深度剖析針對臺灣金融業的 Operation Cache Panda 組織型供應鏈攻擊

CYCRAFT

Everything Starts From CyCraft

奧義智慧 (CyCraft) 是世界領先的臺灣 AI 資安科技公司，於日本、新加坡、美國設有子公司，以創新 AI 技術自動化資安防護，整合 EDR、CTI、TIG 並建構新一代 AI 資安戰情中心，獲關鍵領域企業採用，市佔率國內第一。

2021 年入選 Gartner《大中華區 AI 新創公司指南》與 IDC《Intelligence-led Cybersecurity》等報告代表性案例。從端點到網路、從調查到阻擋、從自建到託管，CyCraft AIR 提供客戶主動式防禦，達到「威脅，視可而止」。

前言

隨著金融科技的技術持續發展，金融產業使用了更多的資訊系統，便也代表著比起過去任何時候，潛藏了更多未知的資安威脅，而駭客入侵所造成的影響，往往也牽一髮而動全身，有著絕不可小覷的風險。

2021 年底一連串我國證券商與期貨商遭受駭客撞庫攻擊、導致下單系統異常的新聞，在當時引發了社會上一片軒然大波。奧義智慧研究團隊在參與事件調查 (Incident Response, IR) 時，成功挖掘出關於金融攻擊事件的更多內幕，本篇文章將帶您深入瀏覽與探討，來自中國國家級駭客的金融產業供應鏈攻擊手法剖析、惡意程式技術，與對應的緩解措施等。

事件緣起

去年臺灣發生多起證券、期貨商遭到撞庫攻擊，甚至出現下單異常案件的情況，研判應為系統性問題而非單一個案，並且對於交易秩序的影響相當嚴重。該攻擊事件疑似為特定組織型駭客所發起，長期且有目的性的滲透行動，從攻擊手法中可以觀察到，駭客具有針對不同目標環境開發對應後門、躲避安全軟體偵測的能力，並十分擅長於企業內網攻擊，操作手法亦相當熟稔。

奧義智慧科技 (CyCraft) 於 2021 年 11 月底到 2022 年 2 月初，監控到一系列大範圍且專門針對臺灣金融單位軟體系統的供應鏈攻擊事件，遂而開始展開進一步詳細的調查。初步發現，攻擊者準確利用了我國金融單位常用的軟體系統之漏洞，第一波攻擊於 2021 年 11 月底出現受駭案例，第二波活動的高峰期則在 2022 年 2 月 10 至 13 號之間，攻擊者來源 IP 位於香港。

經調查，本次攻擊事件所使用之後門程式為 QuasarRAT，經過分析啟動方式、保護機制與使用之 C2 中繼站等情資後，研判應為中國國家級駭客 APT10 所發起的新活動，主要針對國內金融業發動攻擊。

由於在過去的資安研究之中，源於中國的 APT 組織一般較少以經濟獲益為目標，然而，本起行動中則明確有著盜竊金融資料的行為，因此奧義研究團隊以「咬錢熊貓」(Operation Cache Panda) 這項代稱來命名此行動。

攻擊手法剖析

Operation Cache Panda 行動中，利用到了一項證券軟體系統管理介面的網站服務漏洞。首先，攻擊者上傳了中國駭客常用之 ASPXCSharp WebShell 進行網站主機控制，之後便開始利用知名內網滲透工具 Impacket 掃描內網電腦，試圖大範圍植入 DotNet 後門程式，並意圖竊取受駭單位資料。

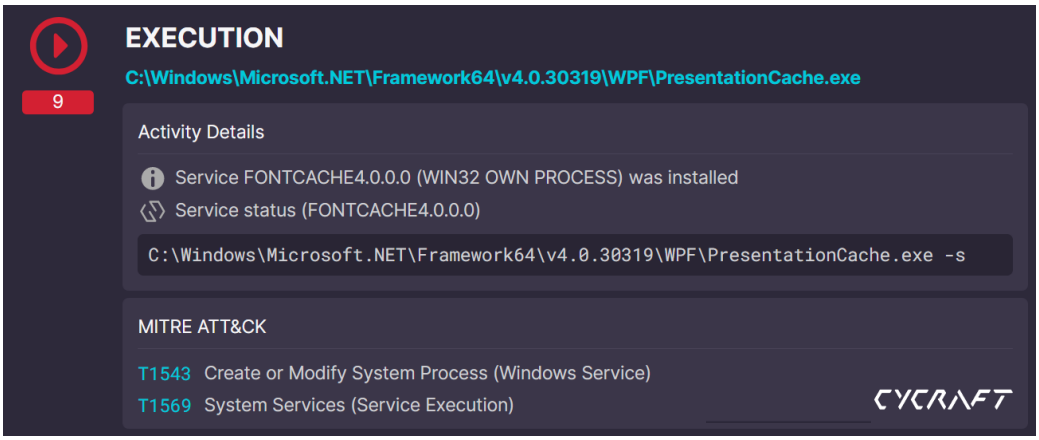
攻擊者大量使用了動態載入 DotNet 組件檔案 (DotNet Assembly) 的技術，透過攻擊手法 Reflective Code Loading（MITRE ATT&CK 編號 T1620），動態注射惡意 DotNet Assembly 程式碼到系統以合法執行程序。

此次事件除了使用到可編譯不同平台 Shellcode、透過 In-Memory 的方式執行 DotNet Assembly 的開源專案 Donut 外，亦發現使用部分 SharpSploit 程式碼注入 DotNet 惡意程式，可以達到無惡意模組落地的隱匿效果，藉以降低被防毒軟體偵測機率。

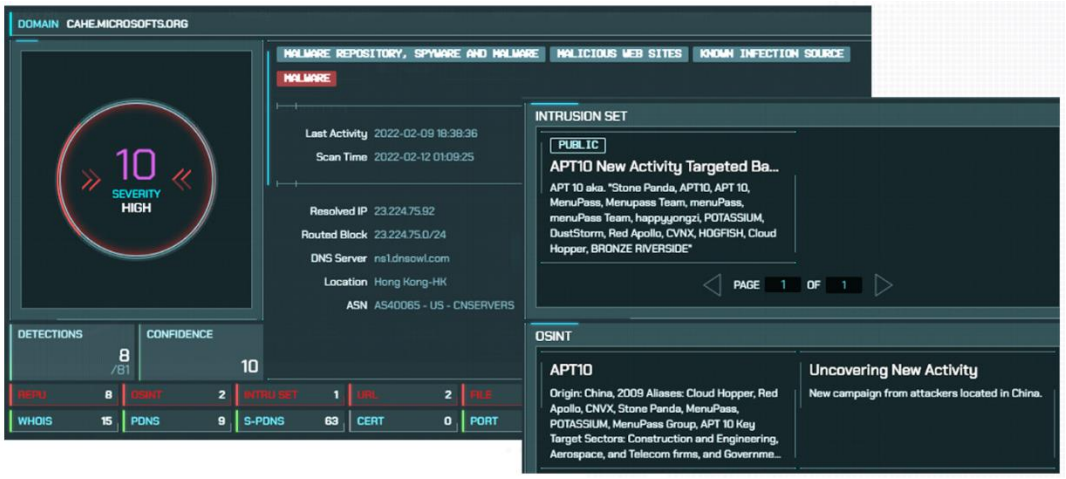
其後攻擊者將會搭配 Impacket，透過 Remote Service/WMI 方式橫向擴散到內部主機。一旦成功取得內部主機的控制權，攻擊者便會建立 Reverse Tunnel RDP，使其更容易地透過遠端桌面操作受駭主機。

在本次調查當中，我們發現駭客使用了名為文叔叔的中國雲端檔案分享服務來下載相關工具，藉以達到一定程度的方便性以及匿名性；不過，也正因如此，駭客在透過 RDP 登入受駭主機時，反而容易留下更多追查線索。

本次遭駭的軟體系統在臺據稱有八成以上的市佔率，屬於金融機構的供應鏈攻擊。據悉已有多家企業遭受 Operation Cache Panda 行動不同程度的影響，建議金融單位立即修補軟體系統漏洞，限制 Web 管理介面的存取範圍，並盤點本文文末所提供的入侵指標 (Indicator of Compromise, IoC)，包含網路 IP、檔案雜湊 (hash) 與惡意程式特徵等，另外也建議安裝奧義智慧的 Xensor EDR，開啟惡意程式保護模組 (Malware Protection Module) 以監控與阻擋相關的惡意活動。



圖一、奧義智慧第一時間監控，並告警駭客內網滲透活動



圖二、奧義智慧全球情資平台 CyberTotal 歸因出攻擊者疑為 APT10

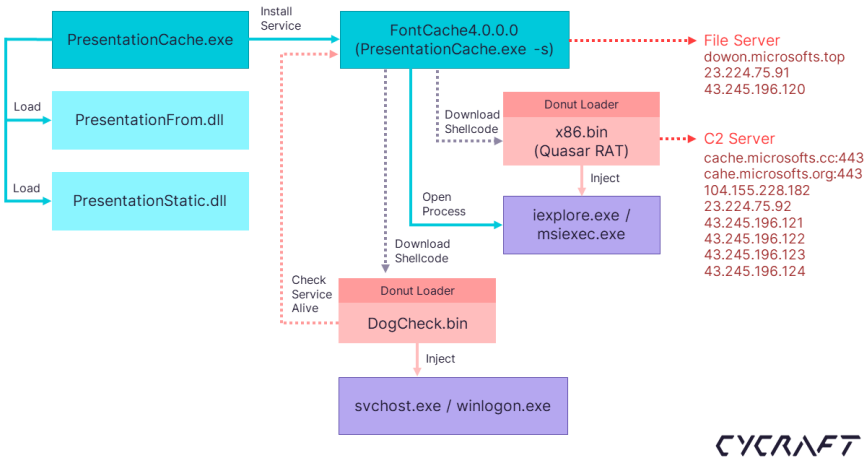
攻擊技術分析

第一階段：突破與建立進入點

本次攻擊所使用的 WebShell 取用於開源專案，此 Webshell 改良了中國駭客常用的蟻劍 WebShell 框架 (As-Exploits)，並加強其動態加載與執行 DotNet Assembly 的能力，透過 GetType[0] 取得和建構出 Payload 的 Run 類型，以確保能做到無惡意檔案落地與不會留下 Web 存取紀錄之效果。

第二階段：移動與潛伏

Operation Cache Panda 事件的攻擊者使用到六隻惡意程式，其中，只有三個檔案會落地，其餘皆在動態下載後載入。這六隻惡意程式各自負責了不同的功能，並串連成了本次的攻擊，整體流程請參照下方圖片。



圖三、惡意程式架構與活動分析

PresentationCache.exe 為後門程式 Quasar 的載入器，首先會將自身註冊為服務，使能夠常駐於系統，且載入 PresentationFrom.dll 及 PresentationStatic.dll 兩個 DLL 檔案。當 PresentationCache.exe 被執行後，則會向外部檔案下載伺服器、抓取 x86.bin 及 DogCheck.bin 檔案，並將這兩個 Shellcode 檔案注入其他的 Process。

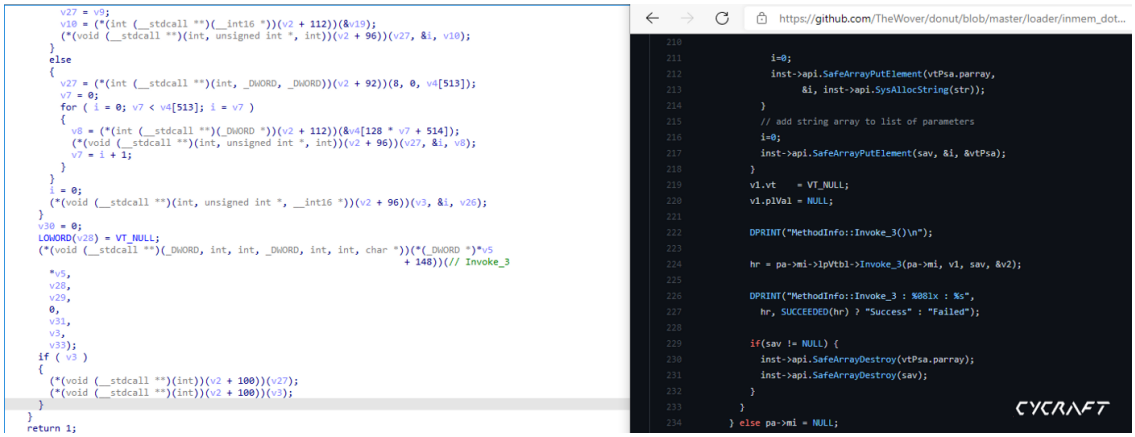
這兩個 Shellcode 會動態加載 DotNET 執行環境，並載入攻擊者的 DotNet Assembly 來負責進行後續行為。在這之中，x86.bin 為後門主體，改自惡名昭彰的 DotNet 後門 Quasar RAT，而 DogCheck.bin 則為守門員，負責檢查後門連線狀況，若是端點沒有連到 C2 Server 時，便會認定程式執行失敗，並且再次將 PresentationCache.exe 重啟。

透過這樣的模式，駭客達到可確保 x86.bin 僅以 In-Memory 的形式執行，使主體惡意程式不落地。而 DogCheck.bin 則用以維持後門運作狀況，加強受駭電腦整體控制度。

PresentationCache 惡意程式技術分析

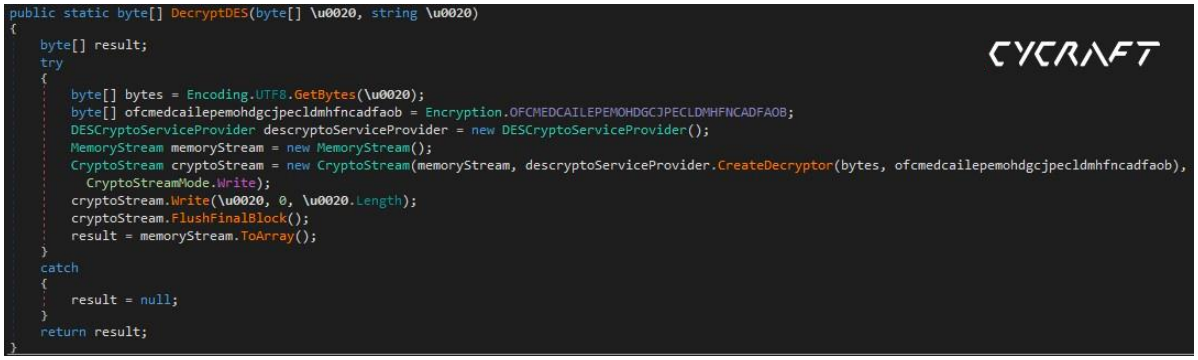
採用 DotNET 相關攻擊技術

本次的攻擊行動，使用到 DotNet Assembly Loader 以及 DotNet Obfuscator，以增加分析調查的難度。而利用了開源專案 Donut 的惡意程式，能夠編譯不同平台的 Shellcode，並負責動態載入 DotNet Assembly。另外，透過 In-Memory 方式所執行的 DotNet Assembly，可以達到 Fileless 的攻擊效果，極力降低留下檔案的機會。因此在調查過程中，時常會因記憶體內的資料已經消失，而無法找到惡意程式主體。



圖四、惡意程式與 Donut 原始碼的比對圖

我們也發現駭客在攻擊過程中，透過商用 DotNET 混淆工具 DotNet Reactor 來增加逆向的難度。 DotNet Reactor 會對程式控制流程進行修改、混淆，也會動態產生 DotNET IL，動態時期才將程式解出並執行。為避免靜態分析，攻擊者使用許多混淆機制，如透過 DES CBC 加密部分字串來避免偵測等。



```
public static byte[] DecryptDES(byte[] \u0020, string \u0020)
{
    byte[] result;
    try
    {
        byte[] bytes = Encoding.UTF8.GetBytes(\u0020);
        byte[] ofcmedcailepemohdgcjpecldmhfncadfaob = Encryption.OFCMEDCAILEPEMOHDGCJPECLDMHFNCADFAOB;
        DESCryptoServiceProvider descryptoServiceProvider = new DESCryptoServiceProvider();
        MemoryStream memoryStream = new MemoryStream();
        CryptoStream cryptoStream = new CryptoStream(memoryStream, descryptoServiceProvider.CreateDecryptor(bytes, ofcmedcailepemohdgcjpecldmhfncadfaob),
          CryptoStreamMode.Write);
        cryptoStream.Write(\u0020, 0, \u0020.Length);
        cryptoStream.FlushFinalBlock();
        result = memoryStream.ToArray();
    }
    catch
    {
        result = null;
    }
    return result;
}
```

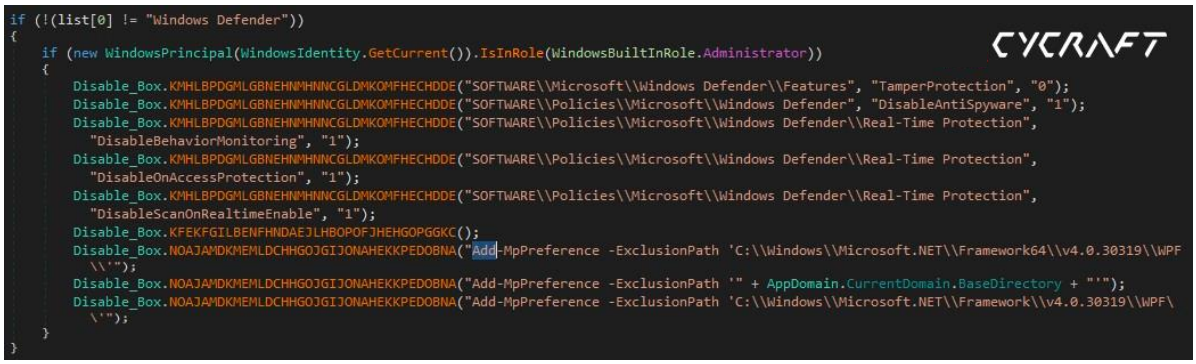
圖五、DES CBC 加密部分字串

值得一提的是，這次攻擊中駭客大量使用開源或商業軟體，並減少運用駭客自行開發的惡意程式，目的為降低被關聯之風險，例如駭客所運用的核心後門程式，便是以 C# 實作與開源的 Quasar RAT。

為了避免被偵測到惡意行動，攻擊者利用了許多 Defense evasion 的技術來迴避，例如把惡意程式加到 Defender 的白名單、檢查 Sandboxie 等；惡意程式會檢查是否有 SbieDLL.dll 的存在，以確認自己是否在 Sandboxie 的沙盒環境中，如果是的話便會立即停止執行，藉以規避掉沙盒分析。

```
public static bool Main()
{
    bool result = false;
    if (Debugger.IsAttached || BoxCheck.KOKAHGFFJAGGHBDKIHLAOMAEPLLKPCOPKAHH() || BoxCheck.GetModuleHandle("SbieDll.dll").ToInt32() != 0 ||
      Process.GetProcesses().Length <= 40)
    {
        result = true;
    }
    return result;
}
```

圖六、檢查 SbieDLL.dll



```
if (!(list[0] != "Windows Defender"))
{
    if (new WindowsPrincipal(WindowsIdentity.GetCurrent()).IsInRole(WindowsBuiltInRole.Administrator))
    {
        Disable_Box.KMHLBPDGMLGBNEHNMHNNCGLDMKOMFHECHDDE("SOFTWARE\\Microsoft\\Windows Defender\\Features", "TamperProtection", "0");
        Disable_Box.KMHLBPDGMLGBNEHNMHNNCGLDMKOMFHECHDDE("SOFTWARE\\Policies\\Microsoft\\Windows Defender", "DisableAntiSpyware", "1");
        Disable_Box.KMHLBPDGMLGBNEHNMHNNCGLDMKOMFHECHDDE("SOFTWARE\\Policies\\Microsoft\\Windows Defender\\Real-Time Protection",
          "DisableBehaviorMonitoring", "1");
        Disable_Box.KMHLBPDGMLGBNEHNMHNNCGLDMKOMFHECHDDE("SOFTWARE\\Policies\\Microsoft\\Windows Defender\\Real-Time Protection",
          "DisableOnAccessProtection", "1");
        Disable_Box.KMHLBPDGMLGBNEHNMHNNCGLDMKOMFHECHDDE("SOFTWARE\\Policies\\Microsoft\\Windows Defender\\Real-Time Protection",
          "DisableScanOnRealtimeEnable", "1");
        Disable_Box.KFEKFGILBENFHNDAEJLHBOPOFJHEHGOPGGKC();
        Disable_Box.NOAJAMDKMEMLDCHHGOJGIJONAHEKKPEDOBNA("Add-MpPreference -ExclusionPath 'C:\\Windows\\Microsoft.NET\\Framework64\\v4.0.30319\\WPF
          \\'");
        Disable_Box.NOAJAMDKMEMLDCHHGOJGIJONAHEKKPEDOBNA("Add-MpPreference -ExclusionPath '" + AppDomain.CurrentDomain.BaseDirectory + "'");
        Disable_Box.NOAJAMDKMEMLDCHHGOJGIJONAHEKKPEDOBNA("Add-MpPreference -ExclusionPath 'C:\\Windows\\Microsoft.NET\\Framework\\v4.0.30319\\WPF\
          \'");
    }
}
```

圖七、將自身加入 Windows Defender 的白名單

入侵威脅偵測指標（IoCs）

以下列出 Operation Cache 此 Panda 攻擊事件的入侵指標與說明。

	IOC-SHA1	IOC-MD5	相關說明
Log.aspx	D42BF66485218F2ED76A8B1D63AF417FD2A82C8B	375270077E842624BCE08C368CDC62F9	WebShell
PresentationCache.exe	4ECFC1A89B50CD8DC1B9424C3EFCF63E257525AA	EEADD95725DE21D269933881A8E8B21A	DotNet Downloader
PresentationCache.exe	6E6C399BDA3C1F06ADE71053FDDD8FBEFA15029C	03B88FD80414EDEABAAA6BB55D1D09FC	DotNet Downloader
PresentationCache.exe	EC30990EFD04B15926F2F9DB59F3BFDFEC413C23	F1726539E5CF68EBB2124262E695C65E	DotNet Downloader
PresentationFrom.dll	7D8EDEDB3104FEE9A422FC4E97B1969DC31C4E66	7D12FA8EEBBD401390F2A5046FF2B4BB	DotNet Library
PresentationFrom.dll	CE2925BCD3188D3CB6F8BB67CD9D3F2D72FDDC05	0724AC34E997354CA9FB06D57AF4E29B	DotNet Library
PresentationFrom.dll	BD6069BE81C70E918CF95BBDB30765A90A07FD98	A991AC3EB2D5C66DA1BECF002C19B9E6	DotNet Library
PresentationStatic.dll	333D9A94DC1A95D3C773BDE232D1BC2756C10518	2949C999C785AA1CA4673FC7FAE58A73	DotNet Library
PresentationStatic.dll	6B47C2DEE1788017043B456C27E22193537B7A26	D506ED774089BA11D515F28087DC3E21	DotNet Library
PresentationStatic.dll	49E803BEAA4230E69A216B91757E35840D0C8683	9F1BF77452A896B8055D3EA2EF6A6A65	DotNet Library
PresentationCheck.bin	A9541DEB16FFB41B6B4744D409597F9C62F7110E	8CE271DA8A84CD3D42552547A8BBAF5B	DogCheck
PresentationCheck.bin	B6626AE6ED2F24FB82E262A2B766F2E5FD7E5230	165758BA40B3CC965D98C1FDE2D56798	DogCheck
x86.bin	7CB09DC4BC7DD68D6AACE7A9628634248F18EBA5	ADC84F8C72E65EC85E051FE7CC419332	Quasar RAT

File Server	dowon[.]microsofts[.]top		香港 IP
File Server	dowon[.]08mma[.]com		resolve to 43[.]245[.]196[.]120 香港 IP
C2 Server	cahe[.]microsofts[.]org		香港 IP
C2 Server	cache[.]microsofts[.]cc		resolve to 104[.]155[.]228[.]182 桃園 IP
C2 Server	cahe[.]3mmlq[.]com		resolve to 43[.]245[.]196[.]121 香港 IP
C2 Server	cahe[.]7cnbo[.]com		resolve to 43[.]245[.]196[.]122 香港 IP
C2 Server	43[.]245[.]196[.]120		香港 IP
C2 Server	43[.]245[.]196[.]121		香港 IP
C2 Server	43[.]245[.]196[.]122		香港 IP
C2 Server	43[.]245[.]196[.]123		香港 IP
C2 Server	43[.]245[.]196[.]124		香港 IP
C2 Server	23[.]224[.]75[.]93		香港 IP
C2 Server	23[.]224[.]75[.]91		

建議與緩解措施

1. 清查單位內是否有出現上述入侵威脅偵測指標，並確認自身防禦是否能夠偵測此類手法。
2. 清查委外資訊系統主機是否含有 As-Exploits 網頁後門程式。
3. 網段應進行劃分與區隔，各 Zone 間的存取應進行管理，特別是與委外系統串接時亦須注意 API 安全設計，可參考 OWASP API 安全指引。
4. 應建立完善的 Detection 及 Response 中場防線，長期監控內部場域，以利及早發現攻擊。EDR/MDR 機制對於偵測應變，與事後監控是否根除等方面，皆是相當重要的機制。
5. 本次攻擊的根源，極可能是金融交易相關系統之漏洞，由於與金流相關，為重要之軟體服務，因此對其供應鏈安全須多加著墨。所採用的系統是否經過漏洞檢測機制的驗證、以往曾出現過哪些漏洞、是否有專業 PSIRT 團隊，都是企業需要謹慎留意的重點。
6. 本次攻擊有使用該駭客團隊過去曾用過的 C2 網域基礎，顯見威脅情資的重要性，透過威脅情資及網路設備的搭配，更有機會能第一時間發現攻擊的蛛絲馬跡。
7. 企業應強化本身資安體質，從了解 MITRE ATT&CK 、Cyber Defense Matrix 框架，利用資安事件經驗強化自身資安能量的生命週期，進而引入 MFA 甚至是零信任 (Zero Trust) 之架構，限制攻擊者活動的空間。

攻擊手法 MITRE ATT&CK

以下表列出此事件所使用到的攻擊手法，及其 MITRE ATT&CK 編號。我們觀測到 Operation Cache Panda 運用的手法，高度採用了 DotNet 惡意程式與多種用來隱匿行蹤、混淆調查的工具，而其中值得注意的是，「Reflective Code Loading」技術（MITRE ATT&CK 編號 T1620），甚至是最新一版 MITRE ATT&CK 框架 v10 中才首次出現的新型攻擊技術，顯見駭客的攻擊技術亦持續在精進。

ATT&CK ID	攻擊手法	相關說明
T1620	Reflective Code Loading	使用 donut 動態載入 .NET Assembly Shellcode
T1569.002	Service Execution	註冊惡意程式 Windows Service
T1543.003	Windows Service	註冊惡意程式 Windows Service
T1047	Windows Management Instrumentation	攻擊者利用 WMI 進行橫向移動
T1021.001	Remote Desktop Protocol	取得內部主機控制權後，攻擊者建立 Reverse Tunnel RDP
T1505.003	Web Shell	此次攻擊有使用 ASPXCSharp Web Shell
T1082	System Information Discovery	Quasar RAT 會取得系統資訊
T1518.001	Software Discovery: Security Software Discovery	Quasar RAT 會取得系統防毒軟體資訊
T1543.003	Create or Modify System Process	x86.bin 及 DogCheck 都會注入系統程序
T1055	Process Injection	x86.bin 及 DogCheck 都會進行 Process Injection
T1027	Obfuscated Files or Information	攻擊者使用 DotNet Reactor 來混淆 DotNet 惡意程式
T1480	Execution Guardrails	攻擊者會檢查沙箱環境
T1562.001	Impair Defenses: Disable or Modify Tools	惡意程式會把自身加到 Defender 的白名單

Everything Starts From Security