تاریخ دریافت مقاله: ۹۶/۰۲/۱۱
تاریخ پذیرش مقاله: ۹۶/۰۴/۰۹

بررسی مفاهیم باج‌افزارها و تحلیل کیفی روش‌های کشف نمونه

سمیرا ثنائی‌فر
کارشناسی ارشد نرم‌افزار، دانشکده مهندسی و علوم کامپیوتر، دانشگاه شهید بهشتی
پست الکترونیکی: samira.sanaeifar@gmail.com

علیرضا خلیلیان
دانشجوی دکتری نرم‌افزار، دانشکده مهندسی کامپیوتر، دانشگاه اصفهان
پست الکترونیکی: khalilian@eng.ui.ac.ir

مجتبی وحیدی اصل*
استادیار مهندسی نرم‌افزار، دانشکده مهندسی و علوم کامپیوتر، دانشگاه شهید بهشتی
پست الکترونیکی: mo_vahidi@sbu.ac.ir

چکیده

مسئله این است که تحقیقات حاضر برای کشف باج‌افزارها[1] بسیار معدودند و روش‌های کشف هم محدود و خاص‌منظوره هستند. از طرفی باج‌افزارها به‌تازگی رواج گسترده‌ای پیدا کرده‌اند و به‌شدت در حال رشد هستند. به‌همین دلیل، محققان نیاز به شناخت ماهیت این نوع بدافزار[2] و رفتار آن و عملکرد روش‌های موجود و کاستی‌های آن‌ها دارند تا بتوانند در تحقیقات بعدی روش‌های مؤثرتری طراحی نمایند. برای برطرف کردن این نیاز، مقاله حاضر با بررسی ۹ کار تحقیقاتی اخیر، زمینه تحقیقات آتی را فراهم می‌سازد. برای نیل به این هدف، تمام مفاهیم و دسته‌بندی‌های لازم در رابطه با باج‌افزارها تشریح شده‌اند و سپس هر یک از ۹ روش در ساختار سه بخشی «روش، آزمایش، نقاط قوت و ضعف» مورد بررسی قرار می‌گیرد. مطالعه‌ها نشان می‌دهند که چند عامل در طراحی روش‌های خوب مؤثرند: پایش عملیات روی فایل‌ها، پایش فراخوانی‌های سیستمی، شبکه‌ای و APIها و بررسی آنتروپی قبل و بعد از خواندن فایل‌ها.

کلید واژه‌ها: بدافزار، باج‌افزار، حمله، رمزگذاری، کشف بدافزار.

۱- مقدمه

دنیای بدافزارها هر روز گونه‌های جدیدی را تولید می‌کند که به شیوه‌های متفاوت سیستم‌های کاربران را مورد حمله و تخریب قرار دهند. یکی از بدافزارهایی که اخیراً رشد فزاینده‌ای پیدا کرده باج‌افزار است. باج‌افزار تهدیدی است در حال رشد که فایل‌های کاربر را رمزنگاری کرده و کلید رمزگشایی را تا پرداخت باج توسط قربانی نگه می‌دارد. این نوع بدافزار سالانه، وسیله‌ای برای ده‌ها میلیون دلار اخاذی است [۳]. طبق گزارش مک‌آفی، در سه ماه اول ۲۰۱۵، ۱۶۵٪ و در سه ماهه آخر سال ۲۰۱۵ حدود ۲۶٪ افزایش در نمونه‌های جدید باج‌افزار دیده شده است. بیشتر قربانیان که توسط مک‌آفی کشف شده‌اند در شمال آمریکا (۵۰٪) قرار گرفته‌اند؛ ۳۵٪ در اروپا و سهم آسیا

* نویسندهٔ مسئول

1- Ransomware
2- Malware

هفت درصد اســت [۲]. با توجه به رشد قابل توجه حملات باج‌افزار، نیاز امروز شــناخت ســاختار داخلــی و رفتار عملکردی این نوع بدافزارهاســت [۵]. شناختی که به‌دست می‌آید راهنمای طراحی فنونی برای کشــف باج‌افزارها یا مقابله با اثرات مخرب آن‌ها می‌شود. برای این منظور مقاله حاضر بر شــناخت باج‌افزارها و روش‌های کشــف آن‌ها تمرکز می‌کند.

متدولــوژی تحقیــق در این مقاله چیزی شــبیه مرور نظام‌یافته ادبیات است که پیرامون یک سؤال تحقیقی خاص تحقیق شــده اســت. در این مقاله به‌دنبال پاسخ این سؤال هســتیم که چگونه می‌توان روش مؤثری برای کشــف یا مقابله با باج‌افزارها طراحی کرد. اما تفاوت این مقاله با یک مقالــه مروری نظام‌یافته این اســت که فقط تعداد اندکی از کارهای اخیر مورد بررسی قرار گرفته است و از این جهت بسیار محدود است. اما باید توجه کرد که مقالات منتشره پیرامون باج‌افزارها در کل بســیار معدودند و این موضوع است که تحقیقات این مقاله و نتایج حاصل از آن‌را ارزشمند می‌ســازد. تا جایی که نویســندگان اطلاع دارند مقاله‌ای با ساختار و محتوای کنونی در حوزه باج‌افزار اولین بار است که مورد توجه و تحقیق قرار می‌گیرد.

برای این‌که بتوانیم پاســخ سؤال تحقیق را بیابیم، ابتدا باید ســاختار داخلی باج‌افزارها و شــیوه عملکرد آن‌ها را بشناسیم. این موضوع شامل گونه‌های مختلف باج‌افزارها، شیوه‌های مختلف آلوده‌ســازی، تخریب و حمله می‌شود. همچنین باید دانشــی از رویکردهای موجود برای کشــف بدافزارها به‌دســت آوریم. لازم است دقیقاً مشخص شود که باج‌افزارها چه ویژگی‌های متمایزی از ســایر بدافزارها در ســاختار کد یا رفتار دارنــد. همچنین باید نقاط ضعف این بدافزارها شناســایی گردد. ســپس لازم است بهترین مطالعه‌های اخیر در کشــف باج‌افزارها مورد بررسی قرار گیرد. این مطالعه ضمن فراهم‌ســازی دیدگاهی از رویکرد کشــف باج‌افزارها و شــناخت دقیق از عملکرد روش‌های موجود، امکان کشف نقاط ضعف و نقاط توسعه روش‌های موجود را فراهم می‌ســازد. برای نیل به هدف مذکور، باید مطالعه‌های مورد بررسی ارزیابی و مقایسه شوند. در نهایت با تحلیل فنون و مطالعه‌ها می‌توان سمت و سوی تحقیقات آتی و شــیوه طراحی روش‌های مؤثر کشف باج‌افزارها را به‌دست آورد تا سؤال تحقیق پاسخ داده شود. پژوهشگران و کاربران در محیط‌های واقعی و صنعتی از محتوا و نتایج حاصل از این مقاله چنین می‌توانند بهره ببرند:

- کمک به شــناخت بهتر باج‌افزارها و روش‌های کشف موجود و طراحی روش‌های جدید مؤثرتر
- اســتفاده از روش‌ها و ابزارهای موجــود در کاربرد مناســب و مناسب‌ســازی و ویژه‌ســازی آن‌ها برای یک کاربرد اختصاصی

نوآوری‌های این مقاله عبارتند از:

- بررســی باج‌افزار و مفاهیم مرتبط و روش‌های مؤثر کشف
- معرفی چند دسته‌بندی برای ارزیابی روش‌های کشف باج‌افــزار و ایجاد چارچوب پایه بــرای قرار دادن کارهای بعدی
- ارائه تحلیلی از کارکرد و سودمندی روش‌های موجود
- برجسته کردن مشکلات موجود و سیر تحقیقات آتی

ساختار ادامه به این شرح است: در بخش دوم باج‌افزار و مفاهیــم مربوط بــه آن و روش‌های مربوط به کشــف بدافزارها ارائه می‌شــود. بخش سوم به تشریح تفصیلی ۹ مورد از مطالعه‌های اخیر اختصاص دارد. هر کار تحقیقی در قالب ســه بخش روش، آزمایش‌ها و نتایج و نقاط قوت و ضعف بررسی می‌گردد. در بخش چهارم، روش‌ها با هم مقایسه می‌شــوند و مشکلات موجود تشریح شده و سیر تحقیقات آتی باز می‌شــود. بخش پنجم هم با نتیجه‌گیری مقاله را پایان می‌دهد.

۲- مفاهیم ضروری

ایــن بخش مفاهیم ضروری را مــرور می‌کند که برای مطالعه بخش سوم لازم هســتند. بدافزار معرفی می‌شود

و باج‌افزار تشـــریح می‌گردد. ســـپس رویکردهای کشـــف بدافزارها مرور می‌شوند.

۲-۱ بدافزار و باج‌افزار

بدافـــزار نرم‌افـــزار مخربی اســـت که به‌طـــور عمدی رفتارهای بدخواهانه‌ای از خود نشان می‌دهد [۱۲]. تاکنون بدافزارها به شـــکل‌هایی همچون ویـــروس، کرم، زامبی و بمب‌های منطقی ظاهر شده‌اند. نسل جدیدتر بدافزارها شامل «باج‌افزار»، «آگهی‌افزار»[۳] و «جاسوس‌افزار»[۴] هستند. مفهوم باج‌افزار جدید نیســـت، (بعضی از حملات حداقل مربوط به دهه ۱۹۸۰ اســـت) ولی در چند ســـال اخیر فعالیت‌هایش به‌شدت گسترده‌تر شده است. عملیات بدخواهانه باج‌افزار معمولاً قفل کردن صفحه رومیزی کاربر به منظور غیرقابل دسترس ســـاختن سیســـتم یا رمزنگاری، رونویسی و یا حذف فایل‌های کاربران است.

باج‌افـــزار بـــه رده‌ای از بدافزارها اطلاق می‌شـــود که قربانیـــان خود را به‌طـــور دیجیتالـــی وادار می‌کند هزینه مشخصی را به‌نام باج بپردازند [۱۷]. باج‌افزارها را در یک دســـته‌بندی می‌توان به «پربقاء»[۵] و «کم‌بقاء»[۶] تقسیم نمود [۱۶]: باج‌افـــزار خاصیت پربقاء دارد اگـــر کنترل یک منبع میزبانی حیاتی را آن‌چنان نگه دارد که دسترســـی به آن‌را تنها در هنگام نیاز فراهم می‌سازد و اگر باج‌افزار تغییر پیدا کند یا حذف شود، آن منبع به‌طور دائمی غیرقابل دسترس می‌گردد و فرایند رمزگشایی را تنها از طریق کلید کارساز[۷] کنتـــرل و فرمان بعـــد از پرداخت باج می‌تـــوان انجام داد. باج‌افزار کم بقاء چنین ویژگی ماندگاری ندارد.

در دســـته‌بندی دیگـــر باج‌افزارهـــا را بـــه دو دســـته «غیررمزنـــگاری»[۸] و «رمزنگاری»[۹] تقســـیم می‌کنند [۱۶]: دســـته اول باج‌افزارها صرفاً ســـعی می‌کننـــد «تعامل» با سیستم را محدود کنند. برای این منظور ممکن است صفحه نمایش را قفل نمایند یا تغییراتی در جدول فایل‌های سیستم انجام دهند. عملیات تخریبی این‌گونه باج‌افزارها ضعیف و ســـاده است و به‌همین دلیل به‌ســـادگی بدون پرداخت باج قابل برگشت اســـت. باج‌افزارهای رمزنگاری با استفاده از الگوریتم‌های رمزنگاری فایل‌ها و داده‌های سیستم قربانی را «غیرقابـــل دســـترس» می‌کنند و تا باج پرداخت نشـــود امکان رمزگشایی را فراهم نمی‌سازند. مثلاً باج‌افزار ممکن است پس از ورود به سیستم قربانی شروع به رمزنگاری کند و پـــس از اتمام عملیات، با پیغامی به کاربر اطلاع دهد که داده‌ها رمز شده‌اند.

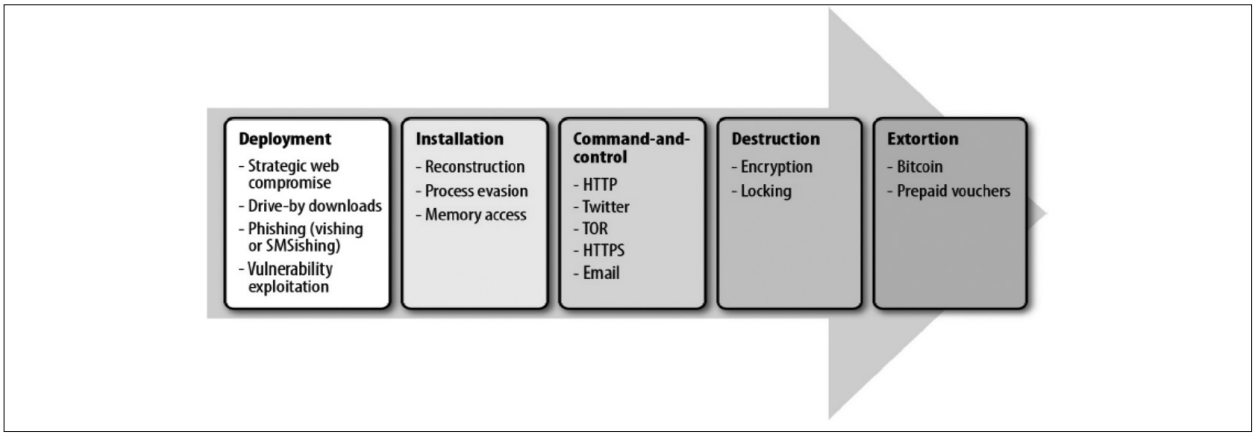
بـــا توجه به نوع سیســـتم رمزنـــگار مورد اســـتفاده، باج‌افزارهای رمزنگار به سه دسته کلید «متقارن»[۱۰]، «کلید عمومی»[۱۱] و «ترکیبی» تقســـیم می‌شوند [۱۶]. حمله‌ای که باج‌افزار رمزنگار متقارن انجام می‌دهد، قابل تعمیر اســـت زیرا عملیات بدافزار از نگاه بدافزارنویس و تحلیلگر بدافزار (شـــکننده) متقارن اســـت. پس با پایش رفتـــار باج‌افزار و مهندسی معکوس یا «جســـتجوی فراگیر»[۱۲] اغلب می‌توان حمله را خنثی کرد یا به کلید دسترسی پیدا کرد.

امـــا در مورد باج‌افزار کلید عمومی، چون رمزگشـــایی با کلید خصوصـــی صورت می‌گیرد، چنین ســـهولتی در خنثی‌ســـازی حمله و یافتـــن کلید وجود نـــدارد. باج‌افزار رمزنـــگار کلید عمومـــی دو ایراد دارد: یکـــی کندتر بودن محاسبات است؛ دیگری این‌که اگر کلید خصوصی را برای یکی از قربانیان فاش کند، ایـــن قربانی کلید خصوصی را منتشر می‌نماید و سایر قربانیان نیز بهره‌مند خواهند شد.

روش ترکیبی با هدف حل مشکلات دو روش متقارن و کلید عمومی طراحی شده است. کلید عمومی و خصوصی تولید می‌شـــود و کلیـــد عمومی داخل باج‌افزار جاســـازی می‌شـــود. اما برای هر قربانی یک کلیـــد متقارن تصادفی ساخته می‌شود و داده‌های سیستم قربانی با آن رمزگذاری می‌گردد. سپس این کلید با کلید عمومی رمزگذاری می‌شود. برای رمزگشـــایی، سیستم قربانی کلید متقارن رمزگذاری

3- Adware
4- Spyware
5- High Survivable Ransomware (HSR)
6- Low Survivable Ransomware (LSR)
7- Server
8- Non-cryptographic
9- Cryptographic

10- Symmetric key
11- Public key
12- Brute-force



شکل ۱: پیکره عمومی حمله باج‌افزار [۱۷]

شـــده را به مهاجم می‌فرســـتد، او با کلیـــد خصوصی باز می‌کند و پس می‌فرستد تا داده‌های رمزگذاری شده سیستم قربانی رمزگشایی شوند.

فعالیت‌های اصلی که باج‌افـــزار انجام می‌دهد می‌تواند به سه رده تقسیم شـــود[۳] : «رونویسی درجا»، «انتقال و رونویسی»، «رمزگذاری مســـتقل». الف) باج‌افزارهایی که به‌صورت درجا محتوای رمزنگاری را رونویسی می‌کنند و ممکن است به‌صورت اختیاری نام فایل را تغییر دهند. ب) این رده گسترش یافته رده (الف) است که فایل را به خارج از فهرست اسناد کاربر (به‌طور مثال به یک فهرست موقت) انتقال می‌دهند، ســـپس محتـــوای آن‌را خوانـــده، محتوای رمزنگاری شـــده را نوشـــته و فایل را به فهرســـت کاربر منتقل می‌کنند. ممکن است زمانی که فایل به فهرست اصلی برگردانده می‌شـــود نام متفاوتی بگیـــرد. از آنجایی که نام فایل مقصد ممکن اســـت با نام اصلی آن همخوانی نداشته باشـــد، وضعیت فایل باید به دقت هر زمانی که فایل منتقل می‌شود، بررسی شود. پ) باج‌افزار فایل اصلی را خوانده، سپس یک فایل جدید و مستقل که شامل محتوای رمز شده است، ایجاد کرده و فایل اصلی را حذف کرده یا فایل اصلی را رونویسی می‌کند. این رده از دو جریان دسترسی برای خواندن و نوشتن داده‌ها استفاده می‌کند.

حملـــه موفقیت‌آمیز باج‌افزار شـــامل یک یا چند فعالیت اســـت: بعد از انجام موفقیت‌آمیز آلودگی باج‌افزار، برنامه مخرب معمولاً پیغامی به قربانی نشان می‌دهد. این «پیغام باج»، کاربر را از قفل شدن سیستم و دستورالعمل چگونگی پرداخت باج برای دسترســـی مجدد، آگاه می‌کند. این پیغام می‌تواند از طرق مختلف ایجاد شود. یک فن رایج، فراخوانی توابع API اختصاص یافته برای ایجاد یک صفحه رومیزی جدیـــد و ایجاد آن به‌صورت پیکربنـــدی پیش فرض برای قفل سیســـتم قربانی است. نویسندگان بدافزار می‌توانند از HTML یا شکل‌های دیگری از پنجره‌های دائمی برای نشان دادن پیغام استفاده کنند. نمایش یک پیغام دائمی یک فعالیت قدیمی در بســـیاری از حمـــلات باج‌افزار اســـت. کلیدهای رمزنـــگاری می‌توانند به‌صـــورت محلی توســـط بدافزار روی سیســـتم قربانی یا به صورت از راه دور روی رایانه کارســـازهای کنترل و فرمان ایجاد شده و سپس به رایانه در خطر، تحویل داده شـــوند. یک مهاجم می‌تواند از توابع مخرب سفارشـــی یا APIهای ویندوز برای حذف فایل‌های اصلی کاربر استفاده کند [۴].

همچنیـــن مهاجـــم می‌توانـــد فایل‌هـــا را «بـــا نســـخه رمزگذاری‌شده رونویسی» کرده یا از حذف مطمئن از طریق واسط «برنامه کاربردی[۱۳] کشف امن ویندوز» استفاده کند. رمزنـــگاری انتخابی و حذف فایل‌هـــای خصوصی کاربر بر اســـاس خصوصیات مشـــخصی مانند انـــدازه، تاریخ دسترســـی و پسوند انجام می‌شـــود. به منظور جلوگیری از شناسایی، شمار زیادی از نمونه‌های باج‌افزار فایل‌های خصوصی کاربر را به صـــورت انتخابی رمز می‌کنند. در

13-API

ساده‌ترین شکل، باج‌افزار می‌تواند فایل‌ها را بر اساس تاریخ دسترسی فهرست کند. در سناریوهای پیچیده‌تر، بدافزار می‌تواند یک نرم افزار مانند word.exe را باز کرده و فایل‌هایی را که اخیرا مورد دستیابی قرا گرفته‌اند فهرست کند. همچنین باج‌افزار می‌تواند کد آلوده را به هر برنامه ویندوز برای به‌دست آوردن این نوع اطلاعات (مانند فرایند خواندن مستقیم حافظه) وارد کند [۴].

در مقایسه با بدافزارهای سنتی، باج‌افزارها رفتارهای متفاوتی از خود نشان می‌دهند. برای مثال، بدافزار سنتی معمولاً به‌صورت پنهانی کار خود را انجام می‌دهد بنابراین می‌تواند اطلاعات بانکی یا کلیدهای فشرده شده را بدون افزایش سوء ظن جمع‌آوری کند. در مقابل، رفتار باج‌افزار در تضاد با کار پنهانی است و تمام هدف حمله، آگاه ساختن کاربر از آلوده شدن سیستم او است [۴]. پیکره کلی حمله باج‌افزار را می‌توان در شکل ۱ مشاهده کرد.

۲-۲ کشف بدافزار

تعیین این‌که یک برنامه مفروض ویروس (بدافزار) است یا خیر، یک مسئله غیرقابل تصمیم‌گیری است [۸, ۹]. کشف بدافزارها را از جنبه‌های مختلف می‌توان تقسیم‌بندی کرد: الف) از نظر وجود اطلاعات قبلی به دو دسته «شناخته شده»[۱۴] و «شناخته نشده»[۱۵] [۱۵]؛ ب) از نظر نوع تحلیل به دو دسته «ایستا» و «پویا»؛ پ) از نظر رویکرد کشف به دو دسته «مبتنی بر امضاء»[۱۶] و «مبتنی بر ناهنجاری»[۱۷] [۱۰]؛ ت) از نظر فنون کشف به دسته‌های «وارسی کننده صحت»[۱۸]، «یادگیری ماشینی» همچون «داده کاوی»، «شبکه عصبی»، «مدل مخفی مارکوف»[۱۹] و «شناسایی مبتنی بر مشخصه»[۲۰].

باج‌افزار شناخته شده بدافزاری است که ساختار کد آن، عملیات بدخواهانه آن و حملات آن شناخته شده است و بنابراین به‌سادگی با دیدن نشانه‌های مشخصی، می‌توان آن را به‌طور قطعی شناسایی کرد. باج‌افزار شناخته نشده ساختار کد جدیدی دارد یا رفتار بدخواهانه و حمله آن جدید است. بنابراین کشف آن‌ها دشوار است و معمولاً از طریق پایش رفتارهای مشکوک می‌توان به‌طور تقریبی به وجود آن‌ها پی برد.

روش‌های ایستا بسیار محبوب و رایجند زیرا بدون اجرای بدافزار و با تحلیل کدش قادرند آن‌را کشف نمایند. روش‌های ایستا سربار محاسباتی کمتری دارند، کم خطرترند زیرا بدافزار را اجرا نمی‌کنند ولی به‌خاطر «تخمین‌های دست بالا»[۲۱]، ممکن است منجر به «مثبت کاذب»[۲۲] شوند. در مقابل روش‌های پویا با اجرای بدافزار می‌توانند آن‌ها را کشف کنند. «همانندسازی»[۲۳] و «پایش»[۲۴] نمونه‌هایی از روش‌های پویا هستند [۱۳]. از محدودیت‌های روش‌های پویا می‌توان سربار اجرایی بالا، بررسی یک یا چند مسیر اجرایی محدود و «منفی کاذب»[۲۵] به‌خاطر «تخمین‌های دست پایین»[۲۶] را نام برد.

برای جلوگیری از صدمه به سیستم کامپیوتری، برای اجرا و تحلیل فایل‌های مشکوک از روش «جعبه شنی»[۲۷] یا «انفجار فلز محض»[۲۸] استفاده می‌شود. در روش اول، سیستم مرزی ورودی و خروجی فایلی را دریافت کرده و بعد از پیمایش، آن را درون محیط مجازی برای اجرا قرار می‌دهد. این‌کار جعبه شنی یا محیط مجازی ایمنی می‌سازد که بدافزار احتمالی در آن اجرا می‌شود و رفتار بدخواهانه‌اش را بروز می‌دهد. مشکل اینجاست که گاهی بدافزارهای پیچیده با روشی می‌توانند از محیط مجازی اجرایی آگاه شود و کاری کند که شناسایی نشود؛ مثلاً اجرای خود را خاتمه داده یا کار بیهوده‌ای انجام دهد[۲۹]. برای حل این مشکل از روش انفجار فلز محض یا کنترل مستقیم روی سخت‌افزار استفاده می‌شود. در این روش

14- Known
15- Unknown
16- Signature-based
17- Anomaly-based
18- Machine learning
19- Hidden Markov Model (HMM)
20- Specification-based
21- Overapproximation
22- False positive
23- Emulation
24- Monitoring
25- False negative
26- Underapproximation
27- Sandbox
28- Bare-metal detonation
29- Outlast, outsmart

به‌جای محیـط مجازی از ماشـین فیزیکـی واقعی برای ارسال و اجرای فایل استفاده می‌شود. در این‌صورت نیاز به چندین کامپیوتر با سیسـتم‌های عامل و پیکربندی‌های گوناگون داریم.

در کشف بدافزار با امضاها، «پایگاه داده‌ای» از امضای بدافزارهـا وجود دارد و فایل مشـکوک در بانک امضاها جستجو می‌شود که معلوم شود با یکی از امضاها مطابقت می‌کنـد یا خیر. ضد بدافزارهای تجـاری معمولاً با روش مبتنـی بر امضاء کار می‌کنند، اما به‌واسـطه اسـتفاده از روش‌های «مبهم‌سازی» در ساخت ویروس‌ها، روش‌های مبتنی بر امضـاء دیگر برای شناسـایی بدافزارها کفایت نمی‌کنند. روش کشف مبتنی بر امضا کارآمد است ولی نیاز به تخصص و دانش انسانی دارد که پایگاه داده امضاها را تهیه نماید و بدافزار جدیدی که امضایش موجود نباشـد، قابل کشف نخواهد بود. به‌علاوه بزرگ شدن پایگاه امضاها خودش ممکن است باعث افت کارایی گردد.

مبهم‌سازی یا «فراریختی» یکی از روش‌هایی است که در سال‌های گذشـته توسط بدافزارهایی همچون ویروس و کرم به‌طور گسـترده مورد اسـتفاده قرار گرفته است. فراریختی روشـی است که سـاختار کد بدافزار را بدون تغییر رفتـار آن تغییر می‌دهد. به ایـن ترتیب، روش‌های مبتنی بر امضاء از کشـف بدافزار فراریخت شـده کاملاً ناتوان می‌شوند. فراریختی فنی است که روی انواع بدافزار از جملـه باج‌افزارها هم قابل اسـتفاده اسـت. خلیلیان و وزین‌دل [۲۰] با بررسـی ۴۰ مقاله شـاخص اخیر، به‌طور مبسوط فنون مبهم‌سـازی فراریختی و روش‌های کشف بدافزارهای فراریخت را مورد بررسی قرار داده‌اند. سپس با ارائه معیارهایی، آن‌ها را با هم مقایسـه کرده و سـیر تحقیقات آتی را برجسته ساخته‌اند.

به‌جـای شناسـایی بدافزار با جسـت‌وجوی امضاها، نرم‌افـزار ضـد بدافـزار می‌توانـد با کمـک روش‌های «ابتکاری»[۳۰]، رفتار غیرعادی و ناهنجار را شناسایی کند. این روش کشف بدافزارها دو مرحله دارد و شامل «یادگیری» و «شناسایی» اسـت. در مرحله یادگیری شناساگر سعی می‌کند رفتار عادی سیسـتم، برنامه یا هر دو را بشناسد. روش‌های ابتکاری کشـف ناهنجاری ممکن است ایستا یا پویا باشـند. روش‌های ابتکاری ایستا از تحلیل ایستای کد استفاده می‌کنند تا ساختارهای مشکوک را در فایل بیابند؛ مثل «حلقه رمزگشایی»، «کدهای خود تغییرده[۳۱]»، «استفاده از فراخوانی‌های API مستند نشده»، «تغییر بردار وقفه‌ها» و غیره [۱۲].

روش‌هـای ابتکاری پویا با تحلیل رفتـار فایل اجرایی هنگام اجرا، سـعی می‌کنند بفهمند آیـا فایل اجرایی آلوده شده اسـت یا خیر. از جمله محاسـن سیستم‌های کشف مبتنی بر ناهنجاری این‌است که قادرند در مقابل «حمله‌های جدید»[۳۲] مثل بدافزارهای شـناخته نشـده یـا گونه جدید بدافزار شـناخته شـده هم حفاظت نمایند. در مقابل هزینه پیاده‌سـازی آن بالاست و نرخ هشـدارهای مثبت و منفی کاذب در سیستم‌های کشف مبتنی بر ناهنجاری در مقایسه با سایر روش‌های کشـف معمولاً بالاتر است [۱۰]. ضمن این‌که تعییـن ویژگی‌هایی که حین مرحلـه یادگیری باید توسـط شناسـاگر مورد توجه قرار بگیرد، کار سـاده‌ای نیسـت. «شناسایی مبتنی بر مشخصه»[۳۳] یک زمینه خاص شناسـایی مبتنی بر ناهنجاری است که با کمک چند قانون یا مشـخصه در مـورد بدخواهی برنامه تحت بررسـی تصمیم‌گیری می‌نماید.

باج‌افزارها به شکل‌های بسیار متفاوتی کار می‌کنند که شـامل قفل کردن صفحه رومیزی سیستم عامل کاربر و رمزگذاری همه فایل‌های او می‌شـود [۴]. مسئله اینجاست که باج‌افزارها رفتارهای متفاوتی نسـبت بـه بدافزارهای معمول از خود نشـان می‌دهند. بـرای نمونه، بدافزارهای سـنتی سـعی می‌کنند خود را پنهان نگه دارنـد تا بتواند اطلاعاتی را از سیسـتم کاربر سـرقت نمایـد. اما رفتار باج‌افزارها کاملاً با مخفی کاری در تضاد است چون نقطه اصلی حمله اینست که باج‌افزار به قربانی خبر دهد که آلوده

30- Heuristic

31- Self-modifying
32- Zero-day
33- Specification-based

شـــده اســـت. به‌عنوان مثالی دیگر، یکی از روش‌های رایج برای کشف بدافزارهای معمول، تحلیل پویاست که بدافزار را در محیطـــی مجـــازی اجرا می‌کنند تا رفتـــار آن را ثبت کنند و بر اســـاس آن نوع فایل مشکوک را تشخیص دهند. اما سیستم‌های کشـــف بدافزار که با فرض رفتار مخفیانه بدافزار طراحی شده‌اند، نمی‌توانند موفق به کشف باج‌افزار در چنین محیطی شـــوند. علت اینست که رفتار باج‌افزارها شباهت بسیاری به فایل‌های سالمی دارد که برای کارهای عادی از رمزنگاری و فشرده‌سازی استفاده می‌کنند. نتیجه این‌که برای کشـــف باج‌افزارها نیاز به روش‌های کشـــف اختصاصی هستیم.

مبارزه با باج‌افزار به دلایلی دشوار است: الف) به‌دست آوردن و یا ایجاد این نوع بدافزار آســـان اســـت و موجب بـــازده فوری و ایجـــاد فرصت‌های ســـودآور برای حمله می‌شـــود؛ ب) تشخیص و تمایز عملیات انجام شده توسط باج‌افزار از نرم‌افزارهای بی‌خطر بســـیار دشوار است؛ پ) باج‌افزارها اغلـــب کاربرانی را مورد حمله قرار می‌دهند که از داده‌های خود به‌طور منظم پشتیبان‌گیری نمی‌کنند [۳].

یکـــی از ویژگی‌هـــای مؤثر بـــرای کشـــف باج‌افزارها «آنتروپی»[۳۴] فایل‌های دستکاری شده قبل و بعد از خواندن و نوشـــتن اســـت و باج‌افزارها تمایل دارنـــد فایل‌هایی که آنتروپی بالا دارند را مورد حمله قرار دهند که اختلاف بین آنتروپی خواندن و نوشتن کم است و شناسایی را دشوار می‌سازد [۴]. در نظریه اطلاعات، «آنتروپی» معیاریست از میزان اطلاعاتی که قبل از دریافت، از دست می‌رود و گاهی به آن آنتروپی شانون گفته می‌شود. در فایل‌ها و اطلاعات دیجیتال، اندازه تصادفی بـــودن داده و اندازه‌گیری مقدار داده‌هایـــی را که در یک فایل وجود دارد آنتروپی می‌نامند. آنتروپـــی داده‌هـــای تصادفی مشـــابه داده‌هـــای معمولی کاربـــران نیســـت و از این ویژگی می‌توان برای بررســـی این‌که فایل رمزگذاری یا فشرده شده یا خیر استفاده کرد. پس در حوزه کشـــف بدافزار از محاسبه آنتروپی می‌توان برای بررســـی این‌که آیا رمزگذاری شـــده و آیا توســـط نرم‌افزارهایی بسته‌بندی شده یا خیر استفاده کرد.

دو روش عمده دیگر برای کشـــف باج‌افزارها استفاده شده است: «بررسی عملیات روی فایل‌ها»[۳۵] شامل خواندن، نوشتن، بازکردن و بســـتن؛ «رشته و توالی فراخوانی‌های سیســـتمی، شبکه‌ای و واســـط برنامه کاربردی»[۳۶]. مسئله این است که این‌گونه اطلاعات باج‌افزارها شباهت بسیاری بـــا فایل‌های ســـالم دارند و نیاز به روش‌هایی هســـت که پالایش اساســـی و هوشـــمندانه‌ای انجام دهند تا داده‌های متمایزکننده‌ای به‌دست آید.

برای محاســـبه میزان سودمندی یک روش کشف رایج اســـت که روی تعدادی نمونه آزمایش می‌شـــود و مقادیر مثبـــت واقعـــی و کاذب و منفـــی واقعی و کاذب محاســـبه می‌شوند. سپس مقادیر «صحت»[۳۷]، «دقت»[۳۸]، «بازیابی»[۳۹]، «معیار اِف»[۴۰] (میانگین هم‌ســـاز دقـــت و صحت) و «مقدار ROC» محاسبه و مقایسه می‌شوند.

۳- مروری بر روش‌های موجود

این بخش شـــرح ۹ مـــورد از روش‌های جدید کشـــف باج‌افزارهاســـت. برای هر مقاله ابتدا روش پیشنهادی بیان می‌شود و سپس جزئیاتی از ارزیابی روش و نتایج به‌دست آمده گزارش می‌شود.

۳-۱ روش تحلیـــل و تشـــخیص خـــودکار باج‌افزار اندروید

روش: یانگ و همکاران [۶] در ســـال ۲۰۱۵ یک مفهوم تحلیل خودکار را که ترکیبی از تحلیل پویا و ایســـتا برای برنامه‌های موبایل است، پیشنهاد دادند. بیشتر برنامه‌های امنیتی به تحلیل مجوزها و مســـئله نفـــوذ اطلاعاتی برای یافتن برنامه‌های مشکوک می‌پردازند که برای روش ایستا مناسب هســـتند. اما اگر یک برنامه مشکوک در پس‌زمینه اجرا شـــود و داده‌های حساس به ســـرقت برود، مناسب

34- Entropy

35- File operations: Read, Write, Open, Close
36- System, network, and API calls
37- Accuracy
38- Precision
39- Recall
40- F-measure

نیســـتند. برای شناســـایی برنامه‌های مشـــکوک در حال اجرا از تحلیل بیدرنگ که تحلیل آلودگی نامیده می‌شـــود، استفاده شده است. مقاله مذکور به بررسی توابع باج‌افزار و رده‌هـــای اصلی آن و مجوزهای قفل کردن، از بین بردن فرایند، Boot receiver و پرداخت می‌پردازد.

در این مقاله مفهوم تحلیل خودکار که ترکیبی از تحلیل پویا و ایســـتا برای برنامه‌های موبایل است، پیشنهاد شده اســـت. کلید این مفهوم، یکی شدن حالت‌های داده و اجرای نرم افزار «مسیر آزمون[41] بحرانی» است. روش پیشنهادی دو مرحله دارد: مرحله اول شامل تحلیل ایستای آزمایشی است که ابتدا حمله احتمالی مبتنی بر واسط برنامه کاربردی اندروید و الگوی حمله‌های موجود را شناســـایی می‌کند و تحلیل پویا از مســـیر هدایت شـــده، برنامه را در یک دامنه محـــدود و متمرکز برای تشـــخیص احتمال حمله با تطابق مسیر شناسایی شده با الگوهای موجود، اجرا می کند. در مرحله دوم اجرای تحلیل پویا، بررسی پویا، نوع سناریوی حمله را با توجه به نوع نشت داده‌های محرمانه مثل کوکی مرورگر وب و غیره بدون دسترسی به داده‌های بحرانی و منابع داده محافظت شده در موبایل، گزارش می‌دهد.

زمانی که یک فایل APK وارد می‌شـــود، اولین قسمت، تحلیل ایستاســـت. تحلیل ایســـتا شـــامل: برگردان APK؛ «تشخیص بســـته‌بندی مجدد»[42]؛ استخراج خصوصیات و مقایسه. تحلیل ایســـتا نمی‌تواند با حالت‌هایی مثل ابهامات کد و رمزنگاری روبرو شـــود. بنابراین نیاز به تحلیل پویا وجـــود دارد. تحلیل پویا به نظارت بر رفتار برنامه در حال اجرا و کشـــف این‌که آیا رفتارهای آن با فعالیت مشکوک مطابق اســـت یا نه، نیاز دارد. ویژگی‌های مشـــکوک که در تحلیل پویا قابل توجه هستند: مسیر و جریان داده بحرانی؛ دسترســـی به دامنه بدخواه؛ هزینه بدخواه (یک برنامه که بدون اطلاع واضح برای یک ســـرویس، شـــارژ می‌شود، «هزینه‌افزار»[43] اســـت. اگر یک مقصد نامعلوم وجود داشته باشد، نرم‌افزار باید آن را به‌عنوان بدافزار در نظر بگیرد.)؛ دورزدن مجوزهـــای اندرویـــد (اگر یـــک برنامه بعضی از مجوزها را اعلام نکند اما برخی کارهای مرتبط را با همان مجوزها اجرا کند، دورزدن مجوز گفته می‌شـــود. اگر یک برنامه دسترســـی مدیر داشته باشد، می‌تواند کار حساس را بدون هرگونه مجوز انجام دهد.)

آزمایش ها و نتایج: نویســـندگان روش پیشـــنهادی را ارزیابی نکرده‌اند و صرفاً یک مفهوم بیان شده است.

تحلیـــل روش و نقاط قوت و ضعف: پیشـــنهاد روش کشـــف برای بن‌ســـازه‌های خاص همچون تلفـــن همراه و سیســـتم عامل اندروید ارزشمند است. ترکیب تحلیل ایستا و پویا نیز ایده مناســـبی است زیرا این دو اطلاع مکمل هم هســـتند. متأســـفانه، ایراد اصلی مطالعه یانگ و همکاران فقدان ارزیابی است. عدم وجود آزمایش‌های تجربی روی نمونه‌های واقعی موجب می‌شـــود خواننده هیچ بینشی از ســـودمندی و محدودیت‌های روش نداشته باشد. به‌همین دلیل روش پیشنهادی قابل استناد نیست.

۳-۲ روش تشخیص تمام خودکار هِل‌دِریود

روش: آندرانیـــو و همکاران [۷] در ســـال ۲۰۱۵ روش تمام خودکار هِل‌دِریود[44] را که نمونه‌های شـــناخته شده یا ناشـــناخته باج‌افزار و ترس‌افـــزار را از نمونه‌های بی‌خطر شناســـایی می‌کرد، ارائه دادند. روش‌هـــای قبلی عمومیت نداشـــته و همچنین محدود به نمونه‌های شـــناخته شـــده هســـتند، به راحتی از آن‌ها می‌توان فـــرار کرد و در مقابل انواع جدید، بی‌اثر هســـتند. از دید کاربر، روش‌های مبتنی بـــر امضا باید به‌طور دائم با تعاریف جدید بروز رســـانی شده که به ندرت موثر است. مقاله مورد نظر روی باج‌افزار موبایل متمرکز شـــده اســـت و چالش‌هایی را که منحصر به مسائل شناسایی باج‌افزار اســـت (فنون شناسایی قفل، عملیات رمزنگاری و پیغام‌های تهدیدآمیز) مورد بررســـی قـــرار می‌دهد. هدف این مقاله غلبـــه بر جنبه‌های نامطلوب روش‌های مبتنـــی بر امضا و شناســـایی انـــواع باج‌افزار شناخته شده و جدید اســـت. دستیابی به این هدف چالش

41-Test path
42- Repackaging detection
43-Chargeware
44-HELDRIOD

برانگیز اســت و به منظور این کــه هر دو باج‌افزار از پیش شناخته شده و همچنین نمونه‌های جدید، شناسایی شوند از روشــی تطبیق پذیر استفاده می‌شود. وجود متن تهدید آمیــز به عنوان یک اجبــار برای نویســنده باج‌افزار برای رســیدن به هدف خود، در نظر گرفته شــده است. روش پیشــنهادی مبتنی بر شناســایی عناصر اصلی اســت که برای پیاده‌ســازی یک برنامه مخرب موبایل نیاز هســتند. اختصاصاً هِل‌دِریود برنامه‌ای که در حال تلاش برای قفل کردن یا رمزنگاری دستگاه، بدون رضایت کاربر باشد و یا درخواست‌های باج روی صفحه نشان داده شود، شناسایی می‌کند. این روش نیازی ندارد که یک نمونه از یک خانواده مشــخص از قبل در دسترس باشد. همچنین تعیین می‌کند که آیا یک برنامه موبایل، کاربر را تهدید می‌کند، دســتگاه را قفــل می‌کند، داده‌ها را رمزنــگاری می‌کند یا ترکیبی از این فعالیت‌ها را انجام می‌دهد. هِل‌دِریود برای تحلیل ایستا و پویا پیاده‌ســازی شده اســت. به‌خصوص این روش از «تحلیل آلودگی[45]» ایستا و شبیه‌سازی سطحی برای یافتن جریان فراخوانی‌های تابعی که رفتارهای قفل دســتگاه یا رمزنگاری فایل را موجب می‌شــوند، استفاده می‌کند. این روش برای شناسایی رفتارهای تهدیدآمیز، از فنی مبتنی بر یادگیــری که پردازش زبان طبیعی[46]نام دارد و عبارت‌های مشــکوک را تشــخیص می‌دهد، اســتفاده می‌کند. هرچند هِل‌دِریــود اختصاصاً برای باج‌افزار طراحی شــده اســت اما مبتنی بر خانواده خاصی نیســت. علاوه براین ویژگی های شناســایی، پارامتری بوده و قابلیت انطباق‌پذیری با خانواده‌های آینده را دارد. هِل‌دِریود فایل APK اندروید را مورد بررسی قرار می‌دهد و سه شاخص مستقل که به‌طور موازی اجرا می‌شــوند را به‌کار می‌گیرد که هر شــاخص برای یک ویژگی بدافزار مشخص شده است.

آزمایش‌هــا و نتایج: مجموعه داده M1 شــامل بیش از ۴۰۰ برنامه مخرب اندروید از دســامبر ۲۰۱۴ اســت، بجز آن‌هایــی که قبــل از آن در بقیــه مجموعــه داده‌ها و هر باج‌افزار شــناخته شده، وجود داشــتند. مجموعه داده R1 باج‌افزارهای شناخته شده و مجموعه داده R2 باج‌افزارهای ناشــناخته برای نمونه‌هایی اســت که در دسامبر ۲۰۱۴ تا ژانویه ۲۰۱۵ مشــاهده شده‌اند. برای ارزیابی سه آزمایش انجام شــد که آزمایش اول برای آزمون توانایی شناسایی و آزمایش دوم برای آزمون مثبت کاذب و آزمایش ســوم به منظور سرعت شناسایی انجام گرفتند. در آزمایش اول تمام ۲۰۷ نمونه باج‌افزار در R1 شناسایی شدند: ۱۹۴ عدد با استخراج ایســتای متن و ۱۳ عدد باقیمانده با استخراج متن از رایانه کارساز کنترل و فرمان. آزمایش انجام گرفته روی R2، از بیــن ۴۴۳ نمونه کل در R2، ۳۷۵ عدد از آن‌ها به درستی به عنوان باج‌افزار یا ترس‌افزار شناخته شدند و ۴۹ عدد به‌طور صحیح هیچیک از این دو، برچســب‌گذاری نشــدند. با تحلیل روی منفی کاذب معلوم شــد نمونه‌هایی که شناســایی نشــده بودند به علت فقدان مدل‌های زبانی بود. به‌طور مثال هِل‌دِریود را روی زبان اســپانیایی تنظیم کرده‌اند و متون تهدیدآمیز را با اســتفاده از مترجم گوگل، تبدیل به زبان اســپانیایی کرده و به آن اضافه کردند، بعد از آزمایش دوباره تمام نمونه‌های قبلی که شناسایی نشده بوند، به‌طور موفقیت آمیز به عنوان باج‌افزار برچسب‌گذاری شدند.

در آزمایــش دوم کــه روی مجموعــه داده M1 انجام شــد (حاوی بدافزار بود اما باج‌افزار نداشت) هیچ نمونه‌ای به عنوان باج‌افزار برچســب زده نشــد. در آزمایش سوم، طبقه‌بندی کننده متن بین تمام رده‌ها سریع‌تر عمل می‌کند. اگر هِل‌دِریود مجبور به اســتفاده از جعبه شنی خارجی به منظور اســتخراج متن تولید شده به‌طور پویا داشته باشد، پنج دقیقه در این پیاده‌ســازی طول می‌کشد، اما این برای تحلیــل پویا غیرقابل اجتناب اســت. اگرچه این اتفاق برای تعداد کمی از نمونه‌ها می‌افتد. این روش روی صدها هزار نمونه که شامل بدافزار عمومی و باج‌افزارها بودند، اعمال شــد که به درســتی تمام نمونه‌های باج‌افزار را شناسایی کرده و در مورد برنامه‌های بی‌خطرکه شباهت به باج‌افزار

45-Taint analysis
46-NLP

داشتند، دچار ابهام نشد.

تحلیل روش و نقاط قوت و ضعف: به‌طور کلی هِل‌دِریود روشی پیشرفته را برای شناسایی بدافزار موبایل به کار گرفته است. همچنین در مورد نمونه‌های ناشناخته به خوبی عمل کرده و فقط تعدادی که زبان آن‌ها پشتیبانی نمی‌شد، گم شدند که این مورد هم به راحتی در عرض ۳۰ دقیقه حل شد (یک کادر متن به زبان اسپانیایی ایجاد و دوباره دسته‌بندپردازش زبان طبیعی اجرا شد). نتایج شناسایی‌های هِل‌دِریود نشان می‌دهد که فقط ۱۲مثبت کاذب در تمام صدهزار برنامه غیر باج‌افزار اتفاق افتاده است. با استفاده از پایگاه داده‌ای مرکب از برنامه‌های باج‌افزار، ترس‌افزار، مشکوک و بی‌خطر، هِل‌دِریود توانست ۹۹٪ از نمونه‌های اصلا دیده نشده را شناسایی کند. انتقال روش هِل‌دِریود به دنیای غیر موبایل شدنی است. بدون شک نویسنده بدافزار می‌تواند از این سیستم به‌وسیله استفاده از کد محلی یا تعبیه کردن رمزنگاری، فرار کند.

۳-۳ روش مطالعه جامع انواع باج‌افزار

روش: خراز و همکاران [۵] در سال ۲۰۱۵ نشان دادند که با نظارت فعالیت غیرعادی سیستم فایل، امکان طراحی سیستم‌های دفاعی برای متوقف ساختن تعداد زیادی از نمونه‌های باج‌افزار وجود دارد. طبق این مقاله، بسیاری از گزارش‌های امنیتی اخیر درباره باج‌افزار به جای ارزیابی‌های علمی، تکیه بر رویه‌های غیر قابل تعمیم دارند. علاوه بر این، گزارش‌های اخیر به‌طور عمده روی پیشرفت‌ها در حملات باج گیرافزار و سطح پیچیدگی آن‌ها به جای ایجاد بینشی قوی درباره فنونی که باید ضد این تهدیدها به کار گرفت، متمرکز هستند. در این مقاله نتیجهٔ مطالعه حملات باج‌افزارهای مشاهده شده بین سال‌های ۲۰۰۶ تا ۲۰۱۴ نشان داده می‌شود. در واقع دید جامعی از ۱۳۵۹ نمونه باج‌افزار متعلق به ۱۵ خانواده و چگونگی رفتار آن‌ها گفته شده است. این مطالعه‌ها نشان می‌دهد که باوجود بهبودهای پیوسته در روش‌های رمزنگاری، تعداد خانواده‌هایی با توانایی‌های مخرب پیچیده هنوز کم است. همچنین تحلیل‌های انجام گرفته مشخص می‌کند که متوقف سازی حملات پیشرفته باج‌افزار به پیچیدگی که گزارش شده، نیست.

این مقاله نشان می‌دهد که با نظارت فعالیت غیرعادی سیستم فایل، امکان طراحی سیستم‌های دفاعی برای متوقف کردن تعداد زیادی از نمونه‌های باج‌افزار وجود دارد. یک آزمایش روی فعالیت‌های سیستم فایل از چند نمونه باج‌افزار، مشخص می‌کند که با نظارت برای درخواست‌های ورودی/خروجی و محافظت از «جدول اصلی فایل[۴۷]» در سیستم فایل NTFS، امکان شناسایی و جلوگیری از تعداد زیادی از حملات باج‌افزار وجود دارد. همچنین تحلیلی روی روش‌های پرداخت پول که توسط خانواده‌های مختلف باج‌افزارها انجام شده و تراکنش‌های ۱۸۷۲ بیت‌کوین که توسط حمله کریپتولاکر استفاده می‌شده است، بررسی شده است. تحلیل تراکنش‌ها نشان می‌دهد که مجرمان فضای مجازی از روش‌های گسترده‌ای (مانند استفاده از نشانی‌های جدید برای هر آلودگی) به منظور پنهان کردن فعالیت مجرمانه حساب‌های بیت کوین استفاده می‌کنند. تعیین نشانی‌های مشکوک در شبکه بیت کوین مبتنی بر تاریخچه تراکنش، بخصوص زمانی که مجرم از چند نشانی مستقل با مقدار کمی از بیت کوین‌ها استفاده می‌کند، مشکل است.

آزمایش‌ها و نتایج: ۳۷/۹٪ از نمونه‌ها از آنوبیس و ۴۸/۳۸٪ با جستجو بین منابع عمومی بدافزار و ۱۳/۸٪ به‌صورت دستی با جستجو بین انجمن‌های امنیتی به‌دست آمده است. مجموعه داده مربوطه شامل ۱۳۵۹ باج‌افزار فعال است. ۱۸۷۲ نشانی بیت کوین جمع آوری شد. جدول سه فهرست خانواده های به‌کار رفته در این مقاله را نشان می‌دهد. این مطالعه، امکان پیاده‌سازی روش‌های دفاعی را ضد حملات مخرب باج‌افزار بررسی می‌کند. با توجه به تحلیلی که روی فعالیت سیستم فایل نمونه‌های باج‌افزاری که فایل‌های کاربر را هدف قرار می‌دهند، انجام شده است، معلوم شد که رده‌های متفاوت حملات باج‌افزار با

47-Master File Table (MFT)

سطوح مختلف پیچیدگی، ویژگی‌های بسیار مشابه را از یک سیستم فایل به علت ماهیت این حملات، به اشتراک می‌گذارد. همچنین تحلیل‌ها نشان می‌دهند، در سیستمی که تحت حمله قرار گرفته، یک هشدار می‌تواند یک تغییر مهم در فعالیت سیستم فایل باشد، چون فرایند مشکوک می‌تواند تعداد زیادی درخواست‌های دسترسی سیستم فایل مشابهی تولید کند. در نتیجه اگر فعالیت سیستم فایل (به‌طور مثال تغییرات در جدول اصلی فایل و انواع بسته‌های درخواستی (ورودی/خروجی IRP به سیستم فایل)، مورد بررسی قرار گیرند، امکان شناسایی انواع باج‌افزار که فایل‌های کاربر را هدف قرار می‌دهند، وجود دارد. توسعه سازوکارهای دفاعی ضد این حملات به علت مهندسی سیستم فایل NTFS امکان‌پذیر است. راهبردهای کاهش حملات که در این مقاله گفته شده است، به‌صورت زیر هستند:

الف) پایش فراخوانی واسط‌های برنامه کاربردی[48]: تعداد زیادی از نمونه‌های باج‌افزار از واسط‌های برنامه کاربردی ویندوز برای قفل صفحه رومیزی کاربر استفاده می‌کنند. بنابراین با نظارت فراخوانی‌های واسط برنامه کاربردی ویندوز، با این که روش جدیدی نیست، اما حملات باج‌افزارهایی که از روش‌های پیچیده استفاده نمی‌کنند را شناسایی می‌کند.

ب) پایش فعالیت سیستم فایل[49]: یک طبقه‌بندی کننده می‌تواند برای تشخیص مدخل‌های مشکوک و بی‌خطر جدول اصلی فایل که تحت حمله باج‌افزار هستند، به کار برود. در واقع مدخلی از جدول که تغییر کرده باشد، مشکوک است. به منظور تمایز بین فعالیت بی‌خطر و مخرب روی سیستم فایل، روش ممکن دیگری که می‌تواند به‌کار برود، نظارت تمام درخواست‌های سیستم فایل که فرایندهای حالت کاربر تولید می‌کنند، می‌باشد. بازیابی فایل‌های حذف شده توسط حمله باج‌افزار می‌تواند به راحتی امکان‌پذیر باشد، اگر صفت $DATA در مدخل جدول اصلی فایل، مقیم باشد، محتوای فایل می‌تواند به مکان دیگری رونوشت شود. برای صفت‌های $DATA که مقیم نیستند، نیاز به تحلیل کردن RUNLIST در مدخل جدول اصلی فایل و رونوشت داده‌های اولیه به مکان دیگر و انجام عمل بازیابی می‌باشد.

پ) به‌کاربردن منابع تله: مهاجمان می‌توانند توسط حملاتی که رفتار عادی کاربر را شبیه‌سازی می‌کنند، از شناسایی فرار کنند. یک راه‌حل می‌تواند استفاده از فایل‌های تله در چندین محل دیسک باشد که منظم نظارت می‌شوند. این روش می‌تواند، با وجود این‌که نمونه‌های باج‌افزار از فنون جدید سیستم‌های رمزنگاری استاندارد یا سفارشی شده استفاده کنند، فرصتی برای تشخیص فرایندهای مخرب در مراحل اولیه حملات باشد.

تحلیل روش و نقاط قوت و ضعف: نقطه قوت این روش بررسی تعداد بسیاری از نمونه باج‌افزارهاست که اطلاعات وسیع و عمومی‌تری از رفتار باج‌افزارها فراهم می‌سازد. با وجود آزمایش‌های گسترده و یافته‌های ارزشمند، نویسندگان روش جدیدی را بر مبنای یافته‌هایشان ارائه نکرده‌اند.

۳-٤ روش آن‌وِیل

روش: خراز و همکاران [۴] در سال ۲۰۱۶ روشی ارائه دادند که تعاملات برنامه مشکوک با سیستم فایل را مورد پایش دقیق قرار می‌دهد تا سیستم بتواند به‌طور دقیق رفتار رمزنگاری باج‌افزار را توصیف کند. سیستم‌های فعلی رایج که برای شناسایی بدافزارها وجود دارند، به‌طور اختصاصی به شناسایی باج‌افزارها نمی‌پردازند. سیستم‌های شناسایی بدافزارها روی عملکردهای پنهانی بدافزار متمرکز هستند (مثل عملکرد مشکوک سیستم عامل برای «ثبت رویدادهای کلیدها»[50]) که در شناسایی باج‌افزارها با شکست مواجه می‌شوند. علت اینست که باج‌افزارها مانند برنامه‌های کاربردی که عمل رمزنگاری یا فشرده سازی را انجام می‌دهند، ظاهر شده و فعالیت می‌کنند. همچنین این سیستم‌ها برای شناسایی رفتارهای ویژه‌ای که باج‌افزارها

48-API Call Monitoring
49- Monitoring File System Activity
50-Keylogging

به‌کار می‌برند زیاد مناسب نیستند؛ مثل طبقه‌بندی نادرست خانواده‌های باج‌افزار توسط پویشگرهای AV. ارزیابی این مقاله نشان می‌دهد که سیستم‌های فعلی شناسایی بدافزارها، مدل‌های رفتاری درستی برای شناسایی رده‌های مختلف حملات باج‌افزار ندارند.

بعضی از روش‌ها روی تحلیل جریان کنترلی برنامه متمرکز شده‌اند، مثلاً یک روش به مدل‌سازی برنامه بر طبق جریان کنترل در سطح دستور می‌پردازد. کارهای دیگری که انجام شده، تحلیل و شناسایی بدافزار با استفاده از توصیف معنایی سطح بالای رفتار زمان اجرای آن‌ها که از ترتیب فراخوانی‌های سیستمی و دسترسی منابع سیستم عامل نشئت گرفته شده است، می‌باشد. در گذشته از هدف مصنوعی یا تله برای شناسایی نقص‌های امنیتی استفاده می‌شده است. روش پیشنهادی مقاله مذکور فرض می‌کند که نمونه‌های باج‌افزار می‌توانند از تمام روش‌هایی که دیگر نمونه‌های بدافزار ممکن است به کار ببرند، استفاده کنند. همچنین سیستم پیشنهادی فرض می‌کند که حملات موفقیت آمیز باج‌افزار یک یا چند از فعالیت‌های زیر را انجام می‌دهند:

– پیامی که بعد از حمله موفقیت‌آمیز به‌طور ثابت نشان داده می‌شود.

– رمزنگاری و حذف فایل‌های خصوصی کاربر

– رمزنگاری و حذف فایل های خصوصی کاربر بر اساس ویژگی‌های مشخص (اندازه، تاریخ دسترسی، پسوند)

در روش مقاله مورد نظر، سیستم به‌طور خودکار محیط اجرایی مصنوعی واقع‌گرایانه‌ای ایجاد کرده و نظارت می‌کند که باج‌افزار چگونه با آن محیط تبادل دارد. این پایش دقیق تبادلات با سیستم فایل امکان توصیف دقیق رفتار رمزنگاری باج‌افزار را برای سیستم فراهم می‌سازد. همزمان، سیستم تغییراتی از سیستم را که شبیه به رفتارهای باج‌افزار است ردیابی می‌کند. کلید اصلی برای موفق بودن این است که باج‌افزار نیاز به دسترسی به فایل‌های قربانی یا صفحه رومیزی دارد. روش آن‌وِیل امکان تحلیل نمونه‌های زیادی از بدافزارها را در مقیاس وسیع فراهم می‌کند و آن‌هایی که رفتارهای شبیه به باج‌افزارها دارند را شناسایی نموده و علامت گذاری می‌نماید. علاوه براین سیستم قادر است بینشی را ایجاد نماید که بتوان فهمید باج‌افزار چگونه عمل کرده و چگونه به‌طور خودکار بین رده‌های مختلف باج‌افزار، تمایز قائل می‌شود. روش‌های جدیدی در این مقاله برای شناسایی باج‌افزار پیشنهاد شده است که به‌عنوان «قفل‌کننده فایل»[51] شناخته می‌شود و فایل‌هایی را که روی کامپیوتر قربانی قرار دارند، هدف‌گیری می‌کند. این فن مبتنی بر پایش سیستم فایل مورد دسترسی در ترکیب با استقرار محیط‌های مصنوعی است که به‌طور خودکار برای راه‌اندازی باج‌افزار ایجاد شده است.

آزمایش‌ها و نتایج: نمونه اولیه آن‌وِیل در ویندوز و در «جعبه شنی کوکو»[52] که چارچوبی متن باز و شناخته شده است و از طریق راه‌اندازهای هسته ویندوز که قابلیت‌های نظارت را فراهم می‌کنند، پیاده‌سازی شده است. علاوه براین تعدادی مؤلفه بیرون از جعبه شنی برای نظارت بر واسط کاربری سیستم کامپیوتری هدف، اضافه شده است. تحلیل طولانی مدتی روی ۱۴۸۲۲۳ نمونه باج‌افزار انجام شده که نشان می‌دهد آن‌وِیل می‌تواند ۱۳۶۳۷ نمونه باج‌افزار را از چندین خانواده شناسایی کند.

همچنین حین اجرای نمونه بدافزار روی سیستم، یک فرایند، فعالیت اصلی کاربر مانند راه‌اندازی یک مرورگر و و رفتن به وبگاه‌ها یا کلیک کردن روی صفحه رومیزی را شبیه سازی می‌کند. این تبادل به‌طور تصادفی تولید شده اما در سراسر اجرا ثابت می‌باشد. هر نمونه برای ۲۰ دقیقه در محیط تحلیل اجرا می‌شود. محیط‌های کاربری برای هر اجرا تولید می‌شود، ردیابی‌های ورودی/خروجی سیستم فایل ثبت شده و تصویر لحظه‌ای از صفحه، قبل و بعد از اجرا گرفته می شود. بعد از هر اجرا ماشین مجازی به وضعیت پاک برای جلوگیری از هرگونه دخالت در سراسر

51-File Locker
52-Cuckoo

اجرا، «فراخوانی برگشتی»[53] می‌کند. تمام آزمایش‌ها بر طبق دستورالعمل‌های تجربی برای آزمایش‌های بدافزار انجام شده است.

آزمایش اول: در این آزمایش تأثیر آن‌وِیل را روی مجموعه داده‌ای برچسب شده ارزیابی کرده و نمونه‌های متفاوتی از قفل‌کننده صفحه را برای تعیین بهترین مقدار آستانه tsim مورد آزمایش قرار داده است. نمونه‌های باج‌افزار از نظرگاه برخطی که نمونه‌های بدافزار را به اشتراک می‌گذارند، جمع‌آوری شده اند. همچنین نمونه‌های باج‌افزار برچسب شده از دو شرکت شناخته شده گرفته شده است و در کل ۳۱۵۶ نمونه جمع‌آوری شده است. به‌منظور اطمینان از این که آن نمونه‌های باج‌افزار فعال بودند، آن‌ها را در محیط آزمون خود اجرا کرده‌اند. ۲۱۲۱ نمونه را به‌عنوان باج‌افزار فعال تأیید کرده‌اند. بعد از هر اجرا، فعالیت سیستم فایل هر نمونه را برای هر علامت حمله به داده کاربر بررسی شده است. مجموعه داده اکثر باج‌افزارها را پوشش می‌دهد. علاوه بر این برای باج‌افزارهای برچسب‌گذاری شده، مجموعه داده‌ای که شامل نمونه‌های غیر باج‌افزار است نیز ایجاد شده است. این نمونه‌ها به «بن‌سازه تحلیل آنوبیس»[54] ارسال شده و شامل مجموعه‌ای از نمونه‌های مشکوک بی‌خطر است. ۱۴۹ برنامه کاربردی اجرایی بی‌خطر که رفتار شبیه باج‌افزار مانند حذف و رمزگذاری و فشرده‌سازی داشتند، انتخاب شد. همچنین ۳۸۴ نمونه بدافزار غیر باج‌افزار از ۳۶ خانواده بدافزار برای ارزیابی نرخ آن‌وِیل آزمون شده است. تفاوت اصلی نام فایل‌ها و پسوندهای اصلاح شده با نویسه‌های تصادفی برای به حداقل رساندن شانس بازیابی فایل‌ها بر اساس نام آن‌ها در جدول اصلی فایل در سیستم فایل NTFS است.

آزمایش دو: این آزمایش دقت آن‌وِیل را زمانی‌که به مجموعه داده بزرگی از نمونه‌های بدافزار در دنیای واقعی اعمال می‌شود، ارزیابی می‌کند. سپس نتایج به‌دست آمده با گزارشی که توسط پویشگرهای AV در انجمن ویروس بیان شده بود، مقایسه شده است. نمونه‌ها از ۱۸ می ۲۰۱۵ تا فوریه ۲۰۱۶ جمع‌آوری شد. از این رو قبل از اجرای آزمایش، ورودی‌های آنوبیس با حذف آن‌هایی که آشکارا اجرایی نبودند، پالایش شده است (مثل تصاویر، پی‌دی‌اف‌ها). بعد از مرحله پالایش، مجموعه داده شامل ۱۴۸۲۲۳ نمونه متفاوت بوده و هر نمونه برای دستیابی به ردگیری ورودی/خروجی و نمرات تمایز تصویر صفحه رومیزی قبل و بعد از اجرا به آن‌وِیل ارسال شده است.

نتیجه به‌دست آمده از آزمایش اول نشان می‌دهد که با tsim= ۰/۳۲ بیش از ۹۷٪ از نمونه‌های باجگیر افزار در سراسر هر دو نمونه صفحه نمایش و قفل فایل‌ها با ۱۰۰٪ دقت کشف شدند. در آزمایش دوم از این شباهت برای شناسایی باج‌افزار قفل کننده صفحه در یک بدافزار ناشناخته به آن‌وِیل استفاده شده است. نتیجه آزمایش دوم بدین صورت است که با حد آستانه شباهت tsim=۰/۳۲، آن‌وِیل، (۱۳۶۳۷ ۹/۲٪) از مجموعه داده نمونه را به‌عنوان یک باج‌افزار برچسب‌گذاری کرد که این‌ها شامل هم نمونه‌های قفل‌کننده فایل و هم قفل‌کننده صفحه رومیزی بودند.

تحلیل روش و نقاط قوت و ضعف: آن‌وِیل توانست نوع جدیدی از باج‌افزار را که تا قبل از آن توسط هیچ شرکت امنیتی گزارش نشده، شناسایی کند. آن‌وِیل می‌تواند به راحتی روی هر سیستم تحلیل بدافزاری به‌وسیله ضمیمه‌سازی به راه‌انداز سیستم فایل در محیط تحلیل، استقرار یابد. ارزیابی نشان می‌دهد که این روش در عمل به خوبی کار می‌کند (دستیابی به نرخ مثبت واقعی ۹۶/۳٪) و در خودکارسازی تعریف نمونه‌های باج‌افزار که برای سیستم‌های تحلیل و شناسایی پیشنهاد شده، مفید است. از نقاط ضعف آن می‌توان گفت این امکان وجود دارد که مهاجم راه‌هایی برای شناسایی محیط کاربری تولید شده پیداکرده و از آن اجتناب کند که البته هزینه بالایی برای مهاجم دارد. امکان دیگر این است که بدافزار ممکن است

53- Callback
54-Anubis

فقط یک بخش مشـــخص از یک فایل را بـــه جای کل فایل رمز کند یا محتوای فایل را با استفاده از الگوهایی غیرقابل خواندن نماید. نویسندگان بدافزار ممکن است از روش‌های دیگری مثل ویدئو یا فایل‌های صوتی برای هشدار به جای قفل صفحه به کاربر اســتفاده کننــد، روش مذکور در این مقاله زمانی اســتفاده می‌شــود که بدافزار ابتدا قادر باشد فایل‌های کاربر را قفل کند. واحد اســتخراج متن باید بهبود پیدا کند، چون بدافزار ممکن است از کلمات نامتعارف برای نشــان دادن یادداشت خود اســتفاده نماید. آن‌وِیل داخل هسته اجرا می‌شــود و هدف آن شناســایی باج‌افزارهای ســطح کاربر اســت که اگر باج‌افزار در سطح هسته اتفاق بیفتد، خطر محســوب می‌شود. البته حمله در سطح هسته پیچیدگی‌هایی برای مهاجمان دارد.

۳-۵- روش سیستم تشخیص زودهنگام هشدار برای متوقف ساختن حملات باج‌افزار روی داده‌های کاربر

روش: اِســکایف و همکاران [۳] در سال ۲۰۱۶ سیستم کریپتولاک برای تشــخیص زودهنگام هشدار را پیشنهاد دادنــد که کاربر را در طی فعالیت مخرب فایل آگاه کرده و با استفاده از مجموعه‌ای از شاخص‌های رفتاری می‌تواند یک فرایند را متوقف نماید. مشکل شناسایی بدافزارهایی که قبلا شناسایی نشده‌اند، امکان فرار بدافزارها از این سیستم شناســایی با اســتفاده از امضای پیچیده و سخت، ممکن است. روش پایشگر صحت زمانی که فایل‌های مهم سیستم تغییر می‌کنند، مدیر را آگاه می‌کند که این روش برای کاربر مزاحم و خسته کننده است. تمرکز این مقاله روی محافظت از داده‌های کاربر برای از دســت‌رفتن کلی اســت. بر این اســاس نویسندگان سیستم تشــخیص زود هنگام هشدار پیشــنهاد کرده‌اند که کاربر را در طی فعالیت مخرب فایل آگاه کــرده و با اســتفاده از مجموعه‌ای از شــاخص‌های رفتــاری می‌تواند یک فرایند را که بــدون اجازه می‌خواهد به مجموعه زیادی از داده‌های کاربر دسترســی پیدا کرده و دســتکاری کند، متوقف نماید. با ترکیب شاخص‌ها برای باج‌افزارها، سیستم می‌تواند برای شناسایی سریع با مثبت کاذب پایین استفاده شود. حد متوسط گم شدن فایل‌ها، ۱۰ فایل از ۵۱۰۰ فایل در دســترس اســت. این سیستم تعداد فایل‌های گم شده قربانی را کم می‌کند.

در این سیســتم نظــارت روی تغییــر داده‌های کاربر به جای تلاش برای شناســایی باج‌افــزار از طریق اجرا یا محتوای آن، انجام می‌شــود. این سیســتم از سه شاخص اصلی برای شناسایی تغییرات مشــکوک فایل که هرکدام از این شــاخص‌ها یک جنبه از رفتار باج‌افزار را تحت نظر میگیرند، استفاده می‌کند و زمانی هر سه آشکار می‌شوند که یک فایل باج‌افزار اجرا شــود. این سه شاخص، پیش از این در سیســتم تشــخیص دیگری به کار نرفته است. در این سیســتم تحلیل گســترده رمزنگاری باج‌افــزار انجام می‌شــود. نرخ ۱۰۰٪ مثبت واقعی در ۴۹۲ باج‌افزار متمایز از ۱۴ خانواده، نشــان داده می‌شــود. یعنی حداقل صفر و حد متوسط ۱۰ (۰/۲٪) از فایل‌ها گم شده است. این سیستم تعداد فایل‌های گم شده را کم می‌کند، بنابراین پرداخت باج کم می‌شود.

کریپتولاک روی شناســایی باج‌افزار از طریق نظارت بیدرنگ داده‌های کاربر، متمرکز است. اتحاد شاخص‌های منحصربه‌فــرد یک معیار قوی برای مشــکوک شــدن به یک فرایند اســت. با ردگیری این شــاخص‌ها و نظارت بر وضعیــت یک فرایند در حال اجــرا، می‌توان امتیازی برای «شــهرت»[55] تعیین کرد کــه مشــخص‌کننده این موضوع اســت که آیا فرایند رفتاری مشــکوک دارد یا خیر. زمانی که به حد آســتانه رســید، کریپتولاک کاربر را آگاه کرده و فرایند مربوطــه را معلّق می‌کند. در واقع جلوی باج‌افزار را از رمزنــگاری کامل داده‌ها، می‌گیرد. مســئله اصلی در رابطه با این سیستم، شناسایی زودهنگام باج‌افزار با مثبت کاذب پایین است که بتواند سیستم را برای اجرا عملی کند. اتحاد شــاخص‌های مطرح شــده، قابلیت شناسایی سریع با مثبت کاذب پایین را ممکن می‌ســازد. این سیســتم سه شاخص اصلی و دو شــاخص ثانویه را در نظر می‌گیرد. شاخص‌های اصلی: الف) تغییرات نوع فایل؛ ب) اندازه‌گیری

55-Reputation

شباهت[۵۶]؛ پ) آنتروپی شـــانون. شاخص‌های ثانوی: الف) حذف؛ ب) محدود کردن نوع فایل[۵۷].

این مقاله فعالیت‌های باج‌افزار را به ســـه رده تقســـیم می‌کند که در بخش دوم مقاله به آن‌ها اشاره شد و به‌ترتیب رده‌های A، B و C هستند.

آزمایش‌هـــا و نتایج: با مطالعه چگونگی توزیع فایل در تمام سیستم فایل و فهرست‌ها و اسناد کاربر، یک مجموعه ســـند ۵۰۹۹ فایل کـــه در درخت فهرســـت تودرتو با ۵۱۱ فهرســـت گسترش یافته است، مطالعه شده است. فایل‌های جمع‌آوری شده از فایل‌های xlsx، pptx و docx و فایل‌های فشرده صوتی هستند. ترکیب این‌ها ۱۱۸۰۹ فایل را تشکیل می‌دهد. از هر مجموعه چند فایل به تصادف انتخاب کرده و آن‌هـــا را در فهرســـت قرار می دهند. ســـرانجام درخت فهرست در پوشه اســـناد کاربر در جعبه شنی کوکو قرار داده می‌شود. در این آزمایش ۲۶۶۳ نمونه بدافزار از طرق مختلف به‌دست آورده می‌شـــود. هر نمونه ۲۰ دقیقه اجرا شده و درنهایت ۲۱۷۱ نمونه بدافزار به دلیل این‌که فایلی را تغییر نداده بودند، حذف می‌شوند. این مطالعه نزدیک چهار برابر (۱۴خانواده بدافزار) تعداد خانواده‌های قبلی مطالعه شده توسط خراز را تحت پوشش قرار می دهد. کریپتولاک تمـــام ۴۹۲ نمونه را با ۳۳ یا تعداد کمتری فایل گم شـــده، شناســـایی می‌کند. آزمایش انجام شده دارای مثبت کاذب کم و سرعت بالای تشخیص است. تعداد فایل‌های گم شده قبل از شناســـایی مهمترین معیار این روش است. به‌طور متوسط سیستم بعد از گم شـــدن ۱۰ فایل بین ۵۰۹۹ فایل آزمایشی (۰/۲٪) باج‌افزار را کشف می‌کند. رده B بیشترین آمار گم شـــدن فایـــل را دارد. تمام ســـه شـــاخص اولیه باارزش بوده و سیســـتم را در شناسایی زودهنگام یاری می‌کنند. در واقع اتحاد شـــاخص برای تســـریع شناسایی، حیاتی است.

پیش از بررســـی رفتـــار داده محور باج‌افـــزار انتظار می‌رفت که نمونه‌ها به جســـتجوی ژرفای در فهرســـت‌ها بپردازند، اما رفتاری متفاوت در نمونه‌ها دیده شـــد. برای مثال تِلساکریپت[۵۸] جستجوی ژرفای داشته و به فایل‌ها در عمیق‌ترین فهرست دستیابی پیدا می‌کند. در اولین فهرست مورد دستیابی پیغام باج را قرار داده و در فهرست دومی که دست می‌یابد شروع به رمزنگاری می‌کند. سی‌بی‌تی‌لاکِر[۵۹] بـــه فایل‌هایی با پســـوند مشـــخص (.txt و .md) به ترتیب صعودی انـــدازه فایل، حمله می‌کند. جی‌پی‌کُد[۶۰] از ریشـــه فهرست شروع کرده و به سمت پایین درخت حرکت می‌کند. این نمونه چون هیچکدام از فایل‌های آزمایشـــی را قبل از کشف شدن، تغییر نداده یا حذف نمی‌کند، مورد توجه است. رده C فایل‌ها را از هر فهرســـت که دسترســـی پیدا کرده، می‌خواند و در یک فایل جدید می‌نویسد و تلاش می‌کند که محتوای اصلی را حذف کند. ترتیب فایل‌هایی که به‌وســـیله بدافزار مورد حمله قرار گرفته اند، ســـرعت کریپتولاک را برای شناســـایی و متوقف کردن باج‌افزار تحت تاثیر قرار می‌دهند.

تحلیـــل روش و نقاط قوت و ضعـــف: از نقاط قوت آن می‌توان گفت، از آنجایی که کریپتولاک به‌جای تمرکز روی محتوای بدافزار یا اجرای آن، روی تغییرات داده‌های کاربر متمرکز است، این سیستم با استفاده از APIهای سطح بالا موقعیت خوبی برای متوقف کردن بدافزارهایی که سیستم فایل را دســـتکاری می‌کنند، دارد. نقاط ضعف کریپتولاک عبارتنـــد از: الف) کریپتولاک قادر به تعیین هدف تغییراتی که تحت نظر دارد، نیســـت. مثلاً نمی‌تواند متوجه شود که کاربر یا باج‌افزار، مجموعه زیادی از فایل‌ها را رمزنگاری می‌کنند. این محدودیت باعث می‌شود که سیستم از کاربر درخواست کند که تصمیم نهایی را بگیرد؛ ب) نسخه اولیه کریپتولاک بهینه‌سازی نشده است. این نسخه سربارهای تاخیر بـــرای عملیات بازکـــردن و خواندن فایـــل کمتر از یک میلی‌ثانیه، بســـتن متوســـط تأخیـــر ۱/۵۸ میلی‌ثانیه، عملیـــات تغییر نـــام و نوشـــتن (بیـــن ۹ و ۱۶ میلی‌ثانیه) دارد.

56- Similarity measurement
57-File type funneling

58-Telsacrypt
59-CBT-locker
60-GPcode

۳-۶ روش کنترل کننده اتصال و قطع کننده اتصال

روش: از آنجائی‌کـه خطرناک‌ترین نـوع باج‌افزار، نوع رمزنگاری ترکیبی اسـت، در ایـن مقاله [۱۶] روش کنترل اتصـال و قطع کننده اتصـال که چارچوبـی جدید برای تشـخیص اکثر انواع خطرنـاک باج‌افـزار و جلوگیری از رمزنگاری فایل‌های قربانی اسـت، بیان شده است. بعد از تحلیـل بیش از ۴۰ باج‌افـزار و در نظر گرفتن فناوری‌های ضد بدافزار رایج و بر طبق پیش‌بینی‌های بدافزارها و ضد بدافزارها، اولین نسخه این چارچوب بر پایه ایده‌ای مرتبط با مرحله تبادل کلید، طراحی شـده اسـت. در این مرحله اکثر باج‌افزارهای پیشرفته از الگوریتم تولید دامنه استفاده می‌کننـد. در چارچوب پیشـنهادی، ناظـر اتصال طراحی شده است که می‌تواند درخواسـت دامنه DNS که توسط الگوریتم‌های تولید دامنه تولید می‌شود را تشخیص دهد.

در ایـن روش از «زنجیره مارکوف»[۶۱] و یک مدل تغییر حرف به حرف متون از انگلیسـی و فارسـی که با الفبای انگلیسـی برای مرحله آموزش نوشـته می‌شود، استفاده شده اسـت. برای مثال در انگلیسی انتظار می‌رود که بعد از حرف «q» حرف «u» را دشـته باشـیم امـا اگر غیر از آن باشـد، که البته با احتمال کمی اتفـاق می‌افتد، می‌تواند کمی هشـداردهنده باشـد. بر طبـق این ایـده، این روش می‌توانسـت کاربران را از درخواست‌ها آگاه کرده و مانع باج‌افزارهای پربقاء شـود. در این چارچوب، ناظر اتصال، تمام ترافیک‌های خروجی قابل اجرا را بررسـی کرده و از کاربر برای پذیرش اتصال یا چشم‌پوشـی از آن، سـوال می‌شود. در حالت پیشرفته چارچوب پیشنهادی یک تاییدیه امضای کد، افزوده می‌شـود. علاوه بر این نویسندگان به توسعه‌دهندگان پیشـنهاد کرده‌اند که نشـان‌های اتصال برنامه‌ها را بـه «مرکز صدور گواهی»[۶۲] آن‌ها بفرسـتند. ایـن مرکز بعد از بررسـی فهرسـت را تایید کـرده و به عنوان نشـان‌های درخواسـت اتصال تایید شـده[۶۳] تایید می‌کند.

61- Markov chain
62- CA
63- VRCA

آزمایش‌هـا و نتایـج: روش پیشـنهادی روی بیش از ۲۰ نمونه باج‌افزار جدید رایج، آزمایش شـده اسـت. این چارچوب تمـام باج‌افزارهای پربقاء را قبل از تبادل کلید و تکمیل شدن رمزنگاری آن‌ها، تشخیص می‌دهد. نمونه‌های مورد اسـتفاده از «نـکات بدافزارها»[۶۴] و «بـوق رایانه»[۶۵] انتخاب شده‌اند، چون بسیار مشهور بوده و پیچیده هستند. نرخ تشـخیص باج‌افزارهای پربقاء در روش مذکور ۱۰۰ درصد با منفی کاذب صفر درصد است.

تحلیل روش و نقاط قوت و ضعف: نقطه ضعف روش پیشـنهادی این اسـت که فقط می‌تواند مانع باج‌افزارهای پربقاء از رمزکردن داده‌ها شود و غیر باج‌افزارهای پربقاء می‌توانند حمله خود را کامل کنند. خوشبختانه آسیب‌های ناشـی از غیر باج‌افزارهـای پربقاء برگشـت‌پذیر بوده و محصولات ضد بدافزار قادر هسـتند مشکل را حل کرده و داده‌های مورد حمله را بدون پرداخت باج، رمزگشایی کنند. مزیت ایده مطرح شـده این اسـت که این چارچوب اولین چارچوبی اسـت که روی مسئله باج‌افزار به‌وسیله نظارت بر اتصالات مشـکوک و ممانعت از رمزنگاری داده کاربر متمرکز شده است. ارزیابی‌ها نشان می‌دهند که چارچوب پیشـنهادی می‌تواند اکثر باج‌افزارهای پربقاء خطرناک را به‌طور موفقیت‌آمیز خنثی کند که این یک مشـکل در زمینه مهاجرت بدافـزار بود. این چارچوب در شناسـایی انواع دیگر نرم‌افزارهای مشـکوک مانند بدافزارهای کاوشـگر بیت‌کوین‌هـا و بات‌نت‌ها و بدافزارهای راه‌اندازی از طریق دریافت نیـز مفید اسـت. به‌دلیل این که ایـده مختص به باج‌افزارهای پربقاء نیسـت، می‌تواند روش دفاعی مفیدی علیه بسیاری از انواع حملات باشد.

۳-۷ چارچوب تشخیص باج‌افزارهای پربقاء

روش: احمدیـان و همـکاران [۱۵] در سـال ۲۰۱۶ چارچـوب جدیدی را بـرای تشـخیص خطرناک‌ترین و مخرب‌ترین باج‌افزار یعنی باج‌افزار پربقاء پیشـنهاد دادند که آن را 2entFOX نامیدنـد. در این روش، رفتار باج‌افزار

64- https://malwaretips.com/
65- https://www.bleepingcomputer.com/

تحلیل شده و ویژگی‌های مناسبی که در تشخیص این نوع بدافزارها با دقت زیاد و مثبت کاذب پایین مفید هستند، پیدا می‌شوند. خروجی این مرحله از کار، استخراج ۲۰ ویژگی است. نویسندگان ادعا کرده‌اند که برای اولین بار در این حوزه توانستند به مجموعه‌ای مناسب برای تشخیص باج‌افزارهای پربقاء، دست یابند. معماری پیشنهادی آن‌ها برای تشخیص بر اساس شبکه باور بیزین طراحی شده است.

آزمایش‌ها و نتایج: ابتدا مطالعه‌ها و تحلیل‌ها روی ۲۰ نمونه مناسب باج‌افزار انجام می‌شود، سپس بر اساس معماری روش پیشنهادی، ویژگی‌های مطلوب شناسایی و استخراج می‌شود. بعد از طراحی ساختار موتور تشخیص، «جدول‌های احتمال شرطی[66]» در شبکه بیزین مبتنی بر تحلیل نمونه نمونه باج‌افزار پربقاء، محاسبه می‌شود، همچنین در مرحله یادگیری، تحلیل چهار نرم‌افزار بی‌خطر و سه نوع نمونه بدافزار استفاده شده است. روش 2entFOX یکی از انواع سیستم‌های تشخیص مبتنی بر ویژگی است با این تفاوت که در این سیستم تشخیص، رفتار غیرقانونی باج‌افزارهای پربقای مورد هدف، به‌طور دستی استخراج شده و به‌عنوان یک ویژگی عمومی و بر اساس این دانش، موتور تشخیص در مرحله یادگیری، طراحی شده است. سپس در طی مرحله آزمایش، در قالب موتور تشخیص و بر اساس معماری، هر برنامه‌ای که انطباق زیادی با این ویژگی‌ها داشته باشد به‌عنوان باج‌افزار پربقاء شناخته می‌شود. حد آستانه ۸۵ بر طبق آمارهای به‌دست آمده از باج‌افزارهای پربقاء و نرم‌افزار مرزی و سایر انواع بدافزار (شامل بات‌نت‌هایی که از ویژگی مرحله تبادل کلید با اهداف دیگر استفاده می‌کنند) برای 2entFOX در نظر گرفته شده است. انتخاب این آستانه بر اساس آزمایشی است که از نظر نویسندگان، احتمال این که یک نرم‌افزار بی‌خطر هر دو رده خصوصیت را داشته باشد غیرممکن است، اما اگر نرم‌افزار بی‌خطری با این دو رده ویژگی وجود داشته باشد، با در نظر گرفتن شرایط عادی ویژگی‌های دیگر به احتمال تقریبی ۸۴/۹٪ می‌رسد، بنابراین حد آستانه ۸۵ در نظر گرفته شده است. بعد از آزمایش‌ها مقدار بازیابی سیستم تشخیص براساس نمونه‌های ارزیابی شده ۱۰۰٪ است. دقت روش بر طبق تشخیص باج‌افزارهای کم بقاء در میان نمونه‌ها ۸۷/۵٪ است و بنابراین مقدار معیار اِف ۹۳/۳۳٪ میباشد.

تحلیل روش و نقاط قوت و ضعف: این روش نرخ مثبت کاذب پایینی دارد اما توسعه مجموعه ویژگی کامل و با دقت، زمان و هزینه زیادی نیاز دارد. استخراج ویژگی‌های دقیق باج‌افزارهای پربقاء آسان نیست، و استخراج ویژگی‌های کافی از گروه‌های ویژگی‌های رفتاری، بسیار چالش برانگیز است و این قدرت 2entFOX است. از طرف دیگر از آنجایی که رفتار برنامه از ویژگی ناشی می‌شود، این روش مشابه روش‌های مبتنی بر امضا هستند اما چون رفتار باج‌افزارهای پربقاء به شکل عمومی استخراج می‌شود، این سیستم تشخیص قادر است هم باج‌افزارهای شناخته شده و هم ناشناخته پربقاء را برخلاف روش های مبتنی بر امضا تشخیص دهد.

۳-۸- تحلیل و دسته‌بندی پویا برای تشخیص باج‌افزار

روش: گاندورا[67] و همکاران [۱۸] با استفاده از روش‌های یادگیری ماشینی روی اطلاعات حاصل از تحلیل پویا، «اِلدِرَن»[68] را طراحی و ارائه کرده‌اند. استفاده از تحلیل پویا برای طراحی روش پیشنهادی به این علت بوده که باج‌افزارها از روش‌های متعددی برای «طبقه‌بندی»[69] استفاده می‌کنند و این موضوع روش‌های مبتنی بر تحلیل ایستا را دچار شکست می‌نماید. باج‌افزارها نشانه‌هایی دارند و روش پیشنهادی عملیات زمان اجرای فایل مشکوک را پایش می‌کند و به‌دنبال این نشانه‌ها می‌گردد. این نشانه‌ها از رفتارهای اجرایی مشترک بین باج‌افزارها هستند که از پیش درون روش پیشنهادی جاسازی می‌شوند.

66-CPT

67- Sgandurra
68- EldeRan
69- Packing

آزمایش‌ها و نتایج: برای ارزیابی اِلدِرَن، ۵۸۲ باج‌افزار و ۹۴۲ فایل سالم مورد استفاده قرار گرفت. دسته‌بند مورد استفاده برای تعیین وضعیت نوع فایل مشکوک به دقت ۹۹/۵٪ دست پیدا کرد که با منحنی ROC اندازه‌گیری شده است. باج‌افزارهای مورد استفاده در آزمایش از ۱۱ خانواده از نمونه‌ها انتخاب شده‌اند.

تحلیل روش و نقاط قوت و ضعف: مزیت روش پیشنهادی اینست که برای پیاده‌سازی آن نیاز نیست خانواده کاملی از باج‌افزارها از قبل موجود باشد. بنابراین روش پیشنهادی در کشف نمونه‌های جدید و ناشناخته هم می‌تواند مؤثر عمل کند. مشکل روش پیشنهادی این است که اگر باج‌افزار هیچ یک از رفتارهای مشترکی که در دسته‌بند جاسازی شده را انجام ندهد، اِلدِرَن قادر به شناسایی آن نخواهد بود. باج‌افزار ممکن است رفتار فراریختی از خود نشان دهد، یعنی رفتار مخرب شناخته شده را به شکل دیگری انجام دهد. در این‌صورت، روش پیشنهادی قطعاً سودمندی روش پیشنهادی کاهش می‌یابد.

۳-۹- مدل کمّی برای جلوگیری از باج‌افزار

روش: مسئله اینست که باج‌افزار را ممکن است بتوانیم با نرم‌افزار ضد بدافزار حذف کنیم. اما فایل‌های رمزگذاری شده از دسترس کاربرد خارج می‌شوند. متأسفانه این اثرات مخرب باج‌افزارها باقی می‌ماند و قابل برگشت نیست یا به‌سختی احیاء می‌شود. به‌همین دلیل کیم و کیم[۷۰] [۱۹] مدل کمّی طراحی کردند که عملیات رمزگذاری باج‌افزار را کشف کند و از بروز حمله جلوگیری نماید. از آنجائی‌که یکی از ریشه‌های موفقیت حمله باج‌افزارها، روش‌های «مهندسی اجتماعی»[۷۱] است، مدل کمی ارائه شده رفتارها و ایده‌هایی که در مهندسی اجتماعی استفاده می‌شود را لحاظ کرده است.

آزمایش‌ها و نتایج: هیچ ارزیابی آزمایشی یا صوری برای ارزیابی روش پیشنهادی صورت نگرفته است.

تحلیل روش و نقاط قوت و ضعف: ارائه مدل‌های رسمی و کمّی برای کشف می‌تواند استحکام بیشتری به روش پیشنهادی ببخشد و امکان تحلیل‌های ریاضی قابل اعتماد را فراهم سازد. متأسفانه این مقاله هیچ نوع ارزیابی آزمایشی یا درستی آزمایی صوری نکرده است که از درستی روش پیشنهادی اطمینان پیدا کنیم. ابعاد کارکرد روش پیشنهادی نامعلوم است و هزینه محاسباتی آن و سودمندی در عمل مشخص نیست.

۴- مقایسه فنون، مشکلات موجود و سیر تحقیقات آتی

جدول ۱ نتایج مقایسه روش‌های مورد بررسی را نشان می‌دهد. سطرهای جدول از شماره‌های ۱ تا ۹ به روش‌های مورد بررسی در بخش سوم به‌ترتیب اختصاص دارد. ستون‌های این جدول پنج معیار را نشان می‌دهند که روش‌ها بر اساس آن‌ها با هم مقایسه شده‌اند. این معیارها کیفی هستند و طوری انتخاب شده‌اند که ضمن سادگی، به‌طور دقیق و علمی با مطالعه روش پیشنهادی قابل سنجش باشند. اگرچه تعداد معیارها محدود است، ولی نتایج به‌دست آمده از تحلیل‌ها ارزشمند است. معیار اول نوع تحلیل‌های روش پیشنهادی را مشخص می‌نماید (۱: ایستا؛ ۲: پویا؛ ۳: ترکیبی). معیار دوم نوع سیستم عامل و بن‌سازه اجرایی باج‌افزارها را مشخص می‌سازد (۱: ویندوز و لینوکس؛ ۲: اندروید؛ ۳: هر دو). معیار سوم شیوه ارزیابی روش پیشنهادی را نشان می‌دهد (۱: آزمایش تجربی؛ ۲: صوری؛ ۳: ترکیبی؛ ۴: بدون ارزیابی). معیار چهارم رویکرد اصلی روش را بیان می‌کند (۱: جلوگیری؛ ۲: شناسایی و کشف). در نهایت، معیار پنجم ایده اصلی روش ارائه شده را نشان می‌دهد.

آن‌چنان که از نتایج جدول ۱ مشخص است، همه روش‌ها یا فقط از تحلیل‌های پویا استفاده کرده‌اند یا ترکیب تحلیل پویا و ایستا. این موضوع نشان می‌دهد که تحلیل ایستا برای شناسایی باج‌افزارها سودمندی کافی نداشته است و هر روش مؤثری در آینده باید نوعی تحلیل پویا

70- Donghyu Kim and Seoksoo Kim
71- Social engineering

جدول ۱: ارزیابی روش‌ها با معیارهای چهارگانه

شماره روش/ معیار	تحلیل‌ها	سیستم عامل هدف	ارزیابی	رویکرد	ایده روش ارائه شده
۱	ترکیبی	اندروید	بدون ارزیابی	شناسایی / کشف	ترکیب ویژگی‌های ایستا همچون بسته‌بندی مجدد فایل و اطلاعات پویا همچون نظارت رفتار برنامه و انطباق مسیر داده بحرانی
۲	ترکیبی	اندروید	آزمایش تجربی	شناسایی / کشف	شناسایی عملیاتی مثل قفل‌گذاری، رمزنگاری و پیغام‌های تهدیدآمیز و تحلیل آلودگی
۳	پویا	ویندوز / لینوکس	آزمایش تجربی	شناسایی / کشف	صرفاً تحلیل نمونه‌های موجود
۴	پویا	ویندوز / لینوکس	آزمایش تجربی	شناسایی / کشف	پایش تعاملات با سیستم فایل
۵	پویا	ویندوز / لینوکس	آزمایش تجربی	شناسایی / کشف	بررسی فعالیت مخرب روی فایل
۶	پویا	ویندوز / لینوکس	آزمایش تجربی	شناسایی / کشف	پایش ترافیک خروجی مرحله تبادل کلید با کارساز وب
۷	پویا	ویندوز / لینوکس	آزمایش تجربی	شناسایی / کشف	استخراج ویژگی‌های رفتاری مثل توابع رمزنگاری و رجیستری ویندوز
۸	پویا	ویندوز / لینوکس	آزمایش تجربی	شناسایی / کشف	پایش پویای عملیات زمان اجرا روی فایل‌ها، رجیستری و API
۹	ترکیبی	ویندوز / لینوکس	بدون ارزیابی	جلوگیری	مدل کمّی روی عملیات رمزگذاری باج‌افزار

روی اطلاعات زمان اجرای فایل مشکوک داشته باشد. روش‌هایی که از تحلیل پویا استفاده می‌کنند اطلاعاتی همچون الگوی دسترسی به فایل‌ها، آنتروپی فایل‌ها قبل و بعد از نوشتن، ترافیک شبکه و حوزه‌های اطلاعاتی رد و بدل شده، الگوی استفاده از فراخوانی‌های کاربردی، سیستمی و شبکه‌ای را مورد استفاده قرار داده‌اند. حاصل کار، طراحی روش سودمندی بوده است که باج‌افزارهایی که رفتار غالب آنها، اطلاع مورد بررسی روش بوده است را با دقت بالا شناسایی کرده‌اند. بنابراین روش‌های سودمند آینده باید بر جمع‌آوری اطلاعات پویایی تمرکز کنند که عمومیت بیشتری در بین خانواده‌های مختلف باج‌افزار دارد. این اقدام منجر به روش‌هایی می‌شود که عمومیت بیشتری دارند؛ یعنی هم خانواده‌های بیشتر و هم نمونه‌های ناشناخته را با دقت بالا می‌توانند شناسایی نمایند.

برای نشان دادن سودمندی روش پیشنهادی، روش‌های مورد بررسی از آزمایش تجربی در مقیاس‌های مختلف روی نمونه‌های واقعی انجام داده‌اند. تنها نتایج آزمایش‌های تجربی است که به ما نشان می‌دهد هر روشی چقدر در عمل قابلیت استفاده دارد؛ حتی اگر تحلیل‌های نظری، سودمندی آن را نشان داده باشند. دو مورد از مقالات مورد بررسی هیچ‌گونه ارزیابی نکرده‌اند. این موضوع باعث می‌شود از نتوانیم از صحت ادعاها و درستی عملکرد روش پیشنهادی مطمئن باشیم و هر گونه تفسیر یا نتیجه‌ای قابل استناد نیست.

اغلب روش‌های پیشنهادی ویندوز یا لینوکس را به‌عنوان بن‌سازه اجرا و تحلیل باج‌افزارها انتخاب کرده‌اند. این روش‌ها فوق‌العاده ارزشمندند این دو سیستم عامل، نرم‌افزارهای پایه اکثر سیستم‌های رایانه‌ای امروزی است. با این حال، تحقیقات آتی باید روش‌های مناسبی برای سیستم‌های عامل گوشی‌های موبایل همچون اندروید طراحی کند و سودمندی روش‌ها روی این بن‌سازه‌ها مورد ارزیابی قرار گیرد.

نتیجه‌گیری

رواج رو به رشد باج‌افزارها و فقدان روش‌های جامع و مؤثر نیاز به تحقیقات گسترده و طراحی روش‌های سودمند و کارامد را ضروری می‌سازد. این مقاله با تجمیع مفاهیم باج‌افزارها، روش‌های کشف و ۹ مورد از تحقیقات مطرح اخیر، زمینه مطالعه و تسلط بر کارهای موجود و نقاط قوت و ضعف آن‌ها را فراهم ساخت. محققان حوزه بدافزار با

مطالعه این مقاله مروری قادر می‌شوند تحقیقات خود را در سمت نیاز روز پیش برده و از بهترین یافته‌های مطالعه‌های قبلی برای طراحی روش‌های بهتر بهره‌مند گردند.

مراجع

1- Baysa, Donabelle, Richard M. Low, and Mark Stamp. "Structural entropy and metamorphic malware." Journal of computer virology and hacking techniques9, no. 4 (2013): 179-192.

2- McAfee Labs Threats ReportMarch. Retrieved March 10, 2017, from https://www.mcafee.com/us/security-awareness/articles/mcafee-labs-threats-report-mar-2016.aspx.

3- Scaife, Nolen, Henry Carter, Patrick Traynor, and Kevin RB Butler. "Cryptolock (and drop it): stopping ransomware attacks on user data." InDistributed Computing Systems (ICDCS), 2016 IEEE 36th International Conference on, pp. 303-312. IEEE, 2016.

4-Kharraz, Amin, Sajjad Arshad, Collin Mulliner, William Robertson, and Engin Kirda. "UNVEIL: A Large-Scale, Automated Approach to Detecting Ransomware." In 25th USENIX Security Symposium (USENIX Security 16), pp. 757-772. USENIX Association, 2016.

5-Kharraz, Amin, William Robertson, Davide Balzarotti, Leyla Bilge, and Engin Kirda. "Cutting the gordian knot: A look under the hood of ransomware attacks." In International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, pp. 3-24. Springer International Publishing, 2015.

6-Yang, Tianda, Yu Yang, Kai Qian, Dan Chia-Tien Lo, Ying Qian, and Lixin Tao. "Automated detection and analysis for android ransomware." In High Performance Computing and Communications (HPCC), 2015 IEEE 7th International Symposium on Cyberspace Safety and Security (CSS), 2015 IEEE 12th International Conferen on Embedded Software and Systems (ICESS), 2015 IEEE 17th International Conference on, pp. 1338-1343. IEEE, 2015.

7-Andronio, Nicoló, Stefano Zanero, and Federico Maggi. "HelDroid: Dissecting and detecting mobile ransomware." In International Workshop on Recent Advances in Intrusion Detection, pp. 382-404. Springer International Publishing, 2015.

8-Cohen, Fred. "Computer viruses: theory and experiments." Computers & security 6, no. 1 (1987): 22-35.

9-Chess, David M., and Steve R. White. "Undetectable Computer Viruses." Virus 107 (2000).

10-Idika, Nwokedi, and Aditya P. Mathur. "A survey of malware detection techniques." Purdue University 48 (2007).

11-Lin, Da, and Mark Stamp. "Hunting for undetectable metamorphic viruses."Journal in computer virology 7, no. 3 (2011): 201-214.

12-Aycock, John. Computer viruses and malware. Vol. 22. Springer Science & Business Media, 2006.

13- Wagener, Gérard, and Alexandre Dulaunoy. "Malware behaviour analysis."Journal in computer virology 4, no. 4 (2008): 279-287.

14- Dixit, Nitesh Kumar, Lokesh Mishra, Mahendra Singh Charan, and Bhabesh Kumar Dey. "The new age of computer virus and their detection." International Journal of Network Security & Its Applications 4, no. 3 (2012): 79.

15-Ahmadian, Mohammad Mehdi, and Hamid Reza Shahriari. "2entFOX: A framework for high survivable ransomwares detection." In Information Security and Cryptology (ISCISC), 2016 13th International Iranian Society of Cryptology Conference on, pp. 79-84. IEEE, 2016.

16-Ahmadian, Mohammad Mehdi, Hamid Reza Shahriari, and Seyed Mohammad Ghaffarian. "Connection-monitor & connection-breaker: A novel approach for prevention and detection of high survivable ransomwares." InInformation Security and Cryptology (ISCISC), 2015 12th International Iranian Society of Cryptology Conference on, pp. 79-84. IEEE, 2015.

17-Liska, Allan., Timothy Gallo. Ransomware Defending Against Digital Extortion. O'Reilly Media Inc., 2017.

18-Sgandurra, Daniele, Luis Muñoz-González, Rabih Mohsen, and Emil C. Lupu. "Automated Dynamic Analysis of Ransomware: Benefits, Limitations and use for Detection." arXiv preprint arXiv:1609.03020 (2016).

19-Kim, Donghyun, and Seoksoo Kim. "Design of quantification model for ransom ware prevent." World Journal of Engineering and Technology 3, no. 3 (2015): 203-207.

۲۰- خلیلیـان، علیرضا، و نوشـین وزیندل. «کشـف ویروسـهای فراریخت: دستاوردها و چالشها» علوم رایانشی، شماره اول (۱۳۹۵): ۲۵-۱۳.