

Passive Monitoring of DNS Anomalies

Bojan Zdrnja¹, Nevil Brownlee¹, and Duane Wessels²

¹ University of Auckland, New Zealand,
`{b.zdrnja,nevil}@auckland.ac.nz`

² The Measurement Factory, Inc.,
`wessels@packet-pushers.com`

Abstract. We collected DNS responses at the University of Auckland Internet gateway in an SQL database, and analyzed them to detect unusual behaviour. Our DNS response data have included typo squatter domains, fast flux domains and domains being (ab)used by spammers. We observe that current attempts to reduce spam have greatly increased the number of A records being resolved. We also observe that the data locality of DNS requests diminishes because of domains advertised in spam.

1 Introduction

The Domain Name System (DNS) service is critical for the normal functioning of almost all Internet services. Although the Internet Protocol (IP) does not need DNS for operation, users need to distinguish machines by their names so the DNS protocol is needed to resolve names to IP addresses (and vice versa).

The main requirements on the DNS are scalability and availability. The DNS name space is divided into multiple zones, which are a “variable depth tree” [1]. This way, a particular DNS server is authoritative only for its (own) zone, and each organization is given a specific zone in the DNS hierarchy. A complete domain name for a node is called a Fully Qualified Domain Name (FQDN). An FQDN defines a complete path for a domain name starting on the leaf (the host name) all the way to the root of the tree. Each node in the tree has its label that defines the zone. An example of an FQDN is “www.auckland.ac.nz.”. A domain is a subdomain when it is contained in another domain; in the previous example “auckland.ac.nz” is a subdomain of “ac.nz”.

As DNS is not centrally controlled, the domain names can be abused by attackers outside any organization. Besides domain name trading, attackers can shift domain name records quickly, making access blocking difficult. Another advantage for attackers is that from the client point of view security is often relaxed around DNS traffic, even in tightly controlled organizational networks. Most organizations have strict firewall policies at least on their perimeter firewalls, but DNS traffic is usually unrestricted because it is used by many other protocols. Attackers are commonly abusing this fact, not only to covertly send data over DNS, but also to deploy rogue DNS servers that can be used to completely control victim’s Internet behavior.

This paper describes a passive DNS anomaly detection project based on data captured at the University of Auckland Internet gateway. Our original

motivation for deploying the passive DNS monitor was to detect and correlate domains used for botnet controls. We quickly realized that the database is also a rich source of information about spam, anti-spamming tools, typosquatting, and other anomalies.

2 Related Work

Florian Weimer presented a passive DNS replication project at the FIRST 2005 conference [17]. As a result of his project a web site was established by RUS CERT [2] that allows public access to data collected “from the public Domain Name Service (DNS) system.” Weimer’s software, *dnslogger* consists of sensors deployed around a network that send captured DNS responses to a central collection service. Sensors encapsulate captured DNS responses in new UDP packets which are then relayed (in real time) to the collector. The collector analyzes received UDP packets and imports them into a database. Weimer’s passive DNS replication project is very similar to the one deployed at the University of Auckland, however, our setup is simpler and our database stores more information, for a longer period of time.

The University of Amsterdam [3] based their DNS capture project on Weimer’s work. Schonewille et al modified Weimer’s program to capture outgoing DNS queries in order to identify machines in the local network that have been compromised. Malware-infected machines tend to emit DNS queries that allow them to be easily identified.

John Kristoff’s DNSwatch [4] software can be used in a similar manner, as described by Elton et al [5], but it requires an external black list of well known malicious IP addresses (servers used to spread malware or contacted by malware).

3 Data Capture Methodology

DNS traffic uses either UDP or TCP on port 53 for communication [7]. Most DNS communication happens over UDP, which is the default protocol used by resolvers, i.e. applications that communicate with DNS servers on behalf of other applications when they need to resolve a DNS query. TCP was originally used only for zone transfers, but RFC 1123 [18] expanded the use of TCP as a backup communication protocol when the answer needs to be larger than 512 octets. In cases like this, the first UDP DNS response contains only partial answers. The truncation bit is set so that the resolver can repeat the query over TCP. However, RFC 2671, “EDNS0” [19], defined a new opcode field/pseudo resource record that allows UDP DNS traffic to be bigger than 512 octets. Because almost all of today’s DNS traffic uses UDP as its transport protocol, the deployment at the University of Auckland ignores TCP traffic.

DNS data is captured passively by sensors at the network edge, using an architecture designed to make implementation of sensors as simple as possible. A sensor is connected to a router SPAN port in order to get complete access to

all network traffic. Sensors run *tcpdump*, configured to write captured packets to a pcap file. Since we are only interested in DNS messages, we used the following *tcpdump* filter: `udp port 53 and (udp[10] & 0x04 != 0)`

Note that our filter only captures UDP DNS replies from authoritative sources, since we filter on their ‘Authoritative Answer’ bit [7]. We ignore TCP (for now) to simplify our parsing code, and because we observe relatively little TCP DNS traffic at the router. Since DNS replies always include the query data (in the Question section), there is little need to also collect DNS queries. Alas, our filter can cause some problems on certain large responses. If the DNS reply is larger than the path MTU, the UDP message will be fragmented. If that occurs, the first fragment usually contains enough information for anomaly detection.

Since our sensor is placed at the network perimeter, we see two types of DNS responses: those destined for the University’s local caching resolvers, and responses leaving the University’s own authoritative nameservers. The former are most interesting for our purposes here, but we did not attempt to filter out the latter from our database.

The sensors have a cron job that runs every hour. First, a new *tcpdump* process is launched. Then, the existing *tcpdump* process is killed. The pcap file containing data from the previous hour is compressed and sent to the collector.

Our database resides on the collector. The database holds only collected DNS data relevant for our research. The relevant data includes:

- Query name (name of the original query)
- Resource Record (RR) type (query type [7], ie A for address records)
- Resource Record data (answer returned by the authoritative DNS server)
- TTL (Time To Live) – value in seconds, set by the authoritative server, that allows the client DNS server or resolver to cache the answer
- First Seen Timestamp – timestamp showing when our sensor first saw this record

Rows in the database correspond to resource records in the Answer section of the DNS reply. We do not store records from the Authority or Additional sections.

Incoming pcap files are preprocessed by a program that unpacks the DNS messages and removes any duplicate entries. Duplicates typically occur for popular names with short TTLs. Since the only timestamp in our database is the First Seen column, a duplicate answer does not update the database and can be safely discarded. After all the new pcap files have been properly parsed, the program imports the data to the database. The collector runs on a system with an Intel Pentium 4D 3GHz processor and 2 GB of RAM. During peak times the collector imports 270,000 DNS messages in approximately 3 minutes.

Our sensors and collector have been running at the University of Auckland since 15 May 2006. As of 15th of January 2007 we have 260 GB of raw DNS data (uncompressed pcap files) and 50 million DNS records in the database. We archive raw pcap files on the collector, but only after zeroing out the source and destination IP addresses with Minshall’s *tcpdpriv* utility [8].

4 Results

4.1 Collected data

Captured DNS data shows a high number of NX (non-existent) DNS domains. Fig. 1 shows received authoritative DNS replies for the University of Auckland sensor, with separate traces showing non-existent domain (NX) responses and “valid” (see Table 1) responses. The month of September 2006 exhibits a very

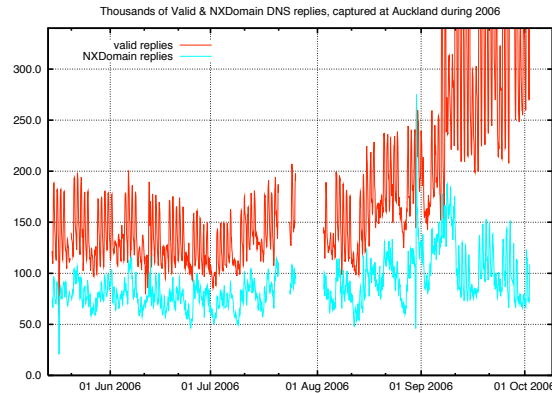


Fig. 1. Authoritative DNS replies captured at Auckland in 2006

different pattern from the previous months. During this month (and several months later), the University of Auckland network was flooded with incoming spam e-mail messages. Since the deployed anti-spam system tries to resolve all domain names and IP addresses seen in e-mail messages, this resulted in a huge increase in processed DNS replies.

4.2 Resource Record Type Prevalence

The current version of the dnsparse application running on the collector can successfully parse 15 resource record types. These types were identified as most commonly used in the first two weeks of captured data. Table 1 shows the distribution by resource record type of valid DNS response records in the collected data.

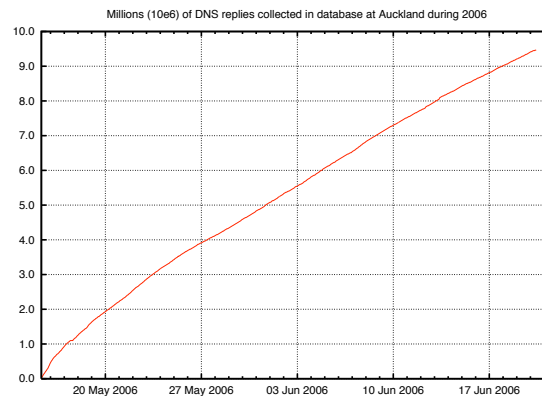
4.3 Impact of Anti-Spam Tools On The DNS System

Table 11 shows that address (A) resource records are responses to the majority of queries. While Jung et al [9] attributed this type of behaviour to user activities (web site browsing) our analysis shows that the biggest contributors to a high rate of A queries are anti spam engines. Spam detection depends

Table 1. Distribution of resource record types in DNS replies (answer section only)

RR type	Number of records	Percentage
A (1)	24096932	57.00
NS (2)	757825	1.79
CNAME (5)	652126	1.54
SOA (6)	16281	0.04
PTR (12)	11261024	26.64
MX (15)	2433120	5.76
TXT (16)	3047556	7.21
AAAA (28)	2202	0.005
SRV (33)	705	0.002
Total:	42267771	100%

on DNS to retrieve data from various real time black lists (RBLs). Spam software installed at the University of Auckland includes SpamAssassin, which will query several RBLs by default. For every domain detected in a message that is scanned, SpamAssassin will attempt to resolve it by issuing an A query for the domain in question. If the domain is successfully resolved, SpamAssassin will query various RBLs in order to determine if the IP address has been black-listed as sending spam. Queries to RBLs are also A type queries and answers. Depending on whether the tested IP address is present in the block list or not, the RBL DNS server will either return an authoritative DNS response in the 127.0.0.0/8 range (various codes are used, depending on the queried block list) or a “no such record” (NX) response. The database contains 12.2 million re-

**Fig. 2.** Size of the database (distinct FQDNs) at Auckland during 2006

source records that were responses to RBL queries. This accounts for 29% of all valid DNS responses received by the University of Auckland. We believe that this

number is even higher for NX domain responses. The number of TXT resource records, while not very high, is also related to e-mail processing. The gateway at the University of Auckland uses SPF [10] to verify whether the e-mail sender’s address has been spoofed, and SPF uses TXT resource records to list legitimate e-mail servers for a particular domain.

Figure 2 shows the size of our database as more FQDNs were added to it during 2006. Clearly, the database growth shows no sign of slowing; further evidence that spammers continue to fill DNS with more and more domains.

4.4 Typo Squatter Domains

Typo squatting is based on incorrect URLs entered by end users in their browsers. Mistyping of domains is very common and can be generally divided into several categories [11]:

- Spelling mistakes (www.auckland.ac.nz or www.auckland.ac.nz)
- Typing mistakes (www.eikipedia.org)
- Top-level domain appending(www.auckland.ac.nz.com) [12]

We found, by manual inspection, several highly exposed typo squatting domains in the database. Some of these are shown in Tables 2 and 3. The IP address shown in Table 2 hosted 1377 domain names, all of which were typo squatter domains. The content on all hosted web sites was the same and consisted of a search engine with various advertisements. Microsoft recently published a list of temporarily unused (parked) typo squatting domains as a result of the Strider URL Tracer with Typo-Patrol [13].

Table 2. Typo squatting domains based on mistyped words

DNS query	Answer	RR type	Entry added	TTL
www.gmaio.com	64.20.33.131	A	16/5/2006	7200
openopffice.org	64.20.33.131	A	17/5/2006	7200
www.forcasts.org	64.20.33.131	A	18/5/2006	7200
www.hontmail.com	64.20.33.131	A	19/5/2006	7200
www.eikipedia.org	64.20.33.131	A	19/5/2006	7200
economist.com	64.20.33.131	A	23/5/2006	7200

A lot of typo squatter domains that have been identified in our database use wildcard DNS records. As wild card DNS zones allow an administrator to setup resolution of any query in the zone he is controlling (for example, a “*.auckland.ac.nz” wildcard shown below will return the same response for any query for a host or subdomain in the auckland.ac.nz domain), all records that have been collected for such zones are directly a result of end users’ activities. While the investigation of detected typo squatter domains targeting large populations (such as those targeting Wikipedia or University of Auckland) did not reveal any

Table 3. Typo squatter domains attacking University of Auckland users

DNS query	Answer	RR type	Entry added	TTL
auckland.ac.nz	70.85.154.28	A	16/5/2006	43200
auckland.ac.nz	64.111.218.142	A	22/12/2006	43200
www.auckland.ac.nz	auckland.ac.nz	CNAME	16/5/2006	43200
www.cs.auckland.ac.nz	auckland.ac.nz	CNAME	17/5/2006	43200
webmail.ec.auckland.ac.nz	auckland.ac.nz	CNAME	29/5/2006	43200
gateway.auckland.ac.nz	auckland.ac.nz	CNAME	18/7/2006	43200

malicious activities, the risk associated with them is high as users who mistyped a subdomain rely on visual detection. Using cryptographic technologies for verification, such as SSL, is also of no help in this example if the attacker can install a SSL certificate for the hosted, typo squatter domain. In such an attack, the victim would have to detect the mistyped URL in order to detect the attack.

4.5 Fast Flux Domains

Fast flux DNS domains are those that have rapidly changing resource records. They also typically have low TTLs. Fast flux DNS domains are typically used for command and control servers [14] by worms. Once a target machine has been infected, it will talk to a central command and control server for further instructions (other stages of malware download, attacks etc.). To prevent easy location and take-down of the control and command server, attackers hard code a DNS domain in malware and frequently change the IP address it points to. This makes address-based perimeter network control of infected machines difficult as an administrator can not block IP traffic towards a particular IP address. Instead, an administrator needs to block access to a certain domain name, which can be done only on the main DNS server in an organization.

We have also observed another typical use of fast flux DNS domains, on web sites running on compromised machines. Spamming operations typically use fast flux domains to change IP addresses of the target web sites per different spam runs. The domain shown in table 4 was used only for three days, for a

Table 4. Fast flux domain records

DNS query	Answer	RR type	Entry added	TTL
contryloansnow.com	82.155.116.90	A	22/5/2006 07:52:15	5
contryloansnow.com	80.192.79.212	A	22/5/2006 07:52:17	5
contryloansnow.com	217.209.81.86	A	22/5/2006 08:21:18	5
contryloansnow.com	62.167.58.207	A	22/5/2006 08:22:21	5
contryloansnow.com	68.85.56.47	A	22/5/2006 08:22:24	5
contryloansnow.com	193.77.253.115	A	22/5/2006 08:25:07	5

limited spam run and changed its IP address 80 times. By reverse resolving IP addresses and geographically locating them we see (table 5) that they are scattered around the world and mainly located on cable/DSL line connected machines. Fast flux DNS domains can be detected by deploying external agents

Table 5. PTR records and geographical location of hosts used for a fast flux domain

IP address	PTR record(s)	Geographical location
82.155.116.90	bl6-116-90.dsl.telepac.pt	Portugal, Europe
80.192.79.212	80-192-79-212.cable.ubr01.edin.blueyonder.co.uk	U.K., Europe
217.209.81.86	h86n2fls33o1110.telia.com	Sweden, Europe
62.167.58.207	adsl-62-167-58-207.adslplus.ch	Switzerland, Europe
68.85.56.47	c-68-85-56-47.hsd1.ga.comcast.net	United States
193.77.253.115	BSN-77-253-115.dial-up.dsl.siol.net	Slovenia, Europe

that query the database and sort results by number of associated resource records in various time intervals. This way it is possible to detect potentially malicious DNS domains, if a certain threshold has been reached. TTL will generally have a low value for fast flux domains as the attacker needs client machines to resolve the domain name frequently, otherwise they will try to connect to the old cached IP addresses.

4.6 Anomalous records

Sorting the captured records by various criteria can be used to detect unusual records or activities. While searching for records with low TTL values can generally be useful in detection of fast flux domains, in order to detect anomalous records we need to perform a full database search.

A typical abuse can be detected by sorting DNS names (queries) by number of associated responses. Besides easy detection of fast flux domains, which will have hundreds, and sometimes thousands, of associated A records, this method detected some anomalous activities, as shown in Table 6, for the `ntc.net.pk` domain. The `ntc.net.pk` domain has in total 1319 A records associated. It is not clear what is the purpose of such DNS responses nor how and why were they resolved by systems or users at the University of Auckland. The WHOIS database [15] confirms that addresses 202.83.160.0–202.83.175.255 belong to the National Telecom Corporation in Pakistan so it seems that the name `ntc.net.pk` resolves to almost all IP addresses used by NTC. Manually querying the DNS server for `ntc.net.pk` returns only 8 IP addresses, which seem to randomly change every time this domain is resolved. This means that, in order to populate the DNS database, this DNS domain was resolved at least 480 times (3840 addresses in the block divided by 8 addresses per reply) by University of Auckland users.

Table 6. DNS records for ntc.net.pk domain

DNS query	Answer	RR type	Entry added	TTL
ntc.net.pk	202.83.160.238	A	15/5/2006 22:15:27	15
ntc.net.pk	202.83.168.98	A	16/5/2006 11:16:17	15
ntc.net.pk	202.83.168.7	A	16/5/2006 15:34:34	15
ntc.net.pk	202.83.174.29	A	16/5/2006 15:39:53	15
ntc.net.pk	202.83.175.65	A	16/5/2006 15:40:17	15
ntc.net.pk	202.83.174.174	A	16/5/2006 15:41:14	15

4.7 Record reputation

In a vast majority of cases, spammers sell their product through various web sites. The creation of SURBL (Spam URI Realtime Blocklist [16]) caused spammers to increasingly start using different domains per spam run, so called ‘throw away’ domains. The idea behind this is to register a new domain, run a spam campaign using that domain and then switch to a different domain. By doing this, spammers are trying to avoid their domain being blacklisted on SURBL; by the time the domain is blacklisted, the spammers have sent enough e-mails and will switch to a different domain.

By checking historical behavior of an IP address and associated DNS resource records with it, particularly NS records, it is possible to calculate the ‘reputation’ of a new DNS domain. The link in establishing whether the new domain is good or bad is through one of its NS records. The reputation can be calculated by checking the history of a particular record to see how many (and which) domains referred to it, or to a particular IP address. Table 7 lists domains

Table 7. Domains using ns0.quijindeshkinmas.com DNS server

DNS query	Answer	RR type	Entry added	TTL
funhderinmdasewio.com	ns0.quijindeshkinmas.com	NS	24/9/2006	300
vertionmdefunshjin.com	ns0.quijindeshkinmas.com	NS	24/9/2006	300
...
saderuijtungandsunastre.com	ns0.quijindeshkinmas.com	NS	8/12/2006	300
...
badesuijintunfeungan.com	ns0.quijindeshkinmas.com	NS	13/12/2006	300

that have been used in various spam runs, and are pointing to one DNS server. These are also the only domains associated with the ns0.quijindeshkinmas.com DNS server. Checking the ‘reputation’ of a DNS server in this way, we can determine whether a newly registered/seen domain has spam/malicious elements or not. For example, if the anti-spam system detects a new domain that has ns0.quijindeshkinmas.com as its NS record, the system can automatically deduce that this domain is malicious or used for spam because historically there have

been no legitimate records related to this DNS server. This information can then be used similarly to all other rules in SpamAssassin.

Detected spam related domains shared the following characteristics:

- FQDNs end in a top level domain, such as .com
- Domain names are not English words
- All domains use ns0.quijindeshkinmas.com and ns0.kilonherunhasedun.com as their DNS servers.
- A records for particular domains are used only while the spam run associated with this domain is active. After it ends, the domain is left idle. A records are also spread around various providers.

5 Conclusion And Future Work

Passively collected DNS data stored in a database allows one to determine historical behavior of particular DNS records, and of the linkages between them. Since the quality of data and possibilities for analysis rise with the number of sensors (or the clients whose DNS traffic is being monitored), installing additional sensors, around the world should enable better detection of anomalies.

Automated analysis of data in the database could quickly detect anomalies and malicious attacks and thereby serve as an early alert system against spam and worm attacks.

The data should be crawled with specialized agents, such as Microsoft's Strider URL Tracer [13] to allow for near to real time detection of malicious domains. Unfortunately, our data has been already seen, i.e. a client tried to resolve it, but it should still be possible to black list the domain and alert other users which makes the viable time for an attack shorter and an attackers job more difficult.

We hope to establish a set of six to ten geographically dispersed sensors that would allow collection of DNS data from different user groups. We invite readers to contact us if they are willing to participate. We will also make the web interface for querying the database available to the public.

5.1 Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. 0427144.

References

1. P. V. Mockapetris, K. J. Dunlap: Development of the Domain Name System. ACM Symposium proceedings on Communications architectures and protocols (SIGCOMM 88), vol. 18, issue 4, 1998
2. RUS-CERT: Passive DNS replication.
<http://cert.uni-stuttgart.de/stats/dns-replication.php>

3. A. Schonewille, D. v. Helmond: The Domain Name Service as an IDS. Research Project for the Master System- and Network Engineering at the University of Amsterdam, February 2006
4. J. Kristoff: DNSWatch. <http://aharp.ittns.northwestern.edu/software/dnswatch>
5. N. Elton, M. Keel: A Discussion of Bot Networks. EDUCAUSE 2005, <http://www.educause.edu/ir/library/pdf/SPC0568.pdf>, April 2005
6. TCPDUMP/libpcap public repository. <http://www.tcpdump.org>
7. P. Mockapetris: Domain Names Implementation and Specification. RFC 1035, November 1987
8. Tcpsdpriv – A program for eliminating confidential information from packets collected on a network interface. <http://ita.ee.lbl.gov/html/contrib/tcpsdpriv.html>, October 2005
9. J. Jung, E. Sit, H. Balakrishnan, R. Morris: DNS Performance and the Effectiveness of Caching. ACM Transactions on Networking, Vol. 10, No. 5, pp. 589-603, October 2002
10. M. Wong: Sender Authentication What To Do. A Messaging Anti-Abuse Working Group White Paper, available at <http://www.openspf.org/whitepaper.pdf>, November 2004
11. Sequitur IPS: Domain name disputes, cybersquatting and UDRP cases. <http://www.sequitur-ips.com/domain-name-disputes/library.html>
12. E. Gavron: A Security Problem and Proposed Correction With Widely Deployed DNS Software. RFC 1535, October 1993
13. Y. Wang, D. Beck, J. Wang, C. Verbowski, B. Daniels: Strider Typo-Patrol: Discovery and Analysis of Systematic Typo-Squatting. Microsoft Research Technical Report (to be submitted to the 2nd Usenix Workshop on Steps to Reducing Unwanted Traffic on the Internet (SRUTI 06)), <http://research.microsoft.com/URLTracer>
14. Gadi Evron, SecuriTeam Blog: Looking behind the smoke screen of the Internet: DNS recursive attacks, spamvertised domains, phishing, botnet C&Cs, International Infrastructure and you. <http://blogs.securiteam.com/index.php/archives/298>
15. L. Daigle: WHOIS Protocol Specification. RFC 3912, September 2004
16. SURBL Spam URI Realtime Blocklists. <http://www.surbl.org>
17. F. Weimer: Passive DNS Replication. FIRST 2005, April 2005
18. Internet Engineering Task Force: Requirements for Internet Hosts Application and Support. RFC 1123, October 1989
19. P. Vixie: Extension Mechanisms for DNS (EDNS0). RFC 2671, August 1999