Use of Deprecated Solidity Functions

Description

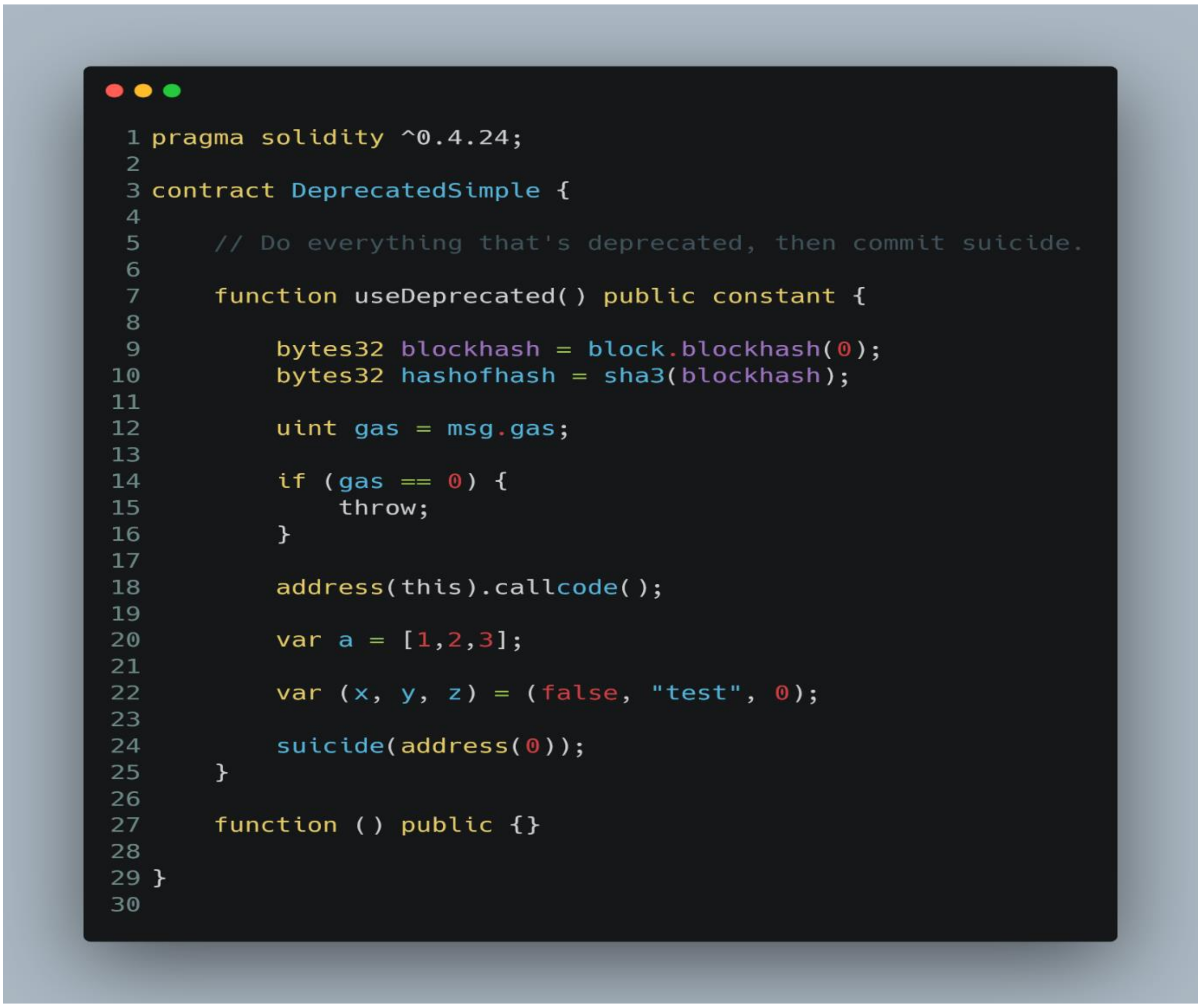
Several functions and operators in Solidity are deprecated. Using them leads to reduced code quality. With new major versions of the Solidity compiler, deprecated functions and operators may result in side effects and compile errors.

Remediation

Solidity provides alternatives to the deprecated constructions. Most of them are aliases, thus replacing old constructions will not break current behaviour. For example, sha3 can be replaced with keccak256.

Example:

Code:



```
pragma solidity ^0.4.24;

contract DeprecatedSimple {

    // Do everything that's deprecated, then commit suicide.

    function useDeprecated() public constant {

        bytes32 blockhash = block.blockhash(0);
        bytes32 hashofhash = sha3(blockhash);

        uint gas = msg.gas;

        if (gas == 0) {
            throw;
        }

        address(this).callcode();

        var a = [1,2,3];

        var (x, y, z) = (false, "test", 0);

        suicide(address(0));
    }

    function () public {}

}
```

Explanation:

In the above example, author has used the suicide function instead of self-destruct function. It is always advised to utilize the alternative function instead of the deprecated one in order to minimize security issues.

Please refer the below deprecated functions and their alternatives.

Deprecated	**Alternative**
suicide(address)	selfdestruct(address)
block.blockhash(uint)	blockhash(uint)
sha3(...)	keccak256(...)
callcode(...)	delegatecall(...)
throw	revert()
msg.gas	gasleft
constant	view
var	corresponding type name