Pheasting on the Crumbs of Misconfigured Networks

© Tor Houghton <torh@bogus.net>

31 January 2006

Abstract

This paper attempts to describe some risks associated with badly configured Domain Name System (DNS) resolvers (and provide possible reasons why many are thusly configured). In an attempt to reach a wider audience, the subject matter has deliberately been written in a style that may by some be considered somewhat "glib".

Introduction

The Domain Name System is a fundamental component of the modern Internet. It's the glue that holds the Internet together and makes it possible for human consumption. Everyone uses it (even relies on it), but it's probable that in excess of 95% of the Internet's users don't comprehend even its very basics. Why should they? To most people, the Internet "just works".

Of course, there are threats to the DNS (and the clients which use it), such as hijacking and poisoning (which may result in e.g. "pharming" attacks). However, this paper is not going to talk about these. In fact, it is not going to talk about threats to the DNS at all.

Rather, it is going to talk about service threats, information leakage, attack discovery as well as possible advertising revenue resulting from people's blind reliance on (or faith in) advice found on the Internet.

DNS Recap

When a client (such as an FTP client, web browser, mail server and so forth) accesses DNS in order to obtain a host's numerical address, it does this through the host's *resolver* library. This component is normally configured at the operating system level and describes which DNS servers to use and what domains to try in turn if a client has not requested the full name[1] of a host (e.g. "www" instead of "www.bogus.net"[2]).

When "bogus.net" was registered back in 1995, the name was chosen partly as a joke, reflecting the state of the equipment that the network consisted of. Over the years, however, it appears that people have been using this domain for *their* local networks.

So here's when that blind reliance thing comes in. Authors are advising their readers to use "bogus.net" during network configuration[3]. Others use the domain as examples in their product documentation[4].

Whatever the reason, there are a lot of resolvers out there configured to use a domain which they do not belong to. On any given day, the average number of DNS lookups for hosts in the "bogus.net" domain is around 20000 (not counting repeat requests, this amounts an average of roughly 6300 unique "A" record queries each day).

The following chart shows the averages over a 14-day period, where red shows the number of unique requests, blue the

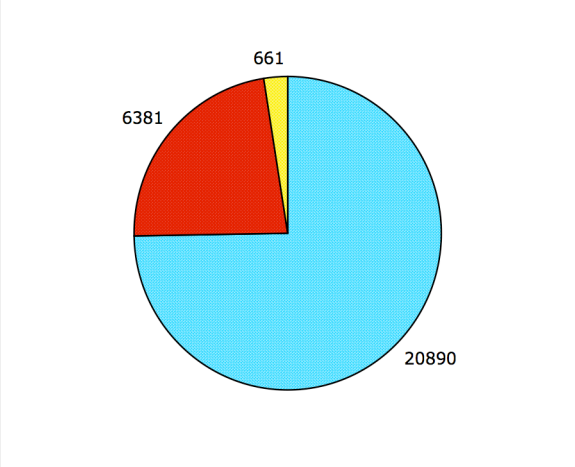
[1] FQDN, or Fully Qualified Domain Name.

[2] Please note that the automatic addition of top level domains (e.g. ".com") when a user types in "google" in some web browsers has nothing to do with the resolver subsystem.

[3] http://www.isaserver.org/tutorials/ISA_Server_Security_Checklist__Part_1_Securing_the_Operating_System_and_the_Interface.html

[4] http://linux.com.hk/PenguinWeb/manpage.jsp?section=8&name=addclients

total number of requests and yellow indicates the total number of unique clients (often the DNS server of the client's service provider):



(This analysis was done before the proliferation of viruses and worms such as "Mytob" which make a great number of repeat requests for possible mail relays.)

What's the fuss?

Let's take a few examples. First, there is the relative non-issue of information leakage. If someone relies on split-horizon DNS to protect the namespace of internal hosts, a simple typo may expose both the internal address space and the naming convention used. For example (packet dump modified for brevity):

```
10:28:24    195.x.x.x.36634 >
       193.214.208.178.53: A?
       172.29.30.1.bogus.net.
10:28:31    195.x.x.x.36634 >
       193.214.208.178.53: A?
       172.29.30.2.bogus.net.
..
..
10:28:39    195.x.x.x.36634 >
       193.214.208.178.53: A?
       172.29.30.31.bogus.net.
```

This only reveals the address of the host making the lookup[5], not necessarily the source of the leak itself. To find out this information, we would have to add a CNAME wildcard record for "172.29.30" to the *bogus.net* zone file, point them to a host of our choice, and wait for the lookups to occur again.

Depending on the outbound security policy of the source, it may be possible to determine the source of the leak as well as any services it is attempting to communicate with.

One attack vector that presents itself is that of rogue services. Once we have established what the client is trying to talk to, an attacker could provide bogus services which could be used to harvest sensitive information such as email, user-names, passwords and so on.

Perhaps more serious are the options which present themselves when a user has misspelled the host portion of a URL. Typical typos include "eager typing" (such as "wwww.google.com" and "wwww.altavista.com"), "fat fingers" (failing to hit the "Enter", or "m" keys, such as "www.google.com]" and "www.google.con") and errors provoked by programming mistakes (e.g. "${domain}.bogus.net").

Live tests show that users do not always appear to spot the difference between the "real" Google and e.g. wwww.google.com.bogus.net ("Boogle"). During testing, using a simple page with an HTML `<IFRAME>` pointing to the real Google[6], logging showed that the same user (as determined by their IP address and browser "UserAgent" information) would revisit Boogle time and again.

Armed with this information, it is quite plausible that an attacker could successfully launch a more sinister attack: The harvesting of usernames and passwords by creating a Trojan web mail client for services such as Hotmail, Yahoo! or Google Mail or through inducing the user to launch malicious code. Secondly, because the user is accessing a web ad-

[5]Often the client's DNS server (i.e. belonging to an ISP).

[6]Ensuring a) that a user's search actually completes and b) that user submitted data is only sent to Google.

dress that is not spoofed, the Trojan site could even have a legitimate SSL certificate signed by a trusted Certificate Authority, thus circumventing one of the controls that usually thwart "phishing" attacks (browser warnings about unsigned or invalid SSL certificates).

Additionally, sensitive information such as usernames, passwords or session state tokens are at risk of being deposited in server logs, e.g.:

```
206.x.x.x - -
        [21/Jan/2006:04:22:12] "GET
        //XXXXXXX:XXXXXXX@XXXXX.net
        /] 404 ..
 67.x.x.x - -
        [24/Jan/2006:03:37:34] "GET
        //YYYYYY:YYYYYY@YYYYY.com
        /] 404 ..
```

Revenue?

Another twist to this tale is that of traffic redirection. By continuously monitoring queries made to the bogus.net DNS servers, one could create a number of "valid" CNAME records for common mistyped hosts, a "top ten" for the month, for instance. Instead of directing them to a bogus web page (see "Boogle" above), it is quite possible to instead direct the client to another website altogether.

Conclusion

Blindly using domain names suggested in documentation is a bad idea. Instead of using some imaginary domain, writers should suggest their readers use their own domain name if they have one, or use one of the test domains assigned by IANA[7]. These are "example.com", "example.net" and "example.org" and have IANA addresses as their NS records. The domains have been earmarked for testing purposes and their servers will respond with a name resolution error ("NXDOMAIN") for any queries to them.

Other domains to look out for: are "contoso.com", "foo.com", "bar.com", "domain.com", "mydomain.com" and "localdomain.com".

This paper has only discussed threats that occur through the misconfiguration of DNS resolvers. Everything and more is possible if the resolver was reconfigured by a Trojan or worm, e.g. by changing the domain search order or indeed changing the "nameserver" entries to point at a "rogue" DNS server.

(Updated versions of this document may appear on http://www.bogus.net/~torh/.)

Credit

Simon Walker, who sensibly told me to scrap the original title of the paper ("Phisting") and use his suggestion instead.

[7]The Internet Assigned Number Authority, http://www.iana.org.