MoonWalk：深入了解 APT41 更新后的武器库（第 2 部分）

admin：

介绍

这是我们对 APT41 的新工具 DodgeBox 和 MoonWalk 进行技术深入研究的第二部分。有关 DodgeBox 的详细信息，请参阅 第 1 部分。

在本博客系列的第 2 部分中，我们将研究 APT41 工具包的新成员 MoonWalk 后门。延续我们在 第 1 部分中对 DodgeBox 加载程序的分析，我们发现 MoonWalk 共享了几种规避技术。它利用 Google Drive 进行命令和控制 (C2) 通信，并滥用 Windows Fibers（一个鲜为人知的 Windows 功能）来规避防病毒 (AV) 和端点检测与响应 (EDR) 解决方案。

关键要点

- APT41 是一个总部位于中国的民族国家威胁行为者，以在东南亚开展活动而闻名，据观察，它使用了一种名为 MoonWalk 的新后门。
- MoonWalk 与 DodgeBox 共享一个通用的开发工具包，重复使用实现规避技术的代码，例如 DLL 挖空、导入解析、DLL 解除挂钩和调用堆栈欺骗。此外，MoonWalk 还采用了进一步的规避策略，包括使用 Google Drive 作为其 C2 通道以融入合法网络流量，以及利用 Windows Fibers 来规避 AV/EDR 安全解决方案。
- MoonWalk 的模块化设计使攻击者可以轻松更新其功能、修改其行为并针对不同场景定制功能。

技术分析

攻击链

这篇博文的重点是攻击链的后半部分，从内存中执行 MoonWalk 后门开始。一旦 DodgeBox 成功加载 MoonWalk 后门，恶意软件就会解密并反射加载两个嵌入式插件（C2 和 Utility）。C2 插件使用自定义加密的 C2 协议与攻击者控制的 Google Drive 帐户进行通信。

下图显示了使用 DodgeBox 加载器部署 MoonWalk 的攻击链。

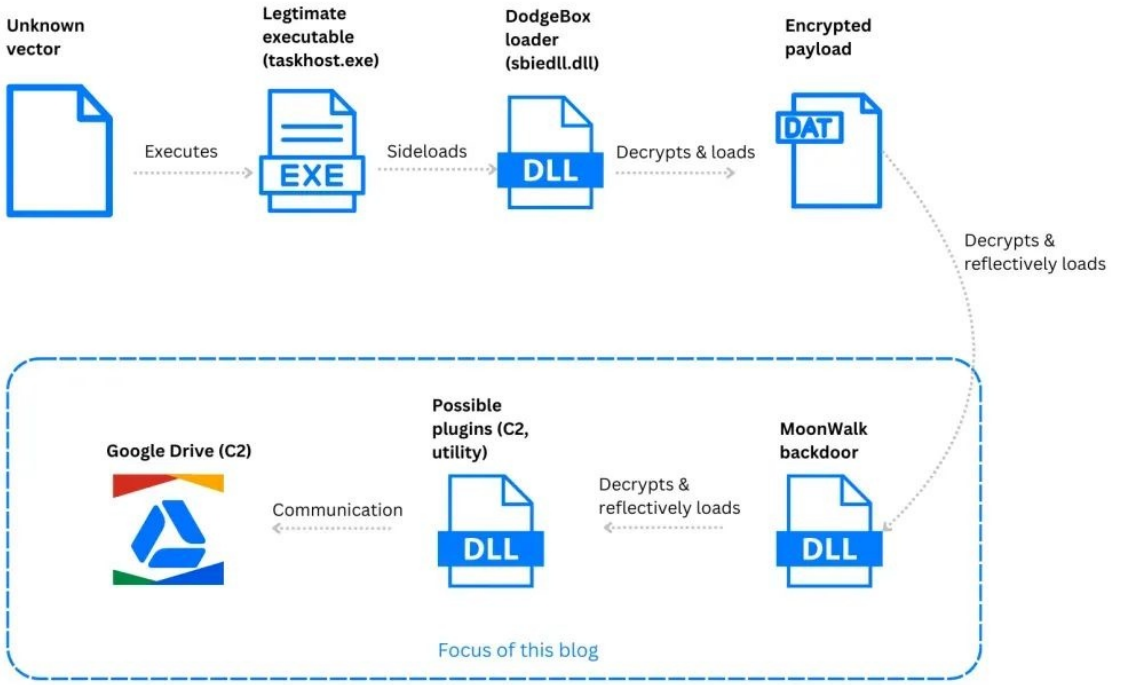

图 1：用于部署 DodgeBox 加载程序和 MoonWalk 后门的攻击链。

MoonWalk 分析

MoonWalk 是一个用 C 语言编写的恶意软件后门，其代码与 DodgeBox 有许多相似之处，表明它们使用相同的开发工具包。它整合了 DodgeBox 的许多逃避相关功能，包括与以下内容相关的功能：

- DLL 挖空
- 导入分辨率
- DLL 解除挂钩
- 调用堆栈欺骗

此外，MoonWalk 使用与 DodgeBox 相同的 DLL 阻止列表。

ThreatLabz 分析揭示了 MoonWalk 的模块化设计，使其能够根据需要加载不同的插件组件。ThreatLabz 检查的样本包含两个嵌入式插件，一个用于 C2 通信的 C2 插件，以及一个提供与压缩和公钥加密相关的功能的实用程序插件。这种模块化架构使 MoonWalk 具有高度的适应性，使攻击者能够针对不同场景定制其功能。

在下面的部分中，我们将重点介绍 MoonWalk 的几个显著功能。

卸载 DodgeBox 装载机

MoonWalk 首次初始化时，会使用与 DodgeBox 相同的算法解析其导入。然后，根据 DodgeBox 配置参数 Config.fShouldUnloadStealthVector，MoonWalk 从内存中卸载 DodgeBox DLL 并将其与进程环境块 (PEB) 解除链接。这减少了 MoonWalk 的内存占用，并混淆了其来源，从而阻碍了内存取证分析。

使用 Windows Fiber

接下来，MoonWalk 初始化用于管理 Windows Fibers 的全局结构。Windows Fibers 是一种轻量级线程机制，自 Windows NT SP5 开始在 Windows 操作系统中可用。与由操作系统调度的传统线程不同，Fibers 由应用程序本身协作

调度。这允许开发人员针对特定工作负载调整应用程序的性能。然而，由于使用 Windows Fibers 的复杂性以及计算机硬件性能的改进，Windows Fibers 并未得到广泛采用，并且仍然是一个鲜为人知的功能。

然而，随着近年来人们对网络安全的关注度不断提高，研究和红队社区对 Windows Fibers 的兴趣也日益浓厚。已经发表了多篇研究论文（1、2、3） 和 开源概念 验证(POC)，利用 Windows Fibers 来逃避 AV/ EDR解决方案。

APT41 可能一直在关注这些发展，因为他们已将 Windows Fibers 整合到 MoonWalk 后门中。从高层次来看，MoonWalk 维护着一个全局的 Fiber 数组。当需要将某个函数作为 Fiber 执行时，将使用 API 创建 Fiber CreateFiber。然后，将此 Fiber 与函数的地址及其参数和其他元数据一起打包，并插入到全局数组中。然后，主 Fiber 安排这些 Fiber 执行。这种 Windows Fibers 的使用有助于 MoonWalk 逃避不支持扫描 Windows Fibers 的 AV 和 EDR，并且通过分解控制流使分析变得具有挑战性。

配置

MoonWalk 解密其配置，该配置在其部分中被硬编码 .lrsrc。与 DodgeBox 一样，MoonWalk 使用 MD5 进行配置验证，并使用 AES 密码反馈 (AES-CFB) 进行解密。

然而，MoonWalk 的配置更为复杂，具有嵌套结构和数组。此配置包含各种执行参数，包括以下内容：

- 互斥名称
- 光纤配置
- 心跳间隔
- 加密密钥
- C2相关数据

在我们分析的样本中，MoonWalk 的配置（称为 Config）包括用于与攻击者控制的 Google Drive 帐户进行身份验证的 OAuth 机密，以及其他值得注意的字段，如下所示：

```
Config.szClientID:
XXXXXXXX3108-0pm3bsjc0mto2e1k4kp2u8817lgk3e3v.apps.googleusercontent.com

Config.szClientSecret:
XXXXXXXXBiuo8VPZUH1dBHkv86mC1xFU_Z3

Config.szRefreshToken: XXXXXXXXiYDPmH9cCgYIARAAGAkSNwF-
L9IrcM7YiuxWrNuyIfKINyNc_pEVytGNNK750ZyyIm32qH6Wh3dGIBTvdPJ2v92xAohHwWw

Config.rgbXorKey:
a8e6bd132daf0360b1af1f5eea15e42f8c6f1dcd7d34376ae4e83a1a4f5907c0

Config.szMutexName:
GlobalctXjvsAxpzyqElmk

Config.szName:
default
```

加载默认配置后，MoonWalk 会在 处搜索新配置文件C:ProgramData[MD5(Config.rgbIDBytes)]。如果找到，恶意软件将解密并加载此文件。本博客的附录中提供了 MoonWalk 解密配置的示例，供参考。

解压并加载插件

然后，MoonWalk 从该部分中提取嵌入的插件 .lrsrc。在我们分析的 MoonWalk 样本中，此部分嵌入了两个插件：一个用于 C2，另一个提供公钥加密和压缩等实用功能的插件。

该 .lrsrc部分中的每个插件都以 72 字节的元数据作为前缀，其中包括 AES-CFB 密钥、MD5 校验和以及插件类型信息。插件类型信息字段提供有关插件功能的信息。这些字段有助于识别插件是用作命令处理程序、C2 还是实用程序。有关插件元数据结构的更多详细信息，请参阅附录部分。

MoonWalk 通过将这些插件注册到全局链接列表中来组织它们。然后，MoonWalk 通过此列表使用 DLL 挖空技术加载 C2 插件及其依赖项（例如实用程序插件）。此过程类似于我们之前在 第 1 部分中针对 DodgeBox 描述的过程。与 DodgeBox 一样，此 MoonWalk 示例将主机 DLL 的副本存储在 中 C:WindowsMicrosoft.NETassemblyGAC_MSILSystem.Data.Trace。

网络通信

加载 C2 插件后，MoonWalk 准备与 C2 服务器建立通信。MoonWalk 使用 Google Drive 进行 C2 通信。这有助于 MoonWalk 逃避检测，因为往返于信誉良好的云服务的流量不太可能引起怀疑，尤其是当目标已经在使用此服务时。奇怪的是，MoonWalk 在发出 HTTP 请求时使用字符串 curl/7.54.0 作为其用户代理，即使它没有在其 C2 插件中使用 libcurl，而是使用 WinHTTP 系列 API。

从高层次来看，MoonWalk 通过以下方式通过 Google Drive 进行通信：

步骤	描述
	`Config.szClientID`MoonWalk 通过利用其配置中的 OAuth 机密（ 、 `Config.szClientSecret`和 ） 从 Goo 权服务器获取访问令牌。`Config.szRefreshToken`
1 - 初始化	MoonWalk 生成 16 个随机字节，并对其进行十六进制编码，得到一个字符串，例如： `f137da1a9019849fbc2aac49a4b6f2c3`.我们将此字符串引用为 `SessionID`。
	MoonWalk 使用 Google Drive API 来检索目录的 ID `/data`。
	MoonWalk 检索目录的 ID `/data/temp`。
2 -加密握手（客户端 Hello 和服务器 Hello）	MoonWalk 会在 /data/temp 中搜索以生成的文件命名的文件 `SessionID` （即 `f137da1a9019849fbc2aac49a4b6f2c3`）。如果找不到该文件，MoonWalk 会生成并上传件 `/data/temp/[SessionID]`来启动加密握手并与服务器交换 AES 密钥。
	然后，MoonWalk 会查找 `/data/[SessionID]` 目录及其子目录 `/data/[SessionID]/s1`。标题为的目录 `s[nu` 似乎是 MoonWalk 检索和下载即将到来的 C2 指令的指定位置。
	最后，MoonWalk 搜索 `/data/[SessionID]/s1/1`文件。当文件可用时，MoonWalk 会下载并处理它，并完握手。
3 -信息收集	MoonWalk 随后检查目录是否存在 `/data/[SessionID]/c1`，如果不存在则创建目录。然后，MoonWalk 收机名称、用户名和操作系统版本等信息，并将其上传到文件 `/data/[SessionID]/c1/1`。
	然后，MoonWalk 通过使用当前 Unix 时间戳作为字符串更新名为“ ”的文件，定期向 C2 服务器发送心跳。`temp.txt`
4 -心跳	MoonWalk 还会定期轮询 `/data/[SessionID]/s1`目录中是否有新文件。如果发现新文件，MoonWalk 会对处理并将其响应上传到目录中 。在我们对 MoonWalk 的分析过程中，我们只观察到了 ping 命令，MoonV 过将包含当前 Unix 时间戳的`/data/[SessionID]/c1`编码文件上传到目录来做出响应 。`/data/[SessionID]`

表 1：使用 Google Drive 的 MoonWalk C2 通信协议的高级视图。

加密握手（客户端问候）

在加密握手阶段，MoonWalk 使用自定义协议与服务器交换 AES 密钥。因此，如果无法访问这些 AES 密钥（这些密钥仅存在于 MoonWalk 的进程内存中），则很难甚至无法解码加密的 C2 消息。

该过程首先由 MoonWalk 使用自定义随机数生成器生成 32 字节 AES 密钥 (rgbClientAESKey) 和 16 字节初始化向量 (IV) (rgbClientAESIV)。然后，AES 密钥被视为椭圆曲线 Diffie-Hellman (ECDH) 私钥，以使用该函数生成相应的 ECDH 公钥 (rgbECDHPublicKey) curve25519_donna。

然后，MoonWalk 通过将 ECDH 公钥和 AES IV 与 MoonWalk 配置中的 XOR 密钥进行 XOR 运算来对它们进行编码 (Config.rgbXorKey)。通过对 、ECDH 公钥和 AES IV 的串联执行 MD5 哈希 Config.rgbXorKey，然后取哈希的前四个字节来创建校验和。最后，MoonWalk 将此数据上传到路径 的 Google Drive /data/temp/[SessionID]。

下图显示了上传文件的内容：

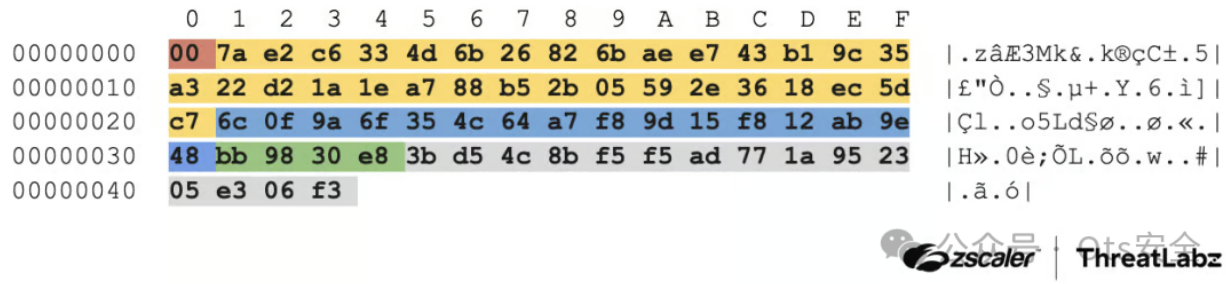

图 2：MoonWalk Client Hello 密钥交换消息的内容。

下表提供了上传文件中包含的各个字段的描述：

Offset	大小（以字节为单位）	描述
0x00	1	未知字段，可能是消息类型枚举。

0x01	32	`rgbECDHPublicKey`与异或 `Config.rgbXorKey`
		
		异或运算前的rgbECDHPublicKey为：
		
		`d2 04 7b 20 60 c4 25 e2 da 01 f8 1d 5b 89 d1 8c ae bd 07 d3 da bc 82 41 e1 b1 14 2c 57 b5 5a 07`
		
		`rgbClientAESIV`与异或 `Config.rgbXorKey`

0x21	16	`rgbClientAESIV`XOR 操作之前是： `c4 e9 27 7c 18 e3 67 c7 49 32 0a a6 f8 be 7a 67`
0x31	4	前四个字节 `MD5 (Config.rgbXorKey | rgbECDHPublicKey | rgbClientAESIV)`
0x35	15	未知字节。

表2：MoonWalk Client Hello 密钥交换消息的描述。

加密握手（服务器问候）

然后，MoonWalk 下载位于 的文件 /data/[SessionID]/s1/1。此文件包含服务器对上述 MoonWalk 握手的响应。

此文件以及所有后续上传或下载的文件均使用自定义方案进行编码。这里，我们将以 Server Hello 文件为例，介绍此方案的解码过程。

下图是编码后的 Server Hello 文件的整体布局示例：

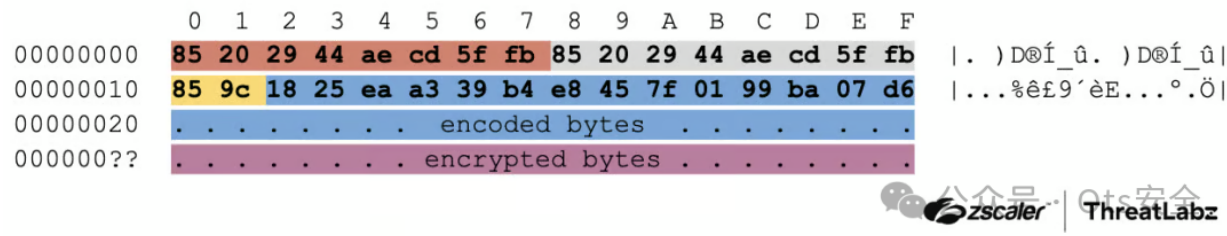

图 3：MoonWalk Server Hello 消息格式。

下表显示了这些字段的描述。

Offset	大小（以字节为单位）	描述

Offset	大小（以字节为单位）	描述
		`rgbFileXorKey`
0x00	8	用于解码的 XOR 密钥 `rgbEncodedBytes`。
0x08	8	未知，可能是消息类型字段。
		`dwNumEncodedBytes`
0x10	2	后面的编码字节数。此字段用编码 `rgbFileXorKey`。解码此字段显示此文件内有 0xbc 编码字节。 `85 20 XOR 85 9c = 00 bc`
		`rgbEncodedBytes`

0x12	编码字节数	此文件内的编码字节。这些字节似乎包含消息元数据，例如 Google Drive 文件 ID、消息标头或垃圾字节。
		
		要解码这些字节， `rgbFileXorKey`从 XOR 密钥的第三个字节开始。
		
		`18 25 ea a3 39 b4 e8 45 7f 01 99 ba 07 d6`
		
		`XOR`
		
		`29 44 ae cd 5f fb 85 20 29 44 ae cd 5f fb`
		
		`=`
		
		`31 61 44 6e 66 4f 6d 65 56 45 37 77 58 2d`

		`rgbEncryptedBytes`
0x??	多变的	文件的其余部分未被编码，因为此部分通常使用 AES-CFB 加密，使用在加密握手阶段交换的 AES 密钥。

图 3：MoonWalk Server Hello 消息格式。

下表显示了这些字段的描述。

```
          0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
00000000  85 20 29 44 ae cd 5f fb 00 00 00 00 00 00 00 00   |................|
00000010  00 bc 31 61 44 6e 66 4f 6d 65 56 45 37 77 58 2d   |.¼1aDnfOmeVE7wX-|
00000020  42 63 77 7a 2d 6e 70 68 47 34 47 58 79 2d 69 64   |Bcwz-nphG4GXy-id|
00000030  57 68 69 00 7e ab 88 27 ff 95 aa cf 53 d2 67 eb   |Whi.~«.'ÿ.ªÏSÒgë|
00000040  eb 91 1e 33 c3 56 a6 4d 66 17 b8 23 a7 b0 d4 54   |ë..3ÃV¦Mf.¸#§°ÔT|
00000050  cf 2e 9f 00 61 36 b0 61 f5 ca 62 1b ba f0 d4 ab   |Ï...a6°aõÊb.°ðÔ«|
00000060  87 0f 82 15 b8 1d 5f 31 ac 21 25 a5 2f 77 fa b4   |....¸._1¬!%¥/wú´|
00000070  d2 c9 86 20 60 9f 32 bc 92 c4 a3 19 4f 17 5d 6f   |ÒÉ. `.2¼.Ä£.O.]o|
00000080  77 c3 66 37 8c a0 16 9f 8c e6 7c 51 bb 1e 9d 1f   |wÃf7. ...æ|Q»...|
00000090  05 00 9c b1 67 5a 0a ef ff 60 3e 61 24 d1 a0 9d   |...±gZ.ïÿ`>a$Ñ .|
000000a0  c9 32 02 29 b7 8c f8 d3 67 7a a2 69 af a1 2d c8   |É2.)·.øÓgz¢i¯¡-È|
000000b0  af eb 88 eb 2a d1 db 5d 3e 60 53 7b 83 5e 03 18   |¯ë.ë*ÑÛ]>`S{.^..|
000000c0  91 fc e7 6f 3b 74 ee 54 f9 b2 c3 a6 3a b6 df 64   |.üço;tîTù²Ã¦:¶ßd|
000000d0  d9 00 29 b9 97 ba 84 7d 01 e6 cd c2 d1 d0 8e e5   |Ù.)¹.°.}.æÍÂÑÐ.å|
000000e0  5a 48 2b 75 1e 31 2f d3 12 38 bd 30 3a fa ad 9b   |ZH+u.1/Ó.8½0:ú..|
000000f0  6f 1e 08 27 98 17 5e 67 fb c1 33 44 d1 82 05 f5   |o..'..^gûÁ3DÑ..õ|
00000100  65 bc 83 3b a5 6f ab 58 3b b7 49 7b ed cf 0a      |e¼.;¥o«X;·I{íÏ.|
```

zscaler ThreatLabz

图 4：解码后的 MoonWalk Server Hello 消息的示例内容。

解码后的 Server Hello 字段如下表所示。

Offset	大小（以字节为单位）	描述
0x00	8	`rgbFileXorKey` XOR密钥，用于解码 `rgbEncodedBytes`。
0x08	8	未知
0x10	2	`dwNumEncodedBytes`
0x12	多变的	`szHeartBeatFileID` 心跳文件的 Google Drive ID `temp.txt`，。
0x34	多变的	未知 编码缓冲区，与 进行 XOR 编码 `Config.rgbXorKey`。 解码后将显示以下字段：

0xce	四十八	`rgbServerECDHBasePoint` - 用作 ECDH 基点，MoonWalk 稍后使用它来生成服务器使用的共享 AES 密钥。
		`77 82 64 13 04 16 94 da 35 d2 1e b8 27 d7 35 ff`
		`02 8a 47 85 56 41 29 5b cb 3b 28 22 f2 69 3d 3a`
		解码后的剩余字节包含校验和以及其他未知字节。

0xfe	4	校验和生成者 `MD5 (rgbServerECDHBasePoint | Config.rgbXorKey.)`
0x102	多变的	未知

表 4：MoonWalk Server Hello 消息中字段的描述。

rgbECDHServerPublicKey利用此信息，MoonWalk使用 函数生成公钥 ( ) curve25519_donna。然后，rgbECDHServerPublicKey 对其进行异或运算 Config.rgbXorKey以生成服务器 AES 密钥。

```
Curve25519_Donna(
    a1->rgbECDHServerPublicKey,
    // Public Key (out):
    // 000001e6`246391ec b5 8f a7 ee 0b da d6 79-79 60 85 79 bf 32 ad 91
    // 000001e6`246391fc 24 a3 39 66 4c 4b 49 97-6c 71 92 d3 55 45 4b 3e
    a1->rgbClientAESKey,
    // Private Key:
    // 000001e6`2463920c 54 be fd a7 f4 0f 62 15-fb 22 9a 48 04 e3 6e 90
    // 000001e6`2463921c 85 4b b9 c7 f2 5f de 57-65 59 9c 90 18 04 d9 d1
    a1->rgbECDHServerBasepoint);
    // Basepoint:
    // 000001e6`24639251 77 82 64 13 04 16 94 da-35 d2 1e b8 27 d7 35 ff
    // 000001e6`24639261 02 8a 47 85 56 41 29 5b-cb 3b 28 22 f2 69 3d 3a

rgbServerAESKey = rgbECDHServerPublicKey ^ Config.rgbXorKey
// 1d 69 1a fd 26 75 d5 19-c8 cf 9a 27 55 27 49 be
// a8 cc 24 ab 31 7f 7e fd-88 99 a8 c9 1a 1c 4c fe
```

通过这种方式，MoonWalk 与其 C2 交换 AES 密钥，从而完成加密握手。

信息收集

在此阶段，MoonWalk 会收集有关环境的信息并将其上传到 Google Drive。收集的数据包括处理器架构、Windows 产品类型、版本和内部版本号、计算机和用户名以及 IP 地址等详细信息。然后使用 LZ4 压缩此信息。然后使用 32 位 MurmurHash2 算法添加校验和，使用自定义混合常数，其中 r设置为 15，初始种子设置为 0x12345678。然后使用服务器的 AES 密钥使用 AES-CFB 加密这些字节，并使用上面详述的自定义方案进行打包，然后上传到 Google Drive。

本博客的附录中提供了有关收集的环境信息的更多详细信息。

心跳

MoonWalk 还会定期向服务器发送心跳。它会 使用在加密握手过程中检索到的temp.txt文件 ID，将当前 Unix 时间戳以纯文本形式上传到 Google Drive 上的一个文件 中。szHeartBeatFileID

后门功能

在我们对 MoonWalk 的分析中，我们没有观察到 C2 发送任何其他命令或插件。如果dwPluginTypePart2 == 1 未找到命令处理程序插件（附录中描述），MoonWalk 默认使用内置处理程序列表。这些处理程序包含以下功能：

- 收集环境信息（类似于上面的信息收集步骤）
- 窃取令牌（令牌冒充）
- 创建令牌（使用给定的凭据登录到 Windows 机器）
- 下载新配置
- 执行命令行命令

注意：此列表并不完整，因为需要进一步分析。

结论

MoonWalk 是一种复杂且模块化的后门，它采用了 DodgeBox 中常见的规避技术。它还引入了创新技术，包括使用不常见的 Windows Fibers。这些规避技术与使用自定义复杂 C2 通信协议相结合，滥用 Google Drive 与合法流量混合，凸显了 APT41 威胁对手的高超技能。

我们将继续密切监视该威胁行为者的最新策略、技术和程序 (TTP)，以保护我们的客户并与安全社区分享研究成果。

Zscaler 覆盖范围

Zscaler 的多层云安全平台检测到与 DodgeBox 相关的各个层面的指标，威胁名称如下。

- Win64.Backdoor.Moonwalk

攻击指标 (IOC)

MD5 哈希	描述
5b1e8455291d99a1724327b9a7fc2616	MoonWalk 后门（与 DodgeBox 加载程序相关，MD5 为：d72f202c1d684c9a19f075290a60920f）。
b69984cbf52b418673bd08279ca845d6	实用插件
5217b8552321556ea434474377cfcd02	C2插件
bfd6286bb39a0e24a2af28c63bd8e194	MoonWalk 后门（与 DodgeBox 加载程序相关，MD5 为：393065ef9754e3f39b24b2d1051eab61）。
75bfb7d5199bf0c4e62525099b33e14f	C2插件
f68ef9e40462c9760bf9c829edd9f4a9	实用插件

实体	描述
GDrive OAuth 客户端 ID #1	XXXXXXX5917-dudeis843uv3v1lrm1n12jbq9l9a86lq.apps.googleusercontent.com
GDrive 客户端秘密 #1	XXXXXXXX8OPdXrMnPIbIvODh4bnYTVtdKJY
GDrive 刷新令牌 #1	XXXXXXXXEqC4HrQVCgYIARAAGAkSNwF-L9IrS7n6zr6G_vE7_huP5uJuMT6aMtOnu3WgmTMRiEc5QJaQgVX4gbUV7ItUbFXVmd5
GDrive OAuth 客户端 ID #2	XXXXXXX3108-0pm3bsjc0mto2e1k4kp2u8817lgk3e3v.apps.googleusercontent.com
GDrive 客户端秘密 #2	XXXXXXXXBiuo8VPZUH1dBHkv86mC1xFU_Z3
GDrive 刷新令牌 #2	XXXXXXXXiYDPmH9cCgYIARAAGAkSNwF-L9IrcM7YiuxWrNuyIfKINyNc_pEVytGNNK750ZyyIm32qH6Wh3dGIBTvdPJ2v92xAohHw
威胁行为者的电子邮件地址（链接到 GDrive）	jsonmakesam@gmail.com

心跳相关的网络请求	描述

动词：
`PATCH`
网址：
`https://www.googleapis.com/upload/drive/v3/files/[redacted_id]`
URL 参数：
`uploadType=media&fields=id,name,size,mimeType,modifiedTime`
用户代理：
`curl/7.54.0`
	MoonWalk `temp.txt`使用 I 间戳进行更新。

动词：
`GET`
网址：
`https://www.googleapis.com/drive/v3/files`
URL 参数：
`fields=files(id,name,size,mimeType,modifiedTime)&q=[redacted_id]%20in%20parents%20and%20trashed%20%3D%20false%20&pageSize=300`
用户代理：
`curl/7.54.0`
	MoonWalk 正在查询新命令

MITRE ATT&CK 框架

策略	ID	技术	描述
防御规避	T1027	模糊文件或信息	MoonWalk 使用 AES-CFB 来加密字符串、配置和捆绑的有效负载。
防御规避	T1027.007	模糊文件或信息：动态 API 解析	MoonWalk 使用加盐的 FNV1a 哈希来动态解析 API。
防御规避	T1620	反射代码加载	MoonWalk 利用 DLL 挖空技术反射加载插件 DLL。
防御规避	T1106	本机 APINative API	MoonWalk 使用 Windows 原生 API，例如 `NtCreateFile`、、 `LdrLoadDll` 和 `NtAllocateVirtualMemory`，而不是 Win32 对应 API。
防御规避	T1562.001	削弱防御能力：禁用或修改工具	MoonWalk 在调用 API 来监控安全软件时利用堆栈欺骗。 MoonWalk 会在自己的地址空间内执行扫描，以检测任何更改，例如挂钩或调试器断点。发现任何修改迹象，DodgeBox 会采取行动从磁盘恢复原始代码，从而有效地撤消对其做的任何未经授权的更改。
指挥与控制	T1102.002	Web 服务：双向通信	MoonWalk 有一个 C2 插件，它利用攻击者控制的 Google Drive 帐户来实现 C2 通信通道
指挥与控制	T1573	加密通道	MoonWalk 利用自定义网络协议来交换加密的 C2 消息。
侦察	T1592	收集受害者主机信息	MoonWalk 收集有关受害者主机的硬件和软件配置的信息。
侦察	T1590	收集受害者网络信息	MoonWalk 收集受害者主机的 IP 地址。

收集环境信息

这是 MoonWalk 在初始化过程中收集并上传到 GoogleDrive 的环境信息列表。

- 受害者哈希（计算机名称和机器的 GUID连接的 FNV1a 哈希 ）。Config.rgbIDBytes
- Windows 主版本号和次版本号
- Windows 内部版本号
- 计算机名称
- 用户名
- 可执行路径（当前进程的可执行文件的完整路径）。
- 冒充状态
- CPU 启动时间
- IPv4 地址
- IPv6 地址
- Config.rgbIDBytes
- Config.wszConfigName
- MoonWalk 配置中的各种心跳相关间隔字段。

原文始发于微信公众号（Ots安全）：MoonWalk：深入了解 APT41 更新后的武器库（第 2 部分）