APPENDIX

Algorithm 1 ($p2$) shows the steps for benchmarking individually each of the 86 JS features. It receives as input a JS file (JS); a file containing JS interpreter's features with the values to test (*file with features*); the performance metric values measured by executing JS using *Duktape* default features (JSI); and the number of runs. CPU time and memory usage may vary between runs due to CPU and operating system schedulers, we set $runs = 10$ to control for random errors. For sake of simplicity, we do not present $p1$ and $p3$ because they are similar to $p2$. All the scripts used in this study are in the replication package [30].

A row in *file with features* (Line 2) is a pair {feature name, test value}. Lines 3 to 4 use *Duktape* configuration script to generate a new JS interpreter $mJSI$ using a configuration file (*cfg*) generated from row. For example, row `DUK_USE_ARRAY_BUILTIN, FALSE` in *cfg* generates a new interpreter without array built-in. If the generation process fails, then the script selects the next row. The generation fails if a feature required to run JS is deactivated or required other features. For example, storing strings in the ROM of the target device requires four activated features. Line 8 measures the size of the compiled interpreter ($codeSize\ mJSI$) using the Linux command `stat`, which reports the number of bytes of the $mJSI$. Line 9 computes the impact of the selected feature compared to their default values. Line 10 stores the performance metrics values of the multiple runs in a file. $codeSize\ mJSI$ does not change between runs so we measure it only once. Lines 11 to 18 benchmark JS using Linux *mallinfo* command to measure memory usage and `/usr/bin/time` to measure CPU time and report the total number of seconds that the process spent in user mode. The output is a CSV file with the percentage change (δ) of each feature ($p \in P$), defined in Equation (5):

$$\delta(p) = \frac{\text{median}(p(mJSI)) - \text{median}(p(JSI))}{\text{median}(p(JSI))} \quad (6)$$

where JSI is the JS interpreter generated using default features, and $mJSI$ the miniaturized JS interpreter. Negative values indicates an improvement in p value, and positive values a detriment. Note that the median is computed on the number of runs chosen.

Algorithm 1: Steps to benchmark JS interpreter features.

Input : JS, *file with features*, $codeSize\ JSI$, $memUs\ JSI$, $CPUTime\ JSI$, $runs$
Output: $report\_file$

1 $report\_file = \emptyset$
2 **forall** *row* $\in$ *file with features* **do**
3 Save feature and its test value on (cfg)
4 Generate $mJSI$ based on cfg
5 **if** *generation fails* **then**
6 Continue with next row
7 **end**
8 $codeSize\ mJSI$ = measure code size ($mJSI$)
9 $\delta codeSize\ JSI = \frac{codeSize\ mJSI - codeSize\ JSI}{codeSize\ JSI}$
10 Open report file
11 **for** 1 **to** $runs$ **do**
12 Execute JS using $memUs\ mJSI$ = measure memory usage ($mJSI$)
13 $\delta memUs = \frac{memUs\ mJSI - memUs\ JSI}{memUs\ JSI}$
14 $CPUTime\ mJSI$ = measure CPU time ($mJSI$)
15 $\delta CPUTime = \frac{CPUTime\ mJSI - CPUTime\ JSI}{CPUTime\ JSI}$
16 Write $\delta codeSize, \delta memUs, \delta CPUTime$ to $report\_file$
17 **end**
18 Close $report\_file$
19 **end**
20 **return** $report\_file$

Table 12: List of *Duktape* features used in this work.

ID	Property	Default value	Modified value	*Duktape* Category	Bin. value
1	DUK_USE_ALLOW_UNDEFINED_BEHAVIOR	FALSE	TRUE	Platform and portability options	0
2	DUK_USE_FATAL_MAXLEN	128	64	Platform and portability options	1
3	DUK_USE_EXEC_PREFER_SIZE	FALSE	TRUE	low Memory management options	0
4	DUK_USE_LEXER_SLIDING_WINDOW	TRUE	FALSE	low Memory management options	1
5	DUK_USE_LIGHTFUNC_BUILTINS	FALSE	TRUE	low Memory management options	0
6	DUK_USE_PREFER_SIZE	FALSE	TRUE	low Memory management options	0
7	DUK_USE_ROM_STRINGS	FALSE	TRUE	low Memory management options	0
8	DUK_USE_ROM_OBJECTS	FALSE	TRUE	low Memory management options	0
9	DUK_USE_ROM_GLOBAL_INHERIT	FALSE	TRUE	low Memory management options	0
10	DUK_USE_HSTRING_ARRIDX	TRUE	FALSE	low Memory management options	1
11	DUK_USE_REFERENCE_COUNTING	TRUE	FALSE	Garbage collection options	1
12	DUK_USE_PARANOID_ERRORS	FALSE	TRUE	ECMAScript Edition 5 (ES5) options	0
13	DUK_USE_FUNC_NAME_PROPERTY	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
14	DUK_USE_DOUBLE_LINKED_HEAP	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
15	DUK_USE_ARRAY_BUILTIN	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
16	DUK_USE_AUGMENT_ERROR_CREATE	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
17	DUK_USE_AUGMENT_ERROR_THROW	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
18	DUK_USE_BOOLEAN_BUILTIN	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
19	DUK_USE_DATE_BUILTIN	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
20	DUK_USE_ERRCREATE	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
21	DUK_USE_ERRTHROW	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
22	DUK_USE_FUNCTION_BUILTIN	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
23	DUK_USE_FUNC_FILENAME_PROPERTY	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
24	DUK_USE_GLOBAL_BUILTIN	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
25	DUK_USE_JC	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
26	DUK_USE_JSON_BUILTIN	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
27	DUK_USE_JSON_SUPPORT	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
28	DUK_USE_JX	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
29	DUK_USE_MATH_BUILTIN	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
30	DUK_USE_NUMBER_BUILTIN	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
31	DUK_USE_OBJECT_BUILTIN	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
32	DUK_USE_REGEXP_SUPPORT	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
33	DUK_USE_SOURCE_NONBMP	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
34	DUK_USE_STRING_BUILTIN	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
35	DUK_USE_TRACEBACKS	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
36	DUK_USE_VERBOSE_ERRORS	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
37	DUK_USE_PC2LINE	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
38	DUK_USE_VERBOSE_EXECUTOR_ERRORS	TRUE	FALSE	ECMAScript Edition 5 (ES5) options	1
39	DUK_USE_BYTECODE_DUMP_SUPPORT	TRUE	FALSE	API options	1
40	DUK_USE_BASE64_SUPPORT	TRUE	FALSE	Codecs	1
41	DUK_USE_HEX_SUPPORT	TRUE	FALSE	Codecs	1
42	DUK_USE_DUKTAPE_BUILTIN	TRUE	FALSE	*Duktape* specific options	1
43	DUK_USE_BUFFEROBJECT_SUPPORT	TRUE	FALSE	ECMAScript 2015 (ES6) options	1
44	DUK_USE_ES6	TRUE	FALSE	ECMAScript 2015 (ES6) options	1
45	DUK_USE_ES6_PROXY	TRUE	FALSE	ECMAScript 2015 (ES6) options	1
46	DUK_USE_ES6_UNICODE_ESCAPE	TRUE	FALSE	ECMAScript 2015 (ES6) options	1
47	DUK_USE_HTML_COMMENTS	TRUE	FALSE	ECMAScript 2015 (ES6) options	1
48	DUK_USE_SHEBANG_COMMENTS	TRUE	FALSE	ECMAScript 2015 (ES6) options	1
49	DUK_USE_REFLECT_BUILTIN	TRUE	FALSE	ECMAScript 2015 (ES6) options	1
50	DUK_USE_SYMBOL_BUILTIN	TRUE	FALSE	ECMAScript 2015 (ES6) options	1
51	DUK_USE_ES7	TRUE	FALSE	ECMAScript 2016 (ES7) options	1
52	DUK_USE_ES7_EXP_OPERATOR	TRUE	FALSE	ECMAScript 2016 (ES7) options	1
53	DUK_USE_ES8	TRUE	FALSE	ECMAScript 2017 (ES8) options	1
54	DUK_USE_ES9	TRUE	FALSE	ECMAScript 2018 (ES9) options	1
55	DUK_USE_ENCODING_BUILTINS	TRUE	FALSE	ECMAScript 2018 (ES9) options	1
56	DUK_USE_ARRAY_FASTPATH	TRUE	FALSE	Perfomance options	1
57	DUK_USE_ARRAY_PROP_FASTPATH	TRUE	FALSE	Perfomance options	1
58	DUK_USE_BASE64_FASTPATH	TRUE	FALSE	Perfomance options	1
59	DUK_USE_CACHE_ACTIVATION	TRUE	FALSE	Perfomance options	1
60	DUK_USE_CACHE_CATCHER	TRUE	FALSE	Perfomance options	1
61	DUK_USE_FAST_REFCOUNT_DEFAULT	TRUE	FALSE	Perfomance options	1
62	DUK_USE_HEX_FASTPATH	TRUE	FALSE	Perfomance options	1
63	DUK_USE_HOBJECT_HASH_PROP_LIMIT	8	64	Perfomance options	1
64	DUK_USE_HSTRING_LAZY_CLEN	TRUE	FALSE	Perfomance options	1
65	DUK_USE_IDCHAR_FASTPATH	TRUE	FALSE	Perfomance options	1
66	DUK_USE_JSON_QUOTESTRING_FASTPATH	TRUE	FALSE	Perfomance options	1
67	DUK_USE_JSON_DECSTRING_FASTPATH	TRUE	FALSE	Perfomance options	1
68	DUK_USE_JSON_DECNUMBER_FASTPATH	TRUE	FALSE	Perfomance options	1
69	DUK_USE_JSON_EATWHITE_FASTPATH	TRUE	FALSE	Perfomance options	1
70	DUK_USE_LITCACHE_SIZE	256	FALSE	Perfomance options	1
71	DUK_USE_REGEXP_CANON_BITMAP	TRUE	FALSE	Perfomance options	1
72	DUK_USE_STRTAB_MINSIZE	1024	128	Perfomance options	1
73	DUK_USE_STRTAB_MAXSIZE	268435456	128	Perfomance options	1
74	DUK_USE_STRTAB_SHRINK_LIMIT	6	0	Perfomance options	1
75	DUK_USE_STRTAB_GROW_LIMIT	17	65536	Perfomance options	1
76	DUK_USE_VALSTACK_GROW_SHIFT	2	FALSE	Perfomance options	1
77	DUK_USE_VALSTACK_SHRINK_CHECK_SHIFT	2	FALSE	Perfomance options	1
78	DUK_USE_VALSTACK_SHRINK_SLACK_SHIFT	4	FALSE	Perfomance options	1
79	DUK_USE_VALSTACK_UNSAFE	FALSE	TRUE	Perfomance options	0
80	DUK_USE_DEBUG_BUFSIZE	65536	2048	Debugger options	1
81	DUK_USE_COROUTINE_SUPPORT	TRUE	FALSE	Execution options	1
82	DUK_USE_PERFORMANCE_BUILTIN	TRUE	FALSE	Performance API (High Resolution Time)	1
83	DUK_USE_VOLUNTARY_GC	TRUE	FALSE	Garbage collection options	1
84	DUK_USE_FASTINT	FALSE	TRUE	Performance options	0
85	DUK_USE_JSON_STRINGIFY_FASTPATH	FALSE	TRUE	Performance options	0
86	DUK_USE_REGEXP_CANON_WORKAROUND	FALSE	TRUE	Performance options	0

Table 13: Results of preliminary study of JS interpreter features and their impact on performance metrics.

id	value	*harness* size	mem. us.	δCS	δMU	median PT	median δET
1	TRUE	555896	104816	0	0	0.71	-13.41
2	64	555896	104816	0	0	0.7	-14.63
3	TRUE	490824	104816	-11.71	0	0.895	9.15
4	FALSE	555888	104816	0	0	0.7	-14.63
5	TRUE	555896	65584	0	-37.43	0.715	-12.805
6	TRUE	551696	104784	-0.76	-0.03	1.595	94.51
11	FALSE	518176	115952	-6.79	10.62	0.72	-12.2
12	TRUE	555976	104816	0.01	0	0.705	-14.02
13	FALSE	555896	103440	0	-1.31	0.71	-13.41
15	FALSE	546424	99728	-1.7	-4.85	0.69	-15.85
16	FALSE	555728	104816	-0.03	0	0.705	-14.02
17	FALSE	555840	104816	-0.01	0	0.7	-14.63
18	FALSE	555776	104080	-0.02	-0.7	0.705	-14.02
19	FALSE	555280	93152	-0.11	-11.13	0.71	-13.41
20	FALSE	555896	104816	0	0	0.715	-12.805
21	FALSE	555896	104816	0	0	0.71	-13.41
22	FALSE	555728	103616	-0.03	-1.14	0.72	-12.2
23	FALSE	555896	104720	0	-0.09	0.71	-13.41
24	FALSE	555320	102192	-0.1	-2.5	0.775	-5.49
25	FALSE	555896	104816	0	0	0.755	-7.93
26	FALSE	555792	104160	-0.02	-0.63	0.83	1.22
28	FALSE	555680	104816	-0.04	0	0.79	-3.66
29	FALSE	549800	97696	-1.1	-6.79	0.71	-13.41
30	FALSE	555432	102640	-0.08	-2.08	0.705	-14.02
31	FALSE	555144	99120	-0.14	-5.43	0.705	-14.02
33	FALSE	555904	104816	0	0	0.71	-13.41
34	FALSE	546280	98384	-1.73	-6.14	0.72	-12.2
35	FALSE	555840	104816	-0.01	0	0.705	-14.02
36	FALSE	543328	104816	-2.26	0	0.7	-14.63
37	FALSE	555736	104640	-0.03	-0.17	0.7	-14.63
38	FALSE	555896	104816	0	0	0.71	-13.41
39	FALSE	547216	104816	-1.56	0	0.7	-14.63
40	FALSE	555896	104816	0	0	0.71	-13.41
41	FALSE	555896	104816	0	0	0.71	-13.41
42	FALSE	555536	101568	-0.06	-3.1	0.71	-13.41
43	FALSE	532664	82400	-4.18	-21.39	0.705	-14.02
44	FALSE	550952	100720	-0.89	-3.91	0.715	-12.805
45	FALSE	551488	104448	-0.79	-0.35	0.705	-14.02
46	FALSE	555896	104816	0	0	0.705	-14.02
47	FALSE	555896	104816	0	0	0.71	-13.41
48	FALSE	555896	104816	0	0	0.715	-12.805
49	FALSE	555680	102192	-0.04	-2.5	0.705	-14.02
50	FALSE	555896	104816	0	0	0.71	-13.41
51	FALSE	555896	104816	0	0	0.71	-13.41
52	FALSE	551800	104816	-0.74	0	0.715	-12.805
53	FALSE	555768	103808	-0.02	-0.96	0.71	-13.415
54	FALSE	555896	104816	0	0	0.71	-13.41
55	FALSE	555424	102768	-0.08	-1.95	0.72	-12.2
56	FALSE	555744	104816	-0.03	0	0.71	-13.41
57	FALSE	555760	104816	-0.02	0	0.735	-10.37
58	FALSE	555808	104816	-0.02	0	0.71	-13.41
59	FALSE	555832	104592	-0.01	-0.21	0.71	-13.41
60	FALSE	555840	104704	-0.01	-0.11	0.72	-12.2
61	FALSE	535880	104816	-3.6	0	0.705	-14.02
62	FALSE	555808	104816	-0.02	0	0.7	-14.63
63	64	555896	100832	0	-3.8	0.705	-14.02
64	FALSE	555880	104816	0	0	0.71	-13.41
65	FALSE	555848	104816	-0.01	0	0.7	-14.63
66	FALSE	555888	104816	0	0	0.71	-13.41
67	FALSE	555848	104816	-0.01	0	0.71	-13.41
68	FALSE	555840	104816	-0.01	0	0.695	-15.24
69	FALSE	555840	104816	-0.01	0	0.71	-13.41
70	FALSE	555896	104816	0	0	0.7	-14.63
71	FALSE	555840	104816	-0.01	0	0.71	-13.41
74	0	555896	104816	0	0	0.705	-14.02
75	65536	555896	104816	0	0	0.71	-13.41
76	FALSE	555896	104384	0	-0.41	0.71	-13.41
77	FALSE	555896	104352	0	-0.44	0.7	-14.63
78	FALSE	555896	104816	0	0	0.7	-14.63
79	TRUE	555896	104816	0	0	0.7	-14.63
80	2048	555896	104816	0	0	0.71	-13.41
81	FALSE	551568	103872	-0.78	-0.9	0.7	-14.63
82	FALSE	555848	104304	-0.01	-0.49	0.705	-14.02
83	FALSE	555848	104816	-0.01	0	0.725	-11.59
84	TRUE	592960	104816	6.67	0	0.63	-23.17
85	TRUE	560144	104816	0.76	0	0.715	-12.805
86	TRUE	687016	104816	23.59	0	0.7	-14.63

Table 14: Results of preliminary study of JS interpreter features and their impact on performance metrics (features with dependencies).

id	value	*harness* size	mem. us.	δCS	δMU	median PT	median δPT
11_14	FALSE	514080	115632	-7.52	10.32	0.72	-12.2
26_27	FALSE	537264	104160	-3.35	-0.63	0.71	-13.41
32_34	FALSE	519208	96656	-6.6	-7.79	0.72	-12.2
7 to 10	vary	696048	12656	25.21	-87.93	0.71	-13.41
72_73	128	555736	97648	-0.03	-6.84	0.705	-14.02