Mind Your Own Cryptocurrency!

Ege Tekiner, Abbas Acar, and A. Selcuk Uluagac
Florida International University, USA

NDSS Paper Title: **A Lightweight IoT Cryptojacking Detection Mechanism in Heterogeneous Smart Home Networks**

Learning from Authoritative Security Experiment Results (LASER) Workshop
24-28 April 2022

Outline

Cryptojacking

- **Cryptojacking** is an act of using victims' processing power without their knowledge and consent.
- Examples:
  - US DOD
  - UK Governmental Services
  - YouTube
  - Nintendo game consoles

Bug hunter finds cryptocurrency-mining botnet on DOD network

Monero-mining botnet infects one of the DOD's Jenkins servers.

Cryptojacking attack hits ~4,000 websites, including UK's data watchdog

Natasha Lomas @riptari / 6:38 AM EST • February 12, 2018

The process is known as crypto-jacking, and it's a growing problem

Cryptojacking Types

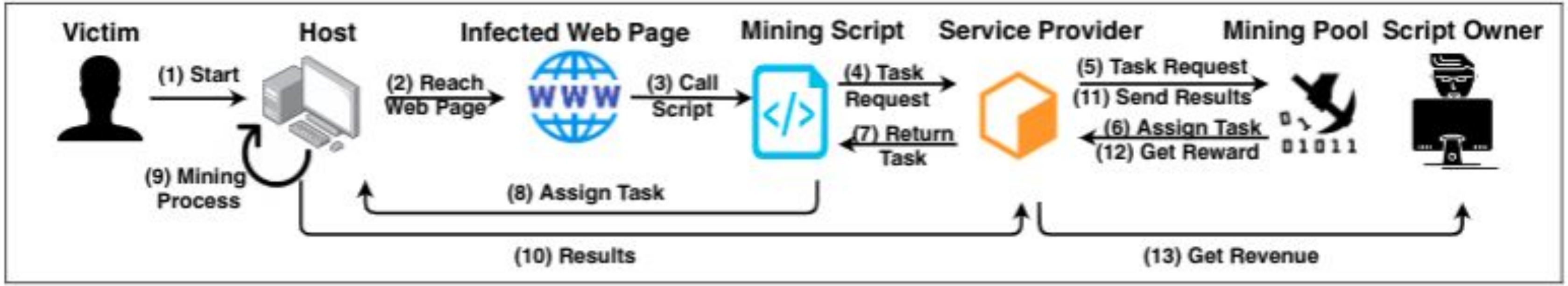
Cryptojacking Malware

In-Browser Cryptojacking

Host-based Cryptojacking

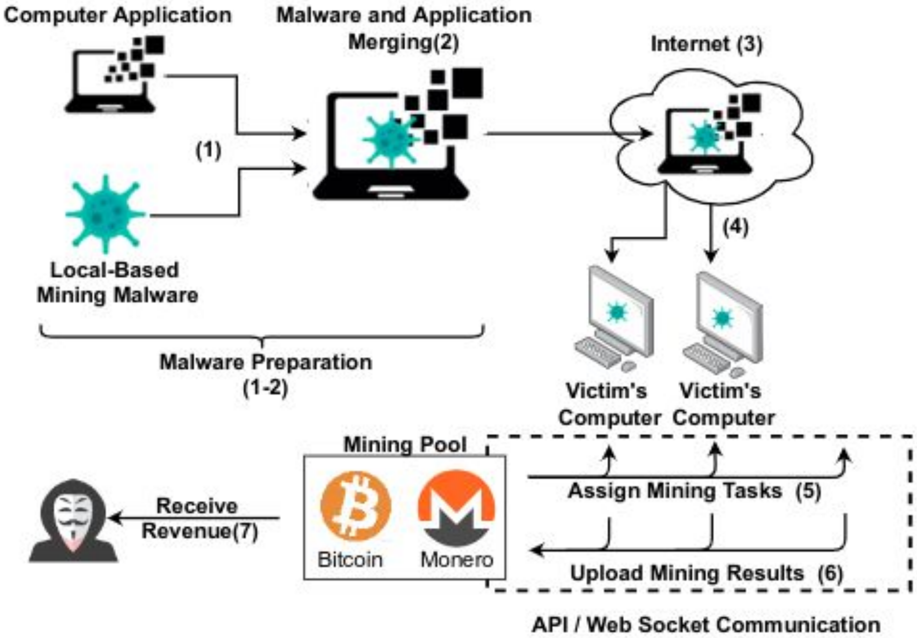
In-browser Cryptojacking

- Takes advantage of interactive web content technologies.
- Connects to victims' host devices to access the computational resources of the victim (e.g., CPU).
- Performs mining as long as the victim keeps the webpage open.



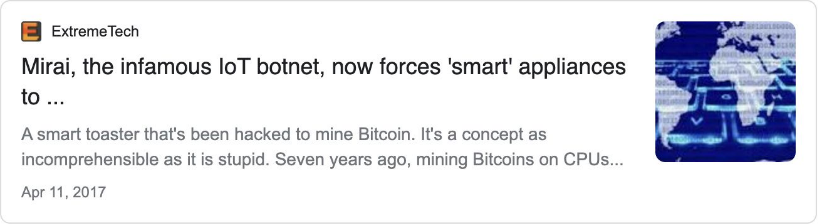
Host-based Cryptojacking

- Turns victims' host devices into a miner for the malware owner.



IoT Cryptojacking

- **New favorite toy** of the attackers.
- **Not individually profitable**.
- **Botnet attacks** to take control of the IoT devices **at scale**.
- Mirai-inspired botnet attacks used this network to mine Bitcoin and turn the botnet network into a **giant cryptojacking mining pool.**


ExtremeTech

Mirai, the infamous IoT botnet, now forces 'smart' appliances to ...

A smart toaster that's been hacked to mine Bitcoin. It's a concept as incomprehensible as it is stupid. Seven years ago, mining Bitcoins on CPUs...

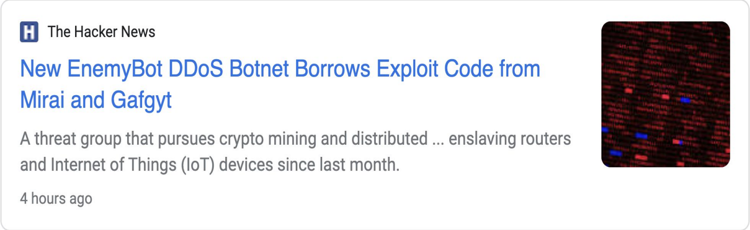
Apr 11, 2017

IoT Cryptojacking

- Another Mirai-inspired botnet, **LIQUOR IoT botnet** started to mine Monero on its victims' IoT devices.
- **BASHLITE** updated with mining and backdoor commands.
- **EnemyBot**, **Spring4Shell**, **Glupteba**, **TickBot** and more IoT botnets are weaponized to mine cryptocurrency.


Bashlite IoT Malware Updated with Mining and Backdoor Commands, Targets WeMo Devices

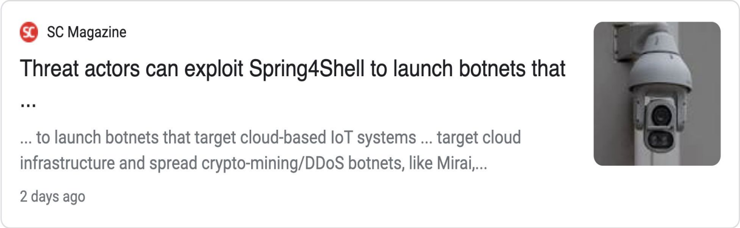
We uncovered an updated Bashlite malware designed to add infected IoT devices to a DDoS botnet. Based on the Metasploit module it exploits, the malware targets devices with the WeMo Universal Plug and Play (UPnP) application programming interface (API).


The Hacker News

New EnemyBot DDoS Botnet Borrows Exploit Code from Mirai and Gafgyt

A threat group that pursues crypto mining and distributed ... enslaving routers and Internet of Things (IoT) devices since last month.

4 hours ago


SC Magazine

Threat actors can exploit Spring4Shell to launch botnets that ...

... to launch botnets that target cloud-based IoT systems ... target cloud infrastructure and spread crypto-mining/DDoS botnets, like Mirai,...

2 days ago

Existing Solutions

- Existing cryptojacking detection methods:
  - <u>Hardware-level features:</u>
    - CPU Events,
    - Memory Activities,
    - Hardware Counters,
    - System Calls.
  - <u>Browser-specific features:</u>
    - JS Compilation Times,
    - Static Source Code Analysis.

Our Approach

- We used **network traffic** because :
  - It does not require devices to be programmed.
  - It can collect the traffic from all device types, communication protocols, hardware types.
  - It works on the encrypted traffic, i.e., **only metadata** is needed.
- It is challenging:
  - **Evasion techniques** such as CPU limiting (i.e., throttle),
  - **Minimized communication** to hide the cryptomining operations,
  - High device **diversity** and **heterogeneous** network traffic.

Thread Model 1

- **<u>Cryptojacking With Service Providers:</u>**
- Service Providers
  - Coinhive, Authedmine, Browsermine, Coinimp, Cryptoloot, DeepMiner, JSECoin, Monerise, Webmine, WebminerPool, Webminepool.
- The attackers merge these framework capabilities with known vulnerabilities and abuse them to run their cryptojacking malware inside of these devices.
- We used **Webmine** and **Webminepool.**

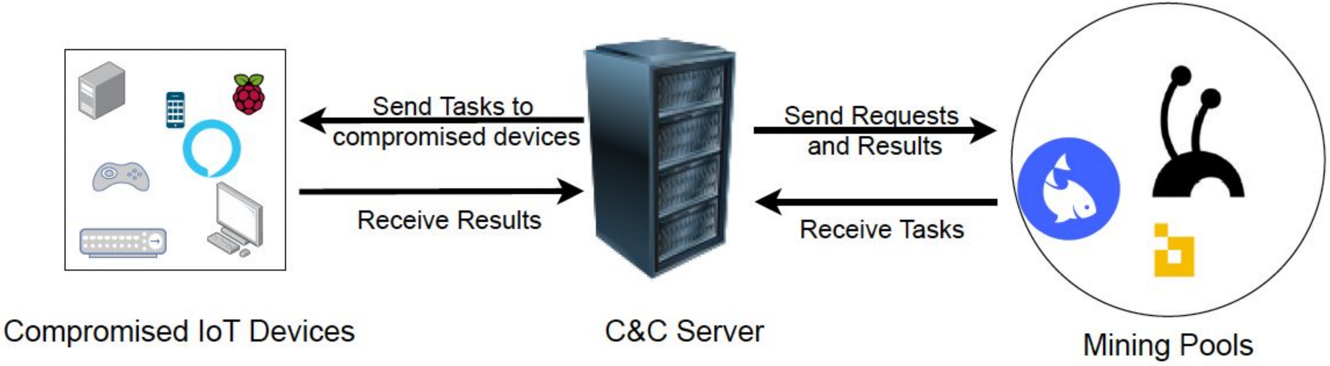
Coinhive

Coinimp

Webminepool

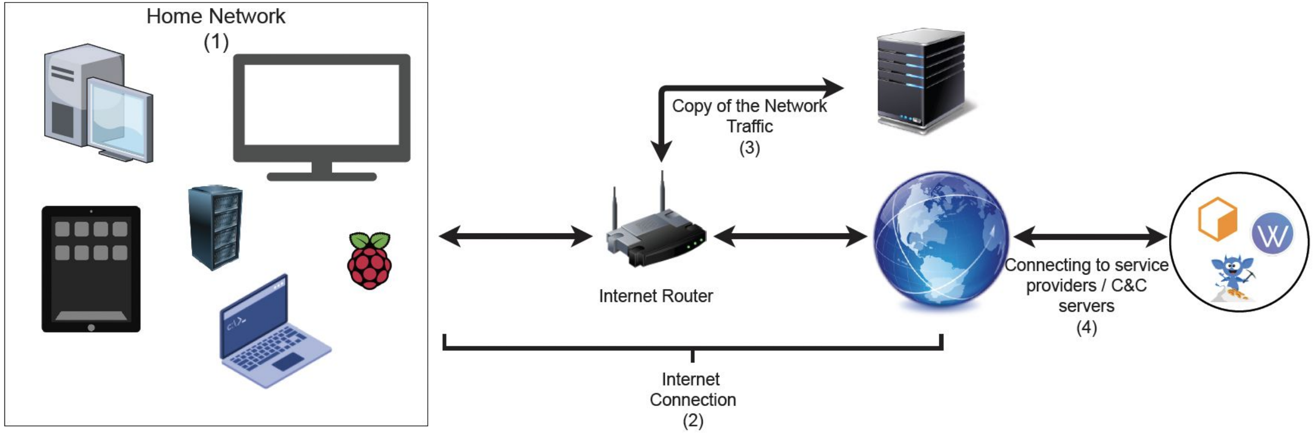
Thread Model 2

- **<u>Cryptojacking With C&C Servers:</u>**
- We focused on the communication pipeline **between the compromised device and the C&C server.**



Setup

- **Regular home-networking settings**.
- All devices are connected to the **same router.**
- One computer is responsible to **collect networking data**.
- **Compromised devices** are also using the network pipeline to connect C&C servers.



Devices

Device	Representation	Hardware	Operating System
Raspberry Pi	IoT Device	Cortex-A72 64-bit SoC 4GB RAM	Raspberry OS
LG Smart TV	IoT Device	LF Quad Core Processor	WebOS 2.0
Laptop	Regular Device	Intel Core i7 9th Generation CPU 16 GB DDR4 RAM	Ubuntu 18.04 LTS
Tower Server	Powerful Device	Intel. Xeon. Gold 6314U Processor 192 GB DDR4 RAM	Ubuntu 20.04
Router	Internet Routing	Atheros QCA9563 Processor	OpenWRT V.19.07.1

Dataset Collection Methodology

- **Same methodology** for benign and malicious data collection.
- ARP poisoning to **re-route** the data communication path.
- **Labelled** the collected networking data during the data collection process.
- Three datasets:
  - Benign Dataset-1
  - Benign Dataset-2
  - Malicious Dataset

Benign Dataset-1

- Downloaded a network data from a public repository:

  https://data.mendeley.com/datasets/5pmnkshffm/1

- It dataset includes following user activities:
  - Interactive
  - Bulk Data Transfer
  - Web Browsing
  - Video Playback
  - Idle Behaviour

Benign Dataset-1

Benign Dataset- 1

Dataset Name	Domain	Total time (Minutes)	Packet Count	Packets Per Second (PPS)	Average Packet Size (Bytes) (APS)
Bulk Data	Internet Data	18	2204727	2636.50	1114.5
Web Multiple	Internet Data	14.56	95388	91.78	567.25
Interactive	Internet Data	20.33	26144	355.97	249
Video	Internet Data	9.55	140009	243.33	956.3333333
Web Single	Internet Data	12.08	51381	71.46	638
Total		**74.52**	**2517649**		

Benign Dataset-2

- Our own benign dataset with the same set of the devices.
- Regular user activities:
  1. Idle Behavior,
  2. Web Browsing,
  3. Watching Video,
  4. Large File Download,
  5. Interactive.
- Only watching video activity from LG Smart TV.
- 16 dataset (3 devices x 5 activities + LG Smart TV).

Benign Dataset

Benign Dataset-2

Dataset Name	Device	Activity	Total time (Minutes)	Packet Count	Packets Per Second (PPS)	Average Package Size (Bytes) (APS)
Laptop_idle_benign	Laptop	Idle	10.24	113602	184.9	929
Laptop_interactive_benign	Laptop	Interactive	22.1	81681	61.6	668
Laptop_webbrowsing_benign	Laptop	Web Browsing	11.43	99235	144.7	764
Laptop_download_benign	Laptop	Download	4.19	442866	1761.6	925
Laptop_video_benign	Laptop	Video	32.45	29010	14.9	1109
Raspberry_idle_benign	Raspberry	Idle	30.25	73	0	113
Raspberry_interactive_benign	Raspberry	Interactive	17.27	104241	100.6	764
Raspberry_webbrowsing_benign	Raspberry	Web Browsing	23.22	123298	88.5	946
Raspberry_download_benign	Raspberry	Download	4.11	276808	1122.5	1267
Raspberry_video_benign	Raspberry	Video	31.26	57205	30.5	1177
Server_idle_benign	Server	Idle	20.21	13459	11.1	142
Server_interactive_benign	Server	Interactive	18.01	123728	114.5	1143
Server_webbrowsing_benign	Server	Web Browsing	14.37	43713	50.7	1233
Server_download_benign	Server	Download	4.15	564831	2268.4	3438
Server_video_benign	Server	Video	14.18	109487	128.7	1069
WebOS_video_benign	WebOS	Livestream and Video	4.07	177704	727.7	930
Total			**261.51**	**2360886**		

Malicious Data Collection

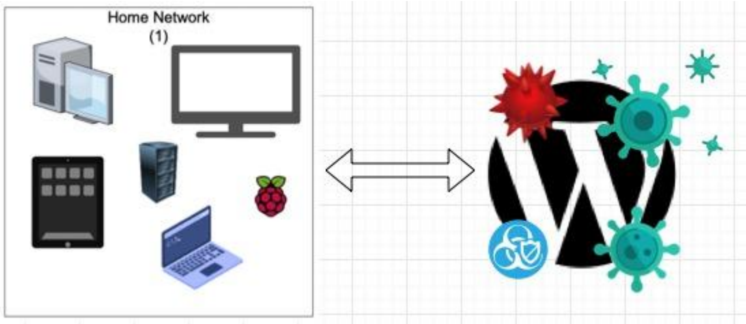
Malicious Data Collection

Implementing
In-Browser Cryptojacking

Implementing
Host-based Cryptojacking

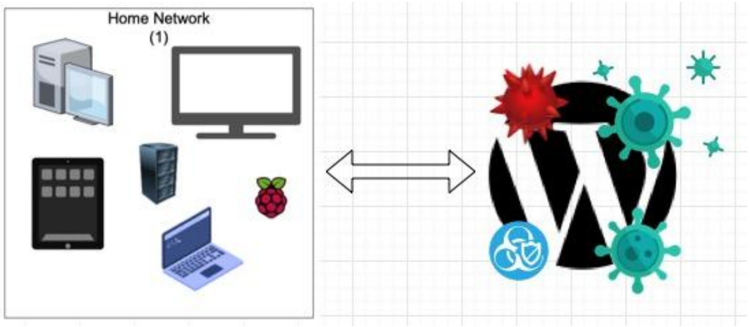
Implementing In-browser Cryptojacking

- In-browser cryptojacking use service providers to **connect and receive mining tasks** and start performing cryptomining.
- We created a **Wordpress webpage server**.
- Cryptojacking malware samples from **different service providers**.



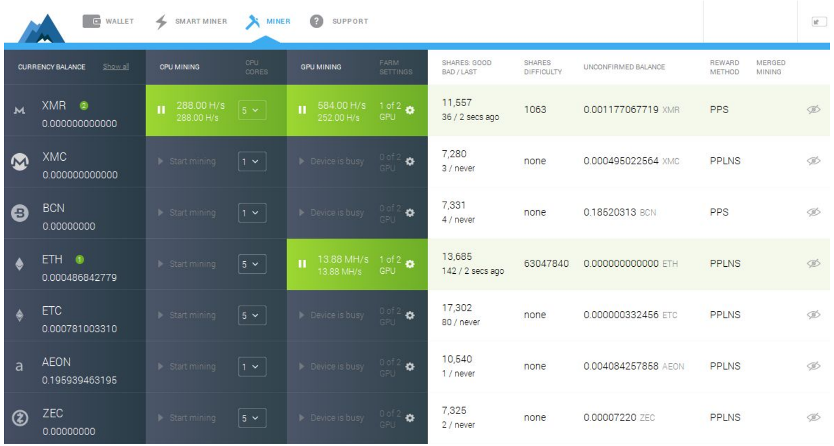
Implementing In-browser Cryptojacking

- LG WebOS operating system **does not support WASM and JS** libraries.
- We used **LG WebOS SDK's cryptographic libraries** to implement cryptojacking on LG devices.
- We collected network traffic data for **at least 12 hours** for every use case scenario.



Implementing Host-based Cryptojacking

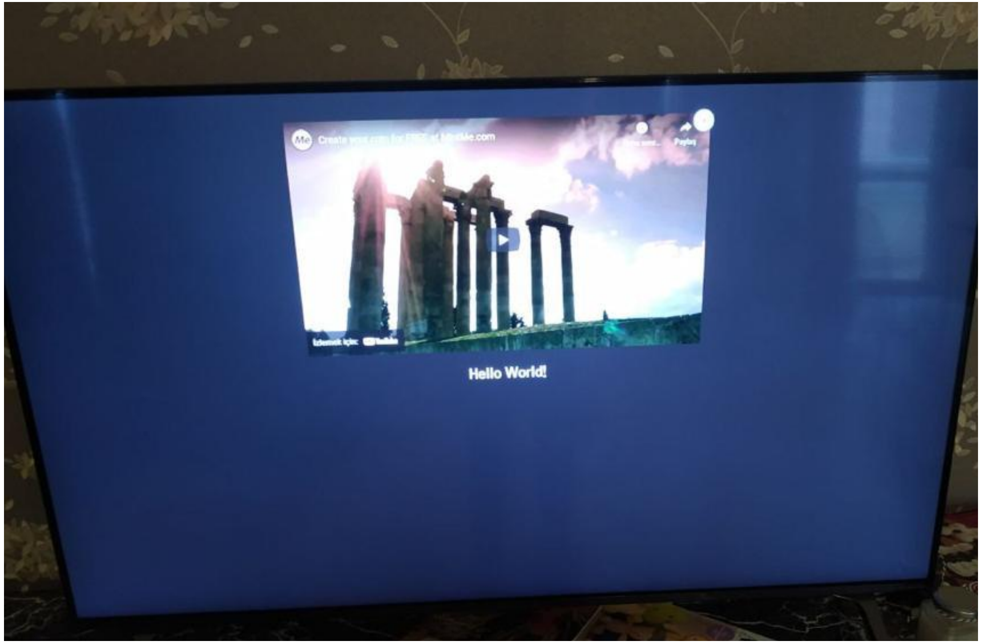
- Implementing on **Raspberry Pi and Server** were straightforward.
    - Downloaded the cryptocurrency mining binary MinerGate V1.7 and run it on our test device.



Implementing Host-based Cryptojacking

- Implementation of host-based cryptojacking on the LG Smart TV is more challenging
- The malware binary needed to be located in a suitable way.
- We used **LG WebOS development framework**.
- We developed **a basic IP TV application** that runs cryptojacking malware as long as the application running.

Implementing Host-based Cryptojacking



Implementing Host-based Cryptojacking

- We implemented the application with two settings;
  - CC server receives the mining tasks from a mining pool:

```
131.94.186.113 - - [23/Mar/2021 13:10:12] "GET /api/hash HTTP/1.1" 200 -
131.94.186.113 - - [23/Mar/2021 13:10:16] "GET /api/hash HTTP/1.1" 200 -
45.146.165.157 - - [23/Mar/2021 13:10:19] "POST /api/jsonws/invoke HTTP/1.1" 404 -
45.146.165.157 - - [23/Mar/2021 13:10:19] "POST /vendor/phpunit/phpunit/src/Util/PHP/eval-stdin.php HTTP/1.1" 404 -
131.94.186.113 - - [23/Mar/2021 13:10:19] "GET /api/hash HTTP/1.1" 200 -
45.146.165.157 - - [23/Mar/2021 13:10:19] "GET /vendor/phpunit/phpunit/src/Util/PHP/eval-stdin.php HTTP/1.1" 404 -
45.146.165.157 - - [23/Mar/2021 13:10:19] "GET /solr/admin/info/system?wt=json HTTP/1.1" 404 -
45.146.165.157 - - [23/Mar/2021 13:10:20] "GET /index.php?s=/Index/\think\app/invokefunction&function=call_user_func_a
rray&vars[0]=md5&vars[1][]=HelloThinkPHP21 HTTP/1.1" 404 -
45.146.165.157 - - [23/Mar/2021 13:10:20] "GET /?a=fetch&content=<php>die(@md5(HelloThinkCMF))</php> HTTP/1.1" 404 -
45.146.165.157 - - [23/Mar/2021 13:10:20] "GET /?XDEBUG_SESSION_START=phpstorm HTTP/1.1" 404 -
45.146.165.157 - - [23/Mar/2021 13:10:20] "GET /console/ HTTP/1.1" 404 -
45.146.165.157 - - [23/Mar/2021 13:10:20] "POST /Autodiscover/Autodiscover.xml HTTP/1.1" 404 -
45.146.165.157 - - [23/Mar/2021 13:10:21] "GET /wp-content/plugins/wp-file-manager/readme.txt HTTP/1.1" 404 -
45.146.165.157 - - [23/Mar/2021 13:10:21] "GET /_ignition/execute-solution HTTP/1.1" 404 -
```

Implementing Host-based Cryptojacking

- CC server runs its own node to create mining tasks:

```
th-node_1 | INFO [12-09|21:36:29.403] Imported new block receipts   count=974  elapsed=67.202ms   number=2,914,297 hash=a060c8..e0a56a age=3y3mo3w  size=3.37MiB
th-node_1 | INFO [12-09|21:36:29.459] Imported new state entries    count=1536 elapsed=10.792ms   processed=9,810,636 pending=138,227 trieretry=0  coderetry=0  duplicate=1314 unexpected=8546
th-node_1 | INFO [12-09|21:36:29.754] Imported new state entries    count=1920 elapsed=9.614ms    processed=9,812,556 pending=137,429 trieretry=0  coderetry=0  duplicate=1314 unexpected=8546
th-node_1 | INFO [12-09|21:36:29.936] Imported new state entries    count=1152 elapsed=1.920ms    processed=9,813,708 pending=138,791 trieretry=0  coderetry=0  duplicate=1314 unexpected=8546
th-node_1 | INFO [12-09|21:36:30.058] Imported new state entries    count=1152 elapsed=2.350ms    processed=9,814,860 pending=138,920 trieretry=0  coderetry=0  duplicate=1314 unexpected=8546
th-node_1 | INFO [12-09|21:36:30.269] Imported new block headers    count=2048 elapsed=241.406ms  number=2,951,552 hash=4b5db4..b89b98 age=3y3mo2w
th-node_1 | INFO [12-09|21:36:30.510] Imported new block headers    count=2048 elapsed=232.629ms  number=2,953,600 hash=c81fa3..5a2a85 age=3y3mo2w
th-node_1 | INFO [12-09|21:36:30.534] Imported new state entries    count=2304 elapsed=4.166ms    processed=9,817,164 pending=137,184 trieretry=0  coderetry=0  duplicate=1314 unexpected=8546
th-node_1 | INFO [12-09|21:36:30.598] Imported new block receipts   count=661  elapsed=52.150ms   number=2,914,958 hash=e8153c..b86c58 age=3y3mo3w  size=2.04MiB
th-node_1 | INFO [12-09|21:36:30.743] Imported new state entries    count=1152 elapsed=2.392ms    processed=9,818,316 pending=137,026 trieretry=0  coderetry=0  duplicate=1314 unexpected=8546
th-node_1 | INFO [12-09|21:36:30.845] Imported new state entries    count=1152 elapsed=5.262ms    processed=9,819,468 pending=138,178 trieretry=0  coderetry=0  duplicate=1314 unexpected=8546
th-node_1 | INFO [12-09|21:36:31.041] Imported new state entries    count=1152 elapsed=1.308ms    processed=9,820,620 pending=139,356 trieretry=0  coderetry=0  duplicate=1314 unexpected=8546
th-node_1 | INFO [12-09|21:36:31.126] Imported new state entries    count=1536 elapsed=2.974ms    processed=9,822,156 pending=138,565 trieretry=0  coderetry=0  duplicate=1314 unexpected=8546
th-node_1 | INFO [12-09|21:36:31.506] Imported new state entries    count=1152 elapsed=2.768ms    processed=9,823,308 pending=139,493 trieretry=0  coderetry=0  duplicate=1314 unexpected=8546
th-node_1 | INFO [12-09|21:36:31.560] Imported new block receipts   count=2048 elapsed=126.151ms  number=2,917,006 hash=2e1078..54cb14 age=3y3mo3w  size=6.69MiB
th-node_1 | INFO [12-09|21:36:31.585] Imported new state entries    count=1920 elapsed=5.036ms    processed=9,825,228 pending=138,168 trieretry=0  coderetry=0  duplicate=1314 unexpected=8546
```

Malicious Dataset

Malicious Samples

Dataset Name	Cryptojacking Type	Device	Software	Attacker	Currency	Total time (Minutes)	Packet Count	Packets Per Second (PPS)	Average Packet Size (Bytes) (APS)
Raspberry_Webmine.io_Robust	In-browser	Raspberry Pi 4	Webmine.io	Robust	Monero	52	3621	1.2	**479**
Raspberry_Webmine.io_Aggressive	In-browser	Raspberry Pi 4	Webmine.io	Aggressive	Monero	735	14156	0.3	**163**
Raspberry_WebminePool_Stealthy	In-browser	Raspberry Pi 4	WebminePool	Stealthy	Monero	521	10285	0.3	146
Raspberry_WebminePool_Robust	In-browser	Raspberry Pi 4	WebminePool	Robust	Monero	527	7708	0.20	141
Raspberry_WebminePool_Aggressive	In-browser	Raspberry Pi 4	WebminePool	Aggressive	Monero	1080	24476	0.40	145
Server_WebminePool_Robust	In-browser	Server	WebminePool	Robust	Monero	382	18460	0.8	498
Server_WebminePool_Aggressive	In-browser	Server	WebminePool	Aggressive	Monero	60	3106	0.9	297
Desktop_WebminePool_Aggressive	In-browser	Desktop	WebminePool	Aggressive	Monero	726	234892	5.4	3128
Raspberry_Binary	Host-based	Raspberry Pi 4	MinerGate	Aggressive	Monero	983	22111	0.4	95
Server_Binary	Host-based	Server	MinerGate	Aggressive	Monero	1024	1213354	**19.7**	154
WebOS	Host-based	LG Smart TV	AntMiningPool	Aggressive	Monero	61	43173	**11.80**	242
Total				**6145**	**1558831**				

Initial Observations

- The highest **malicious PPS and APS** rates << The highest **benign PPS and APS** rates.
- Very small amount of PPS rate and APS rate for in-browser mining.
- **Binary mining** samples do not have any intonation to minimize their communication.
- For binary mining, the APS and PPS rates are **directly correlated with the computational power of the device**.
- **All device types** give almost t**he same PPS and APS** rates for in-browser mining applications.

Evaluation Methodology

- Four sets of experiments:
  1. IoT cryptojacking detection mechanism using Machine Learning
  2. Different adversarial behaviors
  3. Various smart home network settings
  4. Sensitivity of the classifier

IoT Cryptojacking Detection

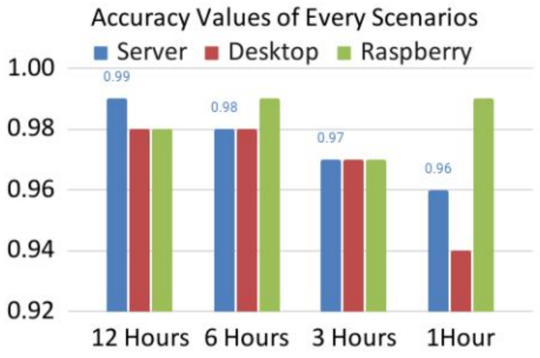
- **Designing the optimum IoT cryptojacking detection mechanism using Machine Learning** (Scenario 0)
  - Feature Extraction
  - Feature Selection
  - Best-performing Classifier
  - Varying Training Sizes

IoT Cryptojacking Detection - Results

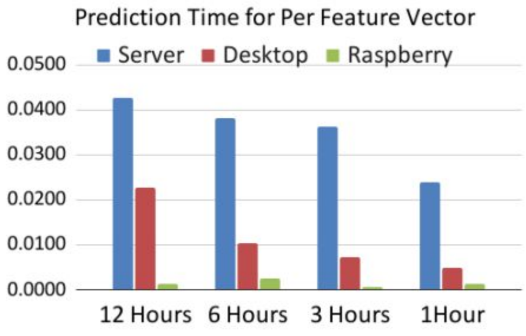
Classifier	Accuracy	Precision	Recall	F1 Score	Test ROC
Logreg	0.97	0.97	0.97	0.97	0.988
KNN	0.98	0.98	0.98	0.98	0.99
SVM	0.99	0.98	0.98	0.98	0.99
GNB	0.96	0.96	0.96	0.96	0.97

Dataset		**Dataset Sample Sizes**			
		12 hours	**6 Hours**	**3 Hours**	**1 Hour**
Server	Malicious	838627	419313	209656	69885
	Benign	837701	418850	209425	69808
Desktop	Malicious	234272	117136	58568	19522
	Benign	234448	117224	58612	19537
Raspberry	Malicious	7829	3914	1957	978
	Benign	8265	4132	2066	1033

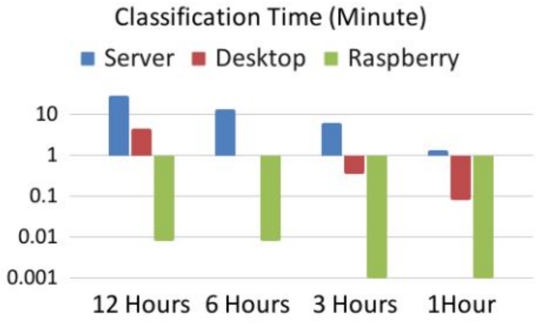
IoT Cryptojacking Detection - Results



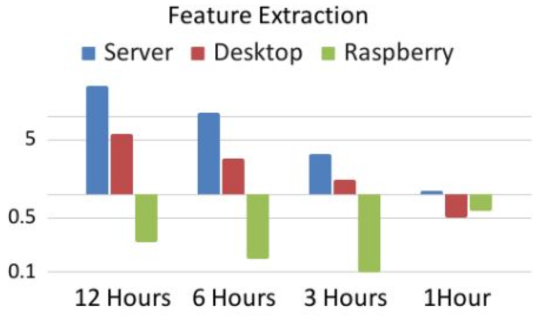
(a)



(b)



(c)



(d)

Adversarial Behavior

- **Different adversarial behaviors**
  - Victim Device Type (Scenario 1)
    - Server vs. Desktop vs. IoT
  - Profit Strategies (Scenario 2)
    - Aggressive vs. Robust vs. Stealthy
  - Cryptojacking Type (Scenario 3)
    - In-browser vs. Host-based

Adversarial Behavior - Results

	Attack Case	Accuracy	Precision	Recall	F1 Score	Test ROC
Scenario 1	Server	0.99	0.99	0.99	0.99	0.99
	Desktop	0.98	0.98	0.98	0.98	0.99
	IoT	0.93	0.93	0.93	0.93	0.96
Scenario 2	Aggressive	0.98	0.98	0.98	0.98	0.99
	Robust	0.87	0.87	0.87	0.87	0.94
	Stealthy	0.91	0.92	0.91	0.91	0.98
Scenario 3	In-Browser	0.95	0.95	0.95	0.95	0.98
	Host-Based	0.99	0.99	0.99	0.99	0.99

Smart Home Settings

- **Various smart home network settings**
  - Fully Compromised (Scenario 4)
    - Overall
  - Partially Compromised (Scenario 5)
    - IoT + Laptop
  - Single Device Compromised (Scenario 6)
    - IoT
  - IoT Compromised (Scenario 7)
    - IoT + IoT

Smart Home Settings - Results

	Test Case	Accuracy	Precision	Recall	F1-Score	Test ROC
Scenario 4	Fully compromised (Overall)	0.98	0.98	0.98	0.98	0.99
Scenario 5	Partially compromised (IoT + Laptop)	0.98	0.98	0.98	0.98	0.99
Scenario 6	Single compromised (IoT)	0.94	0.94	0.94	0.94	0.95
Scenario 7	IoT compromised (IoT + IoT)	0.92	0.92	0.92	0.92	0.96

Classifier Sensitivity

- **The sensitivity of the proposed classifier**
  - Imbalance Dataset (Scenario 8)
    - Timely Balanced
    - Timely Balanced with Oversampling
    - Same Device
  - Classifier Transferability (Scenario 9)
    - Service Provider
    - Device Type
    - Cryptojacking Type
  - Non-default Parameters (Scenario 10)

Classifier Sensitivity - Results

	Attack Case		Accuracy	Precision	Recall	F1-Score	Test ROC
Scenario 8	Timely Balanced		0.99	0.99	0.99	0.99	0.96
	Timely Balanced with Oversampling		0.98	0.98	0.98	0.98	0.98
	Same Device	Server vs. Server	0.98	0.98	0.98	0.98	0.99
		Laptop vs. Laptop	0.99	0.99	0.99	0.99	0.99
		Raspberry vs. Raspberry	0.97	0.97	0.97	0.97	0.96
		WebOS vs. WebOS	0.97	0.97	0.97	0.97	0.99

	Attack Case	Accuracy	Precision	Recall	F1-Score	Test ROC
Scenario 9	Service Provider-1	0.87	0.92	0.87	0.88	0.93
	Service Provider-2	0.69	0.92	0.69	0.75	0.97
	Binary-1	0.87	0.84	0.87	0.81	0.99
	Binary - In-Browser - 1	0.90	0.90	0.90	0.89	0.99
	Binary - In-Browser - 2	0.99	0.99	0.99	0.99	0.99
	Binary - In-Browser - 3	0.99	0.99	0.99	0.99	0.99
	In-Browser - 1	0.99	0.99	0.99	0.99	0.99
	In-Browser - 2	0.97	0.96	0.97	0.96	0.99

Classifier Sensitivity - Results

	Classifier (SVM)			Accuracy	Precision	Recall	F1-Score	Test_ROC
	Kernel	C	Gamma					
Scenario 10	Linear	1	Scale	1.0	1.0	1.0	1.0	1.0
	Poly	1	Scale	0.83	0.83	0.83	0.83	0.92
	RBF	1	Scale	0.83	0.84	0.83	0.83	0.91
	Sigmoid	1	Scale	0.72	0.72	0.72	0.72	0.76
	Linear	1	Auto	1.0	1.0	1.0	1.0	1.0
	Poly	1	Auto	0.88	0.88	0.88	0.88	0.93
	RBF	1	Auto	0.66	0.80	0.66	0.61	0.70
	Sigmoid	1	Auto	0.52	0.27	0.52	0.35	0.5
	Linear	2	Scale	1.0	1.0	1.0	1.0	1.0
	Poly	2	Scale	0.84	0.84	0.84	0.84	0.82
	RBF	2	Scale	0.87	0.88	0.87	0.87	0.92
	Sigmoid	2	Scale	0.73	0.73	0.73	0.73	0.76
	Linear	2	Auto	1.0	1.0	1.0	1.0	1.0
	Poly	2	Auto	0.88	0.88	0.88	0.88	0.93
	RBF	2	Auto	0.66	0.80	0.66	0.61	0.70
	Sigmoid	2	Auto	0.52	0.27	0.52	0.35	0.50

Experimental Challenges

- Implementing cryptojacking malware on LG WebOS
- Expansion to the other devices
  - Amazon Echo
  - Apple HomePod
  - Philips Hue Environment
- ML training on a huge volume of data
- IP → MAC

Code Snippets

```
###################################################################
#                                                                 #
#                malicious csv files import                       #
#                                                                 #
###################################################################

df1 = pd.read_csv('/malicious/WebOS_binary.csv') #
df2 = pd.read_csv('/malicious/Server_Binary.csv') #
df3 = pd.read_csv('/malicious/Raspberry_Webmine_Robust.csv')
df4 = pd.read_csv('/malicious/Raspberry_Binary.csv') #
df5 = pd.read_csv('/malicious/Raspberry_Webmine_Aggressive.csv')
df6 = pd.read_csv('/malicious/Raspberry_WebminePool_Aggressive.csv')
df7 = pd.read_csv('/malicious/Server_WebminePool_Aggressive.csv') #

df32 = pd.read_csv('/malicious/Server_WebminePool_Robust.csv') #
df33 = pd.read_csv('/malicious/Raspberry_WebminePool_Stealthy.csv') #
df34 = pd.read_csv('/malicious/Raspberry_WebminePool_Robust.csv') #
df35 = pd.read_csv('/malicious/Desktop_WebminePool_Aggressive.csv') #


###################################################################
#                                                                 #
#                benign csv files import                          #
#                                                                 #
###################################################################

############### LAPTOP #############

df8 = pd.read_csv('/benign-2/Laptop/Laptop_download_benign.csv')
df9 = pd.read_csv('/benign-2/Laptop/Laptop_idle_benign.csv')
df10 = pd.read_csv('/benign-2/Laptop/Laptop_interactive_benign.csv')
df11 = pd.read_csv('/benign-2/Laptop/Laptop_video_benign.csv')
df12 = pd.read_csv('/benign-2/Laptop/Laptop_webbrowsing_benign.csv')

############### Raspberry ##########

df13 = pd.read_csv('/benign-2/Raspberry/Raspberry_download_benign.csv')
df14 = pd.read_csv('/benign-2/Raspberry/Raspberry_idle_benign.csv')
df15 = pd.read_csv('/benign-2/Raspberry/Raspberry_interactive_benign.csv')
df16 = pd.read_csv('/benign-2/Raspberry/Raspberry_video_benign.csv')
df17 = pd.read_csv('/benign-2/Raspberry/Raspberry_webbrowsing_benign.csv')

############### Server ############


df18 = pd.read_csv('/benign-2/Server/Server_download_benign.csv')
df19 = pd.read_csv('/benign-2/Server/Server_idle_benign.csv')
df20 = pd.read_csv('/benign-2/Server/Server_interactive_benign.csv')
df21 = pd.read_csv('/benign-2/Server/Server_video_benign.csv')
df22 = pd.read_csv('/benign-2/Server/Server_webbrowsing_benign.csv')
```

Code Snippets

Benign: 2360951

In [30]:

```
# Prune the datasets for labeling process for malicious data


# For WebOS = 18:56:80:17:d0:ef
index_names = df1[((df1['HW_dst'] != '18:56:80:17:d0:ef') & (df1['Hw_src'] != '18:56:80:17:d0:ef'))].index
df1.drop(index_names, inplace = True)

# Big_Server_Monero_mining_data = a4:bb:6d:ac:e1:fd

index_names = df2[((df2['HW_dst'] != 'a4:bb:6d:ac:e1:fd') & (df2['Hw_src'] != 'a4:bb:6d:ac:e1:fd'))].index
df2.drop(index_names, inplace = True)

# ege_data_rasberry = dc:a6:32:67:66:4b

index_names = df3[((df3['HW_dst'] != 'dc:a6:32:67:66:4b') & (df3['Hw_src'] != 'dc:a6:32:67:66:4b'))].index
df3.drop(index_names, inplace = True)

# Rasberry_binary_monero_mining = dc:a6:32:68:35:8a

index_names = df4[((df4['HW_dst'] != 'dc:a6:32:68:35:8a') & (df4['Hw_src'] != 'dc:a6:32:68:35:8a'))].index
df4.drop(index_names, inplace = True)

# Rasberry_network_data_2 = dc:a6:32:67:66:4b

index_names = df5[((df5['HW_dst'] != 'dc:a6:32:67:66:4b') & (df5['Hw_src'] != 'dc:a6:32:67:66:4b'))].index
df5.drop(index_names, inplace = True)

# Rasberry-Webmine = dc:a6:32:67:66:4b
index_names = df6[((df6['HW_dst'] != 'dc:a6:32:67:66:4b') & (df6['Hw_src'] != 'dc:a6:32:67:66:4b'))].index
df6.drop(index_names, inplace = True)

# Server_Webmine_Network_data = a4:bb:6d:ac:e1:fd

index_names = df7[((df7['HW_dst'] != 'a4:bb:6d:ac:e1:fd') & (df7['Hw_src'] != 'a4:bb:6d:ac:e1:fd'))].index
df7.drop(index_names, inplace = True)

# Server_%50_Mining = a4:bb:6d:ac:e1:fd

index_names = df32[((df32['HW_dst'] != 'a4:bb:6d:ac:e1:fd') & (df32['Hw_src'] != 'a4:bb:6d:ac:e1:fd'))].index
df32.drop(index_names, inplace = True)

# Rasberry_webmine_%10 = dc:a6:32:67:66:4b

index_names = df33[((df33['HW_dst'] != 'dc:a6:32:67:66:4b') & (df33['Hw_src'] != 'dc:a6:32:67:66:4b'))].index
df33.drop(index_names, inplace = True)

# Rasberry_webmine_%50 = dc:a6:32:68:35:8a

index_names = df34[((df34['HW_dst'] != 'dc:a6:32:68:35:8a') & (df34['Hw_src'] != 'dc:a6:32:68:35:8a'))].index
```

Code Snippets

```python
print("After droppping NAN rows: ")
print("malicious: {}".format(len(df_malicious)))
print("benign: {}".format(len(df_benign)))

start = timer()

results_all_combined_imbalanced = run_process(df_malicious,df_benign,df_results)

end = timer()
print(end - start)
```

```
malicious: 9741
benign: 1634689
0 NAN in malicious!
0 NAN in benign!
After droppping NAN rows:
malicious: 9741
benign: 1634689
Feature Extraction: 100%|█████████████████████| 140/140 [00:02<00:00, 59.85it/s]
Feature Extraction: 100%|█████████████████████| 160/160 [05:24<00:00,  2.03s/it]
let the ml starts
SVM
              precision    recall  f1-score   support

   malignant       0.99      1.00      1.00     40881
      benign       1.00      0.01      0.03       230

    accuracy                           0.99     41111
   macro avg       1.00      0.51      0.51     41111
weighted avg       0.99      0.99      0.99     41111

     fit_time  score_time  test_accuracy  test_precision_weighted  \
0   31.783958   15.123198       0.994568                 0.991921
1   29.676754   14.451717       0.993351                 0.986747
2   31.666967   15.263540       0.994608                 0.989245
3   29.483381   14.580888       0.993838                 0.993876
4  327.497598   14.551876       0.993473                 0.986988

   test_recall_weighted  test_f1_weighted  test_roc_auc model
0              0.994568          0.991939      0.944027   SVM
1              0.993351          0.990038      0.960835   SVM
2              0.994608          0.991920      0.963235   SVM
3              0.993838          0.990806      0.953221   SVM
4              0.993473          0.990220      0.960038   SVM
1343.248294252
```

Concluding Remarks

- A novel, accurate, and robust IoT-based cryptojacking detection system
- Designed **novel experiment scenarios.**
- Different **adversarial behaviors.**
- Different **network settings.**
- **The dataset and code are publicly available in:**
    - github.com/cslfiu/IoTCryptojacking

Q&A - Thanks!

- Lab: csl.fiu.edu
- github.com/cslfiu/IoTCryptojacking

Ege Tekiner
eteki001@fiu.edu
egetekiner.com

Abbas Acar
aacar001@fiu.edu
web.eng.fiu.edu/aacar

Selcuk Uluagac
suluagac@fiu.edu
web.eng.fiu.edu/selcuk

This work was supported by the U.S. National Science Foundation (Award #: NSF-CAREER CNS-1453647).