مجله علمی - پژوهشی "پدافند الکترونیکی و سایبری"
سال چهارم، شماره ۴، زمستان ۱۳۹۵، ص ۴۶-۲۵

بررسی تحلیلی شبکه‌های بات و روش تشخیص آن‌ها

رضا جلایی*[۱]، محمدرضا حسنی‌آهنگر[۲]

۱- دانشجوی دکتری، ۲- دانشیار، دانشگاه جامع امام حسین(ع)

(دریافت: ۹۵/۰۴/۱۲، پذیرش: ۹۵/۰۸/۱۰)

چکیده

باتنت‌ها امروزه به یکی از تهدیدهای جدی و خطرناک برای امنیت صدها میلیون رایانه در معرض خطر و آلوده در فضای سایبر شناخته شده‌اند. آن‌ها شامل شبکه‌ای از میزبان‌های در معرض خطر هستند که تحت کنترل یک نفوذگر قرار دارند و ریشه اولیه بسیاری از حملات و فعالیت‌های جعلی در اینترنت نظیر حملات منع سرویس توزیع‌شده، فیشینگ، ارسال هرزنامه، دزدی اطلاعات و امثال آن هستند. مطالعه‌های انجام‌شده نشان می‌دهند که بین ۱۶ تا ۲۵ درصد رایانه‌های متصل به اینترنت به بات‌ها آلوده بوده و توسط هکرها، تحت کنترل هستند. مقاله حاضر، در خصوص باتنت‌ها و تحقیقات و مطالعات مرتبط با آن‌ها بحث می‌کند، به‌گونه‌ای که سیر مراحل تکامل این بدافزارها را نشان بدهد. مفاهیمی مثل چرخه عمر، مدل‌های فرمان و کنترل، پروتکل‌های ارتباطی، پروتکل‌های باتنت، روش‌های تشخیص باتنت‌ها و ابزارهای تشخیص در این تحقیق بیان شده‌اند. همچنین حمله‌های متصور توسط باتنت‌ها و نیز آماری از حمله‌های انجام شده تاکنون توسط آن‌ها به‌صورت یک تاریخچه آورده شده است. در ادامه در خصوص چالش‌های موجود در خصوص باتنت‌ها بحث شده است. کارهای توسعه‌ای آینده که قابل ادامه دادن توسط محققین است مانند استفاده از روش‌های پنهان نگاری و کانال‌های پنهان در تشخیص و یا قدرتمند سازی سرویس‌دهنده‌های شبکه بات در انتهای این تحقیق نیز بررسی شده است.

واژه‌های کلیدی: باتنت، مدیر بات، پروتکل‌های ارتباطی، حملات، هانی‌نت، سیستم تشخیص‌نفوذ.

۱- مقدمه

واژه Bot در اصل متشکل از واژه "ro – Bot" بوده [۱] که در مقاله حاضر به شکل فارسی بات نگارش شده و به‌کار برده می‌شود. بات مفهومی است که برای توضیح یک اسکریپت یا مجموعه‌ای از اسکریپت‌ها طراحی شده است و می‌تواند در انجام یکسری توابع خودکار شکل‌دهی شود. بانتنت مجموعه‌ای از بات‌ها یا مجموعه‌ای از رایانه‌های آسیب‌پذیر است که از راه دور به‌وسیله مدیر بات کنترل می‌شود [۲].

به‌طور عام باتنت برای نامیدن شبکه‌ای از رایانه‌های آلوده به‌کاربرده می‌شود که به آن بات‌ها[۱] گفته می‌شود. بات‌ها توسط یک اپراتور انسانی کنترل می‌شوند که معمولاً مدیر بات[۲] نامیده می‌شود. بات‌ها از ماشین‌های آلوده برای اجرای دیگر بدافزارها مثل بهره‌برداری[۳] راه دور از آسیب‌پذیری‌های نرم‌افزاری، مهندسی اجتماعی و غیره استفاده می‌کنند [۳]. این کار معمولاً از طریق ایجاد یک زیرساخت سرویس‌دهنده‌های فرمان و کنترل[۴] بین ماشین‌های آلوده و مدیر بات انجام می‌گیرد.

به‌طورکلی اختلاف اصلی بین باتنت و دیگر بدافزارها، وجود این سرویس‌دهنده[۵]های فرمان و کنترل است. بنابراین، روش‌های کشف نیز تمرکز بر تشخیص و یافتن این سرویس‌دهنده‌ها دارند. به‌عبارت دیگر قدرت یک بات ارتباط مستقیم با قدرت سرویس‌دهنده‌های فرمان و کنترل آن دارد.

امروزه باتنت‌ها، به تهدیدی جدی در عرصه اینترنتی تبدیل‌شده‌اند [۴]. تقریباً بین ۱۶ تا ۲۵ درصد از رایانه‌های متصل به اینترنت به باتنت‌ها متصل بوده و عضوی از آن‌ها محسوب می‌شوند [۵]. چنین شبکه‌هایی برای اجرای فعالیت‌های غیرقانونی در ابعاد گسترده طراحی و آماده می‌شوند و این گستره چنان وسیع است که می‌تواند فعالیت سرویس‌های عمومی و خصوصی را در کشورهای مختلف به خطر اندازد. مهاجم یا به‌عبارتی همان مدیر بات، کنترل ماشین‌ها را به دست گرفته و

* رایانامه نویسنده مسئول: Rjalaei@ihu.ac.ir

1 -Bots
2 -Bot Master or Bot herder
3-Exploiting
4-C&C
5 -Servers

اقدام به فعالیت‌های غیرقانونی و جنایی مثل سرقت اطلاعات یا هویت افراد، حملات برای قطع خدمات و سرویس‌ها، ارسال پیام‌های ناخواسته و دیگر فعالیت‌های غیرقانونی می‌کند [۶].

برخی گزارش‌ها نشان می‌دهند که نزدیک به ۸۰ درصد از حجم انتقال و ترافیک ایمیل‌ها را هرزنامه[1]ها به خود اختصاص داده‌اند و این در حالی است که اغلب این پیام‌ها با استفاده از بات‌نت‌هایی چون Grum، Cutwail و Rustock ارسال می‌شوند [۷].

توجه به این نکته مهم است که بات‌نت‌ها اهدافی متحرک هستند که ممکن است با گذشت زمان تمام ابعاد چرخه عمر آن‌ها تغییر و تکامل یابد و هیچ‌یک از روش‌های ردیابی، تشخیص و یا کاهش اثر برای همیشه مؤثر نباشند، زیرا بازیگران مختلفی از قبیل دولت‌ها، شبکه‌های خصوصی و شرکتی و سرویس‌دهندگان اینترنت[2] با اهداف و تفکراتی گوناگون مسائل مربوط به بات‌نت‌ها را موردبررسی قرار می‌دهند. در این مقاله، مروری از فناوری‌های جاری شکل دهنده بات‌نت‌ها ارایه می‌شود.

تحقیق‌های [۹ -۸] چرخه عمر بات و تکنیک‌های ردیابی بات‌نت‌ها را پوشش می‌دهند. ژو[3] و همکارانش در [۸] یک طبقه‌بندی برای انواع بات‌ها، واحدهای اندازه‌گیری ابعاد بات‌نت‌ها و راه‌حل مناسب و اقدامات متقابل علیه هرزنامه‌ها را مشخص کرده‌اند. مرجع [۹] ساختارهای فرمان و کنترل را به دو صورت متمرکز و غیرمتمرکز دسته‌بندی کرده و تکنیک‌های تشخیص بات‌نت‌ها را با تفکیک بیشتری به انواع مبتنی بر ناهنجاری[4]، مبتنی بر DNS و مبتنی بر کاوش[5] تقسیم‌بندی می‌کند. این مطالعه همچنین درباره ابعاد خاصی از تحقیقات انجام‌شده درزمینه بات‌نت توضیحی کوتاه ارائه می‌دهد که مراحل رشد و آینده بات‌نت‌ها، سرعت و روش تکثیر، پیچیدگی‌های طراحی، قابلیت ردیابی و اندازه آن‌ها را شامل می‌شود.

شین[6] و ایم[7] در مرجع [۱۰] توضیحات مختصری در خصوص پیکربندی و فعالیت‌های بات‌نت‌ها ارائه کرده‌اند. اگرچه این توضیحات مباحث مفید و جالبی درباره نتایج، پشتیبانی، دفاع و چالش‌های مرتبط با بات‌نت‌ها و حملات از کار اندازی سرویس[8] را مطرح می‌کنند، اما موضوعاتی که در این مطالعه بررسی می‌شود تا حدی محدود است و پوشش جامعی از مطالعات پیشین و مرتبط ارائه نمی‌دهد.

در مرجع [۹]، مطالعه‌ای در خصوص بات‌نت‌ها انجام‌شده است که در آن نویسندگان، رشد و توسعه بات‌نت‌ها را مطابق با سیر تحول تاریخی بدافزارهای گوناگون که از سال ۱۹۹۳ تا سال ۲۰۰۷ استفاده شده‌اند را شرح می‌دهند. نویسندگان، در این مطالعه، روش‌های بات‌نت را به شکل زیر دسته‌بندی کرده‌اند، روش‌های آلوده‌سازی، رفتار بدخواهانه، مدل‌های فرمان و کنترل، پروتکل‌های ارتباطی، تشخیص بات‌نت‌ها و دفاع در برابر آن‌ها.

مقاله [۱۱] در پژوهشی موردی به بررسی SpyBot با استفاده از خصوصیات اصلی آن می‌پردازد که این خصوصیات عبارت‌اند از، آلوده‌سازی، ساختار فرمان و کنترل، و روش فعالیت‌ها و حملات.

مطالعه و بررسی موجود در [۱۲] درباره ایجاد و استفاده از بات‌نت‌ها و چرخه زندگی بات است و بات‌های مبتنی بر IRC و P2P را توصیف می‌کند. این مطالعه همچنین ویژگی‌های بات‌های معروفی از قبیلAgobot، SDBot ، SpyBotو GT Bot را توصیف کرده و درباره فعالیت‌های بدخواهانه و غیرقانونی بحث می‌کند که با استفاده از بات‌ها اجرایی می‌شوند. این مطالعه صرفاً به بررسی و بحث مرتبط با تشخیص بات‌نت‌ها می‌پردازد.

گزارش فنی ارائه‌شده در [۱۳]، بات‌نت‌ها را با توجه به تکنیک‌های اندازه‌گیری و تشخیص، اقدامات متقابل، و توصیه‌های عملکردی مناسب موردبحث و بررسی قرار می‌دهد و در ادامه به توصیف گرایش‌های محتمل در آینده می‌پردازد.

در [۱۴]، نویسندگان یک بات‌نت نظیر به نظیر[9] ترکیبی را موردبحث قرار داده‌اند که به‌عنوان یک بات پیشرفته عمل می‌کند و به‌گونه‌ای قدرتمند است که حتی اگر تعدادی از بات‌ها حذف شوند به کار خود ادامه داده، همبندی خود را حفظ می‌کند. این بات می‌تواند به‌راحتی با مدیر بات خود ارتباط برقرار کرده و در مقابل تشخیص ابزارهای دفاعی شبکه از طریق الگوهای ترافیک ارتباطی خود مقاومت کند. این بات‌نت به دو بخش خدمتکار[10] و سرویس‌گیرنده[11] تقسیم می‌شود. بات‌های خدمتکار برای ارتباط با دیگر گره‌های بات‌های خدمتکار که دستورات به سمت آن‌ها ارسال می‌شوند؛ از جداول مسیریابی استفاده می‌کند. دلیل این قدرتمندی در پایداری شبکه، این ایده است که هر گره صرفاً در خصوص تعداد کمی از همسایه‌های خود اطلاعات دارد. همچنین برای ارتباط‌های بین میزبان‌های خدمتکار از یک پروتکل رمز

1 -Spam
2 -ISP(Internet service Provider)
3 -Zhu
4-Anomaly - Based
5-Mining - Based
6 -Shin
7 -Im
8-DDoS
9- Peer to Peer
10 -Servant
11 -Client

سفارشی استفاده می‌کند.

در [۱۵]، آقای گاراسیا و همکارانش یک روش تشخیص باتنت HTTP برپایه کاوش الگوهای متوالی ارائه کرده‌اند. مقاله حاضر از یک روش جدید برای تشخیص باتنت استفاده می‌کند که مبتنی بر الگوریتم اپرایوری[1] و مهر زمانی است. این مقاله همچنین یک روش تشخیصی ارائه کرده است که نیاز به اطلاعات اولیه از باتنت مثل امضاها ندارد.

در مرجع [۱۶]، محمد مسعود و همکارانش یک روش دسته‌بندی داده‌های جریانی[2] هوشمند برای تشخیص باتنت‌های نظیر به نظیر ارائه کرده‌اند. آن‌ها یک روش داده‌کاوی مبتنی بر دسته‌بند رای چند سطحی[3] پیشنهاد کرده‌اند که برای دسته‌بندی داده‌های جریانی با محتوای متغیر[4] کاربرد دارد. این مقاله با تقریب خوبی نسبت به روش‌های دیگر کارایی خود را اثبات کرده است، همچنین نقطه قوت آن، استفاده از داده‌های واقعی در حین کار بوده است. ایده اصلی مقاله که موجب افزایش کارایی آن شده است، ذخیره دسته‌بندهای آموزشی بجای ذخیره داده ثابت است.

ادامه مقاله بر این اساس سازمان‌دهی شده که در بخش ۲، تاریخچه باتنت‌ها بیان شده است. در بخش ۳ عملکرد باتنت‌ها مورد بررسی قرار می‌گیرد. بخش ۴، دسته‌بندی و معماری باتنت‌ها را به‌تفصیل شرح می‌دهد. پروتکل‌های باتنت در بخش ۵ برشماری شده و حملات صورت گرفته توسط باتنت‌ها در بخش ۶ ارائه می‌شود. بخش ۷ روش‌های تشخیص باتنت‌ها را فهرست کرده است. روش‌های گریز باتنت‌ها در بخش ۸ ارائه شده و در بخش ۹ موضوعات قابل تحقیق برای آینده و برخی چالش‌ها در خصوص باتنت‌ها فهرست می‌شوند. در نهایت در بخش ۱۰، نتیجه‌گیری مقاله بیان می‌شود.

۲- تاریخچه و مفهوم باتنت‌ها

نقطه شروع باتنت‌ها را می‌توان IRC[5] نامید که یک سیستم گفتگوی مبتنی بر متن است و برقراری ارتباط و گفتگوی افراد را از طریق کانال‌هایی سازمان‌دهی می‌کند [۱۷]. ایده اصلی در ایجاد و استفاده از باتنت‌ها کنترل ارتباطات و تعاملات در اتاق‌های گفتگوی IRC بوده است. باتنت‌ها می‌توانستند فرمان‌های ساده را بفهمند، پشتیبانی اجرایی فراهم کنند، بازی‌های ساده و دیگر خدمات را در اختیار کاربران اتاق‌های گفتگو قرار دهند و اطلاعاتی درباره سامانه‌های در حال فعالیت، افرادی که وارد سیستم شده‌اند، آدرس پست‌های الکترونیکی، اسامی مستعار، و غیره را به دست آورند.

Eggdrop، اولین بات IRC شناخته‌شده، در سال ۱۹۹۳ منتشر شد [۱۸] و پس‌ازآن نیز توسعه یافت. پس از آن بود که باتهای IRC مخرب در محیط اینترنت ظاهر شدند که با استفاده از همان ایده اولیه به وجود آمده بودند. هدف اصلی در طراحی این باتها حمله به کاربران IRC یا حتی حمله به تمام سرویس‌دهنده‌ها بود. کمی بعد، حملاتی برای عدم ارائه یا قطع خدمات و سپس قطع وسیع و توزیع‌شده خدمات به توانایی این باتها اضافه شد. در ادامه و با پیشرفت فناوری، باتهای جدیدی به وجود آمدند که برای برقراری ارتباط با مدیر بات از روش‌های پیچیده استفاده می‌کردند و از دیگر پروتکل‌های موجود نیز بهره می‌بردند. این باتها از روش‌های تهاجمی جدید، منسجم، و قدرتمند استفاده می‌کردند و روی‌هم‌رفته همین ویژگی‌ها بود که باتها را به ابزاری پیچیده و قدرتمند تبدیل کرد. باتها می‌توانستند مثل کرم‌ها تکثیر شوند، مانند ویروس‌ها مخفی بمانند، و حملاتی گسترده و هماهنگ را انجام دهند. برای اشاره به چنین باتهایی می‌توان به AgoBot [۱۹] و SDBot [۲۰] اشاره کرد.

از زمان به وجود آمدن AgoBot و دیگر انواع آن بود که باتها به‌عنوان تهدیدی جدی در محیط اینترنت مطرح شدند [۲۱]. نسل کنونی باتها می‌توانند در میان شبکه‌های اشتراک اطلاعات، شبکه‌های نظیر به نظیر، فایل‌های ارسال‌شده از طریق پست الکترونیک و وب‌سایت‌های آلوده منتشر شوند، یا ممکن است از قبل به‌صورت مخفیانه نصب‌شده باشند. جدول (۱) دوره زمانی تعدادی از مهم‌ترین باتها و ویژگی‌های آن‌ها را نشان می‌دهد.

باتنت‌ها شبکه‌هایی هستند که توسط رایانه‌های میزبان اسیرشده، شکل‌دهی شده‌اند و درنهایت شبکه بات نامیده می‌شوند. بات که مشتق شده از کلمه روبات است توسط یک یا چند مهاجم که مدیر بات نام دارند کنترل می‌شود و برای انجام فعالیت‌های بدخواهانه و غیرقانونی مورداستفاده قرار می‌گیرد [۶]. به‌عبارت دیگر باتها همان کدهای مهاجم و خراب‌کاری هستند که بر روی رایانه‌های میزبان فعالیت می‌کنند و به مدیران بات اجازه می‌دهند از راه دور رایانه‌های میزبان را کنترل کنند و با استفاده از آن‌ها به فعالیت‌های مختلف بپردازند.

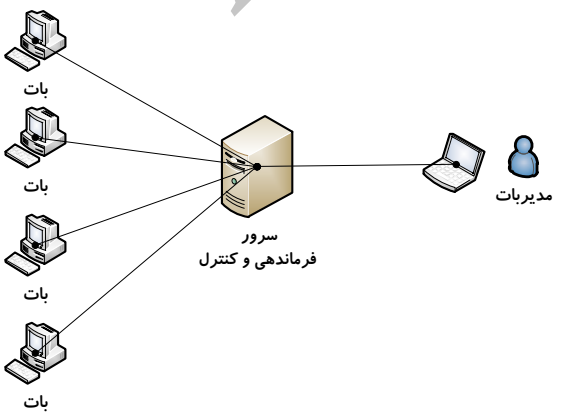
1 -Apriori algorithm
2 -Stream
3 -multi-level ensemble classifier
4 -Concept – drifting data streaming
5 -Internet relay chat

جدول (۱). دوره‌های زمانی ایجاد مهم‌ترین بات‌ها و معماری آن‌ها

ردیف	سال انتشار	نام بات	معماری یا پروتکل
۱.	۱۹۹۳	EggDrop	متمرکز/ IRC
۲.	۱۹۹۸	GTbot	متمرکز
۳.	۲۰۰۲	SDbot	متمرکز/ IRC
		Agobot	
۴.	۲۰۰۳	Spybot	متمرکز
		Sinit	غیرمتمرکز/P2P
۵.	۲۰۰۴	Bagle	متمرکز
		Forbot	متمرکز
۶.		Phatbot	غیرمتمرکز/P2P
۷.	۲۰۰۶	SpamThru	غیرمتمرکز/P2P
۸.		Nugache	غیرمتمرکز/P2P
۹.		Jrbot	متمرکز
۱۰.		Rxbot	متمرکز/ IRC
۱۱.		Rustock	متمرکز/ HTTP
۱۲.	۲۰۰۷	Storm	متمرکز
		Peacomm	غیرمتمرکز/P2P
		Pushdo	متمرکز/ HTTP
		Srizbi	متمرکز/ HTTP
		Zeus/Zbot	متمرکز/ HTTP
		Mega-D	غیرمتمرکز/P2P
۱۳.	۲۰۰۸	Lethic	متمرکز
		Asprox	متمرکز/ HTTP
		Bobax	متمرکز/ UDP/HTTP
		Kraken	متمرکز
		Torpig	متمرکز
		Conficker	غیرمتمرکز/P2P
۱۴.	۲۰۰۹	Waledac	غیرمتمرکز/P2P
		Donbot	متمرکز/ TCP
۱۵.	۲۰۱۰	Festi	متمرکز/ HTTP
۱۶.	۲۰۱۱	TDL-4	غیرمتمرکز/P2P
۱۷.		Zeroaccess	غیرمتمرکز/P2P
۱۸.	۲۰۱۲	Flashfake	غیرمتمرکز/P2P
۱۹.	۲۰۱۳	Boatnet	غیرمتمرکز

اجزای تشکیل‌دهنده بات‌نت‌ها همگی یکسان نیستند، شکل (۱) این اجزا را کنار هم چیده است. بات‌نت برنامه‌ای نرم‌افزاری است که بر روی میزبان آسیب‌پذیر نصب می‌شود و می‌توان آن را به روش‌های مختلف روی ماشین‌های قربانی نصب کرد که از میان آن‌ها می‌توان به روش‌های مهندسی اجتماعی، انتشار ویروس و یا وب‌سایت‌های آلوده اشاره کرد. آن‌ها معمولاً به‌نحوی برنامه‌ریزی می‌شوند که با هر بار راه‌اندازی رایانه‌های قربانی شروع به کار می‌کنند.

میزبان‌های آسیب‌پذیر ماشین‌هایی در اینترنت هستند که به بدافزارها متصل شده‌اند. این بدافزارها توسط مدیر بات با روش‌های تکثیر در اینترنت منتشر می‌شوند و پس از آلوده شدن، این ماشین‌ها تبدیل به قربانی[1] یا برده[2] می‌شوند و ممکن است از آن‌ها به‌عنوان سکویی برای حمله به دیگر میزبان‌های آسیب‌پذیر یا حمله برای قطع خدمات استفاده شود.



شکل (۱). اجزای یک شبکه بات به‌طور عام [۲۲]

1 -Zombie
2- Slaves

حیاتی‌ترین جزء یک بات‌نت، زیرساخت فرمان و کنترل آن است که شامل بات‌ها و یک بخش کنترل است که می‌تواند به‌صورت متمرکز یا غیرمتمرکز مورداستفاده قرار گیرد. برای فرمان دادن به رایانه قربانی و هماهنگ کردن فعالیت آن‌ها، مدیر بات از یک یا چند پروتکل ارتباطی استفاده می‌کند. با توجه به نوع استفاده‌ای که برای یک بات‌نت در نظر گرفته می‌شود، به همان نسبت مجموعه دستورات و کارکرد آن‌ها نیز بسیار گسترده و متنوع هستند.

Rustock، بزرگ‌ترین بات‌نت شناخته‌شده در سال ۲۰۱۰، بیش از یک‌میلیون بات را تحت کنترل خود داشته است [۲۳]. در یک بازه زمانی پنج‌ماهه، پس از بررسی و رصد نزدیک به ۱۸۰ بات‌نت، بیش از ۳۰۰۰۰۰ آدرس IP منحصر به‌فرد را ردیابی کرد که حداقل به یکی از کانال‌های بات‌نت تحت‌نظر متصل بودند. محققان این‌چنین نتیجه گرفتند که بیش از یک‌میلیون ماشین از مسیر فعالیت‌های اولیه خود منحرف‌شده و از راه دور توسط مدیر بات کنترل می‌شدند.

با توجه به مطالعات فرلینگ[1] و همکارانش [۲۴]، می‌توان گفت که دفاع واکنشی[2] رایج‌ترین نوع دفاع است. در این روش، راه‌کار اصلی این است که در ابتدا فعالیت مخرب و بدخواهانه رهگیری شود و سپس با تلاش برای کاهش سطح ترافیک این فعالیت، به‌میزان قابل قبولی در برابر آن واکنش نشان داده شود. اما استفاده از این رویه دونقطه‌ضعف اساسی دارد: نخست این‌که این روش نیاز به ایجاد زیرساختی باقدرت محاسبه‌گری بالا و ذخیره اطلاعات زیاد برای تحلیل (ترجیحاً به‌طور هم‌زمان) حجم زیادی از اطلاعات تحت کنترل دارد. مشکل دوم مربوط به زمان‌بندی است، ازآنجاکه حمله پیش از آن‌که ردیابی شود آغازشده است، کاربران و سرویس‌دهندگان خدمات در برابر آن تا حدی آسیب‌دیده‌اند. اما از سویی دیگر، روش‌های پیشگیرانه یا پیش‌کنش‌گرانه[3] تلاش می‌کنند مانع از اجرای حملات احتمالی به‌ویژه حملات از کار اندازی توزیع‌شده[4] و ارسال هرزنامه‌ها شوند، یا به قربانی کمک کنند تا از دام حملات خارج شوند که این کار را برای مثال با افزایش منابع قربانی، تغییر و اصلاح زیرساخت‌های شبکه و یا احراز هویت کاربران انجام می‌دهند.

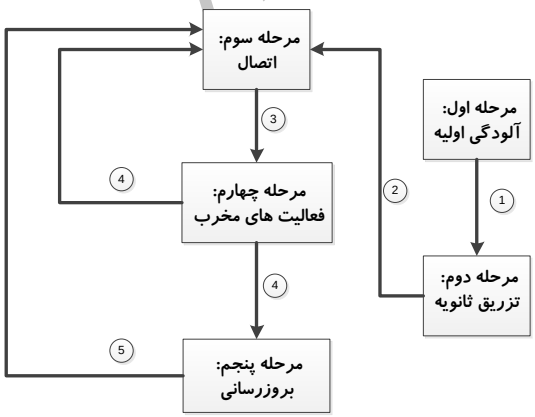
در سال‌های اخیر، شمار روزافزونی از مطالعات برای دست‌یابی به چگونگی ردیابی بات‌نت‌ها و از کار انداختن آن‌ها انجام‌شده است. خطرات و تهدیدات بات‌نت‌ها به‌تازگی در حال مشخص شدن هستند. محققان، مقامات قضایی، کاربران انفرادی و تجار، به‌تازگی به‌دنبال بحث و تبادل‌نظر پیرامون روش‌های مقابله با بات‌نت‌ها هستند که این خود نشانگر این است که احتمالاً در جامعه اینترنتی امروز، بات‌نت‌ها به‌عنوان بزرگ‌ترین تهدید امنیتی مطرح هستند [۲۵].

۳- عملکرد بات‌نت‌ها

فعالیت یک بات با دستوراتی مشخص که مدیر بات از طریق کانال فرمان و کنترل برای آن‌ها ارسال می‌کند آغاز می‌شود. نکته قابل‌توجه این است که بات‌ها نقاط ضعف سامانه‌ها یا برنامه‌های کاربردی نیستند، بلکه برنامه‌هایی هستند که توسط کرم‌ها منتشر می‌شوند یا برای نصب درب‌های پشتی از آن‌ها استفاده می‌شود. آنچه بات را از سایر بدافزارها متمایز می‌کند، کانال فرمان و کنترل آن است. زیرساخت فرمان و کنترل، به‌عنوان تنها راه کنترل بات‌ها در یک شبکه بات به‌حساب می‌آید، از این‌رو حفظ اتصالی مطمئن درون این زیرساخت مشخص می‌کند که شبکه بات به چه میزان قدرتمند و پایدار است و زمان واکنش آن به چه اندازه است.

همان‌طور که ذکر شد، بات‌ها برنامه‌های کاربردی[1] هستند که مانند نسل‌های پیشین ویروس‌ها و کرم‌ها خودبه‌خود تکثیر می‌شوند تا بتوانند میزبان‌های آسیب‌پذیر را آلوده کنند. مطابق نتایج به‌دست‌آمده در مطالعه رجب[5] و همکارانش [۲۶]، بات‌ها برای افزایش تعداد خود، مجهز به چندین نوع مسیرهای تکثیر مدرن هستند. مدیران بات اغلب به دنبال قربانیانی هستند که ویژگی‌های مطلوبی ازجمله سرعت انتقال بالا، دسترسی آسان، سطح امنیتی پایین، میزان کنترل و مانیتورینگ پایین، دوری مسافت و پراکندگی را داشته باشند [۲۷].

برای تبدیل یک میزبان آلوده به یک بات فعال و در نتیجه بخشی از یک بات‌نت شدن، میزبان باید یک سری مراحل را پشت سر بگذارد که به چرخه عمر بات‌نت معروف است [۹ -۸]. این چرخه در شکل (۲) نشان داده شده است.



شکل (۲). چرخه عمربات نت ها به طور عام [۱۷]

1 - Freiling
2- Reactive
3- Proactive
4-DDoS

5- Rajab at al.

اولین مرحله، تزریق اولیه است. زمانی که میزبان، آلوده‌شده و به یک بات محتمل یا شبه بات تبدیل می‌شود. این مرحله با یک فرآیند عادی آلوده شدن رایانه آغاز می‌شود که می‌توان آن را همانند آلوده‌کردن با ویروس‌ها به روش‌های گوناگون، برای مثال از طریق دریافت ناخواسته بدافزار از وب‌سایت‌ها، فایل‌های آلوده متصل به ایمیل‌ها، حافظه‌های قابل‌حمل آلوده و مواردی از این قبیل اجرایی کرد [۹ -۸]. مرحله دوم یا همان تزریق دوم مستلزم این است که مرحله اول با موفقیت به پایان رسیده باشد. در این مرحله، میزبان آلوده، برنامه‌ای را اجرا می‌کند که در پایگاه داده شبکه به دنبال کدهای اجرایی بدافزار خواهد بود. پس از واکشی و اجرای این کدهای اجرایی، میزبان به یک بات واقعی یا زامبی تبدیل می‌شود. به‌طور معمول دریافت کدهای اجرایی بات‌ها از طریق پروتکل‌های FTP, HTTP و یا شبکه‌های نظیر به نظیر صورت می‌گیرد [۹ -۸]. در خلال چرخه عمر و در برخی اوقات، بات جدید باید برای دریافت دستورات و به‌روزآوری‌های جدید با سرویس‌دهنده فرمان و کنترل خود ارتباط برقرار کند. یکی شدن[1] [۲۸] به فرآیندی اطلاق می‌شود که در آن اتصالی با سرویس‌دهنده فرمان و کنترل برقرار می‌شود. برخی نویسندگان این مرحله را مرحله اتصال[2] می‌نامند [۲] . در واقع، این مرحله طوری برنامه‌ریزی‌شده که با هر بار راه‌اندازی مجدد میزبان، به کار می‌افتد تا بدین ترتیب به مدیر بات اطمینان دهد که بات نقش خود را در شبکه بات، به‌خوبی ایفا می‌کند و آمادگی لازم را برای دریافت دستورات اجرای فعالیت‌های مخرب دارد. بنابراین، ممکن است مرحله اتصال چندین بار در طول چرخه حیات بات تکرار شود [۲۹].

ازآنجاکه بات‌ها باید در طول این مرحله با سرویس‌دهنده فرمان و کنترل در ارتباط باشند، ممکن است در این مرحله آسیب‌پذیر شوند. در اغلب موارد بات‌ها به‌صورت پیش‌فرض با فرمان و کنترل ارتباط برقرار می‌کنند و با این کار، روش‌هایی ایجاد می‌شود که از طریق آن‌ها می‌توان الگوهای ترافیکی را شناسایی کرده و به‌این‌ترتیب اجزای بات‌نت یا حتی سرویس‌دهنده فرمان و کنترل تشخیص داده می‌شود.

تزریق دوم و مراحل اتصال، به یکدیگر مرتبط هستند و طبق نظر برخی نویسندگان ممکن است به شکل یک مرحله باشند. برای مثال، اگر سرویس‌دهنده فرمان و کنترل به‌عنوان مخزنی[3] از کدهای اجرایی باشد، به‌احتمال‌زیاد هر دو مرحله هم‌زمان باهم رخ خواهند داد.

برای یافتن مخزن کدهای اجرایی یا سرویس‌دهنده فرمان و کنترل قربانی، بدافزاری که در مرحله آلوده شدن ابتدایی نصب‌شده است و یا خود بات باید آدرس ماشین‌ها را در اختیار داشته باشد. این آدرس‌ها می‌تواند به‌صورت مستقیم به شکل فهرستی از آدرس‌های ثابت IP یا با استفاده از فهرستی از اسامی دامنه‌های ثابت یا متغیر رمزگذاری شوند که از کار انداختن کانال فرمان و کنترل را مشکل می‌سازند. بدیهی است که این کار، به کنترل درآوردن یا مسدود کردن یک سرویس‌دهنده فرمان و کنترل را مشکل‌تر می‌کند و استفاده تنها از یک نام دامنه ثابت به‌طور مستمر موجب شکل‌گیری نقطه شکست[4] خواهد شد.

پس از برپایی کانال فرمان و کنترل، بات منتظر دستورات باقی می‌ماند تا با دریافت آن‌ها فعالیت‌های مخرب را آغاز کند [۹ -۸]. بنابراین، بات وارد مرحله ۴ شده و برای اجرای حمله آماده است. در این زمان، ممکن است رد و بدل شدن پیام‌ها با شدت بیشتری انجام شود و در مدت کوتاهی چندین پیام جابه‌جا شود. به‌هرحال، ترافیک سرویس‌دهنده فرمان و کنترل از حجم بالایی برخوردار نیست و پنهان‌کاری زیادی در شبکه ایجاد نمی‌کند. ازاین‌رو، روش‌های مبتنی بر ناهنجاری ممکن است قادر به شناسایی ترافیک فرمان و کنترل بات‌نت نباشند [۹].

آخرین مرحله چرخه حیات بات؛ حفظ، نگهداری و به‌روزآوری بدافزارها است. اگر مدیر بات بخواهد ارتش زامبی‌های خود را حفظ کند، به‌روزآوری کدها ضروری است. مدیر بات از این کار برای فرار از روش‌های تشخیص، اضافه کردن ویژگی‌های جدید یا استخدام یک سرویس‌دهنده فرمان و کنترل جدید استفاده می‌کند [۱۹ -۱۸]. این مرحله معمولاً مرحله‌ای آسیب‌پذیر است، زیرا هنگامی‌که مدیر بات بخواهد به‌سرعت به‌روزآوری‌ها را منتشر کند، برخی الگوهای رفتاری ایستگاه‌های متعلق به شبکه ممکن است ظاهرشده و بات‌نت را مشخص و قابل‌ردیابی کند. پس‌ازآن‌که بات‌ها به‌روزآوری شدند، باید اتصال‌های جدیدی با زیرساخت فرمان و کنترل برقرار کنند.

۴- دسته‌بندی و معماری بات‌نت‌ها

برای دسته‌بندی بات‌نت‌ها می‌توان از روش‌های مختلفی استفاده کرد، ولی آنچه از مستندات و مقاله‌ها استخراج می‌شود به‌طور عمده بیان می‌دارد که تقسیم‌بندی بات‌نت‌ها بر اساس معماری فرمان و کنترل آن‌ها که مهم‌ترین جز یک شبکه بات است صورت می‌گیرد [۳۰]. بنابراین، این بخش بر معماری بات‌نت‌ها تمرکز می‌کند. لازم به‌ذکر است که می‌توان این دسته‌بندی را بر اساس پروتکل‌های ارتباطی هم انجام داد، ولی به جهت نقش تعیین‌کننده‌ای که پروتکل‌های ارتباطی دارند، ترجیح داده شد که آن‌ها در بخش جداگانه‌ای، بخش ۵، مورد بحث قرار گیرند.

اگر بات‌نت‌ها عمیق‌تر مورد بررسی قرار گیرند ، آنچه فارغ از شیوه‌های ارتباطی، عملیاتی و یا کارکردی موجب تفکیک آن‌ها می‌شود، معماری بات‌نت‌ها است. در ادامه، این معماری‌ها در چهار

1 -Rallying
2 -Connection Phase
3 -Repository
4 -Point of failure

بخش متمرکز، غیرمتمرکز، ترکیبی و تصادفی مورد بحث قرار می‌گیرند.

۴-۱- مدل فرمان و کنترل متمرکز

فرمان و کنترل متمرکز مشابه رویکرد مدل رایج مشتری-سرویس‌دهنده در مفاهیم شبکه‌ای است که در بات‌نت‌های مبتنی بر پروتکل IRC از آن‌ها استفاده می‌شود. در زیرساخت فرمان و کنترل متمرکز، تمامی بات‌ها کانال ارتباطی خود را با یک نقطه اتصال یا تعداد اندکی از این نقاط برقرار می‌کنند. این نقاط معمولاً سرویس‌دهنده‌های فرمان و کنترل هستند که مسئول ارسال دستورات به بات‌ها و به‌روزآوری آن‌ها هستند. مزیت این ساختار، زمان عکس‌العمل پایین و هماهنگی مناسب است. بازخورد مستقیم به مدیر بات اجازه می‌دهد موقعیت بات‌نت را به‌آسانی بررسی و کنترل کند و در خصوص برخی ویژگی‌های حیاتی آن، اطلاعات مهمی شامل تعداد بات‌های فعال یا میزان انتشار آن‌ها به دست آورد. اما مشکل اصلی در ساختار متمرکز این است که سرویس‌دهنده فرمان و کنترل خود یک نقطه شکست مرکزی است [۳۰] که این موضوع به‌راحتی موجب خاموش شدن یک شبکه بات می‌شود.

پروتکل‌های اصلی ساختارهای متمرکز، IRC و HTTP هستند. در بات‌نت‌های IRC، مدیر بات کانال‌های IRC را روی سرویس‌دهنده فرمان و کنترل ایجاد می‌کند. در ادامه، زامبی‌ها به آن متصل شده و منتظر دریافت دستورات برای انجام عملیات مخرب می‌شوند. پروتکلHTTP عموماً برای محدودیت‌های ترافیک IRC در شبکه‌ها به‌کارگیری می‌شود و مزیت اصلی آن در مقابل IRC این است که در بیشتر شبکه‌ها، ترافیک HTTP پذیرفته می‌شود و ارتباط میان بات و مدیر بات را پنهان می‌کند.

۴-۲- مدل فرمان و کنترل غیرمتمرکز

در مرجع [۳۱] ساختار نامتمرکز و متغیر برای کانال فرمان و کنترل برای اولین بار موردبررسی قرار گرفته است. از هم جدا کردن بات‌نت‌هایی که از ساختار غیرمتمرکز استفاده می‌کنند دشوارتر است، زیرا سرویس‌دهنده فرمان و کنترل مرکزی وجود ندارد که کشف و از کار انداخته شود، ازاین‌رو پیدا کردن چند و یا حتی تعداد زیادی بات، به‌معنی از دست رفتن تمام شبکه بات نیست.

چنین بات‌نت‌های غیرمتمرکزی معمولاً مبتنی بر طیفی از شبکه‌های نظیر به نظیر هستند. ایده اصلی به‌کارگیری این شبکه‌ها توسط بات‌نت‌ها، استفاده از توانایی بالای شبکه‌های نظیر به نظیر برای ارتباط باهم، در عین عدم استفاده از سرویس‌دهنده مرکزی بوده است. بات‌نت‌های نظیر به نظیر دارای نقاط ضعفی هستند که برخی از آن‌ها شامل موارد زیر است، وابستگی زیاد به گره‌های قبلی یا گره‌های هم‌جوار، در صورت وجود اتصال ضعیف به‌راحتی ردیابی و کشف می‌شود، در مقایسه با بات‌نت‌های HTTP، رمزنگاری و احراز هویت قوی ندارد و درنهایت برای محدوده آدرس آی‌پی‌های زیاد، ساختار شبکه، پیچیده شده و ترافیک زیادی تولیدی می‌کنند [۳۲].

نظیرهای غیرساخت‌یافته[1]: این نظیرها با پیکربندی‌های تصادفی با درجات مختلف انتشار به وجود آمده و اجازه مسیریابی را نمی‌دهند. این لایه‌ها روش‌های سیل‌آسا و random walk و انواع دیگر همین روش‌ها را برای جستجو به کار می‌گیرند.

نظیرهای ساخت‌یافته: در این نظیرها، مسیردهی میان محتوا و مکان آن شکل می‌گیرد و برای مسیریابی معمولاً از یک جدول درهم‌سازی توزیع‌شده[2] استفاده می‌شود [۳۳] . یک الگوریتم مبتنی بر این جدول درهم‌سازی توزیع‌شده است که در Overnet، eMule و BitTorrent مورداستفاده قرار گرفته است.

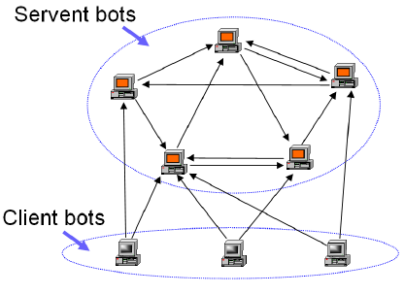
ابر نظیرها[3]: در این شبکه‌ها، تمامی نظیرها یکسان نیستند و بخش کوچکی از زیرمجموعه نظیرها به‌صورت خودکار به‌عنوان سرویس‌دهنده‌های موقت برای پشتیبانی از عملکرد شبکه مانند جستجو و کنترل انتخاب می‌شوند. شبکه‌های Skype، FastTrack و Gnutella از این نوع هستند. ازآنجاکه شبکه‌های ابر نظیر، قابل رویت‌تر و در برابر حملات هدفمند آسیب‌پذیرتر هستند، بات‌نت‌های کارآمد از چنین طراحی استفاده نمی‌کنند. بات‌نت‌هایی که به این دسته تعلق دارند معمولاً آدرس آی‌پی معتبری دارند و تحت تأثیر دیواره‌آتش یا DHCP نیستند. به‌طور معمول ساختن یک بات‌نت نظیر به نظیر شامل دو مرحله است، اول انتخاب کاندیدهای نظیر و دوم اعمال فعالیت‌های لازم برای آنکه کاندیدهای نظیر عضوی از شبکه بات شوند. برای انتخاب کاندیدها، سه زیرمجموعه متفاوت از بات‌نت‌های نظیر به نظیر در [۳۴] ذکر شده‌اند، و هر یک از این زیرمجموعه‌ها بر روی نوع خاصی از کاندیدها تمرکز دارد.یک مثال بات که از ساختار نظیر به نظیر برای ارتباط استفاده می‌کند Nugache است، که از یک فهرست با ۲۲ آدرس جایگزین استفاده می‌کند تا در طول تزریق دوم با آن تماس برقرار شود و فهرست نظیرهای موجود در شبکه را دریافت کند. بااین‌حال، تمام بات‌نت‌های نظیر به نظیر نیاز به کار با فهرست‌های سرویس‌دهنده‌های از قبل مشخص‌شده ندارند.

۴-۳- مدل فرمان و کنترل ترکیبی

ساختارهای ترکیبی هم از ویژگی‌های بات‌نت‌های متمرکز و هم از ویژگی‌های بات‌نت‌های غیرمتمرکز بهره می‌برند.

1-Unstructured P2P Overlayes
2-Distributed Hashing Table (DHT)
3-Superpeer

در مرجع [۳۵]، ساختاری پیشنهاد شده است که در آن باتهایی که متعلق به یک باتنت ترکیبی نظیر به نظیر هستند به دو گروه مجزای باتهای خدمتکار و باتهای مشتری تقسیم می‌شوند. عملکرد باتهای خدمتکار همانند مشتری‌ها و سرویس‌دهنده‌ها است، آن‌ها با آدرس‌های آی‌پی ثابت و با قابلیت مسیریابی تنظیم می‌شوند. از سویی دیگر، باتهای مشتری اتصال‌های جدید را نمی‌پذیرند و با آدرس‌های آی‌پی که به‌صورت پویا اختصاص یافته‌اند یا غیر قابل‌ردیابی هستند، تنظیم و سامان‌دهی می‌شوند. شکل (۳) این معماری را نشان می‌دهد.



شکل (۳). سه شاخص اصلی نهان نگاری ساختار معماری فرمان و کنترل پیشنهادی در مرجع [۳۵]

۴-۴- مدل فرمان و کنترل تصادفی

کوک[1] و همکارانش [۳۱] مدل تصادفی را به‌عنوان مدلی برای باتنت‌های آینده‌ای که بخواهند برای مدت‌زمان طولانی به کار خود ادامه دهند طراحی و ارائه کردند. در این مدل، بات به‌صورت فعال به مدیر بات یا باتهای دیگر متصل نمی‌شود و در عوض منتظر تماس از سوی مدیر بات باقی می‌ماند. برای انجام حمله، مدیر بات، شبکه را برای یافتن زامبی جستجو می‌کند و اگر موفق به پیدا کردن یکی از آن‌ها شود، آنگاه دستورات را برای بات ارسال می‌کند. مزیت این مدل این است که به‌راحتی می‌توان آن را اجرایی کرد و نسبتاً نیز انعطاف‌پذیر است، زیرا خصوصیات ارتباطی عادی میان بات و مدیر بات وجود ندارد و این خود تشخیص‌دادن و مختل‌کردن عملکرد را دشوارتر می‌کند. به دلیل نیاز به بررسی و جستجو در این ساختار مسائل و مشکلاتی در خصوص مقیاس‌پذیری و هماهنگی برای حمله وجود دارد. در حال حاضر هیچ‌یک از باتهای موجود از این روش استفاده نکرده و این روش تنها یک مدل نظری محسوب می‌شود.

۵- پروتکل‌های باتنت‌ها

محدوده وسیعی از پروتکل‌های ارتباطی با محوریت شبکه‌های رایانه‌ای برای برقراری یک شبکه بات وجود دارد. برای درک بهتر این ارتباط و نحوه به‌کارگیری آن‌ها توسط شبکه بات، در ادامه موضوعات مرتبط با هر پروتکلی که به‌هرنحوی در فرآیند چرخه عمر یک باتنت دخالت دارد، توضیح داده می‌شود.

۵-۱-پروتکل IRC

باتهای مبتنی بر این پروتکل، به‌طور عمومی از روش‌های متمرکز در ارتباطات خود استفاده می‌کنند. IRC یک سیستم گفتگوی مبتنی بر متن است که برقراری ارتباط و گفتگوی افراد را از طریق کانال‌هایی سازمان‌دهی می‌کند. در گذشته که مفهوم باتها الزاماً برای رفتار مخرب و مضر نبوده، ایده استفاده از این پروتکل، ایجاد و استفاده از باتنت‌ها برای کنترل ارتباطات و تعاملات در اتاق‌های گفتگوی IRC بوده است.

ویژگی پروتکل IRC، ممکن بودن ارتباط چندگانه از طریق گروه‌هایی با نام "کانال‌های ارتباطی" یا ارتباط یک‌سویه خصوصی میان دو عضو است. این ویژگی به مدیر بات اجازه می‌دهد کنترل انعطاف‌پذیری روی باتنت خود داشته باشد. برای مثال، می‌تواند گروه خاصی از باتها را برای انجام حمله انتخاب کند. از دیگر ویژگی‌های مفیدی که در باتنت‌های IRC برای مهاجمان وجود دارد، فراوانی، مقیاس‌پذیری و تطبیق‌پذیری است که به باتها اجازه استفاده دوباره از کدها را داده و باعث ایجاد باتنت‌های جدید می‌شوند [۳۶].

به‌علاوه، باتنت‌های IRC امروزی به سمت مبهم کردن محتوای فرمان و کنترل در پیغام‌های IRC با استفاده از یک زبان خارجی می‌روند. یا یک روش مبهم و آشفته کردن مانند XOR ساده، جایگزینی، یا درهم کردن را به‌کار می‌گیرند. این باتنت‌ها با استفاده از پیام‌های آشفته و مبهم IRC ، ممکن است از تشخیص مبتنی بر امضاء و روش‌های تشخیص مبتنی بر هانی‌پات فرار کنند. درواقع، بخش بزرگی از باتنت‌های IRC که امروزه مورداستفاده قرار می‌گیرند، از ارتباطات فرمان و کنترل مبهم استفاده می‌کنند.

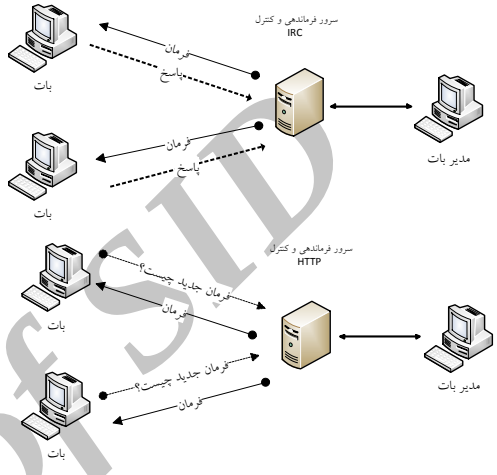
اگرچه پروتکل IRC برای استفاده در کانال فرمان و کنترل بسیار انعطاف‌پذیر و مناسب است، بااین‌حال دارای محدودیت‌های بسیار جدی است، زیرا به‌طورمعمول تشخیص و اخلال در عملکرد باتنت IRC آسان و راحت است. ازآنجاکه ترافیک IRC رایج و معمول نیست و به‌ندرت در شبکه‌ها استفاده می‌شود، تشخیص آن را نیز آسان‌تر می‌کند. درواقع، این نوع ترافیک معمولاً مسدود می‌شود. بنابراین، مدیر یک شبکه ممکن است بتواند به‌راحتی با تشخیص ترافیک IRC در شبکه و مسدود کردن آن با ابزارهایی مثل دیواره آتش، از فعالیت باتنت IRC جلوگیری کند [۳۷].

۵-۲-پروتکل HTTP

پروتکل HTTP، یک پروتکل مرسوم در باتنت‌ها بوده که روش ارتباطی آن بر پایه ارسال و دریافت متدهای GET و POST

1 -Cook

است. با توجه به رواج این پروتکل در شبکه‌ها و استفاده گسترده از آن، ردیابی باتنت‌های مبتنی بر این پروتکل کار آسانی نیست. استفاده از این پروتکل به یک بات این اجازه را می‌دهد تا سامانه‌های دفاعی شبکه مثل دیواره آتش را دور بزند که این کار در باتنت‌های IRC ممکن نبود. باتنت‌های مبتنی بر این پروتکل معایبی دارند که مهم‌ترین آن‌ها معماری متمرکز آنهاست که یک نقطه شکست محسوب می‌شود [۲]. شکل (۴) استفاده از این معماری را برای برقراری یک شبکه بات نشان می‌دهد.



شکل (۴). دو معماری از سرویس‌دهنده‌های باتنت

۵-۳- پروتکل (سرویس) DNS

این روش اگرچه به‌عنوان یک پروتکل مستقل نام‌گذاری نمی‌شود اما یک روش ارتباطی است. بات‌ها برای دریافت دستورات، با سرویس‌دهنده فرمان و کنترل خود، اتصالی را برقرار می‌کنند که از طریق پرس و جوی‌های DNS است. بات‌ها با این کار سرویس‌دهنده خاصی را پیدا می‌کنند که فراهم آورنده DNS معمولاً از آن میزبانی می‌کند. همین امر اساس ردیابی و تشخیص باتنت‌ها را با روش‌های داده‌کاوی و تحلیل ترافیک DNS تشکیل می‌دهد. تحلیل کردن پرس‌وجوهای DNS می‌تواند اطلاعات مناسبی در مورد وجود باتنت و مکان‌های سرویس‌دهنده فرمان و کنترل در اختیار سامانه‌های تشخیص قرار دهد. هنگامی‌که بات‌ها در دامنه‌های یکسان به جستجو می‌پردازند، رابطه‌ای در خلال این بازدیدهای فراوان بات‌ها در آن دامنه مشاهده می‌شود و اطلاعاتی مثل طول عمر دامنه، فراوانی جستجو و فراوانی دامنه به‌دست می‌آید. درواقع با کمک این پروتکل می‌توان روش‌های پاسخ مختلف را بررسی کرد و با استفاده از میزان درخواست‌های قانونی DNS و قیاس‌های تراکم DNS، باتنت‌هایی که ترافیک DNS را تشکیل می‌دهند را موردبررسی قرار داد.

۵-۴- پروتکل SMTP

پروتکل انتقال ساده نامه، یک استاندارد برای نامه‌های الکترونیکی از طریق پروتکل اینترنت است و معمولاً از درگاه ۲۵ روی TCP استفاده می‌کند. اجزای این پروتکل شامل سرویس‌گیرنده (عامل ارسال‌کننده) و سرویس‌دهنده (عامل گوش‌کننده) است. این درگاه توسط SSL با سرنام SMTPS امن شده است.

جدول (۲). مقایسه باتنت‌های مهم در سال‌های اخیر

ردیف	نام پروتکل	مزایا	محدودیت‌ها	کارایی (میزان تشخیص)
۱	IRC	- انعطاف‌پذیری انتخاب بات توسط مدیر بات - فراوانی، مقیاس‌پذیری و تطبیق‌پذیری استفاده از بات‌های جدید - مبهم‌سازی در ارتباطات فرمان و کنترل	تشخیص آسان به دلیل رایج نبودن ترافیک آن امکان مسدودسازی آسان توسط ابزارهای دفاعی شبکه	پایین
۲	HTTP	دشواری ردیابی به دلیل فروانی استفاده از این پروتکل	وجود معماری متمرکز و نقطه شکست	متوسط
۳	SMTP	- امکان توسعه برای بات‌های مختلف بر مبنای امضاهای متفاوت - توانایی استفاده برای حجم بالایی از نامه‌های الکترونیکی و هرزنامه‌ها	حجم کم اطلاعات ارسالی در هر بات امکان مسدودسازی آسان توسط ابزارهای دفاعی شبکه	متوسط
۴	DNS	استفاده از الگوریتم‌های مختلف تولید دامنه‌ها برای فرار از تشخیص	امکان بررسی روش‌های پاسخ مختلف و قیاس تراکم DNS	مشکل

اکثر بات‌ها این پروتکل را با اختلاف اندکی پیاده‌سازی می‌کنند، به‌همین دلیل این امکان وجود دارد که بتوان امضاهای مبتنی بر شبکه را به‌طور مؤثری برای تمایز بین بات‌نت‌های مختلف توسعه داد. بات‌نت‌ها از این پروتکل برای ارسال حجم زیادی از نامه‌های الکترونیکی و هرزنامه‌ها استفاده می‌کنند.

در جدول (۲) پروتکل‌های ارتباطی بات‌نت‌ها از نظر مزایا و کارایی مقایسه شده‌اند.

۶- انواع حمله‌ها

در این بخش انواع حملاتی را که یک شبکه بات می‌تواند در آن شرکت کند، برشماری می‌شوند. طبق یافته‌های محققین و آمار بیان‌شده از سوی مراکز آن‌ها، اگر مهاجم ی بتواند کنترل بیش از یک‌میلیون ماشین را به دست گیرد، در آن صورت در برابر حملات چنین بات‌نتی نمی‌تواند دفاع مؤثری انجام داد. چنین حمله‌ای می‌تواند خسارات سنگینی بر جای بگذارد که از آن جمله می‌توان به کاربرد آن به‌عنوان سلاحی خطرناک در جنگ با کشورها اشاره کرد. جای تعجب نیست که در محیط سایبری از بات‌نت‌ها به‌عنوان یکی از بزرگ‌ترین تهدیدها نام‌برده می‌شود [۹، ۱۳ و ۲۵]، تهدیدی که برای مقابله با آن به تلاش و همکاری گسترده نیاز است.

این فعالیت‌های مخرب و بدخواهانه می‌تواند شامل طیف گسترده‌ای مانند حملات زیر باشد، سرقت اطلاعات، انجام حملات ازکاراندازی توزیع‌شده، انتشار بدافزارها مثل ثبت‌کننده کلیدها[1] تروجان‌ها، ابزارهای جاسوسی[2]، اخاذی، سرقت منابع رایانه‌ای، بررسی و کنترل ترافیک شبکه، جستجوی رایانه‌های آسیب‌پذیر، ارسال هرزنامه، سرقت اطلاعات، سرقت هویت، دست‌کاری در بازی‌ها و نظرسنجی‌های برخط و غیره. جدول (۳)، انواع حملات و فراوانی آن‌ها را برحسب تاریخ فهرست کرده است.

۷- روش‌های تشخیص بات‌نت‌ها

برای تشخیص تهدید مهمی به نام بات‌نت، محققین روش‌های زیادی ارایه داده‌اند که اکثر آن‌ها جنبه ساختاری داشته و روش‌های مختلف را طبقه‌بندی کرده‌اند [۹، ۱۲ و ۳۸]. در ادامه به برخی روش‌های مهم که عموماً مبتنی بر عملکرد بات‌نت‌ها است، پرداخته می‌شود. در شکل (۵)، روش‌های اصلی تشخیص، طبقه‌بندی‌شده که در ادامه به‌طور مختصر توضیح داده می‌شوند.

در مرجع [۴۶]، روش‌های تشخیص به دو دسته اصلی مبتنی بر ایجاد هانی‌نت و مبتنی بر سامانه‌های تشخیص نفوذ تقسیم شده که خود نیز به دو زیرمجموعه مبتنی بر امضاء و مبتنی بر ناهنجاری تقسیم می‌شوند.

برای جمع‌آوری اطلاعات از بات‌ها، استفاده از هانی‌نت‌ها بهترین انتخاب هستند. پس از جمع‌آوری اطلاعات، می‌توان به فناوری استفاده‌شده پی برد و آن را یاد گرفت و تحلیلی از ویژگی‌های اصلی بات‌نت را در اختیار داشت. اغلب می‌توان سیستم تشخیصی با امضای بات تهیه کرد که بتواند سرویس‌دهنده‌های فرمان و کنترل، نقاط آسیب‌پذیری که پیش از این کسی از آن‌ها آگاه نبوده، ابزار و تکنیک‌های مورداستفاده مهاجمان و نیت مهاجم را متوجه شد. همچنین می‌توان از هانی‌نت‌ها برای به‌دست آوردن کدهای اجرایی و نفوذ در این بات‌نت‌ها استفاده کرد. در برخی تکنیک‌ها نیز از هانی‌نت‌ها برای به‌دام انداختن بات‌ها استفاده می‌شود [۳۹ و ۳۱].

هانی‌نت‌ها برای فهمیدن ویژگی‌ها و فناوری بات‌نت ضروری هستند، اما باید به این نکته توجه داشت که هانی‌نت‌ها چندین محدودیت به شرح زیر دارند، مقیاس محدود فعالیت‌های بدخواهانه‌ای که می‌توانند ردیابی کنند، نمی‌توانند بات‌هایی را که از روش‌های تکثیر استفاده نمی‌کنند، به‌دام اندازند مگر بات‌هایی که بر اساس جستجو و پویش، هرزنامه و دریافت‌های تحت وب هستند و اینکه تنها قادرند اطلاعات ماشین‌های آلوده‌ای که به‌عنوان تله استفاده شده‌اند را گزارش کنند.

روش‌های مبتنی بر امضاء در سیستم تشخیص بات‌نت [۴۰ و ۴۱] از امضاهای بات‌های موجود مانند SNORT استفاده می‌کنند. ایده اصلی آن‌ها، کاوش و به دست آوردن اطلاعات درباره خصوصیاتی مانند الگوها از بسته‌های ترافیکی کنترل‌شده و ثبت آن‌ها در پایگاه اطلاعاتی بات‌ها است. این روش‌ها تنها می‌توانند بات‌نت‌های شناخته‌شده و مشهور را تشخیص دهند. بنابراین، استفاده از این راهکار در مورد بات‌های ناشناخته و حملات بات‌هایی که به‌تازگی شروع به کار کرده‌اند ممکن نیست.

1 -Key Loggers
2 -Spywares

جدول (۳). انواع حملات توسط بات‌نت‌های مهم و فراوانی آن‌ها در سال‌های اخیر

ردیف	نام بات‌نت	پروتکل	سال تولید	تعداد بات‌ها	ظرفیت تولید هرزنامه (بیلیون در روز)	نوع حمله
۱	Bagle	متمرکز	۲۰۰۴	۲۳۰/۰۰۰	۵/۷	هرزنامه پست الکترونیکی
۲	Rustock	متمرکز/ HTTP	۲۰۰۶	۱۵۰/۰۰۰	۳۰	هرزنامه پست الکترونیکی و ازکاراندازی سرویس توزیع‌شده
۳	Srizbi	متمرکز/ HTTP	۲۰۰۷	۴۵۰/۰۰۰	۶۰	ارسال کدهای مخرب و هرزنامه
۴	Cutwail	غیرمتمرکز/P2P		۱/۵۰۰/۰۰۰	۷۴	ازکاراندازی سرویس توزیع‌شده
۵	Waledac	غیرمتمرکز/P2P	۲۰۰۸	۸۰/۰۰۰	۱/۵	هرزنامه پست الکترونیکی
۶	Conficker	غیرمتمرکز/P2P		۱۰/۵۰۰/۰۰۰	۱۰	حملات دزدی رمز عبور
۷	BredoLab	متمرکز/IRC و HTTP[1]	۲۰۰۹	۳۰/۰۰۰/۰۰۰	۳/۶	حملات تروجانی
۸	Grum	متمرکز/ HTTP		۵۶۰/۰۰۰	۳۹/۹	هرزنامه پست الکترونیکی
۹	LowSec	غیرمتمرکز	۲۰۱۰	۱۱۰۰۰	۰/۵	هرزنامه پست الکترونیکی
۱۰	Zeroaccess	غیرمتمرکز/P2P	۲۰۱۱	۹/۰۰۰/۰۰۰	۰/۰۲۹	روت‌کیت، کلیک حقه
۱۱	SpyEye	غیرمتمرکز	۲۰۱۲	۵۰۰	۰/۰۱	دزدی اطلاعات بانکی مشتریان
۱۲	shylock	غیرمتمرکز/P2P	۲۰۱۳	۱۰۰/۰۰۰	--	دزدی اطلاعات بانکی مشتریان
۱۳	Necurs	ترکیب متمرکز و P2P	۲۰۱۴	۱/۷۰۰/۰۰۰	۱۵	دزدی اطلاعات، هرزنامه پست الکترونیکی، باج افزار
۱۴	Cridex (Bugat V5)	غیرمتمرکز/P2P	۲۰۱۵	۴۰۰۰	۱۶۰۰۰	دزدی اطلاعات بانکی مشتریان، ازکاراندازی سرویس توزیع‌شده، ارسال کدهای مخرب و هرزنامه

شکل (۵). روش‌های تشخیص بات‌نت‌ها [۱۷]

۱- پروتکل ارتباطی این بات، IRC است. بنابراین برای عبور از دیواره‌های آتش از پروتکل HTTP استفاده می‌کند.

این روش همچنین نمی‌تواند بات‌های مشابهی که دارای امضایی با تغییرات اندک هستند را تشخیص دهد. نقطه‌ضعف دیگر در روش تشخیص مبتنی بر امضاء این است که همواره باید سعی شود پایگاه اطلاعاتی با امضاهای جدید به‌روزآوری شود که این امر خود افزایش هزینه‌های مدیریتی و کاهش عملکرد کلی را در پی دارد. توجه به این نکته مهم است که بات‌های جدید ممکن است پیش از آنکه پایگاه اطلاعاتی به‌روزآوری شود، شروع به حمله کنند [۴۲].

روش مبتنی بر ناهنجاری را می‌توان مهم‌ترین حوزه تحقیق در مبحث روش‌های تشخیص بات‌نت دانست. در این روش، ایده اصلی این است که عملیات تشخیص بات‌نت با در نظر گرفتن چند ناهنجاری متفاوت در ترافیک شبکه انجام شود، که این ناهنجاری‌ها عبارت‌اند از تأخیر زیاد شبکه، میزان بالای ترافیک، وجود ترافیک در درگاه‌های غیرعادی و رفتار غیرطبیعی سیستم. روش‌های مبتنی بر ناهنجاری خود به دو دسته مبتنی بر میزبان و مبتنی بر شبکه تقسیم می‌شوند.[۴۳].

در روش مبتنی بر میزبان، ماشین انفرادی (میزبان) موردبررسی قرار می‌گیرد تا هرگونه رفتار مشکوک شامل پردازش بیش‌ازحد در آن و دسترسی به فایل‌های مشکوک مشخص شود. این روش، معمولاً قابل اندازه‌گیری نیست زیرا تمامی ماشین‌های شبکه باید مجهز به ابزار نظارت باشند تا عملکرد مؤثری داشته باشند. از سوی دیگر، روش‌های مبتنی بر شبکه، ترافیک شبکه را به‌صورت فعال و غیرفعال تحلیل می‌کنند. در نظارت فعال برای اندازه‌گیری پاسخ شبکه، بسته‌ها به درون شبکه تزریق می‌شوند. مطالعه [۴۴] به ارائه روش فعالی به نام BotProbe می‌پردازد. ایده این روش این است که به شکلی پویا بسته‌هایی تزریق و وارد شود تا مشتری درونی را وارسی کنند و دریابند در آن‌سوی ارتباط یک انسان قرار دارد یا یک بات. نقطه‌ضعف بزرگ روش‌های فعال، بالا بردن حجم ترافیک شبکه است که با ارسال بسته‌های اضافی به ماشین‌های مشکوک رخ می‌دهد. علاوه بر این، و از همه مهم‌تر، تزریق و واردکردن بسته‌ها به ابزارهایی که تشخیص را ردیابی می‌کنند کمک می‌کند و ممکن است تحت پیگرد قانونی قرار بگیرند.

روش‌های نظارت غیرفعال به کنترل ترافیک داده در شبکه می‌پردازند و به دنبال ارتباطات مشکوک می‌گردند که ممکن است توسط بات‌ها یا سرویس‌دهنده‌های فرمان و کنترل فراهم شوند. با استفاده از امضاهایی که از قبل جمع‌آوری شده‌اند یا روش‌های مبتنی بر ناهنجاری، ترافیک داده تحلیل می‌شود.

یکی از فرضیه‌های مهم در نظارت غیرفعال این است که بات‌های موجود در یک بات‌نت تمایل به ارائه الگوهای ارتباطی یکسان دارند [۴۵] و این موضوع هم در ساختارهای متمرکز و هم در ساختارهای غیرمتمرکز صدق می‌کند (برای مثال در شبکه‌های نظیر به نظیر). علت این امر این است که بات‌ها از پیش به نحوی برنامه‌ریزی شده‌اند که ارتباطات معمول یکسانی با سرویس‌دهنده فرمان و کنترل و با همان مدیر بات داشته باشند. ازآنجاکه مدیران بات باید برای اجرای حمله با بات‌ها ارتباط برقرار کنند، الگوهای ترافیکی رایجی در شبکه متصل به هر مرحله از چرخه حیات بات وجود دارد. علاوه بر این، پروتکل‌های شبکه‌ای یکسانی در برقراری ارتباطات و انجام فعالیت‌های مخرب مورداستفاده قرار خواهند گرفت. و این نکته اساس روش‌های تشخیص را پایه‌گذاری کرده است که در تشخیص غیرفعال، تلفیقی از روش‌های مختلف استفاده می‌شود و شامل روش‌های آماری، کاوش ترافیک، تجسم، نظریه گراف، خوشه‌بندی[1]، همبستگی، مدل‌های تصادفی، آنتروپی، شبکه‌های عصبی، درخت تصمیم، انتقال گسسته فوریه، CUSUM، یادگیری ماشین، سری‌های زمانی گسسته، تحلیل گروهی و در برخی موارد ترکیبی از روش‌های موجود است [۱۷].

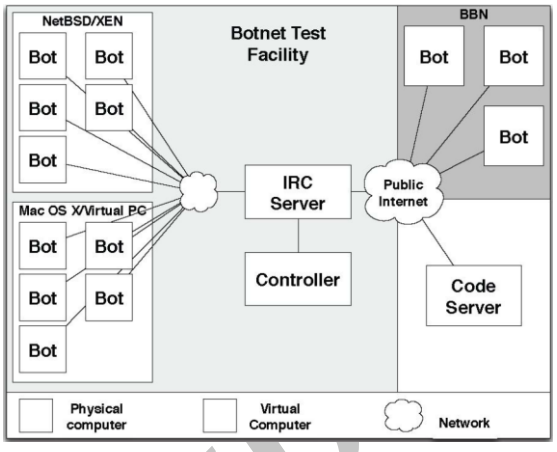
البته روش‌های تشخیص بنابر پروتکل‌های مختلف و عملکرد آن‌ها متفاوت خواهند بود که در میان آن‌ها، روش‌های مبتنی بر پروتکل DNS و شبکه‌های نظیر به نظیر دارای فراوانی تحقیق بیشتری در مقالات می‌باشند که پرداختن به آن‌ها از حوصله این نوشتار خارج است. برای اطلاعات بیشتر به مراجع [۵۰ - ۴۷] رجوع شود.

در ادامه بر اساس چرخه عمر یک بات‌نت نشان داده شده در شکل (۲)، روش‌های مختلف تشخیص موردبررسی و مقایسه قرار می‌گیرند. این چرخه عمر به‌طور مختصر به سه قسمت شکل‌گیری، فرمان و کنترل، و حمله تقسیم می‌شوند و پنج مرحله بیان‌شده در شکل مذکور را در خود جا دارند، هرکدام از این مراحل روش‌های تشخیصی متفاوتی دارند که در ادامه توضیح داده می‌شوند.

در مرحله شکل‌گیری لیویداس[2] و همکاران [۵۱] روشی برای تشخیص بات‌نت‌های مبتنی بر IRC ارائه کرده‌اند. این روش دومرحله‌ای یک دسته‌بند از شبکه بیزی برای تمایز بین ترافیک IRC و غیر IRC استفاده می‌شود، سپس با استفاده از این دسته‌بند ترافیک بات‌نت از ترافیک IRC تشخیص داده می‌شود. محدودیت‌های این روش وابسته بودن به پروتکل خاصی از کانال فرمان و کنترل، و استفاده از داده‌های برچسب‌گذاری شده برای یادگیری است. معماری ابزار این روش در شکل (۶) نمایش داده

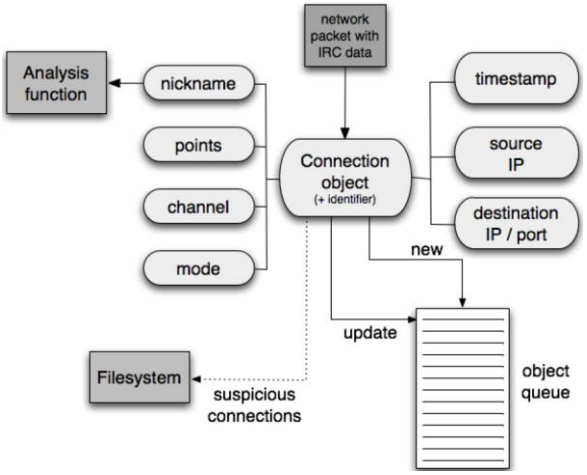
1-Clustering
2 -Livadas

شده است.



شکل (۶). شبکه ارزیاب مورداستفاده در [۵۱] برای تولید ترافیک باتنت

جیوبل[1] و همکاران [۵۲] روش تشخیصی به‌نام ریشی[2] ارائه کرده‌اند که بر باتنت‌های متمرکز مبتنی بر IRC تمرکز دارد. این روش از یک تحلیل *n*-گرم[3] و یک تابع امتیازدهی بهره می‌گیرد. در این روش بسته‌های TCP حاوی کلمات کلیدی رایج IRC استخراج‌شده و با تعیین نام‌های مستعار مرتبط با آن‌ها یک تابع امتیازدهی ایجاد می‌شود. اگر امتیاز این نام‌های مستعار یک حد آستانه بیشتر باشد، هشدار تشخیص باتنت داده می‌شود. نرخ هشدار نادرست این روش که صرفاً برای باتنت‌های متمرکز مبتنی بر IRC است، نسبتاً بالا است. اگر بات‌ها از نام‌های مستعار متشکل از ترکیب کاراکترهای تصادفی استفاده کنند که برای آن‌ها عبارت باقاعده‌ای وجود نداشته باشد، این روش نرخ خطای بالای خواهد داشت. شکل (۷) این روش را نمایش می‌دهد.



شکل (۷). روش تشخیصی ریشی [۵۲]

ونگ[4] و همکاران [۵۳] یک روش تشخیصی مبتنی بر IRC در مرحله شکل‌گیری از چرخه حیات ارائه کرده‌اند که با تحلیل سطح گروهی ناشی از یک بخش ثابت رشته‌ای و یک بخش تصادفی دارای الگوی یکسان بر اساس نام‌های مستعار باتنت‌ها کار می‌کند. در این روش یک معیار فاصله‌ای درکانال جهت محاسبه شباهت نام‌های مستعار درون یک کانال تعریف شده و سپس از این معیار در شناسایی کانال‌های باتنتی استفاده می‌شود. همان‌طور که بیان شد این نام‌های مستعار دارای ساختار یکسانی هستند. نرخ هشدار نادرست بالا در این روش مانند روش ریشی وجود دارد. ولی این روش باتنت‌های IRC ناشناخته را می‌تواند تشخیص دهد. در مرحله فرمان و کنترل از چرخه حیات نیز روش‌های تشخیصی ارائه‌شده است که در ادامه به آن‌ها پرداخته می‌شود.

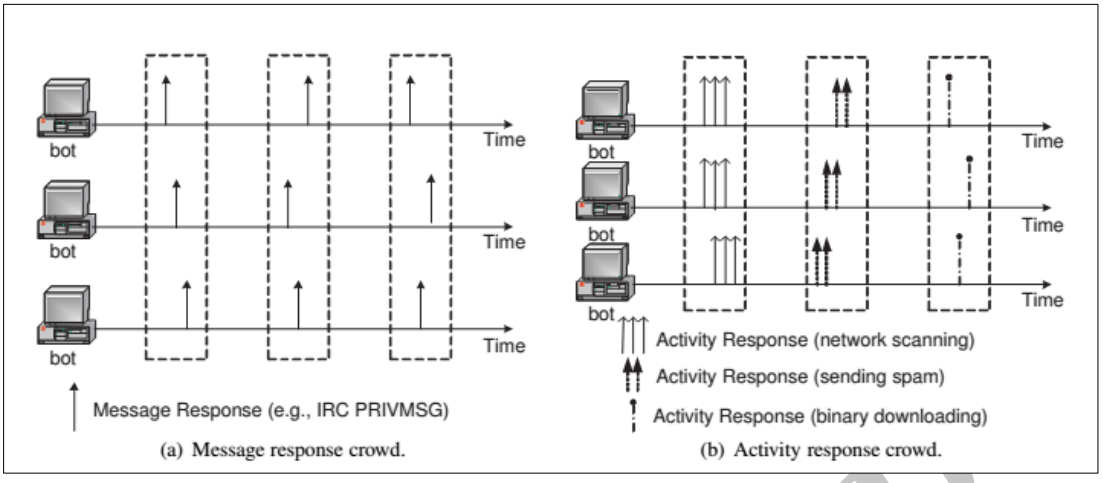
جو[5] و همکاران [۵۴] سیستمی به نام بات‌هانتر[6] ارائه کرده‌اند که به‌صورت غیرفعال و از طریق شناسایی آلودگی موفق باتنت برای تشخیص میزبان‌های آلوده به بات عمل می‌کند. بات‌هانتر با دنبال کردن جریان‌های ارتباطی بین میزبان‌های داخلی و خارجی و استخراج دنباله‌ای از شواهد داده‌های مبادله‌ای مطابق با یک مدل آلودگی مبتنی بر حالت اقدام به تشخیص می‌کند. بات‌هانتر با دنبال کردن جریان‌های ارتباطی بین میزبان‌های داخلی و خارجی و استخراج دنباله‌ای از شواهد داده‌های مبادله‌ای مطابق با یک مدل آلودگی مبتنی بر حالت اقدام به تشخیص می‌کند.

بات اسنیفر [۵۵] روشی است که از تشخیص مبتنی بر ناهنجاری برای شناسایی کانال‌های فرمان و کنترل استفاده می‌کند. اساس کار این روش این است که بات‌ها در یک باتنت یکسان، به جهت فعالیت‌های هماهنگ و یکسان در کد اجرایی خود، هم‌زمانی و همبستگی قابل‌توجهی در عملکرد خود از دید پاسخ یا فعالیت‌های خود دارند (شکل ۸)، درحالی‌که رفتار هماهنگ و همبسته در فعالیت‌های شبکه‌ای هنجار وجود ندارد. در این روش با استفاده از الگوریتم‌های اعمالی، با مشاهده نمونه رفتارهای مشابه و همبسته، باتنت‌ها تشخیص داده می‌شوند.

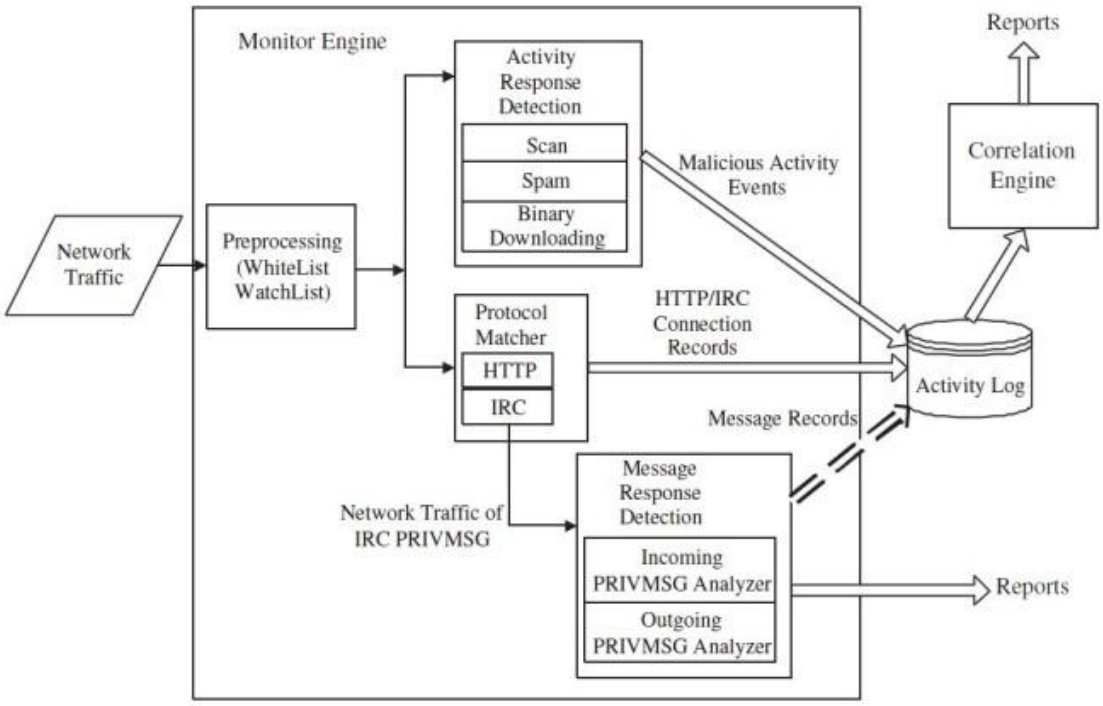
معماری بات‌اسنیفر در شکل (۹) نمایش داده شده است. این روش صرفاً باتنت‌های متمرکز را تشخیص می‌دهد.

1 -Goebel
2 -Rishi
3 -*n*-gram

4 -Wang
5 -Gu
6 -BotHunter



شکل (۸). شباهت و همبستگی مکانی- زمانی در پاسخ‌های بات‌ها (پاسخ‌های پیامی و فعالیتی) [۵۵]



شکل (۹). معماری بات اسنیفر [۵۵]

ورزینگر[1] و همکاران [۵۶] سیستم تشخیص بات‌نتی ارائه کرده‌اند که میزبان‌های آلوده به بات را بدون توجه به کانال‌های فرمان و کنترل و روش انتشار بات‌نت شناسایی می‌کند. سیستم آن‌ها یک مدل تشخیص دو حالته است؛ حالت اول مدل، مشخص‌کننده ارسال فرمانی ویژه برای بات‌ها است. حالت دوم پاسخ بات‌ها به فرمان‌های صادرشده است. در صورتی‌که میزبان‌ها به حالت دوم بروند، هشدار آلودگی به بات اعلام می‌شود. تحلیل محتوای بسته‌ها از نیازمندی‌های شناسایی این سیستم است و لذا نمی‌تواند بات‌نت‌هایی را تشخیص دهد که کانال‌های فرمان و کنترل آن‌ها رمز شده هستند و یا فرمان‌های آن‌ها مبهم‌سازی و پیچیده شده‌اند. در مرحله حمله روش‌های تشخیصی زیر موردبررسی قرار گرفته و ارائه می‌شوند.

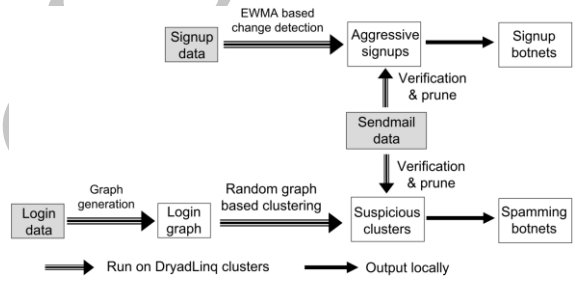
کستل[2] و همکاران [۵۷] روشی تشخیصی برای بات‌نت‌های تولید هرزنامه ارائه کرده‌اند. در این روش با استفاده از ترکیب اطلاعات سرآیند یک پیام پست الکترونیکی و اطلاعات پاکت نامه، مجموعه‌ای از سرآیندهای ترکیبی تولید می‌شود که مقادیر این سرآیندهای ترکیبی با یک بازه محدودی از نشانه‌ها را برای ایجاد یک الگوی نمایشی هنجار شده جایگزین می‌کنند. درنهایت، این الگوها گروه‌بندی‌شده تا خوشه‌هایی با تعداد زیادی از پیام‌های پست الکترونیکی مربوط به آن‌ها تولید شوند. این

1 -Wurzinger

2 -Castle

خوشه‌ها به‌عنوان خوشه‌های مشکوک به بات‌نت شناسایی می‌شوند. محدودیت این روش این است که صرفاً برای ارسال هرزنامه کاربرد دارد.

ژائو[1] و همکاران [۵۸] نیز روش تشخیصی به نام BotGraph برای شناسایی بات‌نت‌های ارسال‌کننده هرزنامه ارائه کرده‌اند که از طریق ساخت گراف بزرگ کاربر-کاربر و شناسایی همبستگی بین فعالیت‌های یک بات‌نت حساب‌های کاربری نامه‌های الکترونیکی را شناسایی می‌کند این حساب‌ها برای ارسال هرزنامه استفاده می‌شود. BotGraph دو مؤلفه اصلی دارد، مؤلفه اول تشخیص ثبت‌نام‌های مشکوک و مؤلفه دوم تشخیص دزدی کاربر بات است. مؤلفه اول BotGraph، حساب‌های کاربری تحت اختیار ارسال‌کننده هرزنامه را محدود می‌کند و مؤلفه دوم با استفاده از یک تحلیل همبستگی در ورود به‌حساب‌های کاربری آن‌ها را شناسایی می‌کند. شکل (۱۰)، معماری این روش را نشان می‌دهد.



شکل(۱۰). معماری روش BotGraph [۵۸]

موارد بیان‌شده روش‌های تشخیصی در مراحل مختلف چرخه حیات بات‌نت‌ها در جدول (۴) نشان داده شده‌اند.

جدول (۴). مقایسه روش‌های تشخیص بات‌نت مبتنی بر چرخه حیات

روش تشخیص بات‌نت	مرحله تشخیص	محدودیت
لیویداس [۵۱]	مرحله شکل‌گیری	وابسته بودن به پروتکل خاص
ریشی [۵۲]		صرفاً برای بات‌نت‌های متمرکز مبتنی بر IRC
ونگ [۵۳]		نرخ هشدار نادرست بالا
بات‌هانتر [۵۴]	مرحله فرمان و کنترل	نیاز به بهینه‌سازی خروجی
بات‌اسنیفر [۵۵]		وابستگی زیاد به مدل آلوده‌سازی
ورزینگر [۵۶]		عدم‌تشخیص کانال‌های فرمان و کنترل رمز شده یا مبهم شده
کستل [۵۷]	مرحله حمله	کاربرد صرفاً در تشخیص ارسال هرزنامه
بات‌گراف [۵۸]		کاربرد صرفاً در تشخیص ارسال هرزنامه

1- Zhao

با وجود این که روش‌های زیادی برای تشخیص بات‌نت وجود دارد، با این‌حال تشخیص امری نسبتاً مشکل است و شاید همواره با موفقیت همراه نباشد. ترافیک بات‌نت شبیه به ترافیک معمولی است و ممکن است برای فرار از تشخیص، به‌خصوص در روش‌های مبتنی بر امضا و تحلیل‌های محتوای بسته‌ها، از رمزگذاری و پنهان نگاری استفاده شود. در تشخیص معمولاً تحلیل حجم بزرگی از داده‌ها لازم و ضروری است که زمان واقعی اجرای آن مشکل و گاهی غیرممکن است و همین امر تشخیص در شبکه‌هایی با ابعاد بزرگ را کاری بس دشوار و مشکل می‌کند. در ادامه چند ابزار مفید تشخیص بات‌نت‌ها که در مطالعات قبلی معرفی‌شده‌اند به‌طور مختصر معرفی می‌شوند.

۷-۱- ابزارهای تشخیص بات‌نت‌ها

در این زیر بخش، بعضی از ابزارهای بکار گرفته‌شده جهت تشخیص بات‌نت‌ها مرور می‌شوند. لازم به ذکر است که ملاک انتخاب این روش‌ها، ارائه روش‌های تحقیقی منتشرشده در مقالات علمی معتبر است.

BotGrep -۱-۱-۷

ناگاراجا[2] و همکارانش [۵۹] ابزاری طراحی کردند که به‌وسیله آن می‌توان بر اساس تحلیل‌های گراف شبکه، بات‌نت‌های نظیر به نظیر را تشخیص داد، برای مثال اطلاعاتی در خصوص اینکه کدام جفت از گره‌ها با یکدیگر در ارتباط هستند که به آن گراف ارتباطات[3] می‌گویند. این روش مبتنی بر ماهیت آمیختن سریع[4] گراف ساخته‌شده فرمان و کنترل بات‌نت است. الگوریتم BotGrep مرتباً گراف ارتباطات را به قسمت‌هایی با سرعت آمیختن زیادتر و کمتر تقسیم می‌کند و نهایتاً به قسمت آمیختن سریع محدود می‌شود. در تحلیل گراف شبکه، فرض این است که میزبان‌های متعلق به بات‌نت نظیر به نظیر، بیشتر از دیگر میزبان‌ها متصل هستند.

BotHunter -۲-۱-۷

در [۵۴]، روشی ارائه‌شده است که در آن، سیستم تشخیص، بات نامحسوس و منفعلی است که از همبستگی گفتگوی‌های IDS، برای مرتبط‌کردن رخدادهای IDS با مدل‌های آلودگی بات استفاده می‌کند. ازآنجاکه هدف BotHunter تشخیص رفتار بات در سطح شبکه است، بات‌های پنهان‌کار و مخفی می‌توانند با اجتناب از همبستگی زمان‌بندی رخداد یا انجام حملات موضعی (مانند پاک کردن فایل‌ها)، بدون هرگونه فعالیت‌های شبکه‌ای از

2- Nagaraja
3- Communication graph
4- Fast-mixing

تشخیص فرار کنند. نویسندگان این مطالعه با توجه به فعالیت‌های زیر، چرخه حیات آلودگی باتنت را مدل‌سازی کرده‌اند، پیمایش هدف، بهره‌برداری از آلودگی، واکشی و اجرای کد اجرایی، راه‌اندازی کانال فرمان و کنترل، و اسکن رو به بیرون شبکه.

در [۶۰]، یک شبکه محلی ابداعی به نام Sub – Botnet برای ارزیابی از این ابزار در سراسر تحقیق خود استفاده شده است که می‌تواند الگوی خوبی برای محقق‌هایی باشد که نیاز به تجربه کار عملی با این ابزار دارند.

Botsniffer-۳-۱-۷

در [۵۵] ابزاری طراح شده است که از روش‌های تشخیص ناهنجاری مبتنی بر شبکه استفاده می‌کند این ابزار برای تشخیص باتنت‌های IRC و HTTP در یک شبکه حوزه محلی طراحی شده است. Botsniffer بات‌های درون یک باتنت را که احتمالاً در پاسخ‌ها و فعالیت‌های خود شباهت‌های زیادی را بروز می‌دهند موردبررسی قرار می‌دهد، یعنی پیمایش و ارسال ایمیل‌های هرزنامه و درنتیجه به اشتراک گذاشتن محتوای ارتباطی مشترک را و مشاهده می‌کند.Botsniffer از روش تشخیصی به نام همبستگی زمانی- مکانی استفاده می‌کند بدان‌معنا که تمامی باتنت‌ها، برخلاف انسان‌ها، تمایل به برقراری ارتباط به نحوی کاملاً هم‌زمان دارند.

Botminer -۴-۱-۷

تکنیکی است که برای تشخیص ترافیک فرمان و کنترل باتنت، از روش‌های داده‌کاوی استفاده می‌کند [۴۵]. Botminer درواقع روش Botsniffer را ارتقا می‌بخشد [۵۵]. Botminer ارتباطات مشابه و ترافیک مخرب را خوشه‌بندی می‌کند؛ سپس همبستگی میان دسته‌ای را برای شناسایی میزبان‌هایی که هم ارتباطات مشابه و هم الگوهای فعالیت مخرب دارند اجرا می‌کند.Botminer ابزار پیشرفته تشخیص باتنت است که مستقل از پروتکل و ساختار باتنت است. Botminer قادر به تشخیص باتنت‌های زمان-واقعی شامل باتنت‌های مبتنی بر IRC، HTTP و P2P، با درصد خطای پایین است. نقطه‌ضعف این روش نیز همان نقطه‌ضعف روش‌های مبتنی بر امضا است که در آن، ترافیک باتنت بدون الگوها و امضاها قابل‌تشخیص نیست.

BotMagnifier-۵-۱-۷

این ابزار در [۶۱] برای پشتیبانی از شناسایی و تعقیب بات‌هایی است که هرزنامه ارسال می‌کنند. این ابزار دسته‌ای از آدرس‌های آی‌پی ابتدایی که با بات‌های هرزنامه در ارتباط هستند را به‌عنوان ورودی دریافت می‌کند و رفتار ارسال هرزنامه آن‌ها را یاد می‌گیرد. این روش به‌خوبی می‌تواند رفتار هرزنامه را مدل‌سازی کند؛ نیاز به در اختیار داشتن فهرست دسته‌ای آی‌پی ابتدایی، یک نقطه‌ضعف بزرگ این ابزار است.

۸- روش‌های گریز باتنت‌ها

تشخیص بدافزار پنهان‌کار و مخفی مثل باتنت‌ها و جاسوس‌افزارها به دلیل آنکه فعالیت‌های آن‌ها دقیق و ظریف است و شبکه را برخلاف حملات ازکاراندازی و کرم‌های مهاجم مختل نمی‌کنند دشوار است. بات‌ها روز به روز پیشرفته‌تر می‌شوند، ازاین‌رو روش‌های گریختن نیز برای فریب دادن روش‌های تشخیص توسعه یافته‌اند و به باتنت‌ها این اجازه را می‌دهند تا زمان اجرایی طولانی‌تری داشته باشند [۶۲]. استینسون و میشل[۶۳] چارچوبی را برای ارزیابی توانایی گریختن و مناسب بودن یک روش تشخیص ارائه کرده‌اند.

چندین روش گریختن مختلف به‌طورمعمول استفاده می‌شوند که شامل، تونل زدن[1] از طریق HTTP، ICMP، یا پروتکل‌های VoIP، تونل‌زدنIpv6، شبکه‌های fast-flux، تغییرات در الگوهای آماری، استفاده از DNS پویا، ترافیک رمزگذاری شده، محول کردن وظایف مختلف به بات‌های موجود در یک شبکه، و تصادفی کردن الگوهای ارتباطی بات می‌شود[۴۵،۶۲]. توسعه و ساخت روش‌های جدید گریختن منجر به توسعه روش‌های جدید تشخیص می‌شود و این مسابقه بین مهاجمان و مدافعان شکل گرفته و همواره ادامه دارد.

در [۶۴] مدل باتنت جدیدی ارائه‌شده که برای ساختن جایگذاری[2] پارازیتی از یک شبکه جایگذاری مانند Skype استفاده می‌کند و با این کار ردیابی مدیر بات و از کار انداختن باتنت بی‌آنکه به کاربران مجاز Skype آسیبی وارد شود را بسیار دشوار می‌کند. این روش از قابلیت اطمینان و توانایی موجود در شبکه Skype برای عبور آسان از دیواره آتش وNAT بهره می‌برد. بات‌ها با استفاده ازSkype API ساخته می‌شوند و پیام‌های رد و بدل شده میان بات‌ها و مدیر در شبکه مانند پیام‌های مجاز کاربردی منتقل می‌شوند. این امر باعث می‌شود ترافیک باتنت میان ترافیک Skype غیرقابل‌شناسایی شود. اگرچه در این مطالعه از Skype استفاده شده است، اما می‌توان از دیگر روش‌های گریختن که از برنامه‌های کاربردی مشهور استفاده و بهره‌برداری می‌کنند نیز استفاده کرد.

اگرچه طرح‌های تشخیص باتنت مبتنی بر دسته‌بندی مؤثر هستند، می‌توان با تصادفی کردن الگوهای ارتباطی و محول کردن نقش‌های مختلف به هر بات از آن‌ها اجتناب کرد. همان‌طور که Nugache نشان می‌دهد، بات مورد مطالعه از

1-Tunneling
2-Overlay

درگاه‌های تصادفی با شماره بالا استفاده می‌کند. در این بات، هر نظیر، درگاه شماره بالای تصادفی تولید شده خود را انتخاب می‌کند تا روی آن گوش کند که شماره درگاه از ۱۰۲۵ تا ۶۵،۵۳۵ متغیر است و می‌تواند در بیشتر مواقع، از تشخیص فرار کند. به همین ترتیب، Storm برای برقراری ارتباط یک درگاه تصادفی با شماره بالا را انتخاب می‌کند و آن درگاه را در تمام بسته‌های ارسالی منتشر می‌کند.

اغلب بات‌نت‌های جدید از شبکه‌های fast-flux [۱۱،۳۷،۶۶،۶۵] به‌عنوان روش فرمان و کنترل خود استفاده می‌کنند. fast-flux یک روش DNS است که برای پنهان کردن حملات فیشینگ و سایت‌های توزیع بدافزار استفاده می‌شود. این شبکه‌ها پشت شبکه‌ای از میزبان‌های به‌خطرافتاده استفاده می‌شوند که همواره در حال تغییر هستند و به عنوان پروکسی عمل می‌کنند. fast-flux همچنین می‌تواند اشاره به ترکیب شبکه‌سازی نظیر به نظیر، فرمان و کنترل توزیع‌شده، متوازن کردن بار مبتنی بر وب، و تعیینِ جهت مجدد پروکسی مورداستفاده قرار گیرد و برای هرچه مقاوم‌تر کردن شبکه‌های بدافزار در مقابل کشف و اقدامات متقابل،کاربرد داشته باشد. در این روش، برای ناشناس ماندن، بات‌ها ابتدا به یک میزبان موافق، که به‌عنوان یک پروکسی عمل می‌کند، متصل می‌شوند تا با این کار دستورات بات به سرویس‌دهنده واقعی فرمان و کنترل ارسال شود و پاسخ‌ها از سرویس‌دهنده فرمان و کنترل به بات‌ها منتقل شوند. در این روش، سوابق DNS یک وب‌سایت واقعی، رایانه‌ها را در یک شبکه fast-flux نشان می‌دهد. شبکه‌های fast-flux برای منتشر کردن دستورات بات در تعداد زیادی میزبان‌های موافق، آدرس‌های IP را در یک روش چرخشی عادلانه[1] با مقادیری که TTL کوتاهی برای هر سابقه منبع DNS خاص و مشخص دارند ترکیب می‌کنند. برای مدیریت هم‌زمان میزان دسترسی برای هزاران دامنه روی یک میزبان، گره‌های موجود در شبکه‌های fast-flux همواره از سرویس‌های DNS و HTTP میزبانی می‌کنند. Storm [۶۷] نوعی بدافزار جدید است که از این روش استفاده می‌کند. در همین راستا و بر پایه روش Fast-flux، در بات‌نت‌های نسل جدید مثل Conficker، Kraken، Cycbot و Murofet از الگوریتم‌های تولید نام دامنه[2] [۶۸] برای یافتن آدرس‌های سرویس‌دهندگان فرمان و کنترل استفاده می‌شود. در این بات‌نت‌ها، هر میزبان آلوده (بات)، با استفاده از یک الگوریتم از پیش تعریف‌شده تعداد زیادی نام دامنه تولید کرده و آن‌ها را با استفاده از پرس‌وجوهای DNS ارسال می‌کند تا آدرس‌های سرویس‌دهندگان فرمان و کنترل را به‌دست آورد (به عنوان مثال، در نسخه جدید بات‌نت Conficker هر میزبان آلوده روزانه ۵۰۰۰۰ نام دامنه تولید می‌کند). اما از آنجاکه تعداد کمی از این نام‌های دامنه توسط مهاجم ثبت می‌شوند، بسیاری از این پرس‌وجوها با شکست مواجه می‌شوند. در بات‌نت‌های مختلف از ورودی‌های متفاوتی برای الگوریتم‌های تولید نام دامنه استفاده می‌شود. به عنوان مثال، Conficker با ارسال یک پرس‌وجو به سایت‌های معتبر (از قبیل Google) زمان و تاریخ فعلی را دریافت کرده و به‌عنوان ورودی الگوریتم تولید نام دامنه استفاده می‌کند. یا Kraken به‌صورت تصادفی کلماتی از ترکیب حروف الفبای زبان انگلیسی ایجاد کرده و به انتهای این کلمات، پسوندهای رایجی را از قبیل ly- و able- اضافه می‌کند [۶۹].

مهاجم با تولید تعداد زیادی نام دامنه و انتخاب طول عمر کوتاه برای آن‌ها آدرس سرویس‌دهندگان فرمان و کنترل را به‌صورت پویا تغییر می‌دهد. بنابراین، سیستم‌های تشخیص بات‌نت باید نام‌های دامنه تولید شده را به‌صورت الگوریتمی شناسایی کرده و آن‌ها را به فهرست‌های سیاه اضافه کنند. اما برای شناسایی دقیق این نام‌های دامنه، ابتدا باید ورودی الگوریتم تولید نام دامنه با مهندسی معکوس مشخص شود که به هزینه و زمان زیادی نیاز دارد. این عامل باعث شده که استفاده از الگوریتم‌های تولید نام دامنه از محبوبیت زیادی در بات‌نت‌های نسل جدید برخوردار باشد [۷۰]. Yadav و همکاران [۷۰] روشی مبتنی بر شناسایی نام‌های دامنه الگوریتمی برای تشخیص بات‌نت‌های نسل جدید ارائه کرده‌اند که از توزیع کاراکترهای حرفی- عددی در پرس‌وجوهای DNS استفاده می‌کند. مرجع [۷۱] این روش را بررسی کرده و وابستگی به تعداد زیاد پرس‌وجوهای DNS به دلیل عدم توجه به سابقه فعالیت‌های گروهی مشکوک در میزبان‌های شبکه را علت بالارفتن نرخ هشدار نادرست در این روش ذکر کرده است. آن‌ها در [۷۲] روش دیگری ارائه کرده‌اند که از شکست‌ها در پرس‌وجوهای DNS برای سرعت بخشیدن به فرآیند تشخیص استفاده می‌کند. Choi و همکاران [۷۳] روشی به نام BotGod ارائه کرده‌اند که از فعالیت‌های گروهی در پنجره‌های زمانی یکسان از ترافیک DNS برای تشخیص بات‌نت‌ها استفاده می‌کند. این روش در [۷۱] تحلیل شده و عنوان شده که نرخ تشخیص آن تا حد زیادی وابسته به انتخاب اندازه مناسب برای پنجره‌های زمانی است و به دلیل عدم توجه به شکست‌ها در ترافیک DNS از کارایی مناسبی برای تشخیص بات‌نت‌های نسل جدید برخوردار نیست. Huang [۷۴] روشی برای تشخیص میزبان‌های آلوده به بات‌های شناخته‌شده در یک شبکه محلی پیشنهاد کرده است. در این روش، ابتدا شکست‌ها در جریان‌های TCP و UDP برای هر میزبان آلوده و غیر آلوده جمع‌آوری می‌شوند. سپس با توجه به

1-Round - Robin
2-Domain Name Generation Algorithms (DGAs)

این شکست‌ها تعدادی بردار ویژگی استخراج می‌شود. این بردارهای ویژگی به‌عنوان ورودی به الگوریتم دسته‌بندی C4.5 داده‌شده و از مدل ایجادشده برای تشخیص هر میزبان آلوده استفاده می‌شود. این روش متکی به شکست‌ها در ترافیک هر میزبان است و با توجه به اینکه به ویژگی‌های ذاتی بات‌نت‌ها مثل فعالیت‌های گروهی را درنظر نمی‌گیرد نرخ هشدار نادرست بالایی دارد.

طبق مطالعه پلوهمن[1] و همکارانش [۱۳]، رویه دیگر، بهبود در تمهیدات امنیتی مهاجمان است. اولین بات‌نت‌ها مرتباً از زیرساخت و کد خرابکارانه ساده بهره می‌برند که تمام سناریو را شامل می‌شوند، اما این موضوع در حال تغییر است. مهاجمان در هر قدمی که برمی‌دارند محتاط‌تر عمل می‌کنند. زمانی که متوجه موضوعی مشکوک و عجیب می‌شوند، از رمزگذاری کلید عمومی، VPN توزیع‌شده، fast-flux، رمزگذاری PHP، مبهم و تیره‌کردن جاوا اسکریپت، پک‌کننده‌های هسته[2]، کانال‌های مخفی و جدا کردن خودکار استفاده می‌کنند.

۹- موضوعات آینده پژوهی

در این بخش ابتدا به برخی چالش‌ها در مطالعات بات‌نت‌ها اشاره می‌شود و در ادامه موضوعاتی که در این زمینه برای آینده قابل تحقیق هستند مطرح می‌شود. در مطالعات بات‌نت‌ها چند چالش اساسی وجود دارد که در مطالعات با آن‌ها برخورد شده است. مهم‌ترین آن‌ها عدم وجود داده‌های واقعی است که کمترین اثر آن انعکاس غیرواقعی رفتار بات‌نت‌ها است. در بیشتر موارد شبکه‌های دانشگاهی واقعیت شبکه‌های ناهمگن را منعکس نمی‌کنند و قادرند مواردی زیادی که برای شبکه در یک محیط کنترل‌شده و نسبتاً امن، ناشناخته‌اند را پنهان کنند. این درحالی است که اغلب ملاک‌های تشخیص، شبکه‌های آزمایشگاهی هستند. به عبارت دقیق‌تر، تمام بات‌نت‌ها از یک سری وضعیت‌های استاندارد پیروی نمی‌کنند، بنابراین ممکن است رفتار خود را در مثلاً در یک ماشین مجازی، بسیار متفاوت‌تر از یک شبکه واقعی نشان دهند. یکی دیگر از چالش‌ها این است که معمولاً در روش‌های تشخیص ارائه‌شده، مدل‌ها را برای داده‌های برخط در نظر نمی‌گیرند البته در [۱۶] از داده‌های جریانی برای تشخیص بات‌نت‌ها در شبکه استفاده شده است که در نوع خود ارزشمند است اما به‌طور عام نتایج داده‌ها در حالت غیرواقعی است. این سؤال اساسی باقی می‌ماند که یک روش تشخیص جدید تا چه میزان به پیشرفت در تشخیص بات‌نت‌ها کمک می‌کند. چالش بعدی عدم وجود یک معیار ارزیابی قابل‌قبول برای ارزیابی روش‌های تشخیص بات است و همین نبود یک معیار استاندارد مقایسه کمی کارهای موجود را دشوار کرده است.

موضوعات گوناگونی برای تحقیق در خصوص شبکه‌های بات وجود دارد که به برخی از موارد مهم اشاره می‌شود.

۹-۱- بات‌های شبکه‌های اجتماعی

استفاده از شبکه‌های اجتماعی می‌تواند به‌عنوان یک منبع عظیم زیرساخت‌های فرمان و کنترل برای مدیران بات مورداستفاده قرار بگیرد و دلیل آن دشوار بودن تمایز فعالیت‌های فرمان و کنترل از ترافیک فعالیت‌های شبکه‌های اجتماعی است [۷۵]. بات‌های اجتماعی[3] برنامه‌هایی هستند که حساب‌های شبکه‌های اجتماعی را کنترل می‌کنند و به تقلید از کاربران واقعی می‌پردازند تا شبکه‌های بات‌اجتماعی را بسازند. این بات‌ها زیرمجموعه‌ای از بات‌های اجتماعی تطبیق‌پذیر بوده و به سبک فرمان و کنترل هماهنگ شده‌اند. شبکه‌های بات‌اجتماعی Facebook، NazBot، و Koobface اولین نمونه‌ها از شبکه بات‌اجتماعی به شمار می‌آیند.

Koobface [۷۶] اولین بات‌نت در حیطه شبکه‌های اجتماعی برخط[4] است. Koobface در اولین قدم، حساب‌های کاربری را روی شبکه پیدا می‌کند و از این حساب‌ها برای ایجاد پیام محرک با یک فراپیوند[5] مورداستفاده قرار می‌دهد.

بات‌های اجتماعی می‌توانند برای تأثیرگذاری بر روی کاربران درون یک شبکه اجتماعی مورداستفاده قرار گیرند و قادرند برخی فعالیت‌ها شامل، پست کردن پیام‌ها و ارسال درخواست اتصال را انجام دهند. از این‌رو می‌توان از آن برای مقاصد بدی چون انتشار اخبار و اطلاعات نادرست و تبلیغات برای هدایت افکار عمومی به سمتی خاص استفاده کرد. با رشد شبکه‌های اجتماعی برخط و با وابستگی کاربران بیشتر به این شبکه‌ها، در سال‌های آینده انتظار می‌رود موارد بیشتری از شبکه‌های بات‌اجتماعی به وجود آیند، بنابراین می‌تواند به‌طور جدی برای موضوع تحقیق مورد توجه قرار گیرند.

۹-۲- کانال‌های پنهان

همواره برای مدیران بات‌نت، مخفی ماندن ارتباطات مهم بوده است و روش‌های زیادی نیز برای رسیدن به آن به‌کارگیری شده است. تحقیق [۷۷] بیان می‌کند که یکی از چالش‌های اصلی طراحی بات‌نت‌های آینده، پوشیده بودن آن‌ها است.کانال پنهان هر روش ارتباطی است که برای انتقال غیرمجاز اطلاعات مورداستفاده قرار می‌گیرد [۷۸]. کانال‌های پنهان به سه دسته انباشتی، زمانی و یا رفتاری تقسیم می‌شوند. درزمینه کانال‌های انباشتی مطالعات بیشتری ثبت شده است و در میان آن‌ها

1-Plohmann
2-kernel packers
3-SbN
4-On - line
5-hyperlink

کانال‌های شبکه‌ای فراوانی بیشتری در پیاده‌سازی و استفاده را دارا هستند. در این کانال‌ها، رسانه انتقال، محیط شبکه است یعنی، خطوط انتقال، مسیریاب‌ها، دیواره‌های آتش و مانند آن، انواع پروتکل‌هایی مثل TCP، IP، ICMP، MAC و AODV برای کانال پنهان مورداستفاده قرار می‌گیرند که به آن‌ها می‌توان روش‌های پنهان نگاری را افزود اگرچه برخی مراجع این روش را در زمره روش‌های صرفاً مخفی سازی اطلاعات به شمار می‌آورند [۷۸]. برای پنهان کردن اطلاعات معمولاً از دو روش کلی استفاده می‌شود، ایجاد بسته‌های جدید و تغییر بسته‌ها ازنظر محتوایی. در بخش تغییر محتوایی بسته‌ها در کانال‌های انباشتی و مبتنی بر پروتکل‌های شبکه نیز به دو طریق کلی استفاده از سرآیند[1] پروتکل و قسمت داده[2] پروتکل استفاده می‌شود که تغییر قسمت داده بسته‌ها به دلیل طول بیشتر و بدون ساختار بودن آن‌ها نسبت به سرآیند بسته‌ها، مکانی آسان‌تر برای انتقال اطلاعات پنهان هستند[۷۹]. نکاتی در طراحی و ارزیابی کانال‌های پنهان وجود دارد که از حوصله این نوشتار خارج بوده و محققین را به مراجع منتشرشده در این زمینه ارجاع می‌دهیم. و درمجموع روش‌های، بهره‌گیری از کانال‌های پنهان موضوعی است که می‌تواند در تحقیقات بعدی موردتوجه قرار گیرد.

۹-۳- تحلیل جریان ترافیک

وجود ترافیک ارسالی و دریافتی در شبکه و قدرت تحلیل آن از روش‌های مختلف موردتوجه محققان است. بررسی پارامترهایی مثل حجم ترافیک، مسیر و محتوای آن و شبیه‌بودن نظم ترافیک عبوری می‌تواند در بررسی بات‌نت‌ها مورداستفاده قرار گیرد و بر اساس آن روش‌های تشخیص مختلفی تاکنون ارائه شده است و به نظر می‌رسد که هنوز در این زمینه نیاز به تحقیق وجود دارد.

۹-۴- DNS

به دلیل ماهیت ارتباطی بات‌نت‌ها و این که در هر صورت برای ارتباط بات با مدیر خود نیاز به استفاده از یک روش ترجمه آدرس وجود دارد، بررسی این سرویس و نحوه برخورد با آن می‌تواند یکی از موضوعات مهم در تشخیص و بررسی عملکرد بات‌نت‌ها باشد که از آن جمله به شبکه‌های Fast- Flax نیز می‌توان اشاره کرد.

۱۰- نتیجه‌گیری

نخستین قدم لازم در مبارزه با تهدیدات بات‌نت، توسعه روش‌های تشخیص مؤثر است. این روش‌ها به دو دسته سیستم تشخیص مبتنی بر هاني‌نت و سیستم تشخیص نفوذ تقسیم می‌شوند. در روش اول، هدف، جمع‌آوری اطلاعات از بات‌ها برای پی بردن به رفتار و چگونگی تشخیص آن‌ها است و دومی به دو زیرمجموعه روش‌های مبتنی بر امضا و مبتنی بر ناهنجاری تقسیم می‌شوند. البته بیشتر مطالعات در زمینه تشخیص بات‌نت‌ها به بررسی روش‌های مبتنی بر ناهنجاری می‌پردازند.

پس از تشخیص، قدم مهم بعدی به‌دست آوردن راه‌هایی برای از کار انداختن زیرساخت‌های بات‌نت‌ها و مختل کردن عملکرد آن‌ها است. رایج‌ترین روش‌های به‌کار گرفته‌شده برای تحقق این کار شامل، قطع کانال فرمان و کنترل و جلوگیری از ارسال دستورات از سوی مدیر بات به بات‌ها هستند که در این مقاله پیشنهادهای مختلف و مرتبط ارائه شده است.

روش‌های متنوعی برای تشخیص بات‌نت‌ها وجود دارد که بررسی و بحث شد آنچه از منظر مدیر بات مطرح است این است که روش‌های فرار نیز درحال‌توسعه هستند که بات‌نت‌های موجود را مخفی کرده و با افزایش سرعت تغییر و از روش‌های مختلف، ردیابی آن‌ها را دشوار می‌سازند. امروزه بات‌نت‌ها برای عملیاتی شدن روی پایگاه‌های جدید شامل گوشی‌های هوشمند، تبلت‌ها و دیگر وسایل همراه فعال‌تر شده‌اند. در این حیطه چالش‌های فراوانی باقی‌مانده است. در این مقاله روش‌های تحقیقی جدیدی که قابل توسعه هستند بررسی و ارایه شد اما موضوع مهمی که محققان با آن مواجه هستند، دشواری آزمون و ارزیابی روش‌های تشخیص در سناریوهای واقعی یا در استفاده از داده‌های واقعی است. برخی ابتکار عمل‌ها، مانند ایجاد انبارهای ردپا، با درصدی موفقیت نسبی اجرایی شده‌اند، اما دسترسی به داده‌ها گاهی کنترل می‌شود یا محدود به برخی موارد خاص می‌شود.

۱۱- مراجع

[1] SANS Institute Info Sec Reading Room provided a description on "Bot & Botnet: An overview," research on topics in information security, 2003.

[2] H. Rouhani Zeidanloo, A. Bt Manaf, P. Vahdani, F. Tabatabaei, and M. Zamani, "Botnet Detection Based on Traffic Monitoring," IEEE transaction, 2010.

[3] C. Li, W. Jiang, and X. Zou, "Botnet: Survey and Case Study," 4th International Conference on Innovative Computing, Information and Control, 2009.

[4] Botnets cams are exploding. http://www.contentagenda.com/articleXml/LN760999245.html?industryid=45177.

[5] B. AsSadhan, J. Moura, D. Lapsley, C. Jones, and W. Strayer, "Detecting botnets using command and control traffic," in: Eighth IEEE International Symposium on Network Computing and Applications, pp. 156–162, 2009.

[6] P. Bacher, T. Holz, M. Kotter, and G. Wicherski, "Know Your Enemy: Tracking Botnets (using honeynets to learn more about bots)," Technical Report, The Honeynet Project, 2008.

[7] T. M. S. Labs, "M86 Security, Spam Statistics," 2011. <http://www.m86security.com/labs/spamstatistics.asp>.

1- Header
2- Payload

phenomenon," In: Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement, IMC'06, ACM, New York, NY, USA, pp. 41–52, 2006.

[27] R. Puri, Bots & Botnet: An Overview, SANS Institute InfoSec Reading Room, 2003.

[28] C. Schiller and J. Binkley, "Botnets: The Killer Web Applications," Syngress Publishing, 2007.

[29] L. Liu, S. Chen, G. Yan, and Z. Zhang, "BotTracer: Execution-Based Bot- Like Malware Detection," In: T. Wu, C. Lei, V. Rijmen, D. Lee (Eds.), Information Security, Lecture Notes in Computer Science, vol. 5222, Springer, Berlin/Heidelberg, pp. 97–113, 2008.

[30] T. Micro, "Taxonomy of Botnet Threats," Technical Report, Trend Micro White Paper, 2006.

[31] E. Cooke, F. Jahanian, and D. McPherson, "The zombie roundup: understanding, detecting, and disrupting botnets," In: Proceedings of the Steps to Reducing Unwanted Traffic on the Internet on Steps to Reducing Unwanted Traffic on the Internet Workshop, USENIX Association, Berkeley, CA, USA, p. 6, 2005.

[32] M. Jelasity and V. Bilicki, "Towards automated detection of peer-to-peer botnets: on the limits of local approaches," In: USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET'09), USENIX Association, Boston, MA, 2009.

[33] P. Maymounkov and D. Mazi'eres, "Kademlia: a peer-to-peer information system based on the xor metric," In: Revised Papers from the First International Workshop on Peer-to-Peer Systems, IPTPS'01, Springer-Verlag, London, UK, pp. 53–65, 2002.

[34] P. Wang, L. Wu, B. Aslam, C. Zou, "A systematic study on Peer-to-Peer botnets," In: Proceedings of 18th International Conference on Computer Communications and Networks, ICCCN 2009, pp. 1–8, 2009.

[35] P. Wang, S. Sparks, and C. Zou, "An advanced hybrid peer-to-peer botnet," In First Workshop on Hot Topics in Understanding Botnets, 2007.

[36] J. B Grizzard, V. Sharma, C. Nunnery, B. B Kang, and D. Dagon, "Peer-to-peer botnets: overview and case study," In: Proceedings of the First Conference on First Workshop on Hot Topics in Understanding Botnets, USENIX Association, Berkeley, CA, USA, p. 1, 2007.

[37] E. W. Middelesch, "Anonymous and hidden communication channels: a perspective on future developments," 2015.

[38] H. R. Zeidanloo, M. J. Shooshtari, P. V. Amoli, M. Safari, and M. Zamani, "A taxonomy of botnet detection techniques, 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT)," vol. 2, pp. 158–162, 2010.

[39] F. C. Freiling, T. Holz, and G. Wicherski, "Botnet tracking: exploring a rootcause methodology to prevent distributed denial-of-service attacks," Lecture Notes in Computer Science, vol. 3679, Springer, Berlin/Heidelberg, pp. 319–335, 2005.

[40] P. Wurzinger, L. Bilge, T. Holz, J. Goebel, C. Kruegel, and E. Kirda, "Automatically generating models for botnet detection," Lecture Notes in Computer Science, vol. 5789, Springer, Berlin/Heidelberg, pp. 232–249, 2009.

[41] J. Goebel and T. Holz, "Rishi: identify bot contaminated hosts by IRC nickname evaluation," Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets, USENIX Association, Berkeley, CA, USA, p. 8, 2007.

[42] Y. Kugisaki, Y. Kasahara, Y. Hori, and K. Sakurai, "Bot detection based ontraffic analysis," The 2007 International Conference on Intelligent Pervasive Computing, IPC, pp. 303–306, 2007.

[8] Z. Zhu, G. Lu, Y. Chen, Z. J. Fu, P. Roberts, and K. Han, "Botnet research survey, in: 32nd Annual IEEE, International Computer Software and Applications," COMPSAC'08, pp. 967–972, 2008.

[9] M. Feily, A. Shahrestani, and S. Ramadass, "A survey of botnet and botnet detection," in: Emerging Security Information, Systems and Technologies, SECURWARE '09, Third International Conference on, pp. 268–273, 2009.

[10] Y. Shin and E. Im, "A survey of botnet: consequences," defenses and challenges basic knowledge of botnet, Challenges, 2009.

[11] Symantec, "Spybot worm," 2003.

[12] J. Liu, Y. Xiao, K. Ghaboosi, H. Deng, and J. Zhang, "Botnet: classification, attacks, detection, tracing, and preventive measures," EURASIP Journal of Wireless Communication Networks, pp. 91–911, 2009.

[13] D. Plohmann, E. Gerhards-Padilla, and F. Leder, "Botnets: Detection, Measurement, Disinfection & Defence, Technical Report," The European Network and Information Security Agency (ENISA), 2011.

[14] B. AsSadhan, J. Moura, D. Lapsley, C. Jones, and W. Strayer, "Detecting botnets using command and control traffic, in: Eighth IEEE International Symposium on Network Computing and Applications," NCA, pp. 156–162, 2009.

[15] S. S. Garasia, D. P. Rana, and R. G. Mehta, "Http Botnet Detection Using Frequent Patterset Mining," IJESAT, vol. 2, Issue-3, pp. 619 – 624, May – Jun 2012.

[16] M. M. Masud, J. Gao, L. Khan, J. Han, and B. Thuraisingham, "Mining Concept-Drifting Data Stream to Detect Peer to Peer Botnet Traffic," ACM, 2012.

[17] S. S. C. Silva, R. M. P. Silva, R. C. G. Pinto, and R. M. Salles, "A Survey computer networks," Elsevier, 2012.

[18] EggHeads, "EggHeads.org-eggdrop development," 1993. <http://eggheads.org/>.

[19] T. Micro, "Worm AgoBot," 2004. <http://about-threats.trendmicro.com/ArchiveMalware.aspx?language=us&name=WORMAGOBOT.XE>.

[20] T. Micro, "Worm SDBot," 2003. <http://about-threats.trendmicro.
com/ArchiveMalware.aspx?language=us&name=WORMSDBOT. AZ>.\

[21] J. B. Grizzard, V. Sharma, C. Nunnery, B. B Kang, and D. Dagon, "Peer-to-peer botnets: overview and case study, in: Proceedings of the First Conference on First Workshop on Hot Topics in Understanding Botnets," USENIX Association, Berkeley, CA, USA, p. 1, 2007.

[22] S. S. C. Silva, R. M. P. Silva, R. C. G. Pinto, R. M. Salles, "A Survey computer networks," Elsevier, 2012.

[23] M. Fossi, G. Y. Egan, K. Haley, E. Johnson, T. Mack, T. Adams, J. Blackbird, M. K. Low, D. Mazurek, D. McKinney, and P. Wood, "Symantec Internet Security Threat Report – Trends for 2010," Technical Report Volume 16, Symantec, 2011.

[24] Freiling, C. Felix, T. Holz, and G. Wicherski, "Botnet tracking: Exploring a root-cause methodology to prevent distributed denial-of-service attacks," European Symposium on Research in Computer Security, Springer Berlin Heidelberg, 2005.

[25] H. Choi, H. Lee, and H. Kim, "BotGAD: detecting botnets by capturing group activities in network traffic," In: Proceedings of the Fourth International ICST Conference on communication System software and middleware, COMSWARE '09, ACM, NewYork, NY, USA, pp. 21–28, 2009.

[26] M. A. Rajab, J. Zarfoss, F. Monrose, and A. Terzis, "A multifaceted approach to understanding the botnet

Information, Systems and Technologies, Cap Esterel, France, August 2008.

[58] Y. Zhao, Y. Xie, F. Yu, Q. Ke, Y. Yu, Y. Chen, and E. Gillum, "BotGraph: Large Scale Spamming Botnet Detection", in Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation, Boston, MA, USA, April 2009.

[59] S. Nagaraja, P. Mittal, C.-Y. Hong, M. Caesar, and N. Borisov, "Botgrep: finding p2p bots with structured graph analysis," In: Proceedings of the 19th USENIX Conference on Security, USENIX Security'10, USENIX Association, Berkeley, CA.USA, p. 7, 2010.

[60] B. Shirley and C. D. Mano, "A Model for Covert Botnet Communication in a Private Subnet," Networking 2008, pp. 624-632, 2008.

[61] A. Nappa, A. Fattori, M. Balduzzi, M. Dell'Amico, and L. Cavallaro, "Take a deep breath: a stealthy, resilient and cost-effective botnet using skype," In: Proceedings of the 7th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, DIMVA'10, Springer-Verlag, Berlin, Heidelberg, pp. 81–100, 2010.

[62] T. Micro, "Taxonomy of Botnet Threats," Technical Report, Trend Micro White Paper, 2006.

[63] E. Stinson and J. C. Mitchell, "Towards systematic evaluation of the evadability of bot/botnet detection methods, In: Proceedings of the 2nd Conference on USENIX Workshop on Offensive Technologies, USENIX Association, Berkeley, CA, USA, pp. 5:1–5:9, 2008.

[64] D. Zhang, C. Zheng, H. Zhang, and H. Yu, "Identification and analysis of skype peer-to-peer traffic," In: Fifth International Conference on Internet and Web Applications and Services (ICIW), pp. 200– 206, 2010.

[65] J. Nazario and T. Holz, "As the net churns: fast-flux botnet observations," In: 3rd International Conference on Malicious and Unwanted Software, MALWARE 2008, pp. 24–31, 2008.

[66] A. Caglayan, M. Toothaker, D. Drapaeau, D. Burke, and G. Eaton, "Behavioral analysis of fast flux service networks," In: Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies, CSIIRW'09, ACM, New York, NY, USA, pp. 48:1-48:4, 2009.

[67] T. M. S. Labs, "Security Labs Report January – June 2011 Recap," Technical Report, Security Labs, 2011.

[68] T. Barabosch, A. Wichmann, F. Leder, and E. Gerhards-Padilla, "Automatic Extraction of Domain Name Generation Algorithms from Current Malware," In Proceedings of the NATO Symposium IST-111 on Information Assurance and Cyber Defense, Koblenz, Germany, September 2012.

[69] S. Yadav, A. K. Krishna Reddy, and A. L. Narasimha Reddy, "Detecting Algorithmically Generated Domain-flux Attacks with DNS Traffic Analysis," IEEE/ACM Transactions on Networking, vol. 20, no.5, pp. 1663–1677, October 2012.

[70] B. Stone-Gross, M. Cova, B. Gilbert, L. Cavallaro, C. Kruegel, G. Vigna, and R. Kemmerer, "Your Botnet is My Botnet: Analysis of a Botnet Takeover," In Proceedings of the 16th ACM Conference on Computer and Communications Security, pp. 635–647, Chicago, IN, USA, November 2009.

[71] R. sharifnia and M. Abadi, "A Novel Reputation System to DetectDGA-Based Botnets," In proceeding of the 3th ICCKE, Mashhad 2013.

[72] S. Yadav and A. L. Narasimha Reddy, "Winning with DNS Failures: Strategies for Faster Botnet Detection," In Proceedings of the 7th International ICST Conference on

[43] J. R. Binkley and S. Singh, "An algorithm for anomaly-based botnet detection," Proceedings of the 2nd Conference on Steps to Reducing Unwanted Traffic on the Internet, vol. 2, USENIX Association, Berkeley, CA, USA, p. 7, 2006.

[44] G. Gu, V. Yegneswaran, P. Porras, J. Stoll, and W. Lee, "Active botnet probing to identify obscure command and control channels," In: Computer Security Applications Conference, ACSAC'09, Annual, pp. 241–253, 2009.

[45] G. Gu, R. Perdisci, J. Zhang, and W. Lee, "BotMiner: clustering analysis of network traffic for protocol-and structure-independent botnet detection," Proceedings of the 17th Conference on Security Symposium, USENIX Association, Berkeley, CA, USA, pp. 139– 154, 2008.

[46] A. Nappa, A. Fattori, M. Balduzzi, M. Dell'Amico, and L. Cavallaro, "Take a deep breath: a stealthy, resilient and cost-effective botnet using skype," Proceedings of the 7th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, DIMVA'10, Springer-Verlag, Berlin, Heidelberg, pp. 81–100, 2010.

[47] H. Choi, H. Lee, and H. Kim, "BotGAD: detecting botnets by capturing group activities in network traffic," Proceedings of the Fourth International ICST Conference on COMmunication System software and middlewaRE, COMSWARE '09, ACM, New York, NY, USA, pp. 21–28, 2009.

[48] D. Dagon, C. Changchun Zou, and W. Lee. "Modeling Botnet Propagation Using Time Zones," *NDSS*. vol. 6, 2006.

[49] A. Madhukar and C. Williamson, "A longitudinal study of p2p traffic classification," 14th IEEE International Symposium on Modeling Analysis and Simulation, pp. 179-188, 2006.

[50] J. Erman, A. Mahanti, M. Arlitt, and C. Williamson, "Identifying and discriminating between web and peer-to-peer traffic in the network core," Proceedings of the 16th International Conference on World Wide Web, WWW'07, ACM, New York, NY, USA, pp. 883–892, 2007.

[51] C. Livadas, R. Walsh, D. Lapsley, and W. Strayer, "Using Machine Learning Techniques to Identify Botnet Traffic," In Proceedings of the 31st Annual IEEE Conference on Local Computer Networks, FL, USA, November 2006.

[52] J. Goebel and T. Holz, "Rishi: Identify Bot Contaminated Hosts by IRC Nickname Evaluation," In Proceedings of the 1st Workshop on Hot Topics in Understanding Botnets, Cambridge, MA, USA, April 2007.

[53] W. Wang, B. Fang, Z. Zhang, and C. Li, "A novel approach to detect irc-based botnets," In Proceedings of the International Conference on Networks Security, Wireless Communications and Trusted Computing, Wuhan, Hubei, China, April 2009.

[54] G. Gu, P. Porras, V. Yegneswaran, M. Fong, and W. Lee, "BotHunter: detecting malware infection through IDS-driven dialog correlation," In: Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium, USENIX Association, Berkeley, CA, USA, pp. 12:1–12:16, 2007.

[55] G. Gu, J. Zhang, and W. Lee, "BotSniffer – detecting botnet command and control channels in network traffic," In: 15th Annual Network & Distributed System Security Symposium, The Internet Society (ISOC), San Diego, 2008.

[56] P. Wurzinger, L. Bilge, T. Holz, J. Goebel, C. Kruegel, and E. Kirda, "Automatically generating models for botnet detection", in Proceedings of the 14th European Symposium on Research in Computer Security, Saint Malo, France, September 2009.

[57] I. Castle and E. Buckley, "The automatic Discovery, Identification and Measurement of Botnets," In Proceedings of the 2nd International Conference on Emerging Security

[73] H. Choi and H. Lee, "Identifying Botnets by Capturing Group Activities in DNS Traffic," Computer Networks, vol. 56, no. 1, pp. 20–33, January 2012.

[74] C.-Y. Huang, "Effective Bot Host Detection Based on Network Failure Models," Computer Networks, vol. 57, no. 2, pp. 514–525, February 2013.

[75] E. Kartaltepe, J. Morales, S. Xu, and R. Sandhu, "Social Network-Based botnet Command-and-Control: emerging threats and countermeasures," In: J. Zhou, M. Yung (Eds.), Applied Cryptography and Network Security, Lecture Notes in Computer Science, vol. 6123, Springer, Berlin/Heidelberg, pp. 511–528, 2010.

[76] J. Baltazar, J. Costoya, and R. Flores, "The real face of koobface," The largest web 2.0 botnet explained, In: Trend Micro Research, 2009.

[77] S. Nagaraja, A. Houmansadr, P. Piyawongwisal, V. Singh, P. Agarwal,and N. Borisov, "Stegobot: a covert social network botnet," In: Information Hiding Conf. (IH), 2011.

[78] C. Serdar, C. E. Adviser-Brodley, and E. H. Adviser-Spafford, "Network covert channels: design, analysis, detection, and elimination," 2006.

[79] Z. Sebastian, G. J. Armitage, and P. Branch, "A survey of covert channels and countermeasures in computer network protocols," IEEE Communications Surveys and Tutorials 9.1-4, 2007.

A Analytical Survey on Botnet and Detection Methods

M. R. Hasani Ahangar, R. Jalaei*

*Imam Hossein University

(Received: 17/01/2016 , Accepted: 31/10/2016)

ABSTRACT

Botnets have recently been identified as one of the most important and emerging threats to the security with hundreds of millions of computers compromised and infected in cyber space. Botnets are a network of compromised hosts or bots that are under the control of an attacker. They are considered as a primary root-cause for most of the attacks and fraudulent activities on the Internet, such as distributed denial of service (DDoS) attacks, phishing, spamming, information theft, and so on. Some studies show that about between 16 to 25 percent of computers which are connected to the internet all over the world are infected by bots and controlled by attackers. This pape, discusses in detail about Botnet and related research including Botnet evolution, life-cycle, command and control models, communication protocols, Botnet detection methods, Botnet mechanisms and their models, possible attacks performed by various types of Botnet. This paper also discusses the prominent research problems that have remained open and could be continued by researches.

Keywords: Botnet, Botmaster, Communication Protocols, Attacks, Honey Net, IDS

* Corresponding Author Email: Rjalaei@ihu.ac.ir