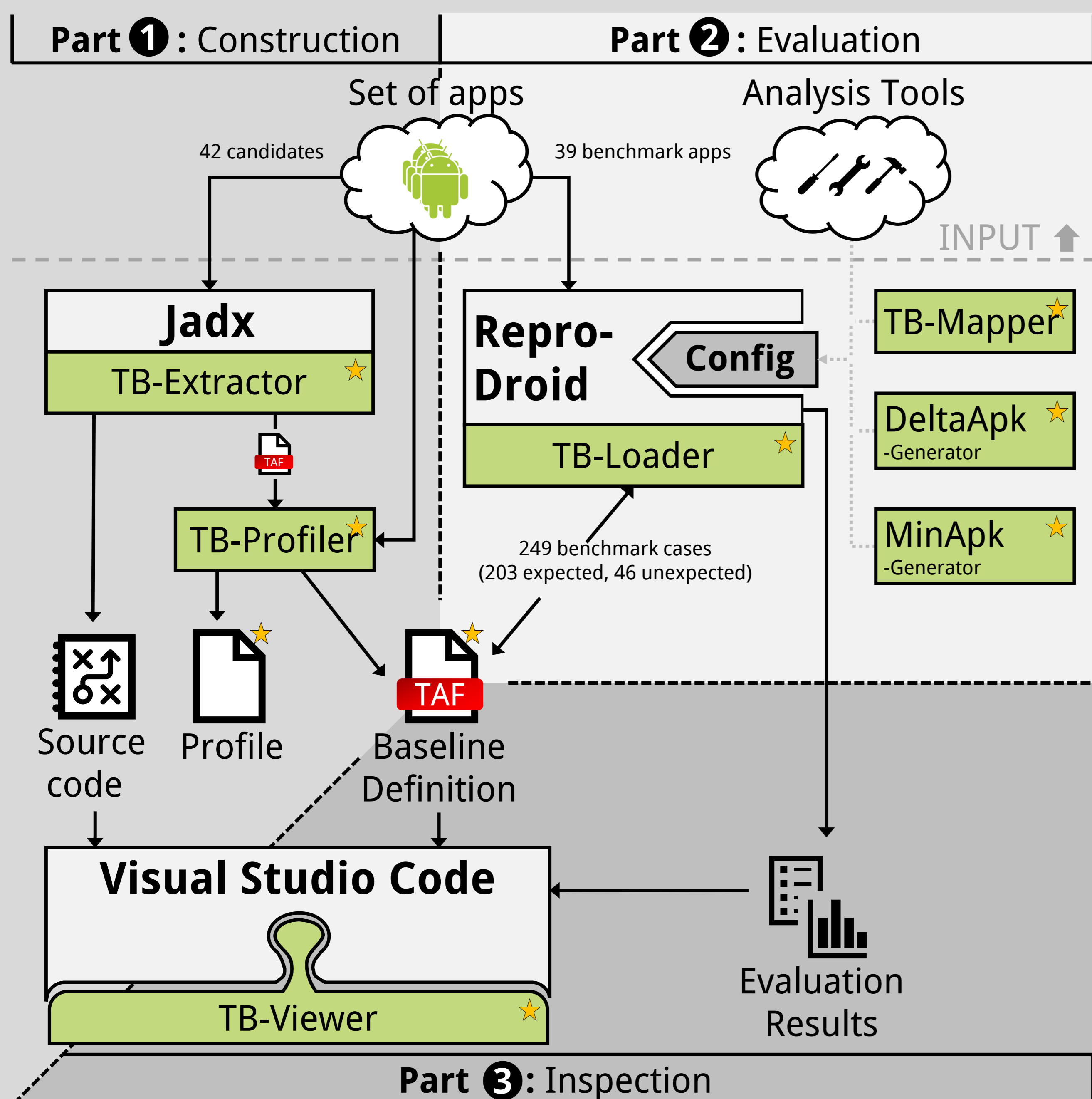


### 2 TaintBench Framework

The TaintBench Framework consists of three parts that handle different tasks. For the 1<sup>st</sup>, 2<sup>nd</sup> and 3<sup>rd</sup> part we extended JadX, ReproDroid and VSC respectively.



The figure below shows the entities of each part.



### 3 TaintBench Suite

[TaintBench.github.io](https://github.com/TaintBench)

The Taint Analysis Benchmark Format (TAF) is used to transfer information between these parts. The complete suite consists of:

- 39 Benchmark Apps
- 249 Benchmark Cases (203 expected, 46 unexpected)

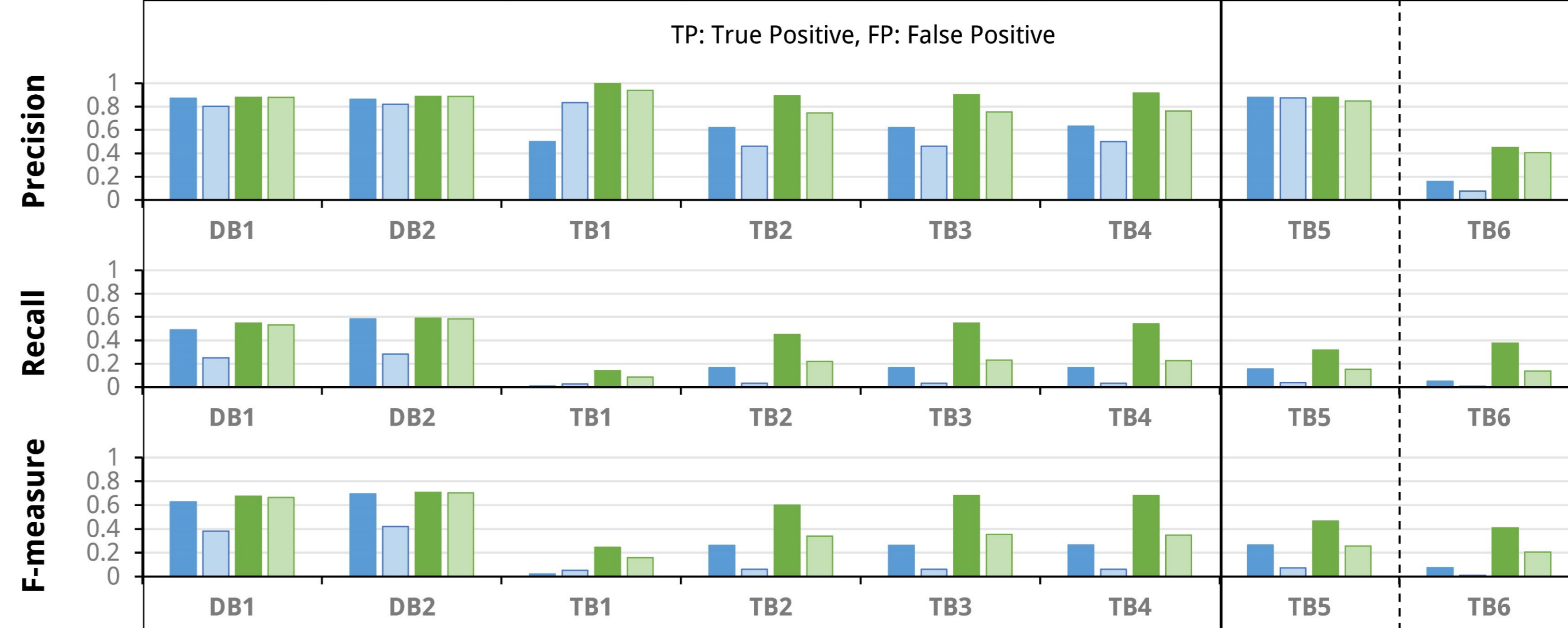
```

{ "findings": [{
  "ID": 1,
  "isNegative": false,
  "description": "This malicious flow leaks IMEI via SMS.",
  "source": {
    "statement": "String s = getIMEI();",
    "methodName": "onCreate",
    "className": "MainActivity",
    "lineNo": 1,
    "targetName": "getIMEI",
    "targetNo": 1,
    "IRs": [{"type": "Jimple", "IRstatement": "$r2 = virtualinvoke ..."}]},
  "sink": {
    "statement": "sendMessage(Logger.imei);", ...},
  "intermediateFlows": [
    {"ID": 1, "statement": "s = \"IMEI: \" + s;", ...},
    {"ID": 2, "statement": "Logger.imei = s;", ...}],
  "attributes": {
    "staticField": true,
    "appendToString": true,
  },
}, { ... } ]...}
    
```



DroidBench				TaintBench																											
Expected Cases: 163		Unexpected Cases: 41		Expected Cases: 186		Unexpected Cases: 35		DB1		DB2		TB1		TB2		TB3		TB4		TB5		TB6									
TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP								
80	12	95	15	2	2	31	19	31	19	31	18	29	4	8	42	41	10	46	10	5	1	6	7	6	7	6	6	7	1	1	12
89	12	96	12	26	0	84	10	102	11	101	9	59	8	51	62	87	12	95	12	16	1	41	14	43	14	42	13	28	5	23	34

- Amandroid
- Amandroid\*
- FlowDroid
- FlowDroid\*



### 4 Evaluation Results

Two versions of Amandroid and FlowDroid were employed during evaluation – the newer version is always marked by \*. As benchmarks the micro benchmark DroidBench (DB) and TaintBench (TB) have been used and compared.

Overall, six experiments were conducted:

- Default Sources & Sinks
- Suite-Level Sources & Sinks
- App-Level Sources & Sinks
- Case-Level Sources & Sinks
- Minified Apps
- Delta Apps



Over-Adaption: Tools are less accurate on TaintBench than on DroidBench  
 Regressions: Newer tool versions are less accurate