مهندسی معکوس بدافزار

نویسنده گان:

Ahmet BALCI

Dan UNGUREANU

Jaromír VONDRUŠKA

CCDCOE
NATO COOPERATIVE CYBER DEFENCE CENTRE OF EXCELLENCE
مرکز همکاری دفاع سایبری سازمان ناتو

ترجمه و تنظیم :

بهادر اکرمی

CFCL
www.cfcl.ir
Computer For Convenient Life

تالین ۲۰۲۰

مرکز همکاری دفاع سایبری ناتو (CCDCOE) یک مرکز مدافعتی سایبری با اعتبار ناتو است که بر روی تحقیقات، آموزش و تمرینات تمرکز دارد. این مرکز جامعه‌ای از ۲۵ کشور را نماینده می‌شود و دید ۳۶۰ درجه از دفاع سایبری را با تخصص در زمینه‌های فناوری، استراتژی، عملیات و قانون ارائه می‌دهد. قلب این مرکز یک گروه متنوع از کارشناسان بین‌المللی از پس‌زمینه‌های نظامی، دولتی، آموزشی و صنعتی است. CCDCOE میزبان Tallinn Manual 2.0 است، جامع‌ترین راهنمایی در مورد اینکه قانون بین‌المللی چگونه برای عملیات سایبری اعمال می‌شود. این مرکز بزرگ‌ترین و پیچیده‌ترین تمرین دفاع سایبری با آتش باز بین‌المللی جهان، Locked Shields را برگزار می‌کند و کنفرانس بین‌المللی در مورد تعارض سایبری، CyCon را برگزار می‌کند. همچنین به‌عنوان رئیس دپارتمان آموزش و آموزش عملیات فضای سایبری، CCDCOE مسئول شناسایی و هماهنگی راه‌حل‌های آموزشی و آموزشی در زمینه عملیات دفاع سایبری برای تمام ارگان‌های ناتو در سراسر اتحادیه است. این مرکز توسط کشورهای عضو خود تأمین مالی می‌شود و در حال حاضر اتریش، بلژیک، بلغارستان، جمهوری چک، دانمارک، استونی، فنلاند، فرانسه، آلمان، یونان، مجارستان، ایتالیا، لتونی، لیتوانی، هلند، نروژ، لهستان، پرتغال، رومانی، اسلواکی، اسپانیا، سوئد، ترکیه، انگلستان و ایالات متحده آمریکا هستند.

www.ccdcoe.org publications@ccdcoe.org

سلب مسئولیت

این متن یک سری اطلاعات قانونی را در مورد مرکز همکاری دفاع سایبری ناتو (CCDCOE) ارائه می‌دهد. این متن بیان می‌کند که این انتشار یک محصول از CCDCOE است و نظرات و سیاست‌های این مرکز یا ناتو را بیان نمی‌کند. همچنین، CCDCOE مسئول هیچ‌گونه خسارت یا ضرر ناشی از استفاده از اطلاعات موجود در این انتشار نیست و مسئول محتوای منابع خارجی، از جمله وب‌سایت‌های خارجی مورد اشاره در این انتشار نیست. این انتشار می‌تواند برای استفاده داخلی در ناتو و برای استفاده شخصی یا آموزشی برای اهداف غیرانتفاعی و غیرتجاری تولید شود، به شرطی که نسخه‌ها دارای یک استناد کامل باشند.

فهرست مطالب

چکیده

بدافزار یک تهدید روبه‌رشد است که برای افراد، شرکت‌ها و مؤسسات هزینه قابل‌توجهی ایجاد می‌کند. ازآنجایی‌که دفاع‌های ضدویروس مبتنی بر امضاهای ساده، در برابر تهدیدات بدافزارهای تازه ظاهر شده یا حملات APT، کارآمد نیستند، برای یک محقق، داشتن مهارت‌های بنیادین برای تجزیه‌وتحلیل این تهدیدات ضروری است. درحالی‌که برای موارد خاص، اقدامات خاصی باید انجام شود، این کتابچه راهنما به‌طورکلی، نحوه تجزیه‌وتحلیل نمونه‌های بدافزار را در یک محیط بسته با استفاده از تکنیک‌های تجزیه‌وتحلیل بدافزارهای استاتیک یا پویا، به شما معرفی می‌کند. اطلاعات موجود در این کتابچه راهنما بر روی بنیان مهندسی معکوس از دیدگاه بدافزار تمرکز دارد و جزئیات بی‌ربط حذف شده‌اند. برای دسترسی به منابع معرفی شده در این کتابچه راهنما، می‌توانید با جستجوی ساده در اینترنت اقدام کنید.

در این کتابچه راهنما، هیچ کار نوآورانه‌ای ارائه نشده است، زیرا می‌توان آن را به‌عنوان گام‌های اولیه در تحقیقات بدافزار در نظر گرفت. خواننده با ابزارهای متداول و منبع بازی که توسط محققان در سراسر جهان برای تجزیه‌وتحلیل بدافزار استفاده می‌شود، آشنا خواهد شد. یادداشت‌ها و بهترین روش‌ها نیز شامل شده است. با استفاده از تکنیک‌ها و ابزارهای ارائه شده در اینجا، یک تحلیل‌گر می‌تواند قوانین YARA را ایجاد کند که در طول تحقیق به شناسایی تهدیدات یا قربانیان دیگر کمک می‌کند.

۱. چرا تجزیه‌وتحلیل بدافزار انجام می‌شود؟

تجزیه‌وتحلیل بدافزار عبارت اســت از "مطالعه یا فرایند تعیین عملکرد، منشــأ و تأثیر بالقوه یک نمونه بدافزار معین".

تجزیه‌وتحلیل بدافزار به واکنش به یک حادثه، با جمع‌آوری اطلاعات در مورد دقیقاً چه اتفاقی برای کدام فایل‌ها و دستگاه‌ها رخ داده است، انجام می‌شود. تحلیل‌گر باید بفهمد که یک باینری بدافزار خاص چه‌کارهایی را انجام می‌دهد و چگونه می‌توان آن را در سیستم‌ها و شبکه شناسایی کرد، خسارات وارده را ارزیابی کرد، فایل‌هایی که سعی در خروج از سیستم داشته‌اند را شناسایی کرد، روش عملیاتی آن را شناسایی کرد و بسیاری از موارد دیگر.

تعیین نوع بدافزاری که در حال تحلیل است، باعث آسان‌شدن کشف کاری است که بدافزار باتوجه‌به اثرات متداول هر نوع بدافزار انجام می‌دهد. بیشتر بدافزارها می‌توانند با این دسته‌بندی‌ها شناسایی شوند:

Backdoor: روش یا کدی در کامپیوتر هدف اســت که به مهاجم اجازه دســترســی بدون احراز هویت معتبر را می‌دهد.

Botnet: گروهی از کامپیوترها که به همان روشی که در backdoor استفاده می‌شود، آلوده شده‌اند و دستورات را از یک سرور C2 واحد دریافت می‌کنند.

Ransomware: نوعی بدافزار اســت که اطلاعات را در یک سیســتم رمزگذاری می‌کند و دســترســی کاربر را غیرفعال می‌کند. مهاجمان برای کلید رمزگشایی پولی می‌خواهند بدون تضمین ارائه کلید صحیح.

Downloader/Launcher: نرم‌افزاری است که کد مخرب دیگری را دانلود یا اجرا می‌کند.

Information stealing malware/Spyware: اطلاعات را بدون دانســتن کاربر با ثبت کلیدهای صــفحه‌کلید، عکس‌های صفحه‌نمایش و غیره جمع‌آوری می‌کند.

Rootkits: برنامه‌هایی هستند که وجود فایل‌های مخرب، برنامه‌ها، اتصالات شبکه و غیره را پنهان می‌کنند.

Scareware: نوعی بدافزار اســت که کاربر را متقاعد می‌کند که نرم‌افزار امنیتی جعلی را خریداری کند که در واقع فقط Scareware را حذف می‌کند.

Worms and Viruses: کدهای مخربی هســتند که از طریق برنامه‌ها و شــبکه‌ها خود را کپی می‌کنند و کامپیوترهای بیشتری را آلوده می‌کنند.

Fileless malware: تکنیکی است که بر اساس حافظه بدافزاری است که از فایل‌های موجود برای دانلود فایل‌های اجرایی در سیستم استفاده می‌کند. این تکنیک به طور مستقیم از فایل‌ها یا سیستم فایل استفاده نمی‌کند. به‌جای آن، از حافظه یا یک شیء دیگر سیستم‌عامل (API ها، crontabs، کلیدهای ثبت) استفاده می‌کند.

Hybrid malware: ترکیبی از عملیات‌های مختلف بدافزار است، مانند گسترش و فعالیت با هم به‌عنوان‌مثال، تروجان‌ها و ransomware.

Advanced Persistent Threats (APT): معمولاً یک گروه ملی یا حمله‌کننده حمایت شده توسط دولت، با روش‌های پیشرفته ویژه برای هدف خاصی حمله می‌کند.

این لیست می‌تواند با نوع‌های بدافزاری خاص‌تری گسترش یابد، اما این کتابچه راهنما بر روی تکنیک‌های عمومی و رایج‌ترین نوع بدافزارها برای سیستم‌عامل ویندوز تمرکز دارد.

تعریف طبق ویکی‌پدیا:

https://en.wikipedia.org/wiki/Malware_analysis

۲. نحوه راه‌اندازی محیط آزمایشگاه

تنظیم یک محیط امن، امکان کاهش خطرات آشکار در سیستم‌ها از طریق تحلیل بدافزار را فراهم می‌کند. ماشین‌های مجازی و شبکه‌های مجازی، این تنظیم را راحت‌تر، سریع‌تر و ایمن‌تر می‌کنند. برخی از پلتفرم‌های مجازی‌سازی موجود در بازار شامل VirtualBox، Parallels، Microsoft Virtual PC، VMware، Microsoft Hyper-V و Xen هستند. ما چند نمونه را با استفاده از Oracle VM VirtualBox، یک هایپروایزر میزبان رایگان و منبع‌باز توسعه‌یافته توسط شرکت Oracle Corporation، که در زمان نوشتن این متن می‌توان از این لینک دانلود کرد: https://www.virtualbox.org/wiki/Downloads، نشان می‌دهیم. تنظیمات شبکه برای هر محیط شبیه‌سازی شده را می‌توان به‌راحتی در VirtualBox با هفت نوع اتصال شبکه انجام داد: Not Attached: در این حالت، یک آداپتور مجازی در یک ماشین مجازی نصب شده است، اما اتصال شبکه وجود ندارد، به‌طوری‌که مانند قطع کابل اترنت عمل می‌کند.

NAT: این حالت به ماشین مجازی اجازه می‌دهد تا به اینترنت متصل شود، اما به دیگر ماشین‌های مجازی اجازه ارتباط نمی‌دهد.

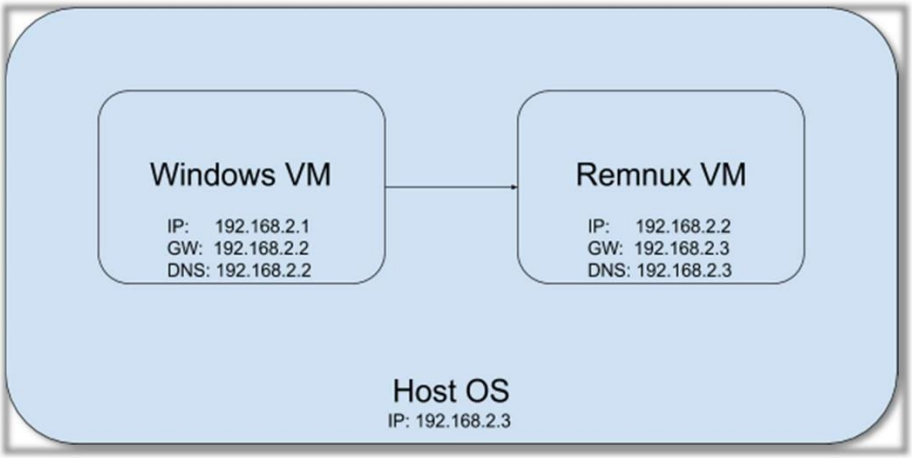
NAT Network: حالت NAT Network بسیار شبیه به حالت NAT است و برای مهمان‌های داخل شبکه NAT ارتباط فراهم می‌کند.

Bridged: حالت Bridged برای اتصال آداپتور مجازی یک ماشین مجازی به سیستم میزبان شبکه فیزیکی استفاده می‌شود.

Internal: این حالت به ماشین‌های مجازی اجازه می‌دهد تا در یک شبکه از همدیگر استفاده کنند، اما از این شبکه عایق شده‌اند و نمی‌توانند به سیستم میزبان دسترسی داشته باشند.

Host-only: این حالت یک شبکه NAT بین ماشین میزبان و ماشین‌های مجازی فراهم می‌کند.

Generic Driver: این حالت شبکه به شما امکان می‌دهد تا رابط شبکه عمومی دستگاه میزبان خود را با ماشین مجازی به اشتراک بگذارید. دو زیر حالت برای حالت Generic Driver VirtualBox موجود است. شما می‌توانید یک تونل UDP برای اتصال ماشین‌های مجازی خود به هم ایجاد کنید یا ماشین مجازی خود را به یک شبکه سوئیچ VDE (Virtual Distributed Ethernet) در لینوکس یا FreeBSD متصل کنید.



شکل ۱: نمونه راه‌اندازی آزمایشگاه بدافزار

در این محیط، یک ماشین مجازی ویندوزی قربانی برای اجرای بدافزار نصب شده است و یک ماشین مجازی Remnux برای شبیه‌سازی اینترنت (با استفاده از Inetsim که در بخش ۵.۲.۴ توضیح داده شده است) و تجزیه‌وتحلیل رفتار بدافزار استفاده می‌شود. ازآنجاکه ما از یک اینترنت شبیه‌سازی شده استفاده خواهیم کرد، بدافزار باید از اینترنت واقعی جدا شود. حالت شبکه Host-only به ما این امکان را می‌دهد که هدف را بدون دسترسی به ماشین میزبان یا دیگر ماشین‌ها در شبکه فیزیکی جدا کنیم. این نیاز با استفاده از دروازه‌های پیش‌فرض و تنظیمات شبکه جداگانه در ماشین میزبان برآورده خواهد شد. گزینه Host-only یک رابط شبکه مجازی مشابه رابط حلقه برگشتی در ماشین میزبان ایجاد می‌کند. IP این رابط باید به‌صورت استاتیک پیکربندی شود و با شبکه فیزیکی متفاوت باشد. علاوه بر این، IP ماشین‌های مجازی باید به‌صورت استاتیک پیکربندی

شـوند درحالی‌که دروازه پیش‌فرض ماشـین قربانی به ماشـین Remnux اشـاره دارد و دروازه پیش‌فرض ماشـین Remnux به ماشـین میزبان اشـاره دارد. IP DNS در ماشـین قربانی باید به ماشـین مجازی Remnux پیکربندی شود تا پرس‌وجوهای DNS در Inetsim در حال اجرا در Remnux پایان یابد.

Snapshotting

اسـنپ‌شـات یک تصـویر از دیسـک و حافظه در یک‌لحظه خاص اسـت. با تجزیه‌وتحلیل یک حافظه دامپ با استفاده از ابزارهای فارنزیک، می‌توانید نمای کلی‌تری از نمونه‌ای که در حال بررسی است را به دست آورید. با اسـتفاده از ابزارهایی مانند Volatility یا Rekall، می‌توانید نمونه بدافزار را اسـتخراج کنید، اتصـالات را ببینید و غیره.

توجه: در زمان نگارش، Volatility و Rekall را می‌توان از لینک‌های زیر دانلود کرد:

https://www.volatilityfoundation.org/

https://github.com/google/rekall

- اسـنپ‌شـات گیری یک ویژگی حیاتی برای تجزیه‌وتحلیل بدافزار اسـت. محیط مجازی برای بدافزار می‌تواند پس از اجرای بدافزار یا تغییر پارامتر سـیسـتم به‌راحتی بازیابی شـود. عملکردهای ضـروری عبارت‌اند از:

- بازیابی اسنپ‌شات: دور ریزی تغییرات و استفاده از تصویر ماشین پیش‌شناور.

- حذف اسـنپ‌شـات: ادغام اسـنپ‌شـات ضـبط شـده با حالت کنونی. پس از حذف، نمی‌توانید به تصـویر پیش‌شناور قبلی بازگردید.

- کلون اسنپ‌شات: اسنپ‌شات انتخاب شده را به یک ماشین مجازی جدید "fork" کنید.

۲. محافظت از خود در برابر بدافزار:

با وجود راحتی‌هایی که محیط‌های مجازی فراهم می‌کنند، بدافزارهای جدیدتر سعی می‌کنند تشخیص دهند که آیا در یک محیط مجازی تحلیل می‌شـوند و رفتار خود را پنهان می‌کنند. پارامترهایی که بیشـترین بررسـی را توسـط بدافزارها انجام می‌دهند، شـامل کلیدهای رجیسـتری، سـاختارهای حافظه، کانال‌های ارتباطی، فایل‌ها و سـرویس‌های خاص، آدرس MAC و برخی ویژگی‌های سـخت‌افزاری هسـتند. برخی از مثال‌های این پارامترها برای VirtualBox عبارت‌اند از:

- کلید رجیستری:

- Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Oracle\VirtualBox Guest Additions
- Computer\HKEY_LOCAL_MACHINE\HARDWARE\ACPI\DSDT\VBOX

- پردازش:

- VboxService.exe
- VboxTray.exe

- فایل‌ها:

- C:\Windows\System32\drivers\VBoxMouse.sys
- C:\Windows\System32\drivers\VBoxVideo.sys

- شروع آدرس فیزیکی برابر با 08:00:27 باشد.
- بررسی دستورالعمل CPUID :
  - اجرای این دستور با EAX=0x40000000، رشته شناسایی سازنده CPU در EBX، EDX و ECX برگردانده می‌شود، مانند "GenuineIntel" یا "AuthenticAMD" اما برای VirtualBox، "vboxvboxvbox" برگردانده می‌شود.
  - همچنین، اجرای با EAX=1، بیت ۳۱ از ECX را در یک ماشین مجازی به ۱ تغییر می‌دهد.

یکی از مثال‌های معروف بدافزار در جهان برای بررسی نام CPU، "GootKit" است که همچنین رجیستری، دیسک، BIOS و آدرس MAC را بررسی می‌کند. مثال‌های دیگر شامل "Locky"، "Heodo" یا "Kovter" هستند که انتظار تعامل کاربر را دارند و "QakBot Trojan" که قبل از اجرا کردن چندین ثانیه منتظر می‌ماند.

برای رفع این مشکلات، برخی از مقادیر (آدرس MAC، مقادیر رجیستری، فایل‌های پیکربندی و غیره) می‌توانند به‌صورت دستی تغییر داده شوند؛ فراخوانی‌های API از بدافزار می‌توانند متوقف شوند؛ و خروجی‌های سفارشی به بدافزار ارائه داده شوند تا مکانیزم‌های خودحفاظتی بدافزار متعادل شوند.

۳- آنالیز ایستا بدافزار

۳-۱ توضیحات

تجزیه‌وتحلیل بدافزار به معنای تجزیه‌وتحلیل فایل‌های اجرایی قابل‌حمل (PE) بدون اجرای آن‌ها است. این تجزیه‌وتحلیل در ابتدا با تجزیه ساختار هدر PE انجام می‌شود که شامل اطلاعات مفیدی است که به سیستم‌عامل کمک می‌کند تا فایل را بارگذاری و اجرا کند) مانند سیستم‌های پشتیبانی شده، چیدمان حافظه، ارجاعات کتابخانه‌های پویا برای پیوند، جداول صادرات و ورودی API، داده‌های مدیریت منابع و داده‌های ذخیره‌سازی محلی نخ .(تجزیه و تحلیل استاتیک اولیه می‌تواند با ارائه اطلاعات در مورد عملکرد، گواهینامه‌ها، واردات، تاریخ کامپایل و غیره تأیید کند که یک فایل مخرب است. بر اساس این اطلاعات، تحلیلگر می‌تواند IoC را

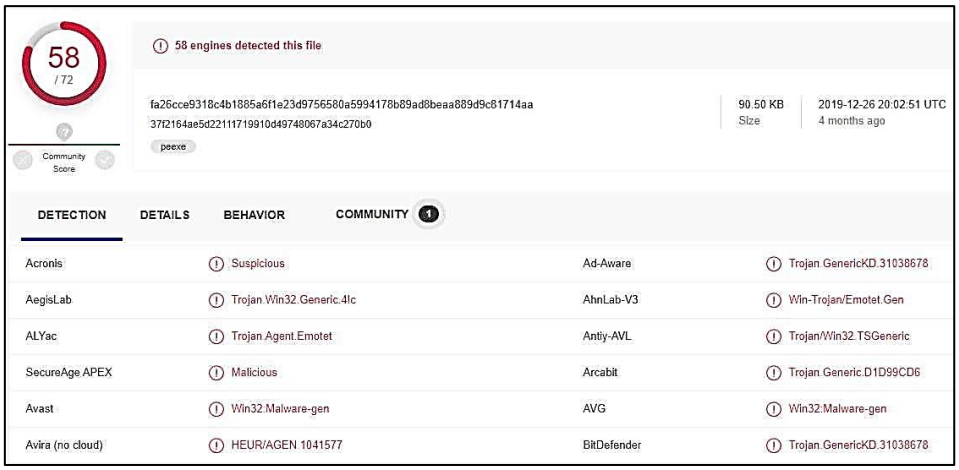
ایجاد کند و از آن برای تحقیقات بیشتر استفاده کند. این تجزیه و تحلیل در مقابل نمونه‌های پیچیده، نسبت به تجزیه و تحلیل استاتیک پیشرفته، که شامل تجزیه و تحلیل کد مخرب درون یک Disassmbeler و بررسی دستورالعمل‌ها است، ناکارآمد است.

در بخش بعدی، ابزارها و تکنیک‌های مختلفی که برای انجام تجزیه و تحلیل بدافزار استفاده می‌شوند، معرفی خواهند شد.

۳-۲ تکنیکها و ابزارهای آنالیز ایستا بدافزار

۳-۲-۱ ابزار VirusTotal

با آپلود کردن یک فایل به VirusTotal و مقایسه آن با لیستی از تشخیص‌های مختلف آنتی‌ویروس، تحلیل‌گر می‌تواند متوجه شود که نمونه مورد بررسی مخرب است یا خیر. این فرآیند همچنین اطلاعاتی را در مورد فایل، مانندSHA256 ، MD5، اندازه فایل، اطلاعات امضا، جزئیات بخش‌ها، واردات و غیره، فراهم می‌کند.



شکل ۲: صفحه وب VIRUSTOTAL

اگر امکان آپلود نمونه برای VirusTotal وجود ندارد، این پلتفرم گزینه جستجوی نمونه‌ای که قبلاً در وب سایت آپلود شده است را با جستجوی مقدار هش نمونه شما فراهم می‌کند.

با این حال، باید به این نکته توجه داشت که استفاده از این ابزار باید با احتیاط انجام شود، زیرا آپلود کردن یک نمونه بدافزار حاوی اطلاعات حساس درباره شرکت شما به VirusTotal می‌تواند باعث ایجاد مشکلات امنیتی برای شرکت شود. در صورت انتشار اطلاعات، افراد سوم ممکن است با استفاده از تابع جستجوی موجود در وب سایت، آن‌ها را پیدا کرده و بهره‌برداری کنند.

۳-۲-۲ ابزار String analysis

تجزیه و تحلیل رشته‌ها، فرایندی است که در آن کاراکترهای خوانا و اسکی و یونیکد از باینری استخراج می‌شوند. همه رشته‌های یافت شده توسط برنامه مورد استفاده نیستند؛ حمله‌کنندگان ممکن است رشته‌های جعلی را نیز درج کنند تا تحقیقات را مختل کنند.

ابزار های استفاده شده برای آنالیز رشته

ابزار Strings2 یک ابزار خط فرمانی است که برای استخراج رشته‌ها از داده‌های باینری استفاده می‌شود. این برنامه نسخه بهبود یافته‌ای از روش رشته‌های Sysinternals کلاسیک است و همچنین می‌تواند رشته‌ها را از فضای آدرس فرآیند استخراج کند.

در زمان نوشتن این متن، Strings2 می‌توانست از لینک زیر دریافت شود:

https://github.com/glmcdona/strings2

همچنین ابزار Flare-Floss یک ابزار حل رشته‌های رمزگذاری شده است که تکنیک‌های مختلف را ترکیب و خودکارسازی می‌کند تا رمزگشایی رشته‌ها را انجام دهد. در زمان نوشتن این متن، ابزار Floss می‌توانست از لینک زیر دریافت شود:

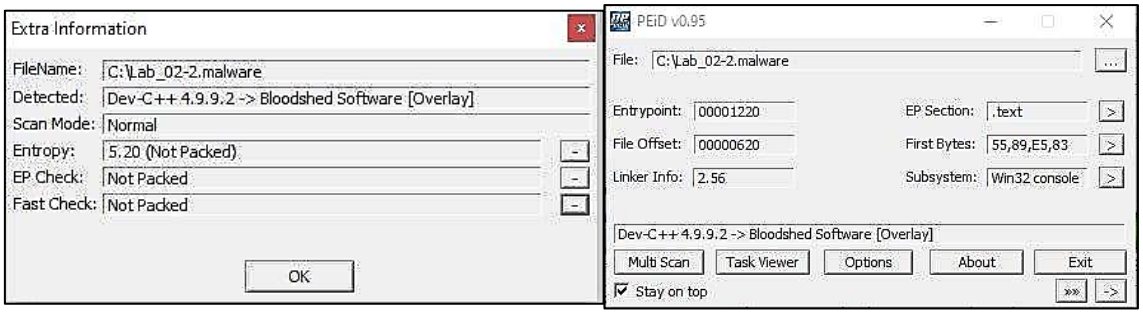
https://github.com/mandiant/flare-floss

همچنین در این متن به این نکته اشاره شده است که رشته‌ها در فرمت‌های ASCII و Unicode هستند و برخی از ابزارها برای استخراج هر دو فرمت رشته نیاز به تعیین نوع رشته دارند.

رشته‌ها به صورت فرمت‌های ASCII و یونیکد هستند (برخی ابزارها نوع رشته مورد نظر برای استخراج در طول تجزیه و تحلیل باید مشخص شود، زیرا برخی ابزارها هر دو فرمت رشته را استخراج نمی‌کنند).

۳-۲-۳ ابزار PEiD Tool

ابزار PEiD برای تجزیه و تحلیل سربرگ PE به کار می‌رود تا به تحلیل‌گر جزئیات بیشتری درباره کریپتورها، بسته‌بندی‌ها و کامپایلرهای موجود در فایل‌های اجرایی بدهد .PEiD با استفاده از امضای‌های استاتیکی که در برنامه ذخیره شده‌اند، این شناسایی را انجام می‌دهد. مثالی که در زیر آورده شده، نتیجه تجزیه و تحلیل با استفاده از ابزار PEiD را نشان می‌دهد. در این مورد، نمونه تحلیل شده بسته‌بندی نشده است و مقدار بی نظمی کم است. ابزار PEiD می‌تواند بیش از ۵۰۰ تعریف امضایی را که از یک فایل پیکربندی به نام "userdb" بارگیری می‌شوند، شناسایی کند.



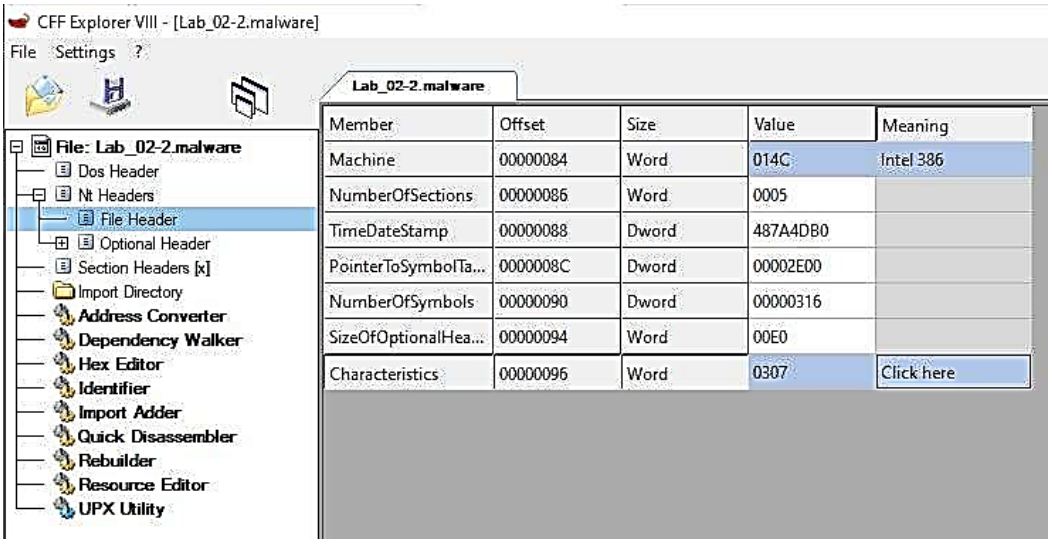
شکل ۳: اسکن نمونه PEID

در زمان نگارش، این ابزار را می توان از لینک زیر دانلود کرد:

https://www.softpedia.com/get/Programming/Packers-Crypters-Protectors/PEiD-updated.shtml

۳-۲-۴ ابزار CFF Explorer

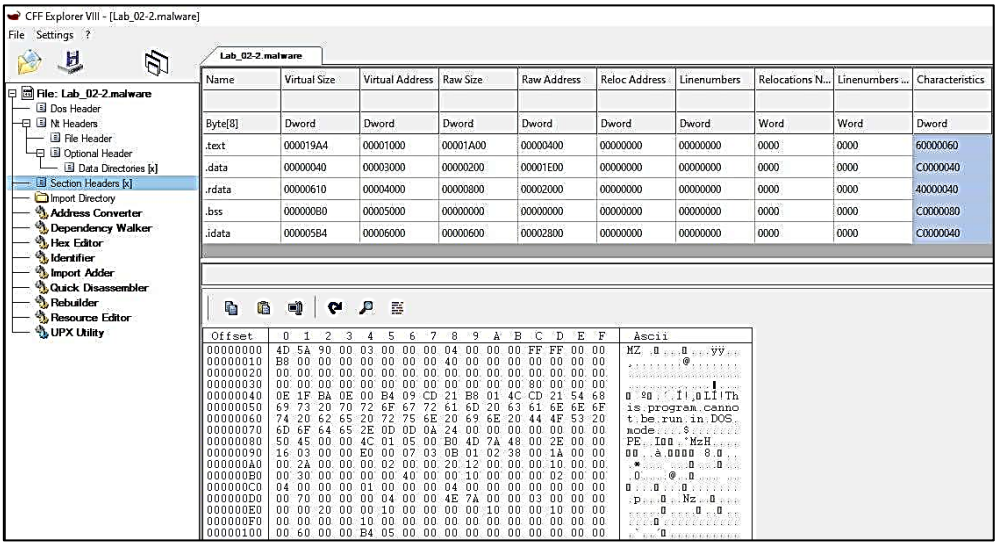
ابزار CFF Explorer یکی از ابزارهای معمول برای انجام تغییرات در داخل سربرگ PE است. این ابزار بر روی سیستم‌عامل ویندوز اجرا می‌شود و قابلیت لیست کردن فرآیندها یا دامپ کردن فرآیند به یک فایل را داراست. با استفاده از این ابزار، تحلیل گر می‌تواند تاریخ کامپایل و نوع معماری نمونه بدافزار تحلیل شده را بر اساس اطلاعات موجود در سربرگ PE استخراج کند. تاریخ کامپایل با استفاده از زمان Epoch Unix Time در بخش "TimeDateStamp" نمایش داده می‌شود. در این مورد، تاریخ "GMT Sunday, July 13, 2008, 6:47:12 PM" است.



شکل ۴: بررسی تاریخ کامپایل در CFF EXPLORER

توجه: اطلاعات مربوط به تاریخ کامپایل نمونه استخراج شده از سربرگ PE، می‌تواند به تحلیل‌گر در پاسخ به سوالات مربوط به رسیدگی به حوادث کمک کند.

با تجزیه و تحلیل بخش سربرگ، تحلیل‌گر می‌تواند تشخیص دهد که آیا بدافزار بسته‌بندی شده است یا خیر. بسته‌بندی‌ها معمولاً نام بخش‌ها را از نام‌های معمولی (text.، data.، rsrc.و غیره) به نام‌های دیگری مانند UPX1 تغییر می‌دهند. در مثال زیر، نمونه بسته‌بندی نشده است.



شکل ۵: بخش هدرها در CFF EXPLORER

ابزار CFF Explorer شامل ویژگی‌های زیر است:

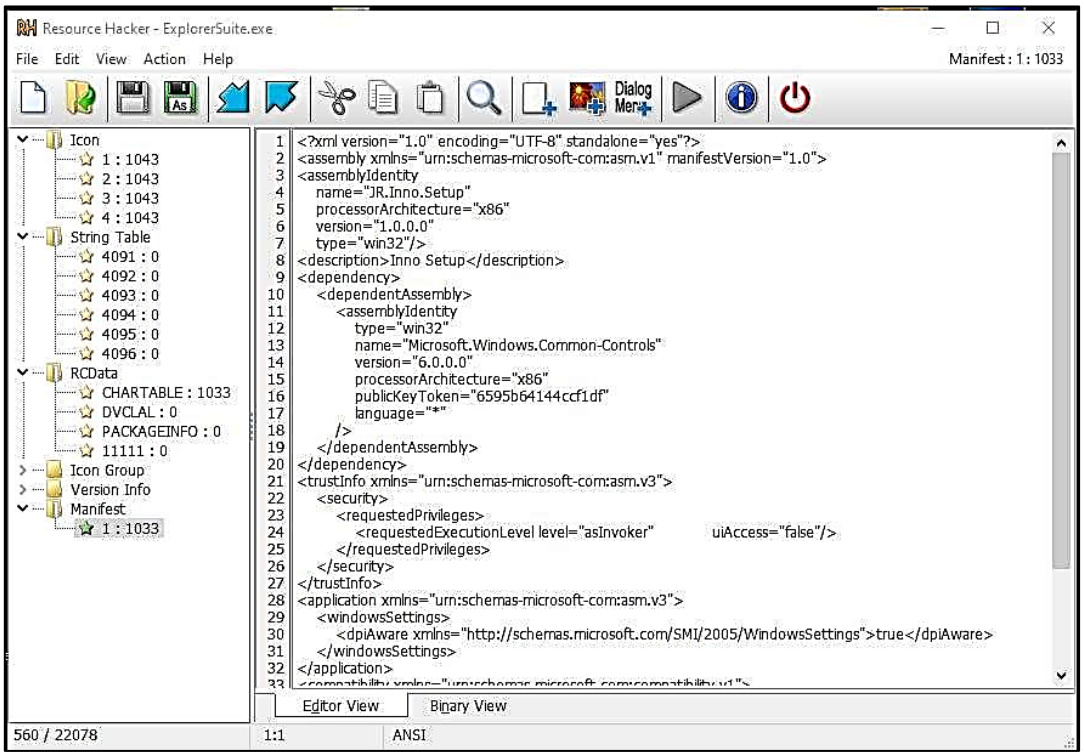
- نمایش دهنده فرآیندها
- ویرایشگر هگزادسیمال
- نمایش دهنده درایورها
- دامپ کردن حافظه و فایل‌های PE
- بررسی اعتبار فایل‌های PE و حافظه
- ویژگی‌های دیگر.

در زمان تهیه کتاب از لینک زیر قابل دانلود است:

https://ntcore.com/?page_id=388

۳-۲-۵ ابزار Resource Hacker

ابزار Resource Hacker یک برنامه رایگان است که برای استخراج، ویرایش و اضافه کردن منابع (تصاویر، دیالوگ‌ها، منوها و غیره) از فایل‌های اجرایی ویندوز استفاده می‌شود. این ابزار قابلیت ویرایش فایل‌های OCX، EXE و DLL را دارد و می‌توانید بدون نیاز به ویرایشگر حرفه‌ای منبع فایل، تغییرات دلخواه خود را ایجاد کنید. این ابزار در ویندوز اجرا می‌شود و می‌تواند منابع را به صورت عکس و یا متن دیکد کند.



شکل ۶: هک منبع – منابع باینری (ICON، MANIFEST)

استفاده از ابزار Resource Hacker می‌تواند در تجزیه و تحلیل نمونه‌های dropper که یک فایل PE اضافی در منابع آن‌ها وجود دارد، به کار گرفته شود. این ابزار همچنین می‌تواند از خط فرمان دسترسی پیدا کند بدون اینکه نیاز به باز کردن رابط کاربری Resource Hacker باشد.

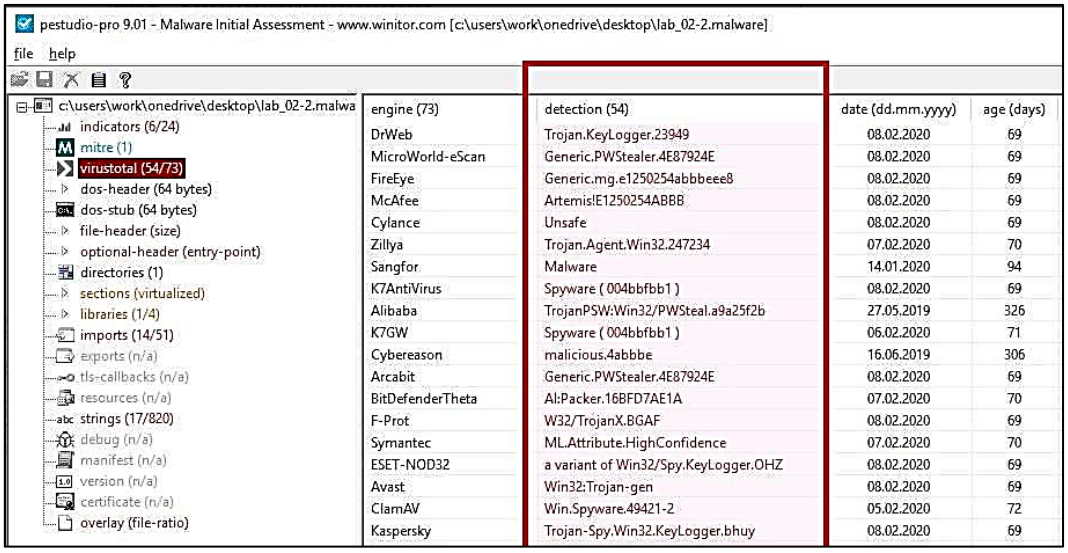
در زمان تهیه کتاب از لینک زیر قابل دانلود است:

http://www.angusj.com/resourcehacker/

۳-۲-۶ ابزار PeStudio

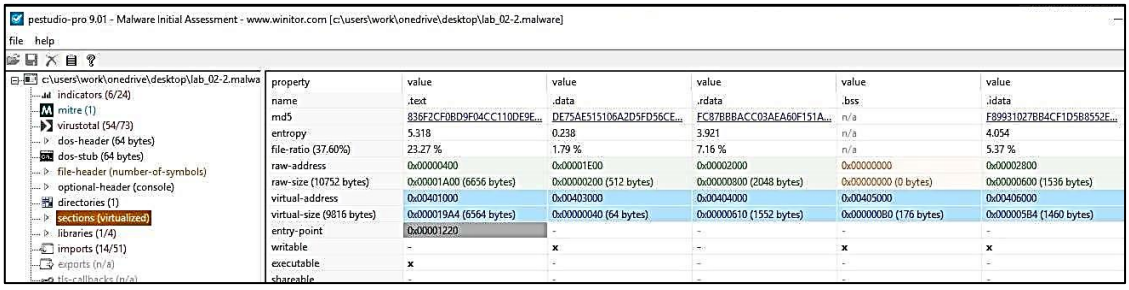
ابزار PeStudio یک ابزار است که برای یافتن نشانه‌های مشکوک در فایل‌های اجرایی استفاده می‌شود تا ارزیابی اولیه بدافزار را شتاب دهد. با استفاده از این ابزار، تحلیل گر می‌تواند به راحتی عملکردهایی که توسط

سازندگان بدافزار برای فعالیت‌های مخرب معمولاً استفاده می‌شوند را شناسایی کند. هنگامی که تحلیل گر نمونه بدافزار را در داخل برنامه باز می کند، اطلاعات کلی مربوط به فایل مانند هش MD5 و بی نظمی به دست می‌آید. سپس مقدار هش نمونه در VirusTotal بررسی می‌شود و نتیجه جستجو در داخل برنامه لیست می‌شود. تصویر زیر نتیجه پرس‌وجو را نشان می‌دهد:



شکل ۷: بررسی VIRUSTOTAL در PESTUDIO

با تحلیل بخش‌های فایل، تحلیل گر می‌تواند هش MD5 برای هر بخش، مقدار بی نظمی و آدرس نقطه شروع (entry-point address)(آدرسی که اجرای فرآیند از آن شروع می‌شود) و همچنین دسترسی خواندن، نوشتن و / یا اجرا برای هر بخش را مشاهده کند. اگر بخش ".rsrc" بیش از حد بزرگ باشد، برنامه می‌تواند یک فایل دیگر را در دیسک "drop" کند. در این مورد، توصیه می‌شود که در زمان تجزیه و تحلیل در حال اجرا، تحلیل گر به فایل‌هایی که بر روی دیسک نوشته شده‌اند، توجه ویژه داشته باشد.



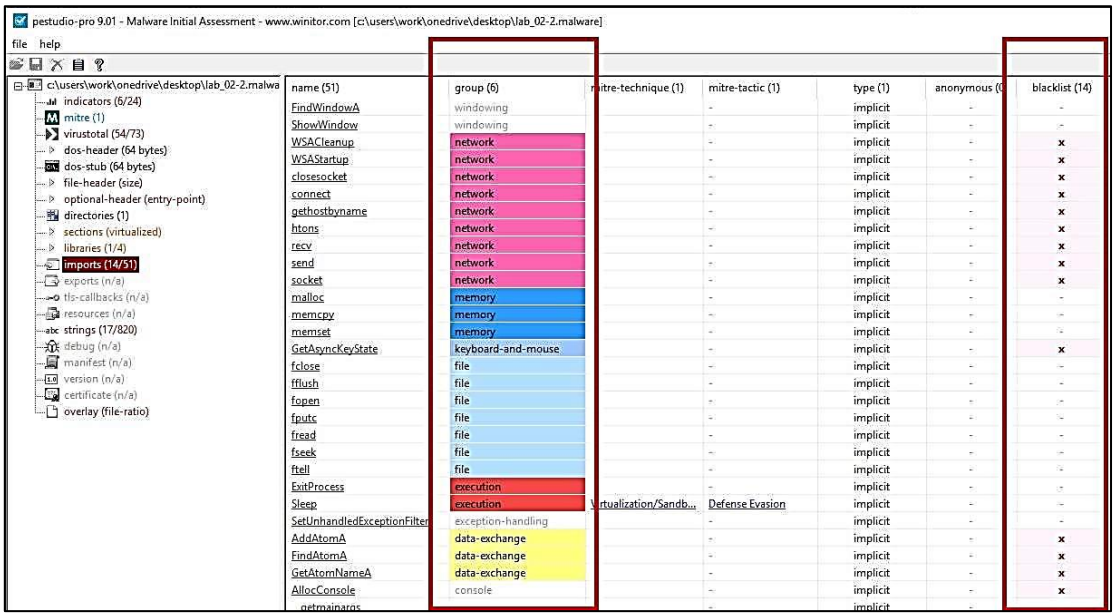
شکل ۸: بخش های هدر در PESTUDIO

با تحلیل بخش‌های فایل، تحلیل گر می‌تواند نام‌های توابع وارد شده را در بخش "Import sections" مشاهده کند. با جستجوی هر تابع در MSDN.microsoft.com ، تحلیل گر می‌تواند شناسایی کند که آن تابع چه کاری

انجام می‌دهد PeStudio .دارای لیستی از واردات"blacklisted" است، که در آن تمام وارداتی که می‌توانند برای فعالیت‌های مخرب استفاده شوند، لیست شده‌اند.

با اســتفاده از ابزارPeStudio، تحلیل گر می‌تواند با بررســی بخش"Imports"، یک نمای کلی از کتابخانه‌های وارد شده اصلی که توسط بدافزار برای فعالیت‌های مخرب استفاده شده‌اند و توسط برنامه PeStudio به لیست سیاه اضــافه شــده‌اند، را بدســت آورد. به عنوان مثال، واردات"connect" ، "gethostbyname"، "socket"، "memcpy"، "send"و "GetAsyncKeyState" به تحلیل گر بدافزار، برخی از عملکردهای اصــلی نمونه تحلیل شده را نشان می‌دهند.

در بخش"Exports" ، توابعی که فایل PE برای اســتفاده در فایل‌های PE دیگر صــادر می‌کند، نمایش داده می‌شوند. در مثال ارائه شده، هیچ صادراتی وجود ندارد.



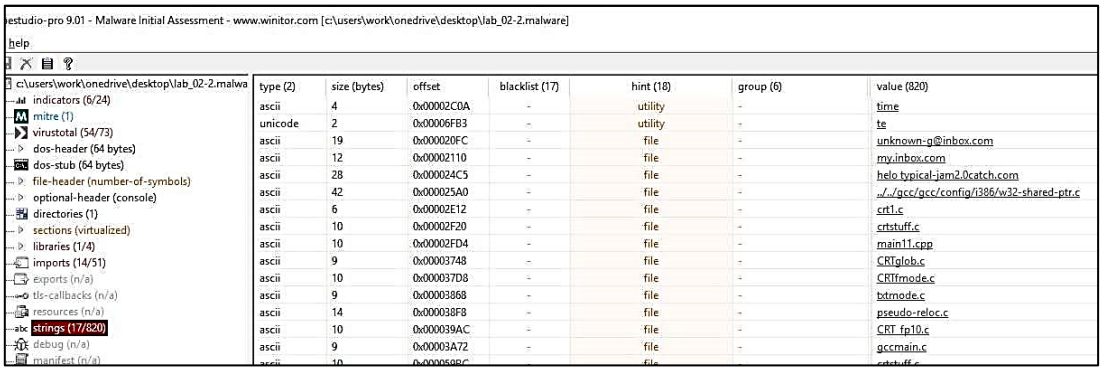
شکل ۹: بخش Imports در PESTUDIO

بخش "Resources" به طور معمول اطلاعات رابط کاربری (آیکون‌ها یا عناصــر پنجره ســفارشــی) را ذخیره می‌کند. اگر برنامه مخرب قابلیت‌های dropper را داشــته باشــد، فایل‌هایی که بر روی دیســک نوشــته شــده‌اند، ممکن است در بخش ".rsrc" ذخیره شوند.

بخش "tls-callback" شــامل کدی اســت که محیط را برای اجرای برنامه تنظیم می‌کند. این کد قبل از نقطه شروع اجرا خواهد شد. با استفاده از این قابلیت، سازنده بدافزار می‌تواند کد را در داخل TLS (Thread Local Storage)پنهان کند که قبل از ایجاد فرآیند توسط سیستم عامل ویندوز، اجرا خواهد شد.

بخش "Strings" نیز یک منبع مفید از اطلاعات برای تحلیل گر است. تمام رشته‌های اجرایی تجزیه و تحلیل شده و در این بخش قرار داده می‌شوند. با بررسی بخش "Strings "، تحلیل گر سعی در شناسایی رشته‌های خوانا مانند آدرس‌های IP و URL و نام فایل‌هایی که در طول تحقیق ممکن است استفاده شوند، دارد. هنگامی که تعداد کاراکترهای خوانا کاهش می‌یابد، برنامه ممکن است بسته‌بندی شده یا مبهم باشد. بخش "رشته‌ها" نمونه تحلیل شده در زیر نشان داده شده است:

Dropper یک نام عمومی برای تروجان‌هایی است که مصنوعات اضافی را روی سیستم آسیب‌دیده رها می‌کنند.



شکل ۱۰: بخش رشته در PESTUDIO

بخش "Certificates" نیز یک قسمت مهم در تجزیه و تحلیل بدافزار است که شامل گواهی استفاده شده برای امضای برنامه می‌باشد. به طور معمول، برنامه‌های مخرب امضا نمی‌شوند یا از گواهیی از یک مرجع گواهی‌دهی استفاده می‌کنند که غیرقابل اعتماد است یا منبع آن گواهی لو رفته است.

ابزار PeStudio همچنین می‌تواند یک گزارش XML برای فایل اجرایی مورد تجزیه و تحلیل ایجاد و صادر کند. گزارش خروجی XML می‌تواند برای تجزیه و تحلیل بیشتر توسط ابزارهای تحلیل شخص ثالث استفاده شود.

در زمان نوشتن کتاب از آدرس زیر قابل دانلود است:

https://www.winitor.com

۴-جداسازها یا Disassembly مانند (IDA & Ghidra)

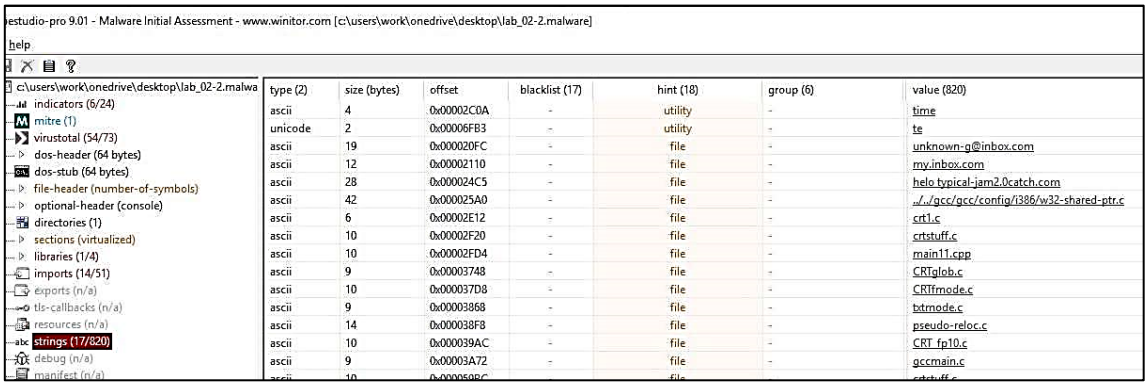
Disassembler یک ابزار بسیار مفید برای بررسی یک فایل اجرایی کامپایل شده و ارائه یک درک کلی از آن است. فایل‌های اجرایی شامل یک کد ماشین در قالب داده‌های دودویی هستند Disassemblerها کد ماشین را به زبان ماشین قابل استفاده‌تر ترجمه می‌کنند.

IDA free ۱-۴

Disassembler IDA6 یک ابزار "استاندارد" است که توسط پژوهشگران بدافزار و مهندسان معکوس استفاده می‌شود. این کتابچه راهنمای فقط بر روی نسخه رایگان IDA (برای استفاده تجاری نیست) تمرکز دارد.

با استفاده از IDA برای تجزیه و تحلیل بدافزار به عنوان یک disassembler (باز کردن فایل‌ها، تجزیه و تحلیل و خواندن کد)، کاربردی برای ایجاد بدافزار در کامپیوتر کاربر ندارد. در مورد قابلیت‌های اشکال‌زدایی IDA، توصیه می‌شود تحلیل‌گر در یک آزمایشگاه جداگانه که به پردازش فایل‌های مخرب اختصاص داده شده است، کار کند تا از بدافزار ناخواسته محیط کاری کسب و کار جلوگیری شود که ممکن است به دلیل اجرای تصادفی کد مخرب در اشکال‌زدایی IDA رخ دهد. برای کسب اطلاعات بیشتر، به فصل ۲ (نحوه تنظیم محیط آزمایشگاه) مراجعه کنید.

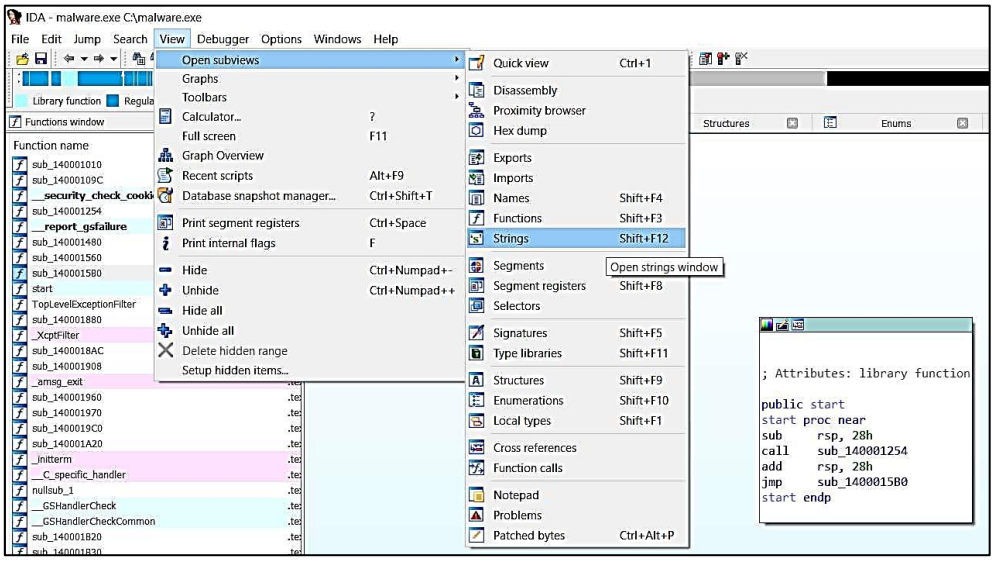
IDA به صورت متنی اساسی کد را نمایش می‌دهد (آدرس، دستور، پارامترها و توضیحات؛ به صورت ردیف به ردیف) یا به صورت نمایش نمودار، که کد ماشین را در بلوک‌های منطقی نشان می‌دهد. تقسیم بندی به بلوک‌ها بر اساس پرش‌ها، شرایط و حلقه‌ها است. روابط بین بلوک‌ها با پیکان‌ها نشان داده می‌شود. نمایش نمودار فقط برای توابع معتبر در دسترس است. نوع نمایش با فشردن کلید فضایی تغییر می‌کند.



شکل ۱۱: نمای متن IDA (در سمت چپ) و نمای نمودار (در سمت راست)

پس از باز کردن یک فایل اجرایی در IDA ، گام‌های اولیه توصیه شده برای آشنایی با ویژگی‌های اساسی فایل اجرایی شامل رشته‌ها، توابع، واردات، صادرات و نام‌ها هستند. همه این‌ها در منو "View" > "Open"

"Strings" <subviews (توابع، واردات، صادرات و نام‌ها در همان مکان قرار دارند) در صورتی که به عنوان یک تب در پنجره کاری اصلی باز نشده باشند، قابل دسترسی هستند.



شکل ۱۲: جداکننده IDA

رشته‌ها – فهرستی از نمایش‌های رشته (متن) در یک فایل اجرایی که می‌تواند در درک بهتر هدف یک فایل اجرایی مفید باشد، مانند آدرس IP، URL یا نام دامنه به فعالیت شبکه اشاره دارد.

بخش‌هایی که در یک فایل اجرایی بارگذاری شده‌اند و توسط یک فایل اجرایی استفاده می‌شوند، در بخش "واردات" قرار دارند. یک تابع API یک کد پیش‌فرض است که یک فایل اجرایی می‌تواند بدون پیاده‌سازی آن در کد خود فراخوانی کند. با توجه به لیست توابع وارد شده، می‌توان به شیوه تعامل یک فایل اجرایی با سیستم عامل و منابع آن (فایل سیستم، رجیستری، شبکه، رمزگذاری و غیره) پی برد.

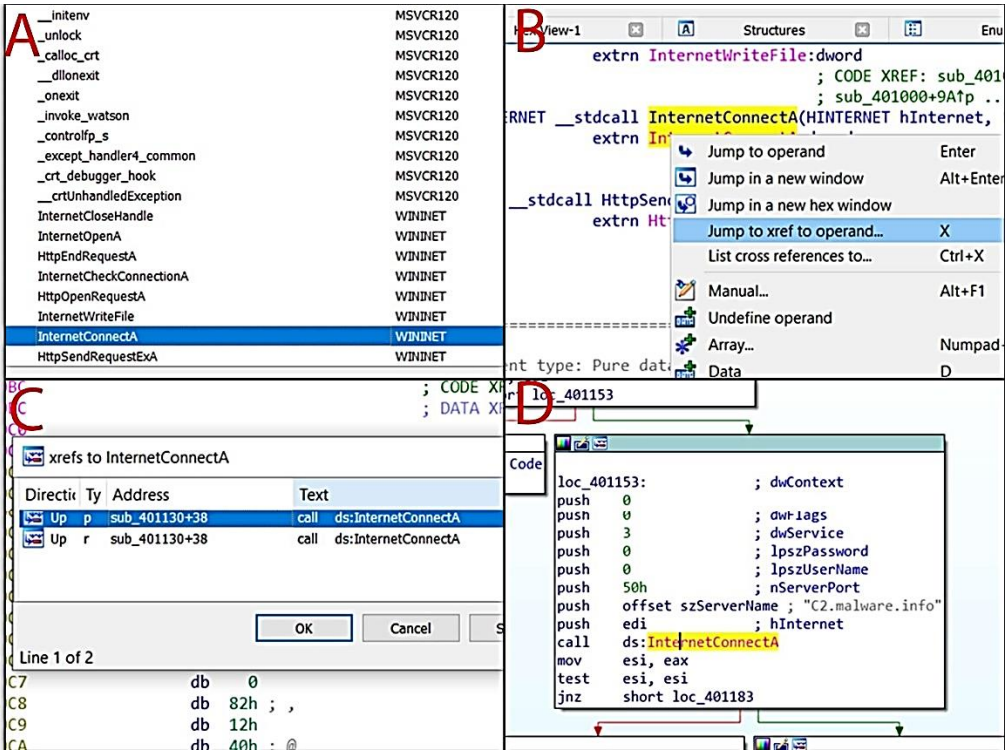
Exports: لیستی از توابعی که از یک فایل اجرایی به محیط خارجی ارائه می‌شوند. توابع صادر شده می‌توانند توسط یک برنامه خارجی فراخوانی و اجرا شوند.

Names: شامل فهرستی از تمام نام‌های موجود در یک فایل اجرایی است، از جمله نام تابع کتابخانه، تابع معمولی، دستور، رشته متنی، داده و نام وارد شده است.

Functions: فهرستی از تمام توابع موجود در کد یک فایل اجرایی. علاوه بر این، ویژگی F.L.I.R.T. (Fast Library Identification and Recognition Technology) به IDA اجازه می‌دهد تا توابع کتابخانه استاندارد تولید شده توسط کامپایلرهای پشتیبانی شده را شناسایی کرده و خوانایی و خواندنی تر شدن تجزیه و تحلیل‌های تولید شده را بهبود بخشد.

توصیه می‌شود در هنگام تجزیه و تحلیل رشته‌ها و واردات، بیشتر به شبکه، رمزگذاری و فایل سیستم تمرکز کنید. اگر موارد جالبی در لیست‌های فوق الذکر یافت شد، باید به دقت بررسی شوند. به عنوان مثال، در بررسی تابع وارد شده "InternetConnectA":

۱. روی آن دوبار کلیک کنید) یا یک بار کلیک کنید و ENTER را فشـــار دهید (تا نمای مونتاژ به آدرسی که تعریف تابع در آن ذخیره شده است، هدایت شود.
۲. نام تابع را برجسته کنید (روی آن کلیک یک بار کنید) و دکمه "x" را فشار دهید) یا روی آن راست کلیک کرده و"...Jump to xref to operand" را انتخاب کنید (تا یک جدول با فهرستی از مواردی که تابع در آن‌ها استفاده شده است، نمایش داده شود.
۳. با دو بار کلیک بر روی موارد، نمای مربوط به کد با تابع "InternetConnectA" را مشاهده می‌کنید و امکان تجزیه و تحلیل متن را فراهم می‌کند.



شکل ۱۳: کار با IDA (A – ورودی، B – نحوه دریافت مراجع متقابل، C– فهرست مراجع متقابل، D – ناحیه کد با تابع API بهره)

تابع "InternetConnectA" با استفاده از دستور "CALL" فعال می‌شود. طبق مستندات رسمی ارائه شده توسط شرکت مایکروسافت، تابع "InternetConnectA" دارای ۸ پارامتر است. پارامترهای خاص به تابع با استفاده از دستور "PUSH" اختصاص داده می‌شوند IDA. قادر است پارامترهای توابع شناخته شده را تشخیص داده و با یک نظر توضـیحی مشـخص کند که به تحلیل گران کمک می‌کند تا بهتر در کد جایگاه بگیرند و آن را درک

کنند. همانطور که در بالا مشاهده می‌شود(شکل ۲ قسمت D)، پارامترها با استفاده از دستور "PUSH" به صورت معکوس به استک منتقل می‌شوند "dwContext" - (هشتمین پارامتر تابع) به عنوان اولین پارامتر PUSH می‌شود. به طور معکوس، "hInternet" (اولین پارامتر تابع) به عنوان آخرین پارامتر PUSH می‌شود.

چگونه کد را درک کنیم؟ پارامتر "dwService" نوع سرویس را تعیین می کند: مقدار ۳ برای HTTP است؛ مقدار ۵۰ h در "nServerPort" به معنای استفاده از پورت استاندارد TCP 80 است (۵۰ به صورت شانزدهی برابر ۸۰ دهدهی) و "szServerName" شامل "C2.malware.info" است که نام دامنه سرور مقصد رمزگذاری شده است.

از آنجایی که چنین تجزیه و تحلیل کد یک فرآیند بسیار کند و زمان بر است، توصیه می شود آن را تجزیه و تحلیل نکنید.

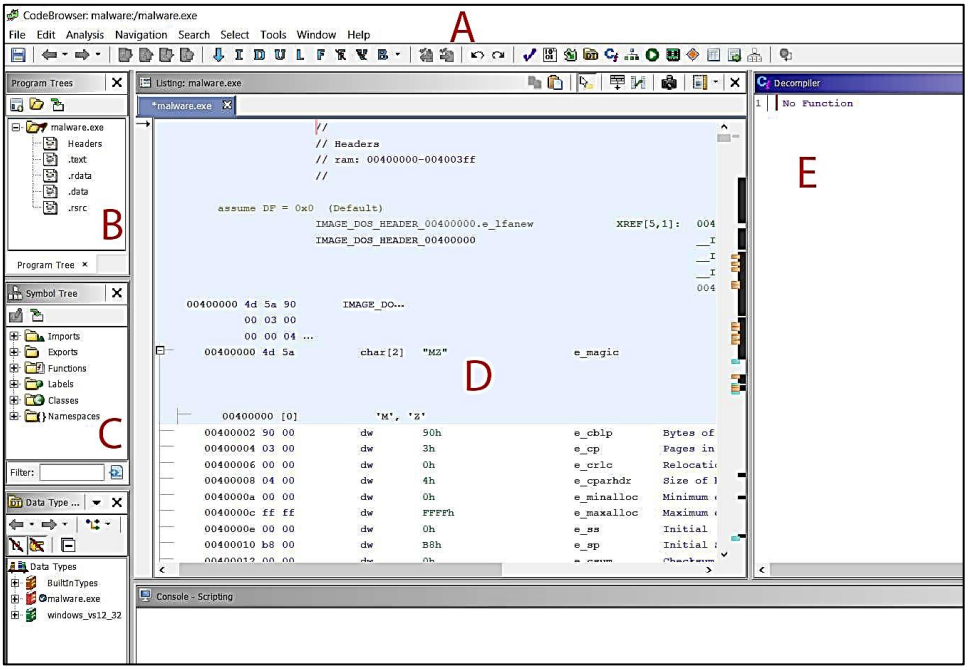
https://docs.microsoft.com/en-us/windows/win32/api/wininet/nf-wininet-internetopena

برای درک کد بهتر است که به صورت کلی، دستور به دستور، از ابتدا به آن نگاه کنید. رویکرد بهتر این است که بلوک‌های جالب کد (بر اساس رشته‌ها، واردات و توابع) را شناسایی کرده و آن‌ها را به دقت تجزیه و تحلیل کنید.

توانایی IDA به راحتی با استفاده از پلاگین‌های قابل برنامه‌ریزی گسترش می‌یابد. پلاگین‌ها می‌توانند برای اتوماسیون کارهای روتین، بهبود تجزیه و تحلیل کد مخرب یا اضافه کردن قابلیت‌های خاص به تجزیه و تحلیل‌گر ماشین باشند. پلاگین‌ها باید به زبان C++ نوشته شوند. آن‌ها می‌توانند به دکمه‌های میانبر یا موارد منو متصل شوند و دسترسی کامل به پایگاه داده IDA داشته باشند و می‌توانند برنامه را بررسی و تغییر دهند یا از توابع I/O استفاده کنند. برخی از پلاگین‌ها فقط برای کاربران ثبت نام شده با اشتراک فعال برای نسخه تجاری در دسترس هستند؛ دیگران به عنوان یک افزونه پرداختی در دسترس هستند مانند (Decompiler Hex-Rays) و همچنین پلاگین‌های منبع باز وجود دارند. یکی از پلاگین‌های پراستفاده‌ی IDA، IDAPython است که امکان نوشتن اسکریپت‌های سفارشی برای IDA را با استفاده از زبان پایتون فراهم می‌کند.

Ghidra ۲-۴

Ghidra یک disassembler است که توسط NSA توسعه داده شده و در سال ۲۰۱۹ به عنوان یک ابزار منبع باز منتشر شده است.



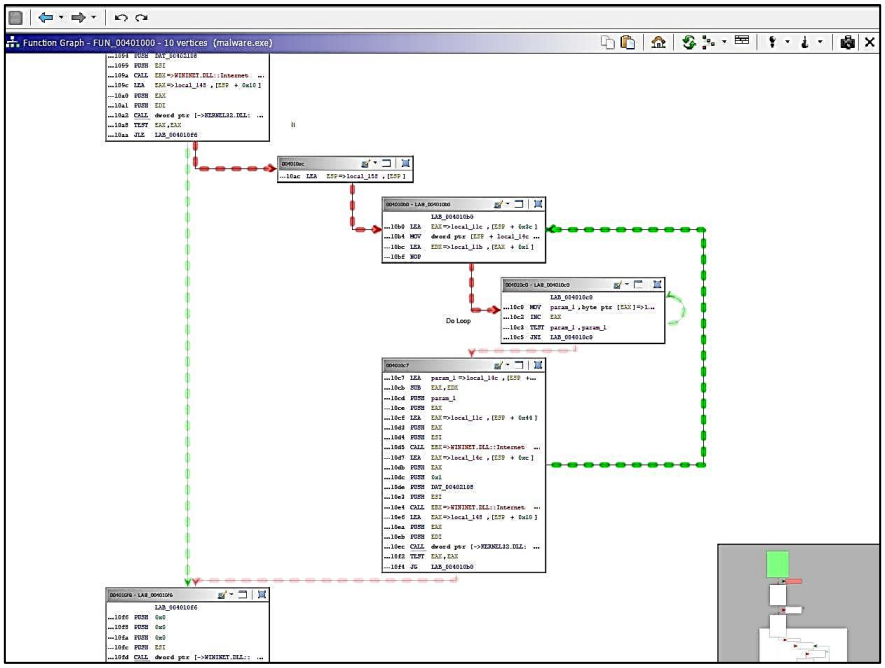
شکل ۱۴: پنجره GHIDRA (A – منو، B – ساختار برنامه، C – واردات، صادرات، توابع، D – مونتاژ، E - Decompiler)

در مقایسه با IDA، Ghidra در اصطلاحات استفاده شده، در ابتدا کمتر کاربرپسند به نظر می‌رسد، شاید به دلیل ظاهر آن. باید به یاد داشت که IDA یک ابزار حرفه‌ای با توسعه تجاری و تاریخچه قابل توجه در زمینه مهندسی معکوس است، در حالی که Ghidra یک ابزار جدید است که تازه منتشر شده است.

https://www.hex-rays.com/products/ida/tech/plugin/

https://ghidra-sre.org/

Ghidra دارای قابلیت‌های مشابه با IDA freeware است، همانطور که در فصل قبل توضیح داده شده است. این فصل ویژگی‌های اضافی آن را نشان می‌دهد. برای تحلیل‌گران مخرب، قابلیت نمایش تفسیر گرافیکی ساختار کد مشابه یک نمودار بلوک (بلوک‌های کد، شاخه‌ها، شرایط و غیره)، درک بهتری از الگوریتم را فراهم می‌کند. برای دسترسی به این عملکرد، روی آیکون "Display Function Graph" در پنل اصلی کلیک کنید یا به منو "Window" > "Function Graph" بروید.

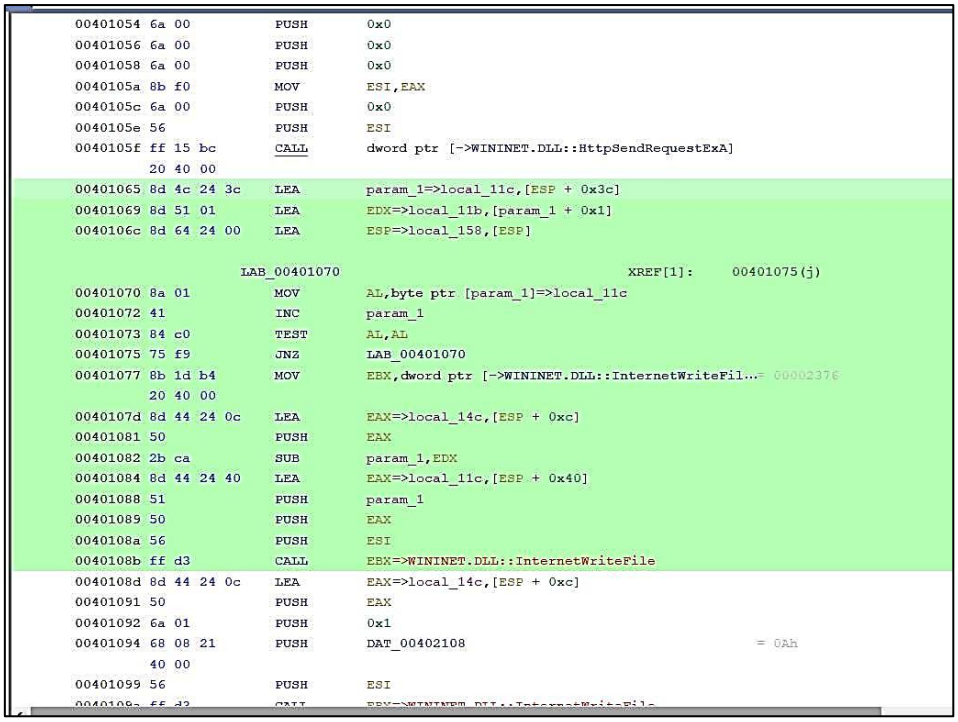


شکل ۱۵: نمودار تابع در GHIDRA

Ghidra با داشتن دیکامپایلر قدرتمند، نسبت به نسخه رایگان IDA، آن را برتری می‌بخشد. در حالی که IDA همچنین قابلیت دیکامپایلر را دارد، اما این قابلیت فقط در نسخه تجاری آن و به عنوان یک افزونه با پرداخت اضافی در دسترس است.

دیکامپایلرها کد اسمبلی را به زبان برنامه‌نویسی سطح بالا ترجمه می‌کنند که باعث کاهش زمان تجزیه و تحلیل می‌شود. زبان سطح بالا نسبت به کد اسمبلی مألوف‌تر است، بنابراین کمترین زمان برای خواندن نیاز است؛ کد به خوبی ساختار یافته شده است و منطق الگوریتم بیشتر آشکار است.

Ghidra با دیکامپایلر خود، کد اسمبلی را به زبان C به صورت پیش‌فرض ترجمه می‌کند. در پنجره پیش‌فرض Ghidra، هم دیکامپایل شده و هم کد اسمبلی وجود دارد. آن‌ها همگام هستند: هنگامی که از کد اسمبلی یا کد C عبور می‌کنید، نشانگر بخش‌های مشابه کد را به صورت همزمان در هر دو پنجره با رنگ سبز مشخص می‌کند، همانطور که در شکل‌های ۱۶ و ۱۷ در صفحه بعدی نشان داده شده است.



```
00401054 6a 00          PUSH    0x0
00401056 6a 00          PUSH    0x0
00401058 6a 00          PUSH    0x0
0040105a 8b f0          MOV     ESI,EAX
0040105c 6a 00          PUSH    0x0
0040105e 56             PUSH    ESI
0040105f ff 15 bc       CALL    dword ptr [->WININET.DLL::HttpSendRequestExA]
         20 40 00
00401065 8d 4c 24 3c    LEA     param_1=>local_11c,[ESP + 0x3c]
00401069 8d 51 01       LEA     EDX=>local_11b,[param_1 + 0x1]
0040106c 8d 64 24 00    LEA     ESP=>local_158,[ESP]

                    LAB_00401070                                    XREF[1]:    00401075(j)
00401070 8a 01          MOV     AL,byte ptr [param_1]=>local_11c
00401072 41             INC     param_1
00401073 84 c0          TEST    AL,AL
00401075 75 f9          JNZ     LAB_00401070
00401077 8b 1d b4       MOV     EBX,dword ptr [->WININET.DLL::InternetWriteFil...= 00002376
         20 40 00
0040107d 8d 44 24 0c    LEA     EAX=>local_14c,[ESP + 0xc]
00401081 50             PUSH    EAX
00401082 2b ca          SUB     param_1,EDX
00401084 8d 44 24 40    LEA     EAX=>local_11c,[ESP + 0x40]
00401088 51             PUSH    param_1
00401089 50             PUSH    EAX
0040108a 56             PUSH    ESI
0040108b ff d3          CALL    EBX=>WININET.DLL::InternetWriteFile
0040108d 8d 44 24 0c    LEA     EAX=>local_14c,[ESP + 0xc]
00401091 50             PUSH    EAX
00401092 6a 01          PUSH    0x1
00401094 68 08 21       PUSH    DAT_00402108                                = 0Ah
         40 00
00401099 56             PUSH    ESI
```

شکل ۱۶: کد مونتاژ در GHIDRA

```
undefined4 uVar2;
int iVar3;
char *pcVar4;
undefined4 local_14c;
_WIN32_FIND_DATAA local_148;

hFindFile = FindFirstFileA("\\*",(LPWIN32_FIND_DATAA)&local_148);
local_14c = 0;
if (hFindFile != (HANDLE)0x0) {
  uVar2 = HttpOpenRequestA(param_1,&DAT_00402100,&DAT_004020fc,0,0,&PTR_s_text/html_00403020,
                           0x80000000,0);
  HttpSendRequestExA(uVar2,0,0,0,0);
  pcVar4 = local_148.cFileName;
  do {
    cVar1 = *pcVar4;
    pcVar4 = pcVar4 + 1;
  } while (cVar1 != 0);
  InternetWriteFile(uVar2,local_148.cFileName,pcVar4 + -(int)(local_148.cFileName +
  1),&local_14c)
  ;
  InternetWriteFile(uVar2,&DAT_00402108,1,&local_14c);
  iVar3 = FindNextFileA(hFindFile,(LPWIN32_FIND_DATAA)&local_148);
  while (0 < iVar3) {
    pcVar4 = local_148.cFileName;
    local_14c = 0;
    do {
      cVar1 = *pcVar4;
      pcVar4 = pcVar4 + 1;
    } while (cVar1 != 0);
    InternetWriteFile(uVar2,local_148.cFileName,pcVar4 + -(int)(local_148.cFileName + 1),
                      &local_14c);
    InternetWriteFile(uVar2,&DAT_00402108,1,&local_14c);
```

شکل ۱۷: – کد زبان c دکامپایل شده در GHIDRA

۵ تحلیل پویا

۵-۱ توضیحات

تجزیه و تحلیل پویا بدافزار با تحلیل کد در حال اجرا انجام می‌شـود. برای مطالعه رفتار فایل اجرایی، توصـیه می‌شـود آن را در داخل یک محیط آزمایشـگاه مجازی اجرا کنید. برای درک عملکرد بدافزار و جلوگیری از گسترش آن، مهندسین معکوس هنگام انجام تجزیه و تحلیل پویا پیشرفته از دیباگرها استفاده می‌کنند.

۵-۲ ابزارهای تحلیل رفتار

۵-۲-۱ ابزار Process Monitor

Process Monitor برای نظارت بر ایجاد یا پایان یک فرآیند یا ارائه اطلاعات بیشتر در مورد یک فرآیند خاص توسـط تحلیل‌گر اسـتفاده می‌شـود. این ابزار ویژگی‌های دو ابزار Sysinternals (Regmon و Filemon) را ترکیب کرده و قابلیت فیلترینگ را اضـافه می‌کند. این ویژگی‌ها باعث می‌شـوند Process Monitor یک ابزار ضروری باشد که هر تحلیل‌گری باید آن را در کیت ابزار شکار بدافزار خود قرار دهد.

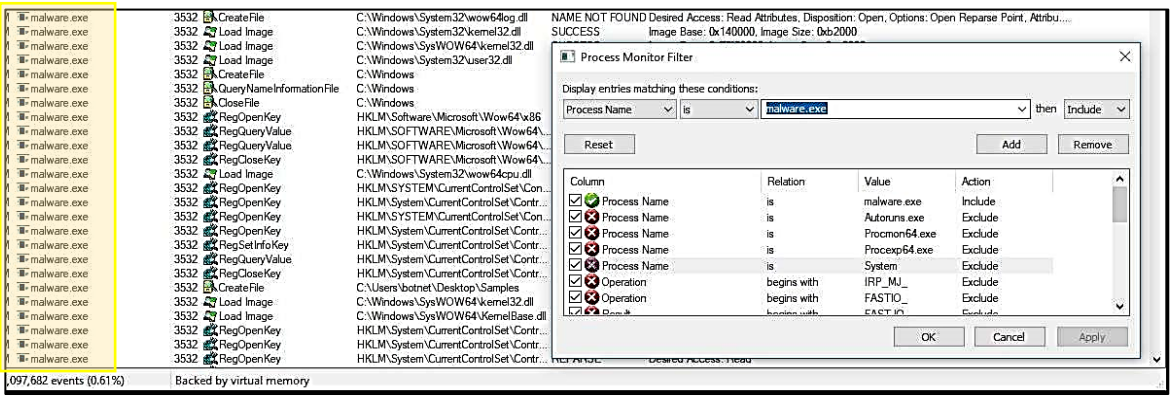
Process Monitor دارای قابلیت نظارت، ضبط و فیلتر کردن چندین نشانه است که در وب‌سایت مایکروسافت توضیح داده شده است.

Process Monitor دارای قابلیت‌های زیر است:

- ضبط بیشتر داده‌های ورودی و خروجی عملیات.
- فیلترهای غیرمخرب به شما اجازه می‌دهند تا فیلترها را بدون از دست دادن داده‌ها تنظیم کنید؛
- ضبط قابل اعتماد جزئیات فرآیند، از جمله مسیر تصویر، خط فرمان، کاربر و شناسه نشست؛
- فیلترها برای هر فیلد داده‌ای، از جمله فیلدهایی که به عنوان ستون پیکربندی نشده‌اند، تنظیم می‌شوند؛
- ابزار درخت فرآیند رابطه تمام فرآیندهای مرجع در یک ردیابی نشان می‌دهد؛
- فرمت لاگ اصـلی تمام داده‌ها را برای بارگیری در یک عملیات Process Monitor متفاوت حفظ می‌کند؛
- ثبت زمان بوت برای تمام عملیات.

برای دریافت تمام رویدادهای فرآیندها و رجیستری، تحلیل‌گر باید ابزار Process Monitor را با دسترسی مدیر اجرا کند. در تصویر زیر، با استفاده از قابلیت فیلتر Process Monitor و اعمال فیلتری که حاوی نام نمونه مورد نظر برای تجزیه و تحلیل اسـت (در این مورد malware.exe)، تحلیل‌گر می‌تواند رویدادهایی که توسـط نمونه ایجاد شده‌اند را مشاهده کرده و بر اساس آن‌ها همبستگی‌هایی برقرار کند، پس از اجرای نمونه.

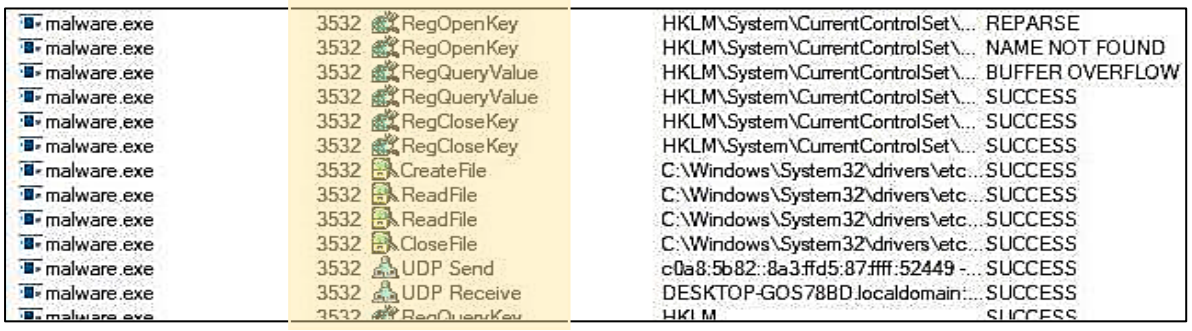
https://docs.microsoft.com/en-us/sysinternals/downloads/procmon



شکل ۱۸: پس از فیلتر نام فرآیند در Process Monitor

در مثال ارائه شـده، پس از بررسـی رویدادها، تحلیل گر تصـویر بهتری از آنچه بدافزار در حال تلاش برای انجام آن است، خواهد داشت. به عنوان مثال، در تصویر زیر، فایل اجرایی 'malware.exe' در حال خواندن کلیدهای رجیستری، ایجاد فایل‌ها و شروع اتصالات شبکه است..

بررسـی تمام اقدامات بدافزار در یک سـیسـتم، تحلیل گر را به یک ایده از هدف و نیت فایل اجرایی مخرب می‌رسـاند. این نوع تجزیه و تحلیل باید قبل از ادامه تجزیه و تحلیل عمیق کد با اسـتفاده از تکنیک‌های تجزیه و تحلیل بدافزار ایستا در دیس‌آسمبلر IDA انجام شود.



شکل ۱۹: مانیتور فرآیند – پس از فیلتر فرآیند "MALWARE.EXE"

برای دنبال کردن رویدادهای مهم در میان رویدادهای متعدد، داشتن فیلترهای مناسب بسیار مهم اسـت. صـفحه وب مایکروسافت ۱۲ لینکی به فایل "Malware Analysis.PMF" دارد که در آن فیلترهای متعددی پیش‌تنظیم شده‌اند.

شامل فیلترها:

- TCP/UDP Send and Receive هر اتصالی که بدافزار ممکن است در حال اجرا آن را استفاده کند
- Load Image بارگذاری DLL/Executable
- Create File ایجاد فایل‌های جدید
- Write/Delete/Rename File هر تغییری در فایل‌ها
- فعالیت‌های رجیستری - ورودی‌های Run برای پایداری بدافزار.

https://learn.microsoft.com/en-us/archive/blogs/motiba/process-monitor-for-dynamic-malware-%20analysis

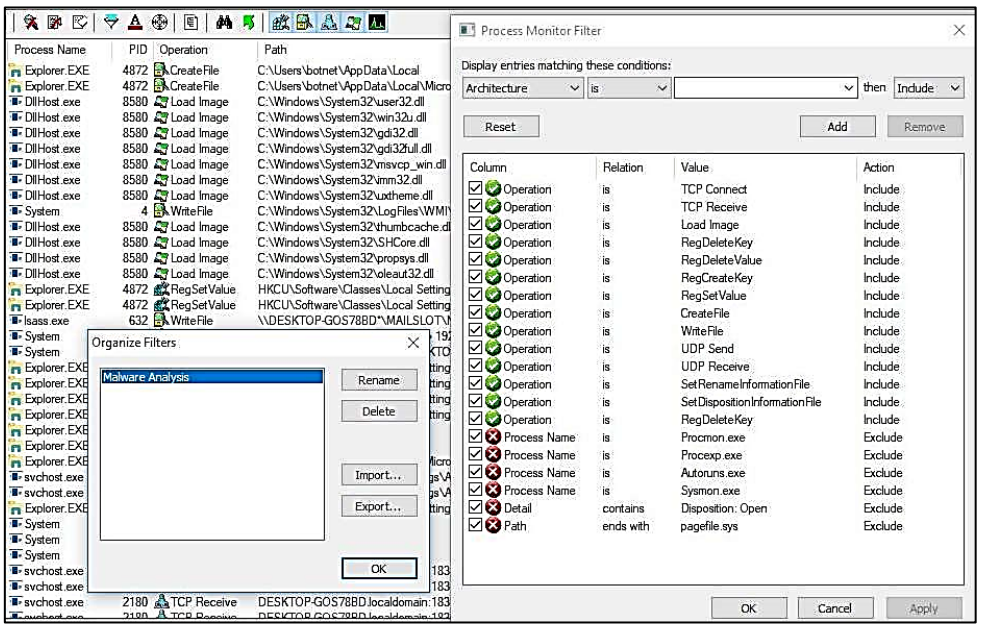
فیلترهای حذف شده ای که معمولاً مرتبط به تجزیه و تحلیل بدافزار نیستند:

Procmon/Procmon64/Autoruns/Sysmon: این‌ها رویدادهای مربوط به ابزارهای Sysinternals را حذف می‌کنند.

Disposition: Open: برای فیلتر کردن هر تماس برای ایجاد فایل که برای باز کردن یک فایل به کار می‌رود.

Page File: فایل صفحه در تجزیه و تحلیل بدافزار کمتر/نامربوط است.

کاربر می‌تواند با استفاده از منوی Filter->Organize Filters فیلتر را به Process Monitor بارگذاری کند و سپس آن را وارد کند.



شکل ۲۰: مانیتور فرآیند - پیکربندی فیلتر تجزیه و تحلیل بدافزار

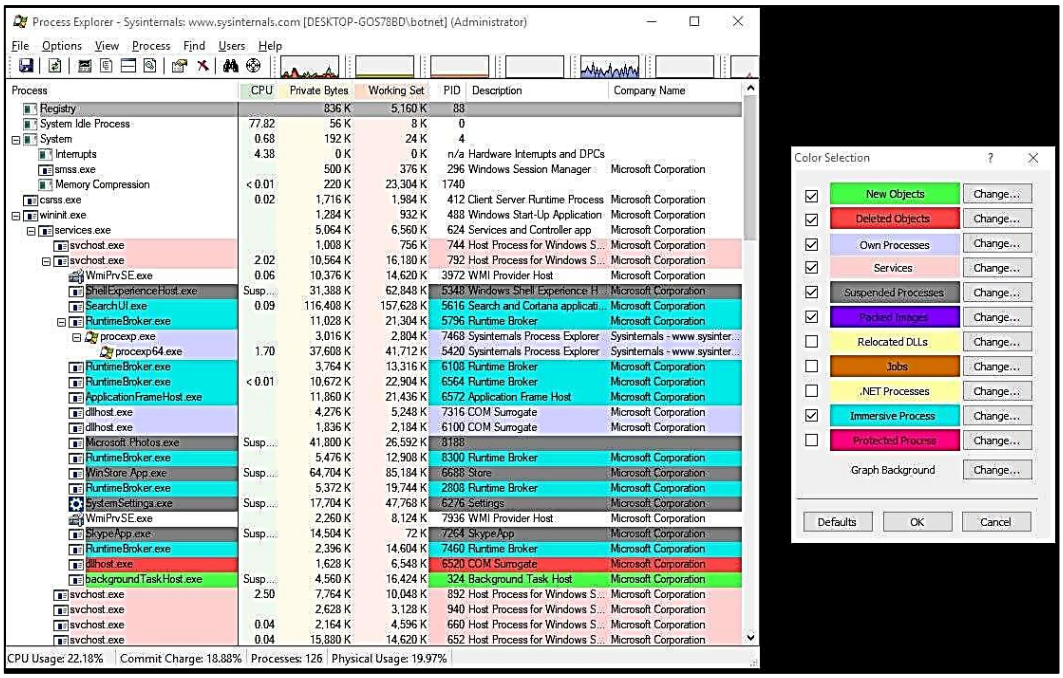
Process Monitor بخشی از بسته SysInternals Suite است و در زمان نگارش مقاله، می‌توانید آن را از وب‌سایت زیر دانلود کنید:

https://learn.microsoft.com/en-us/sysinternals/downloads/procmon

۲-۲-۵ Process Explorer

Process Explorer یک ابزار قدرتمند مدیریت فرآیند است که برای ارائه بینش درباره تمام فرآیندهای در حال اجرا استفاده می‌شود. فرآیندهای در حال اجرا در سیستم در یک ساختار درختی نشان داده می‌شوند که رابطه فرزند و والدین را نشان می‌دهد.

رابط گرافیکی Process Explorer و کد رنگ در زیر نشان داده شده است:



شکل ۲۱: فیلتر انتخاب رنگ در Process Explorer

نمایش اولیه مجموعه ای از ستون ها را در اختیار کاربر قرار می دهد که عبارتند از:

- Process: نام فایل اجرایی به همراه آیکون آن در صورت وجود.
- CPU: درصد زمان پردازنده در ثانیه آخر (یا نسبت به سرعت بروزرسانی).
- Private Bytes: مقدار حافظه اختصاص داده شده به این برنامه به تنهایی.
- Working Set: مقدار RAM واقعی که توسط ویندوز به این برنامه اختصاص داده شده است.
- PID: شناسه فرآیند.

- Description: توضیحات، اگر برنامه دارد.
- Company Name: این یکی مفیدتر از آنچه که فکر می‌کنید است. اگر چیزی درست نیست، با جستجوی فرآیندهایی که توسط مایکروسافت تولید نشده‌اند، شروع کنید.

قابلیت های Process Explorer:

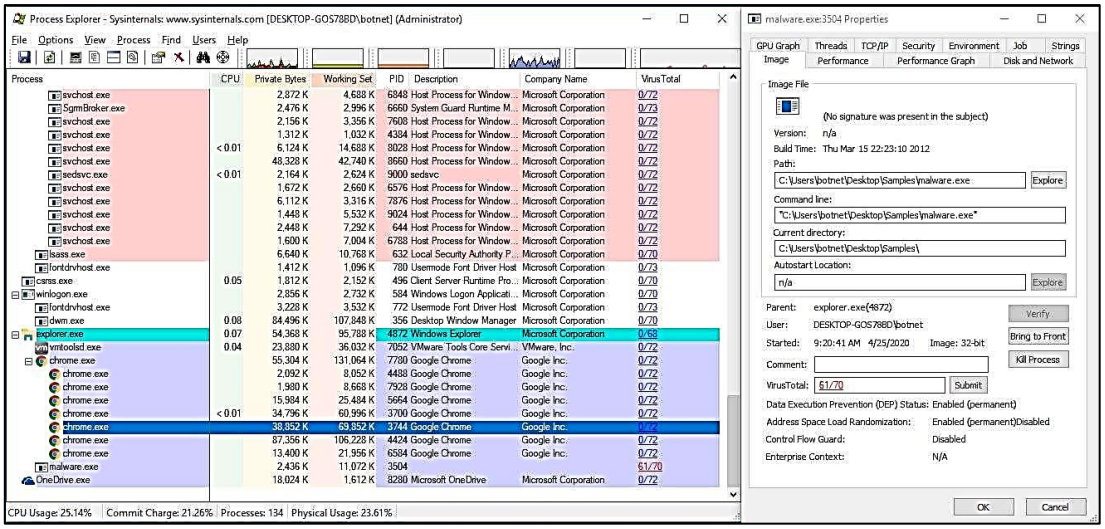
- نمایش ساختار درختی فرآیندها به صورت پیش‌فرض که روابط والدین و فرزندان را با استفاده از رنگ‌ها نشان می‌دهد.
- پیگیری دقیق مصرف CPU برای فرآیندها.
- امکان اضافه کردن چند آیکون سینی به منظور نظارت بر CPU، Disk، GPU، شبکه و موارد دیگر.
- شناسایی فرآیندی که یک فایل DLL را بارگذاری کرده است.
- شناسایی فرآیندی که یک پنجره باز را اجرا می‌کند.
- امکان مشاهده داده‌های کامل در مورد هر فرآیند، از جمله نخ‌ها، مصرف حافظه، دستگیره‌ها، اشیاء و هر اطلاعات مهم دیگر.
- امکان قطع یک درخت فرآیند کامل، از جمله هر فرآیندی که توسط فرآیندی که شما انتخاب می‌کنید شروع شده است.
- امکان تعلیق یک فرآیند و توقف همه نخ‌های آن به گونه‌ای که هیچ کاری انجام ندهند.
- امکان مشاهده نخی در یک فرآیند که بیشترین مصرف CPU را دارد.

https://www.howtogeek.com/school/sysinternals-pro/lesson2/

پیشنهاد می‌شود که از Process Explorer به همراه Process Monitor استفاده شود زیرا Process Explorer ویژگی‌هایی را فراهم می‌کند که به تحلیل‌گر کمک می‌کند تا با فرآیند تعامل کند و رفتار فرآیند مخرب را به صورت دقیق‌تر تجزیه و تحلیل کند.

برای بررسی سریع سیستم و فرآیندهای در حال اجرا، Process Explorer یک گزینه دارد که به تحلیل‌گر اجازه می‌دهد تا تمام هش‌ها را در VirusTotal جستجو کند و تعداد تشخیص‌ها را نمایش دهد. به عنوان مثال، در تصویر زیر، کاربر می‌تواند ببیند که نام فرآیند "malware.exe" که فرآیند فرزند "explorer.exe" است

(۶۱ از ۷۰) تشخیص را دارد، که نشان دهنده احتمال بالایی برای این است که این برنامه مخرب باشد. بررسی پنجره‌های ویژگی‌ها (با دو بار کلیک بر روی فرآیند باز می‌شود) که در سمت راست تصویر نشان داده شده است، می‌تواند مجموعه دیگری از اطلاعات مفید را فراهم کند، به عنوان مثال، کاربری که فرآیند در حال اجرا است، رشته‌های موجود در حافظه، نخ‌های فعال، اتصالات فعال شبکه که بدافزار آن‌ها را شروع می‌کند و مسیر کامل فایل اجرایی در دیسک.



شکل ۲۲: ویژگی های فایل MALWARE.EXE در Process Explorer

Process Explorer بخشی از بسته SysInternals Suite است و در زمان نگارش، می‌توانید آن را از وب‌سایت زیر دانلود کنید:

https://docs.microsoft.com/en-us/sysinternals/downloads/process-explorer

Regshot ۳-۲-۵

Regshot یک ابزار است که به تحلیل گر اجازه می‌دهد تا دو عکسبرداری از رجیستری ویندوز (قبل و بعد از بدافزار) انجام دهد تا تغییراتی که در رجیستری ایجاد شده‌اند یا فایل‌هایی که توسط فایل اجرایی مخرب رها شده‌اند را شناسایی کند. سپس تحلیل گر می‌تواند از این اطلاعات برای ایجاد IoC استفاده کند.

رابط کاربری گرافیکی ابزار Regshot در تصویر زیر ارائه شده است:

Regshot دارای مراحل استفاده زیر است:

۱- اولین عکسبرداری از رجیستری سیستم را هنگامی که سیستم تمیز است انجام دهید.

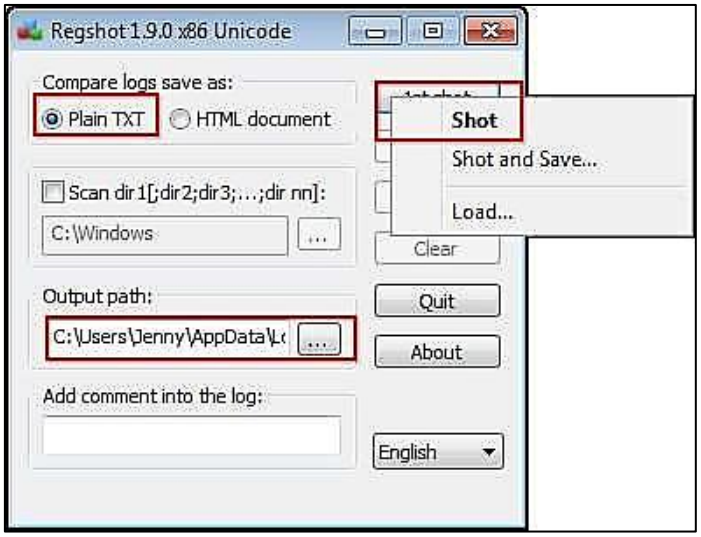
۲- نمونه بدافزار را اجرا کنید.

۳- دومین عکسبرداری از رجیستری سیستم را پس از بدافزار انجام دهید.

۴- دکمه "مقایسه" را برای مقایسه دو عکسبرداری تولید شده فشار دهید.

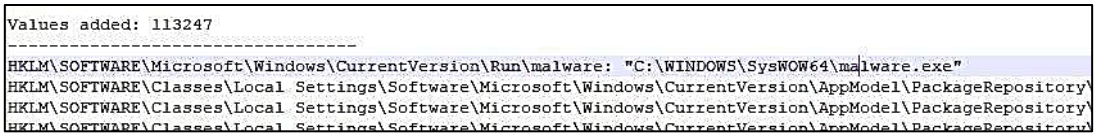
۵- گزارش تولید شده را تحلیل کنید.

۶- بر روی یک سیستم جدید و تمیز شروع کنید.



شکل ۲۳: SNAPSHOT در REGSHOT

در مثال زیر، پس از اجرا و مقایسه عکسبرداری دوم با اولین عکسبرداری انجام شده هنگامی که سیستم تمیز بود، تحلیل‌گر شناسایی کرده است که فایل اجرایی "malware.exe" داده‌ها را در رجیستری در 'HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\malware' با نام 'C:\WINDOWS\SysWOW64\malware.exe' ایجاد می‌کند تا برای پایداری در سیستم استفاده شود. بررسی گزارش کامل رجیستری که تغییرات را در آن ثبت می‌کند، به تحلیل‌گر کمک می‌کند تا تصویر دقیقی از رفتار برنامه مخرب بدست آورد.



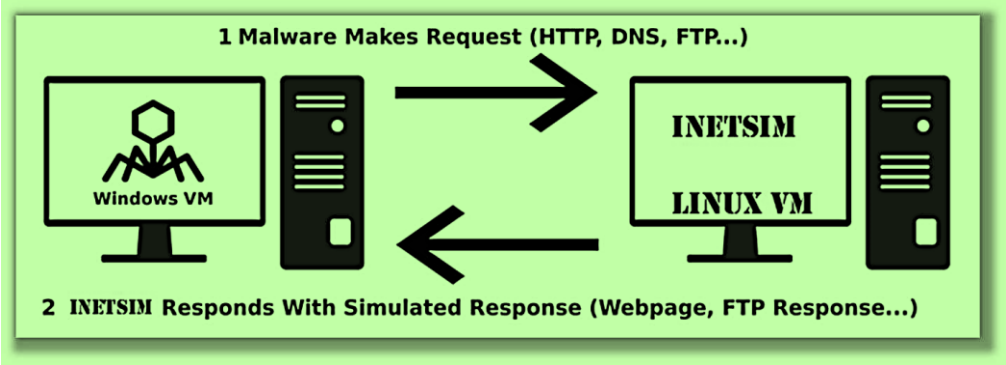
شکل ۲۴: گزارش SNAPSHOT در REGSHOT

در زمان نوشتن، ابزار Regshot را می توان از وب سایت زیر دانلود کرد:

https://sourceforge.net/projects/regshot

INetSim ۴-۲-۵

INetSim یک مجموعه نرم‌افزاری مبتنی بر لینوکس است که به کاربر امکان می‌دهد خدمات متعدد استاندارد اینترنت را در یک ماشین مجازی برای تحقیقات شبیه‌سازی کند. با استفاده از این ابزار، تحلیل گر می‌تواند رفتار شبکه نمونه بدافزار را بدون اتصال به اینترنت نظارت کند. اگر شما در حال انجام تحقیقات در ویندوز هستید، راه ساده‌تر استفاده از این ابزار، استفاده از ماشین مجازی لینوکس (جایی که ابزار INetSim پیکربندی و در حال اجرا است) به عنوان دروازه برای ماشین مجازی ویندوز است. نحوه راه‌اندازی این ابزار در تصویر زیر نشان داده شده است:

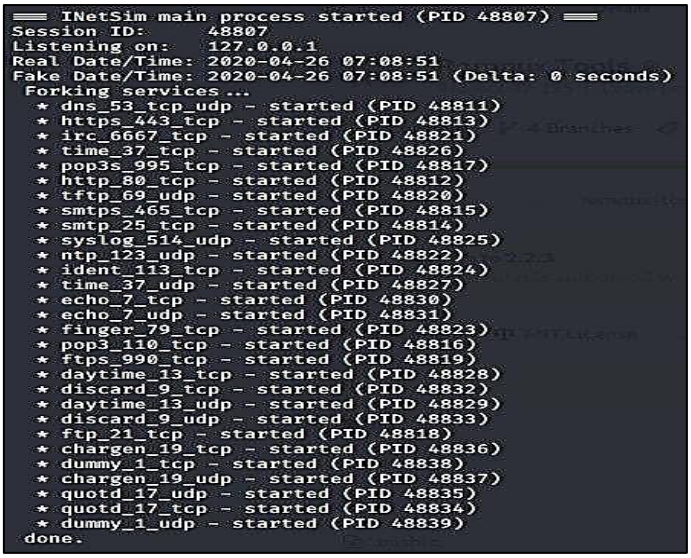


شکل ۲۵: راه اندازی INETSIM

پس از اجرای ابزار، تصویر سمت چپ تمام سرویس های شبیه سازی شده توسط INetSim، از جمله پورت پیش فرض آنها را نشان می دهد.

به منظور تغییر تنظیمات پیکربندی ابزار برای افزودن یا حذف خدمات، کاربر باید فایل 'etc/inetsim/inetsim.conf' را تغییر دهد.

هنگام اجرا، INetSim همه اتصالات ورودی/خروجی را ضبط می کند، بنابراین تحلیلگر می تواند IOC ها را بر اساس اتصالاتی که فایل مخرب در تلاش است ایجاد کند، بسازد.



شکل ۲۶: خروجی سرویس های در حال اجرا در INETSIM

در زمان نوشتن، ابزار Regshot را می توان از وب سایت زیر دانلود کرد:

https://www.inetsim.org/downloads.html

۳-۵ Sandboxing

برای محدود کردن گسترش بدافزار و حفاظت از محیط خود، تحلیل گران بدافزار نمونه بدافزار را درون یک راه‌حل سندباکس اجرا می کنند. ابزارهای سندباکس معمولاً گزینه دامپ حافظه فرآیند را ارائه می‌دهند، بنابراین تحلیل گر می تواند تصویر بهتری از چه اتفاقی در حافظه RAM رخ می‌دهد، بدست آورد.

نویسندگان بدافزار می‌دانند که اگر نمونه بدافزار آن‌ها درون یک ماشین مجازی یا راه‌حل سندباکس اجرا شود، احتمالاً نمونه توسط یک مهندس معکوس یا راه‌حل خودکار تحلیل می‌شود، بنابراین آن‌ها معمولاً یک بررسی متفاوت پیاده‌سازی می کنند. برای کسب اطلاعات بیشتر درباره انواع بررسی‌هایی که بدافزار ممکن است پیاده‌سازی کند، لطفاً بخش مربوط به خودحفاظت بدافزار در فصل دوم را بررسی کنید.

چندین راه حل سندباکس رایگان، که در آن یک تحلیلگر می تواند نمونه را آپلود کند و منتظر گزارش باشد، در اینترنت موجود است. در زمان نگارش، شناخته شده ترین آنها عبارتند از:

- ✓ www.malwr.com
- ✓ www.hybrid-analysis.com
- ✓ www.any.run
- ✓ www.joesandbox.com
- ✓ www.cuckoosandbox.org

- ✓ www.sandbox.anlyz.io
- ✓ www.analyze.intezer.com

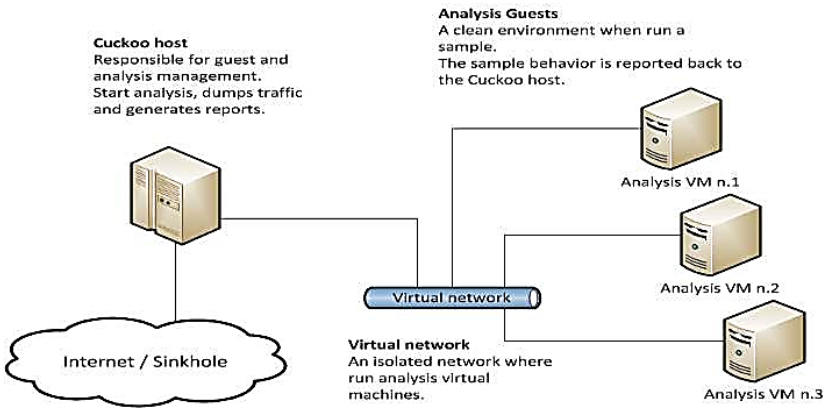
۱-۳-۵ Cuckoo Sandbox

این راهنما ویژگی‌ها و مشخصات Cuckoo Sandbox را به عنوان معروف‌ترین سیستم تحلیل خودکار بدافزار متن باز، معرفی می‌کند. با استفاده از سندباکس، تحلیل‌گران می‌توانند وظیفه تحلیل هر فایل مخرب را در ویندوز، macOS، لینوکس یا اندروید را به صورت خودکار انجام دهند. سندباکس می‌تواند به صورت محلی پیاده‌سازی شود و برای آن نیاز به یک میزبان (نرم‌افزار مدیریت) و چند مشتری سندباکس (ماشین‌های مجازی برای تحلیل) دارد.

Cuckoo Sandbox یک برنامه کاربردی است که ویژگی های زیر را داراست:

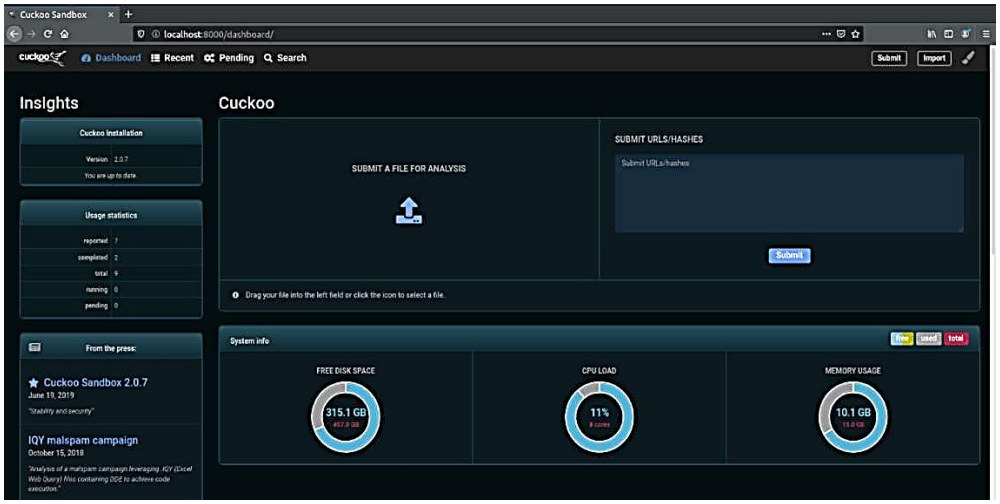
- تهیه عکس از اجرای بدافزار
- از بین بردن و دانلود فایل ها را متوقف می کند
- حافظه فرآیندهای بدافزار را ذخیره می کند
- تحلیل همزمان روی چندین دستگاه اجرا می کند
- ترافیک شبکه تولید شده را به صورت فرمت PCAP ذخیره می کند
- فرآیندهای جدید را به صورت بازگشتی نظارت می کند
- تماس های API مربوط به تحلیل رفتاری را پیگیری می کند
- حافظه کامل VM را به دست می آورد.

نمودار زیر معماری Cuckoo's را نشان می دهد:



شکل ۲۷: معماری سندباکس در Cuckoo

با توجه به طراحی ماژولار آن، "Cuckoo" می تواند به عنوان یک برنامه مستقل یا به صورت یک قسمت از چارچوب های بزرگتر استفاده شود. این محیط شناسایی با استفاده از کنسول وب قابل دسترسی است که نمونه های بدافزار برای تحلیل به آن ارسال می شوند. کنسول وب در تصویر زیر نشان داده شده است:



شکل ۲۸: کنسول وب سندباکس Cuckoo

پس از ارسال فایل ها به محیط شناسایی با استفاده از کنسول وب، آنها اجرا می شوند و تمام فعالیت های آنها ثبت و در گزارش نهایی قرار می گیرد. تحلیلگر می تواند با استفاده از کنسول وب به گزارش دسترسی پیدا کرده و آن را مطالعه کند. محیط شناسایی "Cuckoo Sandbox" چندین فرمت گزارش دارد، از جمله فرمت خوانا برای انسان، فرمت - MAEC (Malware Attribute Enumeration and Characterization) یک زبان استاندارد توسعه داده شده توسط - MITRE و قابلیت صادر کردن گزارش داده به فرمت دیگری.

در زمان نوشتن، اطلاعات بیشتر در مورد نصب و استفاده از راه حل Cuckoo Sandbox را می توانید در صفحه وب بیابید:

https://cuckoo.readthedocs.io/en/latest/installation/host/installation

https://cuckoo.readthedocs.io/en/latest/introduction/what/

۲-۳-۵ Windows Sandbox

در ویندوز ۱۰، نسخه ۱۹۰۳ (بروزرسانی ماه مه ۲۰۱۹)، ویژگی جدیدی به نام Windows Sandbox اضافه شده است. محیط Sandbox به منابع سیستمی زیادی نیاز ندارد و تنها حدود ۱۰۰ مگابایت فضای دیسک را مصرف می‌کند. محیط Windows Sandbox در شکل ۲۹ زیر نشان داده شده است.

شکل ۲۹: محیط گرافیکی سندباکس WINDOWS

برخی از این ویژگی ها:

- x64 architecture
- Virtualisation capabilities enabled in BIOS
- At least 4GB of RAM (8GB recommended)
- At least 1 GB of free disk space (SSD recommended)
- At least 2 CPU cores (4 cores with hyperthreading recommended)

هر بار که تحلیلگر ویژگی Windows Sandbox را اجرا می کند، یک نصب جدید و پاک از ویندوز ۱۰ ایجاد می شود. پس از تحلیل فایل اجرایی، وقتی تحلیلگر محیط Sandbox را می بندد، همه چیزی که در محیط بوده حذف می شود. با استفاده از این تکنیک، تحلیلگر می تواند به راحتی برنامه های مخرب یا ناشناخته را تست کند در حالی که اطمینان حاصل می کند که محیط کاری ایمن و پاک باقی می ماند.

برای استفاده از ویژگی Windows Sandbox، کاربر باید هایپروویزر Microsoft را فعال کند. همچنین، محیط Sandboxامکان سفارشی سازی جوانب مختلف محیط را نیز فراهم می کند. به عنوان مثال:

- GPU مجازی شده را فعال یا غیرفعال کنید
- فعال یا غیرفعال کردن شبکه درsandbox
- پوشه ها را از میزبان به اشتراک بگذارید
- یک اسکریپت یا برنامه راه اندازی را اجرا کنید

Windows Sandbox در نسخه‌های ۶۴ بیتی Windows 10 Pro، Enterprise و Education موجود است. برای نسخه Home در دسترس نیست:

https://docs.microsoft.com/en-us/virtualization/hyper-v-on-windows/about

برای فعال کردن این ویژگی، Sandbox به دنبال یک فایل پیکربندی با پسوند "WSB." می‌گردد. برای کسب اطلاعات بیشتر در مورد نحوه فعال سازی و پیکربندی Windows Sandbox، می‌توانید به وبلاگ Microsoft Community مراجعه کنید.

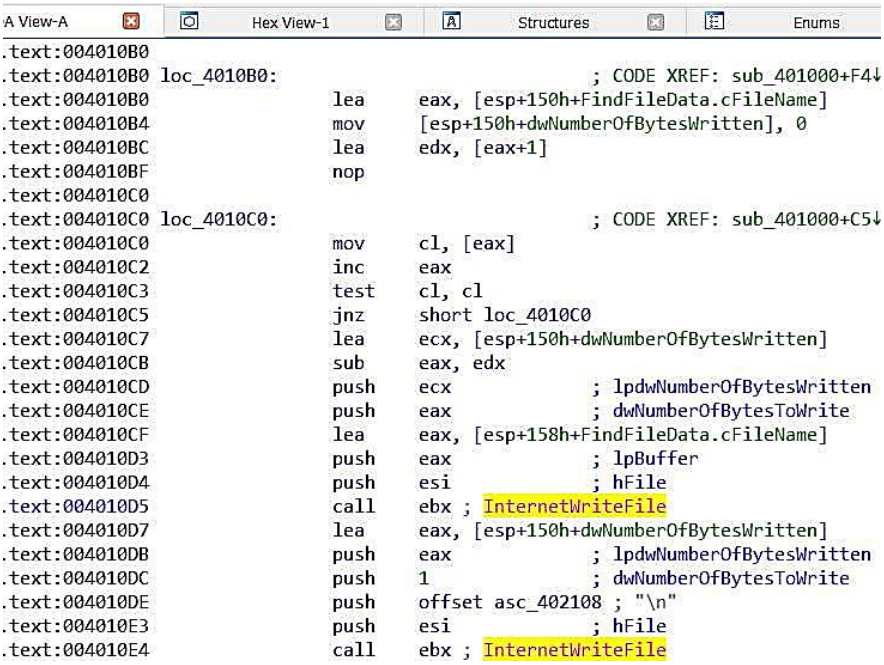
۴-۵ Debuggers

در نگاه اول، دیباگر شبیه به دیس‌آسمبلر به نظر می‌رسد: هر دو کد نمونه‌های مورد بررسی را به صورت اسمبلی نمایش می‌دهند و لیستی از توابع، رشته‌ها و غیره را ارائه می‌کنند. اما تفاوت آنها در این است که دیباگر امکان نظارت دقیق بر اجرای کد مخرب را فراهم می‌کند، از جمله دسترسی به حافظه، رجیسترها، استک و عناصر کنترل. مزیت دیباگر این است که فرصت اجرای کد، کنترل اجرا (دستور به دستور، نقاط وقفه و غیره) و مشاهده مقادیر خاص در رجیسترها، پارامترهای توابع و مقادیر بازگشتی آنها را فراهم می‌کند که باعث بهبود درک کد می‌شود.

چندین دیباگر منبع باز برای فایل های اجرایی وجود دارد، از جمله WinDbg، x64dbg، Immunity Debuggerو OllyDbg در ادامه، نمونه های زیر با استفاده از x64dbg نشان داده شده است.

۱-۴-۵ Breakpoint

هنگامی که یک نمونه مشکوک در IDA تحلیل می‌شود، فراخوانی تابع API 'InternetWriteFile' در آدرس‌های 'x004010D5' و 'x004010E4' شناسایی شد. همانطور که از نام آن پیداست، این تابع داده ها را از طریق شبکه ارسال می‌کند. پارامترهای تابع مقصد('hFile') ، داده هایی که باید ارسال شوند ('lpBuffer') طول داده هایی که باید ارسال شوند ('dwNumberOfBytesToWrite') و مقدار داده هایی که ارسال شده‌اند ('lpdwNumberOfBytesWritten') را تعریف می‌کنند. مقصد در فایل اجرایی به صورت سخت‌کد شده است و قبلاً کشف شده است (بخش ۴.۱ IDA را ببینید). روشن است که تابع در آدرس 'x004010E4' کاراکتر '\n' را ارسال کرده است. اما نوع داده ارسال شده در آدرس 'x004010D5' هنوز مشخص نیست.



شکل ۳۰: پارامترهای تابع "INTERNETWRITEFILE" در IDA

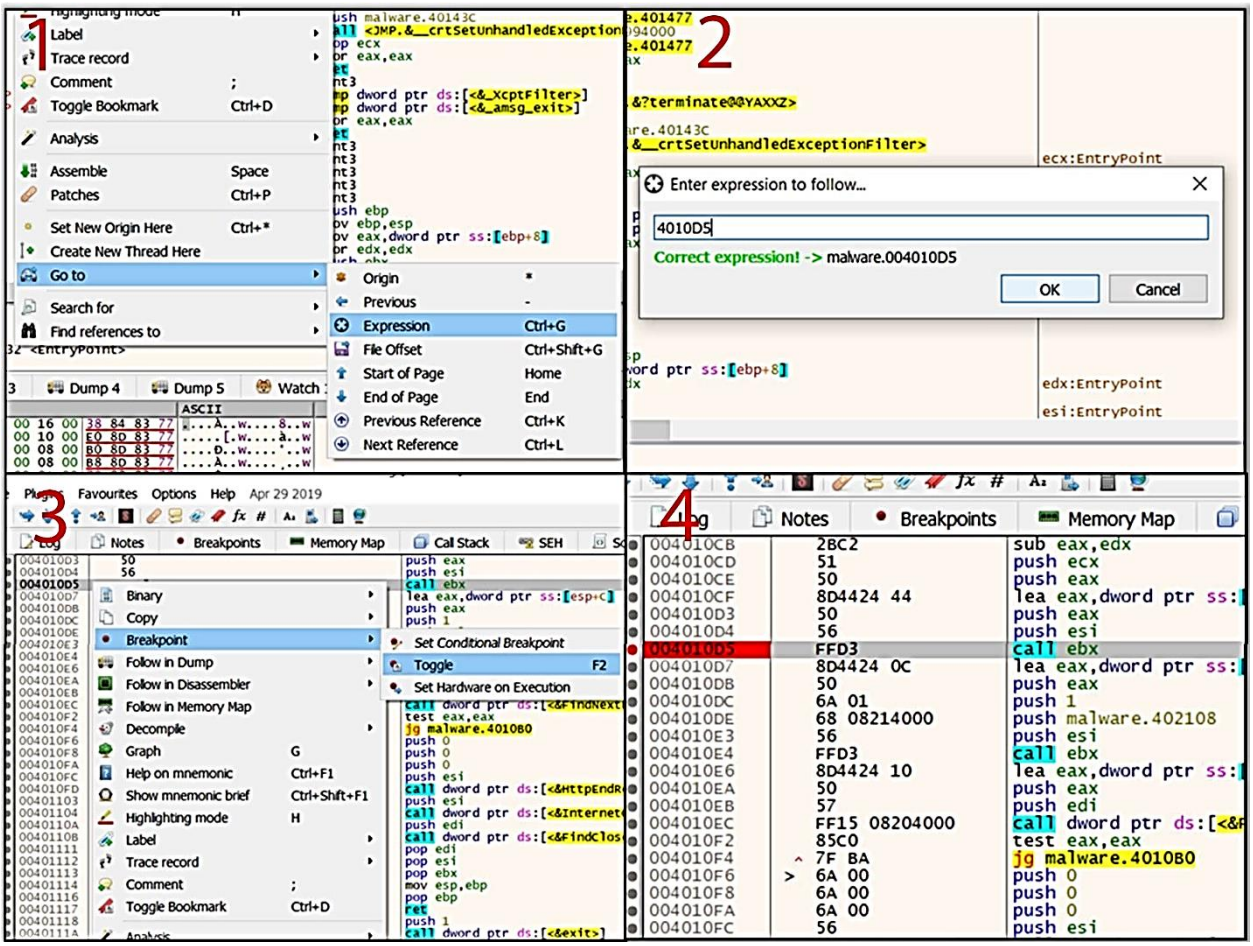
https://techcommunity.microsoft.com/t5/windows-os-platform-blog/windows-sandbox/ba-p/301849
https://docs.microsoft.com/en-us/windows-hardware/drivers/debugger/

https://x64dbg.com/

https://www.immunityinc.com/products/debugger/index.html
http://www.ollydbg.de/

راحت‌ترین روش برای تحلیل آن، نظارت بر آن در دیباگر اســـت. پس از باز کردن فایل اجرایی درx64dbg ، یک breakpoint باید در آدرس 'x004010D5' (که در IDA پیدا شده است، آدرسی که تابع از آن فراخوانی شده است) قرار داده شود:

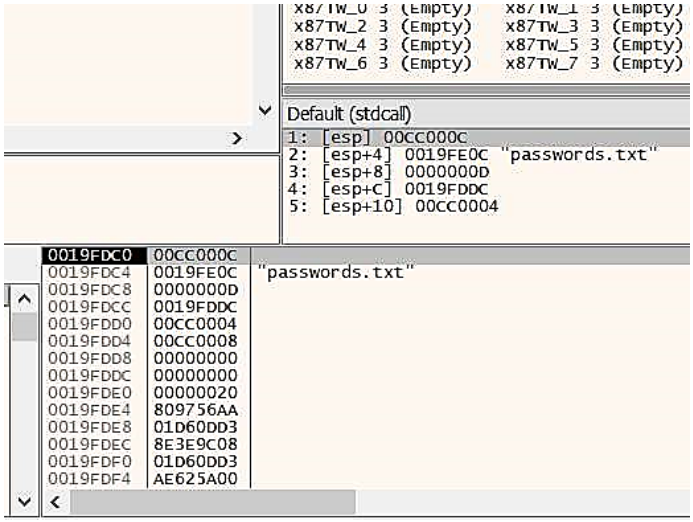
۱. روی کد راست کلیک کرده و 'Go to' > 'Expression' یا کلیدهای (CTRL+G) را انتخاب کنید. یک جعبه گفتگو ظاهر می‌شود.
۲. آدرس را در جعبه گفتگو وارد کرده و روی OK کلیک کنید.
۳. با راست کلیک کردن روی آدرس مورد نیاز F2 or press ('Toggle' > 'Breakpoint' > right-click).
۴. آدرس با breakpoint با رنگ قرمز مشخص می‌شود.



شکل ۳۱: تعیین BREAKPOINT در آدرس خاص X64DBG

سپس با فشردن کلید F9 (یا از طریق منو 'اشکال‌زدایی' > 'اجرا')، دیباگر را برای اجرای فایل اجرایی فعال کنید. دیباگر به breakpoint می‌رسد و متوقف می‌شود. داده‌هایی که توسط 'InternetWriteFile' ارسال می‌شوند، در منطقه استک قابل مشاهده هستند.

احتمال بالایی وجود دارد که توابع API مربوط به شبکه و مدیریت فایل در طول اجرای برنامه چندین بار فراخوانی شوند (حلقه ارسال بیش از یک بسته داده، حلقه پردازش بیش از یک فایل، یک ردیف از یک فایل و غیره). مشاهده داده های دیگری که توسط این توابع API پردازش می شوند، ارزش دارد.

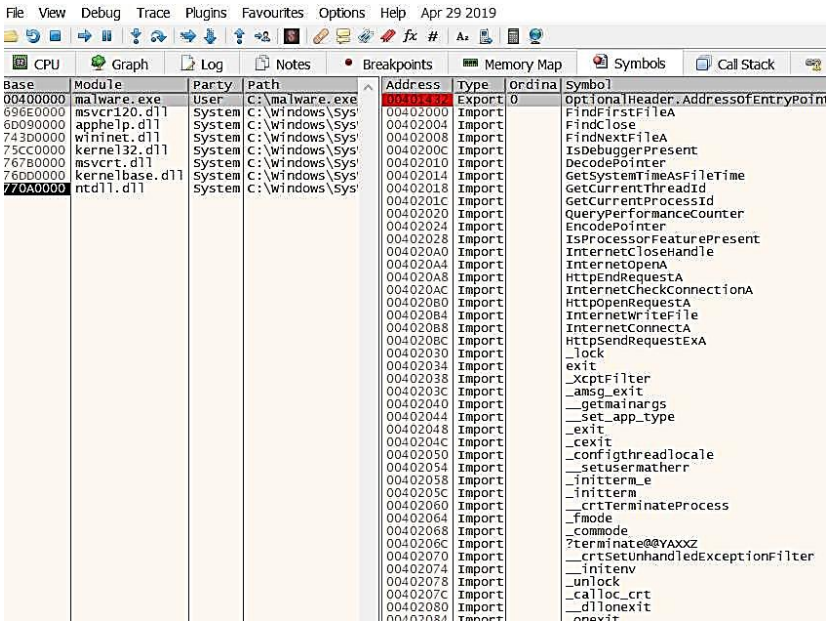


شکل ۳۲: حافظه پشته در X64DBG

۲-۴-۵ Symbols and Intermodular calls

در مثال قبل، آدرسی که تابع مورد نظر از آن فراخوانی شده بود، شناخته شده بود و می‌دانستیم که breakpoint باید در آنجا قرار داده شود. اگر آدرس مورد نظر نامعلوم باشد، باید یک بررسی از توابع و مراجعه‌های متقابل آنها انجام شود و x64dbg ویژگی های داخلی برای این کار دارد: نمادها و تماس های بین ماژولی.

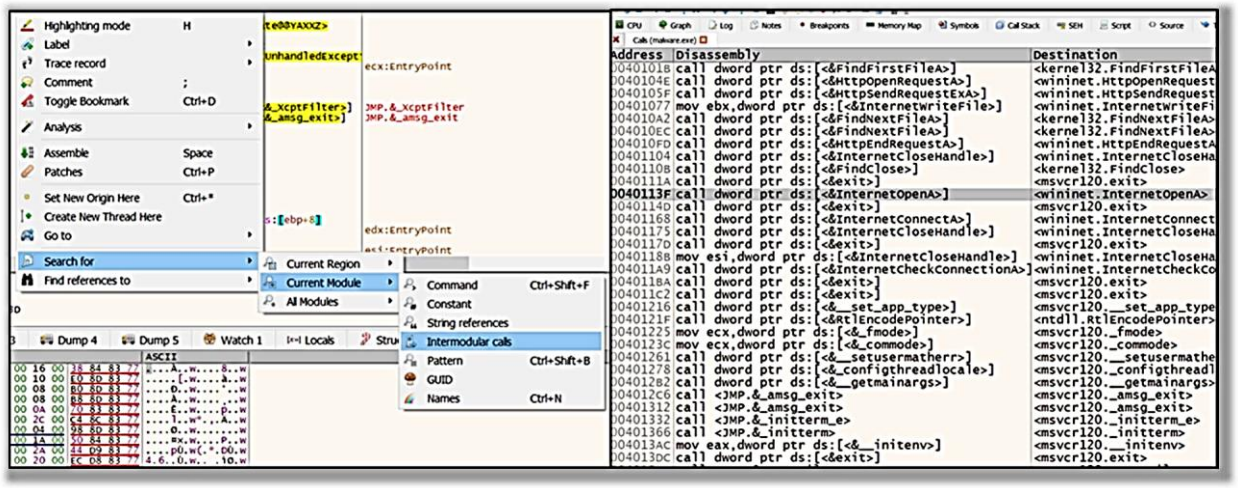
برای دیدن لیستی از نمادها (توابع خارجی وارد شده)، به تب "Symbol" بروید و نام اجرایی را از بین تمام ماژول ها انتخاب کنید) یا کلیدهای CTRL+N را فشار دهید .(این لیست شامل آدرس‌های خاصی که توابع وارد شده از آنها فراخوانی می‌شوند، نیست. این آدرس‌ها در تماس‌های بین ماژولی در دسترس هستند.



شکل ۳۳: نمادها در X64DGB

تماس‌های بین ماژولی از طریق راست کلیک کردن 'Intermodular > 'Current Module' > 'Search for' calls' نشان داده می‌شوند. یک جدول ظاهر می‌شود که شامل اطلاعاتی در مورد اینکه توابع وارد شده چگونه ('Disassembly' column)، کجا ('Address' column) و چه توابع وارد شـــده ('Destination' column) فراخوانی می‌شوند. می‌توان با فشردن کلید F2 تماس‌های مورد نظر را breakpoint گذاشت یا آنها را در منطقه کد بررسی کرد که با دو بار کلیک بر روی آنها قابل تغییر است. لیستی از رخدادهای رشته‌ها نیز می‌تواند تحلیل شود:

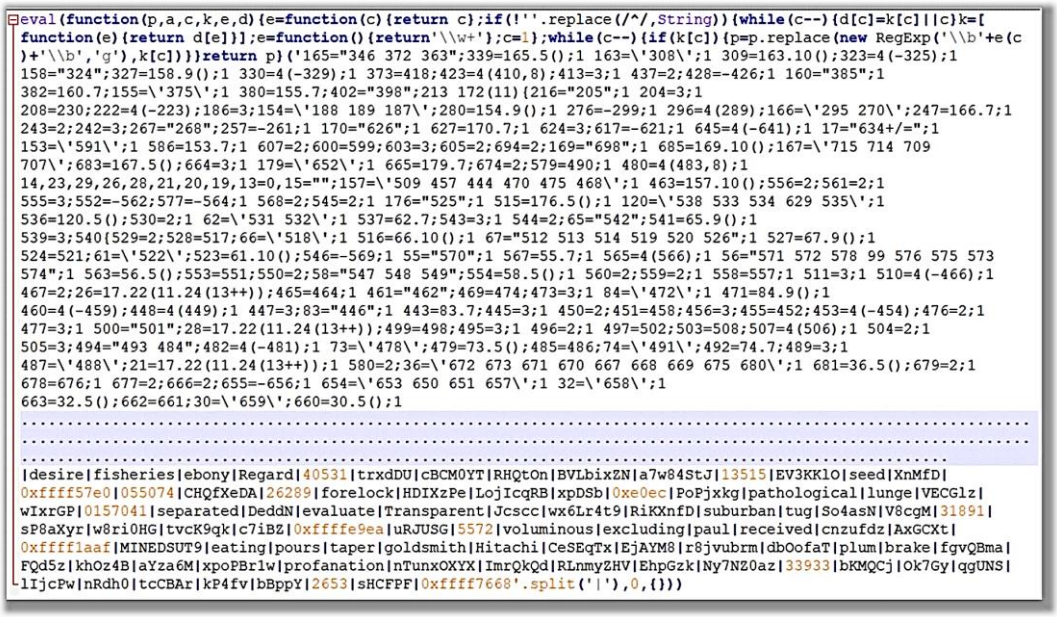
(right-click in 'CPU' > 'Search for' > 'Current Module' > 'String references')



شکل ۳۴: ارتباط بین ماژول‌ها در X64DBG

Deobfuscation ۳-۴-۵

دیباگرها همچنین در مواجهه با اســکریپت‌های مبهم و بدون کامپایل کمک می‌کنند. مثال زیر نحوه تحلیل یک اسکریپت جاوا اسکریپت مبهم به نام 'malware.js' را نشان می‌دهد. این اسکریپت ۱۰ صفحه طول دارد (شکل ۳۵ فقط برای توضیح برش گرفته شده است) و دیباگ کردن دستی آن بسیار چالش برانگیز خواهد بود.



شکل ۳۵: جاوا اسکریپت مبهم

یک فایل جاوا اسکریپت باید توسط یک مفسر اسکریپت اجرا شود. ویندوز یک موتور اسکریپت نیتیو به نام 'wscript.exe'را در دایرکتوری 'C:\Windows\System32' دارد. معمولاً، اسکریپت جاوا اسکریپت مبهم به منظور رها کردن یا دانلود یک فایل جدید مخرب و اجرای آن طراحی شده است. سخت است برآورد کرد که دقیقاً چه می‌تواند باشد و بر روی چه چیزی در دیباگر تمرکز کرد، اما احتمالاً سعی در اجرای یک دستور خودسر در سیستم عامل دارد، بنابراین توابع API از 'shell32.dll' (مانند 'ShellExecute') باید نظارت شوند.

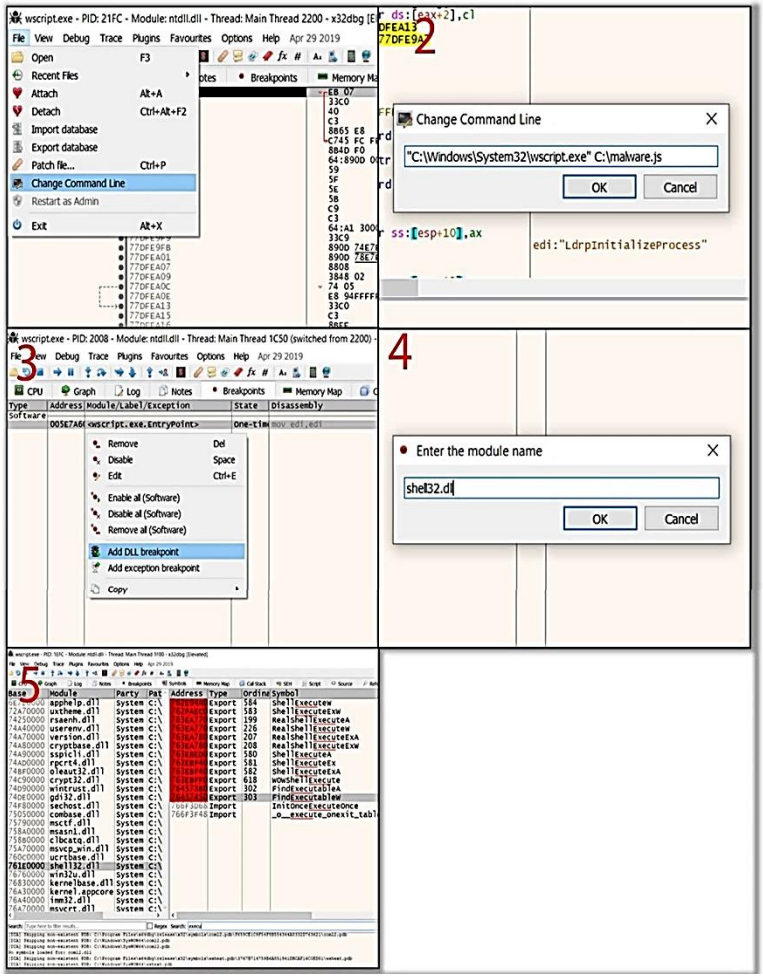
از دیدگاه دیباگر، این به معنای بارگذاری'wscript.exe'، اعلام به 'wscript.exe' برای پردازش فایل جاوا اسکریپت مخرب، قرار دادن breakpoint در 'ShellExecute' و تحلیل محتوای آن هنگام فعال شدن است.

- wscript.exeرا بارگذاری کنید

('File' > 'Open' > 'C:\windows\system32\wscript.exe')

- malware.jsرا به عنوان یک پارامتر اضافه کنید ('File' > 'Change Command Line') و مسیر فایل مخرب را اضافه کنید؛ (به عنوان مثال "C:\Windows\system32\wscript.exe" C:\malware.js'.)
- به پنل 'Breakpoints' بروید راست کلیک کنید 'Add dll breakpoint' را انتخاب کنید و 'shell32.dll' را پر کنید.
- اجرا را اجرا کنید و منتظر شوید تا 'shell32.dll' breakpoint فعال شود(اگر فعال شد، به این معنی است که DLL و نمادهای آن بارگذاری شده اند).

- به پنل 'Symbols' بروید ماژول ها را بین گزینه ها انتخاب کنید توابع 'Execute' را فیلتر کنید و آنها را breakpoint کنید.
- به پنل 'Breakpoints' بازگردید و breakpoint DLL را از مرحله ۴ غیرفعال کنید (در غیر این صورت، تمام اقدامات مرتبط با DLL breakpoint خواهند شد، نه فقط توابعی که به صورت دستی breakpoint شده اند).
- اجرا کنید و منتظر شوید تا یکی از نقاط وقفه 'Execute' فعال شود تا پارامترها را در حافظه استک بررسی کنید.

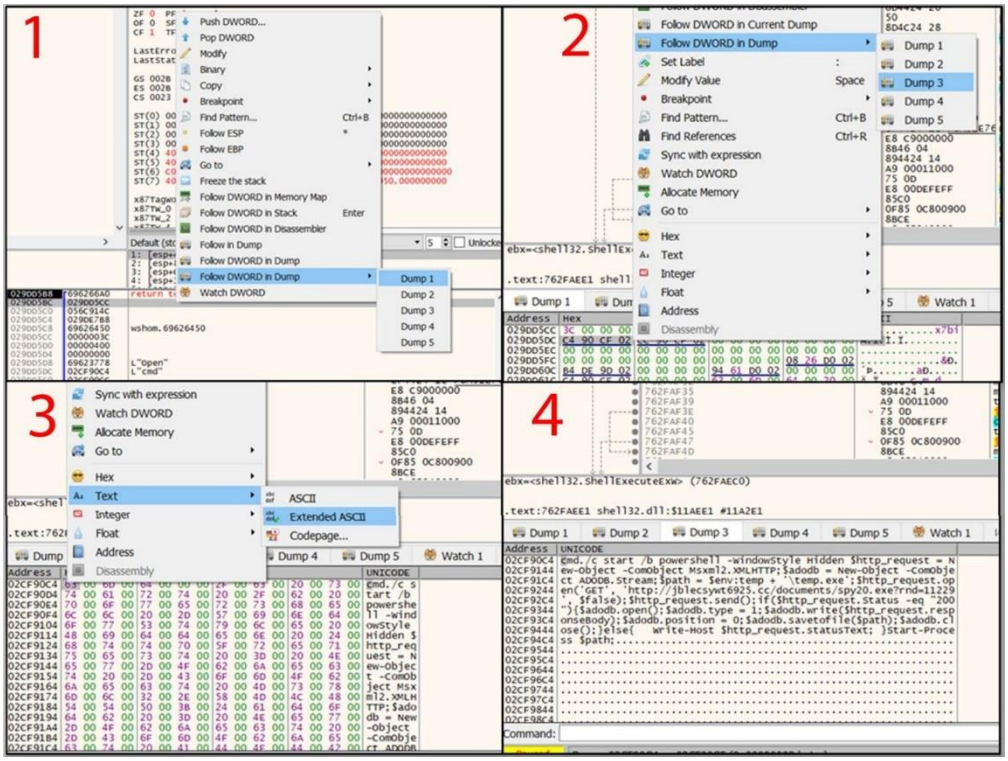


شکل ۳۶: اشکال زدایی جاوا اسکریپت و نقطه شکست DLL در X32DBG

یکی از نقاط وقفه، اجرای کد را در تابع 'ShellExecuteExA' متوقف می‌کند. تابع فقط یک پارامتر دارد – یک اشاره‌گر به ساختار 'SHELLEXECUTEINFOA' برای بررسی آن، روی مقدار اشاره‌گر راست کلیک کرده و 'Follow DWORD in Dump' > 'Dump 1' را انتخاب کنید. مورد پنجم ساختار یک فایل/شیء/فرمان برای اجرا است. برای جزئیات، روی آن در منطقه 'Dump 1' راست کلیک کرده و 'Follow DWORD in

'Dump 2' > 'Dump' را انتخاب کنید و با راست کلیک کردن 'Extended ASCII' > 'Text' فرمت را تنظیم کنید. در این مورد، یک فرمانی که یک اسکریپت کوتاه پاورشل را که یک فایل 'spy20.exe' را از لینک "http://jblecsywt6925.cc/documents/" دانلود می‌کند، آن را به نام 'temp.exe' ذخیره می‌کند و آن را اجرا می‌کند، را آغاز می‌کند.

https://docs.microsoft.com/en-us/windows/win32/api/shellapi/nf-shellapi-shellexecuteexa



شکل ۳۷: بررسی حافظه پشته در X32DBG

Patching ۴-۴-۵

بدافزار می‌تواند مکانیزم‌های دفاعی داشته باشد که برای جلوگیری یا مانع شدن از معکوس‌مهندسی طراحی شده‌اند. این مکانیزم‌ها شامل انواع مختلفی هستند: شناسایی حضور ابزار مانیتورینگ (debugger، Wireshark، Process monitor و غیره)، تست اینکه بدافزار در یک ماشین مجازی اجرا می‌شود یا خیر، بررسی اتصال به اینترنت یا تعامل کاربر، و بسیاری دیگر، از جمله بررسی اینکه آیا در یک محیط شناسایی شده مانند یک سندباکس مورد بررسی قرار می‌گیرد یا خیر. اگر بدافزار هر یک از موارد فوق را شناسایی کند، ممکن است خود را خاتمه دهد یا رفتار خود را به طور ارادی تغییر دهد تا ویژگی‌های واقعی خود را فاش نکند.

تحلیلگر می‌تواند این مکانیزم‌های دفاعی را با روش پچ کردن حذف کند - به عبارت دیگر، کد مخرب را تغییر دهد. برای این کار، تحلیلگر باید مکانیزم دفاعی را در کد شناسایی کرده، آن را تنظیم کرده و آن را به عنوان یک فایل اجرایی جدید ذخیره کند که بدون تأثیر مکانیزم دفاعی مورد بررسی قرار گیرد.

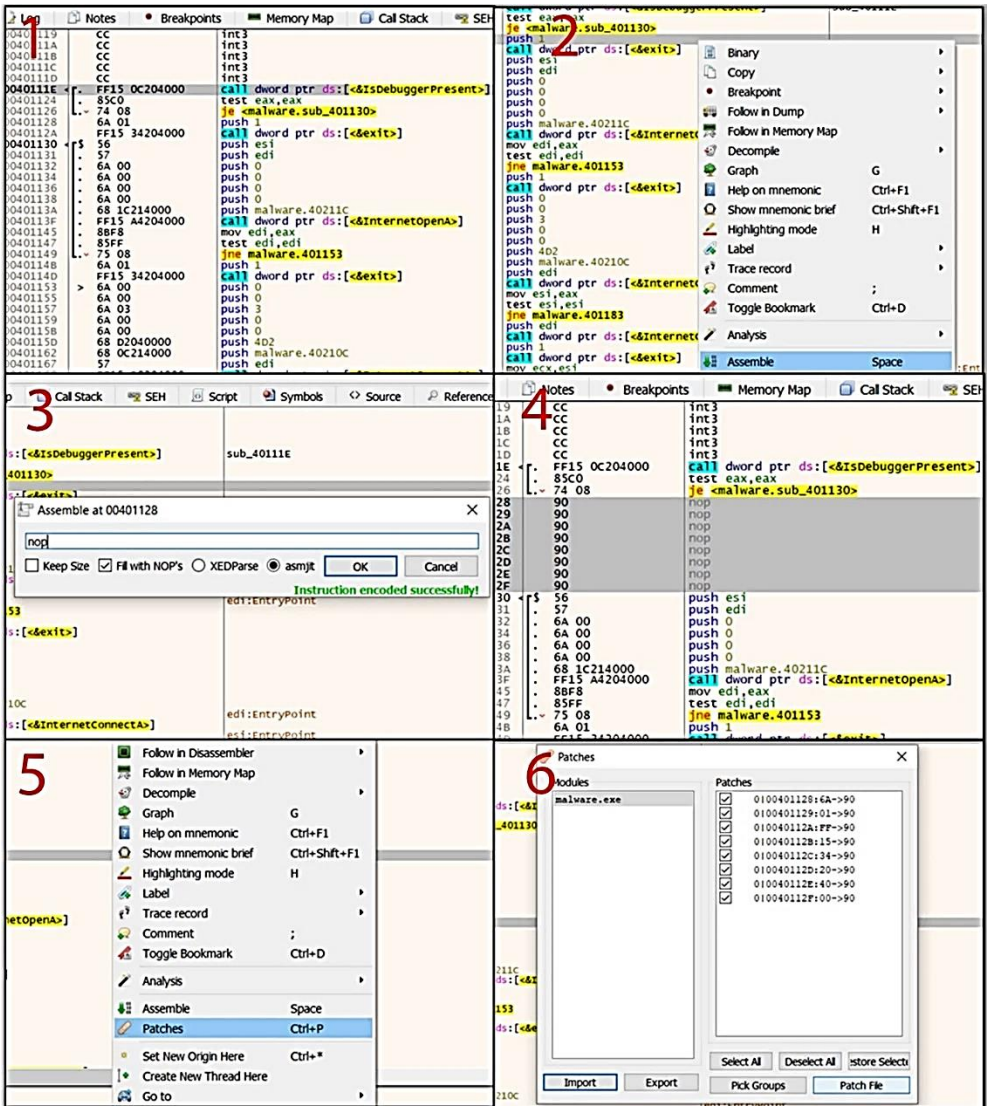
مثال زیر یک مکانیسم دفاعی را نشان می دهد که توسط "IsDebuggerPresent" انجام شده است.

این تابع از کتابخانه استاندارد 'kernel32.dll' است. با فراخوانی این تابع، بدافزار تست می کند که آیا در حضور یک دیباگر در حال اجراست یا خیر. مراحل زیر نشان می‌دهند چگونه مکانیزم دفاعی را غیرفعال کنید:۱ .محل تابع 'IsDebuggerPresent' را در میان فراخوانی‌های ماژول‌های مختلف شناسایی کنید (روی آن راست کلیک کرده و 'Search for' > 'Current Module' > 'Intermodular calls' را انتخاب کنید) و دوبار کلیک کنید.

برای غیرفعال کردن مکانیزم دفاعی، کد را ارزیابی کرده و شناسایی کنید که چگونه این مکانیزم کار می کند و چگونه می‌توان آن را حذف کرد. در این مثال، تابع ('exit' در آدرس 'x0040112A' فراخوانی می‌شود) فرآیند خود را پایان می‌دهد اگر تابع ('IsDebuggerPresent' در آدرس 'x0040111E' قرار دارد) مقدار بولین 'true' را برگرداند (این به معنی اجرای فایل اجرایی در یک دیباگر است). برای اجتناب از این بررسی امنیتی، کافی است فراخوانی تابع 'exit' و دستور 'PUSH 1' قبلی را به عنوان 'nop' بازنویسی کنید، به شرح زیر. هدف دستور 'nop' این است که CPU هیچ کاری انجام ندهد nop (به معنی بدون عملیات است)، که در حذف کد اصلی که نمی‌تواند به سادگی حذف شود، بسیار مفید است و باید با دستورات معتبر جایگزین شود:

۱- محل تابع 'IsDebuggerPresent' را در میان فراخوانی‌های ماژول‌های مختلف شناسایی کنید (روی آن راست کلیک کرده و 'Search for' > 'Current Module' > 'Intermodular calls' را انتخاب کنید) و دوبار کلیک کنید.
۲- خط دستوری که باید جایگزین شود را علامت‌گذاری کرده و فاصله را فشار دهید (یا روی خط راست کلیک کرده و 'Assemble' را انتخاب کنید).
۳- یک پنجره با دستور اصلی ظاهر می‌شود. دستور اصلی را با دستور مورد نیاز (در این مورد 'nop') بازنویسی کرده و روی OK کلیک کنید.
۴- مراحل ۳-۴ را برای تمامی خطوطی که باید تغییر کنند تکرار کنید.
۵- پس از انجام تمامی تغییرات، 'CTRL + P' را فشار دهید (روی آن راست کلیک کرده و 'Patches' را انتخاب کنید).

۶- یک پنجره جدید حاوی خلاصه‌ای از تمامی تغییرات ظاهر می‌شود. روی 'Patch File' کلیک کرده و آن را به عنوان یک فایل جدید ذخیره کنید.



شکل ۳۸: پچ‌های ایجاد شده روی فایل در X32DBG

۶- Network traffic analysis

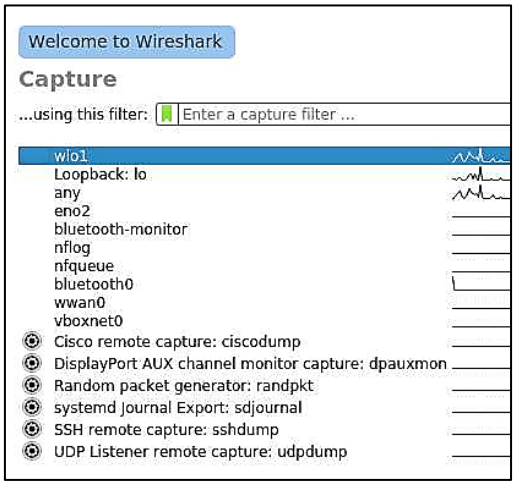
تحلیل ترافیک شبکه در تجزیه و تحلیل بدافزارها بسیار حائز اهمیت است. با نگاه به ترافیک شبکه، تحلیلگر می‌تواند بفهمد کدام فایل‌ها از سیستم خارج می‌شوند، سرورهای C2، نحوه ارتباط بدافزار و بسیاری موارد دیگر را بفهمد. برای تحلیل ترافیک شبکه، از منابع داده‌ای مانندSPAN، پورتهای Mirror و TAP های شبکه استفاده می‌شود. تحلیل ترافیک شبکه به شناسایی ترافیک WAN غیرمجاز و بهکارگیری منابع شبکه کمک می کند، اما ممکن است فاقد جزئیات و اطلاعات زمینهای غنی برای بررسی مشکلات امنیت سایبری باشد.

Wireshark یکی از محبوب‌ترین برنامه‌های تجزیه و تحلیل پروتکل شبکه است که از آن برای ضبط بسته‌های شبکه و نمایش ترافیک شبکه در فایل‌های ضبط شده استفاده می‌شود. با استفاده از Wireshark، می‌توانید جزئیات بسته‌های شبکه را به صورت دقیق مشاهده کنید و از این طریق بتوانید فایل‌هایی که در حال انتقال از شبکه هستند، سرورهای C2، نحوه ارتباطات مخرب و موارد دیگر را شناسایی کنید. Wireshark امکان مشاهده داده‌های بسته‌های شبکه را با جزئیات بالا فراهم می‌کند.

نکته: در زمان نگارش، Wiresharkرا می توان از لینک زیر دانلود کرد:

https://www.wireshark.org/download.html

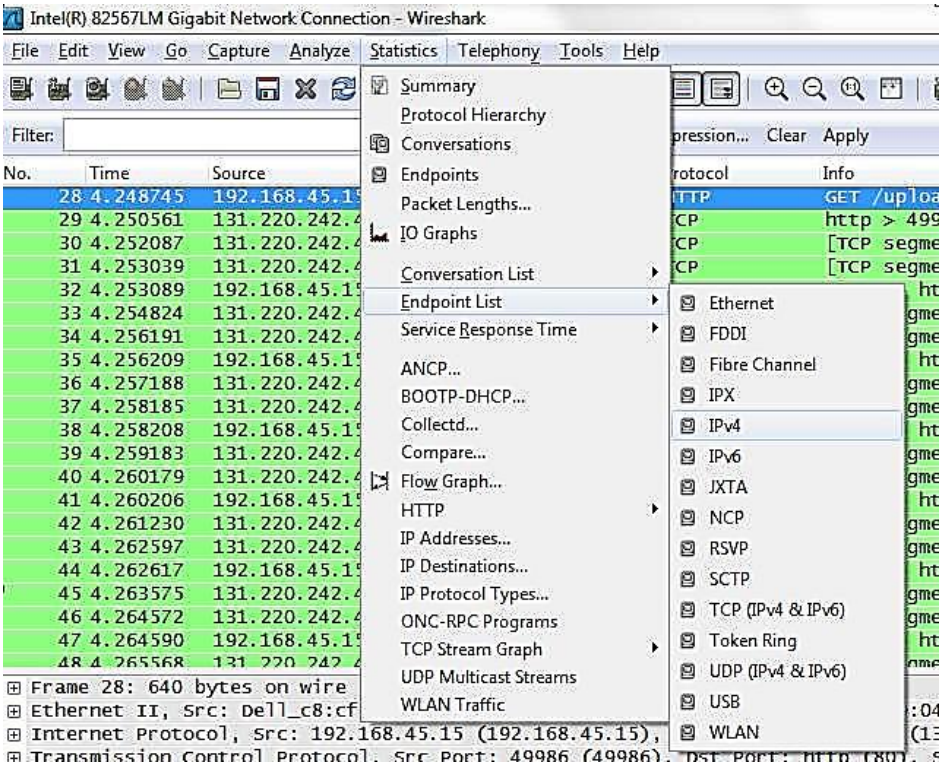
تحلیل ترافیک شبکه بسیار حائز اهمیت است و برای این کار از برنامه‌های تجزیه و تحلیل پروتکل شبکه مانند Wireshark استفاده می‌شود. اما استفاده از Wireshark یا هر نرم‌افزار دیگری برای ضبط بسته‌های شبکه بر روی سیستم قربانی که بدافزار در آن اجرا می‌شود، در نظریه ممکن است، اما دارای مشکلاتی است. بدافزار با مکانیزم‌های خودحفاظت ممکن است تشخیص دهد که در حال نظارت است و رفتار خود را پنهان کند. بنابراین، اجرای Wireshark روی دروازه پیش‌فرض سیستم قربانی، راه‌حل بهتری است. همچنین، می‌توان یک پورت SPANرا بر روی سوئیچ تنظیم کرد تا یک کپی از تمام بسته‌های شبکه‌ای که در پورت سیستم قربانی دیده می‌شوند، ارسال شود.



شکل ۳۹: انتخاب رابط Wireshark

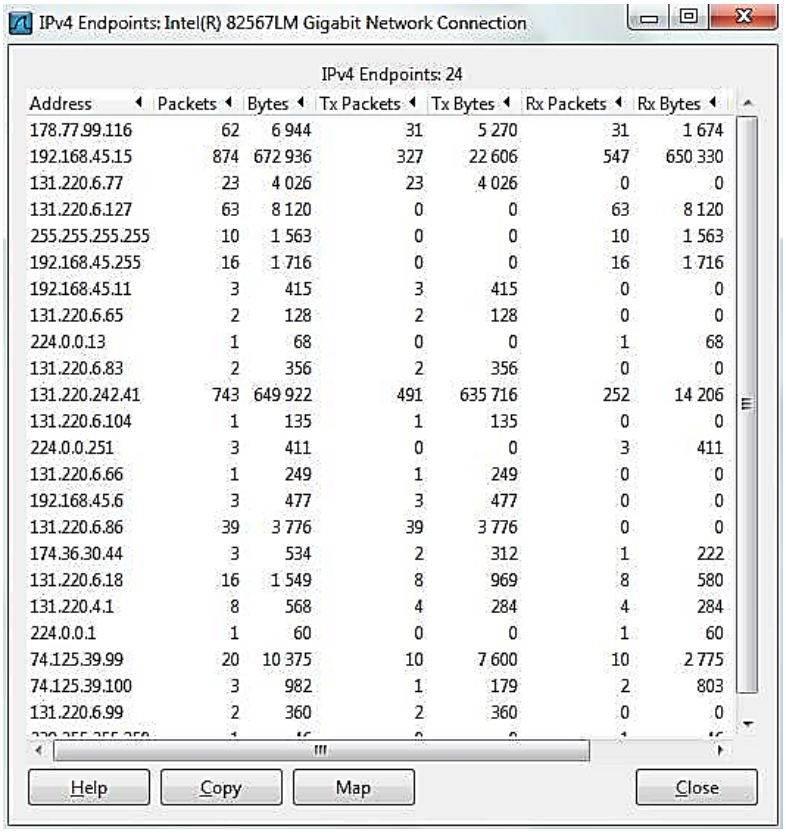
Wireshark با لیست رابط‌های موجود شروع می‌شود. رابطی که بدافزار از آن ارتباط برقرار می‌کند، می‌تواند انتخاب شده و ترافیک می‌تواند ضبط شود. حذف تمام نویزهای موجود در رابط مشخص شده، شناسایی رفتار بدافزار از طریق آن رابط را آسان‌تر می‌کند.

C2 سرور: سرورهای Command and Control سرورهای مهاجم هستند که برای کنترل بدافزارها استفاده می‌شوند. این سرورها شامل مجموعه‌ای از ابزارها و تکنیک‌هایی هستند که مهاجمان برای حفظ ارتباط با بدافزارهایشان استفاده می‌کنند. از جمله کارهایی که می‌توانند توسط C2 سرورها انجام شوند، کنترل بدافزارها، ارسال دستورات به بدافزارها، جمع‌آوری اطلاعات از سیستم‌های قربانی و انتقال اطلاعات به سرورهای مهاجم است.



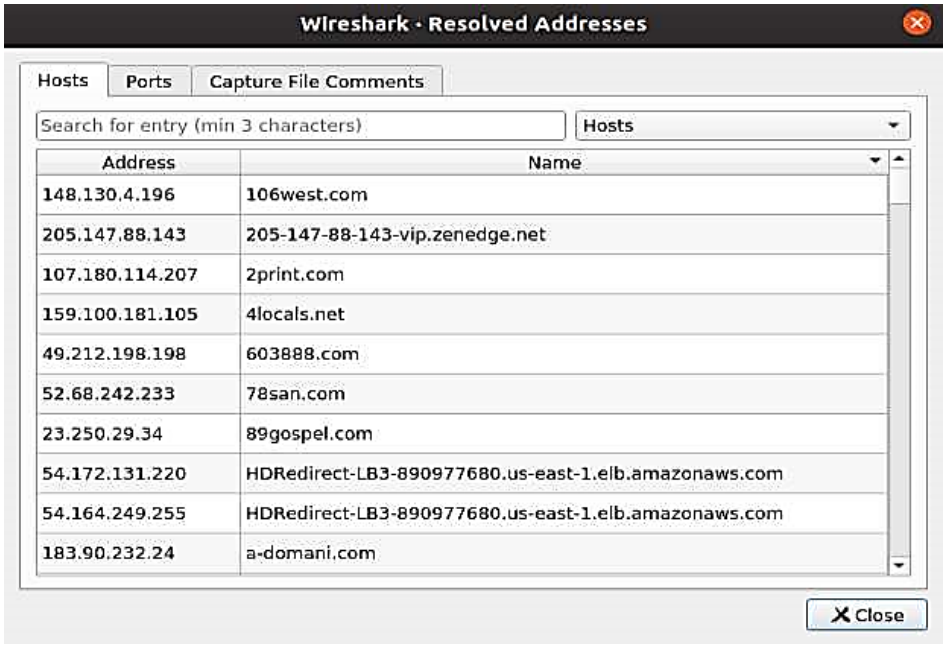
شکل ۴۰: به دست آوردن آمار ترافیک

Wireshark همچنین آمار مفیدی را از دیدگاه تجزیه و تحلیل بدافزار نگه می‌دارد. با استفاده از بخش آمارها، می‌توان انتهای اتصالات و گفتگوها را لیست کرد. در حالی که لیست انتهای اتصالات به مرتب‌سازی انتهای IP با استفاده از تعداد بسته‌های ارسالی اجازه می‌دهد، لیست گفتگوها می‌تواند گفتگوهای بین انتهای اتصالات را بر اساس تعداد بایت‌های منتقل شده بین آن‌ها و مدت زمان تبادل داده‌هایشان مرتب کند. این اطلاعات می‌تواند برای تحلیل رفتار شبکه نامتعارف با آدرس‌های IP مورد استفاده قرار گیرد.



شکل ۴۱: فهرست نقاط پایانی Wireshark

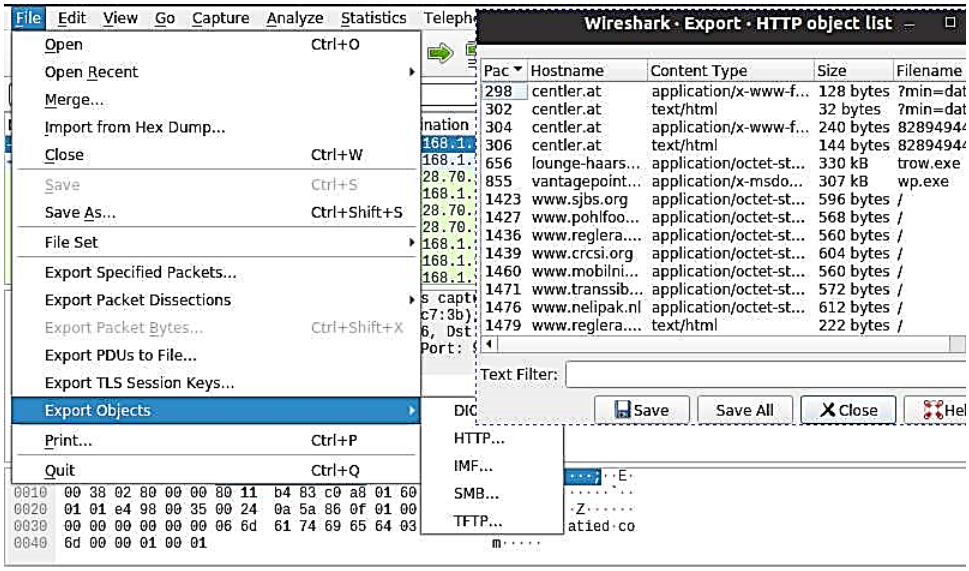
همانطور که در شکل ۴۲ مشاهده می شود، با استفاده از لیست "resolved address"، نام دامنه این آدرس های IP مشکوک را می توان به راحتی و بدون تلاش اضافی پیدا کرد.



شکل ۴۲: فهرست آدرس های حل شده

با اســتفاده ازWireshark، می‌توان با تنظیم فیلترهای نمایش، بســته‌های مورد نظر را تشــخیص داد. این برنامه امکانات فیلترینگ متنوعی را ارائه می‌دهد، از فیلترهای ســاده پروتکل‌ها مانند HTTP، DNS، FTP و غیره تا فیلترهای پیچیده‌تری که می‌توانند با عبارات منطقی ترکیب شــوند. در مثالی که در شــکل ۴۳ نشــان داده شــده اســت، ترافیک HTTP یک آدرس IP منبع مشــکوک را نشــان می‌دهد که بسته‌های آن شــامل رشــته "exe" هستند. در اینجا، با کلیک بر روی"File - Export Objects - HTTP"، می‌توانید این دو فایل را با یک کلیک ذخیره کنید. همه شــیء در ترافیک می‌تواند با اســتفاده از لیســت شــیء صــادر شــود. همچنین Wireshark آمار مفیدی را از دید تحلیل بدافزار نگه می‌دارد.

شکل ۴۳: فیلتر کردن فایل های EXE از یک IP خاص



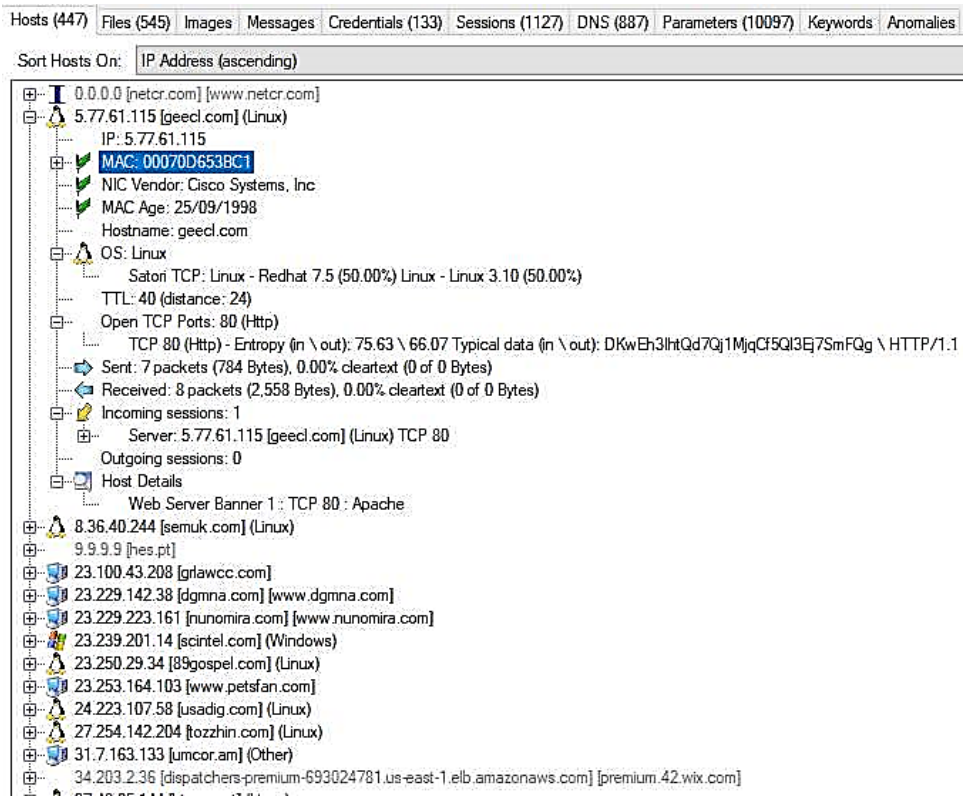
شکل ۴۴: خروجی گرفتن از Objectهای ترافیک

در حالی که Wireshark یک ابزار تجزیه و تحلیل شــبکه همه منظوره برای همه نیازها اســت، ابزار تجزیه و تحلیل شبکه دیگری، Network Miner، از دیدگاه فارنزیک و بدافزار مفیدتر و راحت تر است.

نکته: در زمان نگارش، Wiresharkرا می توان از لینک زیر دانلود کرد:

https://www.netresec.com/?page=NetworkMiner

نمایش تمام جزئیات جمع‌آوری شده درباره میزبانان در یک رابط کاربری کاربرپسند مناسب است. فایل‌ها، اعتبارات و غیره که در ترافیک شبکه منتقل می‌شوند، می‌توانند در تب‌های مختلف لیست شوند. همچنین، لیست‌های جداگانه‌ای از پرس‌وجوهای DNS و جلسات وجود دارد که همه آن‌ها می‌توانند بر اساس نیاز فیلتر شوند.



شکل ۴۵: رابط جستجوگر شبکه در Wireshark

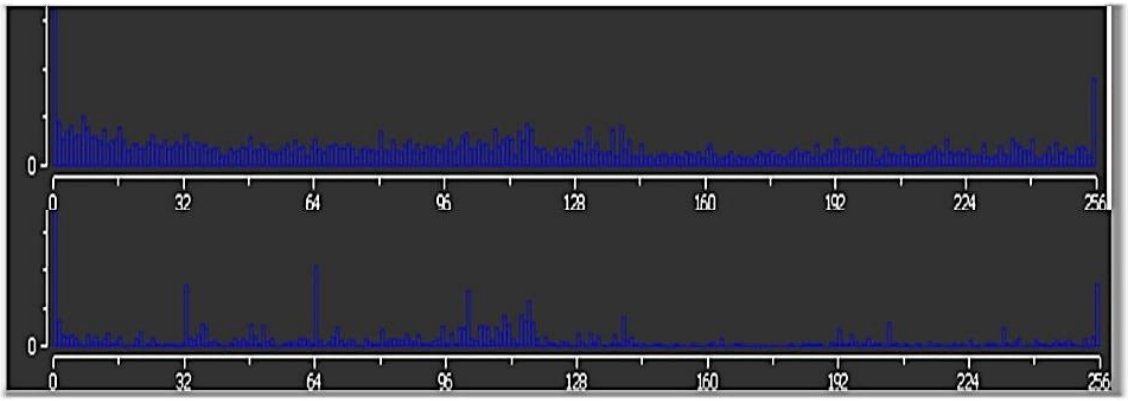
۷- Packed executables/unpacking

بدافزارها بسیار اغلب توسط نویسندگان بسته‌بندی می‌شوند تا از تشخیص آن‌ها توسط آنتی‌ویروس‌ها و بررسی توسط متخصصان معکوس‌سازی جلوگیری شود. این بسته‌بندی به وسیله ابزارهای بسته‌بندی نرم‌افزار استاندارد (مانند UPX، EXEStealth، ASProtect، FastPack، EXELock) یا بسته‌بندی‌های سفارشی انجام می‌شود. هر دوی این روش‌ها به طور کلی قادر به فشرده‌سازی، رمزگذاری و رمزگشایی نرم‌افزارهای مخرب اصلی هستند. یک بسته‌بندی‌کننده نرم‌افزار اجرایی اصلی را رمزگذاری کرده و آن را به عنوان داده خام در یک فایل اجرایی جدید ذخیره می‌کند که شامل کد رمزگشایی است. اگر فایل جدید اجرا شود، کد اصلی در حافظه رمزگشایی و اجرا می‌شود.

Detection ۱-۱-۷

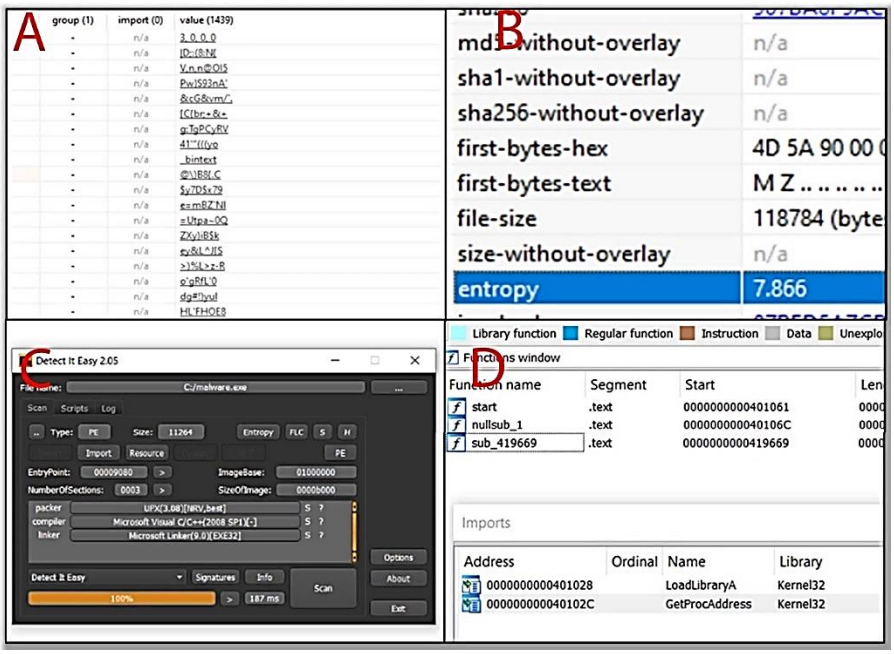
برای تشـــخیص اینکه یک فایل اجرایی بســـته‌بندی شـــده اســـت، چندین روش وجود دارد: فایل‌های اجرایی بسته‌بندی شده شامل تعداد کمی از رشته‌های معنادار، تعداد کمی از واردات و توابع و همچنین بی نظمی بالایی هستند. این به این دلیل است که کد باز کننده تنها بخش خوانا است (کد کوتاه به معنای تعداد کمی از رشته‌ها و کمبود نیاز به واردات یا توابع است) و بخش داده (شامل فایل اجرایی اصلی) رمزگذاری شده است، که به این معنی است که هیچ رشته، واردات یا توابعی وجود ندارد و بی نظمی بالایی دارد.

شکل زیر دو نمودار نشان می‌دهد که تعداد بایت‌های خاص در یک فایل اجرایی بسته‌بندی شده (بالا) و یک فایل اجرایی بازبسته‌بندی شده (پایین) را نشان می‌دهد. تفاوت مهم این است که فایل اجرایی بسته‌بندی شده توزیع یکنواختی از مقادیر بایت دارد، در مقابل فایل اجرایی بازبسته‌بندی شده که شامل چندین قله است که توسط دستورات پراستفاده(MOV، PUSH، CALL و غیره) ایجاد شده‌اند.



شکل ۴۶: هیستوگرام بایت -فایل اجرای بسته بندی شده (بالا) در مقابل. غیر بسته بندی شده (پایین)

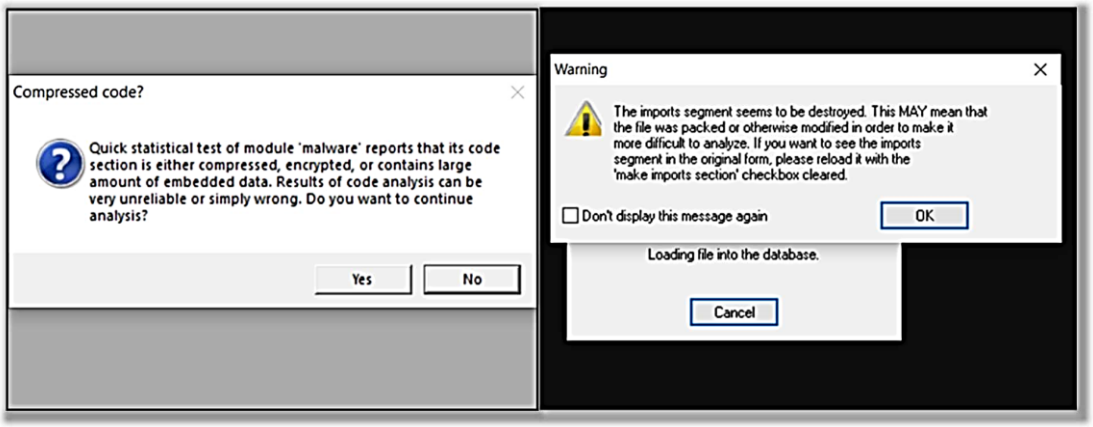
چندین ابزار برای تشخیص یک فایل اجرایی بسته‌بندی شده وجود دارد PeStudio، رشته‌های بی‌معنی را نشان می‌دهد و در صورت وجود فایل‌های اجرایی بسته‌بندی شده یا رمزگذاری شده، آنتروپی بالایی را محاسبه می‌کند؛Detect It Easy ، نوع بسته‌بندی کننده را شناسایی می‌کند (بر اساس پایگاه داده بسته‌بندی‌کننده‌های شناخته شده)؛ لیست توابع و واردات بسیار ضعیف هستند وقتی که فایل‌های اجرایی بسته‌بندی شده را در IDA بازبسته‌بندی می‌کنند.



شکل ۴۷: ویژگی های فایل اجرایی بسته بندی شده (A – رشته ها در PESTUDIO، B - ENTROPY در PESTUDIO، C – تشخیص آسان، D – توابع و واردات در IDA)

IDA Disassembler و OllyDbg Debugger می‌توانند فایل‌های اجرایی بسته‌بندی شده یا بخش‌های خاص آن‌ها را شناسایی کنند. این ابزارها در صورت باز شدن یک فایل اجرایی بسته‌بندی شده، در هنگام پردازش تجزیه خودکار ابتدایی، یافته‌های خود را اعلام می‌کنند. با این حال، تحلیل بیشتر همچنان ممکن است اما نتایج بسیار نامعتبر هستند.

https://github.com/horsicq/DIE-engine/releases



شکل ۴۸: IDA (سمت چپ) و OLLYDBG (سمت راست) به موارد اجرایی بسته بندی شده اشاره می کنند

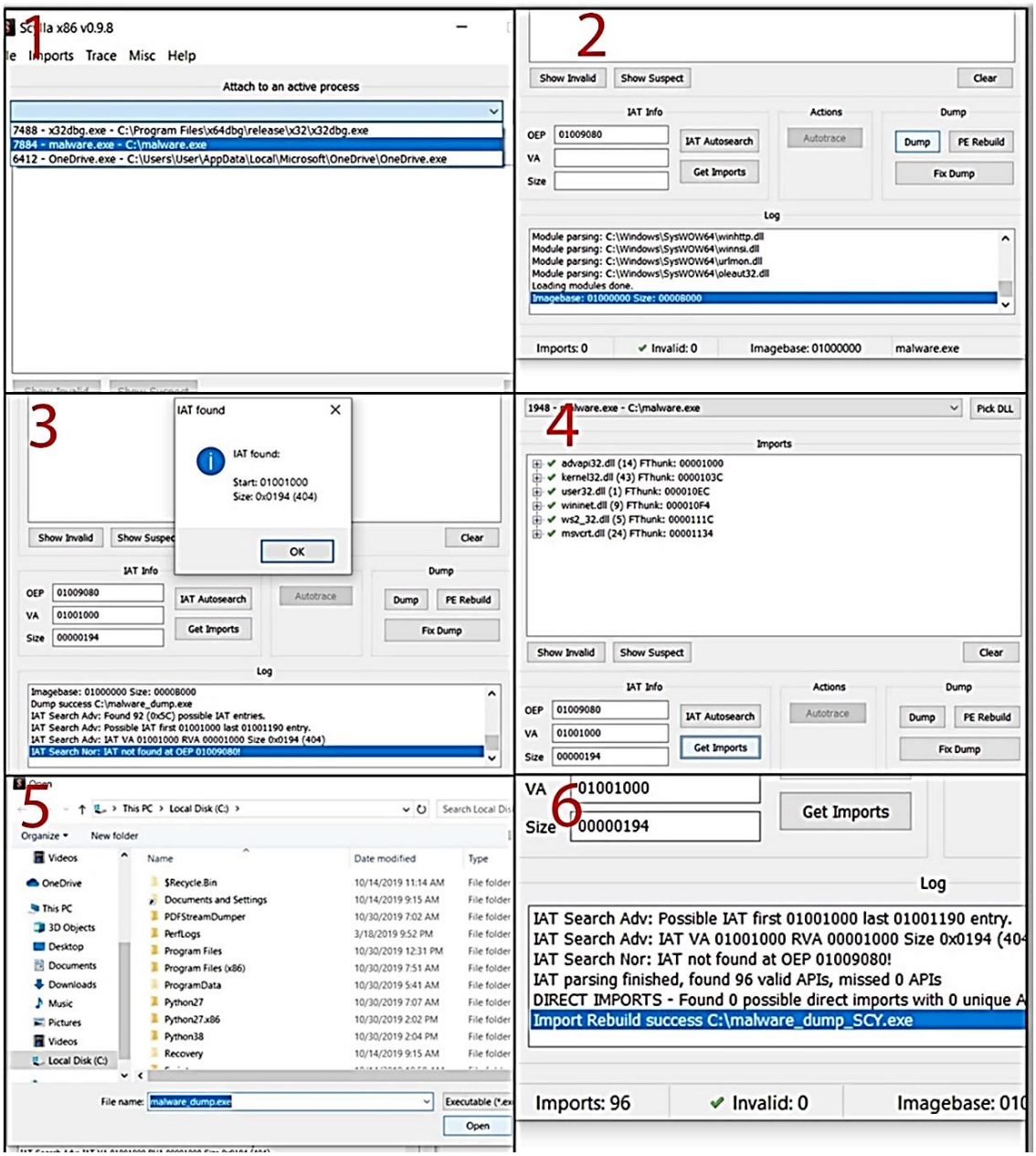
Unpacking ۲-۱-۷

اگر یک فایل اجرایی با یک بسته‌بندی استاندارد شناخته شده شده باشد، احتمالاً یک بازبسته‌بندی کننده عملکردی موجود است، یا یک بازبسته‌بندی رسمی (مانند UPX packer/unpacker) یا یکی توسط تحلیلگران بدافزار یا یک راه‌حل توسعه داده شده توسط جامعه.

برای الگوریتم‌های بسته‌بندی سفارشی ناشناخته، رویکرد متفاوتی لازم است. یک روش چندمنظوره، دامپ کردن کد بازبسته‌بندی شده از حافظه پس از اجرای فایل اجرایی بسته‌بندی شده است و چندین ابزار برای این منظور وجود دارد (ابزارهایPE، Scylla، OllyDumpEx/OllyDump و غیره) .

مراحل بازبسته‌بندی فایل‌های اجرایی با استفاده از Scylla به شرح زیر است:

۱- اجرای فایل اجرایی بسته‌بندی شده.
۲- باز کردن Scylla و اتصال آن به فرآیند فایل اجرایی (کد در این مرحله بازبسته‌بندی می‌شود).
۳- کلیک بر روی "Dump" و ذخیره فایل اجرایی بازبسته‌بندی شده جدید Scylla) پنجره ذخیره فایل جدید را باز می‌کند. (در طول عملیات دامپ، برخی اطلاعات مهم مانند نقطه ورود و جدول آدرس ورودی (IAT) از دست می‌روند.
۴- برای شناسایی IAT از فرآیند متصل شده، بر روی "IAT Autosearch" کلیک کنید.
۵- برای استخراج IAT از فرآیند بر روی "Get Imports" کلیک کنید. گاهی اوقات Scylla با مشکل استخراج تمامی ورودی‌های IAT مواجه می‌شود. اگر این اتفاق رخ داد و Scylla نتوانست برخی ورودی‌ها را استخراج کند (که با یک علامت صلیب قرمز به جای یک علامت تیک سبز نشان داده می‌شود)، این موضوع بر روی تحلیل بعدی تأثیری ندارد، زیرا ممکن است ورودی‌های ناموفق را از لیست حذف کرده و به مراحل بعدی ادامه دهید. اگر تعداد ورودی‌های استخراج نشده بالا باشد، بهتر است کل روند را از ابتدا تکرار کنید (یعنی هر دو Scylla و فرآیند اجرایی را خاتمه دهید و فایل دامپ شده از مرحله ۳ را حذف کنید).
۶- بر روی "Fix Dump" کلیک کرده و فایل دامپ شده از مرحله ۳ را انتخاب کنید.
۷- Scyllaیک فایل جدید با نام مشابه فایل دامپ شده با پسوند "_SCY.exe" ایجاد می‌کند.



شکل ۴۹: باز کردن بسته بندی با Scylla

فایل اجرایی بازبسته‌بندی شده با IAT صحیح، برای تحلیل استاتیک آماده است – کد، رشته‌ها، توابع و واردات قابل مشاهده هستند. Scylla گاهی اوقات نمی‌تواند نقطه ورودی صحیح را استخراج کند که مانعی برای تحلیل پویا بیشتر است. نقطه ورودی اصلی صحیح باید با دیباگ کردن فایل اجرایی بسته‌بندی شده و ثابت در هدرPE فایل اجرایی بازبسته‌بندی شده شود.

۸- Incident response collaboration (Misp & Yara)

قوانین YARA بر اساس الگوهای متنی یا باینری ایجاد شده‌اند. هر قانون شامل مجموعه‌ای از رشته‌ها و یک عبارت بولی است که منطق آن را تعیین می‌کند. به طور کلی، هر قانون YARA دارای دو بخش است: توصیف رشته‌ها و شرطی است. در حالی که بخش حاوی توصیف رشته‌ها در برخی موارد می‌تواند حذف شود، بخشی که شرایط اعلام می‌شود، اجباری است".

یک مثال از یک قانون اساسی YARA در زیر ارائه شده است:

```
rule FirstYaraRule
}
strings:
$text_string = 'malwaredomaine.com"
$hex_string = { A2 24 ?? D8 23 FB }
condition:
$text_string or $hex_string
{
```

در مثالی که در سمت چپ ارائه شده است، تمام فایل‌های باینری که رشته متنی "malwaredomaine.com" یا رشته شانزده‌گانه "A2 24 ?? D8 23 FB" درون فایل دارند، قانون Yara با نام "FirstYaraRule" را فعال می‌کنند. علامت سوال داخل رشته شانزده‌گانه، نشان دهنده کاراکترهای وایلدکارد (بایت‌هایی که نامعلوم هستند و ممکن است با هر چیزی مطابقت داشته باشند) است.

اگر یکی از رشته‌ها (رشته متنی یا رشته شانزده‌گانه) حداقل یک مطابقت با فایل‌های اسکن شده داشته باشد، قانون Yara فعال می‌شود.

برای انجام اسکن قوانین Yara ، محقق باید مجموعه قوانینی که می‌خواهد استفاده کند و هدفی که قرار است اسکن شود (می‌تواند یک فایل، پوشه یا فرآیند در حال اجرا باشد) را داشته باشد. از آنجا که این کتابچه فقط بر روی بدافزارهایی که در سیستم عامل ویندوز اجرا می‌شوند تمرکز دارد، فایل اجرایی که برای انجام اسکن می‌تواند استفاده شود، می‌تواند از این صفحه وب دانلود شود:

https://github.com/virustotal/yara/releases/tag/v4.0.0

سینتکس مورد استفاده در هنگام انجام اسکن به شرح زیر است:

yara [OPTIONS] RULES_FILE TARGET

کل لیست با تمام پارامترهای موجود که می توانند در طول اسکن استفاده شوند در این صفحه وب موجود است :

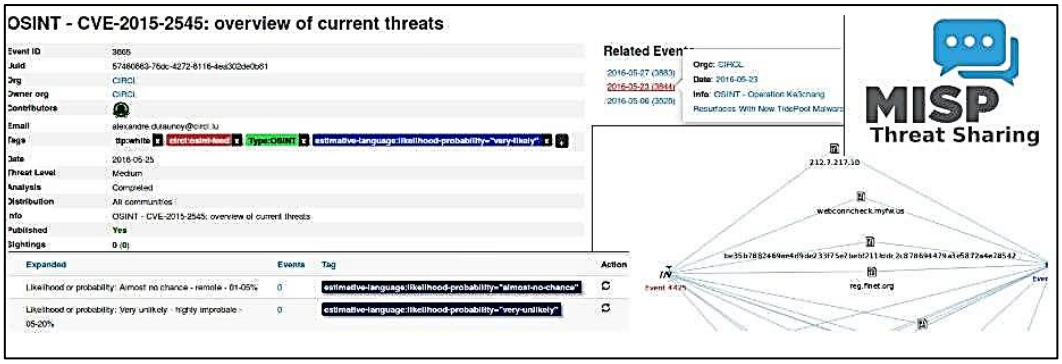
https://yara.readthedocs.io/en/v3.4.0/commandline.html

علاوه بر ایجاد مجموعه ای از قوانین یارا، یک تحلیلگر می تواند یکی از منابع قوانین Yara زیر را نیز مورد استفاده قرار دهد:

- Florian Roth repository:
  - https://github.com/Neo23x0/signature-base/tree/master/yara
- Yara Rules group GNU-GPLv2:
  - https://github.com/Yara-Rules/rules
- Github repository:
  - https://github.com/InQuest/awesome-yara

تمام یافته‌ها، از جمله قوانین Yara کامپایل شده، می‌توانند بر روی پلتفرم MISP (Malware Information Sharing Platform) آپلود، استفاده و سپس به اشتراک گذاشته شوند.

پلتفرم MISP (Malware Information Sharing Platform) یک پلتفرم اطلاعات تهدید منبع باز است که توسط سازمان‌های مختلفی که چندین نمونه MISP برای به اشتراک گذاری IoC اجرا می‌کنند، استفاده می‌شود. محقق می‌تواند تمام نشانگرها را به نمونه‌های MISP خود اضافه کند و بر اساس داده‌های ذخیره شده در پایگاه داده از رویدادهای مرتبط استفاده کند. تصویر زیر یک رویداد را نشان می‌دهد که بر اساس ویژگی‌های آن، پلتفرم MISP با رویدادهای دیگری که قبل از این حادثه در پایگاه داده بودند، همبستگی برقرار کرده است.



شکل ۵۰: صفحه وب MISP

امکان به اشتراک گذاری اطلاعات از طریق پلتفرم MISP بسیار مهم است، زیرا این امکان را برای تحقیقات همکارانه فراهم می کند و از تحلیل همان نمونه که شخص دیگری قبل از شما تحلیل کرده است، جلوگیری می کند.

اطلاعات بیشتر در مورد پلتفرم MISP را می توانید در وب سایت زیر بیابید:

https://www.misp-project.org/index.html

تصویر از وب سایت:Misp

https://www.misp-project.org/index.html

۹- Conclusion

این کتابچه شامل بسیاری از ابزارها و کاربردهای ضروری آنها است. در نظر داشته باشید که هدف آن نشان دادن تمام ویژگی‌های هر ابزار یا تمام مواردی که ممکن است از آنها استفاده شود نیست. برخی از ابزارها قابلیت‌های بسیار مشابه یا همپوشانی دارند. بر عهده خواننده است که ارزیابی کند کدام ابزار برای انجام یک کار تحلیلی خاص مناسب‌تر است. همچنین، راهکارهای دیگری نیز وجود دارند که در این کتابچه ذکر نشده‌اند.

تجزیه و تحلیل کد مجموعه دستوری استاتیک، فرآیندی بسیار زمان‌بر است. برای افزایش کارایی، توصیه می‌شود که با تجزیه و تحلیل کد پویا با استفاده از دیباگرها ترکیب شود. بهتر است با تجزیه و تحلیل استاتیک و رفتاری پایه شروع کرده و سپس با استفاده از دانشی که در دو مرحله اول به دست آورده‌ایم، به تجزیه و تحلیل کد ترکیبی (استاتیک و پویا) ادامه دهیم. هنگام انجام مهندسی معکوس کد، برای جلوگیری از نقض امنیت یا حوادث، مهم است که تحلیل‌گر یک محیط آزمایشی راه‌اندازی کند که جدا از شبکه شرکت فیزیکی باشد.

نتایج تحلیل بدافزار(IoCs) می‌توانند به عنوان ورودی برای تحقیقات جرم شناسی بیشتر درباره حوادث امنیتی جاری، و همچنین به عنوان ورودی برای نظارت امنیتی(Firewall، شبکه یا میزبان IDS/IPS، SIEMو غیره) برای جلوگیری از وقوع حملات مشابه در آینده استفاده شوند.

- Hex Rays SA. 2020. IDA Pro - Hex Rays. [https://www.hex-rays.com/products/ida/].Accessed May 2020.
- Hex Rays SA. 2020. F.L.I.R.T. - Hex Rays. [https://www.hex-rays.com/products/ida/tech/flirt/]. Accessed May 2020.
- Microsoft. 2020. InternetOpenA function (wininet.h) - Win32 apps | Microsoft Docs. [https://docs.microsoft.com/en-us/windows/win32/api/wininet/nf-wininet-internetopena]. Accessed May 2020.
- Hex Rays SA. 2020. IDA Technology: Open Plug-In Architecture - Hex Rays. [https://www.hex-rays.com/products/ida/tech/plugin/]. Accessed May 2020.
- National Security Agency. 2020. Ghidra. [https://ghidra-sre.org/]. Accessed May 2020.
- Microsoft. 2020. Debugging Tools for Windows (WinDbg, KD, CDB, NTSD) - Windows drivers | Microsoft Docs. [https://docs.microsoft.com/en-us/windows-hardware/drivers/debugger/]. Accessed May 2020.
- x64dbg Community. 2020. x64dbg. [https://x64dbg.com/]. Accessed May 2020.
- Immunity Inc. 2020. Immunity Debugger. [https://www.immunityinc.com/products/debugger/index.html]. Accessed May 2020.
- Oleh Yuschuk. 2014. OllyDbg v1.10. [http://www.ollydbg.de]. Accessed May 2020.
- Microsoft. 2020. ShellExecuteExA function (shellapi.h) - Win32 apps | Microsoft Docs. [https://docs.microsoft.com/en-us/windows/win32/api/shellapi/nf-shellapi- shellexecuteexa]. Accessed May2020.
- NTInfo. 2020. Detect It Easy. [https://www.ntinfo.biz/index.html#detect_it_easy]. Accessed May 2020.
- FireEye Labs. Obfuscated String Solver. Github. [https://github.com/fireeye/flare-floss]. Accessed May 2020.
- Strings2. [https://github.com/glmcdona/strings2]. Accessed May 2020.
- Practical Binary Analysis. 2018. Dennis Andriesse. No Starch Press (December 18, 2018)
- Mastering Malware Analysis. 2019. Alexey Kleymenov. Packt Publishing; 1 edition (June 6, 2019)
- Procmon [https://docs.microsoft.com/en- s/sysinternals/downloads/procmon]. Accessed May 2020.
- Process Monitor for Dynamic Malware Analysis. [https://docs.microsoft.com/en-us/archive/blogs/motiba/process-monitor-for-dynamic-malware-analysis]. Windows Sandbox Hari Pulapaka. [https://techcommunity.microsoft.com/t5/windows-kernel-internals/windows-sandbox/ba-p/301849]. Accessed May 2020.
- Practical Malware Analysis. 2012. Michael Sikorski and Andrew Honig. No Starch Press; 1 edition (February 1, 2012)
- Mastering Reverse Engineering – Re-engineer your ethical hacking skills. 2018. Reginald
- Wongs. Packt Publishing; 1 edition (October 31, 2018)
- Hands-On Network Forensics: Investigate Network Attacks and Find Evidence Using

- Common Network Forensic Tools. 2019. Nipun Jaswal. Packt Publishing; 1 edition (March 30 ,2019)
- Yaniv Assor. 2016. Anti-VM and Anti-Sandbox Explained. [https://www.cyberbit.com/blog/endpoint-security/anti-vm-and-anti-sandbox-explained/]. Accessed May 2020.
- Infosec Institute. 2016. How Malware Detects Virtualized Environment (and its Countermeasures). [https://resources.infosecinstitute.com/how-malware-detects- virtualized-environment-and-its-countermeasures-an-overview/]. Accessed May 2020