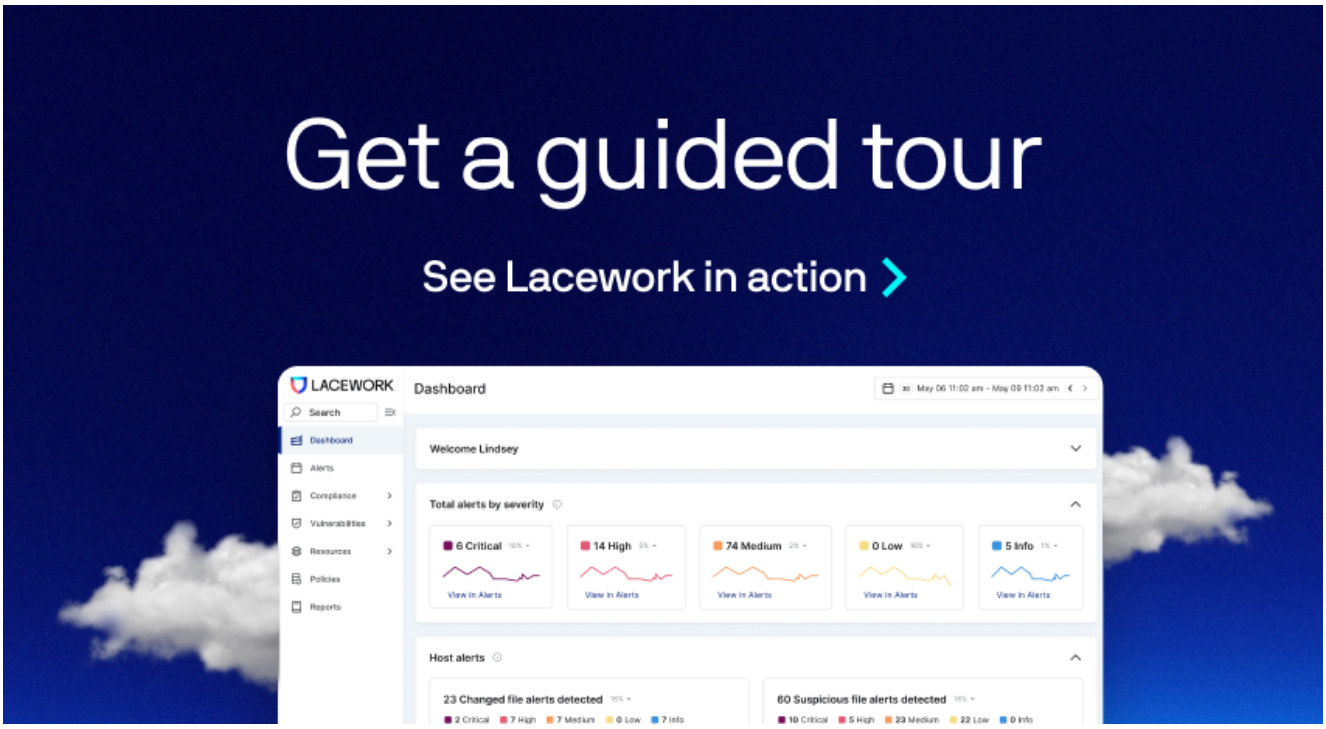
The “Kek Security” Network

lacework.com/blog/the-kek-security-network/

March 18, 2021



Key takeaways:

- Keksec has updated their tactics to include use of DGA and Tor C2, and proxies.
- Based on voluminous polymorphic specimens there is likely widespread infection attributable to the group.
- Keksec is actively exploiting Citrix NetScaler RCE – CVE-2019-19781 and Vmware vCenter Server – CVE-2021-21973
- Tools and indicators are available **here.**

Introduction

Kek Security/Keksec is a prolific threat actor group that was recently detailed in **Checkpoint** (Freakout) and **Netlab360** (Necro) reporting. Keksec exploits several vulnerabilities and targets multiple architectures with polymorphic tools including Linux and Windows payloads, and custom python malware. The group is actively constructing IRC botnets for the purposes of DDoS operations and cryptojacking campaigns using both Doge and Monero.

This blog details the new tools and tactics leveraged by Keksec and includes persona information on the actors. Analysis tools and indicators can be found in the Lacework Labs **Github**.

Kaiten Variants

Beginning as early as August 2020, Keksec started leveraging polymorphic variants of the **Kaiten** IRC DDoS bot. This was observed on VirusTotal with over 1800 uploads of the malware at the time of this blog. The polymorphic nature of the malware renders hash-checks useless and requires a scan to properly identify. This means that the number of VirusTotal uploads for a given variant is a reflection of the malware's footprint in the wild. Analysis of submission data showed 46 unique submitter keys in 23 countries. The ownership of the submitter keys is not disclosed in VirusTotal, so it's unknown how many belong to AV vendors or unique victims.

Variants of the Kaiten source are available on GitHub, however the exact source for the Keksec variants are not.
Some of the specimens were configured to exploit different vulnerabilities including the Citrix NetScaler RCE – CVE-2019-19781 and other IoT exploits often seen in Mirai. Kaiten uses a substitute cipher to encode sensitive strings such as commands and c2 endpoints. Analysis of the Keksec samples revealed the use of two custom ciphers, the second of which was configured in late February.

Keksec's Kaiten cipher	Timeline
xm@_;w,B-Z*j? nvE|sq1o$3″7zKC<F)utAr.p%=>4ihgfe6cba~&5Dk2d!8+9Uy:	August 2020 – February 2021
%q*KC)&F98fsr2to4b3yi_:wB>z=;!k?”EAZ7.D- md<ex5U~h,j|$v6c1ga+p@un	February 2021 – present

The following python code may be used to decode Kaiten strings, the inputs and the cipher value just needs to changed for the respective specimen:

```
encstring_ = "?>K!tF>iorZ:ww_uBw3Bw"
cipher_ = {}

encode = list('%q*KC)&F98fsr2to4b3yi_:wB>z=;!k?"EAZ7.D-
md<ex5U~h,j|$v6c1ga+p@un')
decode =
list('0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ. ')

temp = zip(encode,decode)
for pairs in temp:
    cipher_[pairs[0]] = pairs[1]


decode_temp = []

for s in encstring_:
    try:
        d = cipher_[s]
        decode_temp.append(d)
    except:
        raise

decoded = ''.join(decode_temp)
print(decoded)
```

A significant tactic change employed with the latest Kaiten specimens, was the use of a Tor. This is achieved with an .onion domain **vp3te7pkfczmnnl.onion**, as opposed to hardcoded IPs seen in earlier specimens. The following table shows all observed c2s with the latest cipher (%q*KC)&F98fsr2to4b3yi_:wB>z=;!k?”EAZ7.D-md<ex5U~h,j|$v6c1ga+p@un).

Encoded c2	Decoded	count
?>K!tF>iorZ:ww_uBw3Bw	vp3te7pkfczmnnl.onion	126
C)uqC)uq9)u9K	45.145.185.83	6
C)uq)Ku*%Kuq*C	45.153.203.124	7

Since February, there have been over 120 observed specimens using the .onion c2. These newer binaries are being served off several IPs belonging to DediPath and ColoCrossing which are also two of the top sources of Mirai malware.

Malware URL	ASN
http://45.153.203.242/bins/fagbinz.sh4	DediPath
http://45.153.203.124/S1eJ3/lPxdChtp3zsh4	DediPath
http://172.245.36.128/S1eJ3/lPxdChtp3zx86	ColoCrossing
http://45.144.225.96/S1eJ3/IObeENwjm68k	DediPath (irc.kek.org)
http://45.145.185.83/S1eJ3/IObeENwjsh4	DediPath (irc.kek.org)
http://45.144.225.65/S1eJ3/IObeENwjx86	DediPath

Two of the malware distribution IPs (45.144.225.96 & 45.145.185.83) are also part of **Keksec’s IRC infrastructure** as shown with the irc.keksec.org banner in Shodan (Figure 1). **Note:** keksec.org is only the name for the IRC server is not a registered domain. On one of the endpoints an XMRig installation script was also captured by Shodan scanners.

Figure 1 – Keksec Banners – Shodan

Similar to TeamTNT, Keksec has self-attributed their attacks in the past by using their name in the malware and infrastructure. Despite this, the new Kaiten specimens were not given 'keksec' names, nor were there any Keksec static artifacts. The following table shows historic examples:

Type	Examples
filenames	ayylmao420kekuaintgettindesebinssh4 keksec.ppc keksec.mips
Network	irc.kek.org kek.gay
Strings	keksec keksec rox keksec ROX

Necro

Simultaneously with the new Tor functionality in the Kaiten binaries, Keksec also started using Tor proxies in the custom Necro payloads. Necro is the name for Keksec's obfuscated python IRC malware, with the latest version documented by **Net360**. Key features of note in latest Necro variants:

- Installs Tor, leverages hardcoded proxies
  - A full list of proxies is provided in our indicator list

- Targets both Windows and Linux
  - Rootkit functionality for Windows
- Exploits Laravel, Weblogic vulnerabilities
  - CVE-2020-14882
  - CVE-2021-3129
- Employs new DGA with NO-IP.COM Dynamic domains

Necro is characterized by its unique obfuscations which leverages a combination of zlib compression and a multibyte XOR key. These samples can be decoded with a **script** provided by Lacework Labs.

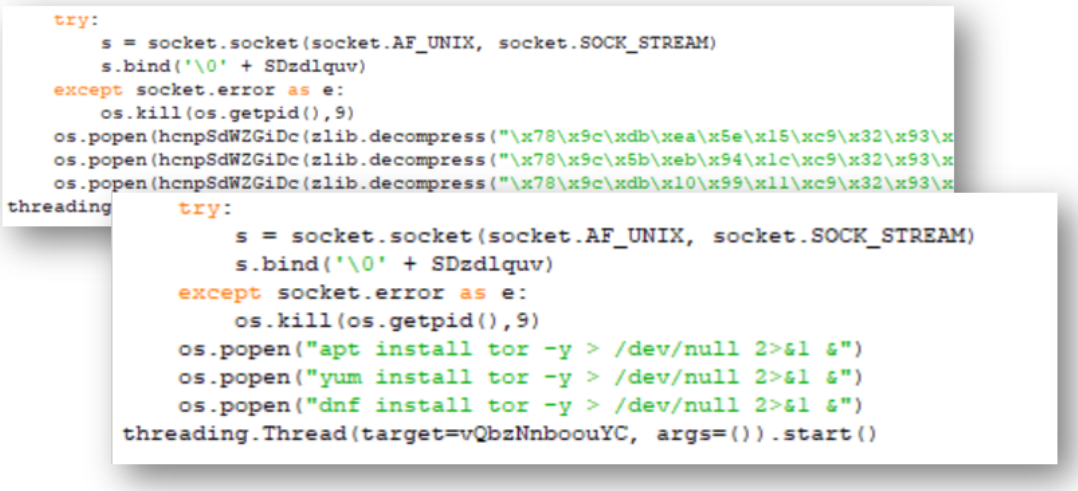


Figure 2 – Before and after example – ZLIB + XOR obfuscation

Following deobfuscation, the proxy configurations and Tor installation commands are easily observable. Similarly, the DGA algorithm is identifiable. The following python generates all possible Necro **DGA domains**. To date, only domain ntxkg0la99w.zapto.org has been seen in the wild.

```
import random

counter_=0

while 1:
    if counter_>=0xFF:
        break
    counter_ +=1

    random.seed(a=0x7774DEAD + counter_)

    dgadomain_=
(''.join(random.choice("abcdefghijklmnopqoasadihcouvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456
789") for _ in range(random.randrange(10,19)))).lower()

dgadomain_+="."+random.choice(["ddns.net","ddnsking.com","3utilities.com","bounceme.net",

"freedynamicdns.net","freedynamicdns.org","gotdns.ch","hopto.org",

"myddns.me","myftp.biz","myftp.org","myvnc.com","onthewifi.com",

"redirectme.net","servebeer.com","serveblog.net","servecounterstrike.com",

"serveftp.com","servegame.com","servehalflife.com","servehttp.com",

"serveirc.com","serveminecraft.net","servemp3.com","servepics.com",

"servequake.com","sytes.net","viewdns.net","webhop.me","zapto.org"])
    print(dgadomain_)
```

Personas

Kek Security is comprised of at least 4 individuals, the most renown of which goes by “Freak”. Handles for other members include “horsewithnoname”, “Tyrant”, and “Moony”. Freak is the author of the Necro malware which dates as far back as 2015. He also maintains a GitHub with various repositories including those with an older **Necro** version, a Windows rootkit loader, darkIrc source code, and exploit code for the recently disclosed **vulnerability** affecting VMware’s vCenter.

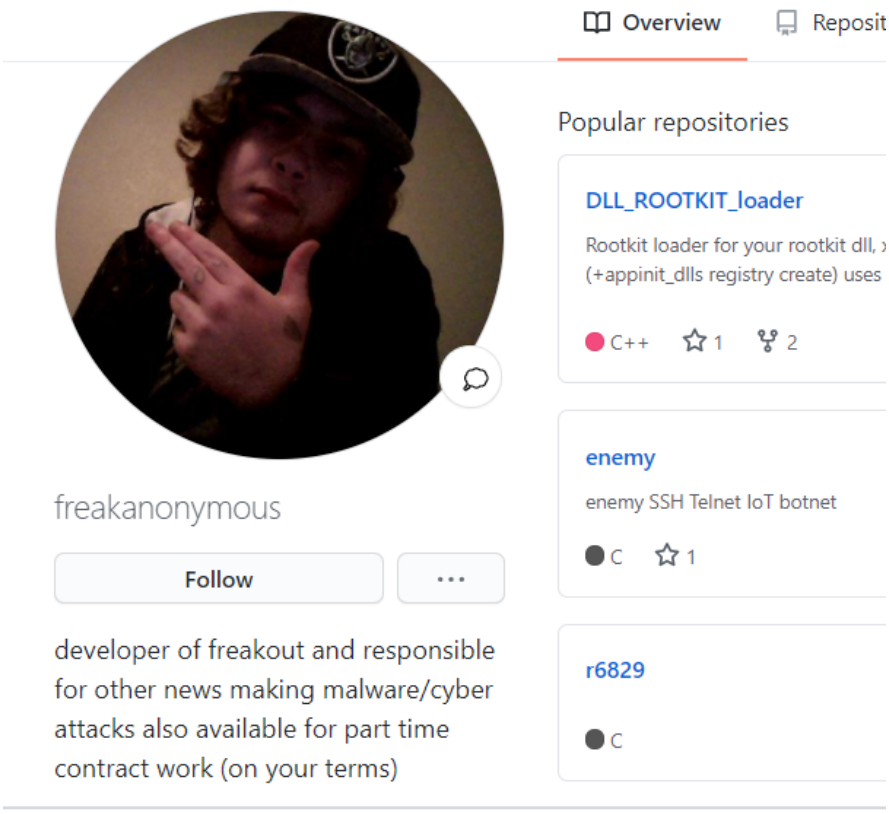


Figure 3 -@freakanonymous

While Keksec's github was recently created in January 2021, they also administer a pastebin with pastes starting in 2014. The Keksec pastebin has various custom tools including scanners, exploits and crypters. One of the pastes is the python code used for obfuscating earlier versions of Necro using zlib compression without the additional XOR encoding.

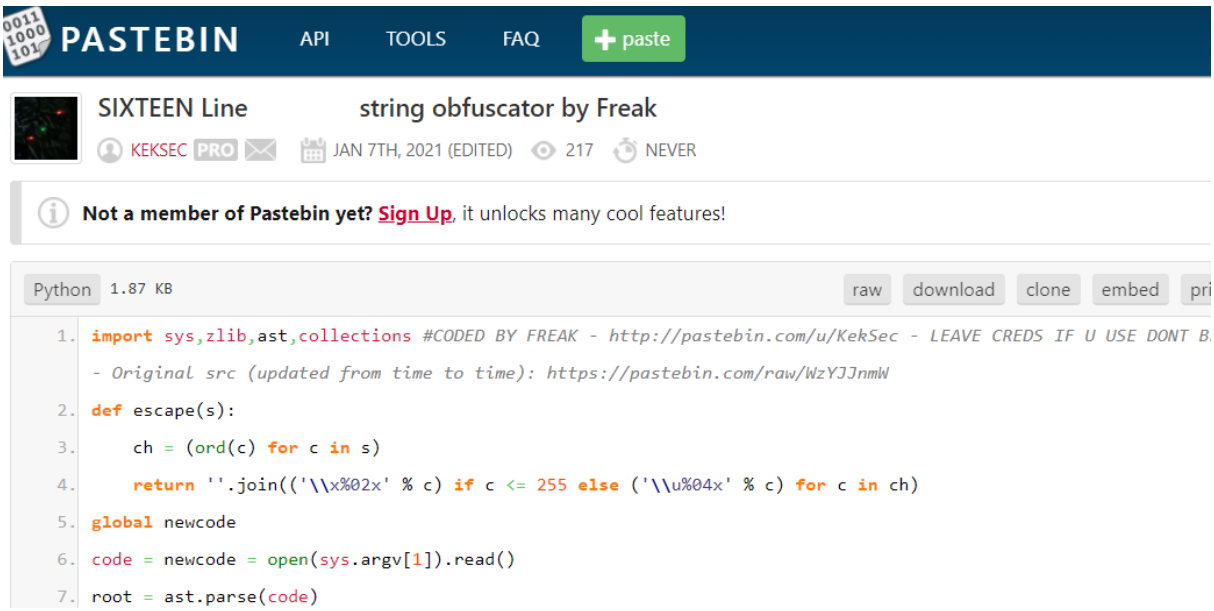


Figure 4 -Necro Obfuscator

Lacework recently blogged about another group named TeamTNT who shares multiple similarities with regards to general tactics. For example, both are self-promoters with a social media presence. Also, both leverage the Kaiten source and self-attribute their code and infrastructure. There is also evidence that the two groups are acquainted at some level as the TeamTNT associated GitHub account, **@jwne** is only one of 5 followers of @freakanonymous. For more details on TeamTNT, refer to our **blog**.

Indicators and Tools for this activity are available on our GitHub. If you found this blog useful then please share and follow us on **Twitter**!