攻防世界 Reverse高手进阶区 3分题 secret-string-400

原创

思源湖的鱼 于 2021-02-02 18:11:19 发布 139 收藏
分类专栏： ctf 文章标签： ctf 攻防世界 reverse

本文链接：https://blog.csdn.net/weixin_44604541/article/details/113568838

前言

继续ctf的旅程
攻防世界Reverse高手进阶区的3分题
本篇是secret-string-400的writeup

发现攻防世界的题目分数是动态的
就仅以做题时的分数为准了

解题过程

得到一个无后缀文件
winhex看了眼是个压缩文件

解压得到html和js文件

task.html

```
<html>
 <head>
  <title>JSMachine</title>
  <meta charset="UTF-8">
  <script type='text/javascript' src='Machine.js'></script>
 </head>
 <body>
  <h1>Secret key</h1><br/>
  <h2>Test your luck! Enter valid string and you will know flag!</h2><br/>
  <input type='text'></input><br/>
  <br/>
  <input type='button' onclick="check()" value='Проверить'></button>
 </body>
</html>
```

Machine.js

```
function createRegisters(obj){
 obj.registers = [];
 for(i=0; i < 256; ++i){
  obj.registers.push(0);
 }
};

function Machine() {
 createRegisters(this);
 this.code = [0]
 this.PC = 0;
 this.callstack = [];
 this.pow = Math.pow(2,32)
};

Machine.prototype = {
 opcodesCount: 16,
 run: run,
 loadcode: function(code){this.code = code},
 end: function(){this.code=[]}
};


function run(){
 while(this.PC < this.code.length){
  var command = parseCommand.call(this)
  command.execute(this);
 }
 //this.end()
}

function getOpcodeObject(){
 var opNum = (this.code[this.PC] % this.opcodesCount);
 this.PC += 1;
 return eval('new Opcode'+opNum);
}

function parseCommand(){
 var opcode = getOpcodeObject.call(this);
 opcode.consumeArgs(this);
 return opcode;
}

var opcCreate = "";
for(i=0;i<16;++i){
 opcCreate += "function Opcode"+i+"(){this.args=[]}\n";
}


eval(opcCreate);


function makeFromImm(obj) {
 var res = obj.code[obj.PC + 2];
 res <<=8;
 res += obj.code[obj.PC + 1];
 res <<=8;
 res += obj.code[obj.PC];
```

```
 res <<=8;
 res += obj.code[obj.PC+3];
 res = res >>> 0;
 return res;
}

function getRegImm(obj){
 this.args[0] = obj.code[obj.PC];
 obj.PC += 1;
 this.args[1] = makeFromImm(obj);
 obj.PC += 4;
}

function getImm(obj){
 this.args[0] = makeFromImm(obj);
 obj.PC += 4;
}

function getTwoRegs(obj){
 this.args[0] = obj.code[obj.PC];
 obj.PC += 1;
 this.args[1] = obj.code[obj.PC];
 obj.PC += 1;
}

function getThreeRegs(obj){
 this.args[0] = obj.code[obj.PC];
 obj.PC += 1;
 this.args[1] = obj.code[obj.PC];
 obj.PC += 1;
 this.args[2] = obj.code[obj.PC];
 obj.PC += 1;
}

function getRegString(obj){
 this.args[0] = obj.code[obj.PC];
 obj.PC += 1;
 this.args[1] = getString(obj);
}

function getRegRegString(obj){
 this.args[0] = obj.code[obj.PC];
 obj.PC += 1;
 this.args[1] = obj.code[obj.PC];
 obj.PC += 1;
 this.args[2] = getString(obj);
}

function getRegTwoString(obj){
 this.args[0] = obj.code[obj.PC];
 obj.PC += 1;
 this.args[1] = getString(obj);
 this.args[2] = getString(obj);
}

function getString(obj){
 var res = "";
 while(obj.code[obj.PC] != 0) {
  res += String.fromCharCode(obj.code[obj.PC]);
```

```
  obj.PC += 1;
 }
 obj.PC += 1;
 return res;
}

Opcode0.prototype = {
 consumeArgs : function(obj){},
 execute: function(){}
};

Opcode1.prototype = {
 consumeArgs: getRegImm,
 execute: function(obj){
  obj.registers[this.args[0]] = (obj.registers[this.args[0]] +  this.args[1]) % 0x100000000;
 }
}



Opcode2.prototype = {
 consumeArgs: getTwoRegs,
 execute: function(obj){
  obj.registers[this.args[0]] = (obj.registers[this.args[0]] + obj.registers[this.args[1]]) % 0x100000000;
 }
}

Opcode3.prototype = {
 consumeArgs: getRegImm,
 execute: function(obj){
  obj.registers[this.args[0]] = ((obj.registers[this.args[0]] -  this.args[1]) % 0x100000000) >>> 0;
 }
}

Opcode4.prototype = {
 consumeArgs: getTwoRegs,
 execute: function(obj){
  obj.regsiters[this.args[0]] = ((obj.registers[this.args[0]] - this.registers[this.args[1]])%100000000) >>> 0
 }
}

Opcode5.prototype = {
 consumeArgs: getThreeRegs,
 execute: function(obj){
  var mult = obj.registers[this.args[0]] * obj.registers[this.args[1]];
  console.log(mult.toString(16));
  obj.registers[this.args[2]] = (mult / obj.pow) >>> 0;
  obj.registers[this.args[2]+1] = (mult & 0xffffffff) >>> 0;
 }
}

Opcode6.prototype = {
 consumeArgs: getThreeRegs,
 execute: function(obj){
  var divs = obj.registers[this.args[0]] * obj.pow + obj.registers[this.args[0]+1];
  obj.registers[this.args[2]]  = (divs / obj.registers[this.args[1]]) >>> 0;
  obj.registers[this.args[2]+1]= (divs % obj.registers[this.args[1]]) >>> 0;
 }
}

Opcode7.prototype = {
```

```
Opcode7.prototype = {
 consumeArgs: getRegImm,
 execute: function(obj) {
  obj.registers[this.args[0]] = this.args[1];
 }
}

Opcode8.prototype = {
 consumeArgs: getImm,
 execute: function(obj){
  obj.callstack.push(obj.PC);
  obj.PC = this.args[0];
 }
}

Opcode9.prototype = {
 consumeArgs: getImm,
 execute: function(obj){
  obj.PC = (obj.PC +  this.args[0]) % obj.code.length;
 }
}

Opcode10.prototype = {
 consumeArgs: function(){},
 execute: function(obj){
  obj.PC = obj.callstack.pop();
 }
}

Opcode11.prototype = {
 consumeArgs: getRegString,
 execute: function(obj){
  obj.registers[this.args[0]] = eval('new '+this.args[1]);
 }
}

Opcode12.prototype = {
 consumeArgs: getRegTwoString,
 execute: function(obj){
  obj.registers[this.args[0]][this.args[1]] = Function(this.args[2]);
 }
}

Opcode13.prototype = {
 consumeArgs: getRegRegString,
 execute: function(obj){
  obj.registers[this.args[0]] = obj.registers[this.args[1]][this.args[2]];
 }
}

Opcode14.prototype = {
 consumeArgs: getRegRegString,
 execute: function(obj){
  obj.registers[this.args[1]][this.args[2]] = obj.registers[this.args[0]];
 }
}

Opcode15.prototype = {
 consumeArgs: getRegRegString,
 execute: function(obj){
```

```
  obj.registers[this.args[0]] = obj.registers[this.args[1]][this.args[2]]();
 }
}
function check(){
machine = new Machine;
machine.loadcode([11, 1, 79, 98, 106, 101, 99, 116, 0, 12, 1, 120, 0, 114, 101, 116, 117, 114, 110, 32, 100, 111
, 99, 117, 109, 101, 110, 116, 46, 103, 101, 116, 69, 108, 101, 109, 101, 110, 116, 115, 66, 121, 84, 97, 103, 7
8, 97, 109, 101, 40, 39, 105, 110, 112, 117, 116, 39, 41, 91, 48, 93, 46, 118, 97, 108, 117, 101, 47, 47, 0, 15,
 3, 1, 120, 0, 14, 3, 1, 117, 115, 101, 114, 105, 110, 112, 117, 116, 0, 12, 1, 121, 0, 119, 105, 110, 100, 111,
 119, 46, 109, 97, 99, 104, 105, 110, 101, 46, 101, 110, 100, 32, 61, 32, 102, 117, 110, 99, 116, 105, 111, 110,
 40, 41, 123, 116, 104, 105, 115, 46, 99, 111, 100, 101, 61, 91, 93, 59, 116, 104, 105, 115, 46, 80, 67, 61, 49,
 55, 51, 125, 47, 47, 0, 15, 3, 1, 121, 0, 12, 1, 122, 0, 97, 108, 101, 114, 116, 40, 49, 41, 59, 47, 47, 11, 23
4, 79, 98, 106, 101, 99, 116, 255, 9, 255, 255, 255, 12, 10, 97, 108, 101, 114, 116, 40, 50, 41, 59, 47, 47, 12,
 234, 120, 255, 118, 97, 114, 32, 102, 61, 119, 105, 110, 100, 111, 119, 46, 109, 97, 99, 104, 105, 110, 101, 46
, 114, 101, 103, 105, 115, 116, 101, 114, 115, 91, 49, 93, 46, 117, 115, 101, 114, 105, 110, 112, 117, 116, 47,
47, 10, 118, 97, 114, 32, 105, 32, 61, 32, 102, 46, 108, 101, 110, 103, 116, 104, 47, 47, 10, 118, 97, 114, 32,
110, 111, 110, 99, 101, 32, 61, 32, 39, 103, 114, 111, 107, 101, 39, 59, 47, 47, 10, 118, 97, 114, 32, 106, 32,
61, 32, 48, 59, 47, 47, 10, 118, 97, 114, 32, 111, 117, 116, 32, 61, 32, 91, 93, 59, 47, 47, 10, 118, 97, 114, 3
2, 101, 113, 32, 61, 32, 116, 114, 117, 101, 59, 47, 47, 10, 119, 104, 105, 108, 101, 40, 106, 32, 60, 32, 105,
41, 123, 47, 47, 10, 111, 117, 116, 46, 112, 117, 115, 104, 40, 102, 46, 99, 104, 97, 114, 67, 111, 100, 101, 65
, 116, 40, 106, 41, 32, 94, 32, 110, 111, 110, 99, 101, 46, 99, 104, 97, 114, 67, 111, 100, 101, 65, 116, 40, 10
6, 37, 53, 41, 41, 47, 47, 10, 106, 43, 43, 59, 47, 47, 10, 125, 47, 47, 10, 118, 97, 114, 32, 101, 120, 32, 61,
 32, 32, 91, 49, 44, 32, 51, 48, 44, 32, 49, 52, 44, 32, 49, 50, 44, 32, 54, 57, 44, 32, 49, 52, 44, 32, 49, 44,
 32, 56, 53, 44, 32, 55, 53, 44, 32, 53, 48, 44, 32, 52, 48, 44, 32, 51, 55, 44, 32, 52, 56, 44, 32, 50, 52, 44,
 32, 49, 48, 44, 32, 53, 54, 44, 32, 53, 53, 44, 32, 52, 54, 44, 32, 53, 54, 44, 32, 54, 48, 93, 59, 47, 47, 10,
 105, 102, 32, 40, 101, 120, 46, 108, 101, 110, 103, 116, 104, 32, 61, 61, 32, 111, 117, 116, 46, 108, 101, 110,
 103, 116, 104, 41, 32, 123, 47, 47, 10, 106, 32, 61, 32, 48, 59, 47, 47, 10, 119, 104, 105, 108, 101, 40, 106,
32, 60, 32, 101, 120, 46, 108, 101, 110, 103, 116, 104, 41, 123, 47, 47, 10, 105, 102, 40, 101, 120, 91, 106, 93
, 32, 33, 61, 32, 111, 117, 116, 91, 106, 93, 41, 47, 47, 10, 101, 113, 32, 61, 32, 102, 97, 108, 115, 101, 59,
47, 47, 10, 106, 32, 43, 61, 32, 49, 59, 47, 47, 10, 125, 47, 47, 10, 105, 102, 40, 101, 113, 41, 123, 47, 47, 1
0, 97, 108, 101, 114, 116, 40, 39, 89, 79, 85, 32, 87, 73, 78, 33, 39, 41, 59, 47, 47, 10, 125, 101, 108, 115, 1
01, 123, 10, 97, 108, 101, 114, 116, 40, 39, 78, 79, 80, 69, 33, 39, 41, 59, 10, 125, 125, 101, 108, 115, 101, 1
23, 97, 108, 101, 114, 116, 40, 39, 78, 79, 80, 69, 33, 39, 41, 59, 125, 47, 47, 255, 9, 255, 255, 255, 12, 10,
97, 108, 101, 114, 116, 40, 51, 41, 59, 47, 47, 15, 1, 234, 120, 255, 9, 255, 255, 255, 12, 10, 97, 108, 101, 11
4, 116, 40, 52, 41, 59, 47, 47, 10, 97, 108, 101, 114, 116, 40, 53, 41, 59, 47, 47, 10, 97, 108, 101, 114, 116,
40, 54, 41, 59, 47, 47, 10, 97, 108, 101, 114, 116, 40, 55, 41, 59, 47, 47, 0, 12, 1, 103, 0, 118, 97, 114, 32,
105, 32, 61, 48, 59, 119, 104, 105, 108, 101, 40, 105, 60, 119, 105, 110, 100, 111, 119, 46, 109, 97, 99, 104, 1
05, 110, 101, 46, 99, 111, 100, 101, 46, 108, 101, 110, 103, 116, 104, 41, 123, 105, 102, 40, 119, 105, 110, 100
, 111, 119, 46, 109, 97, 99, 104, 105, 110, 101, 46, 99, 111, 100, 101, 91, 105, 93, 32, 61, 61, 32, 50, 53, 53,
 32, 41, 32, 119, 105, 110, 100, 111, 119, 46, 109, 97, 99, 104, 105, 110, 101, 46, 99, 111, 100, 101, 91, 105,
93, 32, 61, 32, 48, 59, 105, 43, 43, 125, 47, 47, 0, 12, 1, 104, 0, 119, 105, 110, 100, 111, 119, 46, 109, 97, 9
9, 104, 105, 110, 101, 46, 80, 67, 61, 49, 55, 50, 47, 47, 0, 15, 0, 1, 103, 0, 15, 0, 1, 104, 0])
machine.run();
}
```

打开html

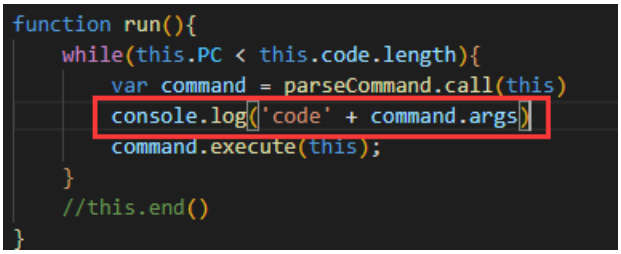
Secret key

Test your luck! Enter valid string and you will know flag!

Проверить

可以看到关键在于check函数

在run里面打印出来



```
function run(){
    while(this.PC < this.code.length){
        var command = parseCommand.call(this)
        console.log('code' + command.args)
        command.execute(this);
    }
    //this.end()
}
```

```
code234,x,var f=window.machine.registers[1].userinput//
var i = f.length//
var nonce = 'groke';//
var j = 0;//
var out = [];//
var eq = true;//
while(j < i){//
out.push(f.charCodeAt(j) ^ nonce.charCodeAt(j%5))//
j++;//
}//
var ex =  [1, 30, 14, 12, 69, 14, 1, 85, 75, 50, 40, 37, 48, 24, 10, 56, 55, 46, 56, 60];//
if (ex.length == out.length) {//
j = 0;//
while(j < ex.length){//
if(ex[j] != out[j])//
eq = false;//
j += 1;//
}//
if(eq){//
alert('YOU WIN!');//
}else{
alert('NOPE!');
}}else{alert('NOPE!');}//
```


整理下

```
f = window.machine.registers[1].userinput//
var i = f.length
var nonce = 'groke';
var j = 0;
var out = [];
var eq = true;
while (j < i) {
 out.push(f.charCodeAt(j) ^ nonce.charCodeAt(j % 5))
 j++;
}
var ex = [1, 30, 14, 12, 69, 14, 1, 85, 75, 50, 40, 37, 48, 24, 10, 56, 55, 46, 56, 60];
if (ex.length == out.length) {
 j = 0;
 while (j < ex.length) {
  if (ex[j] != out[j])
   eq = false;
  j += 1;
 }
 if (eq) {
  alert('YOU WIN!');
 } else {
  alert('NOPE!');
 }
} else {
 alert('NOPE!');
}
```

写脚本

```
nonce = 'groke'
ex = [1, 30, 14, 12, 69, 14, 1, 85, 75, 50, 40, 37, 48, 24, 10, 56, 55, 46, 56, 60]
flag = [0] * 20
for i in range(20):
 flag[i] = ex[i] ^ ord(nonce[(i%5)])
print(''.join(map(chr,flag)))
```

得到flag

```
nonce = 'groke'
ex = [1, 30, 14, 12, 69, 14, 1, 85, 75, 50, 40, 37, 48, 24, 10, 56, 55, 46, 56, 60]
flag = [0] * 20
for i in range(20):
    flag[i] = ex[i] ^ ord(nonce[(i%5)])
print(''.join(map(chr,flag)))
```

```
flag is: WOW_so_EASY
```

结语

关键是得到源码