DNS in Computer Forensics

Neil Fowler Wright
University of Westminster, London UK.
neil@fowlerwright.co.uk
+44 7717 693 706

ABSTRACT

The Domain Name Service (DNS) is a critical core component of the global Internet and integral to the majority of corporate intranets. It provides resolution services between the human-readable name-based system addresses and the machine operable Internet Protocol (IP) based addresses required for creating network level connections. Whilst structured as a globally dispersed resilient tree data structure, from the Global and Country Code Top Level Domains (gTLD/ccTLD) down to the individual site and system leaf nodes, it is highly resilient although vulnerable to various attacks, exploits and systematic failures.

This paper examines the history along with the rapid growth of DNS up to its current critical status. It then explores the often overlooked value of DNS query data; from packet traces, DNS cache data, and DNS logs, with its use in System Forensics and more frequently in Network Forensics, extrapolating examples and experiments that enhance knowledge.

Continuing on, it details the common attacks that can be used directly against the DNS systems and services, before following on with the malicious uses of DNS in direct system attacks, Distributed Denial of Service (DDoS), traditional Denial of Service (DOS) attacks and malware. It explores both cyber-criminal activities and cyber-warfare based attacks, and also extrapolates from a number of more recent attacks the possible methods for data exfiltration. It explores some of the potential analytical methodologies including; common uses in Intrusion Detection Systems (IDS), as well as infection and activity tracking in malware traffic analysis, and covers some of the associated methods around technology designed to defend against, mitigate, and/or manage these and other risks, plus the effect that ISP and nation states can have by direct manipulation of DNS queries and return traffic.

This paper also investigates potential behavioural analysis and time-lining, which can then be used for the development of automated analysis methods during forensic investigations and as DNS is a network protocol, there is a predomination towards network based attacks and discovery. It shows the breadth of possible attacks and the scope of investigative approaches that can be employed.

Overall it is an exploration of the area of DNS in Computer Forensics, additionally providing a foundation for educational exploration and further subject research: it concludes by bringing together all these aspects to support the

importance of DNS analysis in Computer Forensics.

Keywords: DNS, Network Forensics, FQDN, Computer Forensics

1. OVERVIEW

The Domain Name Service (DNS) is a critical and core component of the global Internet and the majority of corporate intranets. It provides resolution services between human-readable name based addresses, e.g. URL's, and the machine operable IP addresses required for creating network level connections. This report explores the value of DNS query data in System and Network Forensics, as advocated by Gary Kessler, (1) and (2), as well as the malicious use of DNS in system attacks and malware, and some of the associated methods around the technologies to mitigate or manage the risks. As DNS is a network protocol, this report predominates towards network based attacks and discovery. It shows the breadth of possible attacks and the scope of investigative approaches that can be employed.

2. ASSUMPTIONS

It is recommended that the reader has a reasonable knowledge of Internet systems, computer forensics, Botnets and basic networking; an understanding of DNS inner functionality is not required. Detailed information is available from the book, DNS & Bind (3), published by O'Reilly.

3. HISTORY

3.1 Pre-DNS

Internet communication could entirely exist using IP addresses; however general use as both a commercial and personal tool could not be achieved without human-readable names for websites, and online services. The entire name-to-IP resolution system was originally based on a single hosts.txt file maintained by the Stanford Research Institute, (3 p. 3), distributed between systems administrators on the fledgling Internet by manual FTP. Originally infrequently updated; the embryonic Internet outgrew this manual update method, with thousands of records in a single file.

3.2 Foundation

From Paul Mockapetris' original architecture proposal in 1984 (FC's 883 and 884, (4), the DNS architecture grew to provide a distributed, de-centralised, hierarchical system for the name-to-IP resolution on a global basis, allowed the IPv4 Internet to grow (3 p. 4). Top level zones managed by different organisations, repeatedly delegate sections of the name-space. Blocks of domain space can then be readily traversed. Functionality therefore is resilient to missing data or systems at the edge nodes. Figure 1, below, shows an example of part of the top levels of the DNS system, and the subsequent delegation to sub organisations.
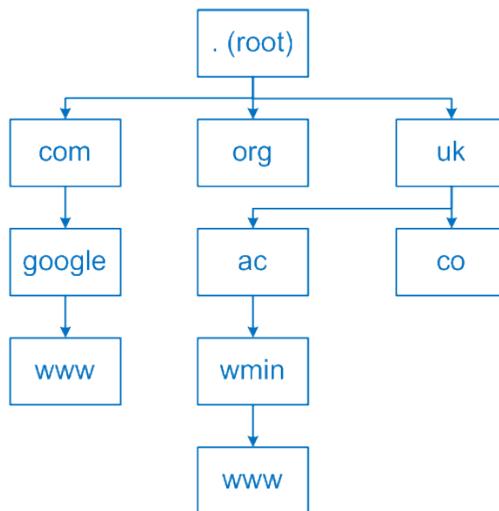
Figure 1

3.3 Expansion

Continued growth of the Internet, now incorporates both IPv6 and IPv4 addressing, with "AAAA" and "IP6.ARPA" records having been registered by NIC's as early as 2004 for the .jp domain, (5 p. 12).

From the top-level ICANN has assigned over 21 gTLD (generic Top Level Domains) zones, (6), and 250 ccTLD's (2-letter country specific top-level domains) zones, (6), on a core set of root-name-servers; large arrays of distributed systems posted at hundreds of key data exchange nodes on the Internet, (7), to individual company systems.

Each countries registrar or NIC (Network Information Centre) is responsible for maintaining or sub-contracting the management of domain name registration and management policies. Different registrars have different rules regarding identity, cost, responsibility, process, etc... E.g. Singapore requires a registration to be made by a company trading within the country and have a local registered company number;

> "A registrant in this category must be, at the point of application, a commercial entity and registered with Accounting and Corporate Regulatory Authority (ACRA), International Enterprise Singapore (IE Singapore) or any professional bodies." (8)

Other jurisdictions have almost no restrictions, and this makes certain domain suffixes particularly attractive to spammers and Botnet masters for use in their illegal services.

ICANN chief, Rod Beckstrom commented regarding the security and continued stability of the domain name system globally; advising certain countries that they would be required to explain themselves for the wholesale criminal activity within their e-borders.

"Parts of that system are only in your countries. I'm going to be writing you a letter and asking you for what is happening in the domain name system in your countries, because we're seeing new levels of wildcarding that's occurring at the telecom service provider level, synthesis of domain name system providers interrupting DNS requests and providing false data and information for commercial or other purposes." (9)

Furthermore the anti-spam group KnujOn reported that 162 of the global registrars were non-compliant with ICANN requirements for whois data, (10).

These comments show an increased level of recognition at the highest echelons for the criticality and vulnerability of the DNS environment, and a willingness to use DNS as part of the control and detection infrastructure for criminal activity.

3.4 Prevalence

DNS is now the de-facto standard for Internet domain name resolution and it comprises millions of physical name servers, and just within the .com gTLD for one delegated registrar, there are almost 90 million active domains with over 315 million names previously deleted, (11). They cite their top registrant as Domaincontrol.com who registered 34,592 domains during 12th August 2010, deleted 23,553, and transferred 22,392 out to other parties, (12). From just one client of one registrar on one day; this is an extreme level of registration volume.

Paul Vixie, President of Internet Systems Consortium Inc, writes that;

"Most new domain names are malicious. ... Every day lots of new names are added to the global DNS, and most of them belong to scammers, spammers, e-criminals, and speculators. The DNS industry has a lot of highly capable and competitive registrars and registries who have made it possible to reserve or create a new name in just seconds, and to create millions of them per day." (13)

KnujOn continues in their full report regarding the prevalence of URLs selling fake pharmaceuticals;

However, illicit product traffic presents an opportunity for Registrars to earn significant amounts money through illicit domain registrations and related domain product services. (14 p. 2)

They continue grouping this seemingly small matter of registrar corruption and whois fraud, with larger more sinister illegal activities;

"Additional security threats like malware deployment, denial of service attacks, trademark hijacking, botnets, spam, WHOIS fraud, network

intrusions, domain hijacking, Registrar corruption, and electronic money laundering are all tools of the global network of illicit drug traffic. Beyond the Internet this traffic impacts the health of the public while funding organized crime and terrorist groups." (14 p. 2)

These examples illustrate the scope of the problem of both crimes which use DNS and the sheer number of domains that any investigation must deal with at any time, and thus the likelihood of DNS related behaviour being relevant to an investigation.

3.5 DNS Service

The core name service software is based on the ISC BIND distribution, which is maintained as *"a reference implementation of those protocols",* (15). Whilst primarily for Un*x and Un*x-like platforms, it is also present in many appliances, e.g. Infoblox, (16), and is a core component of Microsoft's AD server. DNS is therefore present in every domestic, academic, governmental and corporate network, via servers, routers and ADSL modems. The DHCP client software on a PC automatically receives DNS server details when allocated an IP address, (17). For small networks, this is usually an upstream ISP's (Internet Service Provider) DNS cache resolver.

DNS without DNSSEC extensions has no inherent security; it relies on TCP and UDP data connections across IP to the delegated servers and assumes that they are both available and accurate. The protocol does not provide a means to detect DNS interception, and hence packets can be manipulated, as covered later in this paper.

3.6 Overview

As DNS is a core and prevalent component of the Internet, it is both a prime target for attack and a key source of information. The risks and the benefits are therefore from the distributed/delegated nature of the system, and its disparate implementations and standards. This readily allows for local collections of data, and delegated control, but also local influencing of results and sites.

4. DNS IN FORENSICS

Gary Kessler, writes;

"Digital investigators have an increasing need to examine data networks and traffic, either as part of criminal or civil investigations or when responding to information security incidents. To truly understand the contents of the logs and the data packets, examiners need to have a good foundation in the protocols comprising the Transmission Control Protocol (TCP/IP) suite." (2)

4.1 Network Forensics

Whilst it appears that most investigators prioritize their investigation by examining the HTTP traffic for the pages visited, the preliminary step of mapping

the DNS queries and responses and correlating them against subsequent traffic can reveal significant data regarding the activities of the "browser"; human or automated.

4.1.1 DNS in Network Packet Captures

Where complete packet captures are available, the DNS queries can be used to narrow down specific sections of a traffic dump, investigate query timings, or ascertain if HTTP requests are pre-seeded with connection IP addresses e.g. for spoofing attacks, or show failed domain lookups, or those issued as the result of non-human activity.

4.1.2 Data in DNS Queries

For each DNS transaction there is a query and a response. The query contains a single name or IP resolution request in a standard form, easily monitored by Wireshark and a "port 53" filter:

First packet: Request a reverse (in-addr.arpa) resolution of the configured name server e.g. "158.43.204.4", by using the query "PTR 4.204.43.158.in-addr.arpa". If this section is captured, then it demonstrates that the query has been made from the command line, or has not already been cached on the PC by the local resolver. The "nslookup" command skips the local resolver and hosts file cache.

Second packet: Reply from the upstream resolver containing the FQDN (Fully Qualified Domain Name), "cache0004.ns.eu.uu.net". This demonstrates another useful type of information in domain names; the caching name server name reveals structural data about UUNet's operation. In this case they divide their operation by region "eu.uu.net" and then by function "ns.eu.uu.net" and that they utilize multiple classes of name servers "cache", in this case, as well as multiple instances "0004". Whilst not directly relevant to the analysis of a specific PC's behaviour, a sudden change in this data could indicate that the configured resolver has been changed. Also if the IP address in the packet does not match the response, then the DNS connection could be being subverted.

Third packet: Request for the resolution of the provided FQDN, e.g. "www.microsoft.com", as an "A" record.

Fourth packet: Usually the upstream name server then replies with multiple addresses, CNAMES, and their related "A" records. This multiple record response reduces the number of subsequent queries sent by the client PC for the enumeration of a single address.

Fifth packet: Repeats the request but for the IPV6 or "AAAA" query.

Sixth packet: Returns the IPv6 address if available.

Whilst a search for a non-existent record, e.g. "www" with no domain, produces the "No such name" response, and a recommendation that a root-name-server is consulted.

The existence of the IPv6 query shows that the PC has the IPv6 options enabled.

If the PC's DNS configuration is deliberately changed to include a search order of "zzxxxxxxzz.com", "yyyyyyyyy.com" and "zzzzzzzzz.com", Figure 2, then the response to an unqualified domain query is changed as shown in Figure 3.
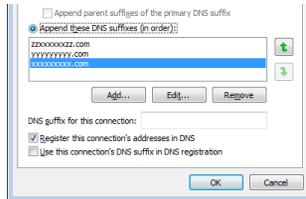

Figure 2

Firstly "www.zzxxxxxxzz.com" is queried and fails, returning "No such name", then "www.yyyyyyyyy.com" is checked and resolves (64.38.232.180). As this has resolved, "www.xxxxxxxx.com" is never checked. This reveals important data regarding the PC's network configuration.

Figure 3

If enumerated over a number of queries all these entries may be identified and the configured search order ascertained.

The behaviour of client DNS queries can be further examined by typing "www.google.com" in to a browser directly, which then generates a number of subsequent queries, Figure 4. Examining these queries and responses reveals how the site was accessed. Clicking on the link "images" at the top of the page then generates no further DNS queries. The suffix ".co.uk" in the reply for the typed suffix ".com" also shows the regionalization of the PC's IP address or its' ISP at that time via the Google load balancing system.

Figure 4

If packet captures are taken external to the local DNS resolver, e.g. on a corporate gateway, then there are many more connections available, and furthermore if both "sides" of the DNS resolver (gateway) are observed, then the difference between client query and connections onwards to other DNS servers would indicate which entries are already in the DNS server cache.

A lack of DNS queries for connections could be IP specific connections, these could be where malware has hardcoded IP addresses for command-and-control servers.

No DNS queries for other HTTP traffic could indicate that the system has already connected to the remote server, and that the IP address is already cached, or that the FQDN is already recorded in the hosts file.

The above examples, only touch the surface of the data that can be extrapolated from the DNS queries, whether the analysis is data based, temporal or another parameter.

4.2 System Forensics

4.2.1 System Hosts File

The hosts file is consulted by the resolver during the first phase of resolution, and where a record exists, will preclude further queries to external systems. Many anti-virus and security applications routinely monitor this file for changes, and several anti-spyware tools pre-seed it with entries containing "127.0.0.x" addresses to foil some malware. Any records in this file should be examined in detail, especially relating to banks or known online systems.

4.2.2 Registry Entry for DNS Settings

For Windows the system registry contains the key, '*HKLM\System\Current ControlSet\Services\Tcpip\Parameters\NameServer="A.B.C.D A.B.E.F"*', (18),

where "A.B.C.D" and "A.B.E.F" are the IP addresses of the allocated name servers. This should be checked to ensure it matches the expected network or DHCP settings.

4.2.3 Resolv.conf

For Linux is the /etc/resolv.conf file holds the DNS search suffixes and the assigned name servers. On a DHCP system, this file may not be fully populated by the DHCP client script, therefore the file /etc/dhcp/resolv.conf needs to be checked.

4.2.4 VPN Tunnels and Addition Network Interfaces

Where a VPN tunnel is set up encompassing a particular network address block, it may be acting as a means to bypass the local ISP or networks name service, and hence change the resolution process. Additional network interfaces can be identified through the key and sub-keys of *"HKLM\System\Current ControlSet\Services\Tcpip\Parameters\Interfaces\ <Interface Name>*", (19)**.** For Linux all network settings are stored in /etc/sysconfig/ network-scripts/<interface name> configuration files.

4.2.5 DNScache Data

The DNScache configuration is held in the registry key, *"HKLM\System\CurrentControl Set\Services\DNScache*". From Windows the DNScache can be flushed to remove polluted cache entries using "ipconfig /flushdns". The contents can be displayed using the command "ipconfig /displaydns" which will list all entries, often many hundreds.

For a real-time analysis of the DNS cache by remotely querying it, Ed Skoudis advises, (20), that you can parse the running cache for specific entries using a predetermined file, names.txt, and a one line command.

> C:\> for /F %i in (names.txt) do @echo %i & nslookup -norecurse %i [DNSserver] | find "answer" & echo.

This is based on the paper by Luis Grangeia, where he explains;

> "The most effective way to snoop a DNS cache is using iterative queries. One asks the cache for a given resource record of any type (A, MX, CNAME, PTR, etc.) setting the RD (recursion desired) bit in the query to zero. If the response is cached the response will be valid, else the cache will reply with information of another server that can better answer our query, or most commonly, send back the root.hints file contents." (21)

Under Linux this would be, (20);

> for i in `cat names.txt`; do host -r $i [nameserver]; done

For a Linux PC with a local caching name server, the DNS cache contents can be dumped using the command "rndc dumpdb -cache" which creates a file called

named_dump.db that can be read or interrogated.

For a system with a local DNS cache, this is extremely useful for DNS query analysis; for DNS servers there is a larger volume of information, but this can be invaluable in tracing malware, and attacks.

4.2.6 DNS server log files and statistics files

On a Linux name server with logging enabled in the DNS service, the extensive file /var/named/chroot/var/named/named.log is created. This file is not rotated on service restart. It contains the time and date stamp and query of every domain lookup received.

Such log files are usually contain too much data to read manually, therefore automated scripts should be used. ISPs maintain very large cache servers for their clients commonly known as "sump" servers, and their cache's and logs can be used to produce a detailed picture of browsing and malware behaviour from their client base. These servers are almost never rebooted.

System log files, can also contain information regarding zone transfers. Where it is suspected that a transfer has been polluted, and provide a record of the attack. Similarly Windows System and Security logs may contain critical data on changes made to the DNS environment.

4.3 Attacks on DNS

There are many attacks on DNS, some included below, for which detailed explanations can be found on various technical websites, including detailed analysis of Pharming methods by Gunter Ollmanns' www.technicalinfo.net, (22). By studying the attacks it is possible to develop detection and investigation routines as demonstrated by the Conficker and Zeus analysis. Studies in Essen, (23), have shown that publishing DNS records is one of the most effective way of attracting malware and attacks to a honeynet for analysis.

4.3.1 Modified Lookup Process

By changing the contents of the hosts file, or by subverting the resolver DLL files, or by changing the DNS server settings on the network interface, or by creating a VPN tunnel with different VPN settings, the client PC can be caused to access alternative IP addresses for specific websites. This is usually detected by detailed analysis of the system configuration, and checking the file contents and/or hashes for files and DLL's or a packet trace.

Modifying the contents of the DHCP server, or inserting a rogue DHCP server/service in to the LAN allows the published DNS settings to be modified. This then allows arbitrary modification of DNS responses. Reviewing the contents of DHCP packets in network captures, and last connection entries in the registry on the system can be compared with expected settings.

4.3.2 Man-in-the-middle Attacks and DNS Spoofing

Compromising routers, or network proxy servers, allows the DNS connections crossing the system to be modified, and the resultant connections to be redirected. Redirection of connections could be made to upstream or transparent proxy servers where man-in-the-middle attacks can additionally be made. Reviewing network security, network architecture, and web proxy settings, can reveal this form of attack. However analysis and comparison of external and internal DNS resolution requests, and identifying data mismatches can also reveal some of the aspects of these attacks.

4.3.3 Domain Name Hijacking

At the domain registrar level, the compromising of an email account, or administrative key (auth-key) can allow the delegation records for a domain to be redirected to alternative domain name servers, where the browser can be redirected to an alternative web site. Ramachandran used domain hijacking techniques on domains used by Bobax in 2005 to identify hosts sending spam trying to communicate back, (24). Detection of this attack is very difficult at the network level, and has to be analyzed using delegation and registrar tools, e.g. "dig" and "whois" as well as online search engines.

Botnets often utilize thousands of domains, where the DNS delegation can be updated automatically, and these are then frequently changed to the current active set of zombie DNS servers, which each hold the current low TTL value zone records. c.f. Dynamic DNS and Fast Flux DNS below.

4.3.4 Similar Domains Names

Another common attack, exploiting the human weaknesses of spelling and typing and for use in phishing attacks and link attacks; an attacker registers a number of similar but misspelt zones and delegates them to name servers they control. Browsing to these addresses reaches the rogue website, which often proxies the real site, or uses the real sites embedded objects. Where the misspelt domain name is used in email phishing attacks, it can be detected by the consistent and repeated accesses to this misspelt site in DNS.

4.3.5 DNS Wildcards

Where the "*" field is used in a DNS zone file, it acts as a catchall for possible host records. These are not commonly used in normal DNS outside MX (Mail eXchanger) records for email processing, and hence were they are used for tracking purposes e.g. by Phishers, they are clearly visible in DNS logs and network captures, and useful for 4.11.4 Mobile Data Exfiltration.

4.3.6 Poorly Managed DNS

A common point for exploiting DNS is to use many of the known vulnerabilities in the DNS service daemons. Where DNS servers are not patched, or zero-day exploits can be utilized, the configuration of the DNS server can be changed and

traffic redirected. Variations on this attack include "zone file injection", where a compromised zone file is added, and "root hints file changes", where the file containing the details of the core root-name-servers are altered. Unless the compromised DNS server is checked in detail, or the compromising traffic is detected, this can be very difficult to detect. Comparison of direct DNS queries and DNS delegation checks will though reveal the discrepancies against the compromised server for the specific domains affected.

4.3.7 DNS Server Denial of Service

Denial of Service (DoS) on a DNS server, includes taking the server or the service offline, either through a direct crash, or by flooding the server with bogus connections. This will cause a redirect for all the clients to their secondary name server and slow down the client PC, but may allow the attacker to redirect the connections to a compromised secondary DNS server. Detection of DoS attacks can be difficult, and research is examining the effectiveness of Neural Networks, (25). Examining the search order on the PC and comparing to captured network traffic, or examining the failed outbound connections to the primary server will identify this scenario.

4.3.8 DNS Cache Poisoning

Where the DNS server is vulnerable to any of the available cache security vulnerabilities, an attacker can inject a fake DNS record in to the existing cache file. This is then propagated through any DNS servers downstream that use this compromised host as a cache-resolver. Where Fast-Flux DNS is used with a Botnet, then the server may be repeatedly attacked with low TTL value records to pollute the cache. Where a record is intended to be cached for an extended period, e.g. Denial of service via cache poisoning, then a large TTL will be provided. This is particularly difficult to resolve, as the DNS caching server will only clear the record on restart, or when the TTL expires. A scan of 2.5 million DNS servers by Dan Kaminsky in 2005, (26), showed that more than 230,000 servers were vulnerable to just one type of cache poisoning threat.
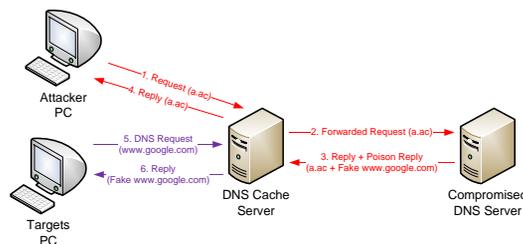

Figure 5

Figure 5, above, shows the data flows of a cache-poisoning attack. Where the attacker first requests a valid DNS name (1), and the cache server queries that from a compromised but authoritative name server (2), the reply (3) contains both

the requested data, and an additional record (or records) to poison the cache. The original data is returned to the requester (4). When a future client requests the poisoned record (5) they get the corrupted data (6) which is already in the cache.

In 1997 Eugene Kashpureff may have used an early form of this attack against large numbers of DNS servers as an apparent marketing ploy or protest attack against Internic, redirecting all traffic to his Alternic site. He faced federal charges, (27), before pleading guilty, and received a 5 year suspended sentence, (28).

To investigate these attacks it is possible to manually follow the DNS delegation path to ensure that the correct DNS records can be established. When monitoring replies from suspected poisoned caches, compare current published TTL's with those in the cache server. Very high and very low TTL values should be suspicious, where most DNS records are 900 to 7200 seconds, occasionally as high as 172800 seconds for some records, and occasionally as low as 15 seconds for some load-balanced services. Other values in packet captures or replies should be examined closely. Cached entries may remain for hours/days after an attack has been mitigated.

4.3.9 DNS ID Spoofing

DNS ID spoofing is where the attacker tries to predict the DNS transaction ID for a given query that will be sent outbound from the cache server. The attacker then floods the server with spoofed reply UDP packets in the hope that the ID number is matched and a spoofed packet is accepted before the real reply is received. If the attacker has access to the local network, then the ID can be sniffed from the query packet and hence immediately spoofed.

In mid 2008 Dan Kaminsky advised of a new flaw present in nearly all current name servers across the world, (29). Patches were swiftly released and automated tools, Figure 6, are available to verify currently exposure.
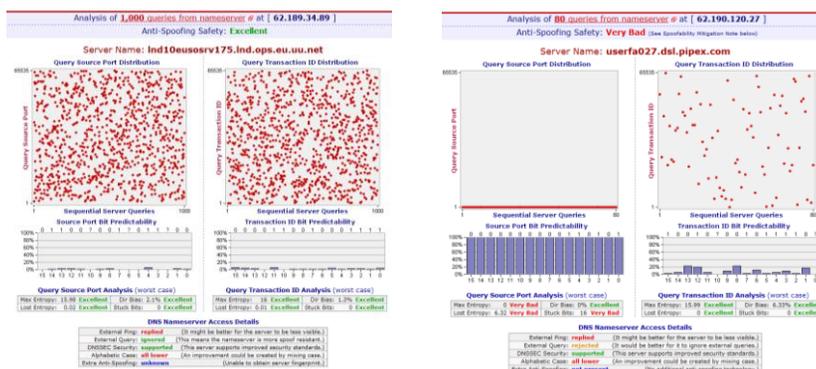

Figure 6

The left-hand imageFigure 6 clearly shows a well configured server whose port numbers and DNS ID's cannot readily be predicted. However the right-hand image**Error! Reference source not found.** shows data for a server which is at substantial risk, in this case due to the use of the stanza "***query-source address * port 53;***" in the /var/named/chroot/etc/named.conf file. This attack is an additional way to create poisoned cache records, andcan be detected in the cache by the same mechanisms as above, however this attack is more readily detected at the network level by monitoring for these spoofed replies.

4.3.10 Birthday Attack

The effectiveness of the above attack can be improved, as detailed by Ollmann;

> "This repetitive behaviour means that a "Birthday Paradox" could be used to mathematically increase the speed and probability of a successful attack by reducing the number of spoofed guesses of the DNS transaction ID from tens of thousands down to a few hundred." (22)

4.4 Identifying Behaviour

As outlined in the Network Analysis above and in the attacks on DNS directly, the data at the packet level is critical to identifying behaviour.

DNS queries can be used to narrow down a search to a specific section of a network traffic dump, or to investigate query timings, or to ascertain if HTTP requests are pre-seeded with connection IP addresses e.g. for spoofing attacks, or to indicate failed domain lookups, or those issued as the result of non-human activity, or as a passive system consequence, or tracking the behaviour of software update tools. The DNS queries made by corporate PC's when they are off-network can reveal internal domain names, IP's and addresses of management, reporting and logging servers.

However when investigating DNS traffic, it is best to take a larger view with an understanding of how and when browsers, for instance, cause the PC operating system to make a DNS request.

4.4.1 Timeline

For example; taking all the DNS replies from a 24 hour period in Wireshark, saving them out as text based packets, and filtering for the reply header, domain name and the time to live. Then filtering with a simple Perl script and returning the sorted data to Excel for graphical processing. Figure 7 shows the offset TTL versus domain name in request order, with the domain names text suppressed at this level.
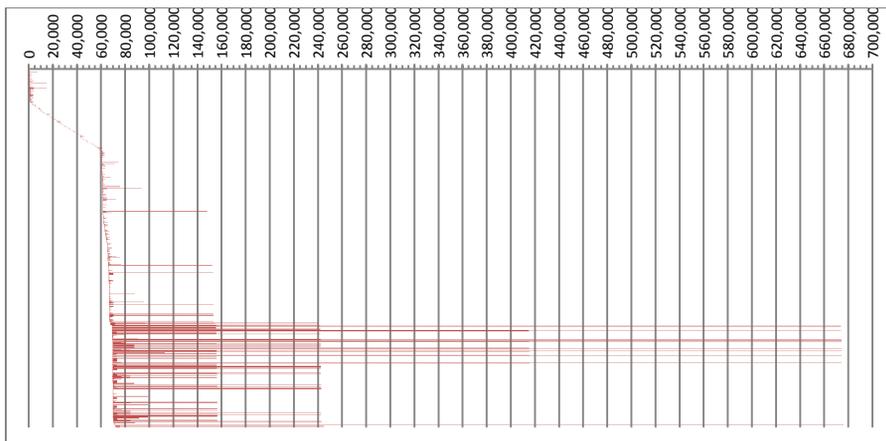
Figure 7

This high-level view contains 2000 line items of data, and could readily be "drilled-down" to specific times or filtered by URLs. It is already apparent that there are different TTL's on different domain names, and that some low-level DNS activity is happening continuously (slopes in graph). Periods of peak activity can also be identified. This analysis should form part of any investigation where the data is available.

4.5 Tracking Malware

The key to detection is monitoring the DNS server logs to spot Malware events, which for investigative purposes correlates with log reviews and traffic captures.

4.5.1 Hard-Coded IPs

Looking for HTTP or other transactions which are NOT preceded by DNS requests can be a key sign of malware activity, (30). Malware is often pre-seeded with connection targets and peer-to-peer nodes for which DNS lookups are not required.

4.5.2 Dynamic DNS (DDNS)

A particular feature of many of the DNS aware Botnets is the use of Dynamic DNS records. Here either the DNS is directly manipulated within the Botnet, via the hosting of delegated DNS servers on zombie hosts, or though third party DDNS servers. This allows a malware site to frequently move location. Given the usual time-lag between seizure and investigation, the DNS logs and network captures are the only place to identify this changing data.

4.5.3 Fast Flux DNS

An extension of DDNS is the use of Fast Flux methods; here the DNS records are changed on every transaction by setting a TTL of 0 or 1 seconds and hence requiring the client to re-lookup the remote address. This is commonly used now

in Botnets to continually move services around the Botnet cloud. Such regularly changing DNS records should be seen as a clear sign in any packet capture or log of an active Botnet on the network. Even load balanced critical infrastructure does not regularly change its' IP addresses in this way.

4.5.4 Conficker

ICANN recently published a report, (31), detailing how the DNS traffic generated by the Conficker worm was tracked and investigated. This represents a significant turning point in counter-worm activities and network forensic analysis, even when we consider most modern Botnets have moved to P2P communications, this change is largely as a result of this and similar investigative and reactive activities.

In summary, the global registrars, law enforcement, ICANN and other parties came together to lock down, pre-register and sinkhole as many as possible of the hundreds of algorithmically generated DNS names being used by the Conficker zombie processes to connect to the command and control infrastructure, effectively cutting them off from the main Botnet Command and Control servers. This use of DNS by the malware authors was to mitigate the risk of network (IP) level takedown of Command and Control systems where they would otherwise be hardcoded in to the malware. These DNS records should be considered as "Rendezvous Logic Points", and were typically registered just a few hours before use for the Botnet malware.

As a result of the effectiveness of this campaign spanning just the core TLD's, was a broadening of the DNS name scope in subsequent variants of the malware. This required a similar broadening by the ICANN team in their contacts with ccTLD and gTLD registrars. Piscitello summarizes below the criticality of DNS in the forensic effort against the malware:

> "The combined efforts of all parties involved in the collaborative response should be measured by more criteria than mitigation alone. The containment measures did not eradicate the worm or dismantle the botnet entirely. Still, the coordinated operational response merits attention because the measures disrupted botnet command and control communications and caused Conficker malware writers to change their behaviour. The collaborative effort also demonstrated that security communities are willing and able to join forces in response to incidents that threaten the security and stability of the DNS and domain registration systems on a global scale." (31)

4.5.5 Cutwail, Ozdok and Zeus

First looking at the correlation of Spam and sending domains using DNS correlation and whois searches, Chun Wei and colleagues found a strong direct correlation, (32). Willa Ehrlich from AT&T Labs later published a paper on their investigation of SMTP spamming malware using network flow data and DNS metadata, (33). By profiling legitimate SMTP traffic against traffic generated by a

spamming Botnet, and using entropy based analysis of hosts behaviour to identify zombie systems. Then for these identified systems, the command and control connections were enumerated using traffic modelling, periodicity and recurrence analysis. Once suspected, the DNS metadata was passively replicated and then analyzed for suspected abuse of the DNS protocol. This was achieved across the AT&T Internet backbone. Ehrlich describes this analysis as follows (reformatted for clarity):

> "The DNS analysis of the IP address of the suspected controller provides the following output:
>
>   i.   *count of all domains resolved to the address historically,*
>   ii.   *count of domains that resolved to the address recently (e.g., last 1 day),*
>   iii. *number of transient domains related to the suspect address. Transient domains are domains that migrate frequently between diverse provider addresses, indicating an evasion effort.*
>
> To determine the transiency of a domain we consider the average time overlap between addresses for the same domain and the diversity of the addresses in terms of AS numbers and IP registration data." (33)

This is perhaps one of the clearest and most concise examples of DNS forensics against real-time network packet captures, and should be used as a model going forwards for other similar analysis. However even the most basic spam analysis involves DNS analysis of email headers, (34).

4.5.6 DNSscraper

On a smaller scale, Ben Jackson created, published and updated a program called "zeusdnsscraper.pl", (35). This simple system uses a non-recursive DNS query, discussed earlier, to query the domain name server for entries in the Zeus Tracker Domain Block List, (36). Any hits could then be investigated back to client PC's to contain any infection of the Botnet on the network. This can of course be expanded to any other Botnet or domain required.

4.5.7 Logs, Caches and Trends

As described in detail above, 4.3 Attacks on DNSthere are many known subversion methods for DNS data payloads. Actively looking for these trends and connection signatures, e.g. with an IDS system, greatly increases the chance of detecting and mitigating malware infections, or detecting malware activity.

Some host based IDS and firewall software contains firewall rule-bases or packet level transaction logs, which can be similarly used. Mapping DNS query packets against known TTL's and subsequent data connections will also highlight relevant issues.

4.6 Trojan Defence

More directly, when faced with a proposed "Trojan Defence", the data in DNS logs, records, caches and hosts files may prove invaluable in proving or disproving the underlying hypothesis. The signatures for malware are well known and can be readily enumerated if the data exists.

4.7 Penetration Testing

The tools used for penetration testing may be useful in analysis, or may be used as part of an attack being investigated. Below, examined briefly, are a small number of interesting DNS related testing tools and methods:

4.7.1 DNS Mapping

The DNSMAP tool, (37), is a prime example of a tool that forms the basis of a penetration test, allowing the wholesale mapping of DNS zones, whois data and other domain related information. This tool includes sub-domain brute-forcing for the enumeration of hosts and sub-domain records from a name server. The importance of whois and methodologies for investigation are covered by Bruce Nikkel, UBS, (38).

4.7.2 Fierce 2.0

Fierce, (39), is a DNS mapping tool for aggressively enumerating, DNS zones via the name server and many other databases. It is capable of enumerating AS numbers, ARIN tables, whois data as well as standard DNS zones.

4.7.3 Information Discovery

Matt via his Attack Vector blog, (40), published a discussion of corporate information discovery methods which underpin the two above tools. This shows how readily information can leak and the importance of the DNS and DNS-related databases.

4.7.4 Information Leakage

Further information on the leakages of possible internal IP space through external corporate DNS servers is presented by dd, (41), giving examples of IP addresses extracted from DNS names inside Dell, Cisco, Facebook and even the SANS Institute.

4.7.5 Future Mapping

As IPv6 substantially increases the IP space available and is deployed inside a corporate network, brute forcing DNS domains in IPv6 will replace ping sweeps in future attacks and penetration tests. All networks will need to be in internal private DNS to function, so attacking the DNS zones and enumerating them becomes the best option. Van Hauser writes;

> "DNS Servers will become primary sources of information => primary targets", (42)

4.8 DNS in Attacks

In addition to those 4.3 Attacks on DNS explored above, there are a number of uses of DNS in other attacks presented below:

4.8.1 DNS Rebinding

According to Jeremiah Grossman, (43), 4 of the top 15 new types of attack developed in 2009 used DNS rebinding in some way. DNS Rebinding, (44), uses XSS (Cross-Site Scripting) and DNS vulnerabilities to pollute DNS data in the client resolver. By getting the user to access a compromised site, which provides a fake cookie for the intended target site, then rebinding the DNS record of the compromised site to that of the target site, the browser can be forced to access the target site with the XSS shell code from the compromised site. This can be used for spamming and scraping attacks on websites, (45). RSnake also explains in detail how a session fixation attack can be carried out using session cookies, compromised DNS servers and very low TTL's, (46), and then extends this further to credential brute force attacks, (47).

This shows the extensive scope for leveraging DNS rebinding, in this case, to enable other forms of attack, e.g. against the admin interface of domestic ADSL routers, (48). However, detecting this attack can be difficult without looking at both DNS data and web session cookies, or working closely with the site address and the host header referenced against DNS. Defeating it however, is achieved through the use of SSL and TLS connections as proposed by David Ross, (49).

4.8.2 Attacking Google

Google, who recently suffered a very high profile attack, presented details of their investigation to the FIRST (Forum of Incident Response and Security Teams) Conference 2010 in Miami. Heather Adkins revealed;

> "The security team also found the use of a hard coded DNS server. When the attackers conducted reconnaissance, they performed DNS queries, data which was useful in the investigation, Adkins said.
>
> The team searched for hosts and the DNS queries, building a picture of the scope of the attack. Concentrated DNS queries in a specific place represented invasive operations, she said. Most traffic mainly reaches out to common sites. A warning sign is when traffic is detected going to a new website that was recently registered and no one had visited before.
>
> "DNS query logs may be the only method you have to find new generations of malware," she said. "The adversary will need to reach other systems to install that malware. We often look for the big [anomalies], but we have to monitor for the subtle too." "(50).

This clearly demonstrates the criticality of DNS logs and registrant data when investigating an attack.

4.9 ISP DNS Manipulation

Whilst technically not an attack; it is important to ensure during any investigation that the role of the ISP and their DNS servers is fully understood. There are many examples where ISP's manipulate DNS, effectively using man-in-the-middle and wildcarding techniques to achieve specific aims, usually advertising.

4.9.1 Advertising Injection

An area of contention exists with the manipulation of DNS replies for non-existent domains, to display advertising pages. This came to a head when BT was accused of illegal monitoring of users communications with the Phorm trial, (51). Whilst advertising injection remains a questionable practice, it is entirely facilitated by DNS manipulation on the ISP's name servers.

4.9.2 YouTube Censorship

Many countries insist on filtering the Internet due to political requirements, censorship directives, or differences in legal statutes. Turkey for example heavily censors approximately 4000 websites, including access to YouTube and recently added Google to the list, (52). Manipulating client DNS settings can be used in some jurisdictions to circumvent this control as explained by Ozan Tuzun in Jonathan Heads article, (52).

4.9.3 Great Firewall of China.

Part of the government censorship system in China for all Internet traffic, it filters DNS and controls how sites like Facebook or Twitter are resolved. An "attack" need not therefore be malicious or deliberately directed; sections of DNS can be impacted by issues with root-name-servers where such censorship is applied to traffic from or for countries outside of the censorship zone. For example Chile and parts of the US suffered a loss of several sites due to such an issue on an independent root-name-server hosted in China, (53).

4.10 Proactive Methods

A number of technologies are now emerging which will change the nature of DNS, increasing security and reliability with the aim to reduce the attack and exploit surface for cyber-criminals and protect the end-user.

4.10.1 DNSSEC

DNSSEC Validator is an alpha-release plug-in to Firefox that allows the browser to display if a domain is secured by DNSSEC and whether those checks were successful, (54). When a page is browsed using this plug-in, the user can check to verify the page has not been spoofed. Zones can readily be secured using online tools, e.g. security-dns.net (55), and implementations of OpenDNSSec, (56). Care must be taken with system and key management however and ensuring keys are not allowed to expire or further issues can arise, as concluded by George Michaelson and colleagues, (57), in their article for the Internet Protocol Journal;

"It is an inherent quality of the DNSSEC deployment that in seeking to prevent lies, an aspect of the stability of the DNS has been weakened. When a client falls out of synchronization with the current key state of DNSSEC, it will mistake the current truth for an attempt to insert a lie. The subsequent efforts of the client to perform a rapid search for what it believes to be a truthful response could reasonably be construed as a legitimate response, if indeed this instance was an attack on that particular client. Indeed, to do otherwise would be to permit the DNS to remain an untrustable source of information. However, in this situation of slippage of synchronized key state between client and server, the effect is both local failure and the generation of excess load on external servers—and if this situation is allowed to become a common state, it has the potential to broaden the failure state to a more general DNS service failure through load saturation of critical DNS servers." (57)

4.10.2 Sump Caching

As could be seen from the previously reviewed Conficker defence paper, (31), pre-seeding or proactively seeding DNS zones can greatly reduce the effectiveness of malware. Using key sump-caches and deliberately injecting such records on a global basis is another effective way of debilitating malware. It has to be achieved on a global scale, which is where organisations like DNS-CERT and ICANN have a potentially key coordination role.

4.10.3 Sinkhole

A DNS Sinkhole, as released recently by the SANS ISC (Internet Storm Center), is an example of a preconfigured DNS sump-cache-resolver, Bruneau explains;

"It is a ready to install DNS Sinkhole server for those who would like to test and/or deploy one in their network as an internal forwarder. I also indicated that inserting a DNS sinkhole in a network is like putting a NIDS/NIPS inline with potentially several thousand signatures (DNS domains). After you loaded your DNS sinkhole list, it hijacks the client's DNS requests to known malicious sites responding with an IP address you control instead of its true address. It could also be used to enforce corporate policies (hacking, adults, gaming, social, etc) with the creation of separate sinkhole lists." (58)

allowing any corporate network to deploy such a device trivially. The use of such sinkholes has been widely discussed over several years and used extensively against Botnets, (59) , but this is a significant development in operability and potential of this technology.

4.10.4 DNSBL

DNSBL's (DNS Black Lists) are used by anti-spam engines to monitor, track and publish in a resolvable manner IP addresses and domain names for use by the email filters. They are RBL's (Real-time Black Lists) published via DNS to

reduce the size of the databases downloaded by subscribers and improve the response time. Any host appearing on a block list is then ignored by a subscribing SMTP mail server, and cannot exchange suspected SPAM email. Jeff Makey compares and contrasts different blacklists and their effectiveness, (60).

Anirudh Ramachandran explains how the combination of honey-trap, sinkhole, and DNSBL could be used to profile the network-level behaviour of spammers, (24).

4.11 Cutting Edge

4.11.1 DNS-CERT

The attacks on global or corporate DNS infrastructure are now considered so serious that there have been active discussions about creating an emergency alert and reaction network around the global DNS infrastructure, (61). Paul Vixie, recently proposed the creation of a DNS-CERT (Community Emergency Response Team), (62), to create just such a global cooperative of DNS NOC's to monitor and react to threats to the DNS landscape. This has received wide spread support, and is expected to be closely related to an existing DNS-OARC framework and models. He proposes the following model;

> 'DNSCERT should be a joint venture across the entire DNS industry, and the 24x7 "watch floor" should be distributed across the globe. Much of the technical and operations work should be outsourced to the participants, who by running a tool set in common and doing training in common including sending personnel to DNSCERT HQ on a quarterly or annual rotation, will form an extremely robust and redundant asset base for the DNSCERT function.' (63)

ICANN have responded with findings from their 2010 DNS-CERT workshop, to directly address current issues;

> "The purpose of the exercise was to discuss various scenarios, and identify how the members of the community around the table presently respond to an array of Internet and DNS security threats, such as malicious names, malware using DNS, denial of service attacks, and security issues involving domain names, registry and registrar operations." (64)

This shows that the industry takes these matters very seriously and is using the expertise and techniques, many of which are outlined above, not just to investigate issues, but to improve the security and stability of this critical system.

4.11.2 RPZ

At the Black Hat Conference July 2010 Paul Vixie presented DNS-RPZ, (13), also known as Response Policy Zones, where owners of DNS resolvers can subscribe to and exchange reputation information regarding DNS resolvers and their domains. This was expected to be incorporated in to the mainstream BIND code in the October 2010. This proposal has already generated much discussion,

and whilst embryonic, it has the potential to allow greater classification of criminal activity, and low-reputation behaviour. If incorporated into browser software, this would reduce some phishing sites effectiveness in a manner similar to DNSSEC.

As with any such additional control, the technology moves the cybercrime problem from one area to another. Whilst this is beneficial to many large organisations, it does not help the community environment of the Internet where these reputational systems would be ill-equipped to cope with such small user communities. This would however be effective against the short-lived domains prevalent in Botnets, but likely be over-ridden by the impact on less well known, private user systems.

4.11.3 Client IP

As part of the extensions proposed by Google and UltraDNS to the existing DNS protocol, they would like to see the clients IP address passed as part of the DNS request. Ostensibly for DNS load balancing, (65), if implemented this would provide further data within the protocol for the positive identification of the client PC and protect against a number of the flood attacks, permitting client connection verification and tracking. How this would function with the largely NAT-ted network environments of most organisations and ADSL networks has not yet been disclosed. It would require a fundamental change to DNS masquerading techniques to avoid the default 192.168.1.2 domestic PC address being seen by the global DNS servers.

4.11.4 Mobile Data Exfiltration

Chris Wysopal, CTO of Veracode, presenting at the 2010 SCForum, (66), explained how mobile devices, which are fundamentally PC's in a small case, use IP connectivity and hence DNS requests for much of their non-call based functionality. With the growth of Android, Windows Mobile, Blackberry and other advanced mobile platforms, the market has grown to a ubiquitous Internet delivered environment.

Where device-based malware may be harvesting data, phone numbers for example, their outbound connectivity may be closely controlled by access control structures inside the mobile OS and additional security software. In this scenario, DNS is usually an allowed protocol; therefore the data being exfiltrated by these malware programs can be encapsulated in the DNS packet space, or as part of the query. Where the malware author controls their own DNS servers, this becomes trivial to implement.

This does not just affect the mobile market, DNS exfiltration appears to be a superb way of bypassing most controls. Where wildcard DNS zones are created and malware hosting DNS servers run by the malware authors, the potential for data loss is unlimited. Where "space" in the protocol is used, this may be limited by deep packet inspecting firewalls and IDS, but hiding this data in the DNS

resolution request would be trivial and largely hidden in the huge global volume of DNS traffic. E.g. "*Standard query A gold1230.miscreant.ac*", could communicate the current $1230 futures price for gold to a name server hosting "miscreant.ac" under the Ascension Islands ccTLD.

This closely matches the communications channels used by previous malware such as Loki, as discussed by Hal Bergal, (67).

4.11.5 Malware Hash Register (MHR)

Team Cymru have proposed the publication and maintenance of a free for use non-commercial licensed, software hash database for use by forensic examiners, and to publish this via a query based DNS system.

> "The Malware Hash Registry (MHR) project is a look-up service similar to the Team Cymru IP address to ASN mapping project. This project differs however, in that you can query our service for a computed MD5 or SHA-1 hash of a file and, if it is malware and we know about it, we return the last time we've seen it along with an approximate anti-virus detection percentage."(68)

This shows an innovative use of DNS for the management and manipulation of hash data for use in forensic analysis. Whilst greatly expanding the size of DNS, this shows the power, and through that, the risk of such a global database structure.

5. CONCLUSION

This report clearly shows that the DNS protocol is both critical and vulnerable. It is at the core of all global networks and is the one function that maps the human-readable 'face' of the internet to the complex machine-operable 'IP' world of routers and servers. It is allowed through almost all firewalls globally and is generally implicitly considered to be trustworthy by end-use systems and hence the end-users. There are few people within the technical arena who truly understand both the DNS protocol and the above detailed risks. Many such parties, Kaminsky, Vixie, RSnake, Morrow, Arends, Liu, and the team at Columbia Tech, etc are repeatedly cited in this report or in the other articles referenced.

The vulnerability of DNS as a whole and end-user systems specifically has been clearly shown, and furthermore expanded to demonstrate some of the techniques and practices that enable forensic investigations to be carried out, within, using, and on the DNS infrastructure.

DNS logs, where captured, have value in confirming browsing behaviour versus malware behaviour, identifying system configuration, as well as providing time-line data for investigations. DNS is therefore an immensely versatile and valuable resource.

It should be noted that many of the techniques and investigations listed above are

shared between system/network forensics, intrusion detection/prevention, proactive system monitoring, system/network management, and procedural controls. This should not be seen as a dilution of the information, but instead reinforce the criticality of DNS data and the integral part it plays in all communication.

REFERENCES

1. *The Case for Teaching Network Protocols to computer Forensic Examiners.* **Kessler, Gary C. and Fasulo, Matt.** Arlington : s.n., 18-20 04 2007, Proceedings of the Conference on Digital Forenscs, Security and Law, pp. 115-137.

2. *On teaching TCP/IP protocol analysis to computer forensics examiners.* **Kessler, Gary C.** 2(1), 2008, Journal of Digital Forensic Practice, pp. 43-53.

3. **Liu, Cricket and Albitz, Paul.** *DNS & Bind.* 5th. Sebastapol : O'Reilly Media, Inc, 2006. p. 618. 978-0-596-10057-5.

4. **IETF.** RFC 1034 - Domain Names - Concepts and Facilities. *Internet Engineering Task Force.* [Online] 11 1987. http://tools.ietf.org/html/rfc1034.

5. **Yasuhiro, Morishita Orange.** DNS Operational Experiences in JPRS/.JP - IPv6. *Japan Registry Services Co., Ltd.* [Online] 22 02 2005. [Cited: 12 08 2010.] http://www.nav6tf.org/documents/arin-nav6tf-apr05/4.IPv6_and_DNS_BM.pdf.

6. **ICANN.** New GTLD Program. *Internet Corporation for Assigned Names and Numbers.* [Online] 10 2009. http://www.icann.org/en/topics/new-gtlds/factsheet-new-gtld-program-oct09-en.pdf.

7. **Davies, Kim.** There are not 13 root servers. *ICANN Blog.* [Online] 15 11 2007. http://blog.icann.org/2007/11/there-are-not-13-root-servers/.

8. **SGNIC.** SGNIC RPPG. *SGNIC.* [Online] 11 2009. [Cited: 18 08 2010.] http://www.nic.sg/sites/default/files/rppg.pdf.

9. **Murphy, Kevin.** Beckdtrom: DNS is under attack. *DomainIncite.com - Domain Name News & Opinion.* [Online] 03 2010. http://domainincite.com/beckstrom-dns-is-under-attack/#more-316.

10. **KnujOn.com LLC.** News. *KnujOn.com.* [Online] 01 07 2010. [Cited: 12 08 2010.] http://www.knujon.com/news.html.

11. **Domain Tools.** Domain Counts & Internet Statistics. *Domain Tools.* [Online] 12 08 2010. http://www.domaintools.com/internet-statistics/.

12. —. Daily Changes by DomainTools, Whois by DomainTools.com. *Daily Changes.* [Online] 12 08 2010. [Cited: 12 08 2010.] http://www.dailychanges.com/.

13. **Vixie, Paul.** Taking Back the DNS. *CircleID - Internet Infrastructure.* [Online] 30 07 2010b. [Cited: 06 08 2010.] http://www.circleid.com/posts/20100728_taking_back_the_dns/.

14. **KnujOn.com LLC.** *KnujOn.com* . [Online] 20 06 2010. [Cited: 12 08 2010.] http://www.knujon.com/knujon_audit0610.pdf.

15. **ISC.** Bind 9.7.1-P2. *Internet Systems Corporation.* [Online] 15 07 2010. [Cited: 03 08 2010.] https://www.isc.org/software/bind.

16. **Infoblox.** Network Service Appliances. *Infoblox.* [Online] 2010. [Cited: 03 08 2010.] http://www.infoblox.com/products/appliances.cfm.

17. **IETF.** RFC 2131 - Dynamic Host Configuration Protocol. *Internet Engineering Task Force.* [Online] 03 1997. http://tools.ietf.org/html/rfc2131.

18. **Savill, John.** Where in the registry are the entries for the DNS servers located? *Windows IT Pro.* [Online] 09 01 2000. [Cited: 12 08 2010.] http://www.windowsitpro.com/article/dns/where-in-the-registry-are-the-entries-for-the-dns-servers-located-.aspx.

19. **Honeycutt, Jerry.** Microsoft Windows Registry Guide. *Microsoft Windows Registry Guide.* 2nd. Redmond : Microsoft PRess, 2005, Appendix B, pp. 467-483.

20. **Skoudis, Ed.** Episode #17: DNS Cache Snooping in a Single Command. *Command Line Kung Fu.* [Online] 30 03 2009. [Cited: 14 08 2010.] http://blog.commandlinekungfu.com/2009/03/episode-17-dns-cache-snooping-in-single.html.

21. **Grangeia, Luis.** DNS Cache Snooping. *rootsecure.net.* [Online] 02 2004. [Cited: 14 08 2010.] http://www.rootsecure.net/content/downloads/pdf/dns_cache_snooping.pdf

22. **Ollmann, Gunter.** The Pharming Guide (Part 2). *Technical Info.* [Online] 2007. [Cited: 13 08 2010.   ] http://www.technicalinfo.net/papers/Pharming2.html.

23. *Efficient Deployment of Honeynets for Statitical and Forensic Analysis of Attacks from the nternet.* **Riebach, Stephan, Rathgeb, Erwin P. and Toedtmann, Birger.** [ed.] R. et al. Boutaba. s.l. : IFIP International Federation for Information Processing, 2005, Networking, pp. 756-767.

24. *Understanding the Network-Level Behaviour of Spammers.* **Ramachandran, Anirudh and Feamster, Nick.** Pisa : s.n., 11-15 09 2006, SIGCOMM, pp. 291-302. 1-59593-308-5/06/0009.

25. *Detection of Denial of Service Attacks against Domain Name System UsingNeural Networks.* **Rastegari, Samaneh, Saripan, M. Iqbal and Rasid, Mohd Fadlee A.** 1, 2009, IJCSI International Journal of Cpomputer Science Issues, Vol. 6, pp. 23-27. 1694-0814.

26. **Evers, Joris.** DNS servers--an Internet Achilles' heel. *Cnet News.* [Online] 03 08 2005. [Cited: 06 08 2010.] http://news.cnet.com/DNS-servers--an-Internet-Achilles-heel/2100-7349_3-5816061.html.

27. **Kornblum, Janet.** Kashpureff to face federal charges. *CNET News.* [Online] 3 11 1997. [Cited: 13 08 2010.] http://news.cnet.com/2100-1023-204961.html.

28. **Long, Peggy and Valiquette, Joseph.** Eugene E. Kashpureff Pleaded Guilty to Unleashing Software on the Internet That Interrupted Service for Tens of Thousands of Internet Users Worldwide. *Department of Justice.* [Online] 22 09 2003. [Cited: 12 08 2010.] http://www.justice.gov/criminal/cybercrime/kashpurepr.htm.

29. **Gibson, Steve.** DNS Nameserver Spoofability Test. *Gibson Research Corporation.* [Online] 2010. [Cited: 12 08 2010.] https://www.grc.com/dns/dns.htm.

30. *DNS-based Detection of Scanning Worms in an Enterprise Network.* **Whyte, David, Kranakis, Evangelos and van Oorschot, P.C.** San Diego : s.n., 03-04 02 2005, Proceedings of the 12th Annual Network and Distributed System Security Symposium.

31. **Piscitello, Dave.** Conficker Summary and Review. *ICANN.* [Online] 07 05 2010. [Cited: 12 08 2010.] http://icann.org/en/security/conficker-summary-review-07may10-en.pdf.

32. *Mining Spam Email to Identify Common Origina for forensic Applcation.* **Wei, Chun, et al.** Ceara, Brazil : ACM, 16-20 03 2008, SAC, pp. 1433-1437. 978-1-59593-753-7.

33. **Ehrlich, Willa K, et al.** Detection of Spam Hosts and Spam Bots Using. *Usenix.* [Online] 05 04 2010. [Cited: 12 08 2010.] http://www.usenix.org/events/leet10/tech/full_papers/Ehrlich.pdf.

34. *Methods to identify spammers.* **Eggendorfer, Tobias.** Adelaide : s.n., 21-23 01 2008, e-Forensics. 978-963-9799-19-6.

35. **Jackson, Ben.** More Malware DNS Cache Scraping. *innismir.net.* [Online] 25 03 2010. [Cited: 14 08 2010.] http://www.innismir.net/article/483.

36. **abuse.ch.** Zues Tracker. *abuse.ch.* [Online] 20 06 2009. [Cited: 14 08 2010.] https://zeustracker.abuse.ch/blocklist.php.

37. **GNUCitizen.** dnsmap. *code.google.com.* [Online] 2010. [Cited: 12 08 2010.] http://code.google.com/p/dnsmap/ .

38. *Domain anme forensics: a systematic approach to investigating an internet presence.* **Nikkel, Bruce J.** 1, s.l. : Elsevier Ltd, 2004, Digital Investigation, pp. 247-255.

39. **RSnake.** ha.ckers Blog. *ha.ckers.org.* [Online] 10 06 2010. [Cited: 12 08 2010.] http://ha.ckers.org/blog/20100610/fierce-20-to-be-released/.

40. **Matt.** Corporate Information Discovery [Part 1]. *AttackVector.org.* [Online] 25 05 2010. [Cited: 12 08 2010.] http://www.attackvector.org/corporate-information-discovery-part-1/.

41. **dd.** Leaking private IP addresses cia DNS. *Sucuri.* [Online] 03 05 2010. [Cited: 12 08 2010.] http://blog.sucuri.net/2010/05/leaking-private-ip-addresses-via-dns.html.

42. **Hauser, van.** Attacking the IPv6 Protocol Suite. *The Hackers Choice (THC).* [Online] 2008. [Cited: 12 08 2010.] http://freeworld.thc.org/papers/vh_thc-ipv6_attack.pdf .

43. **Grossman, Jeremiah.** Top Ten Web Hacking Techniques of 2009 (Official). *Jeremiah Grossman Blog.* [Online] 12 01 2010. [Cited: 12 08 2010.] http://jeremiahgrossman.blogspot.com/2010/01/top-ten-web-hacking-techniques-of-2009.html.

44. **RSnake.** Persistent Cookies and DNS Rebinding Redux. *ha.ckers.org.* [Online] 20 01 2009b. [Cited: 12 08 2010.] http://ha.ckers.org/blog/20090120/persistent-cookies-and-dns-rebinding-redux/.

45. —. DNS Rebinding for Scraping and Spamming. *ha.ckers.org.* [Online] 18

11 2009a. [Cited: 14 08 2010.] http://ha.ckers.org/blog/20091118/dns-rebinding-for-scraping-and-spamming/,.

46. —. Session Fixation Via DNS Rebinding. *ha.ckers.org.* [Online] 16 11 2009c. [Cited: 14 08 2010.] http://ha.ckers.org/blog/20091116/session-fixation-via-dns-rebinding/.

47. —. DNS Rebinding for Credential Brute Force. *ha.ckers.org.* [Online] 17 11 2009d. [Cited: 14 08 2010.] http://ha.ckers.org/blog/20091116/session-fixation-via-dns-rebinding/.

48. **Constantin, Lucian.** DNS Rebinding Attack Can Be Used to Hack Home Routers. *Softpedia.* [Online] 14 07 2010. [Cited: 12 08 2010.] http://news.softpedia.com/news/DNS-Rebinding-Attack-Can-Be-Used-to-Hack-Home-Routers-147508.shtml.

49. **Ross, David.** Current Thoughts on DNS Rebinding. *Random Dross - MSDN Blogs.* [Online] 17 12 2009. [Cited: 14 08 2010.] http://blogs.msdn.com/b/dross/archive/2009/11/17/current-thoughts-on-dns-rebinding.aspx.

50. **Adkins, Heather.** For Google, DNS log analysis essential in Aurora attack investigation. *TechTarget.* [Online] 15 06 2010. [Cited: 12 08 2010.] http://searchsecurity.techtarget.com/news/article/0,289142,sid14_gci15149 65,00.html.

51. **Williams, Chris.** BT's 'illegal' 2007 Phorm trial profiled tens of thousands. *The Register.* [Online] 14 04 2008. [Cited: 14 08 2010.] http://www.theregister.co.uk/2008/04/14/bt_phorm_2007/.

52. **Head, Jonathan.** Turkey goes into battle with Google. *BBC News Europe.* [Online] 02 07 2010. [Cited: 06 08 2010.] http://news.bbc.co.uk/1/hi/world/europe/10480877.stm.

53. **McMillan, Robert.** After DNS problem, Chinese root server is shut down. *Computer World.* [Online] 26 03 2010. [Cited: 06 08 2010.] http://www.computerworld.com/s/article/9174278/After_DNS_problem_C hinese_root_server_is_shut_down.

54. **CZ.NIC.** DNSSEC Validator . *CZ.NIC Labs.* [Online] 2010. [Cited: 12 08 2010.] https://labs.nic.cz/dnssec-validator/.

55. **security-dns.net.** Zone Signing Made Simple. *security-dns.net.* [Online] 2010. [Cited: 06 08 2010.] http://security-dns.net/.

56. **Nichols, Shaun.** OpenDNSSEC service goes live. *v3.co.uk.* [Online] 09 02 2010. [Cited: 02 08 2010.] http://www.v3.co.uk/v3/news/2257605/opendnssec-service-goes-live?.

57. *Rolling Over DNSSEC Keys.* **Michaelson, George, et al.** [ed.] Ole J Jacobsen. 1, s.l. : Cisco Systems, 03 2010, The Internet Protocol Journal, Vol. 13, p. 35. http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_13-1/131_dnssec.html.

58. **Bruneau, Guy.** DNS Sinkhole ISO Available for Download. *SANS Internet Storm Center .* [Online] 19 06 2010. [Cited: 01 08 2010.] http://isc.sans.edu/diary.html?n&storyid=9037.

59. *Real-Time and Forensic Network Data Analysis Using Animated and Coordinated Vizualization.* **Krasser, Sven, et al.** West Point : s.n., 06 2005, Proceedings of the 2005 IEEE Workshop on Information Assurance.

60. **Makey, Jeff.** Blacklists Compared - 31 July 2010. *San Diego Supercomputer Center - Jeff Makey.* [Online] 06 08 2010. [Cited: 12 08 2010.] http://www.sdsc.edu/~jeff/spam/cbc.html.

61. **Rasmussen, Rod.** The Need For A DNS Emergency Alert System. *Security Week.* [Online] 26 07 2010. [Cited: 02 08 2010.] http://www.securityweek.com/need-dns-emergency-alert-system.

62. **Vixie, Paul.** Perspectives on a DNS-CERT. *Internet Systems Consortium.* [Online] 12 03 2010a. [Cited: 12 08 2010.] http://www.isc.org/community/blog/201003/perspectives-dns-cert.

63. —. Towards a DNSCERT Definition. *CircleID.* [Online] 17 06 2010c. [Cited: 12 08 2010.] http://www.circleid.com/posts/20100617_towards_a_dnscert_definition/.

64. **Nazario, Jose, Arends, Roy and Morrow, Chris.** Summary of the April, 2010 DNS-CERT Operational Requirements and Collaboration Analysis Workshop. *ICANN.* [Online] 04 2010. [Cited: 12 08 2010.] http://icann.org/en/topics/ssr/dns-cert-collaboration-analysis-24may10-en.pdf.

65. **van der Gaast, Wilmer and Contavalli, Carlo.** A porposed extension to the DNS Protocol. *Google Code Blog.* [Online] 27 01 2010. [Cited: 12 08 2010.] http://googlecode.blogspot.com/2010/01/proposal-to-extend-dns-protocol.html.

66. *The truth about Mobile Malware.* **Wysopal, Chris.** Oxford Belfry : SC Magazine, 2010. SC Forum.

67. *Hiding Data, Forensics, and Anti-Forensics.* **Berghel, Hal.** 4, 04 2007, Communications of the ACM, Vol. 50, pp. 15-20.

68. **Team Cymru, Inc.** Malware Hash Registry. *Team Cymru.* [Online] 15 02 2010. [Cited: 12 08 2010.] http://www.team-cymru.org/Services/MHR/.

ACKNOWLEDGEMENTS

6. AUTHOR BIOGRAPHY

Neil Fowler Wright has a Masters Degree in Computer Forensics from The University of Westminster, a Bachelor Degree in Physical Electronics from Warwick University and has spent 15 years in the Web, Internet and Computer Security industry working for major financial organisations and law firms. He has a passion for computer security; particularly network security and network forensics, and has worked with DNS since the mid 1990's.