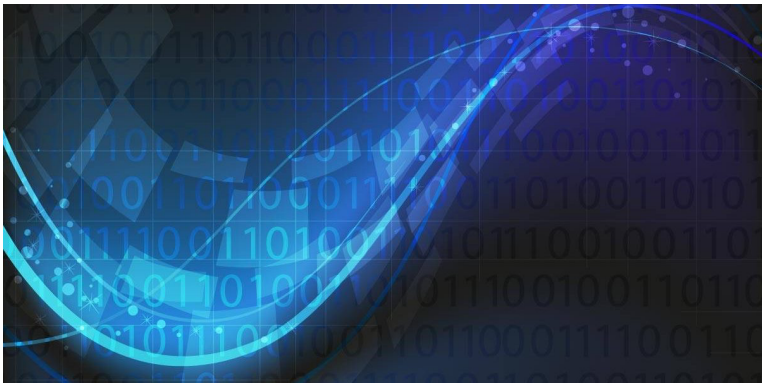
The CozyDuke APT

APT REPORTS 21 APR 2015 8 minute read



// AUTHORS

 KURT BAUMGARTNER  COSTIN RAIU

CozyDuke (aka CozyBear, CozyCar or "Office Monkeys") is a precise attacker. Kaspersky Lab has observed signs of attacks against government organizations and commercial entities in the US, Germany, South Korea and Uzbekistan. In 2014, targets included the White House and the US Department of State, as believed.

The operation presents several interesting aspects

- extremely sensitive high profile victims and targets
- evolving crypto and anti-detection capabilities
- strong malware functional and structural similarities mating this toolset to early MiniDuke second stage components, along with more recent CosmicDuke and OnionDuke components

The actor often spearphishes targets with e-mails containing a link to a hacked website. Sometimes it is a high profile, legitimate site such as "diplomacy.pl", hosting a ZIP archive. The ZIP archive contains a RAR SFX which installs the malware and shows an empty PDF decoy.

In other highly successful runs, this actor sends out phony flash videos directly as email attachments. A clever example is “Office Monkeys LOL Video.zip”. The executable within not only plays a flash video, but drops and runs another CozyDuke executable. These videos are quickly passed around offices with delight while systems are infected in the background silently. Many of this APT’s components are signed with phony Intel and AMD digital certificates.

Recent CozyDuke APT activity attracted significant attention in the news:

Sources: State Dept. hack the ‘worst ever’, CNN News, March 2015
White House computer network ‘hacked’, BBC News, October 2014
Three Months Later, State Department Hasn’t Rooted Out Hackers, Wall Street Journal, February 2015
State Department shuts down its e-mail system amid concerns about hacking, Washington Post, November 2014

Let's examine a smattering of representative CozyDuke files and data. There is much to their toolset.

Office Monkeys dropper analysis

CozyDuke droppers and spyware components often maintain fairly common characteristics, but these files' functionality are modified in slight ways depending on the team's needs. This rapid development and deployment is interesting.

68271df868f462c06e24a896a9494225,**Office Monkeys LOL Video.zip**

Believe it or not, recipients in bulk run the file within:

95b3ec0a4e539efaa1faa3d4e25d51de,**Office Monkeys (Short Flash Movie).exe**

This file in turn drops two executables to %temp%:

2aabd78ef11926d7b562fd0d91e68ad3, Monkeys.exe

3d3363598f87c78826c859077606e514, player.exe

It first launches Monkeys.exe, playing a self-contained, very funny video of white-collar tie wearing chimpanzees working in a high rise office with a human colleague. It then launches player.exe, a CozyDuke dropper maintaining anti-detection techniques:

3d3363598f87c78826c859077606e514,player.exe,338kb,Trojan.Win32.CozyBear.v,CompiledOn:2014.07.02 21:13:33

Anti-detection and trojan functionality

The file collects system information, and then invokes a WMI instance in the rootsecuritycenter namespace to identify security products installed on the system, meaning that this code was built for x86 systems, wql here:

SELECT * FROM AntiVirusProduct
SELECT * FROM FireWallProduct

The code hunts for several security products to evade:

- CRYSTAL
- KASPERSKY
- SOPHOS
- DrWeb
- AVIRA
- COMODO Dragon

In addition to the WMI/wql use, it also hunts through the "SOFTWAREMicrosoftWindowsCurrentVersionUninstall" registry key looking for security products to avoid. Following these checks, it drops several more malware files signed with the pasted AMD digital signature to a directory it creates. These files are stored within an 217kb encrypted cab file in the dropper's resources under the name "A". The cab file was encrypted and decrypted using a simple xor cipher with a rotating 16 byte key: x36x11xddx08xacx4bx72xf8x51x04x68x2ex3ex38x64x32.

The cab file is decompressed and its contents are created on disk. These dropped files bundle functionality for both 64bit and 32bit Windows systems and are all located within one directory:
C:Documents and SettingsuserApplication DataATI_Subsystem

6761106f816313394a653db5172dc487,amdhcp32.dll,54kb ← 32bit dll,CompiledOn:2014.07.02 21:13:24
d596827d48a3ff836545b3a999f2c3e3,aticaldd.dll,60kb ← 64bit dll,CompiledOn:2014.07.02 21:13:26
bc626c8f11ed753f33ad1c0fe848d898,atiumdag.dll,285kb ← 32bit dll, Trojan.Win32.CozyDuke.a, CompiledOn:2014.07.02 21:13:26
4152e79e3dbde55dcf3fc2014700a022,6kb,racss.dat

The code copies rundll32.exe from windowssystem32 to its newly created %appdata%ATI_Subsystem subdirectory as "amdocl_as32.exe" alongside the three dll's listed above. It runs atiumdag.dll with two parameter values, it's only export and an arbitrary pid, i.e.:
"C:Documents and SettingsuserApplication DataATI_Subsystemamdocl_as32.exe" "C:Documents and SettingsuserApplication DataATI_Subsystematiumdag.dll"", ADL2_ApplicationProfiles_System_Reload 1684"

This dll is built with anti-AV protections as well. However, it looks for a different but overlapping set, and the random duplication suggests that this component was cobbled together with its dropper, partly regionally based on target selection.

K7

KASPERSKY

AVG

The code collects information about the system and xml formats this data prior to encryption for proper parsing:

Finally, this process beacons to www.sanjosemaristas.com, which appears to be a site that has been compromised and misused multiple times in the past couple of years. hxxp://www.sanjosemaristas[.]com/app/index.php?{A01BA0AD-9BB3-4F38-B76B-A00AD11CBAAA}, providing the current network adapter's service name GUID. It uses standard Win32 base cryptography functions to generate a CALG_RC4 session key to encrypt the collected data communications and POSTs it to the server.

Executable-Signing Certificates

Samples are usually signed with a fake certificate – we've seen two instances, one AMD and one Intel:

FROM THE SAME AUTHORS

Focus on DroxiDat/SystemBC

DiceyF deploys GamePlayerFramework in online casino development studio

TOP 10 unattributed APT mysteries

Black Hat USA 2022 and DEF CON 30

Andariel deploys DTrack and Maui ransomware

Configuration files:

Some of the malware uses an encrypted configuration file which is stored on disk as “racss.dat”. This is encrypted by RC4, using the key {0xb5, 0x78, 0x62, 0x52, 0x98, 0x3e, 0x24, 0xd7, 0x3b, 0xc6, 0xee, 0x7c, 0xb9, 0xed, 0x91, 0x62}. Here’s how it looks decrypted:

Second stage malware and communications:

The attackers send commands and new modules to be executed to the victims through the C&Cs. The C&C scripts store these temporarily until the victim next connects to retrieve local files. We’ve identified two such files:

- settings.db
- sdfg3d.db

Here's how such a database file appears:

These are BASE64 encoded and use the same RC4 encryption key as the malware configuration.

Decoding them resulted in the following payloads:

59704bc8bedef32709ab1128734aa846, ChromeUpdate.ex_
5d8835982d8bfc8b047eb47322436c8a, cmd_task.dll
e0b6f0d368c81a0fb197774d0072f759, screenshot_task.dll

Decoding them also resulted in a set of tasking files maintaining agent commands and parameter values:

conf.xml

And a set of "reporting" files, maintaining stolen system "info", error output, and "AgentInfo" output, from victim systems:

DCOM_amdocl_ld_API_.raw
Util_amdave_System_.vol
Last_amdpcom_Subsystem_.max
Data_amdmiracast_API_.aaf
7.txt

screenshot_task.dll is a 32-bit dll used to take a screenshot of the full desktop window and save it as a bitmap in %temp%. The number of times the screenshot is repeated is configurable within the xml task file.

cmd_task.dll is a 32-bit dll that maintains several primitives. It is used to create new processes, perform as a command line shell, and several other tasks.

Each of these payloads is delivered together with a configuration file that explains how to run it, for instance:

In another tasking, we notice a tracked victim:

Attackers map a network drive use Microsoft OneDrive to run further tools:

They copy down a base64 encoded document from Microsoft OneDrive to the victim system and decode it there:

Not everything works as planned, so they maintain error reporting facility for the c2 communications:

Furthermore, ChromeUpdate is a 64-bit executable (which appears to be a WEXTRACT package) that oddly drops a 32-bit Dll. Cache.dll is simply stored as a cabinet file in the ChromeUpdate's resource section.

ChromeUpdate.exe starts the file with "rundll32 cache.dll,ADB_Setup"

Cache.dll analysis

Cache.dll was written in C/C++ and built with a Microsoft compiler.

Cache.dll code flow overview

- RC4 decrypt hardcoded c2 and urls
- resolve hidden function calls
- collect identifying victim system data
- encrypt collected data
- send stolen data to c2 and retrieve commands

Cache.dll code details

Structurally, "Cache.dll" is a fairly large backdoor at 425KB. It maintains both code and data in the raw, encrypted blobs of data to be decrypted and used at runtime, and

hidden functionality that isn't exposed until runtime. No pdb/debug strings are present in the code.

It maintains eight exports, including DllMain:

- ADB_Add
- ADB_Cleanup
- ADB_Initnj
- ADB_Load
- ADB_Release
- ADB_Remove
- ADB_Setup

ADB_Setup is a entry point that simply spawns another thread and waits for completion.

Above, we see a new thread created with the start address of Cache.dll export "ADB_Load" by the initial thread.

This exported function is passed control while the initial thread runs a Windows message loop. It first grabs an encrypted blob stored away in a global variable and pulls out 381 bytes of this encrypted data:

The standard win32 api CryptDecrypt uses rc4 to decrypt this blob into a hardcoded c2, url path, and url parameters listed below with a simple 140-bit key “x8BxFFx55x8BxECx83xECx50xA1x84x18x03x68x33xC9x66xF7x45x10xE8x1Fx89x45xFCx8Bx45x14x56”.

The code then decodes this set of import symbols and resolves addresses for its networking and data stealing functionality:

InternetCloseHandle
InternetReadFile
HttpSendRequestA
HttpOpenRequestA
HttpQueryInfoA
InternetConnectA
InternetCrackUrlA
InternetOpenA
InternetSetOptionW
GetAdaptersInfo

Much like the prior office monkey "atiumdag.dll" component, this code collects identifying system information using standard win32 API calls:

- Computer name – GetComputerNameW
- User name – GetUserNameW
- Adapter GUID, ip address, mac address – GetAdaptersInfo
- Windows version – GetVersionExW

It then uses the runtime resolved networking API calls to send the collected data back to a hardcoded c2 and set of urls.

Cache.dll connectback urls:

- 209.200.83.43/ajax/links.php
- 209.200.83.43/ajax/api.php
- 209.200.83.43/ajax/index.php
- 209.200.83.43/ajax/error.php
- 209.200.83.43/ajax/profile.php
- 209.200.83.43/ajax/online.php
- 209.200.83.43/ajax/loader.php
- 209.200.83.43/ajax/search.php

Observed user-agent string on the wire, but it's dynamically generated based on the Windows system settings (retrieved using standard win32 api

"ObtainUserAgentString"):
"User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727; .NET CLR 3.0.04506.648; .NET CLR 3.5.21022)"

Communications with the CozyDuke C2 include key/value pairs passed as URL parameters. Observed keys that remind us of the Cosmicduke communications include:

status=

k=

mode=

ajax=

name=

subNodeId=

nodeId=

r=

t=

id=

item=

item_id=

js=

j=

v=

json=

i=

c=

x=

a=

Connections with MiniDuke/CosmicDuke/OnionDuke:

One of the second stage modules of CozyDuke/Cozy Bear, Show.dll, is particularly interesting because it appears to have been built onto the same platform as OnionDuke. Below we compare Show.dll with the OnionDuke sample MD5: c8eb6040fd02d77660d19057a38ff769. Both have

IN THE SAME CATEGORY

BellaCPP: Discovering a new BellaCiao variant written in C++

Lazarus group evolves its infection chain with old and new malware

Careto is back: what's new after 10 years of silence?

APT trends report Q3 2024

Beyond the Surface: the evolution and expansion of the SideWinder APT group

exactly the same export tables and appear to be called internally “UserCache.dll”:

This seems to indicate the authors of OnionDuke and CozyDuke / Cozy Bear are the same, or working together.

Another interesting comparison of two other files matches a recent second stage tool from the CozyDuke attacks with a second stage component from other Miniduke/Onionduke attacks.

2e0361fd73f60c76c69806205307ccac, update.dll (MiniDuke), 425kb (internal name = “**UserCache.dll**“)
9e3f3b5e9ece79102d257e8cf982e09e, cache.dll (CozyDuke), 425kb (internal name = “**UserCache.dll**“)

The two share identical export function names in their export directories, and the naming appears to be randomly assigned at compile time. The table below presents the function matches based on size data, but the calls, jmps and code all match as well. The contents of only one of these exports in update.dll has no match whatsoever in cache.dll.

Unlike the atiumdag.dll file above, however, cache.dll and update.dll do not maintain anti-AV and anti-analysis functionality sets. Perhaps they plan to pair this stealer with another dropper that maintains the WMI anti-AV functionality. This rotating functionality seems representational for the set, along with other characteristics. Their custom backdoor components appear to slightly evolve over time, with modifications to anti-detection, cryptography, and trojan functionality changing per operation. This rapid development and deployment reminds us of the APT28/Sofacy toolset, especially the coreshell and chopstick components.

We expect ongoing and further activity from this group in the near future and variations on the malware used in previous duke-ish incidents.

For more information about MiniDuke, CosmicDuke and OnionDuke, please see References.

Related MD5s

62c4ce93050e48d623569c7dcc4d0278, 2537.ex_
a5d6ad8ad82c266fda96e076335a5080, drop1.ex_
93176df76e351b3ea829e0e6c6832bdf, drop1.pd_
7688be226b946e231e0cd36e6b708d20, 8.zip
fd8e27f820bdbdf6cb80a46c67fd978a, doc853.ex_
93176df76e351b3ea829e0e6c6832bdf, doc853.pdf
9ad55b83f2eec0c19873a770b0c86a2f, reader_sl.ex_
f16dff8ec8702518471f637eb5313ab2 1.ex_
8670710bc9477431a01a576b6b5c1b2a
93176df76e351b3ea829e0e6c6832bdf,
droppedhppscan854.pdf
f58a4369b8176edbde4396dc977c9008,
droppedreader_sl.ex_
83f57f0116a3b3d69ef7b1dbe9943801
b5553645fe819a93aafe2894da13dae7
acffb2823fc655637657dcbd25f35af8
1a42acbdb285a7fba17f95068822ea4e
d543904651b180fd5e4dc1584e639b5e
d7af9a4010c75af6756a603fd6aef5a4
93176df76e351b3ea829e0e6c6832bdf, 3852.pdf
f2b05e6b01be3b6cb14e9068e7a66fc1,
droppedreader_sl.ex_
57a1f0658712ee7b3a724b6d07e97259, dropped3852.ex_
93176df76e351b3ea829e0e6c6832bdf, 5463.pdf

eb22b99d44223866e24872d80a4ddefd,
dropped5463reader_sl.ex_
90bd910ee161b71c7a37ac642f910059, dropped5463.ex_
1a262a7bfecd981d7874633f41ea5de8
98a6484533fa12a9ba6b1bd9df1899dc
7f6bca4f08c63e597bed969f5b729c56
08709ef0e3d467ce843af4deb77d74d5

Related CozyDuke C&Cs:

```
121.193.130.170:443/wp-ajax.php
183.78.169.5:443/search.php
200.119.128.45:443/mobile.php
200.125.133.28:443/search.php
200.125.142.11:443/news.php
201.76.51.10:443/plugins/json.php
202.206.232.20:443/rss.php
202.76.237.216:443/search.php
203.156.161.49:443/plugins/twitter.php
208.75.241.246:443/msearch.php
209.40.72.2:443/plugins/fsearch.php
210.59.2.20:443/search.php
208.77.177.24:443/fsearch.php
www.getiton.hants.org.uk:80/themes/front/img/ajax.ph
www.seccionpolitica.com.ar:80/galeria/index.php
209.200.83.43/ajax/links.php
209.200.83.43/ajax/api.php
209.200.83.43/ajax/index.php
209.200.83.43/ajax/error.php
209.200.83.43/ajax/profile.php
209.200.83.43/ajax/online.php
209.200.83.43/ajax/loader.php
209.200.83.43/ajax/search.php
```

Appendix: Parallel and Previous Research

The MiniDuke Mystery: PDF 0-day Government Spy Assembler 0x29A Micro Backdoor, Securelist, Feb 2013
Miniduke is back: Nemesis Gemina and the Botgen Studio, Securelist, July 2014
MiniDuke 2 (CosmicDuke), CrySyS, July 2014
COSMICDUKE Cosmu with a twist of MiniDuke [pdf], F-Secure, September 2014
THE CASE OF THE MODIFIED BINARIES, Leviathan Security, October 2014
A word on CosmicDuke, Blaze's Security Blog, September 2014
OnionDuke: APT Attacks Via the Tor Network, F-Secure, November 2014
The Connections Between MiniDuke, CosmicDuke and OnionDuke, F-Secure, January 2015

Kaspersky Lab products detect the malware used by the CozyDuke threat actor as:

HEUR:Trojan.Win32.CozyDuke.gen
Trojan.Win32.CozyBear.*

APT COZYDUKE SPYWARE

The CozyDuke APT

Your email address will not be published. Required fields are marked *

Type your comment here

Name *

Email *

Comment

JIM BISHOP
Posted on April 27, 2015. 12:24 pm

Great post! Very detailed and informative.

Reply

// LATEST POSTS

SOC, TI AND IR POSTS

Attackers exploiting a patched FortiClient EMS vulnerability in the wild

ASHLEY MUÑOZ,
FRANCESCO FIGURELLI,
CRISTIAN SOUZA,
EDUARDO OVALLE,
AREG BAGHINYAN

APT REPORTS

Lazarus group evolves its infection chain with old and new malware

VASILY BERDNIKOV, SOJUN RYU

CRIMEWARE REPORTS

Analysis of Cyber Anarchy Squad attacks targeting Russian and Belarusian organizations

KASPERSKY

MALWARE DESCRIPTIONS

Download a banker to track your parcel

DMITRY KALININ

// LATEST WEBINARS

THREAT INTELLIGENCE AND IR

04 SEP 2024, 5:00PM 60 MIN

Inside the Dark Web: exploring the human side of cybercriminals

ANNA PAVLOVSKAYA

TECHNOLOGIES AND SERVICES

13 AUG 2024, 5:00PM 60 MIN

The Cybersecurity Buyer's Dilemma: Hype vs (True) Expertise

OLEG GOROBETS,
ALEXANDER LISKIN

CYBERTHREAT TALKS

16 JUL 2024, 5:00PM 60 MIN

Cybersecurity's human factor – more than an unpatched vulnerability

OLEG GOROBETS

TRAININGS AND WORKSHOPS

09 JUL 2024, 4:00PM 60 MIN

Building and prioritizing detection engineering backlogs with MITRE ATT&CK

ANDREY TAMOYKIN

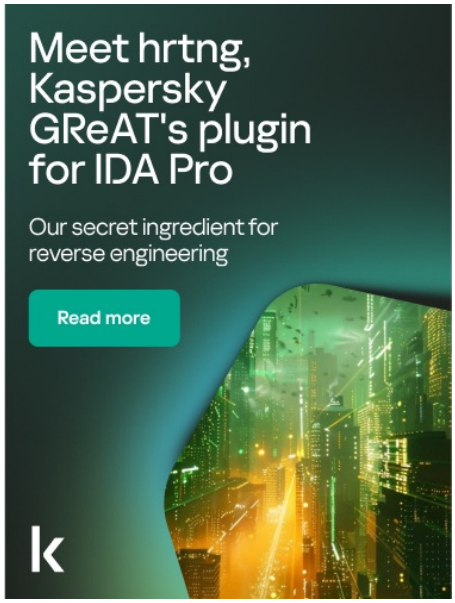
// REPORTS

BellaCPP: Discovering a new BellaCiao variant written in C++

While investigating an incident involving the BellaCiao .NET malware, Kaspersky researchers discovered a C++ version they dubbed “BellaCPP”.

Lazarus group evolves its infection chain with old and new malware

Lazarus targets employees of a nuclear-related organization with a bunch of malware, such as MISTPEN, LPEClient, RollMid, CookieTime and a new modular backdoor CookiePlus.



Careto is back: what’s new after 10 years of silence?

Kaspersky researchers analyze 2019, 2022 and 2024 attacks attributed to Careto APT with medium to high confidence.

APT trends report Q3 2024

The report features the most significant developments relating to APT groups in Q3 2024, including hacktivist activity, new APT tools and campaigns.

// SUBSCRIBE TO OUR WEEKLY E-MAILS

The hottest research right in your inbox

Email

Subscribe

I agree to provide my email address to “AO Kaspersky Lab” to receive information about new posts on the site. I understand that I can withdraw this consent at any time via e-mail by clicking the “unsubscribe” link that I find at the bottom of any e-mail sent to me for the purposes mentioned above.

THREATS

APT (Targeted attacks)
Secure environment (IoT)
Mobile threats
Financial threats
Spam and phishing
Industrial threats
Web threats
Vulnerabilities and exploits
All threats

CATEGORIES

APT reports
Malware descriptions
Security Bulletin
Malware reports
Spam and phishing reports
Security technologies
Research
Publications
All categories

OTHER SECTIONS

Archive
All tags
Webinars
APT Logbook
Statistics
Encyclopedia
Threats descriptions
KSB 2024

Privacy Policy
License Agreement
Cookies