**JSFuzzer Code Style Conclusions**

* **Never leave unreachable code**

Leaving written code in unreachable control flows may lead to unpredicted results.

Some JavaScript interpreters may use optimizations that eliminates unreachable code from your program. These optimizations are sometimes buggy. They may lead to a corruption of the program's global scope, and can sometimes crash the runtime environment.

* **Define your function globally**

Defining functions in conditional flows (if/while…), may lead to an unexpected behavior.

Some interpreters may not accept functions declared in an executed flows (so even if the interpreter reached to the function definition in its runtime flow, it may not be familiar with it).

Other interpreters may be familiar with functions defined in an unreached execution flow.

These two surprising results eliminates most of advantages of conditional function definitions (functions declaration in a conditional statement).

So you can either declare your function globally, or use function expressions instead of function definitions when declared in conditional scope.