Detection of browser-based cryptocurrency mining

Veelasha Moonsamy
Radboud University, The Netherlands

25 June 2019
Blockchain and Cryptocurrencies Security School
University of Padova, Italy
Radboud University, Nijmegen, NL
DiS research areas

▶ (Applied) Crypto
  ▶ Symmetric key crypto
  ▶ Identity-based applications
  ▶ Smart cards and RFID security
▶ Hardware security
  ▶ Side-channel analysis and countermeasures
  ▶ Fault attacks
▶ System Security
▶ Efficient implementations of crypto: hardware and software
▶ Post-quantum crypto
▶ Lightweight crypto: protocols and implementations
▶ Privacy engineering (Privacy & Identity lab)
▶ Read more about DiS members:
  https://www.ru.nl/dis/people/members/
iHUB – latest development

- https://www.ru.nl/ihub/
- Radboud University’s new interdisciplinary research hub on Security, Privacy, and Data Governance
- iHub brings together a diverse range of scholars from across the humanities, social sciences, engineering and natural sciences
- Tackle urgent questions raised by the increased digitalization and datafication of science and society
- Join the mailing list to keep up-to-date: https://mailman.science.ru.nl/mailman/listinfo/ihub-followers
Erasmus+ programme as of January 2019: Nijmegen & Padova

- Allows for students (and staff) to study (and teach) at universities in the EU member states for set periods of time
- Inter-institutional agreement from 2018/19 until 2021/22
- Suitable for both student and staff exchanges
- More about:
  - Bachelor programme: https://www.ru.nl/english/education/bachelors/computing-science/programme-outline/
- All courses are taught in English (both at the Bachelor and Master level)
Summer Schools organized by DiS members

1. Summer school on real-world crypto and privacy (June 2020, Croatia)
   ▶ This year: 17-21 June (last week), with 200 participants
   ▶ [https://summerschool-croatia.cs.ru.nl/](https://summerschool-croatia.cs.ru.nl/)
   ▶ Registration and stipend application will open in February 2020

2. Interdisciplinary Summerschool on Privacy (September, Nijmegen)
   ▶ 1-6 September 2019
   ▶ This year’s theme: Dark Patterns
Summer Schools organized by DiS members

1. Summer school on real-world crypto and privacy (June 2020, Croatia)
   ▶ This year: 17-21 June (last week), with 200 participants
   ▶ [https://summerschool-croatia.cs.ru.nl/](https://summerschool-croatia.cs.ru.nl/)
   ▶ Registration and stipend application will open in February 2020
1. Summer school on real-world crypto and privacy (June 2020, Croatia)
   ▶ This year: 17-21 June (last week), with 200 participants
   ▶ [https://summerschool-croatia.cs.ru.nl/](https://summerschool-croatia.cs.ru.nl/)
   ▶ Registration and stipend application will open in February 2020

2. Interdisciplinary Summerschool on Privacy (September, Nijmegen)
   ▶ 1-6 September 2019
   ▶ This year’s theme: Dark Patterns
Conferences organized by DiS members

- RWC 2021 (Amsterdam), [https://rwc.iacr.org](https://rwc.iacr.org)

Both conferences offer student stipends.
Conferences organized by DiS members

- RWC 2021 (Amsterdam), [https://rwc.iacr.org](https://rwc.iacr.org)

Both conferences offer student stipends.
Conferences organized by DiS members


▶ RWC 2021 (Amsterdam), https://rwc.iacr.org
Conferences organized by DiS members


- RWC 2021 (Amsterdam), [https://rwc.iacr.org](https://rwc.iacr.org)

- Both conferences offer student stipends
Acknowledgment

- Joint collaboration:

**Minesweeper: An In-depth Look into Drive-by Cryptocurrency Mining and Its Defense**

Radhesh Krishnan Konoth  
Vrije Universiteit Amsterdam  
[Email](mailto:r.k.konoth@vu.nl)

Emanuele Vineti  
Vrije Universiteit Amsterdam  
[Email](mailto:emanuele.vineti@gmail.com)

Veelasha Moonsamy  
Utrecht University  
[Email](mailto:email@veelasha.org)

Martina Lindorfer  
TU Wien  
[Email](mailto:martina@iseclab.org)

Christopher Kruegel  
UC Santa Barbara  
[Email](mailto:chris@cs.ucsb.edu)

Herbert Bos  
Vrije Universiteit Amsterdam  
[Email](mailto:herbertb@cs.vu.nl)

Giovanni Vigna  
UC Santa Barbara  
[Email](mailto:vigna@cs.ucsb.edu)

- Paper available at: [www.veelasha.org](http://www.veelasha.org)

- Link to GitHub repo in the paper
Cryptocurrency: the rise of decentralized money

- A cryptocurrency:
  - is a digital asset designed to work as a *medium of exchange*
Cryptocurrency: the rise of decentralized money

- A cryptocurrency:
  - is a digital asset designed to work as a *medium of exchange*
  - uses *cryptography* to secure financial transactions, control the creation of additional units, and verify the transfer of assets
Cryptocurrency: the rise of decentralized money

▶ A cryptocurrency:
- is a digital asset designed to work as a *medium of exchange*
- uses *cryptography* to secure financial transactions, control the creation of additional units, and verify the transfer of assets
▶ In 2009, the first cryptocurrency, ‘Bitcoin’ was introduced
A cryptocurrency:
- is a digital asset designed to work as a medium of exchange
- uses cryptography to secure financial transactions, control the creation of additional units, and verify the transfer of assets

In 2009, the first cryptocurrency, ‘Bitcoin’ was introduced.

Fast forward to 2018, about 1600 cryptocurrencies are in existence, out of which more than 600 still see an active trade.
Cryptocurrency: the rise of decentralized money

- A cryptocurrency:
  - is a digital asset designed to work as a *medium of exchange*
  - uses *cryptography* to secure financial transactions, control the creation of additional units, and verify the transfer of assets
- In 2009, the first cryptocurrency, ‘Bitcoin’ was introduced
- Fast forward to 2018, about **1600** cryptocurrencies are in existence, out of which **more than 600** still see an active trade
- An overall surge in market value across cryptocurrencies, which are mineable without specialized hardware, has renewed interest in *cryptominers*
A cryptocurrency:
- is a digital asset designed to work as a *medium of exchange*
- uses cryptography to secure financial transactions, control the creation of additional units, and verify the transfer of assets

In 2009, the first cryptocurrency, ‘Bitcoin’ was introduced

Fast forward to 2018, about **1600** cryptocurrencies are in existence, out of which **more than 600** still see an active trade

An overall surge in market value across cryptocurrencies, which are mineable without specialized hardware, has renewed interest in cryptominers

... which in turn led to the proliferation of cryptomining services, such as **Coinhive** - introduced in September 2017
Cryptocurrency: the rise of decentralized money

- A cryptocurrency:
  - is a digital asset designed to work as a medium of exchange
  - uses cryptography to secure financial transactions, control the creation of additional units, and verify the transfer of assets

- In 2009, the first cryptocurrency, ‘Bitcoin’ was introduced

- Fast forward to 2018, about 1600 cryptocurrencies are in existence, out of which more than 600 still see an active trade

- An overall surge in market value across cryptocurrencies, which are mineable without specialized hardware, has renewed interest in cryptominers

- ... which in turn led to the proliferation of cryptomining services, such as Coinhive - introduced in September 2017

- Can be easily integrated into a website to mine on its visitors’ devices from within the browser
From September 2017 onwards ...

It started with:

UNICEF Is Mining Crypto to Raise Funds for Children
From September 2017 onwards ...

It started with:

UNICEF Is Mining Crypto to Raise Funds for Children

‘Our Cryptocurrency Mining Policy: Free Content, No Ads!’
From September 2017 onwards …

And things went downhill very quickly:

Cryptocurrency Mining Malware Expected to Explode in 2018

Cryptojackers Found on Starbucks WiFi Network, GitHub, Pirate Streaming Sites

Cryptojacking Attacks Explode by 8,500 Percent

Stealthy miners steal resources and increase vulnerability
Recent update

- 08 March 2019: Coinhive is no longer in operation* 1

1https://coinhive.com/blog/en/discontinuation-of-coinhive
Recent update

- 08 March 2019: Coinhive is no longer in operation*

- Community’s reaction:

  ![Coinhive stops digging, but cryptomining still dominates](https://coinhive.com/blog/en/discontinuation-of-coinhive)
Drive-by mining aka *Cryptojacking*

- Is a web-based attack
- An infected website *secretly* executes a mining script (Javascript code and/or WebAssembly module) in user’s browser to mine cryptocurrencies
- Is considered *malicious* only when user does not explicitly give their consent
Drive-by mining aka *Cryptojacking*

- Is a web-based attack
- An infected website **secretly** executes a mining script (Javascript code and/or WebAssembly module) in user’s browser to mine cryptocurrencies
- Is considered **malicious** only when user does not explicitly give their consent
- In this work: we study the prevalence of drive-by mining attacks on Alexa’s Top 1 million websites
Threat Model

1. User

2. HTTP Request

3. HTTP Response

(Orchestrator Code)

4. Fetch Mining Payload

5. Relay Communication

Webserver

Webserver/External Server

WebSocket

Proxy

Mining Pool

Relay Communication

Mining Pool Communication

WebSocket Proxy
Current detection methods

Two main approaches have been used:
1. Blacklist-based approach
2. High CPU-based approach
Current detection method: Blacklist-based approach

- Existing defenses:

  2 https://gitlab.com/ZeroDot1/CoinBlockerLists
  3 https://github.com/1lastBr3ath/drmine
  4 https://github.com/xd4rker/MinerBlock
Current detection method: Blacklist-based approach

- Existing defenses:
  - CoinBlockerList\(^2\): maintains a blacklist of mining pools and proxy servers that are manually collected from reports on security blogs and Twitter

\(^2\)https://gitlab.com/ZeroDot1/CoinBlockerLists
\(^3\)https://github.com/1lastBr3ath/drmine
\(^4\)https://github.com/xd4rker/MinerBlock
Current detection method: Blacklist-based approach

- **Existing defenses:**
  - CoinBlockerList\(^2\): maintains a blacklist of mining pools and proxy servers that are manually collected from reports on security blogs and Twitter
  - Dr. Mine\(^3\): blocks drive-by mining by means of explicitly blacklisted URLs (based on for e.g. CoinBlockerLists)

\(^2\)https://gitlab.com/ZeroDot1/CoinBlockerLists
\(^3\)https://github.com/1lastBr3ath/drmine
\(^4\)https://github.com/xd4rker/MinerBlock
Current detection method: Blacklist-based approach

- Existing defenses:
  - CoinBlockerList\(^2\): maintains a blacklist of mining pools and proxy servers that are manually collected from reports on security blogs and Twitter
  - Dr. Mine\(^3\): blocks drive-by mining by means of explicitly blacklisted URLs (based on for e.g. CoinBlockerLists)
  - MinerBlock\(^4\): combines blacklists with detecting potential mining code inside loaded JavaScript files

---

\(^2\) [https://gitlab.com/ZeroDot1/CoinBlockerLists](https://gitlab.com/ZeroDot1/CoinBlockerLists)
\(^3\) [https://github.com/1lastBr3ath/drmine](https://github.com/1lastBr3ath/drmine)
\(^4\) [https://github.com/xd4rker/MinerBlock](https://github.com/xd4rker/MinerBlock)
Current detection method: Blacklist-based approach

- **Existing defenses:**
  - CoinBlockerList\(^2\): maintains a blacklist of mining pools and proxy servers that are manually collected from reports on security blogs and Twitter
  - Dr. Mine\(^3\): blocks drive-by mining by means of explicitly blacklisted URLs (based on for e.g. CoinBlockerLists)
  - MinerBlock\(^4\): combines blacklists with detecting potential mining code inside loaded JavaScript files

- **Shortcomings:**
  - Not scalable
  - Prone to high false negatives
  - Easily defeated by URL randomization and domain generation algorithms

---

\(^2\)[https://gitlab.com/ZeroDot1/CoinBlockerLists]
\(^3\)[https://github.com/1lastBr3ath/drmine]
\(^4\)[https://github.com/xd4rker/MinerBlock]
Current detection methods: High CPU-based approach

- Several studies found high CPU usage from the website can be used as an indicator of drive-by mining
Current detection methods: High CPU-based approach

- Several studies found high CPU usage from the website can be used as an indicator of drive-by mining
- Consequently, many drive-by miners started throttling their CPU usage to around 25%
Several studies found high CPU usage from the website can be used as an indicator of drive-by mining.

Consequently, many drive-by miners started throttling their CPU usage to around 25%.

Implications:
- False positives, as there might exist other CPU-intensive use cases (e.g. games).
- False negatives, as cryptominers have started to throttle their CPU usage to evade detection.
Minesweeper: contributions

- Perform first in-depth assessment of drive-by mining
Minesweeper: contributions

- Perform first in-depth assessment of drive-by mining
- Discuss why current defenses based on blacklisting and CPU usage are ineffective
Minesweeper: contributions

- Perform first in-depth assessment of drive-by mining
- Discuss why current defenses based on blacklisting and CPU usage are ineffective
- Propose MineSweeper, a novel detection approach based on the identification of the cryptographic functions (static analysis) and cache events (during run-time)
Drive-by mining in the wild

- Conducted a large-scale analysis with the aim to answer the following questions:
Drive-by mining in the wild

- Conducted a large-scale analysis with the aim to answer the following questions:
  1. How prevalent is drive-by mining in the wild?
Drive-by mining in the wild

- Conducted a large-scale analysis with the aim to answer the following questions:
  1. How prevalent is drive-by mining in the wild?
  2. How many different drive-by mining services exist currently?
Drive-by mining in the wild

- Conducted a large-scale analysis with the aim to answer the following questions:
  1. How prevalent is drive-by mining in the wild?
  2. How many different drive-by mining services exist currently?
  3. Which evasion tactics do drive-by mining services employ?
Drive-by mining in the wild

Conducted a large-scale analysis with the aim to answer the following questions:

1. How prevalent is drive-by mining in the wild?
2. How many different drive-by mining services exist currently?
3. Which evasion tactics do drive-by mining services employ?
4. What is the modus operandi of different types of campaign?
Drive-by mining in the wild

- Conducted a large-scale analysis with the aim to answer the following questions:
  1. How prevalent is drive-by mining in the wild?
  2. How many different drive-by mining services exist currently?
  3. Which evasion tactics do drive-by mining services employ?
  4. What is the modus operandi of different types of campaign?
  5. How much profit do these campaigns make?
Drive-by mining in the wild

- Conducted a large-scale analysis with the aim to answer the following questions:
  1. How prevalent is drive-by mining in the wild?
  2. How many different drive-by mining services exist currently?
  3. Which evasion tactics do drive-by mining services employ?
  4. What is the modus operandi of different types of campaign?
  5. How much profit do these campaigns make?
  6. What are the common characteristics across different drive-by mining services that can be used for their detection?
Large-scale Analysis: experiment set-up

[Diagram showing the flow of data from a Crawler to a Website Data Profile, including HTML, JavaScript, Wasm Asm.js, Web Socket Frames, Service Workers Log, CPU Log, and Enriched Data. The diagram also includes flows to Miner's code, Stratum, WebSocket, Workers, and CPU load, leading to Strong Factors and Weak Factors, which are then analyzed by the Analyser.]
Data collection

- Over a period of one week in mid-March 2018
Data collection

- Over a period of one week in mid-March 2018
- Crawler
  - Crawled landing page and 3 internal pages
  - Stayed on each visited page for 4 seconds
  - No simulated interacted, i.e. the crawler did not give any consent for cryptomining
Data collection

- Over a period of one week in mid-March 2018
- Crawler
  - Crawled landing page and 3 internal pages
  - Stayed on each visited page for 4 seconds
  - No simulated interacted, i.e. the crawler did not give any consent for cryptomining
- Crawled 991,513 websites; 4.6 TB raw data and 550 MB data profiles
Recall: cryptomining code consists of *orchestrator code* and *mining payload*
Recall: cryptomining code consists of orchestrator code and mining payload

Identification of orchestrator code
Recall: cryptomining code consists of orchestrator code and mining payload.

Identification of orchestrator code:
- Websites embed the orchestrator script in the main page.
Recall: cryptomining code consists of *orchestrator code* and *mining payload*

**Identification of orchestrator code**
- Websites embed the orchestrator script in the main page
- Can be detected by looking for specific string patterns
Recall: cryptomining code consists of **orchestrator code** and **mining payload**

Identification of orchestrator code

- Websites embed the orchestrator script in the main page
- Can be detected by looking for specific string patterns

```html
<script src="https://coinhive.com/lib/coinhive.min.js"></script>
<script>
    var miner = new CoinHive.Anonymous('CLIENT-ID',
        {throttle: 0.9});
    miner.start();
</script>
```
Preliminary results: Cryptomining code (1/2)

- Recall: cryptomining code consists of orchestrator code and mining payload
- Identification of orchestrator code
  - Websites embed the orchestrator script in the main page
  - Can be detected by looking for specific string patterns
    ```html
    <script src="https://coinhive.com/lib/coinhive.min.js"></script>
    <script>
      var miner = new CoinHive.Anonymous('CLIENT-ID',
                                          {throttle: 0.9});
      miner.start();
    </script>
    ```
  - Keywords: CoinHive.Anonymous or coinhive.min.js
Preliminary results: Cryptomining code (2/2)

- Identification of mining payload
  - Dump the Wasm (WebAssembly) payload
  - `--dump-wasm-` module flag in Chrome dumps the loaded Wasm modules
  - Keyword-based search: `cryptonight_hash` and `CryptonightWasmWrapper`
Effectiveness of fingerprint-based detection

<table>
<thead>
<tr>
<th>Mining Service</th>
<th>Number of Websites</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coinhive</td>
<td>514</td>
<td>59.35%</td>
</tr>
<tr>
<td>CoinImp</td>
<td>94</td>
<td>10.85%</td>
</tr>
<tr>
<td>Mineralt</td>
<td>90</td>
<td>10.39%</td>
</tr>
<tr>
<td>JSECoin</td>
<td>50</td>
<td>5.77%</td>
</tr>
<tr>
<td>CryptoLoot</td>
<td>39</td>
<td>4.50%</td>
</tr>
<tr>
<td>CryptoNoter</td>
<td>31</td>
<td>3.58%</td>
</tr>
<tr>
<td>Coinhave</td>
<td>14</td>
<td>1.62%</td>
</tr>
<tr>
<td>Minr</td>
<td>13</td>
<td>1.50%</td>
</tr>
<tr>
<td>Webmine</td>
<td>8</td>
<td>0.92%</td>
</tr>
<tr>
<td>DeepMiner</td>
<td>5</td>
<td>0.58%</td>
</tr>
<tr>
<td>Cpufun</td>
<td>4</td>
<td>0.46%</td>
</tr>
<tr>
<td>Monerise</td>
<td>2</td>
<td>0.23%</td>
</tr>
<tr>
<td>NF WebMiner</td>
<td>2</td>
<td>0.23%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>866</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>
Effectiveness of fingerprint-based detection

Detected 866 websites; 59.35% used Coinhive cryptomining services

<table>
<thead>
<tr>
<th>Mining Service</th>
<th>Number of Websites</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coinhive</td>
<td>514</td>
<td>59.35%</td>
</tr>
<tr>
<td>CoinImp</td>
<td>94</td>
<td>10.85%</td>
</tr>
<tr>
<td>Mineralt</td>
<td>90</td>
<td>10.39%</td>
</tr>
<tr>
<td>JSECoin</td>
<td>50</td>
<td>5.77%</td>
</tr>
<tr>
<td>CryptoLoot</td>
<td>39</td>
<td>4.50%</td>
</tr>
<tr>
<td>CryptoNoter</td>
<td>31</td>
<td>3.58%</td>
</tr>
<tr>
<td>Coinhave</td>
<td>14</td>
<td>1.62%</td>
</tr>
<tr>
<td>Minr</td>
<td>13</td>
<td>1.50%</td>
</tr>
<tr>
<td>Webmine</td>
<td>8</td>
<td>0.92%</td>
</tr>
<tr>
<td>DeepMiner</td>
<td>5</td>
<td>0.58%</td>
</tr>
<tr>
<td>Cpufun</td>
<td>4</td>
<td>0.46%</td>
</tr>
<tr>
<td>Monerise</td>
<td>2</td>
<td>0.23%</td>
</tr>
<tr>
<td>NF WebMiner</td>
<td>2</td>
<td>0.23%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>866</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>
Effectiveness of fingerprint-based detection

- Detected 866 websites; 59.35% used Coinhive cryptomining services
- Issues with keyword-based fingerprinting: code obfuscation and manual effort of updating signatures

<table>
<thead>
<tr>
<th>Mining Service</th>
<th>Number of Websites</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coinhive</td>
<td>514</td>
<td>59.35%</td>
</tr>
<tr>
<td>CoinImp</td>
<td>94</td>
<td>10.85%</td>
</tr>
<tr>
<td>Mineralt</td>
<td>90</td>
<td>10.39%</td>
</tr>
<tr>
<td>JSECoin</td>
<td>50</td>
<td>5.77%</td>
</tr>
<tr>
<td>CryptoLoot</td>
<td>39</td>
<td>4.50%</td>
</tr>
<tr>
<td>CryptoNoter</td>
<td>31</td>
<td>3.58%</td>
</tr>
<tr>
<td>Coinhave</td>
<td>14</td>
<td>1.62%</td>
</tr>
<tr>
<td>Minr</td>
<td>13</td>
<td>1.50%</td>
</tr>
<tr>
<td>Webmine</td>
<td>8</td>
<td>0.92%</td>
</tr>
<tr>
<td>DeepMiner</td>
<td>5</td>
<td>0.58%</td>
</tr>
<tr>
<td>Cpufun</td>
<td>4</td>
<td>0.46%</td>
</tr>
<tr>
<td>Monerise</td>
<td>2</td>
<td>0.23%</td>
</tr>
<tr>
<td>NF WebMiner</td>
<td>2</td>
<td>0.23%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>866</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>
Preliminary results: Mining pool communication (1/2)

- Miners use the Stratum protocol to communicate with the mining pool
Preliminary results: Mining pool communication (1/2)

- Miners use the Stratum protocol to communicate with the mining pool
- Use of WebSockets to allow full-duplex, asynchronous communication between code running on a webpage and servers
Preliminary results: Mining pool communication (1/2)

- Miners use the Stratum protocol to communicate with the mining pool
- Use of WebSockets to allow full-duplex, asynchronous communication between code running on a webpage and servers
- Search in WebSocket frames for keywords related to Stratum protocol

<table>
<thead>
<tr>
<th>Command</th>
<th>Keywords</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authentication</td>
<td>type:auth</td>
</tr>
<tr>
<td>Authentication accepted</td>
<td>type:authed</td>
</tr>
<tr>
<td>Fetch job</td>
<td>identifier:job</td>
</tr>
<tr>
<td>Submit solved hash</td>
<td>type:submit</td>
</tr>
<tr>
<td>Solution accepted</td>
<td>command:accepted</td>
</tr>
<tr>
<td>Set CPU limits</td>
<td>command:set_cpu_load</td>
</tr>
</tbody>
</table>
Preliminary results: Mining pool communication (2/2)

- 59,319 (5.39%) websites use WebSockets
- 1,008 websites use Stratum protocol for communication
- 2,377 websites encode the data (Hex code or salted Base64)
  - more on this later
Summary of key findings

- Identified 1,735 websites as mining cryptocurrency, out of which 1,627 (93.78%) could be identified based on keywords in the cryptomining code.
Summary of key findings

- Identified 1,735 websites as mining cryptocurrency, out of which 1,627 (93.78%) could be identified based on keywords in the cryptomining code.
- 1,008 (58.10%) use the Stratum protocol in plaintext, 174 (10.03%) obfuscate the communication protocol.
Summary of key findings

- Identified 1,735 websites as mining cryptocurrency, out of which 1,627 (93.78%) could be identified based on keywords in the cryptomining code.
- 1,008 (58.10%) use the Stratum protocol in plaintext, 174 (10.03%) obfuscate the communication protocol.
- All the websites (100.00%) use Wasm for the cryptomining payload and open a WebSocket.
- At least 197 (11.36%) websites throttle their CPU usage to less than 50%, while for only 12 (0.69%) mining websites we observed a CPU load of less than 25%.
Summary of key findings

- Identified 1,735 websites as mining cryptocurrency, out of which 1,627 (93.78%) could be identified based on keywords in the cryptomining code.
- 1,008 (58.10%) use the Stratum protocol in plaintext, 174 (10.03%) obfuscate the communication protocol.
- All the websites (100.00%) use Wasm for the cryptomining payload and open a WebSocket.
- At least 197 (11.36%) websites throttle their CPU usage to less than 50%, while for only 12 (0.69%) mining websites we observed a CPU load of less than 25%.
In-depth analysis: evasion techniques

We identified three evasion techniques, which are widely used by the drive-by mining services in our dataset

1. Code obfuscation
2. Obfuscated Stratum communication
3. Anti-debugging tricks
In-depth analysis: code obfuscation

- **Packed code**: The compressed and encoded orchestrator script is decoded using a chain of decoding functions at run time.

- **PCharCode**: The orchestrator script is converted to charCode and embedded in the webpage. At run time, it is converted back to a string and executed using JavaScript's `eval()` function.

- **Name obfuscation**: Variable names and functions names are replaced with random strings.

- **Dead code injection**: Random blocks of code, which are never executed, are added to the script to make reverse engineering more difficult.

- **Filename and URL randomization**: The name of the JavaScript file is randomized or the URL it is loaded from is shortened to avoid detection based on pattern matching.
In-depth analysis: code obfuscation

- **Packed code**: The compressed and encoded orchestrator script is decoded using a chain of decoding functions at run time.
- **PCharCode**: The orchestrator script is converted to charCode and embedded in the webpage. At run time, it is converted back to a string and executed using JavaScript’s `eval()` function.
- **Name obfuscation**: Variable names and functions names are replaced with random strings.
- **Dead code injection**: Random blocks of code, which are never executed, are added to the script to make reverse engineering more difficult.
- **Filename and URL randomization**: The name of the JavaScript file is randomized or the URL it is loaded from is shortened to avoid detection based on pattern matching.

All of the above mainly applied to orchestrator code; the only obfuscation on mining payload is *name obfuscation*.
In-depth analysis: obfuscated Stratum communication

- Identified the Stratum protocol in plaintext for 1,008 websites
In-depth analysis: obfuscated Stratum communication

- Identified the Stratum protocol in plaintext for 1,008 websites
- Manually analyzed the WebSocket communication for the remaining 727 websites and found the following:
  - 174 websites obfuscate by encoding the request, either as Hex code, or with salted Base64 encoding before transmitting it through the WebSocket
  - We could not identify any pool communication for remaining 553 websites, either due to other encodings, or due to slow server connections
In-depth analysis: Anti-debugging tricks

- 139 websites used anti-debugging tricks
- Checked code periodically to see whether the user is analyzing the code served by the webpage using developer tools
- If the developer tools are open in the browser, it stops executing any further code
MineSweeper
MineSweeper employs multiple stages in order to detect a webminer:
CryptoNight algorithm (1/2)

- CryptoNight was proposed in 2013 and popularly used by Monero (XMR)
CryptoNight algorithm (1/2)

- CryptoNight was proposed in 2013 and popularly used by Monero (XMR)
- We exploit two fundamental characteristics:
CryptoNight algorithm (1/2)

- CryptoNight was proposed in 2013 and popularly used by Monero (XMR)
- We exploit two fundamental characteristics:
  - It makes use of several cryptographic primitives, such as:
    - Keccak 1600-516, Keccak-f 1600, AES, BLAKE-256, Groestl-256, and Skein-256
CryptoNight algorithm (1/2)

- CryptoNight was proposed in 2013 and popularly used by Monero (XMR)
- We exploit two fundamental characteristics:
  - It makes use of several cryptographic primitives, such as: Keccak 1600-516, Keccak-f 1600, AES, BLAKE-256, Groestl-256, and Skein-256
  - A memory hard algorithm
    - High-performances on ordinary CPUs
    - Inefficient on today’s special purpose devices (ASICs)
    - Internal memory-hard loop: alternate reads and writes to the Last Level Cache (LLC)
CryptoNight allocates a scratchpad of 2MB in memory

On modern processors ends up in the LLC
Wasm analysis

- Linear assembly bytecode translation using the WebAssembly Binary Toolkit (WABT) debugger
- Functions identification - to create an internal representation of the code for each function
- Cryptographic operation count - track the control flow and crypto operands
- Static call graph construction, including identification of loops
CryptoNight detection

- MineSweeper is given as input a CryptoNight fingerprint
- We created a fingerprint for each of CryptoNight’s cryptographic primitives based on operands counts and flow structure
CryptoNight detection - an example

- Assume the fingerprint for BLAKE-256 has 80 XOR, 85 left shift, and 32 right shift instructions

Function `foo()`, which is an implementation of BLAKE-256, contains 86 XOR, 85 left shift, and 33 right shift instructions.

In this case, the similarity score is 3 and the difference score is 2.

All three types of instructions are present in `foo()`; `foo()` contains extra XOR and an extra shift instruction.
CryptoNight detection - an example

- Assume the fingerprint for BLAKE-256 has 80 XOR, 85 left shift, and 32 right shift instructions.

- Function `foo()`, which is an implementation of BLAKE-256, that we want to match against this fingerprint, contains 86 XOR, 85 left shift, and 33 right shift instructions.
CryptoNight detection - an example

- Assume the fingerprint for BLAKE-256 has 80 XOR, 85 left shift, and 32 right shift instructions
- Function $\texttt{foo()}$, which is an implementation of BLAKE-256, that we want to match against this fingerprint, contains 86 XOR, 85 left shift, and 33 right shift instructions
- In this case, the similarity score is 3 and difference score is 2
CryptoNight detection - an example

- Assume the fingerprint for BLAKE-256 has 80 XOR, 85 left shift, and 32 right shift instructions
- Function `foo()`, which is an implementation of BLAKE-256, that we want to match against this fingerprint, contains 86 XOR, 85 left shift, and 33 right shift instructions
- In this case, the similarity score is 3 and difference score is 2
- All three types of instructions are present in `foo()`; `foo()` contains extra XOR and an extra shift instruction
Evaluation of cryptofunction detection

- Identified 40 unique samples among the 748 collected Wasm samples
- Applied the cryptofunction detection routine of MineSweeper on them

<table>
<thead>
<tr>
<th>Detected Primitives</th>
<th>Number of Wasm Samples</th>
<th>Number of Cryptominers</th>
<th>Missing Primitives</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>30</td>
<td>30</td>
<td>-</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>3</td>
<td>AES</td>
</tr>
<tr>
<td>3</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>3</td>
<td>Skein, Keccak, AES</td>
</tr>
<tr>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>0</td>
<td>4</td>
<td>0</td>
<td>All</td>
</tr>
</tbody>
</table>
CPU cache events monitoring

- What if an attack would sacrifice part of the profits for obfuscated Wasm?
CPU cache events monitoring

- What if an attack would sacrifice part of the profits for obfuscated Wasm?
- Solution: CPU cache events monitoring
CPU cache events monitoring

- What if an attack would sacrifice part of the profits for obfuscated Wasm?
- Solution: CPU cache events monitoring
- MineSweeper monitors the L1 and L3 for load and store events caused by the CryptoNight algorithm
CPU cache events monitoring

- What if an attack would sacrifice part of the profits for obfuscated Wasm?
- Solution: CPU cache events monitoring
- MineSweeper monitors the L1 and L3 for load and store events caused by the CryptoNight algorithm
- Also detects a fundamental characteristic of the CryptoNight algorithm: the memory-hard loop!
Evaluation of blacklisting approaches

- For comparison, we evaluate MineSweeper against Dr. Mine
Evaluation of blacklisting approaches

- For comparison, we evaluate MineSweeper against Dr. Mine
- Dr. Mine uses CoinBlockerLists as the basis to detect mining websites
Evaluation of blacklisting approaches

- For comparison, we evaluate MineSweeper against Dr. Mine
- Dr. Mine uses CoinBlockerLists as the basis to detect mining websites
- Visited the 1,735 websites that were mining during our first crawl for the large-scale analysis with both tools
Evaluation of blacklisting approaches

- For comparison, we evaluate MineSweeper against Dr. Mine
- Dr. Mine uses CoinBlockerLists as the basis to detect mining websites
- Visited the 1,735 websites that were mining during our first crawl for the large-scale analysis with both tools
- Dr. Mine could only find 272 websites, while MineSweeper found 785 websites that were still actively mining cryptocurrency
Evaluation of CPU cache events monitoring (1/2)

- We visited 7 pages for the following categories of web applications:
  - Web miners
  - Videoplayes
  - Wasm-based games
  - JavaScript (JS) games
Evaluation of CPU cache events monitoring (2/2)

Our tests confirm us the effectiveness of this detection method on CryptoNight-based algorithms.

Performance counter measurements for the L1 cache for different types of web applications (logscale)

Performance counter measurements for the L3 cache for different types of web applications (logscale)
Conclusion

Drive-by mining is real and can be very profitable for high traffic websites

Current defenses are not sufficient to stop malicious mining

To severely impact their profitability, we need to aim at the core properties of the miners code: cryptographic functions and memory behaviors
Post-Minesweeper related work\(^5\)

\(^5\)This is not an exhaustive list
Post-Minesweeper related work\textsuperscript{5}

  [https://www.usenix.org/conference/usenixsecurity19/presentation/bijmans](https://www.usenix.org/conference/usenixsecurity19/presentation/bijmans)
  - This work builds upon Minesweeper
  - Performs two large studies into the world of cryptojacking, focused on organized cryptomining and the spread of cryptojacking on the Internet.

\textsuperscript{5}This is not an exhaustive list
Post-Minesweeper related work\textsuperscript{5}

  https://www.usenix.org/conference/usenixsecurity19/presentation/bijmans
  - This work builds upon Minesweeper
  - Performs two large studies into the world of cryptojacking, focused on organized cryptomining and the spread of cryptojacking on the Internet.

- *Dissecting Android Cryptocurrency Miners*,
  - Analyzed the Android miners and identified how they work
  - What are the most popular libraries and APIs used to facilitate the development of the mining script
  - What static features are typical for this class of applications

\textsuperscript{5}This is not an exhaustive list
Future directions

▶ Network-based cryptomining detection (e.g. with university or company network)
▶ Detecting “pop-under” windows used for concealing illegitimate mining

The secret lives of students who mine cryptocurrency in their dorm rooms

By Karen Hao • January 6, 2018

Cryptocurrency-Mining Botnet Malware Arrives Through ADB and Spreads Through SSH

Posted on: June 20, 2019 at 5:02 am  Posted in: Botnets, Mobile  Author: Trend Micro
Future directions

- Network-based cryptomining detection (e.g. with university or company network)
Future directions

- Network-based cryptomining detection (e.g. with university or company network)
- Detecting “pop-under” windows used for concealing illegitimate mining
Thank you for your attention!

email@veelasha.org
www.veelasha.org
@veelasha_m