NGS NISR
Next Generation Security Software Ltd.

The Pharming Guide

Understanding & Preventing DNS-related Attacks by Phishers

Abstract

Exploiting well known flaws in DNS services and the way in which host names are resolved to IP addresses, Phishers have upped the ante in the cyber war for control of a customer's online identity for financial gain.

A grouping of attack vectors now referred to as "Pharming", affects the fundamental way in which a customer's computer locates and connects to an organisations online offering. Enabling the Pharmer to reach wider audiences with less probability of detection than their Phishing counterparts, pharming attacks are capable of defeating many of the latest defensive strategies used customer and online retailer alike.

This paper, extending the original material of "The Phishing Guide", examines in depth the workings of the name services of which Internet-based customers are dependant upon, and how they can be exploited by Pharmers to conduct identity theft and financial fraud on a massive scale.

Author

Gunter Ollmann, NGS – email: gunter [at] ngssoftware.com

Section 1: **Background**

This paper focuses upon a recent group of attack vectors used by criminals to target an organisation's customers for identity theft and financial fraud. Closely related to Phishing attacks, this new attack manipulates the ways in which a customer locates and connects to an organisation's named hosts or services through modification of the name lookup process. The attack vectors, commonly referred to as Pharming, have the ability to bypass many traditional Phishing attack prevention tools and affect larger segments of an organisations customer-base.

Given the apparent complexity of this attack vector, this paper seeks to carefully explain many of the background processes all Internet-based customers use on a daily basis to connect to an organisations commercial service, and examines how frailties in them can be exploited by an attacker to conduct a Pharming attack.

Readers should ensure that they fully understand how traditional Phishing attacks are conducted and the defensive strategies that have been adopted in the past to protect against them. Ideally the reader should be familiar with the author's previous paper "The Phishing Guide" as several sections of this paper reference information contained within the earlier whitepaper.

1.1. Pharming History

While the term Phishing has been around since early 1996, it wasn't until the later stages of 2003 that email-based phishing attacks began to become a popular attack vector for cyber criminals as a means to conduct financial fraud and identify theft. By mid 2004, Phishing attacks were headline news around the world and most customers of online financial services or retailers were deeply concerned that they too could fall victim to an attack. Indeed, most customers still feel that way today.

During this time, numerous software vendors and managed service providers developed sophisticated tools and techniques to help protect against the threat of Phishing. Providing solutions for the customer's desktop, the organisations servers and enterprise gateways, they helped protect against many popular forms of Phishing attack.

As has been the case for many years, the thrust and parry between attacker and protector has led to a cyber arms war.

The latest attack vectors being exploited by criminals to achieve identity theft and fraud exploit frailties within the way customers locate and connect to an organisations online service. While Phishers tended to make use of obfuscation methods to disguise the true destination reached by the customer, Pharming attacks manipulate various components of core domain and host naming systems to misdirect the customer to an alternative destination – one completely under their control. When an attack is carried out at this level, many of the security solutions devised for protecting against Phishing attacks are also deceived.

Pulling from a pool of well known exploit techniques – such as DNS hijacking, DNS spoofing, cache poisoning, etc. – Pharmers have been able to alter the DNS resolution information that customers need to resolve (and consequently reach) an organisations online services.

In March 2005 the well known and respected SANS security organisation issued a warning about new attacks by attackers that corrupted some DNS servers so that requests for a number of ".com" sites were directed to alternative servers maintained by the Phishers. Using a rogue DNS server posing as an authoritative DNS server for a particular .com domain, Pharmers were able to cache poison several ISP-level DNS servers and requests for more than 900 unique Internet addresses and more than 75,000 email messages were redirected.

Using even less complex techniques, in January 2005 the domain name for a large New York ISP (Panix) was hijacked to a site in Australia while earlier, in 2004, a German teenager hijacked the eBay.de domain name. Through simple social engineering and a good telephone manner, an attacker can often fool a domain registrar and gain control of a domain. Such a thing happened on the 24th April 2005 when users of the secure webmail service - Hushmail - were redirected to a "defaced" webpage.

Section 2: **Understanding Host Resolution**

2.1. Connecting to Hosts

The attack vectors exploited in Pharming abuse the name associations a customer has with a particular service they wish to connect to. While Phishing attacks typically rely upon interception of traffic destined for a particular host or URL through obfuscation mechanisms, Pharming attacks manipulate the underlying processes used by web browsers (and almost all other Internet services) to identify and connect to a named host.

This section focuses upon the underlying mechanisms used by customers to locate and connect to an organisations online service through name resolution services. Whilst the majority of Pharming attacks currently manipulate the DNS service, in the future alternative resolution services such as autocompleters and search engines are also likely to succumb to attack.

In order to understand how Phishers and Pharmers may carryout their attacks, it is first necessary to understand how these often neglected name resolution services work and appreciate the frailties being exploited.

2.2. Understanding Conventional DNS

Although every person "surfing the net" will transparently make use of its services many hundreds of times per day, DNS is probably the least understood networking service of which the Internet depends upon. Without DNS managing the translation from complex IP addresses to helpful location names and back again, many of today's online businesses and offerings could not function.

While elegantly simple from a superficial perspective – up close DNS is a multifaceted beast packed full of legacy design compromises, layers upon layers of interconnected services, and complex flows of "trusted" information.

In order to understand the role DNS plays in Pharming attacks (and many other attack vectors), it is important to understand how it actually works.

2.2.1. Connecting Computers

The protocols used by the Internet to allow two computers to communicate with each other typically rely upon numeric addresses of the format 123.456.789.012 in order to distinguish one from another. While these numeric addresses are efficient for computers, most humans tend to prefer to use an alias when connecting to different computers. So, instead of referring to a computer as 192.168.20.12, it may be more convenient to call it "Fritz" - the file server in the local library.

In the early days of the Internet (and some time before it was really popular), each computer would have its own list of mappings between an alias and it numeric address (also referred to as its "IP Address"). Typing the alias "Fritz" into the web browser would result in an invisible translation between the name and IP address, and the web browser would then connect to the file server in the local library.

This local file format worked well for small groups of machines, but scalability and manageability issues were a problem. The Domain Naming System (DNS) was invented to solve these problems. DNS servers are deployed throughout the Internet and allow multiple regulatory and commercial entities to manage the translation of unique aliases to computers at a global level.

2.2.2. DNS Hierarchy

Core to DNS functionality is a hierarchical architecture of servers providing linking information to ever more informative and specific servers. This can be visualised as a simple pyramidal structure with information flowing from the top "Root Servers", through to the "Top Level Domain" servers and finally onto the "authoritative" domain servers.

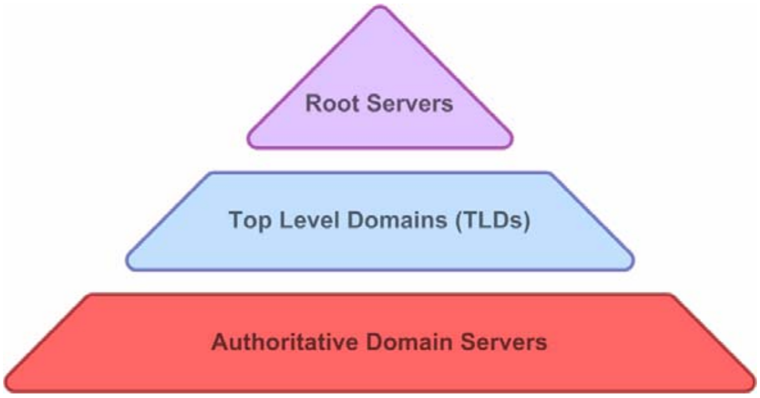


Figure 1: Simplified DNS Hierarchy

This simple structure allows networked users to locate the hosts and services they are looking for anywhere in the world, while allowing organisations to manage their online host names using their own DNS servers.

Consider a customer who wishes to connect to the online web banking service of MyBank Limited which has been designated the name www.mybank.com. (frequently referred to as a "Fully Qualified Domain Name" or FQDN) by that organisation. The IP address of the host is controlled by the company MyBank and is stored within a DNS server under that organisations control. In order for the customer's web browser software to connect to www.mybank.com, it must first discover the IP address of the web server by querying MyBank's DNS server. However, the customer's computer most likely will not know the IP address of the DNS server, and must first uncover these details before it can actually query it. To find out the DNS servers address, a number of queries must be made to other servers that contain details of how to reach it.

Firstly the user's computer will require a DNS resolver to connect to a known root server (sometimes referred to as "**.**" (dot) servers) and obtain the details of a server that knows about **.com.** addresses. The root server will thus provide details of an appropriate Top Level Domain (TLD) server, which is then queried by the DNS resolver for details about the authoritative **.mybank.com.** address. Finally, the resolver queries the organisations authoritative DNS server for the IP address of the "**www**" host (full name www.mybank.com), and the user can then use this IP information to connect to the web banking host.

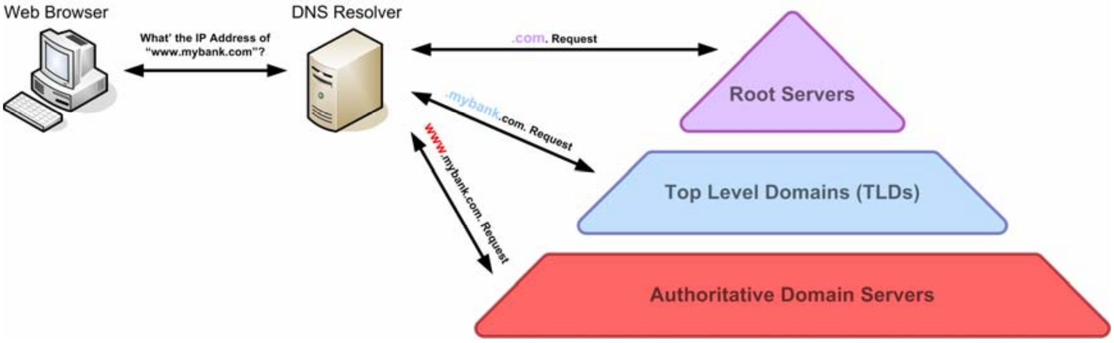


Figure 2: DNS resolution showing the DNS resolver in action

This entire process may only take a few fractions of a second to complete, and is sometimes visible to customers as a pause between typing in a specific web address into their web browsers and hitting go, and seeing the first parts of the web page to appear. To speed things up, the customer's computer may decide to cache the IP address associated with the host name for a period of time – thereby not requiring any repeated DNS lookups for a while.

Root Servers

There are a number of strategically placed Root Servers underpinning the entire Internet – referenced by 13 distinct names – each is assigned a letter ranging from A to M, with the full name containing ROOT-SERVERS.NET. DNS resolvers use hard-coded IP lookup tables for these servers.

Their exclusive role is to point DNS resolvers to the appropriate TLD server for their query.

The following list contains the names and locations of all 13 critical Root Servers. It is important to remember that these Root Servers are not single servers, but clusters of (globally) load balanced servers.

Servers	Location(s)	Historical Name
A.ROOT-SERVERS.NET	Dulles, VA, USA	ns.internic.net
B.ROOT-SERVERS.NET	Marina Del Rey, CA, USA	ns1.isi.edu
C.ROOT-SERVERS.NET	Herndon, VA, USA Los Angeles, CA, USA	c.psi.net
D.ROOT-SERVERS.NET	College Park, MD, USA	terp.umd.edu
E.ROOT-SERVERS.NET	Mountain View, CA, USA	ns.nasa.gov
F.ROOT-SERVERS.NET	Auckland, New Zealand Sao Paulo, Brazil Hong Kong, China Johannesburg, South Africa Los Angeles, CA, USA New York, NY, USA Madrid, Spain Palo Alto, CA, USA Rome, Italy Seoul, Korea San Francisco, CA, USA San Jose, CA, USA Ottawa, ON, Canada	ns.isc.org
G.ROOT-SERVERS.NET	Vienna, VA, USA	ns.nic.ddn.mil
H.ROOT-SERVERS.NET	Aberdeen, MD, USA	aos.arl.army.mil
I.ROOT-SERVERS.NET	Stockholm, Sweden Helsinki, Finland	nic.nordu.net
J.ROOT-SERVERS.NET	Dulles, VA, USA Mountain View, CA, USA Sterling, VA, USA Seattle, WA, USA Atlanta, GA, USA Los Angeles, CA, USA Amsterdam, The Netherlands	
K.ROOT-SERVERS.NET	London, UK Amsterdam, The Netherlands	
L.ROOT-SERVERS.NET	Los Angeles, CA, USA	
M.ROOT-SERVERS.NET	Tokyo, Japan	

Given the current structure and constraints of the DNS service, only 13 root servers are typically referenced. This is due to data limitations requiring all root server names to fit within a single UDP packet given the current naming scheme. However, this is a compromise limitation and there is nothing to prevent the use of shorter names for root servers to allow for more to be listed within a single UDP packet, or the use of multiple UDP packets, in the future should it be so agreed.

Top Level Domain Servers

The Top Level Domain server's role is to point DNS resolvers to an Authoritative Domain server. The TLD layer in the pyramid is actually are divided into two distinct classes – Generic TLDs (gTLD) and Country-code TLDs (ccTLD) – with the ccTLD's also having a range of subdomain servers.

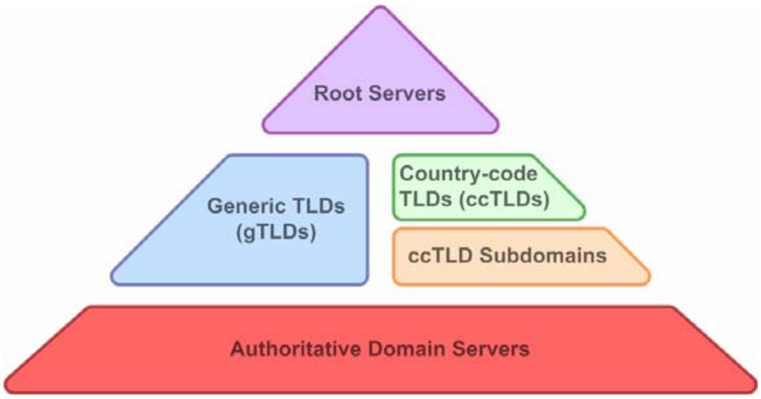


Figure 3: Top Level Domain (TLD) breakdown

Until mid-2000 (prior to a number of high profile denial of service attacks) the root servers also handled all requests for the generic top level domains. This responsibility was later removed from the root servers and led to the creation of dedicated TLD servers. gTLD's provide resolver information for the common .com, .net, .org, .gov, .mil and .gov domain groupings, while ccTLD's provide resolver information for country specific domain groupings – such as .UK for the United Kingdom. In many cases ccTLD's also allow for subdomains – such as .ac.uk for academic institutes within the United Kingdom.

While the Root Servers are critical to the operation of the Internet, gTLDs and ccTLDs get many more requests. Consequently gTLDs and ccTLDs are now more critical to the successful operation of DNS than the Root Servers.

Authoritative Domain Servers

Authoritative Domain Servers (or Name Servers) manage a Zone and either provide IP address lookup information themselves, or delegate the lookup of zone/sub-zone information to other DNS name servers.

The technical specifications for DNS define two classes of name servers: "Primary Masters" and "Secondary Masters". A Primary Master name server maintains a zone file which is stored locally on the host. Secondary Master name servers ideally get their zone information from an authoritative name server for that zone – referred to as its master server – via a process called "zone transfer".

For network resilience reasons, an authoritative domain server would typically have a list of at least one Secondary Master associated with a registered domain name – ideally more Secondary Master name servers are used.

DNS Servers

The DNS server (a generic term for Name Server) for a particular domain provides forward and reverse resolution services between a specific host name and its IP address. For example, the DNS server for MyBank Limited may contain entries such as:

- The IP address of "www" is 100.1.2.10 (www.mybank.com)
- The IP address of "ftp" is 100.1.2.11 (ftp.mybank.com)
- The IP address of "mail" is 100.1.2.14 (mail.mybank.com)
- The IP address of "testserver" is 100.1.10.12 (testserver.mybank.com)

Resolvers

Resolvers are the software applications (typically transparent) used by a customers computer to automatically access names servers and resolve host name to IP address details.

Resolvers are designed to handle the following:

- The querying of a name server,
- The interpretation of responses from a name server (e.g. resource records or an error response),
- The return of gathered information back to the customer programs that requested it.

2.2.3. DNS Host Discovery

Having loosely covered the processes used to resolve a domain name to an IP address, it is important to understand how the global DNS environment works in the real world. While the basic pyramid structure of the previous section was fine, to understand the intricacies of DNS resolution it is necessary to have a closer look at this hierarchical structure.

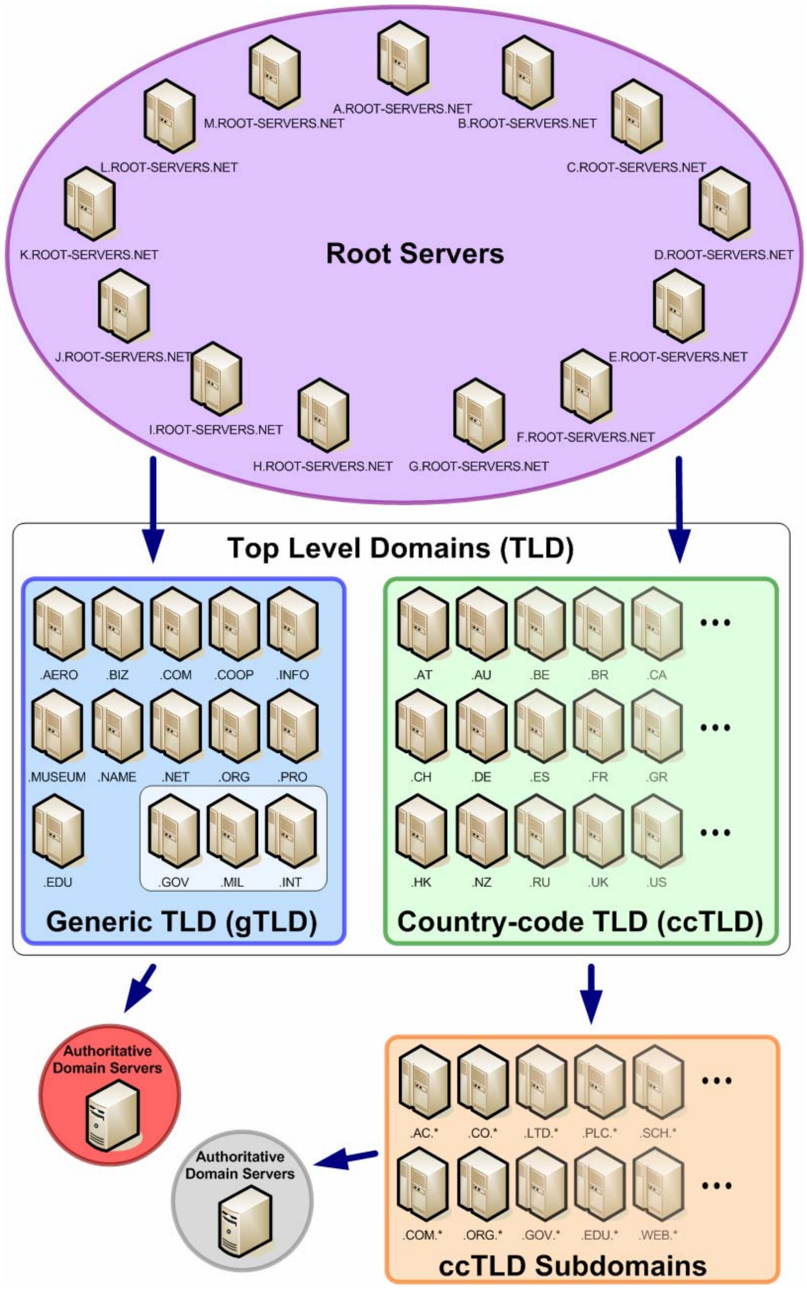


Figure 4: Detailed hierarchical view of the DNS resolution structure

If we now take a closer look at the resolution of the IP address associated with the domain name www.mybank.com – but from two different geographic locations (London and New York) – we will find that the DNS resolver can take two different paths in its discovery of this information:

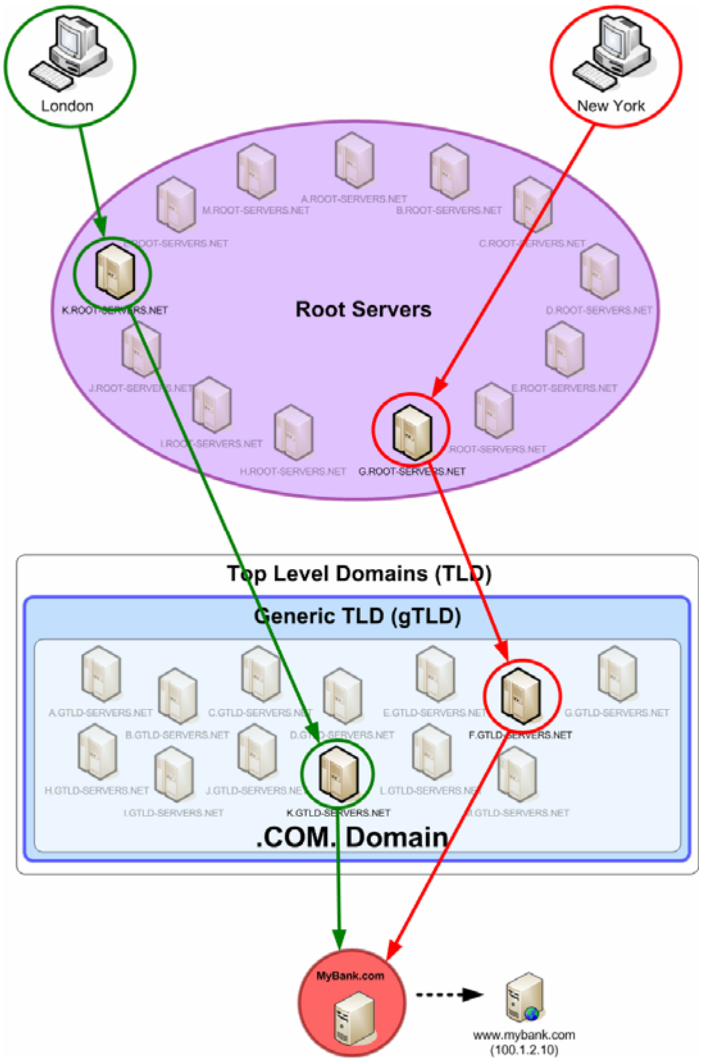


Figure 5: Resolution of MyBank DNS server from two physical locations

From the example above, we see that the London customer queries K.ROOT-SERVERS.NET root server, then the K.GTLD-SERVERS.NET load balanced .com gTLD server, before being directed to the authoritative MyBank.com DNS server – which knows the IP address of www.mybank.com to be 100.1.2.10. Meanwhile, the New York customer has queried the G root server and the F .com gTLD server before reaching the authoritative MyBank.com DNS server. If these two customers were to repeat their lookups at a later date or time they would likely end up querying different servers.

2.2.4. DNS Server Delegation

Things gradually become more complex when you take into account an international organisation that has registered and manages multiple domains.

Consider the example of MyBank Limited which has registered three domains (mybank.com, mybank.co.uk, and mybank.com.au) and manages three DNS servers strategically placed to handle customer host lookups in their three key geographic business regions (Europe, Americas, and the Pacific). Along with these DNS servers, MyBank Limited has separate hosting facilities for key Internet-accessible mail and web services in London, Atlanta and Sydney. The following diagram shows this arrangement.

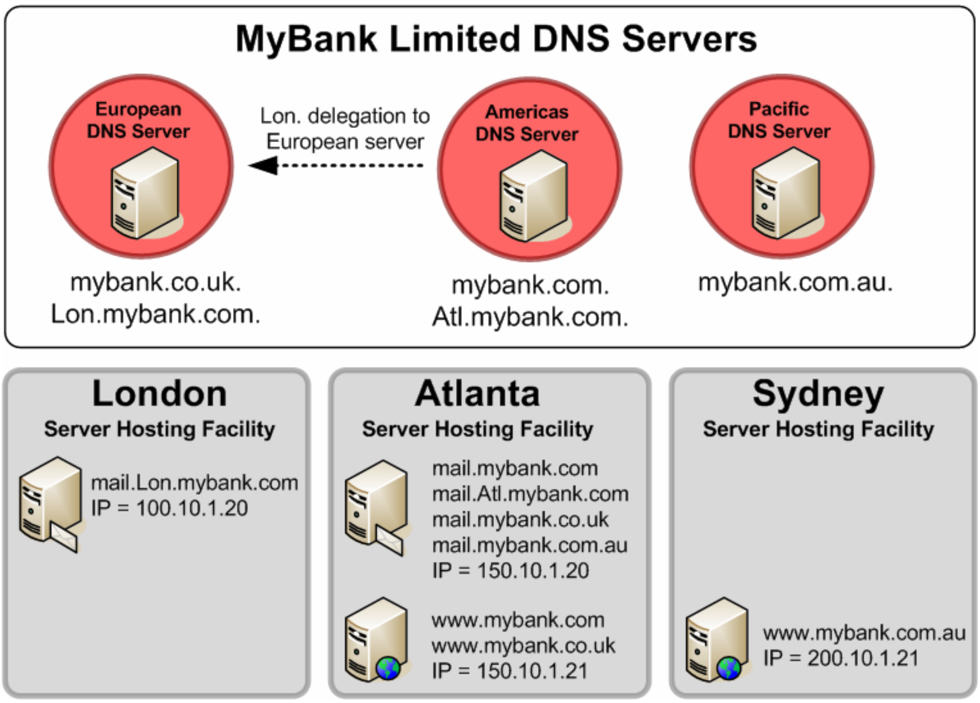


Figure 6: Multiple corporate DNS servers with internationally distributed host servers

MyBank Limited has found that, given the size of their London and Atlanta offices, that they wish to use subdomains to help manage email services – hence the addition of Lon.mybank.com and Atl.mybank.com domains. As part of this plan, MyBank Limited delegates the management of the Lon.mybank.com to the European DNS server (which also manages the mybank.co.uk domain) – even though the authoritative name server for mybank.com is located in the Americas.

In addition, MyBank Limited has rationalised its server hosting such that requests for www.mybank.com and www.mybank.co.uk resolve to the same IP address – which corresponds to a fully redundant cluster of web servers located in Atlanta. Due to regulatory requirements in Australia, MyBank has been required to ensure that all Australian web requests go to a separate/dedicated server.

Finally, MyBank Limited has configured their DNS servers such that all external mail goes to the same mail host located in Atlanta – regardless of which regional gTLD or ccTLD it was destined for. The only exception is for email sent specifically to the London office with the email address of whoever@lon.mybank.co.uk.

This arrangement of DNS servers and hosting means that customer requests for different host services will require different resolver paths to uncover the IP address they require. The following table and graphic helps to illustrate this point.

www.mybank.com	mail.lon.mybank.com	mail.mybank.com.au	www.mybank.co.uk
Root Server C.ROOT-SERVERS.COM	**Root Server** M.ROOT-SERVERS.COM	**Root Server** K.ROOT-SERVERS.COM	**Root Server** I.ROOT-SERVERS.COM
Generic TLD F.GTLD-SERVERS.COM	**Generic TLD** B.GTLD-SERVERS.COM	**.AU Country-code TLD** SEC1.APNIC.NET	**.UK Country-code TLD** NS7.NIC.UK
MyBank Americas NS0.MYBANK.COM	**MyBank Americas** NS0.MYBANK.COM	**.COM.AU ccTLD Sub.** NS2.AUSREGISTRY.NET	**.CO.UK ccTLD Sub.** NS7.NIC.UK
www.mybank.com 150.10.1.21	**MyBank Europe** NSE.MYBANK.COM	**MyBank Pacific** NSP.MYBANK.COM	**MyBank Europe** NSE.MYBANK.COM
	mail.lon.mybank.com 100.10.1.20	**mail.mybank.com.au** 150.10.1.20	**www.mybank.co.uk** 150.10.1.21

Note that the Root Server, Generic TLD, Country-code TLD and ccTLD subdomain host addresses may change with each new DNS lookup request, as they all make use of multiple load-balanced servers to answer lookup requests.

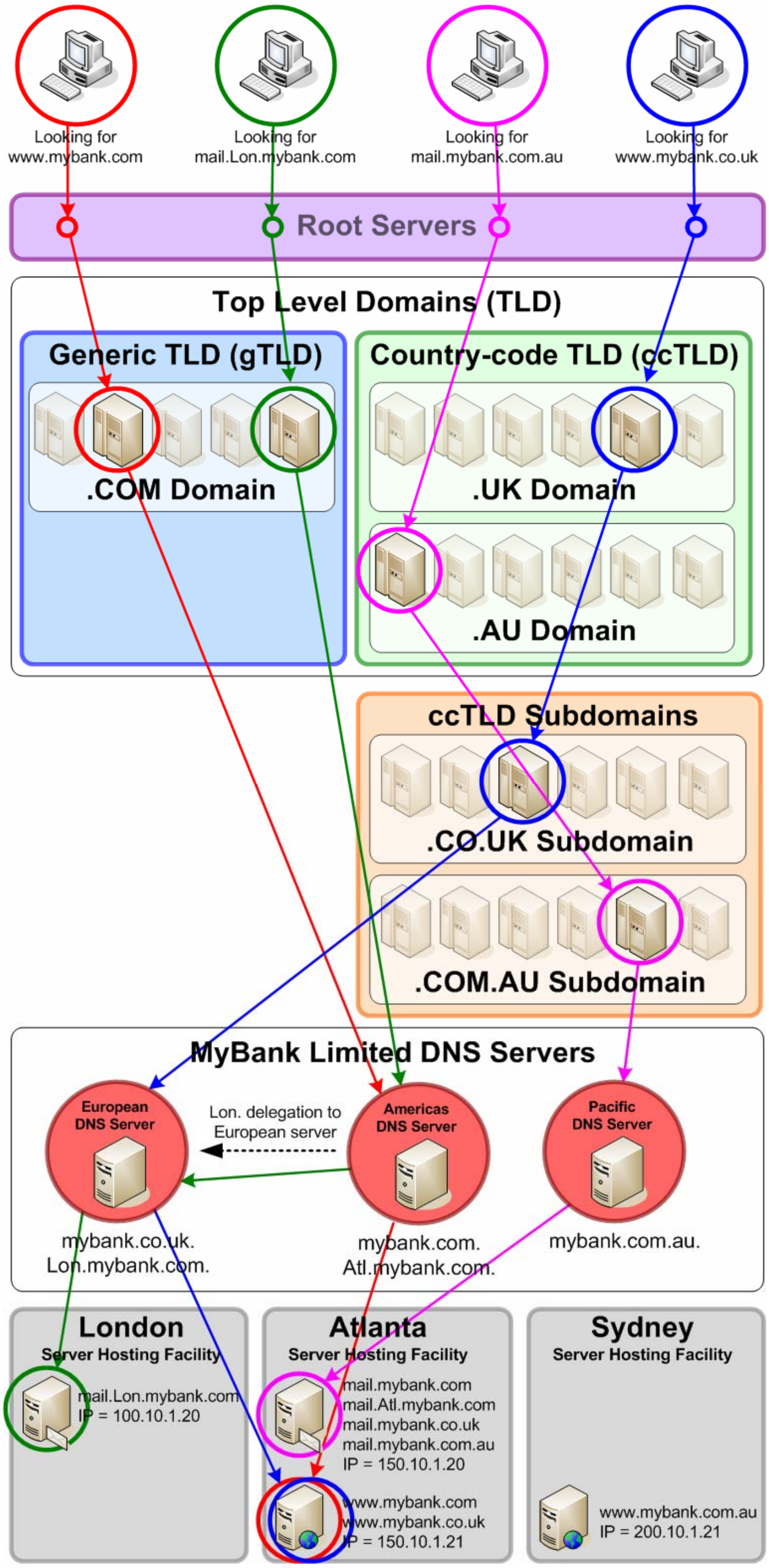


Figure 7: International customers looking up internationally hosted servers

2.2.5. Resolving an IP Address

While it is certainly possible for every Internet computer to carryout its own DNS lookup procedure each and every time, there are more efficient methods of obtaining the IP address information associated with a particular FQDN – methods that can help speed up resolution processes as well as reduce the volume of network traffic.

Within a corporate environment or dialup/broadband subscription ISP where there are many desktop computers accessing the Internet through a managed connection, instead of each customers computer making DNS queries directly (i.e. independently resolving through the Root Servers etc.), they are instead directed to use DNS servers provided by their organisation/ISP. These shared DNS servers are configured to provide DNS resolving services on behalf of their customers and may also temporarily cache previously requested domain name information.

Depending upon the configuration of the corporate/ISP DNS server, instead of attempting to resolve the IP address of a requested FQDN directly, it may in turn ask a larger or better positioned DNS server to resolve it first – just in case it has already cached the information. To illustrate this process, consider the following figure:

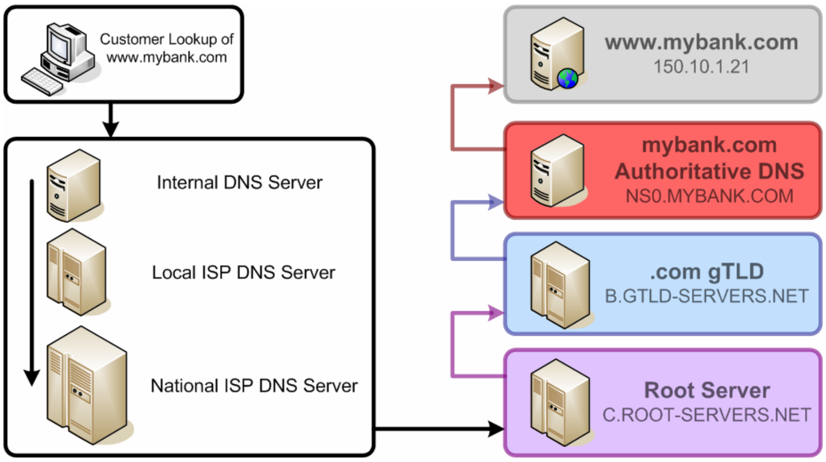


Figure 8: Resolving an IP address using non-authoritative DNS servers

In this scenario, the customer wishes to connect to www.mybank.com. However, the customers computer has not connected to this host before (or doesn't have the details cached in its own memory) and is thus forced to connect to an internal DNS server to find out this information. If the internal DNS server doesn't have this information cached, it may be configured to ask the local ISP's DNS server for the information. In some cases local ISP's may be subsidiaries to larger national ISP organisations and will be configured to ask their nominated national ISP DNS server for connection details of the www.mybank.com FQDN.

If the details are not present in these DNS servers, one of them (most likely the last one asked or so configured) will then resolve the details by querying a root server, then a .com gTLD server, eventually reaching an authoritative DNS server for the mybank.com domain. The mybank.com DNS server then informs the DNS server doing the resolving the location (IP address) of the www host.

Assuming that the national ISP DNS server did the asking, it would most likely cache the details (the length of time these details are cached would most likely be dictated by the TTL information provided by the authoritative mybank.com DNS server) and then pass the information back to the local ISP DNS server. The local ISP DNS server would also cache the information, and then provide the location details up to the internal DNS server (which would most likely also cache the details), before finally passing the details to the original customer computer that made the request for www.mybank.com. Armed with this IP address information, the customer's computer can now attempt to connect to www.mybank.com.

2.2.6. Caching Host and Domain Information

In the process of resolving a hosts IP address, a customer's desktop computer should query their local DNS server for the necessary information. If their local DNS server is not Authoritative for the domain of the host it is looking up, it should ask other servers to get an answer (as explained in the previous section - 2.2.5).

In the majority of cases, a customers computer will connect to the Internet via their corporate network or a local ISP and their default DNS server will also manage the domain (i.e. Authoritative) to which their computer belongs. For example, a customer may use BT ADSL Internet services in the UK and are automatically assigned the name host81-155-234-22.range81-155.btcentralplus.com to their dynamically assigned IP address 81.155.234.22.

If this default DNS server is required to provide all resolution services to each computer on its network, it is likely that the server will be busy fulfilling numerous lookup requests which may slow down its performance - especially if it is also authoritative for its own domain. To help reduce this lookup overhead, the DNS server may be configured to temporarily store any information it gets from other DNS servers inside its own database for a period of time stipulated by the answering (most likely authoritative) DNS server. The storage time is defined by the knowledgeable DNS server in the form of a "time to live" or TTL for short.

The mechanism of temporarily storing lookup information is called Caching, and it is designed to allow a DNS server to respond faster to multiple queries for the same domain or host information. DNS servers that operate in this manner are referred to as Caching DNS servers and are often configured to carryout recursive lookups. Operating in this way, the caching DNS server will first check its current database of lookups for domain or host information. If this information has not expired (checked by reviewing the TTL of the entry), the caching server will provide the requesting computer with this stored information. If the TTL has expired, or the DNS server does not have any knowledge of the domain or host, it would then conduct a recursive lookup to find the correct information. Once found, the caching DNS server would add the information to its database and pass the information back to the requesting computer.

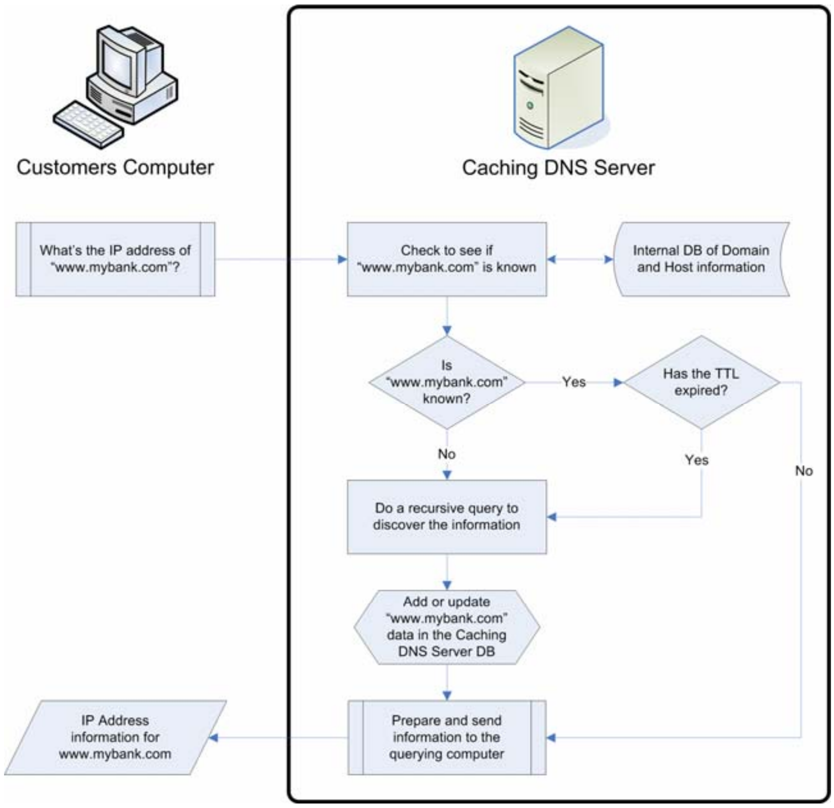

Customers Computer
Caching DNS Server
What's the IP address of "www.mybank.com"?
Check to see if "www.mybank.com" is known
Internal DB of Domain and Host information
Is "www.mybank.com" known?
Yes
Has the TTL expired?
No
Yes
No
Do a recursive query to discover the information
Add or update "www.mybank.com" data in the Caching DNS Server DB
IP Address information for www.mybank.com
Prepare and send information to the querying computer

Figure 9: How a caching DNS server answers a lookup request

The majority of Internet Service Providers (ISPs) use some kind of caching DNS server to support their customers and reduce network loads. By short-circuiting the normal domain or host resolution process, these servers can reduce the answer response time to the computers that use the service. However, because previous answers are stored until their TTL expires, it can take hours or days before any legitimate changes to the original data may filter its way to all areas of the Internet. A comparison of core functionality between a standard DNS server and a caching DNS server is as follows:

	DNS Server	DNS Cache Server
Availability	Should be able to respond to lookup queries from any computer on the Internet	Should only respond to lookup queries that originate from a "local" network
Types of query that it should answer	Non-recursive queries	Recursive queries
Records that it should attempt to resolve	Should only respond with data it is authoritative about	Should attempt to resolve any legitimate request

2.2.7. Local Lookup

As stated earlier, DNS evolved from locally stored reference files that contained mappings from a host name or alias, to a numeric IP address. Almost every popular operating system in use today still provides this kind of service, and may be configured to use the information contained in these locally stored files in preference to that contained on a remote DNS server.

All modern desktop UNIX distributions, Linux distributions and Microsoft Windows operating systems utilise a 'hosts' file. This host file contains the mapping information between a memorable alias and IP address, and is located in the /etc/hosts file on Linux systems and %Systemroot%\System32\Drivers\Etc\hosts on Microsoft Windows systems. An example of a 'hosts' file is as follows:

```
# Copyright (c) 1993-1999 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97     rhino.acme.com          # source server
#       38.25.63.10     x.acme.com              # x client host
127.0.0.1       localhost
192.168.10.5    printer
192.168.10.12   fileserver
192.168.10.12   fs.home
150.10.1.20     mail.lon.mybank.com
150.10.2.20     mail.mybank.co.uk
150.10.2.20     mail.atl.mybank.com
150.10.2.21     www.mybank.com
150.10.2.21     webserver
```

Depending upon the computers operating system, it may be configured to use 'host' files for host resolution in preference to remote DNS services. There are a number of reasons for using a local 'hosts' file first:

- Speed – reading the locally stored file for host naming information will generally be faster than requesting external network resources.
- Updateability – for small networks, or networks that are subject to frequent change, updating a local file may be easier than setting up or modifying a dedicated DNS server.

- Noise – some organisations choose to use hosts files on servers or other hosts to reduce the volume of network traffic which would normally be associated with repeated DNS queries.
- Shortcuts – in many environments, personnel who frequently have to connect to the same hosts may prefer to supply their own shortened or more memorable alias to a host. For example, instead of the name mail.lon.mybank.com, the user could add an entry to their 'hosts' file called "smtp" for that associated IP address.
- Overriding – in some cases it may be necessary to locally override any public/authoritative DNS resolution of a particular host. For instance, when testing a web application within a seperate hosting environment while the real one is still "live", modifying the local hosts 'hosts' file to point www.mybank.com to the IP address of the test server should prevent the user from testing the wrong server.

Other Host Resolution Services

As you would expect, throughout computer history there have been a number of alternative methods of resolving a hosts alias to an IP address. Many operating systems still provide backwards compatibility with these resolution services, and may try to resolve an unknown host name using them if their preferred method does not yield an authoritative result. These services and the order in which they are queried, is important, and can have a security significance.

Unix and Linux systems use a file (called 'hosts.conf') to define the resolution services the host may use, and in which order. For reference, these are typically (in order):

- DNS
- hosts – for lookups in /etc/hosts
- NIS – a legacy UNIX directory service (Network Information Service)

Microsoft Windows operating systems would typically try to resolve a host alias to IP address using a greater number of methods. For reference, these are typically (in order)

- Checks to see if the alias is the computers own name
- hosts – for lookups in %Systemroot%\System32\Drivers\Etc\hosts
- DNS
- WINS – a proprietary lookup protocol to Microsoft Operating Systems
- Network Broadcasts
- LMHOSTS – a file associated with the LAN Manager directory service

2.3. The "New DNS"

While conventional DNS services are responsible for the efficient resolution of host name into an Internet routable IP address, there are a number of additional services used by customers to discover hosts that contain specific information resources they are looking for. These services may be either online or built into the software a customer is using to access the Internet.

These "New DNS" services have been designed to make it easier for customers to find, and connect to, appropriate servers using minimal amounts of information. For simplicity, they can be divided into two key categories:

- "Autocompleters"
- Search Engines

2.3.1. "Autocompleters"

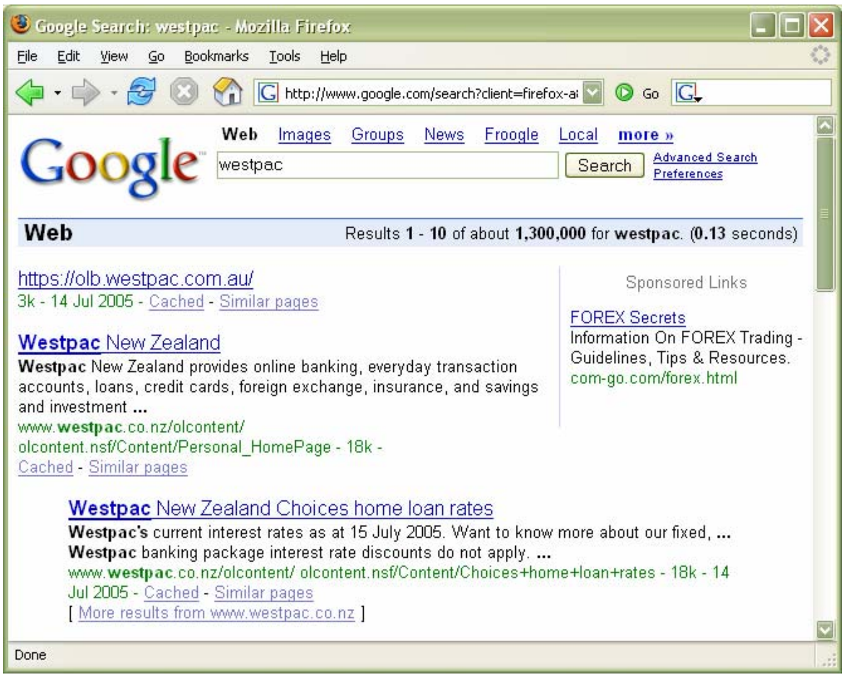
Most web browser implementations or Internet aware applications will try to "autocomplete" incomplete or unknown host names. For instance, typing the word "beer" into a web browsers' address bar will most likely result in one or more of the following actions:

- ★ The browser will try to resolve the host “beer” by see if the name is cached locally or listed in lookup files such as HOSTS or LMHOSTS.
- ★ It will append “http://” to the start of the name and try to connect to the host using the HTTP protocol.
- ★ It will append “www.” to the start of the name and “.com” to the end, and try to connect to the host “www.beer.com”
- ★ It will query the software vendors online database for advice about the best URL associated with this host name or “keyword”. In many cases organisations will pay to have keywords associated with their product or website.
- ★ It will submit a query to a default search engine and automatically follow the resultant link. Again, organisations may pay to ensure that their website is associated with the keyword.
- ★ It will submit a query to a nominated search engine and list all the options returned by the search engine.
- ★ It may just respond with “no such host”.

Whilst this process aids the discovery of *a* host name associated with the word “beer”, there is no guarantee that the customer will be directed to the same host name every time.

2.3.2. Search Engines

Search engines are an increasingly popular mechanism for resolving destination hosts or URLs associated with a particular organisation. In many cases, even though the customer knows the exact URL of the business service they wish to connect to, they find it easier to type in one or two keywords into their favourite online search engine (or browser toolbar) and follow the resultant link.



http://www.google.com/search?client=firefox-a

westpac

Web Results **1** - **10** of about **1,300,000** for **westpac**. (**0.13** seconds)

https://olb.westpac.com.au/
3k - 14 Jul 2005 - Cached - Similar pages

Westpac New Zealand
Westpac New Zealand provides online banking, everyday transaction accounts, loans, credit cards, foreign exchange, insurance, and savings and investment **...**
www.**westpac**.co.nz/olcontent/ olcontent.nsf/Content/Personal_HomePage - 18k - Cached - Similar pages

Westpac New Zealand Choices home loan rates
Westpac's current interest rates as at 15 July 2005. Want to know more about our fixed, **...**
Westpac banking package interest rate discounts do not apply. **...**
www.**westpac**.co.nz/olcontent/ olcontent.nsf/Content/Choices+home+loan+rates - 18k - 14 Jul 2005 - Cached - Similar pages
[More results from www.westpac.co.nz]

Sponsored Links

FOREX Secrets
Information On FOREX Trading - Guidelines, Tips & Resources.
com-go.com/forex.html

Figure 10: Google search for “westpac" to discover links to the New Zealand personal banking page

For example, instead of typing in the entire URL for the Westpac personal banking online service, a customer may find that it is faster to type “westpac” into Google (or a related toolbar search aid) and follow the second link on the page.

These search engines use a number of mechanisms to decide which responses are presented to the viewer, and in what order they are sorted. This process is commonly referred to as “page ranking”. The most common methods of deciding the page ranking of a response include:

- Whether the searched words (or phrase) appear in the name of the host or as part of a URL.
- How many times the searched word appears within a page or site, the order of the words, and how it is used within the structure of the page (e.g. a page title, a paragraph heading, within a sentence, e.g.).
- How many other websites or pages contain links to the page containing the searched words, and what their own page rankings are.
- The frequency of which other people have conducted the same or a similar search and followed a particular resultant link.
- The age the page has existed.
- “Tweaking” of keyword associations by the search engine company to reflect local language interpretations or regional locations of the request.
- Paid for services by the search engine company to ensure that a particular result will always appear at the top of the results list.

Section 3: **Pharming Attack Vectors**

3.1. Understanding the Attack Vectors

It should now be clear that there are a lot of background processes being executed each time a customer wishes to connect to a named host or online service. Each process relies upon an extraordinary number of systems and routines to function correctly, and different results are possible depending upon the physical location of the customer and the timing of their resolution request. Consequently, there exist a number of vectors through which a Pharmer could conduct their attack.

While there are indeed many ways in which the existing name resolution processes (and DNS in particular) can be attacked, not all are useful within a pharming attack. Since the ultimate goal of a Pharmer is to obtain personal information about a customer (ranging from website authentication details, through to complete identity theft), some attack vectors are more successful that others.

This section focuses upon the different attack vectors available to the Pharmer to conduct their attack and, where possible, the consequences of the attack.

3.1.1. Grouping Attack Targets

Before examining each individual attack vector available to the Pharmer, it is important to understand which groups of hosts or services are likely to come under attack. Each grouping typically has its own unique attack vectors and likelihoods of success. The following figure divides the host resolution process into five target groups:

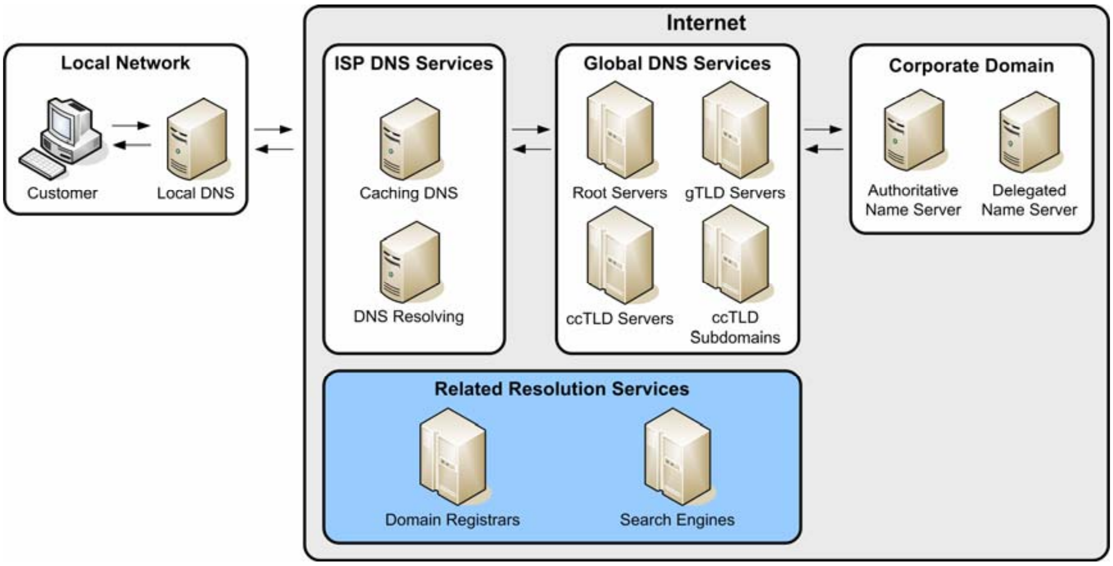


Figure 11: Host resolution services – attack target groupings

Further explanation of groupings:

- **Local Network** – The local network grouping includes the customer's host, the physical LAN, any proxy servers and egress firewalls. In addition, if the customer is located within a business environment, local DNS services may also be included.
- **ISP DNS Services** – This group includes all the DNS servers used by the customer, located on the Internet, used for DNS resolution. It includes ISP DNS servers that cache lookup results as well as and resolving services.
- **Global DNS Services** - This group includes all the globally managed services used as part of the resolving process to identify the authoritative name servers for a domain. It includes all Root and TLD servers.
- **Corporate Domain** – This group includes all the services typically owned by a corporate entity to do carryout the IP address resolution of named hosts. As such it

includes the authoritative name services for their domain, and any other final delegation processes.

- **Related Resolution Services** – This group includes services not directly related to the DNS lookup process, but which have a substantial effect on the resolution of hosts or online resources. As such, it includes the domain registration process and online search engines or portals.

3.1.2. Attack Classification

Using the groupings discussed above makes for an easier classification of vectors likely to be used during a Pharming attack. The following table categorises attack vectors used by Pharmers and their association with the five different target groupings.

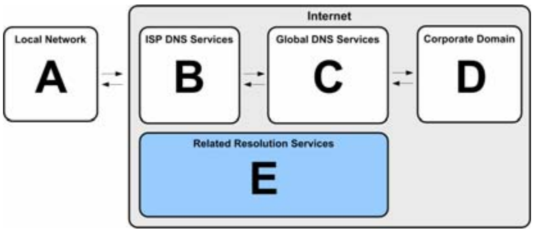


Figure 12: Key to attack targets

Attack Vector	Target Group				
	A	B	C	D	E
Human Factors					
The insider edge	Yes	Yes	Pos.	Yes	
Local Host and Local Network Attacks					
Modification of lookup processes	Yes				
Traffic observation and modification	Yes	Pos.			
Man-in-the-middle attacks	Yes				
Domain Registration Attacks					
Domain hijacking					Yes
Similar domain name registrations					Yes
Botnet name server registration					Yes
Domain Configuration Attacks					
DNS wildcards				Yes	
Poorly managed DNS servers		Yes		Yes	
DNS Spoofing					
DNS cache poisoning	Yes	Yes			
DNS ID spoofing with sniffing	Yes				
DNS ID spoofing without sniffing	Pos.	Yes			
The birthday attack	Pos.	Yes			
The "New DNS" Attacks					
Page rank escalation					Yes

3.2. Human Factors

The automated process of resolving a named host to a particular IP address is wholly dependant upon the careful management and configuration of the hosts by technical administrators. These administrators must often manually edit individual DNS server configuration files in order to tune and optimise the services under their care, as well as manage the important host data the services contain.

Because manual intervention is a necessary part of running the DNS system, there are ample opportunities for human factors to affect the security and integrity of the information it contains. The complexity of the DNS system and the interplay between various globally load-balanced servers, coupled with regional optimizations and content delivery options, means that any flaws in the configuration of a single DNS server or name server can often be difficult to identify. Therefore, should a DNS system administrator choose to, he could most likely make malicious alterations to the information about a particular organisations host which may not be spotted for some time – if ever.

For instance, in January 2005 the DNS address for the domain "panix.com" was changed, and the ownership of the New York State ISP was changed from New York to Australia. As part of the attack, requests to reach the panix.com web services were directed to the United Kingdom while email was redirected to Canada.

3.2.1. The Insider Edge

In the past most of the unauthorised modifications made to DNS host entries have been of nuisance value, and rarely for financial gain. However, in the future, the potential profit to be gained by an attacker undertaking a pharming scam is expected to increase the probability of DNS system administrators making temporary changes for cash. With organised crime now taking a keen interest in online identity theft opportunities, the insider threat has never been higher.

Depending upon the location of the conspiring system administrator and the DNS services they have access to, the following risks are present:

- ★ Internal network DNS servers – Within these networks the local system administrator would typically have no difficulty in modifying any static host records, adding custom entries, or modifying cached information without detection. These modifications would only affect the customers operating within the internal network.
- ★ ISP DNS servers – The opportunity for a rogue system administrator to make short-term modifications to DNS entries and conduct pharming attacks is highest within this network level. Since ISPs typically operate several DNS servers simultaneously to cater for different network subscriber demands, modifications to a single DNS server are unlikely to be detected. Any customers relying upon the corrupted DNS server would receive incorrect IP address information and be directed to a different host. Given the high volume of traffic using an ISP's DNS services, even a single malicious entry is likely to yield substantial numbers succumbing to the pharming attack.
- ★ Corporate name servers – Modifications to resolution information within the authoritative name servers for an organisations domain will yield maximum impact for the Pharmer, but also the highest probability of being detected. Malicious changes to entries within the authoritative name server(s) will result in all network traffic being directed to an alternative IP address. However, unless there is adequate security logging at the name server, short-period (e.g. a few hours each day) alterations are unlikely to be detected by the parent organisation.
- ★ Global DNS Servers – It is possible that system operators could modify values and have a widespread affect on the DNS resolution process. However changes at this level are likely to be noticed fairly rapidly.

3.3. Local Host and Local Network Attacks

Over the last few years there has been a substantial increase in attacks that focus upon gaining control of desktop systems – be that a customers home PC or a corporate workstation. The delivery mechanism chosen for the attack is often related to the type of control over the computer that the attacker wishes to achieve. Since the ultimate goal of the Pharmer is to seal customer identity credentials, there are a number of unique attack vectors focusing upon the local host or LAN that go beyond those previously discussed in "The Phishing Guide".

3.3.1. Modification of Lookup Processes

Each computer used directly by the Customer must use a predetermined routine to resolve a host name to an Internet routable IP address. As discussed in the earlier section “Local Lookup” (2.2.7), there are a number of local methods that may serve as alternatives to using standard DNS servers. If the attacker is able to gain the ability to modify local lookup preferences, or the files they rely upon, it is possible to conduct a pharming attack.

HOSTS file modification

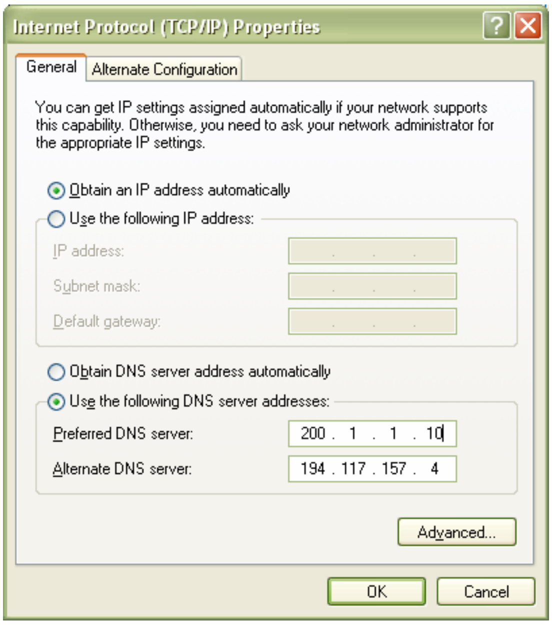
Since the most common desktop operating systems are based upon the Microsoft Windows platform, many of the successful pharming attacks focus upon modifying the operating systems HOSTS file. With access to this file, the Pharmer is able to add extra entries that will divert the customer’s traffic to a different IP address. For instance, by exploiting a vulnerability in the customers web browser software (typically the customer would have browsed an “infected” webpage that contained specific exploit code waiting to trap any person who visited the page and had a vulnerable version of the browser software), the Pharmer may be able to overwrite the existing HOSTS file with a specially crafted one such as the following:

```
# Pharming Hosts file
200.1.1.10      www.mybank.com
200.1.1.10      mail.mybank.com
200.1.1.10      www.hotmail.com
200.1.1.10      passport.microsoft.com
200.1.1.10      login.passport.net
200.1.1.10      webmail.yahoo.com
200.1.1.10      www.hushmail.com
200.1.1.10      mail.google.com
200.1.1.10      www.google.com
```

In this example, the Pharmer knows that the HOSTS file is used in preference to querying external DNS services and has pointed all the above host names to an IP address he controls. Typically the Pharmer would be running a web server with a number of virtual sites that look like each real site. The Pharmer can choose to just capture the customers login credentials as they enter them into the fake sites and generate an error afterwards, or may choose to transparently proxy the requests to the real servers and capture further confidential details as the customer “uses” the real site.

DNS Network Settings

If the Pharmer has control of the local network hosts (e.g. is a corporate network administrator, runs an Internet café, or can pay an insider to do it for them) it is a simple process to modify the network settings of the computer to point all DNS queries to a DNS server that the Pharmer controls.



You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

● Obtain an IP address automatically
○ Use the following IP address:
IP address: . . .
Subnet mask: . . .
Default gateway: . . .

○ Obtain DNS server address automatically
● Use the following DNS server addresses:
Preferred DNS server: 200 . 1 . 1 . 10
Alternate DNS server: 194 . 117 . 157 . 4

Figure 13: Modification of local host DNS server preferences.

3.3.2. Traffic Observation and Modification

With access to the local network, the Pharmer can typically directly observe and modify the destinations of network traffic. While the implications of network sniffing or rogue proxies are obvious, the ability to take control of computer configurations through initialisation services such as DHCP and WPAD are less so.

Rogue DHCP servers

For many networked environments, DHCP is typically used to automatically assign IP addresses and routing information for each computer host as it starts up. These computers must also renew or update the DHCP information from time to time (usually configured centrally by a network administrator). It is possible for an attacker to install a rogue DHCP on a particular network segment and have the local computers use information from it in preference to a centralised server located on a different network segment (mostly due to the speed of response).

By controlling the DHCP settings of a computer, an attacker can state which DNS servers must be used by the customer's computer. If the attacker also has control of a remote DNS server (or has installed his own DNS server somewhere else) he can also provide incorrect host resolution information and direct the customer to hosts of his choice.

In addition, DHCP as implemented in Microsoft operating systems also allows for the definition of a WPAD location.

Rogue WPAD services

The Web Proxy Automatic Discovery (WPAD) service allows web browsers (and other related software) to use a variety of methods to automatically locate suitable proxy services for their traffic. The WPAD service relies upon a number of well-known network protocols to identify and register proxies with computers configured to use the service. Since many popular software products are configured by default to use the service, a rogue WPAD server or suitably constructed entries in the local DNS server can be used by a Pharmer to redirect network traffic to a proxy server of their choice – thereby carrying out a man-in-the-middle attack.

3.3.3. Man-in-the-middle Attacks

Man-in-the-middle attacks often form a key component of a sophisticated Phishing or Pharming attack. With access to a customer's local network, this attack delivery platform is much simpler and often harder to detect. For more discussion about man-in-the-middle attacks, readers are referred to section 2.3.1 of "The Phishing Guide".

3.4. Domain Registration Attacks

Domain registration attacks abuse the way in which a domain may be registered with a registrar. The most common vectors for attack are:

- Domain Hijacking
- Similar domain name registrations
- Botnet name server registrations

3.4.1. Domain Hijacking

In order for an organisation to make use of a domain name, they must first register it. This process is done through various domain registration authorities; typically by paying a small fee to the registration authority to maintain ownership of the domain for a set number of years (typically 1-3 years). Domain Hijacking is the process by which a domain with a lapsed registration is purchased by another person and is then used for some other purpose.

Registration information is managed by various Internet registrars, and can be queried using several tools (such as whois) and other related online services. For instance, querying the registrar for information about the TECHNICALINFO.NET domain, we retrieve the following information:

```
Domain Name: TECHNICALINFO.NET
Registrar: MELBOURNE IT, LTD. D/B/A INTERNET NAMES WORLDWIDE
Whois Server: whois.melbourneit.com
Referral URL: http://www.melbourneit.com
Name Server: NS0.PHASE8.NET
Name Server: NS1.PHASE8.NET
Name Server: NS2.PHASE8.NET
Status: REGISTRAR-LOCK
Updated Date: 14-dec-2004
Creation Date: 20-jun-2002
Expiration Date: 20-jun-2006
```

It is important to note that, in the above example, the ownership of TECHNICALINFO.NET will expire on the 20th June 2006. If the current owner does not renew his registration by that date, anyone could purchase the domain and take ownership.

By 'hijacking' an existing domain, as opposed to registering a new domain, the new owner can take advantage of any existing links to it – thereby guaranteeing a number of "backlinks" and associated traffic. Domain hijacking an increasingly popular mechanism for advertisers and other organisations that generate revenue from customers connecting to their sites.

In a Pharming attack, the attacker would seek to take ownership of the domain as soon as the current owner neglects to re-register the domain. With ownership, the Pharmer would construct a new website (and other related Internet services such as email) to replicate the earlier version and fool any customers who connect to the site.

Alternatively, the Pharmer may choose to utilise a heavily trafficked site (not associated with an organisation they plan to target) to provide links (hidden or otherwise) to an alternative site under their control and increase its search engine rankings. This attack vector is explained fully in a later section.

3.4.2. Similar Domain Names

Perhaps one of the simplest attack vectors of all, the Pharmer registers multiple spelling and key mashing (e.g. "fat fingers" hitting neighbouring keys simultaneously) permutations of the target host name hoping that a customer will mistype it. If a customer does mistype the host name, instead of getting a "no such host" message or connecting to a host that is clearly not their desired destination, the attacker has created a fake version of the site to fool the customer and steal their credentials.

The registration of similar domain names has been abused for many years, but the nature of the attack was often somewhat different. Many adult entertainment or advertising sites make use of alternative domain name registrations to capture the attention of a potential customer or generate site traffic. A past example includes www.whitehouse.com, which was once a porn site and obviously not affiliated with www.whitehouse.gov – a US government information site.

More recently, attackers have made use of key mashing permutations of popular websites to direct customers to malicious websites. For example, in April 2005 an attacker registered googkle.com and msmn.com for the purpose of secretly infecting the computers of users of Google and MSN who has mistyped their host names with Spyware, Adware, and other malicious software.

It is expected that this particular attack vector will become increasingly popular as a mechanism of infecting customer computers with malicious software and stealing personal authentication information.

3.4.3. Botnet Name Server Registration

The domain registration process allows the registrant to list the authoritative servers responsible for managing the IP address lookups of the hosts within that domain. It is normally recommended that at least two name servers be listed (a primary and a backup) and that they be located on different network segments to help cope with network resiliency issues.

Botnets have been used in phishing attacks in the past – mainly to host multiple copies of the faked website at several IP addresses. As each host or IP address is closed down by the owner ISP, the Phisher just modifies his DNS to point to alternative location. If the ISP has control of the DNS entry (i.e. the Phisher is using an ISP's DNS server as their authoritative domain server), it is a simple process for the ISP to remove the entry (or point to the real website) and effectively close down access to all the fake websites in one go.

While it is recommended that at least two name servers be listed, the registrant can choose to list many more. This ability to list multiple entries can be abused during a pharming attack and, when combined with an established Botnet, can be very difficult to shut down once identified.

By ensuring that the multiple name servers registered by the Pharmer are spread across several ISP's, no single ISP can shut down the DNS service. Instead, the organisation being targeted by the attack must attempt to deal with the registrar of the malicious domain to close it down. Unfortunately many registrars do not have formal procedures for dealing with this kind of request.

3.5. Domain Configuration Attacks

3.5.1. DNS Wildcards

DNS Wildcards are special entries within a DNS configuration file for handling "catchall" name resolutions. For instance, an organisation may have two internet hosts – www.mybank.com for web traffic and mail.mybank.com for email – but wish to make use of more intuitive host names for their customers at a later date. Instead of continuously updating their DNS configuration, they may add a pair of wildcard entries to direct network traffic to these two hosts that were destined for other host names.

For example, the authoritative DNS server for the domain mybank.com may be configured to have the following entries:

```
www.mybank.com    IN    A             150.10.1.21
mail.mybank.com   IN    A             150.10.1.20
mybank.com        IN    A             150.10.1.21
mybank.com        IN    MX    10      mail.mybank.com
*.mybank.com      IN    MX    10      mail.mybank.com
*.lon.mybank.com  IN    A             150.10.1.21
```

Here we see that the * wildcard entries are designed to do the following:

- Direct all emails destined for [something].mybank.com to the mail server mail.mybank.com. For example, this would handle emails addressed to security@newyork.mybank.com and gunter@foo.bar.london.mybank.com.
- Direct all connections to hosts with names ending with lon.mybank.com to the IP address 150.10.1.21. For example, this would direct customers requesting http://customergateway.lon.mybank.com to the same host IP address as www.mybank.com.

In the past Phishers have spoofed email source addresses of organisations that did not have DNS wildcard entries for their mail servers (i.e. MX records) so that, should a recipient of one of their fake emails attempt to reply to it, the response would never be received/intercepted by the organisation and thus have a lower likelihood of discovering that their organisation was an unwitting participant in a phishing attack.

Pharmers (and many spammers) make use of DNS wildcard entries to obfuscate the true destination host of their attack. For example:

- If the Pharmer owns the top level domain (e.g. "pharmer.com") he may use a host name http://www.mybank.com.Login.html.134534534.pharmer.com/
- The Pharmer may abuse common link sites to redirect the victim to a server of their choice. For instance, in March 2005 an attacker abused the DNS wildcard configuration of a third-party redirection service (Kickme.to) to target Barclays banking customers with URL's such as:
    - http://barclays.co.uk|YJ3EMOHOqljQ8J5oW2ZKyTaRMQOahSWaxTrFTEQK9l9VVQj6jDtyq10d24r2h0bijh2

  - http://barclays.co.uk|34fdcb4rvdnp9phxbahhvbs6l56a2uyx%2edivxmovies%2ea%74/41pvaw3/

- Spammers often use DNS wildcard entries to embed unique tracking information within the host name to verify real email accounts and bypass anti-spam filtering software.

3.5.2. Poorly Managed DNS Servers

While it is true that new vulnerabilities within core DNS software are being discovered continuously, there is also a parallel stream of patches and security fixes being issued by the various vendors to correct them. If a DNS server is being managed correctly, these patches and updates will be installed shortly after being made available – thereby limiting the window of opportunity for an attacker seeking to exploit the new vulnerability.

Poorly managed DNS servers tend to be several patch cycles behind and tend to suffer from poor configurations and lax authentication controls. This means that an attacker can often easily gain control of the DNS server. Depending upon the role of the DNS server (e.g. ISP-level DNS caching, DNS resolving, corporate name server, etc.), the attacker can use the compromised DNS as part of a successful Pharming attack as if they were an insider (see section 3.2.1).

3.6. DNS Spoofing

A DNS spoofing attack can be defined as the successful insertion of incorrect resolution information by a host that has no authority to provide that information. It may be conducted using a number of techniques ranging from social engineering through to exploitation of vulnerabilities within the DNS server software itself. Using these techniques, an attacker may insert IP address information that will redirect a customer from a legitimate website or mail server to one under the attacker's control – thereby capturing customer information through common man-in-the-middle mechanisms.

According to the most recent "Domain Health Survey" (Feb 2003), a third of all DNS servers on the Internet are vulnerable to spoofing.

The Attack

Operating normally, a customer can expect to query their DNS server to discover the IP address of the named host they wish to connect to. The following diagram reflects this process.

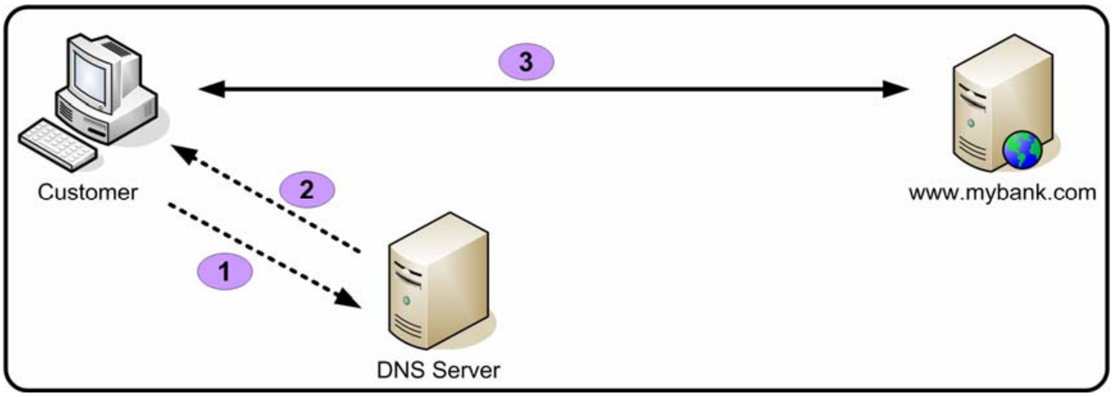


Figure 14: The normal DNS resolution process

1. The customer queries the DNS server – "What is the IP address of www.mybank.com?"
2. The DNS responds to the customer query with "The IP address of www.mybank.com is 150.10.1.21"
3. The Customer then connects to the host at 150.10.1.21 – expecting it to be www.mybank.com.

However, with a successful DNS spoofing attack, the process has been altered. The following diagram reflects this process.

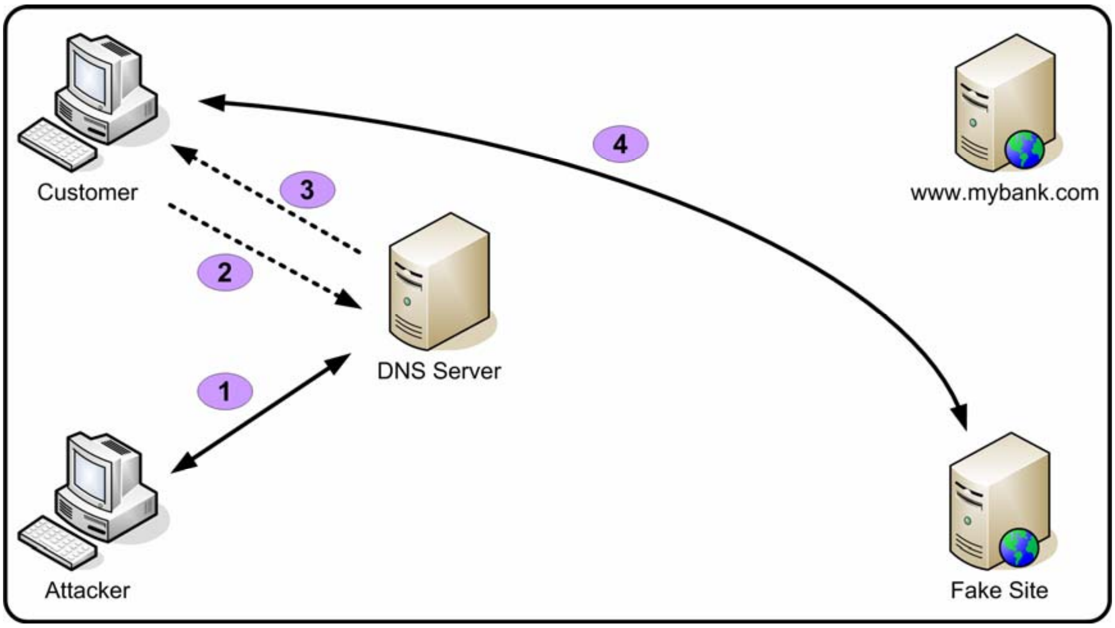


Figure 15: The DNS resolution process having fallen victim to a DNS spoofing attack

1. The attacker targets the DNS service used by the customer and adds/alters the entry for www.mybank.com – changing the stored IP address from 150.10.1.21 to the attackers fake site IP address (200.1.1.10).
2. The customer queries the DNS server – “What is the IP address of www.mybank.com?”
3. The DNS responds to the customer query with “The IP address of www.mybank.com is 200.1.1.10” – not the real IP address.
4. The Customer then connects to the host at 200.1.1.10 – expecting it to be www.mybank.com, but in fact reaching the attackers fake site.

Use of Botnets

Botnets may be used within many of DNS spoofing attacks as a force multiplier in the following ways:

- ★ As a denial of service agent against an authoritative name server to cause it to respond slowly (or not at all) to resolver requests for valid host IP address information.
- ★ As a delivery mechanism for fake UDP responses from the spoofed IP address of the authoritative name server.

3.6.1. DNS Cache Poisoning

One attack vector for DNS spoofing is through cache poisoning. In this attack the attacker abuses caching vulnerabilities within the DNS server to add multiple resolution entries for hosts not originally asked for and is not authorised to provide. While most new DNS service implementations are not vulnerable to cache poisoning, there are still a large number of vulnerable DNS servers that are.

The process in which a DNS server may have its cache poisoned can be explained in the following diagram and walkthrough.

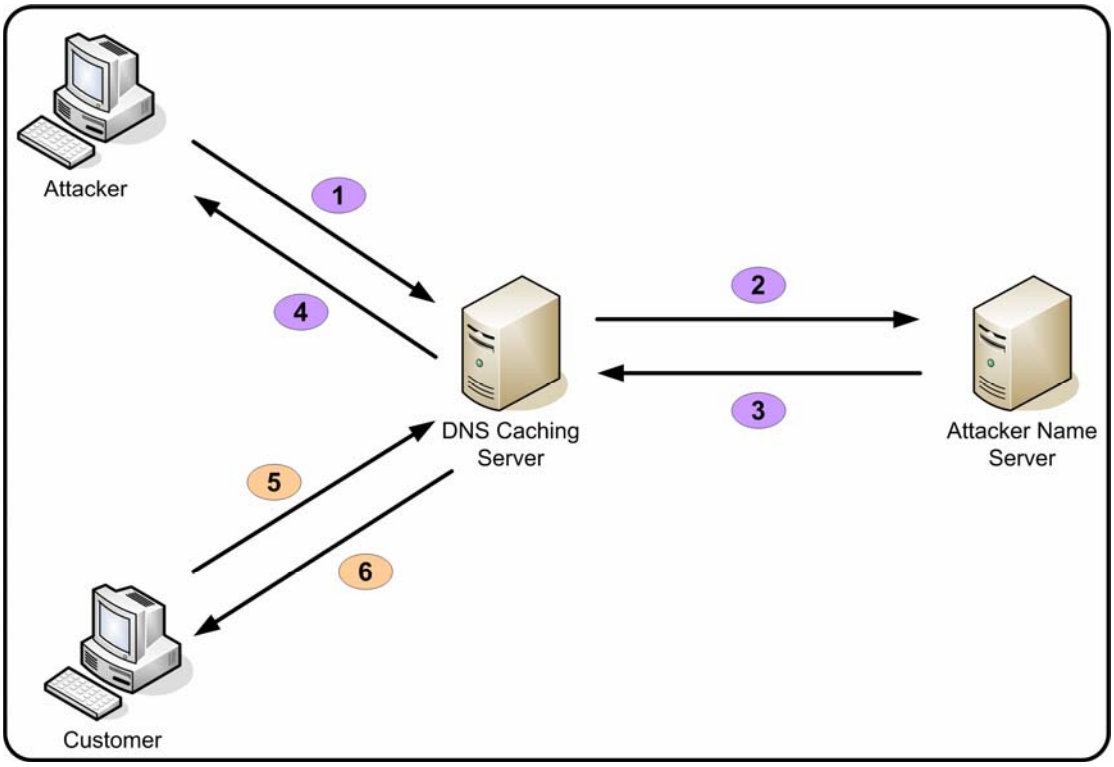


Figure 16: The DNS cache poisoning process

1. The attacker queries the DNS server for the IP address for a host that is managed by a name server owned by the attacker – "What is the IP address of www.attackerowned.com?"
2. The DNS Caching server does not have a cached entry for www.attackerowned.com and must resolve the IP address by querying the authoritative name server for the attackerowned.com domain. This authoritative name server belongs to the attacker.
3. The attackers name server informs the DNS caching server that the IP address of www.attackerowned.com is 200.1.1.10. In addition, the attackers name server also includes additional (faked) resolution records such as:
    a. www.mybank.com is 200.1.1.11
    b. mail.mybank.com is 200.1.1.11
    c. secure.mybank.com is 200.1.1.11
4. The DNS caching server responds to the attacker's original query with – "The IP address of www.attackerowned.com is 200.1.1.10." This result, along with the extra resolution records, is cached by the DNS server for a period equivalent to the TTL supplied by the attackers name server.
5. At a later date, an ordinary customer who also uses this DNS caching server queries it for the IP address of www.mybank.com – "What is the IP address of www.mybank.com?"
6. The corrupted DNS caching server responds to the customer query by supplied the previously cached (and fake) answer – "The IP address of www.mybank.com is 200.1.1.11" – instead of the real 150.10.1.21 address.

For instance, In July 1997 Eugene Kashpureff of AlterNIC used a program to "poison" the caches of major name servers around the world. This caused traffic originally destined for www.internic.net's address to go to the IP address of the AlterNIC web server. No attempt was made to disguise the attack, and customers who tried to reach www.internic.net were confronted with the AlterNIC website.

3.6.2. DNS ID Spoofing with Sniffing

DNS lookup queries by Customer hosts rely upon the UDP protocol (an important Internet protocol that, unlike TCP, does not use any form of handshaking) to request and obtain resolution information. Each UDP-based DNS query originated from the customers computer will also be assigned a unique identifier (ID for short) to help manage multiple lookup responses. Any queried DNS sever is supposed to include the same query ID as the request. If the ID supplied is not the same as that of the originating request, the customers computer should ignore the response.

DNS ID spoofing is an attack vector which focuses upon providing incorrect or malicious DNS resolution information to customer requests after having observed the ID of their request. To be successful, the attacker must observe the customers request (most often achieved by sniffing the network traffic) and be capable of constructing a spoofed response faster than the DNS server can supply the legitimate answer.

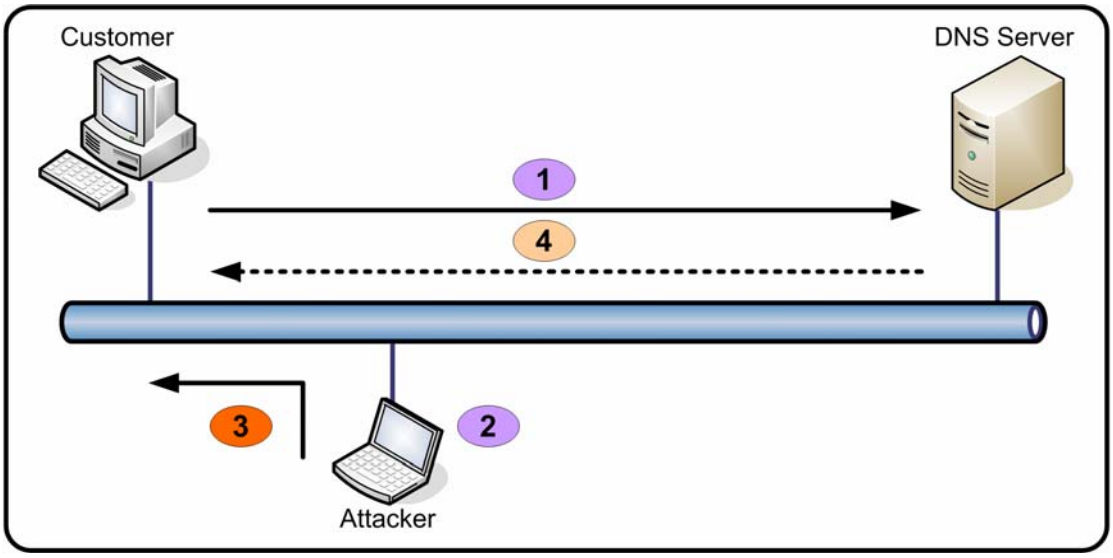


Figure 17: The DNS ID spoofing process

In the figure above, the process of DNS ID spoofing is as follows:

1. The customer sends a UDP-based request to the DNS server – "What is the IP address of www.mybank.com?" – with an ID of 117.
2. The attacker has positioned a laptop on the network to sniff all network traffic and respond to DNS queries.
3. Having identified a DNS request for www.mybank.com with an ID of 117, the attackers laptop automatically responds with "The IP address of www.mybank.com is 200.10.1.11" using the same ID.
4. A few fractions of a second later, the real DNS server sends its response "The IP address of www.mybank.com is 150.10.1.21", but is ignored by the customer's computer since it has already received a response from the attacker's machine.

3.6.3. DNS ID Spoofing without Sniffing

While DNS ID spoofing is a relatively simple process if the attacker is able to monitor a customer's network traffic for DNS lookup requests, the attack is restricted to environments where the attacker can gain physical network access to a network segment shared with the customer. To overcome this limitation, an attacker may use a combination of techniques to achieve ID spoofing without relying upon network sniffing.

A limitation of the UDP-based DNS lookup process is that the ID is coded to 2 bytes, meaning that there are only 65535 possible values. Therefore, for an attacker to succeed in carrying out the previous attack without sniffing, he would have to either guess the correct ID or rapidly produce 65535 spoofed responses before the DNS server could respond.

An additional problem is to know exactly when to launch the attack. This problem can be overcome by using a process similar to the one discussed previously on DNS cache poisoning – the attacker actually launches the initial lookup request and the DNS caching server (the victim in this attack) must query the authoritative name server for the IP address information.

Early versions of a popular DNS software implementation suffered from a security flaw that resulting in all DNS transaction ID's being non-random – instead they were sequential. This of course made the "guessing" of a requests ID a simple process. Following a CERT advisory (CA-1997-22), organisations using this software were advised to upgrade to a new version that implemented random transaction ID's.

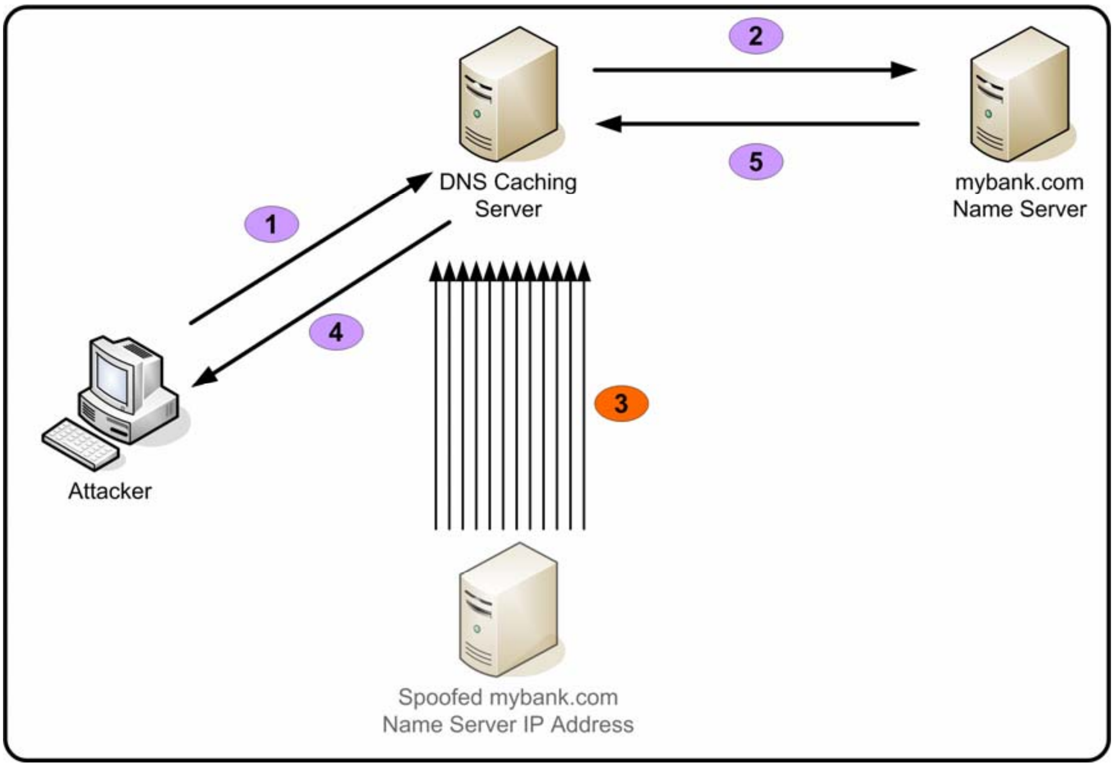


Figure 18: The DNS ID spoofing attack not relying on sniffing

In the figure above, the process of DNS ID spoofing without relying on network sniffing is as follows:

1. The attacker makes a request to the DNS server – "What is the IP address of www.mybank.com?"
2. The DNS server must query the mybank.com authoritative name server (150.10.1.2) for the IP address of www.mybank.com and has assigned a DNS ID of 45889 to this request.
3. While the DNS server is attempting to resolve the IP address of www.mybank.com and awaiting a response from the authoritative name server, the attacker has launched a flood of UDP responses from a host under their control using the spoofed IP address of the mybank.com name server (150.10.1.2). Each spoofed response has a different DNS ID and states that the IP address of www.mybank.com is 200.10.1.11.
   Note: The UDP packets must be sent to a specific port as well. While this in theory must also be guessed, in practice many DNS servers use the same source port for their queries. Therefore the attacker would, in advance of this spoofing attack, have constructed their own authoritative name server and observed the DNS source port after having queried the DNS server for the IP address of a host listed on the attackers name server.
4. If the attacker has succeeded (i.e. managed to guess the DNS response with the correct ID using the spoofed name server IP address to the DNS server before the

real mybank.com name server did) he will receive a response from the DNS caching server stating "The IP address of www.mybank.com is 200.10.1.11". This IP address information is then cached for the stipulated TTL and will be supplied by the DNS server in response to any later customer requests.

5. The mybank.com authoritative name server responds with the correct IP address for www.mybank.com (150.10.1.21), but is ignored by the requesting DNS server if it has already received a "valid" spoofed response from the attackers system.

3.6.4. The Birthday Attack

Closely related to the previous attack vector, a "Birthday Attack" exploits a weakness discovered in 2002 relating to the fact that the most popular DNS implementation (BIND) would send multiple simultaneous recursive queries for the same IP address (now fixed in the latest versions of the software). 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.

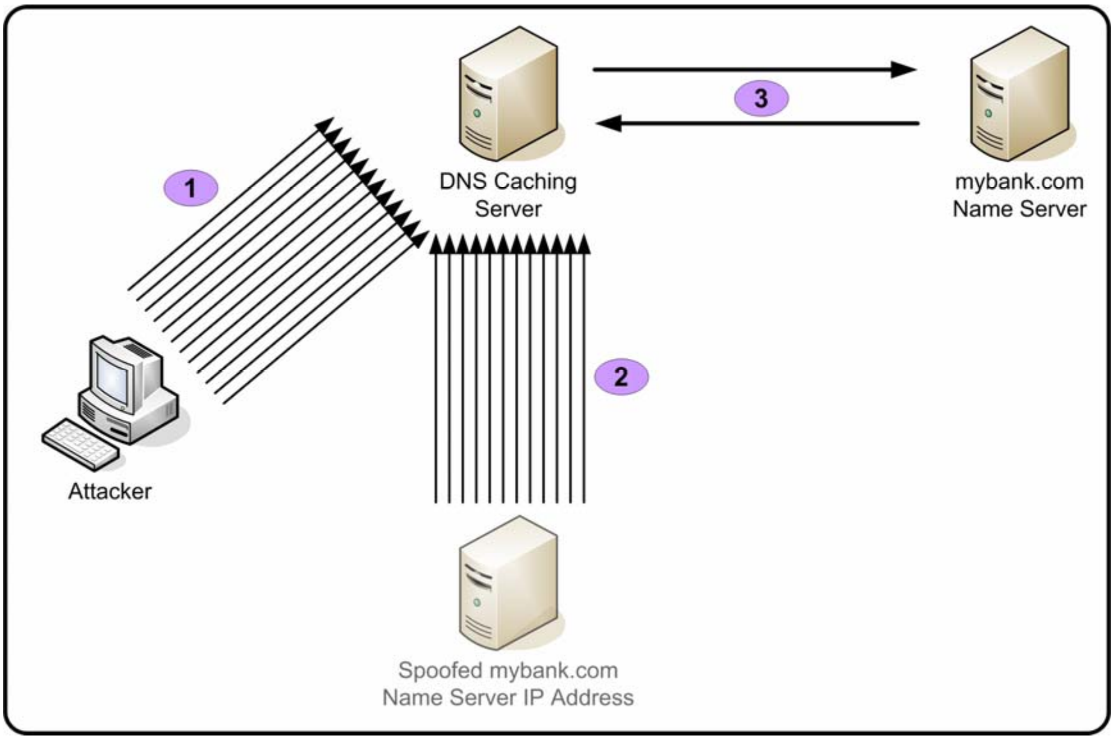


Figure 19: The DNS Birthday Attack

In the figure above, the birthday attack is carried out as follows:

1. The attacker launches repeated requests to the DNS caching server asking "What is the IP address of www.mybank.com" as fast as possible.
2. Simultaneously, the attacker also sends repeated spoofed responses using different DNS transaction ID's stating that "The IP address of www.mybank.com is 200.10.1.11".
3. For each request from the attacker in (1), the DNS server tries to resolve the IP address for www.mybank.com by querying the authoritative mybank.com name server – typically using a different DNS transaction ID for each request. Based upon the mathematical properties of the Birthday Paradox, there is a higher probability that the attacker can "guess" a correct DNS transaction ID (thereby "answering" the DNS servers query) faster than the real name server can respond.

To further increase the odds of the attacker supplying a correct DNS transaction ID with the spoofed message, the attacker could target the authoritative name server with other requests or denial of service techniques to slow down its response to the DNS caching server.

Why does the attack work?

The Birthday Attack is named after a mathematical result that establishes that the probability that two or more people in a group of 23 share the same birthday is greater than 50% - the so called "Birthday Paradox". This mathematical principle can be applied to pseudo-random number generation; which in this case is the process for generating DNS transaction ID's.

During a conventional DNS ID spoofing attack (section 3.6.3) the attacker would send n spoofed responses for a single query – resulting in a probability of success of $n/65535$. During a DNS Birthday Attack the attacker sends n spoofed replies for n queries for which the probability of success (P) becomes:

$$P = 1 - \left(1 - \frac{1}{t}\right)^{\frac{n \times (n-1)}{2}}$$

Plotting this equation, where t represents the maximum number range of DNS transaction ID's (65535), we observe the following results for the first 1000 values of n.

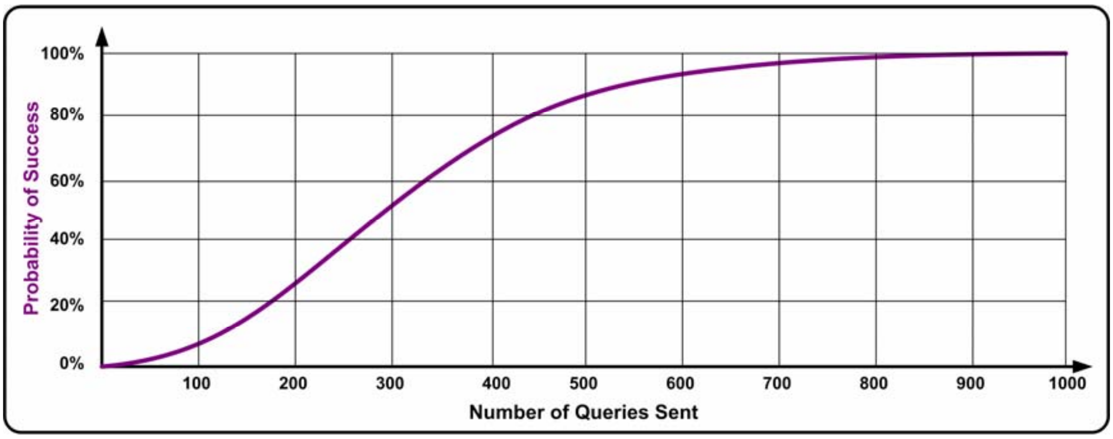


Figure 20: The DNS Birthday Attack probability of success graph

From this quick analysis, we can see that the probability of success reaches 50% when n is approximately 300 and 99% as n approaches 800. Using a conventional DNS ID spoofing attack (section 3.6.3), 300 spoofed responses would have only yielded less than 0.5% success.

3.7. The "New DNS" Attacks

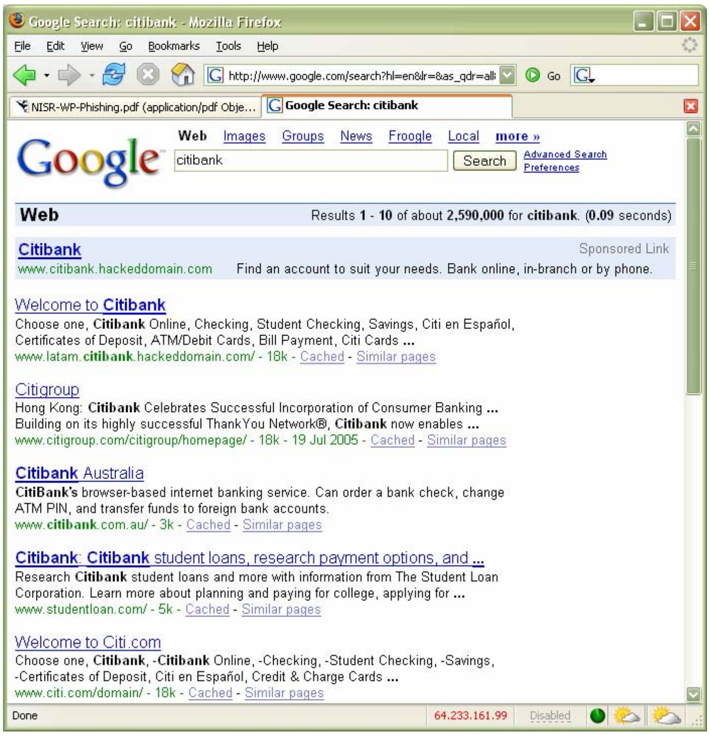
The growing popularity of the "new DNS" coupled with the way customers rely upon these services to locate frequently accessed Internet resources represents an ideal target population – some would say "low hanging fruit" - for Pharmers. With some forward thinking and relatively little effort, a Pharmer can corrupt the names associations these services offer at regional or global levels and target specific customer bases.

The marketing opportunities that many popular search engines provide, means that Pharmers can purchase "sponsored links" or similar services which will place their hyperlinked resources (i.e. link to their faked website) at the top of a customers search page response. Closely related to the exploitation of banner advertising by Phishers (see section 2.2.2 – Web-based Delivery – of "The Phishing Guide"), Pharmers may exploit the validation processes of some search engine providers that are not as rigorous as others, and can provide links to their fake sites using keywords or phrases normally associated with the targeted organisation.

3.7.1. Page Rank Escalation

With prior planning, a Pharmer can seek to increase their search engine page ranking by abusing the way that provider actually calculates the ranking. By taking advantage of the page ranking system the Pharmers goal is to get their fraudulent link (and extracted page text) to appear in the place of where a customer would normally expect to find the real link, preferably first on the list.

For Example, using the Google search engine, customers who type in “Citibank” would normally expect to see several links to Citibank. However, by exploiting the page weightings explained in section 2.3.2, it may be possible for the Pharmer to cause the following to appear in response to the query:



Web Results **1** - **10** of about **2,590,000** for **citibank**. (**0.09** seconds)

Citibank
www.citibank.hackeddomain.com Find an account to suit your needs. Bank online, in-branch or by phone.
Sponsored Link

Welcome to **Citibank**
Choose one, **Citibank** Online, Checking, Student Checking, Savings, Citi en Español, Certificates of Deposit, ATM/Debit Cards, Bill Payment, Citi Cards **...**
www.latam.**citibank**.hackeddomain.com/ - 18k - Cached - Similar pages

Citigroup
Hong Kong: **Citibank** Celebrates Successful Incorporation of Consumer Banking **...** Building on its highly successful ThankYou Network®, **Citibank** now enables **...**
www.citigroup.com/citigroup/homepage/ - 18k - 19 Jul 2005 - Cached - Similar pages

Citibank Australia
CitiBank's browser-based internet banking service. Can order a bank check, change ATM PIN, and transfer funds to foreign bank accounts.
www.**citibank**.com.au/ - 3k - Cached - Similar pages

Citibank: **Citibank** student loans, research payment options, and **...**
Research **Citibank** student loans and more with information from The Student Loan Corporation. Learn more about planning and paying for college, applying for **...**
www.studentloan.com/ - 5k - Cached - Similar pages

Welcome to Citi.com
Choose one, **Citibank**, -**Citibank** Online, -Checking, -Student Checking, -Savings, -Certificates of Deposit, Citi en Español, Credit & Charge Cards **...**
www.citi.com/domain/ - 18k - Cached - Similar pages

Figure 21: Search engine result influencing

In the example figure above, we note that:

- The Pharmer has paid for the “sponsored link” to point to their host – www.citibank.hackeddomain.com. This purchase could have been made through the use of stolen credit card details, or other fraudulent financial transactions.
- The first non-sponsored link of the returned search contains the expected “Welcome to Citibank” message and supporting text. However, it is only by taking a closer look at the URL in green that the customer would notice anything untoward. In this case, the link points to the fake site hosted at www.latam.citibank.hackeddomain.com.
- The fact that the Pharmer has managed to get his fake site listed first is important for customers that make use of search engines that offer an “I’m feeling lucky” search option. With such an option available, the customer would have been automatically directed to this top-listed faked result.

The successful manipulation of search page rankings provides a very good delivery platform for the Pharmer because of the way it can be targeted to a specific customer audience or region (i.e. most popular search engines offer regional responses to search queries) and the difficulty for the victim organisation to identify or shut down the attack.

Section 4: Defence Mechanisms

Pharming attacks tend to be harder to defend against that traditional Phishing attacks due to the distributed nature of the attack focus and the use of resources not under the control of the victim organisation. In addition, the manipulation of the DNS resolution process occurs at such a fundamental level that there are very few methods available to reliably detect any malicious changes.

4.1. Classic Phishing Defences

Many of the defences used to thwart phishing attacks can be used to help prevent or limit the scope of future Pharming attacks. While readers are referred to the detailed coverage of these defence tactics explained in “The Phishing Guide”, a brief summary of these key defences is as follows:

Client-side

- Desktop protection technologies
- Utilisation of appropriate, less sophisticated, communication settings
- User application-level monitoring solutions
- Locking-down browser capabilities
- Digital signing and validation of email
- General security awareness

Server-side

- Improving customer awareness
- Providing validation information for official communications
- Ensuring that the Internet web application is securely developed and doesn’t include easily exploitable attack vectors
- Using strong token-based authentication systems
- Keeping naming systems simple and understandable

Enterprise

- Automatic validation of sending email server addresses,
- Digital signing of email services,
- Monitoring of corporate domains and notification of “similar” registrations,
- Perimeter or gateway protection agents,
- Third-party managed services.

4.2. Additional Pharming-specific Defences

While Phishing attacks typically use email as the attack delivery platform, Pharming attacks do not require any email obfuscation attacks to succeed – therefore Phishing defences that rely upon email security play a lesser role. The defences that will be most successful in preventing Pharming attacks focus upon the following areas:

- Change management, monitoring and alerting
- Third-party host resolution verification
- DNS server patching, updating and configuration
- Search engine control

4.2.1. Change Management, Monitoring and Alerting

The potential for an administrator or other authoritative employee to maliciously modify DNS resolution information without detection is great. As financial incentives increase, organisations and ISP's will need to ensure that adequate change control, monitoring and alerting mechanisms are in place and enforced.

It is recommended that:

- Wherever editing is possible, access to DNS configuration files and caching data is limited to approved employees only.
- A change management process is used to log and monitor all changes to DNS configuration information.
- Auditing of DNS record changes is instigated by a team external to any DNS administrative personnel; with automatic alerting of changes conducted in real time.
- Regular audits and comparative analysis of secondary DNS and caching servers should be conducted.

4.2.2. Third-party Host Resolution Verification Services

Toolbars

Many third-party developed plug-in toolbars originally designed to detect Phishing attacks are deceived by Pharming attacks. Typically, these Phishing toolbars show the IP address and reverse lookup information for the host that the browser has connected to, so that customer can clearly see if he has reached a fake site. Some managed toolbars (normally available through a subscription service) also compare the host name or URL of the current site to an updatable list (or real-time querying) of known phishing sites.

Some toolbars now offer limited anti-pharming protection by maintaining a stored list of previously validated "good" IP addresses associated with a particular web address or host name. Should the customer connect to an IP address not previously associated with the host name, a warning is raised. However, problems can occur with organisations that change the IP addresses of their online services, or have large numbers of IP addresses associated with a particular host name.

In addition, some toolbars provide IP address allocation information such as clearly stating the geographic region associated with a particular netblock. This is useful for identifying possible fake Pharming sites that have been setup in Poland pretending to be for an Australian bank for instance.

Server Certificates

To help prevent pharming attacks, an additional layer can be added to the authentication process, such as getting the server to prove it is what it says it is. This can be achieved through the use of server certificates.

Most web browsers have the ability to read and validate server identification certificates. The process would require the server host (or organisation) obtain a certificate from a trusted certificate authority, such as Verisign, and present it to the customer's browser upon connection for validation.

4.2.3. DNS Server Patching, Updating and Configuration

As with any Internet-based host, it is vial that all accessible services be configured in a secure manner and that all current security updates or patches be applied. Failure to do so is likely to result in an exploitation of any security weaknesses, resulting in a loss of data integrity.

Given the number of possible attacks that can be achieved by an attacker whom manages to compromise an organisation's DNS servers, these hosts are frequently targeted by attackers. Therefore it is vital that security patches and updates be applied as quickly as possible – typically organisations should aim to apply fixes within hours of release.

Similarly, it is important that organisations use up to date versions of the service wherever possible. As we have already discussed in section 3.6, each new version of the DNS software usually contains substantial changes to protect against the latest attack vectors (e.g. randomising DNS ID's, randomising port numbers, etc.)

Configuration

DNS servers typically offer organisations and their administrators an extensive number of configuration options. Therefore great care must be made during the installation and configuration process if the service is to be deployed securely.

Many common configuration mistakes are exploited for spoofing attacks. To help defend against spoofing attacks, the following advice should be heeded:

- Name servers are typically vulnerable to spoofing attacks from the Internet when configured to accept recursive queries. As explained in section 3.6.1, an attacker can query the name server for a host belonging to a domain managed by a DNS server under their control. To help prevent this:
    - If possible, turn of recursion.
    - Restrict the addresses to which the name server will respond to queries from.
    - Restrict the addresses to which the name server will respond to recursive queries from.
- When configured to disable recursion, the name server is operating in a passive mode – not sending queries on behalf of other name servers or resolvers. This has the following effects:
    - Since a non-recursive name server doesn't send out queries, it doesn't cache any data, and is difficult to spoof.
    - It is not advisable to disable recursion on a name server if it is relied upon by other servers for resolution or as a forwarder
- If it is not possible to turn off recursion, organisations should aim to restrict the types of queries that the name server accept.
    - Restrict queries based upon the addresses they came from.
    - Restrict queries based upon the zones that are allowed to ask about.
- "Glue Fetching" is the term used to define the process in which a name server attempts to retrieve A records to accompany any NS record requests. This process has the potential for receiving spoofed responses. By turning off glue fetching, the following occurs:
    - The name server will not generate these queries.
    - The name server will not build up a cache.

4.2.4. Search Engine Control

Internet search engines are undergoing constant development. Many of the methods used by attackers to increase their page ranking statistics are known of by the search engine developers, and a constant cycle of detection and refinement can be observed by both parties. For instance, Google modified its search algorithm to "reset" the page rank statistics of web sites that had recently changed ownership – this was to reduce the impact of instant "backlinks" and the weighting they attach to a ranking.

Traditionally the emphasis on increasing a pages ranking has been for revenue or lead generation – most closely associated with advertising. However, the increasing pace at which customers are relying upon search engines to access key services (such as online banking) means that a Pharmer who can get his fake site ranked at the top is likely to acquire a high number of victims.

Organisations should ensure that they regularly review keyword associations with their online services. Ideally automated processes should be developed to constantly monitor all the popular search engines for key search words or phrases customers are likely to use to locate their key services. It is also important that region-specific search engines also be monitored.

Section 5: **Summations**

Attacks focused upon host name resolution processes are likely to be of increasing importance to attackers seeking financial gain or to conduct identity theft operations. The lack of understanding customers (and many organisations) have of the background processes necessary to resolve IP address information to named hosts or services means that attacks that manipulate these DNS services are likely to go unnoticed.

Building upon the success of Phishing attacks, the new class of Pharming attacks enables the attacker to reach a wider customer audience with very little effort and a lower probability of detection.

To combat the new Pharming threat, it is vital that organisations understand how global DNS resolution services function and how they may be manipulated by an attacker. Armed with this knowledge, organisations can develop better monitoring and alerting processes to detect Pharming attacks early on – attacks that would most likely have escaped detection.

5.1. Resources

“The Phishing Guide”, *Gunter Ollmann, 2004*

“DNS and Bind”, *O’Reilly, 2001*

“Addressing Weaknesses in the Domain Name System Protocol”, *Christoph Schuba, Purdue University, August 1993*

“Security Best Practice: Host Naming and URL Conventions”, *Gunter Ollmann, 2005*

“DNS Cache Poisoning – The Next Generation”, *Joe Stewart, 2003*

“The Anatomy of a Large-Scale Hypertextual Web Search Engine”, *Sergey Brin and Lawrence Page, 2000*

Information Links

Current status and physical location of Root Servers - http://netmon.grnet.gr/stathost/rootns/

Top level generic domain name listing - http://www.iana.org/gtld/gtld.htm

Top level country-code domain name listings - http://www.iana.org/cctld/cctld-whois.htm

DNS Spoofing Techniques - http://www.securesphere.net/download/papers/dnsspoof.htm

Birthday Paradox - http://en.wikipedia.org/wiki/Birthday_paradox

Cache Poisoning - http://www.lurhq.com/cachepoisoning.html

SANS Warning - http://isc.sans.org/diary.php?date=2005-03-04

Abbreviations

ADSL	Asynchronous Digital Subscriber Line
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DSL	Digital Subscriber Line
FQDN	Fully Qualified Domain Name
IP	Internet Protocol
ISP	Internet Service Provider
NIS	Network Information Service
TCP	Transmission Control Protocol
TTL	Time To Live
UDP	User Datagram Protocol
URL	Uniform Resource Locator

WPAD	Web Proxy Automatic Discovery

5.2. List of Figures

About Next Generation Security Software (NGS)

NGS is the trusted supplier of specialist security software and hi-tech consulting services to large enterprise environments and governments throughout the world. Voted “best in the world” for vulnerability research and discovery in 2003, the company focuses its energies on advanced security solutions to combat today’s threats. In this capacity NGS act as adviser on vulnerability issues to the Communications-Electronics Security Group (CESG) the government department responsible for computer security in the UK and the National Infrastructure Security Co-ordination Centre (NISCC). NGS maintains the largest penetration testing and security cleared CHECK team in EMEA. Founded in 2001, NGS is headquartered in Sutton, Surrey, with research offices in Scotland, and works with clients on a truly international level.

About NGS Insight Security Research (NISR)

The NGS Insight Security Research team are actively researching and helping to fix security flaws in popular off-the-shelf products. As the world leaders in vulnerability discovery, NISR release more security advisories than any other commercial security research group in the world.