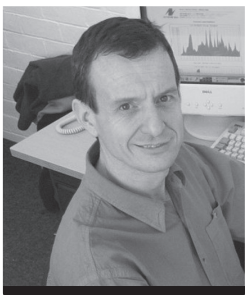
Working the botnet: how dynamic DNS is revitalising the zombie army



Simon Heron

Simon Heron, Managing Director, Network Box (UK)

Years ago hackers were little more than script kiddies who busied themselves in their bedrooms writing malicious code. They caused disruption for kicks, and their reward was kudos from the hacking community.

Those days are behind us. The criminal underground has found its way into online scams, creating a boom in cybercrime. Criminals have realised that they can make a lot of money with relatively little risk. From the criminal's point of view, stealing money electronically poses far less danger than breaking into someone's home or a commercial property and rifling around for valuables. Botnets have helped to facilitate this digital theft.

How botnets began

Botnets emerged in the late nineties, as ecommerce proliferated and the internet's user base grew. Improvements in PC processing speeds and storage capacity, coupled with ever-increasing broadband penetration rates, have created an abundance of relatively insecure, large capacity PCs waiting to be commandeered. Once compromised, these PCs become 'bots', ready to carry out the commands of the botnet controller, or botmaster.

A typical use of botnets is running spamming campaigns. We can see their effect through the global spread of spam sources, as outlined in figure 1.

A botmaster running a spam campaign sends one copy of the spam email to an infected machine. The bot in turn passes it on to other bots in the network. The spam is then forwarded to a list of email addresses which are shared between the bots. Because the capacity and bandwidth belongs to the PC's owner, the botmaster's cost is minimal. Furthermore, any attempt to trace the source of the spam would lead only to the infected PCs.

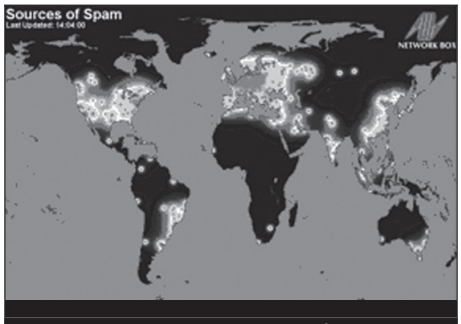


Figure 1: Worldwide sources of spam

Island hopping botnets

A recent ploy favoured by spam-issuing botmasters is island hopping. This tactic operates using the country code top-level domains (ccTLDs) of small island countries as part of the web links contained in spam mails, rather than top-level domains such as .com, .net and .info. These domain names are favoured because they are typically unknown to spam filters, meaning they are less likely to be detected. Among the glut of junk emails from island domains that have been seen are .im, for the Isle of Man, and .st, for Sao Tome. However, despite initially being caught cold, spam-filtering houses are adding these island domains to their spam filters.

Another common use for botnets is to garner information. Most PCs contain sensitive information about the user, and that information can be very profitable if it lands in the wrong hands. A botnet is often used to steal PINs, passwords and other sensitive information from home PC users, which is then fed back to the botmaster. He/she can then use the data to commit fraud and siphon money from the PC owner's bank account, or sell the information on to other criminals.

The repercussions of a botnet attack on a business can be even more serious. If a botmaster infiltrates an organization and turns its PCs into bots, options for exploitation include accessing company usernames and passwords, and financial and confidential information that could cause irreparable damage to a company's reputation.

Harvesting and controlling a huge number of PCs gives the botmaster enormous power. For example, he can launch distributed denial of service (DDoS) attacks for revenge or profit. A DDoS attack is a process used to saturate a server with requests for data so that other users' legitimate requests will not be answered. During such an attack, the attacker must be able to launch many requests from many locations. A single ADSL connection would not be sufficient to achieve this, and it is also easy to block.

Phishing with botnets

Distributing the attack also has the advantage of ensuring that the attacker is untraceable. With many IP addresses flooding a single target, there is no way to tell where an attack is really coming from. And, worryingly, there is a trend for fine-tuning DDoS attacks. For example, bots which were once limited to their creator's control can now be controlled though a simple user interface, namely internet relay chat (IRC). There is also evidence that

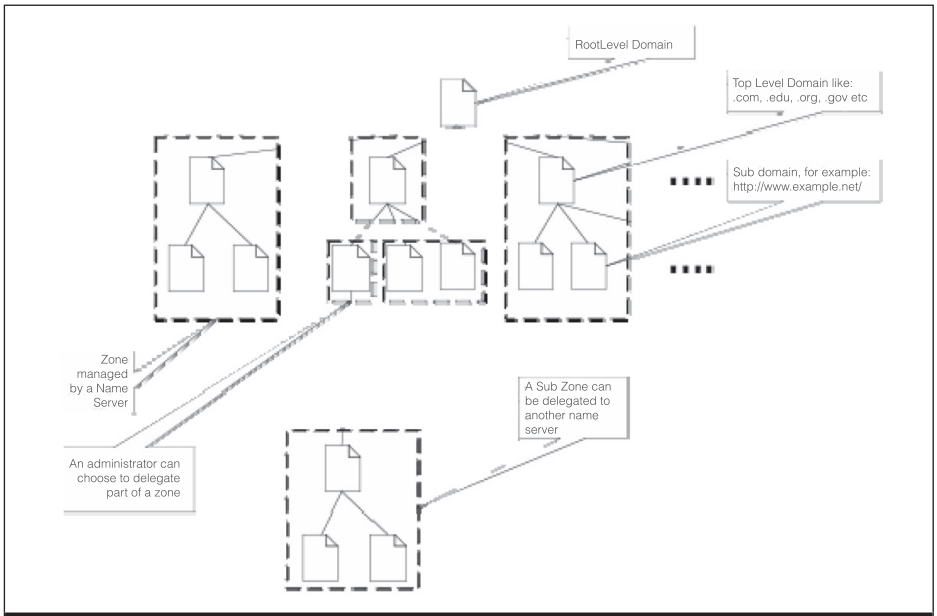


Figure 2: DNS architecture

these botnets are now updating themselves to the latest version of their software in a similar way that Windows Updates get the latest patches for Windows.

The boom in the theft of personal information has also given rise to botnet malware with phishing capabilities. Advanced malware triggers the bot to display a web page mimicking a legitimate site. The user then inputs a username and password into what they believe to be the genuine site. These details will then be logged and forwarded to the botmaster.

Typically, social engineering is used to mask such an attack. This can be done either by forwarding the log-in details to the genuine website after submitting it to the botmaster's server so that the user thinks they have entered the site in the normal way, or by displaying an error message before redirecting the user to the real site to input their details again.

Exploiting DNS

It's clear that botnets and spammers are becoming more adaptive as new, tighter security measures are being introduced. Botnets themselves are as versatile as they are valuable. It follows, then, that being able to make a bot even more agile, to be able to exploit even more vulnerabilities, is an exciting prospect for the botmaster.

One way of doing this is by exploiting dynamic naming system (DNS) servers. DNS servers translate regular domain names, such as domain.com, into numeric IP addresses, such as 215.10.11.12. Figure 2 shows the DNS structure.

When somebody enters a web address into a browser (say www.example.net/products), the browser needs an IP address for this name. The browser sends a request to its designated domain name server for a translation. The DNS server receives the request and searches its database for a resource record for www.example.net. If it finds the record, it will return the IP address associated with it.

However, if the DNS server fails to resolve the address, it makes a request to its parent server asking for name resolution. This request is iterated up the tree until it reaches the authoritative server for the requested domain. This search continues until either the originating DNS server receives an answer or a Time to Live (TTL) is reached. TTL is a time limit set for an IP host requesting a response from another host. If TTL expires, the DNS server returns an error code to the requesting client and the browser issues an error. A successful resolution returns the IP information to the requesting client browser.

Once the DNS server receives resolution for the address www.example.net, it passes the IP address back to the requesting client and the page can be located and served.

Traditionally, a static IP address is needed to host a website. However, by using dynamic DNS, websites can be hosted from an IP address that is constantly changing. Dynamic IP addresses are given out by ISPs, predominantly to home users, either using dial-up or broadband connections. The advantage of dynamic DNS for the user is that it allows a domain to be hosted from a dynamic IP address. Each time a user connects to the internet, they receive a new IP address which in turn is updated against their domain name registered with the dynamic DNS service using third party client software. This way, their website or domain is always available, regardless of the IP address it is using.

Devilish uses for dynamic DNS

However, this service can be used for illegitimate purposes. Criminals can use it to keep phishing sites online for longer. Let's imagine a fictional domain registered with the dynamic DNS service called example.net. dynamic DNS holds an IP address for that site. However, the website at its IP address is a phishing site. When an unsuspecting PC user clicks on a link to example.net they are redirected to the IP address and are met by what appears to a legitimate site. Upon entering their username and password on the fake site, their details are logged and forwarded to the botmaster. The botmaster is then free to use these log-in details.

When the phishing site is discovered and subsequently removed, users accessing example.net end up at a website that no longer exists. When this happens, the botmaster simply logs into the dynamic DNS account and changes the IP address of the fake site to point to a new address hosting another phishing site. Then, the phishing site is active once more.

The same process is used to produce FTP sites for illegal content. When surfers follow a web link to download the illegal content they are redirected to the zombie PC that is unknowingly hosting the content. When this exploitation is

discovered and fixed by the owner of the zombie PC, rendering the links to the hosted content redundant, the criminal redirects the domain name to another zombie PC's IP address. This process can be automated, with the zombie PCs announcing that they are up and the dynamic DNS server choosing one.

Fighting online crime with databases

One way of combating phishing attacks is to build databases that contain lists of all known phishing websites. Some anti-virus vendors already maintain such databases, but they are not made publicly available. The only people to benefit from them are the customers of the anti-virus companies.

OpenDNS, a company that specializes in domain name resolution, is more astute. Rather than keeping the information to itself, OpenDNS has created a phishing database that can be accessed by anyone. When users find a phishing website they can submit the website's address to the OpenDNS database. Other users can then vote on whether it is actually a phishing site or not. Obviously, this database relies on a motivated user base to submit suspicious sites to the database and to verify the validity of each submission.

But even the free-to-access model has its flaws. Any database – free or otherwise – is constantly playing catch-up. As soon as one phishing site is taken down, another one springs up in its place. Furthermore, criminals circumvent phishing databases by using zombie PCs to operate as rogue DNS servers on compromised systems. These rogue DNS servers are similar to dynamic DNS, except that the DNS server, rather than being maintained by a legitimate company, is manually maintained by the botmaster on one of his infected machines.

Fixing DNS

One solution is to make the DNS system more secure. There is a suite of IETF specifications called Domain Name System Security Extensions (DNSSEC), intended to secure certain kinds of information provided by the Domain Name System. This suite can be used to validate the origin of DNS data, and ensure DNS data integrity.

Many believe that deploying DNSSEC will help in securing the Internet as a whole, but there are challenges, including implementing a backwardly compatible standard that can scale and work across a wide variety of DNS servers and clients. Work is nevertheless in progress to resolve problems and there are instances of DNSSEC being implemented.

Another malicious use of dynamic DNS ensures that a botnet is kept online. It does this by checking the validity of the IP address hosting the bot control server, which keeps the zombie PCs operational. Years ago, many botnets could be disabled by isolating and shutting down the control server to which all the bots reported. Compromised PCs then had no central server to report to, or receive instructions from, and were essentially useless.

Dynamic DNS solves this problem for botmasters, by continuously enabling the bots to report to a valid address. When one bot server is discovered, the botmaster logs into his dynamic DNS account and redirects requests for that bot server to a new bot server at a new IP address.

It is also possible for criminals to install a caching DNS server. All they need is a compromised machine, or a server hosted in a country where phishing laws are lax. The hacker then sends out emails containing a link to the malicious website. When the recipient clicks on the link, DNS servers search for the IP address. Eventually, the rogue DNS server will be 'asked' for the IP address – and when it is, it will provide the IP address of the phishing site.

No silver bullet

Will dynamic DNS and domain management be the conduit for future hacking attacks? It's impossible to know. The cat and mouse relationship between the hacker and the security industry means that criminals are forever trying to conjure up imaginative and unpredictable ways to circumvent the traps laid by security professionals. This makes it very hard for the security industry to be proactive. If criminals can see that preventative measures have been put in place before they have investigated that attack vector, there is little point in exploring that avenue further. Until we discover a 'silver bullet' to stop criminal activity, it is shall remain a persistent irritant in the security professional's life.

One obvious solution would be to create a walled garden of safe web sites. While this solution may mitigate the threat of phishing attacks to some extent, it also negates the liberalism that is inherent to the internet. On the net, anyone can access any information from anywhere. In this country we covet freedom – including freedom of information – but with freedom comes risk and responsibility. If we restricted access to portions of the public internet, it would cease to be the immense resource it currently is.

The internet's diversity is a blessing and a curse. What makes it such a valuable resource also makes it a breeding ground for a variety of online threats. Technicians can continue to refine standards and tools at the network's core, but protection from online crime starts at the edge of the network, with the people using it. Education and preventative action are as important as new DNS technologies.

About the author

Simon Heron has over 16 years experience in the IT industry, including eight years experience in internet security. During this time he has developed and designed technologies ranging from firewalls and anti-virus tools, to LANs and WANs.

Prior to Network Box, Heron co-founded and was technical director of Cresco Technologies, a network design and simulation solution company with customers in the USA, Europe and China. Before that he worked for Microsystems Engineering as a project manager, where he implemented network security for the company. He has an MSc in microprocessor technology and applications, and a BSc in naval architecture and shipbuilding.