HTML Guard 2.11
geschützte Seiten knacken

Vorwort

Tja, wieder einmal habt ihr eine Seite entdeckt, deren Quelltext mit „HTML Guard“(www.htmlGuard.com) verschlüsselt wurde. Vielleicht will da jemand böse Sachen tun und ihr sollt nichts mitbekommen. Wenn ihr herausfinden wollt, was jemand da vor euch zu verbergen hat, dann seid ihr hier genau richtig. Gleichzeitig distanziere ich mich von jeder Handlung, die meine Anleitung zur Folge hat und weise auf das Copyright hin, das die Autoren auf ihre Quelltexte haben.

Einführung

Der „HTML Guard“ funktioniert so: Man schreibt eine HTML Seite und lässt sie verschlüsseln. Der „HTML Guard“ analysiert den Quelltext und verschlüsselt einen großen Teil davon. Dann erzeugt der „HTML Guard“ eine HTML Seite mit einem Script, das den verschlüsselten Teil der HTML Seite decodiert und in den Cache des Browsers schreibt. Der Browser zeigt das ganze an und der entschlüsselte Quelltext ist sicher und unerreichbar für uns im Browsercache.
Im Detail geht der „HTML Guard“ noch etwas trickreicher vor. Er verschlüsselt nämlich die Routine, die den verschlüsselten Text entschlüsseln soll auch noch einmal, was unsere Arbeit etwas verkompliziert. Deswegen müssen wir in zwei Schritten entschlüsseln, was lästige Arbeit bedeutet.

Vorarbeit

Der „HTML Guard“ verfügt leider über ein paar zusätzliche Funktionen, die einem die Arbeit schwer machen, aber für echte Hacker kein Problem sind. Er kann die Rechte Maustaste sperren und die Auswahl des Quelltextes verhindern und dazu noch verhindern, das Bilder gespeichert werden und zu allem Überfluss kann man die Seite nicht drucken. Diese Features sind wahlweise und nicht immer anzutreffen, aber einen Hacker kann das nicht aufhalten.
Also, wenn du eine mit „HTML Guard“ geschützte Seite findest, speichere die Seite ab und suchen dir die HTML Seite raus, die den verschlüsselten Code enthält. Diese öffnest du dann mit einem Editor, der kein HTML interpretiert (z.B. Notepad). Sollte das aus irgendeinem Grund nicht erfolgreich sein, dann lassen sie sich den Quelltext anzeigen (IE: „Ansicht“ -> „Quelltext“; Netscape: „Ansicht“ -> „Rahmenquelltext anzeigen“). Den Quelltext kopierst du dann und fügst ihn in eine neu erstellte, leere HTML Datei ein. Dann kann es losgehen.
Hinweise: Mach dir ne Sicherheitskopie von der Datei, du wirst sie noch brauchen und glaubt nicht, dass der Quelltext ruckzuck entschlüsselt ist. Plant mindestens 20 Minuten ein und lest ruhig zweimal, wenn ihr was nicht checkt.

Eine Seite entschlüsseln: Beispiel

Es folgt nun ein Beispiel für eine unverschlüsselte HTML Seite. Hier nun ein Quelltext:

```
<html>
<head>
<title>Eine Testseite für HTML Guard 2.11</title>
</head>
<body>
<h2>Dies ist ein Testtext für die Testseite für HTML Guard.</h2>
<p align=right><font size=2><i>(c) Megaman IV</i></font></p>
</body>
</html>
```

So, und nun der Quelltext derselben Seite, nachdem sie mit „HTML Guard 2.11“ verschlüsselt wurde.

```
<html><head><meta http-equiv="expires" content="2"><title>Eine Testseite für HTML Guard
2.11</title></head><body><script language="JavaScript" type="text/javascript"><!--
var n="",l=54,f="i21tf4\"-sVT>epbx(z#vhg%.0;lH
ud&GaDA<cM=m7wn/rLo3)FIü5",k="";eval(unescape("%66%75%6E%63%74%69%6F%6E%20%6D%28%78%29%7B%76%6
1%72%20%75%3D%27%27%2C%77%2C%7A%2C%6A%2C%6F%3B%66%6F%72%28%77%3D%30%3B%77%3C%78%2E%6C%65%6E%67
%74%68%3B%77%2B%2B%29%7B%7A%3D%78%2E%63%68%61%72%41%74%28%77%29%3B%6A%3D%66%2E%69%6E%64%65%78%
4F%66%28%7A%29%3B%69%66%28%6A%3E%2D%31%29%7B%6F%3D%28%28%6A%2B%31%29%25%6C%2D%31%29%3B%69%66%2
8%6F%3C%3D%30%29%7B%6F%2B%3D%6C%7D%75%2B%3D%66%2E%63%68%61%72%41%74%28%6F%2D%31%29%7D%65%6C%73
%65%7B%75%2B%3D%7A%7D%7D%6E%2B%3D%75%7D%3B%66%75%6E%63%74%69%6F%6E%20%64%64%64%28%29%7B%64%6F%
63%75%6D%65%6E%74%2E%77%72%69%74%65%28%6E%29%7D"));m("cfDxHpun2&fgm-t;;.-ux3L&pLm-;-
uMpHHVbDM2/%m-)-uMpHHbD&&2/%m-t-ecfLux%M3H3Lm-v\"\"iiww-ecf&ecfDxHpux3L&pLm-;-uMpHHVbDM2/%m-;-
uMpHHbD&&2/%m-t-ux%M3H3Lm-vIIIIII-un2&fgm-t;;.-ecfLux%M3H3Lm-v\"\"iiww-uDH2%/m-Mp/fpL-
ecf&ec43/fu4DMpm-<L2DH,u pHhpf2MD,uVD/VsVpL24-uM3H3Lm-vIIIIII-uV2#pm-st-
eapVMgGdd7Hlf#fu72fup2/pLud/Lp%2VfL2pLfp/uTpLV23/uh3/cr43/fecrf&ecrfLecfLux%M3H3Lm-vIIIIII-
uDH2%/m-Mp/fpL-ecf&ec43/fu4DMpm-<L2DH,u pHhpf2MD,uVD/VsVpL24-uM3H3Lm-v\"\"iiww-ecxe
>=ouadDL&crxecr43/fecrf&ecrfLecrfDxHpecrf&e");m("crfLecrfDxHpeucg1eA2pVu2Vfup2/u>pVffp(fu45Lu&
2pu>pVfVp2fpu45Lu
>=ouadDL&0crg1eucbuDH2%/mL2%gfec43/fuV2#pm1ec2ezMFu=p%D7D/uüTcr2ecr43/fecrbe");ddd();document.
write(k);k="";//--></script><noscript>Zur Anzeige dieser Seite ben&ouml;tigen Sie einen
JavaScript-f&auml;higen Browser.</noscript></body></html>
```

Ihr seht, dass da ein Unterschied ist. Der Browser zeigt die verschlüsselte Version genauso an, wie die unverschlüsselte. An diesem Beispiel zeige ich, wie man von einer verschlüsselten HTML Seite wieder den originalen Quelltext bekommt.

Schritt 1: Script entschlüsseln

Das Script, das den Quelltext entschlüsselt ist recht lächerlich verschlüsselt, wenn man überhaupt von einer Verschlüsselung reden kann. Diesen verschlüsselten Teil müssen wir erst sichtbar machen. Kein Problem. Legt spätestens jetzt eine Kopie der verschlüsselten Seite an. An der werden wir jetzt rumspielen.
Sicher ist dir bei genauerer Begutachtung der verschlüsselten Seite dieser Block hier aufgefallen:

```
eval(unescape("%66%75%6E%63%74%69%6F%6E%20%6D%28%78%29%7B%76%61%72%20%75%3D
%27%27%2C%77%2C%7A%2C%6A%2C%6F%3B%66%6F%72%28%77%3D%30%3B%77%3C%78%2E%6C%65
%6E%67%74%68%3B%77%2B%2B%29%7B%7A%3D%78%2E%63%68%61%72%41%74%28%77%29%3B%6A
%3D%66%2E%69%6E%64%65%78%4F%66%28%7A%29%3B%69%66%28%6A%3E%2D%31%29%7B%6F%3D
%28%28%6A%2B%31%29%25%6C%2D%31%29%3B%69%66%28%6F%3C%3D%30%29%7B%6F%2B%3D%6C
%7D%75%2B%3D%66%2E%63%68%61%72%41%74%28%6F%2D%31%29%7D%65%6C%73%65%7B%75%2B
%3D%7A%7D%7D%6E%2B%3D%75%7D%3B%66%75%6E%63%74%69%6F%6E%20%64%64%64%28%29%7B
%64%6F%63%75%6D%65%6E%74%2E%77%72%69%74%65%28%6E%29%7D"));
```

Profis sehen sofort, dass das das verschlüsselte Script ist. Hier nun die Erklärung (ich hoffe ihr könnte ein bisschen programmieren). Es handelt sich um eine verschachtelte

Funktion. Der ganz Block mit den „%56“ oder „%3E“ sind Buchstaben, die in Hexwerte übersetzt wurden. Die Hexwerte werden mit der Funktion „unescape()“ wieder zu einem String verarbeitet. Nämlich dem Script. Und nun kommt der Trick. Mit der Funktion „eval()“ wird der Browser angewiesen, dass, was in der Klammer steht (nämlich das Script) als JavaScript Quelltext zu betrachten und auszuführen, ohne, dass das Script explizit im Quelltext steht (vgl. SelfHTML -> OUA Funktionen). Trickreich oder? An diesem Quelltext müssen wir ran. Dazu bedienen wir uns einer Textbox.
Erstellt mal eine neue HTML Seite, die eine Textarea enthält. So was zum Beispiel:

```
<html>
<body>
<h3>Textarea für das Script</h3>
<form name="forumlar">
<textarea rows="20" cols="70" name="source">Ok</textarea>
</form>
</body>
</html>
```

Die Seite enthält eine Textarea in welcher wir den Source einfügen lassen werden. So, nun nehmen wir uns das verschlüsselte Script und lassen es mit „unescape()“ (ohne das „eval()“!) entschlüsseln und fügen das Ergebnis in die Textarea ein. In unserem Beispiel würde das so aussehen:

```
<html>
<body>
<h3>Textarea für das Script</h3>
<form name="formular">
<textarea rows="20" cols="70" name="source">Ok</textarea>
</form>
<script language="JavaScript">
var x;
x=unescape("%66%75%6E%63%74%69%6F%6E%20%6D%28%78%29%7B%76%61%72%20%75%3D%27
%27%2C%77%2C%7A%2C%6A%2C%6F%3B%66%6F%72%28%77%3D%30%3B%77%3C%78%2E%6C%65%6E
%67%74%68%3B%77%2B%2B%29%7B%7A%3D%78%2E%63%68%61%72%41%74%28%77%29%3B%6A%3D
%66%2E%69%6E%64%65%78%4F%66%28%7A%29%3B%69%66%28%6A%3E%2D%31%29%7B%6F%3D%28
%28%6A%2B%31%29%25%6C%2D%31%29%3B%69%66%28%6F%3C%3D%30%29%7B%6F%2B%3D%6C%7D
%75%2B%3D%66%2E%63%68%61%72%41%74%28%6F%2D%31%29%7D%65%6C%73%65%7B%75%2B%3D
%7A%7D%7D%6E%2B%3D%75%7D%3B%66%75%6E%63%74%69%6F%6E%20%64%64%64%28%29%7B%64
%6F%63%75%6D%65%6E%74%2E%77%72%69%74%65%28%6E%29%7D");
document.formular.source.value = x;
</script>
</body>
</html>
```

Hurra, in unserer Textarea ist nun der Source. Das war billig. Das Ergebnis ist:

```
function m(x){var
u='',w,z,j,o;for(w=0;w<x.length;w++){z=x.charAt(w);j=f.indexOf(z);if(j>-
1){o=((j+1)%l-1);if(o<=0){o+=l}u+=f.charAt(o-1)}else{u+=z}}n+=u};function
ddd(){document.write(n)}
```

Ohne das Script verstehen zu müssen können wir uns voll freuen, denn das ist das Script, was die Seite entschlüsselt. An selbigen werden wir nun rummanipulieren.

Schritt 2: Quelltext anzeigen lassen

Willkommen zu dem aufwendigsten Teil des Ganzen. Profis schaffen das natürlich in einem Schritt und brauchen keine Sicherheitskopie, aber wenn du das hier liest ist die Wahrscheinlichkeit, dass du Profi bist gering.
Nun geht es daran, in die Verschlüsselte Seite eine Textbox einfügen zulassen und das Script so abzuändern, das es den entschlüsselten Quelltext und die Textarea schreibt.
Öffnet die zuvor angelegte Kopie der verschlüsselten Seite (ich hab's euch ja gesagt!). Da muss jetzt eine Textarea rein und dazu musst du in dem Quelltext der Seite direkt zwischen dem Body-Tag und den Script-Tag die Textarea samt Formular einfügen (Blau markierter Teil):

```
<html><head><meta http-equiv="expires" content="2"><title>Eine Testseite für HTML
Guard 2.11</title></head><body>
<form name="formular">
<textarea rows="20" cols="70" name="source">Ok</textarea>
</form>
<script language="JavaScript" type="text/javascript"><!--
var n="",l=54,f="i21tf4\"-sVT>epbx(z#vhg%.0;lH
ud&GaDA<cM=m7wn/rLo3)FIü5",k="";eval(unescape("%66%75%6E%63%74%69%6F%6E%20%6D%28%78
%29%7B%76%61%72%20%75%3D%27%27%2C%77%2C%7A%2C%6A%2C%6F%3B%66%6F%72%28%77%3D%30%3B%7
7%3C%78%2E%6C%65%6E%67%74%68%3B%77%2B%2B%29%7B%7A%3D%78%2E%63%68%61%72%41%74%28%77%
29%3B%6A%3D%66%2E%69%6E%64%65%78%4F%66%28%7A%29%3B%69%66%28%6A%3E%2D%31%29%7B%6F%3D
%28%28%6A%2B%31%29%25%6C%2D%31%29%3B%69%66%28%6F%3C%3D%30%29%7B%6F%2B%3D%6C%7D%75%2
B%3D%66%2E%63%68%61%72%41%74%28%6F%2D%31%29%7D%65%6C%73%65%7B%75%2B%3D%7A%7D%7D%6E%
2B%3D%75%7D%3B%66%75%6E%63%74%69%6F%6E%20%64%64%64%28%29%7B%64%6F%63%75%6D%65%6E%74
%2E%77%72%69%74%65%28%6E%29%7D"));m("cfDxHpun2&fgm-t;;.-ux3L&pLm-;-uMpHHVbDM2/%m-)-
uMpHHbD&&2/%m-t-ecfLux%M3H3Lm-v\"\"iiww-ecf&ecfDxHpux3L&pLm-;-uMpHHVbDM2/%m-;-
uMpHHbD&&2/%m-t-ux%M3H3Lm-vIIIIII-un2&fgm-t;;.-ecfLux%M3H3Lm-v\"\"iiww-uDH2%/m-
Mp/fpL-ecf&ec43/fu4DMpm-<L2DH,u pHhpf2MD,uVD/VsVpL24-uM3H3Lm-vIIIIII-uV2#pm-st-
eapVMgGdd7Hlf#fu72fup2/pLud/Lp%2VfL2pLfp/uTpLV23/uh3/cr43/fecrf&ecrfLecfLux%M3H3Lm-
vIIIIII-uDH2%/m-Mp/fpL-ecf&ec43/fu4DMpm-<L2DH,u pHhpf2MD,uVD/VsVpL24-uM3H3Lm-
v\"\"iiww-ecxe
>=ouadDL&crxecr43/fecrf&ecrfLecrfDxHpecrf&e");m("crfLecrfDxHpeucg1eA2pVu2Vfup2/u>pV
ffp(fu45Lu&2pu>pVfVp2fpu45Lu
>=ouadDL&0crg1eucbuDH2%/mL2%gfec43/fuV2#pm1ec2ezMFu=p%D7D/uüTcr2ecr43/fecrbe");ddd(
);document.write(k);k="";//--></script><noscript>Zur Anzeige dieser Seite
ben&ouml;tigen Sie einen JavaScript-f&auml;higen Browser.</noscript></body></html>
```

Die HTML Seite sieht dann etwas verkrüppelt aus, aber es geht ja um die Funktionalität. So nun kommt der Trick. Wir haben vorhin das Entschlüsselungsscript decodiert. Selbiges muss nun abgeändert werden. Wir hatten:

```
function m(x){var
u='',w,z,j,o;for(w=0;w<x.length;w++){z=x.charAt(w);j=f.indexOf(z);if(j>-
1){o=((j+1)%l-1);if(o<=0){o+=l}u+=f.charAt(o-1)}else{u+=z}}n+=u};function
ddd(){document.write(n)}
```

Der grüne Teil des Scripts sorgt dafür, dass der decodierte Teil der Seite angezeigt wird, indem der Source in den Cache geschrieben wird. Diesen Teil müssen wir nun mit einer Anweisung ersetzten, die den Quelltext nicht in den Cache sondern in die Textarea schreibt, die wir erstellt haben. Der decodierte Text befindet sich in der Variablen n. Das Script sieht dann so aus (Der Grüne Bereich wurde geändert):

```
function m(x){var
u='',w,z,j,o;for(w=0;w<x.length;w++){z=x.charAt(w);j=f.indexOf(z);if(j>-
1){o=((j+1)%l-1);if(o<=0){o+=l}u+=f.charAt(o-1)}else{u+=z}}n+=u};function
ddd(){document.formular.source.value = n;}
```

So, fast geschafft. Jetzt müssen wir dieses gecrackte Script mit dem verschlüsselten Script in der HTML Seite austauschen, damit nicht das Originalscript sondern unser tolles gecrackte Script ausgeführt wird. Also, der folgende Block:

```
eval(unescape("%66%75%6E%63%74%69%6F%6E%20%6D%28%78%29%7B%76%61%72%20%75%3D%27%27%2
C%77%2C%7A%2C%6A%2C%6F%3B%66%6F%72%28%77%3D%30%3B%77%3C%78%2E%6C%65%6E%67%74%68%3B%
77%2B%2B%29%7B%7A%3D%78%2E%63%68%61%72%41%74%28%77%29%3B%6A%3D%66%2E%69%6E%64%65%78
%4F%66%28%7A%29%3B%69%66%28%6A%3E%2D%31%29%7B%6F%3D%28%28%6A%2B%31%29%25%6C%2D%31%2
9%3B%69%66%28%6F%3C%3D%30%29%7B%6F%2B%3D%6C%7D%75%2B%3D%66%2E%63%68%61%72%41%74%28%
6F%2D%31%29%7D%65%6C%73%65%7B%75%2B%3D%7A%7D%7D%6E%2B%3D%75%7D%3B%66%75%6E%63%74%69
%6F%6E%20%64%64%64%28%29%7B%64%6F%63%75%6D%65%6E%74%2E%77%72%69%74%65%28%6E%29%7D")
);
```

Wird ersetzt durch:

```
function m(x){var
u='',w,z,j,o;for(w=0;w<x.length;w++){z=x.charAt(w);j=f.indexOf(z);if(j>-
1){o=((j+1)%l-1);if(o<=0){o+=l}u+=f.charAt(o-1)}else{u+=z}}n+=u};function
ddd(){document.formular.source.value = n;}
```

Beachtet, dass ihr wirklich nur den Teil ersetzten dürft und dass das Semikolon auch dazugehört. Ihr müsstet nun diesen Quelltext vor euch haben (Die blaue Stelle wurde geändert):

```
<html><head><meta http-equiv="expires" content="2"><title>Eine Testseite für HTML
Guard 2.11</title></head><body>
<form name="formular">
<textarea rows="20" cols="70" name="source">Ok</textarea>
</form>
<script language="JavaScript" type="text/javascript"><!--
var n="",l=54,f="i21tf4\"-sVT>epbx(z#vhg%.0;lHud&GaDA<cM=m7wn/rLo3)FIü5",k="";
```

```
function m(x){var
u='',w,z,j,o;for(w=0;w<x.length;w++){z=x.charAt(w);j=f.indexOf(z);if(j>-
1){o=((j+1)%l-1);if(o<=0){o+=l}u+=f.charAt(o-1)}else{u+=z}}n+=u};function
ddd(){document.formular.source.value = n;}
m("cfDxHpun2&fgm-t;;.-ux3L&pLm-;-uMpHHVbDM2/%m-)-uMpHHbD&&2/%m-t-ecfLux%M3H3Lm-
v\"\"iiww-ecf&ecfDxHpux3L&pLm-;-uMpHHVbDM2/%m-;-uMpHHbD&&2/%m-t-ux%M3H3Lm-vIIIIII-
un2&fgm-t;;.-ecfLux%M3H3Lm-v\"\"iiww-uDH2%/m-Mp/fpL-ecf&ec43/fu4DMpm-
<L2DH,upHhpf2MD,uVD/VsVpL24-uM3H3Lm-vIIIIII-uV2#pm-st-
eapVMgGdd7Hlf#fu72fup2/pLud/Lp%2VfL2pLfp/uTpLV23/uh3/cr43/fecrf&ecrfLecfLux%M3H3Lm-
vIIIIII-uDH2%/m-Mp/fpL-ecf&ec43/fu4DMpm-<L2DH,upHhpf2MD,uVD/VsVpL24-uM3H3Lm-
v\"\"iiww-
ecxe>=ouadDL&crxecr43/fecrf&ecrfLecrfDxHpecrf&e");m("crfLecrfDxHpeucg1eA2pVu2Vfup2/
u>pVffp(fu45Lu&2pu>pVfVp2fpu45Lu>=ouadDL&0crg1eucbuDH2%/mL2%gfec43/fuV2#pm1ec2ezMFu
=p%D7D/uüTcr2ecr43/fecrbe");ddd();document.write(k);k="";//--
></script><noscript>Zur Anzeige dieser Seite ben&ouml;tigen Sie einen JavaScript-
f&auml;higen Browser.</noscript></body></html>
```

Achtung: Wenn ihr diesen Source per Copy&Paste in eine neue Seite einfügt, könnte es wegen den Zeilenumbrüchen zu Fehlern kommen!
Hinweis: Das Lila gefärbte ist nur Ablenkung! Das soll euch blos auf eine falsche Fährte locken und euch verarschen, aber nun wisst ihr es besser.
Speichere die Seite ab und starte sie. Wenn du alles richtig gemacht hast, öffnet sich die Seite und dann kannst du dich freuen, denn nun erscheint dieser Text in der Textarea

```
<table width="100%" border="0" cellspacing="3" cellpadding="1"><tr
bgcolor="#445577"><td><table border="0" cellspacing="0" cellpadding="1"
bgcolor="#FFFFFF" width="100%"><tr bgcolor="#445577" align="center"><td><font
face="Arial, Helvetica, sans-serif" color="#FFFFFF" size="-1">Gesch&uuml;tzt mit
einer unregistrierten Version von</font></td></tr><tr bgcolor="#FFFFFF"
align="center"><td><font face="Arial, Helvetica, sans-serif"
color="#445577"><b>HTML Guard</b></font></td></tr></table></td></tr></table>
<h2>Dies ist ein Testtext für die Testseite für HTML Guard.</h2> <p
align=right><font size=2><i>(c) Megaman IV</i></font></p>
```

So, was sehen wir hier eigentlich? Das ist der HTML Quelltext, der verschlüsselt war. Zusätzlich ist hier noch der Code drin, der hinweist, dass ich mit der unregistrierten Version gearbeitet habe. Wenn ihr wollt, könnt ihr den entfernen. Dazu löschst du die erste Tabelle, also alles zwischen dem ersten „<table ...>“ und dem zweiten „</table>“ einschließlich der Tags (In der ersten Tabelle ist noch eine zweite verschachtelt). Dann sollte bei diesem Beispiel nur folgendes Über bleiben:

```
<h2>Dies ist ein Testtext für die Testseite für HTML Guard.</h2> <p
align=right><font size=2><i>(c) Megaman IV</i></font></p>
```

Das ist der entschlüsselte Teil des Quelltextes. Mit diesem Text muss man nun im Quelltext alles zwischen dem „<body>“ und dem „</body>“ austauschen (Der blaugrüne Teil wurde ausgetauscht). Aus:

```
<html><head><meta http-equiv="expires" content="2"><title>Eine Testseite für HTML
Guard 2.11</title></head><body>
```

```
<form name="formular">
<textarea rows="20" cols="70" name="source">Ok</textarea>
</form>
<script language="JavaScript" type="text/javascript"><!--
var n="",l=54,f="i21tf4\"-sVT>epbx(z#vhg%.0;lHud&GaDA<cM=m7wn/rLo3)FIü5",k="";
function m(x){var
u='',w,z,j,o;for(w=0;w<x.length;w++){z=x.charAt(w);j=f.indexOf(z);if(j>-
1){o=((j+1)%l-1);if(o<=0){o+=l}u+=f.charAt(o-1)}else{u+=z}}n+=u};function
ddd(){document.formular.source.value = n;}
m("cfDxHpun2&fgm-t;;.-ux3L&pLm-;-uMpHHVbDM2/%m-)-uMpHHbD&&2/%m-t-ecfLux%M3H3Lm-
v\"\"iiww-ecf&ecfDxHpux3L&pLm-;-uMpHHVbDM2/%m-;-uMpHHbD&&2/%m-t-ux%M3H3Lm-vIIIIII-
un2&fgm-t;;.-ecfLux%M3H3Lm-v\"\"iiww-uDH2%/m-Mp/fpL-ecf&ec43/fu4DMpm-
<L2DH,upHhpf2MD,uVD/VsVpL24-uM3H3Lm-vIIIIII-uV2#pm-st-
eapVMgGdd7Hlf#fu72fup2/pLud/Lp%2VfL2pLfp/uTpLV23/uh3/cr43/fecrf&ecrfLecfLux%M3H3Lm-
vIIIIII-uDH2%/m-Mp/fpL-ecf&ec43/fu4DMpm-<L2DH,upHhpf2MD,uVD/VsVpL24-uM3H3Lm-
v\"\"iiww-
ecxe>=ouadDL&crxecr43/fecrf&ecrfLecrfDxHpecrf&e");m("crfLecrfDxHpeucg1eA2pVu2Vfup2/
u>pVffp(fu45Lu&2pu>pVfVp2fpu45Lu>=ouadDL&0crg1eucbuDH2%/mL2%gfec43/fuV2#pm1ec2ezMFu
=p%D7D/uüTcr2ecr43/fecrbe");ddd();document.write(k);k="";//--
></script><noscript>Zur Anzeige dieser Seite ben&ouml;tigen Sie einen JavaScript-
f&auml;higen Browser.</noscript></body></html>
```

Wird also:

```
<html><head><meta http-equiv="expires" content="2"><title>Eine Testseite für HTML
Guard 2.11</title></head><body>
<h2>Dies ist ein Testtext für die Testseite für HTML Guard.</h2> <p
align=right><font size=2><i>(c) Megaman IV</i></font></p></body></html>
```

Beachtet, dass auch unsere Textarea gelöscht werden muss. Speicher die Datei nun ab und führ sie aus.
Es ist kein Problem, wenn ihr den entschlüsselten Quelltext einfügt, aus dem der Hinweis mit der Registrierung nicht gelöscht wurde. Ihr müsst auf jeden Fall ALLES zwischen „<body ...>“ und „</body> damit austauschen. Dann seid ihr fertig und könnt vor euren Freunden angeben, das ihr verschlüsselte HTML Seiten knacken könnt.

Vergleich

So, bei unserer ganzen Entschlüsselungsaktion sind wir am Ende und haben nun den Ursprünglichen Quelltext wieder hergestellt. Es fällt auf, dass der „HTML Guard“ aus dem Quelltext sämtliche Zeilenumbrüche entfernt hat, was bedeutet, dass man sich den wieder selbst formatieren darf. Außerdem ist im Kopf der HTML Seite ein Meta-Tag (<meta http-equiv="expires" content="2">) dazugekommen ist, aber der tut nichts Böses. Ansonsten ist alles original erhalten geblieben. Bei näherer Betrachtung eröffnet sich eine Schwäche des Schutzes, denn Scripts, die innerhalb des Head-Tags definiert

wurden werden nicht verschlüsselt. Sollte man solche Scripts „schützen“ wollen so muss man sie in den Body-Tag verlegen.

Nachwort

Wie du siehst ist alles, was man braucht Zeit. Der Schutz den „HTML Guard“ bietet ist nicht gerade sicher und kann nur normale Internetbenutzer oder Kleinkinder beeindrucken. Den auf der Webseite von „HTML Guard“ angepriesene Schutz vor Hackern ist also nur heiße Luft. Oft habe ich gelesen, das dieses Programm seine horrenden 15 US$ Wert sei, doch dies sei hier widerlegt. Ich empfehle dem Programmierer Andreas Wulf sich einen komplizierteren Schutz einfallen zu lassen.
Wer wissen will, wie man auch mit der unregistrierten Version Seiten verschlüsselt und kein Copyright Hinweis auf der Seite hat (wie bei meinem Beispiel), der soll eines meiner weiteren Tuts lesen.