Botnets

Pablo Álvarez, Javier Sáez

27 de noviembre de 2017

Índice

1. Introducción

Hoy en día, debido al gran aumento producido en el uso de dispositivos informáticos y a la generalización del IoT (*Internet on Things*), nos preocupa cada vez más la seguridad en la red. Estamos expuestos a una gran variedad de ataques informáticos, mediante los cuales pueden robarnos información, destruirla, suplantar nuestra identidad, etc.

Es por esto que la seguridad en la red es uno de los temas más importantes en la actualidad y, en este trabajo, vamos a profundizar sobre uno de los mecanismos informáticos más usados para realizar ataques: las botnets, o redes de bots informáticos.

2. ¿Qué es una Botnet?

El término Botnet proviene de la unión de las palabras bot (robot) y net (network o red) y hace referencia a un conjunto o red de dispositivos informáticos, generalmente ordenadores, que son controlados de forma remota por otro equipo. El artífice de la botnet puede controlar todos los ordenadores/servidores infectados de forma remota usando canales de comunicación formados por protocolos de red basados en estándares como el *IRC* y el *HTTP*, de los que hablaremos más adelante.

Cada uno de los dispositivos que forman parte de la botnet pueden estar ejecutando uno o más bots, entendiendo por bot cualquier aplicación software que ejecute tareas automáticas (*scripts*) a través de Internet.

Como hemos dicho en la introducción, las botnets representan uno de los delitos cibernéticos más sofisticados y populares de hoy en día, ya que permiten a los hackers tomar el control de muchos equipos a la vez y convertirlos en equipos "zombis", que funcionan como parte de una gran red que se puede usar para propagar virus, generar spam y cometer todo tipo de delitos informáticos y fraudes.

Algunas botnets pueden englobar cientos o un par de miles de equipos informáticos, pero otras pueden llegar a controlar decenas e incluso centenares de miles de zombis, lo que supone un gran poder informático. Pero, ¿cómo se forman dichas redes?

3. Formación de una botnet

Ya hemos visto que las botnets son una amenaza creciente para la seguridad informática, así que ahora vamos a explicar cómo una botnet puede llegar a infectar cientos de miles de dispositivos.

En primer lugar, el ciberdelincuente crea un virus (el *bot*) y lo propaga a través de la red con el objetivo de que llegue a un gran número de equipos. La infección comienza, por tanto, cuando el usuario se descarga ese virus, lo que generalmente se produce al visitar determinados sitios web de dudosa fiabilidad.

En general, la forma en la que un equipo es infectado depende del sistema que estamos usando:

- En sistemas con Windows y MacOS, la forma más habitual de expansión de los bots suele ser mediante la distribución de software ilícito o sospechoso (generalmente cracks y archivos descargados con algún cliente P2P[1]). Dicho software suele contener malware, el cual puede escanear nuestra red de área local, disco duro, e intentar propagarse usando vulnerabilidades conocidas de Windows.

- En sistemas UNIX, la forma más usada para construir y expandir una Botnet es a través de telnet o SSH por medio de un sistema de prueba-error, esto es, probando usuarios y contraseñas comunes contra todas las IPs que se pueda de forma sistemática, o bien mediante ataques a bugs muy conocidos, que los administradores pueden haber dejado sin solucionar.

Una vez que el dispositivo ha sido infectado, el bot se conecta a un servidor C&C determinado, esto es, una infraestructura de mando y control (Command and control) que consta de servidores y otros elementos usados para controlar el malware, lo que permite que el creador de la botnet (botmaster) mantenga registros de cuántos bots están activos y en línea.

El botmaster ya tiene el control sobre los bots y puede llevar a cabo ataques informáticos e incluso vender su red a otras empresas. Generalmente, el valor de esta depende de la cantidad y de la capacidad de los bots. Los bots más recientes pueden incluso escanear automáticamente su entorno y propagarse utilizando vulnerabilidades y contraseñas débiles, lo que aumenta considerablemente su valor.

4. Estructura de una botnet

En este apartado vamos a centrarnos en explicar algunos aspectos técnicos de las botnets: su arquitectura, sus componentes principales, etc.

4.1. Arquitectura

La arquitectura de las botnets ha ido evolucionando en los últimos años con el objetivo de evitar que sean detectadas. Tradicionalmente, los bots eran vistos como clientes que se comunicaban a través de determinados servidores. Esto permitía a la persona que controlaba la red realizar el control desde una localización remota, para no ser descubierto. Muchas de las redes actuales se basan en redes P2P existentes. Estos bots P2P realizan las mismas acciones que en el modelo cliente-servidor, pero no requieren de un servidor central para comunicarse, lo que las hace aún más indetectables.

4.1.1. Modelo cliente-servidor

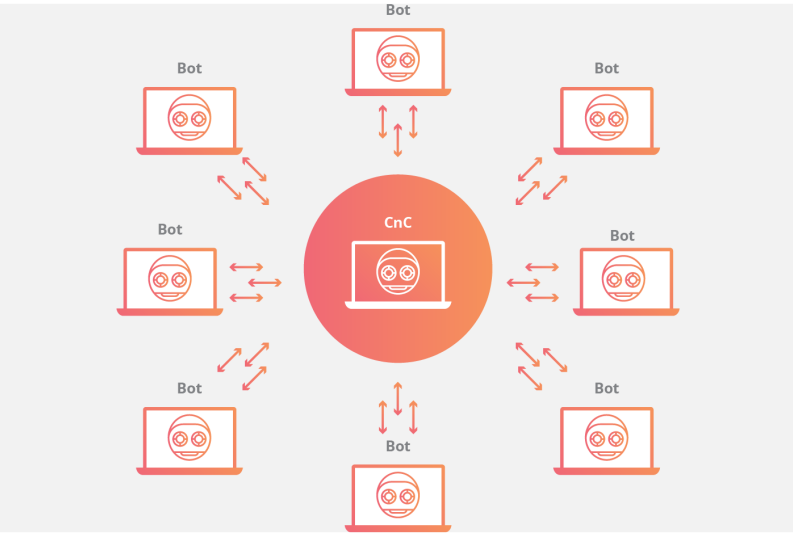
Fue el primer modelo que se creó para botnets. En este modelo, cada máquina individual se conecta a un servidor centralizado (o una pequeña cantidad de estos) para acceder a la información. Cada bot se conectará a un recurso del centro de

[1]Una red P2P (peer-to-peer) es una red de ordenadores en la que todos o algunos aspectos funcionan sin clientes ni servidores fijos, sino una serie de nodos que se comportan como iguales entre sí.

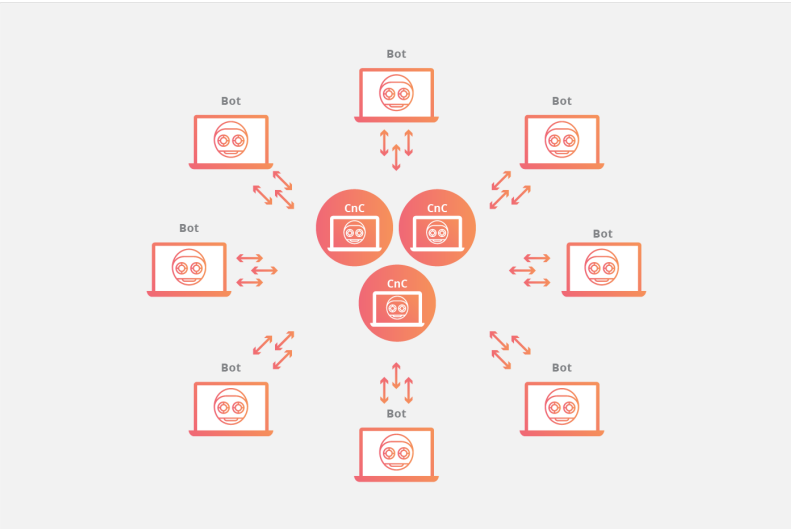
comando y control (C&C) como un dominio web o un canal IRC[2] para recibir instrucciones. Al usar estos repositorios centralizados para servir nuevos comandos para la botnet, un atacante simplemente necesita modificar el material de origen que cada botnet recibe de un servidor C&C para actualizar las instrucciones a las máquinas infectadas. El servidor centralizado que controla la botnet puede ser un dispositivo del atacante, o puede ser un dispositivo infectado.

Existen distintas topologías conocidas de botnets centralizadas, como son:

- Topología de estrella: un servidor central controla cada uno de los bots.

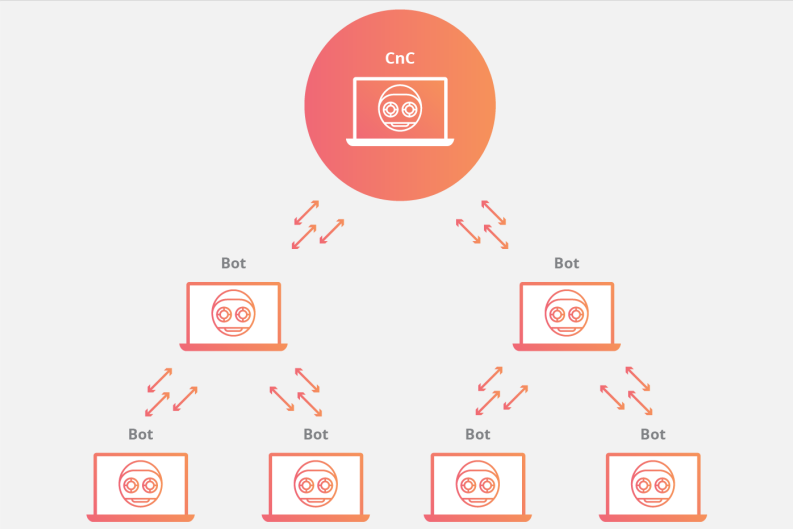


- Topología multi-servidor: los bots son controlados por varios servidores centralizados.



- Topología jerárquica: se establece una relación de jerarquía en la que varios bots controlan otro grupo de bots.

[2]Internet Relay Chat (IRC) es un protocolo de comunicación en tiempo real basado en texto, que permite que dos personas se comuniquen pero que se diferencia con la mensajería instantánea en que los usuarios no acceden a establecer la comunicación de manera previa, así que dos usuarios pueden comunicarse sin tener que haberlo hecho nunca antes.



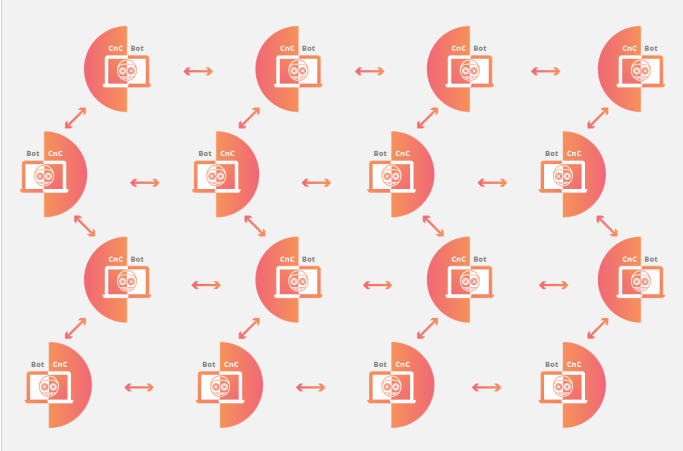
Sin embargo, este modelo tiene un gran vulnerabilidad: para eliminar una botnet con un servidor centralizado, basta con interrumpir dicho servidor. Como resultado de esta vulnerabilidad, los creadores de malware de botnet han evolucionado y se han movido hacia un nuevo modelo que es menos susceptible a estas interrupciones.

4.1.2. Modelo P2P

Para evitar las vulnerabilidades del modelo de cliente/servidor, las botnets más recientes se han diseñado utilizando componentes de distribución de archivos punto a punto descentralizada. Incluir la estructura de control dentro la propia botnet elimina el único punto débil presente en una botnet con un servidor centralizado, dificultando su desmantelamiento. Los bots P2P pueden ser clientes y centros de comando, que trabajan junto con sus nodos vecinos para propagar datos.

Las redes de bots P2P mantienen una lista activa de computadoras de confianza con las que pueden dar y recibir órdenes y actualizar su malware. Al limitar el número de máquinas a las que se conecta cada uno de los bots, cada bot solo está expuesto a dispositivos adyacentes, lo que hace que la red sea más difícil de rastrear. Sin embargo, al carecer de un servidor de comando centralizado, una botnet P2P es más vulnerable al control de otra persona que no sea el creador de la botnet. Para protegerse contra la pérdida de control, las botnets descentralizadas suelen estar encriptadas usando firmas digitales, para que el acceso sea limitado. Ejemplos son Gameover ZeuS y ZeroAccess, que serán comentadas más adelante.

Con esta arquitectura se evita también que si hay un error en alguna parte del sistema, caiga el sistema entero, pues la información está en todos los infectados. De hecho, para encontrar infectados, cada bot infectado prueba direcciones IP aleatorias hasta encontrar otro infectado, que le contesta con su versión del software y le indica los otros bots infectados que este bot ya ha detectado. Si las versiones del software son diferentes, el que tenga la versión más actualizada manda los archivos al otro bot para que pueda actualizarse, y este se lo manda a los bots que ya conocía, por lo que la red sigue creciendo de forma automática.



4.2. Componentes

Las redes botnet están compuestas de varias piezas clave que hacen que la botnet funcione. Estas son:

- Botmaster (o 'pastor', *herder*): se encarga de controlar la botnet de forma remota. Es conocido como el C&C, command and control en inglés, pues se encarga de realizar las órdenes y controlar la botnet. Este pastor realiza un ataque llamado 'Covert channel', que consigue que dos objetos, que se supone que no deben comunicarse según las normas de seguridad, se comuniquen.

- Protocolos de control: IRC, como ya hemos comentado, es un protocolo de control. Un pastor crea un canal con el protocolo IRC para que los clientes infectados se unan al canal y así , si manda un mensaje por el canal le llegará a todos los zombies. Un ejemplo de mensaje puede ser:

  $$: herder!herder@channel.comTOPIC\#attackDDoSwww.google.es$$

  que mandaría atacar a todos los zombies del canal al sitio web 'www.google.es'. Luego, los bots podrían ofrecerle una respuesta del estilo:

  $$: botA!botA@infected.netANSWER\#attackIamattackingwww.google.es$$

  Contestando que está atacando a la página.

  Algunas botnets implementan versiones propias de protocolos conocidos, y las diferencias en la implementación pueden ser usadas para detectar las redes maliciosas. Un ejemplo es MEGA-D (Ozdok), la botnet responsable de mandar el 32 % del spam en todo el mundo, descubierta en 2008, que implementa una modificación del protocolo SMTP para comprobar si puede hacer Spam. Sin embargo, 'tirar' el servidor SMTP de Mega-D hace que la red entera caiga.

- Zombies: Un zombie es un dispositivo que ha sido infectado por un hacker, troyano o virus y que puede ser usado para realizar tareas maliciosas . Es mandado por un pastor. Los zombies suelen ser usados para enviar Spam o

hacer ataques DDoS. La mayoría de las veces, los usuarios no saben que su computador es un zombie y está siendo usado para estos fines. 'Scrimping' es el término que se usa para denominar el momento en el que un ordenador pasa a formar parte de una botnet y dar por tanto sus recursos a la red.

4.3. Protocolos de control

Como hemos comentado, el pastor utiliza el modelo C&C para manejar la red de zombies. Los protocolos que se usan para controlarlos han ido variando con el tiempo, desde IRC como ya hemos comentado hasta versiones mucho más sofisticadas. Las más importantes han sido:

- Telnet: Las botnets que utilizan telnet usan un protocolo simple en el cual el bot se conecta al servidor que manda para hostear la red. Los bots se añaden a la red mediante un script que es ejecutado en un servidor externo y que escanea direcciones IP para poder conectarse mediante Telnet o SSH. Cuando un candidato es encontrado se añade a la lista de infectados y es infectado mediante SSH, siendo convertido en nuevo esclavo. Cuando los servidores están infectados por el controlador, se pueden lanzar los ataques. Famosos ataques con botnets que usan Telnet han conseguido tirar páginas de internet como la de *Xbox* y *PlayStation*.

- IRC: Las redes IRC utilizan comunicaciones sencillas y de bajo ancho de banda. Así, es fácil controlar a los bots, son fáciles de construir y tienen bastante éxito en la coordinación de ataques de denegación de servicio y envío de Spam, a la vez que pueden evitar sencillamente que los ataques sean parados, cambiando el canal de comunicación. Sin embargo, como hemos comentado antes, se consiguió encontrar una manera efectiva de parar las botnets que usan este protocolo, y es el bloqueo de ciertas palabras clave en la comunicación, que hace que esta pierda el sentido para los bots y el ataque cese. Otro problema de IRC es que cada zombie debe saber cuál es el servidor, puerto y canal para poder formar parte del ataque, así que las personas que trabajan en seguridad pueden detener estos canales y servidores para parar el ataque, aunque los clientes sigan infectados, su ordenador no puede formar parte del ataque pues no tiene forma de recibir las instrucciones aunque, como ya hemos explicado, tienden a cambiar de canal de comunicación para evadir este problema.

- P2P. Ya hemos visto que surgió para evitar los problemas de corte de canales en los ataques y evolucionar así a una nueva forma de atacar más efectiva. Se han usado también métodos para encriptar la botnet mediante la encriptación con clave pública, pero esto es tan dificil de implementar como de romper.

- Dominios: Las redes más grandes suelen usar dominios en vez de IRC (véase Rustock y Srizbi). Estos son hosteados con servicios muy difíciles de romper. Si un zombie accede a una página o dominio específicos, obtendrá las órdenes que tiene que hacer. Así, los pastores consiguen como ventaja que pueden mantener una red muy grande con un código muy sencillo. Sin embargo, también necesitan para ello, al contrario que IRC, un gran ancho de banda y los dominios pueden ser fácilmente tirados por agencias del gobierno sin esfuerzo

o pueden ser atacados. Para evitar estos ataques, los pastores utilizan Fast-flux DNS, una técnica en la que se esconde el software malicioso mediante el cambio constante de la red y la actuazión de los hosts como proxies. Así, es dificil controlar los servidores, que pueden incluso cambiar de dominio mediante ciertos algoritmos.

- Extras: Muchas veces, para evitar ser rastreadas de forma sencilla, las botnets utilizan servicios como Github, Twitter y otras redes sociales varias y servicios de mensajería instantánea.

5. Usos habituales de las botnets

5.1. Ataques DoS/DDoS

Un ataque de denegación de servicio consiste en una red de ordenadores que causa que un servicio sea inaccesible a otros usuarios. Esto se suele hacer consumiendo todo el ancho de banda del servicio víctima o sobrecargando los recursos computacionales del recurso atacado. Lo que se pretende es saturar los puertos con múltiples flujos de información y que así el servidor se sobrecargue y no pueda prestar el servicio.
Generalmente, estos ataques se hacen de forma distribuida, por lo que denominan ataques DDoS (Distributed Denial of service attack), generando el flujo de información desde varios puntos de conexión, usando por ejemplo una botnet.

5.2. Envío de Spam

Hacer Spam es hacer uso de sistemas de comunicación electrónica para enviar repetidamente mensajes usualmente no deseados y que suelen contener publicidad. La forma más común de hacerlo es mediante e-mail, pero también puede hacerse mediante mensajería instánea, buscadores de red, anuncios en blogs o anuncios en la red. Los spammers consiguen las direcciones a las que enviarán el spam mediante sitios web, grupos de noticias, grupos de correos electrónicos que mandan chistes, páginas en las que se solicita tu correo para identificarte o incluso por ensayo y error. Una vez que los tienen utilizan un sistema que recorre la lista de usuarios enviando el mensaje a todas las direcciones de forma automática mendiante una botnet.

5.3. Abuso de los servicios de pago por publicidad

Las botnets también son utilizadas para abusar por ejemplo de Google AdSense, que permite a las compañías poner publicidad de google en sus páginas de internet y ganar así dinero. Entonces, el atacante usa su botnet para hacer clicks en los anuncios de forma automática y generar así un beneficio.

5.4. Manipulación de encuestas o juegos online

Los juegos y encuestas online están ganando mucha popularidad y es relativamente sencillo manipularlos con botnets. Como cada vot tiene una dirección IP diferente, en una encuesta cada voto tendrá la misma credibilidad que un voto de una persona real. En los juegos online, también se puede manipular de una forma similar, simulando jugadores.

5.5. Robo de información masivo

En ocasiones, la combinación de varios de los anteriores usos de las Botnets pueden ser usados para obtener información de los usuarios. Por ejemplo, existen emails que falsean ser grandes compañías de bancos o ser Paypal y piden a las víctimas que envíen su información privada por correo. Estos emails son enviados de forma masiva por una botnet mediante spam. También pueden hostear con los zombies websites falsas que simulen ser este tipo de servicios y recolectar información cuando una víctima intenta hacer LogIn en una de estas páginas.

5.6. Minería de bitcoins

Minar un bitcoin consiste en coger un bloque de memoria que contenga ciertos campos y conseguir, mediante la elaboración de muchos cálculos, validar este bloque. Por tanto, las botnets se benefician de esto pues cada zombie en la botnet puede estar validando distintos bloques a la vez y así el pastor consigue validar muchos más bloques en menos tiempo, por lo que consigue así muchos más bitcoins.

5.7. Computación paralela

Las botnets son al fin y al cabo dispositivos que trabajarán de forma paralela. Así, pueden ser usados por empresas que se dediquen a procesar muchos datos para hacer la tarea más eficiente, de forma similar a la minería de bitcoins. Podemos destacar dos de ellas que son SETI(Search for Extraterrestrial Intelligence) que , como su nombre indica, se dedican a buscar inteligencia extraterrestre. Esta empresa recibe las ondas del espacio e intentan interpretar los datos digitalmente. Por tanto, cuanto más poder computacional tengan, más rango de onda pueden cubrir. Otra empresa importante es GPUGRID, que se dedica a investigación biomédica y utiliza las GPU de los zombies para simular biomoléculas. Sin embargo, estas simulaciones requieren mucha capacidad computacional y suelen requerir supercomputadores.

6. Indicios y contramedidas

6.1. Cómo saber si estoy infectado

Como ya hemos visto anteriormente, la estructura de las botnets hace que sean casi imperceptibles, lo que nos puede llevar a preguntarnos si realmente podemos saber si formamos parte de una botnet.

Lo habitual es que los bots permanezcan ocultos hasta que se les ordene llevar a cabo alguna tarea. Dependiendo de la tarea, ésta puede hacer que nuestro equipo

se ralentice (sobre todos en dispositivos antiguos o con poca capacidad de procesamiento), muestre mensajes misteriosos e, incluso, puede llegar a ocasionar fallos concretos, pero esto no tiene por qué significar que estamos infectados, ya que estos problemas pueden deberse a la ocurrencia de otras circunstancias.

Además de estos indicios, siempre podemos usar herramientas y productos específicos que nos ayuden a confirmar nuestras sospechas.

6.2. Contramedidas

Como ya hemos dicho, darnos cuenta de que pertenecemos a una botnet no es una tarea fácil, al igual que, por desgracia, evitarlo.

Evitar pertenecer a una botnet depende en gran medida de cuestiones relacionadas con el sistema operativo que se esté usando y del software de seguridad del que dispongamos, además de las acciones que lleve a cabo el usuario en la red. Por tanto, nunca está de más usar un software de seguridad actualizado y desconfiar de enlaces y páginas web sospechosas.

A continuación vamos a mencionar algunas webs y herramientas que pueden sernos útiles:

- La Oficina de Seguridad del Internauta (OSI) nos ofrece un servicio 'Antibotnets', disponible en `https://www.osi.es/es/servicio-antibotnet`, con el cual podemos conocer si desde nuestra conexión se ha captado alguna brecha de seguridad relacionada con botnets, comprobando si nuestra dirección IP está relacionada con incidentes registrados en su base de datos. En caso de que así sea, esta plataforma nos proporciona diferentes servicios y herramientas con las que podemos proceder a desinfectarnos. También cabe la opción de descargar un *plugin* que nos informe de manera continua de nuestra situación al navegar por la red.

- El Instituto Nacional de Ciberseguridad (INCIBE) también cuenta con una herramienta gratuita que nos muestra automáticamente una alerta en el caso de que alguna de las direcciones IP de una empresa forme parte de una botnet.

- También podemos mencionar varios antivirus específicos, como *Virus Removal Tool* de Kaspersky, *Avira EU Cleaner*, *Constant Guard*, *Norton Power Eraser* y *RuBotted*. Este último monitoriza nuestro ordenador para detectar infecciones potenciales y actividades sospechosas relacionadas con bots y, una vez detectados, los identifica y los elimina.

- Por último, es recomendable emplear un navegador de última generación y limitar los derechos de usuario cuando estemos conectados.

Por otro lado, debemos resaltar que habitualmente las botnets usan un servidor de comando y control (C&C) para conectar todos los bots, por lo que basta con cerrar ese panel para derribarlas, aunque no siempre es tan fácil, ya que pueden usar diversos servidores distribuidos por todo el mundo.

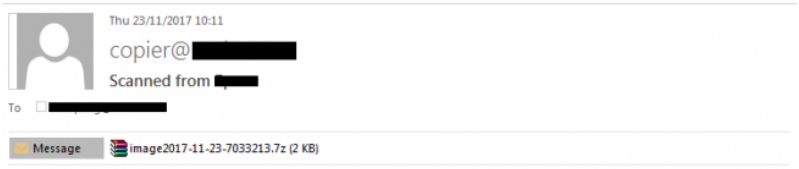
También existen botnets que usan redes P2P para comunicarse, por lo que no existe un servidor centralizado, haciendo más difícil su eliminación.

7. Ejemplos de botnets famosas

En esta sección vamos a mencionar algunos casos de ataques famosos que se han llevado a cabo mediante botnets, algunos de ellos bastante recientes.

- Botnet Necurs: El 23 de noviembre la firma de seguridad *Forcepoint* descubrió que una enorme campaña de *phising* había enviado hasta 12 millones de emails en apenas 6 horas. Dichos emails contienen el *ransomware*[3] Scarab, un malware bastante peligroso que cifra los ordenadores con Windows, desactiva las opciones de recuperación y exige al usuario que realice un pago cuyo valor va a ir incrementando conforme pase más tiempo. Esta campaña ha utilizado la botnet Necurs, la más grande del mundo, para llevar a cabo su cometido.

  El *ransomware* Scarab ya fue detectado en junio, y se propaga a través de emails en los que se lee *'Scanned from HP/Lexmark/Canon'* y contienen un archivo con formato *7zip* adjunto, el cual es un *script* que descarga el *ransomware* al ordenador.



  Una vez que infecta al usuario, añade una copia del archivo *sevnz.exe* en la carpeta App Data, con un mensaje de rescate que te informa de que todos tus archivos han sido cifrados debido a un problema de seguridad de tu PC y te piden que envíes un email con tus datos para confirmar que estás dispuesto a pagar en bitcoins por la llave de descifrado.

  Los expertos advierten que este tipo de ataques será cada vez más común, incluso Google afirma que ser víctima de *phising* es 400 veces más peligroso que formar parte de una brecha de datos.

- Botnets extractoras: Las botnets extractoras son un tipo de *malware* que se ocultan en los ordenadores y son capaces de instalar aplicaciones de extracción de bitcoins y otras criptomonedas. En lo que va de año se han llegado a registrar 1.65 millones de estos ataques.

  Según Kaspersky Lab, estos ataques se realizan mediantes 2 botnets, una de ellas formada por 5000 ordenadores con la que se han obtenido más de 165000 euros al mes y otra con 4000 equipos con la que se han embolsado otros 25000 euros mensuales.

  El malware que contienen estos equipos permite crear las monedas virtuales con tecnología *blockchain* y lleva a cabo la minería de forma oculta. Dicho

[3]Un ransomware (del inglés ransom, 'rescate', y ware, por software) es un tipo de programa dañino que restringe el acceso a determinadas partes o archivos del sistema infectado, y pide un rescate a cambio de quitar esta restricción.

malware se distribuyó mediante programas de *adware*[4] que las víctimas instalaron.

Una vez infectados, el virus descarga el extractor de monedas, lo instala y ejecuta algunas actividades para asegurarse de que trabaja todo el tiempo posible. En cuanto la primera moneda se extrae, se transfiere a las carteras de los criminales, dejando a las víctimas con u ordenador que no rinde lo suficiente y unas elevadas facturas de luz. Se suelen extraer las criptomonedas *Zcash* y *Monero*, pues ofrecen una forma viable de anonimizar las transacciones y las carteras de los propietarios.

- GameOver Zeus: A mediados de junio de 2014, el FBI, en conjunto con otras organizaciones de seguridad, publicó un comunicado anunciando el desmantelamiento de la botnet asociada a la amenaza conocida como GameOver Zeus, responsable de miles de infecciones e íntegramente relacionada con Cryptolocker, un ransomware bastante peligroso.

  GameOver Zeus es una variante de un conocido troyano bancario que se propaga por Internet y es ampliamente utilizado por los cibercriminales para el robo de credenciales. GameOver Zeus no utilizaba el protocolo HTTP para su comunicación, sino que utilizaba redes P2P, por lo que su desmantelamiento fue bastante complicado.

  Esta red constituía una gran amenaza a la privacidad de los usuarios y sobre todo al estado de sus cuentas bancarias, pero los métodos que implementa para el robo de información continúan siendo eficaces y le permiten a un atacante, sin mucho esfuerzo, obtener usuarios y contraseñas. Este método se basa en la inyección de código en el navegador y otros procesos del sistema, lo que le permite al malware interceptar el envío de usuarios y contraseñas para luego reenviarlas a su panel de control.

- ZeroAccess botnet: ZeroAccess es un malware de tipo troyano que infecta ordenadores con sistema operativos Windows, pasando estos a ser parte de una red de bots.

  Los ordenadores infectados tienen como principal objetivo la minería de Bitcoins (usando la red de ordenadores infectados para realizar las complicadas operaciones necesarias para generar esta moneda) y el fraude en campañas online de anuncios (mediante la visita de determinados sitios o anuncios en los que los anunciantes pagan una cantidad de dinero por visitas).

  El virus es distribuido principalmente a través de sitios comprometidos o maliciosos, en los cuales, mediante anuncios o campañas de spam se intenta dirigir a los usuarios a visitar paginas comprometidas preparadas para aprovechar alguna vulnerabilidad en el sistema para infectarlo. También se usa ingeniería social, es decir, un conjunto de técnicas que permiten sacar partido de la confianza del usuario para lograr que ejecute un programa malicioso que infectará el ordenador con ZeroAccess.

[4]Un programa adware es cualquier programa que automáticamente muestra u ofrece publicidad no deseada, ya sea incrustada en una página web mediante gráficos, carteles, ventanas flotantes, o durante la instalación de algún programa al usuario, con el fin de generar lucro a sus autores.