Global Information Assurance Certification Paper

Auditing a BIND DNS Server – An Administrators Perspective

Auditing Networks, Perimeters, and Systems
GSNA Practical Assignment
Version 2.1 (amended July 5, 2002)

Option 1 – Perform an Audit

Norrie Bennie

Table of Contents

Assignment 1 - Research in Audit, Measurement Practice, and Control

Assignment 2 – Create an Audit Checklist

Assignment 1 – Research in Audit, Measurement Practice, and Control

Introduction

A Domain Name Service (DNS) server is an important aspect of the Internet. Without DNS servers we would find it difficult to be able to use the Internet for web browsing or sending email. The main operation of a DNS server is to match domains names (human understandable names) to IP addresses of computers and networks corresponding to the name and vice-versa.

The Domain Name Service itself is regarded as a distributed database (Albitz & Liu, p.4), (Holland), as no one DNS server contains information regarding the whole of the Internet, but rather information on a specific part of the Internet. The Internet is broken up into zones, each zone being delegated by a DNS server. The topmost level domain is referred to as "." (the root domain). From this, the zones are broken down into country, then by organization type, and so on. Each DNS server has a list of hosts which managed the "." domain, these are known as the root servers, and are generally the first point of contact when trying to resolve a domain name that the DNS server does not know about.

Each DNS server contains information about the part of the Internet it knows, and sends this information to other DNS servers searching for the correct IP address for a certain host in a domain. As a result of containing information about hosts on networks, and redirecting other DNS servers or hosts to an IP address, DNS servers are a prime target (see *SANS/FBI Top 20 List*) for an attacker wishing to gain information about a network, or attempt to misdirect Internet traffic. The DNS server is therefore an essential host to protect for any organization, as it is the portal by which customers and the public are able to get access to Internet services that the organization provides.

Identify the system to be audited

The system being audited is a Domain Name Server running BIND 8.2.7 on a Solaris 8 which is running on a Sun Ultra 30. The system is the primary DNS server for the division. It also acts as the authoritative name server for several other domain names. The division has a number of sites located around the country. Each site has a sub domain relating to the state it is located in, and has DNS servers to which these zone are delegated to, but which are also slaves to the primary DNS server being audited. A slave DNS server also exists on the site being audited. All DNS servers are slaves for zones in other regions, as a result the DNS servers are fully meshed to provide redundancy and speed. The DNS is an important part of the organization, as many of the services which are provided by on the network rely on DNS authentication of a host. Richard Stevens (Stevens, p.200) refers to this as a hostname spoofing check. For example client machines must have reverse lookup address in order to access XDMCP, as well as internal and corporate web servers and application servers such as the HR database. The DNS is also important as it provides for a domain user email address "user@domain" rather than "user@host.domain" addresses. The network itself is protected by

a Cisco router which is acting as a firewall. All user services such as mail, web and DNS are situated on the internal network, but have ports open in the router to allow access. As the DNS server represents or delegates the zones for the division and organization, it is important that the DNS is secure, as any compromises may not only affect the division, but also the organization. Since the DNS server is on the internal network and the only other protection is the router, it is important that the operating system and the DNS server itself are hardened, to prevent unauthorized users gaining access to the internal system.

The following figure (Figure 1) shows the DNS server's configuration in the network, along with the other main DNS servers to which it talks.

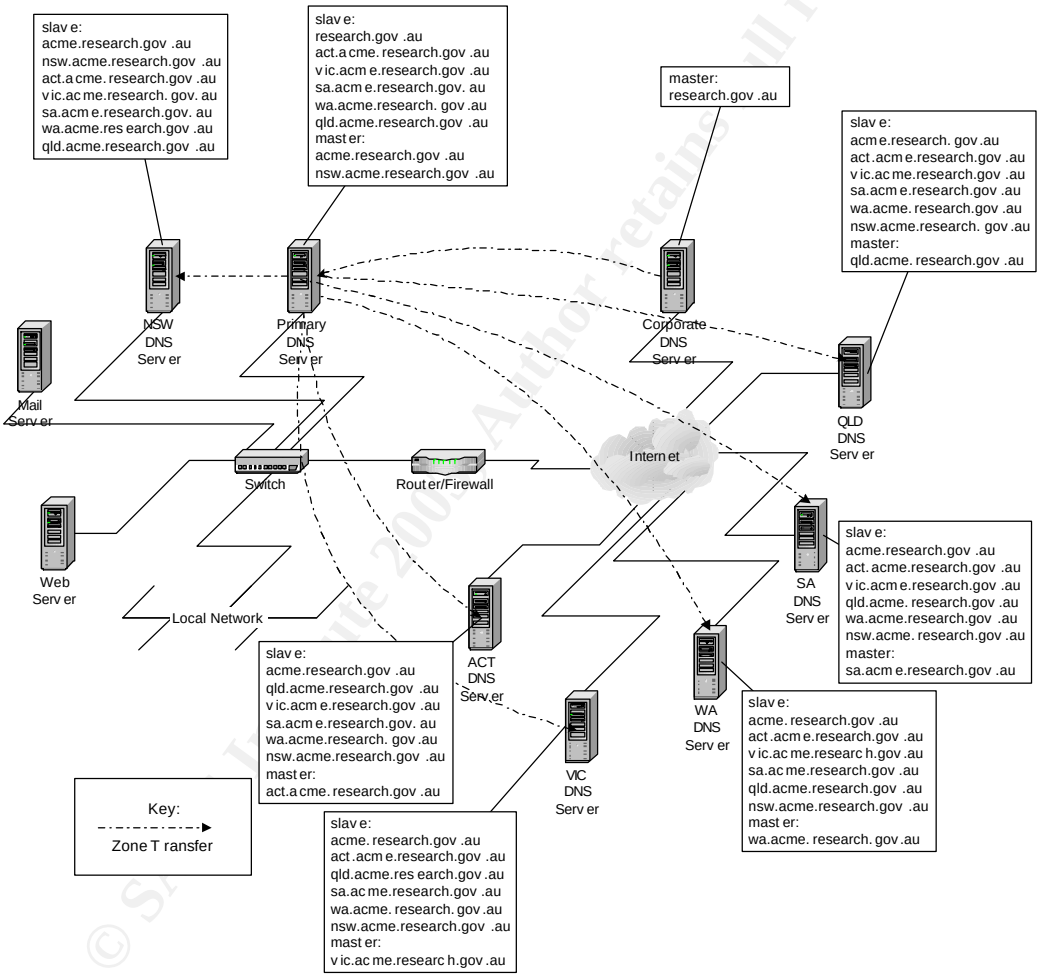

Figure 1 – DNS layout of network

The scope of the audit is on the primary DNS server, and does not include and audit the slave or other DNS servers within the division or external DNS servers to which the primary DNS communicates. In terms of usage, the DNS servers are used more frequently during the day when there are lots of users at work, so it is safer to do the audit out of work hours, as at least if a denial of service is caused, it will be less of a concern as there will not be too many

users affected locally. However, there is also a nightly backup of all the main servers and hosts on the network, so if the DNS server is brought down, it may cause the backup to fail. There is a three to four hour period between when users leave and the backup starts, this provides an opportunity to perform the audit during this time. Alternatively, after the backup completes, and before users come into work, there will be about a six to seven hour window of opportunity to perform the audit – at least an audit of those items which may cause the DNS to fail. While the DNS server is comprised of the operating system and the bind application, this audit is focused on DNS security aspects, rather than the operating system itself, however the operating system does needs to be audited. There are many audit lists available for Unix based operating systems, and many specifically focused on the Solaris operating system, I have suggested an audit checklist already developed, specifically one on auditing Solaris from CISECURITY (https://www.cisecurity.org/tools2/solaris/SolarisBenchmark.pdf). It seems to be a very complete and thorough audit checklist.

Evaluate the risk to the system

The Berkley Internet Name Domain (BIND) DNS server application has had many vulnerabilities over it's evolution, as can be seen from the many vulnerability announcements (seen in the list of references for CERT/CC, ISS and ISC) While this can be said about many applications, the DNS application is at the heart of the Internet and Internet communications, and without it we would find it hard to browse the Internet and talk to business partners and even send email, unless we were able to memorize all the IP addresses for machines that could be found on the Internet (an impossible task). The fact that we can use names instead of IP numbers to reference machines makes it a whole lot easier to try and remember web site names and email addresses (or at least make a good evaluated guess at it). This is the importance of DNS in regard to the Internet, being able to find a host by specifying it's name rather than IP address. The security concerns regarding BIND have made is part of the SANS/FBI (see *SANS/FBI Top 20 List*) top vulnerabilities list for UNIX.

In terms of software vulnerabilities there are numerous risks when running a DNS server. Many of these can be found in the CERT advisories and CVE databases, as well as the SANS and ISS web sites.

There are several risks associated with a DNS server. The main risks are DNS Flooding, DNS cache poisoning, Information leakage, DNS hijacking, DNS ID spoofing, BIND software vulnerabilities, and the DNS protocol itself. With respect to accessing the risk, I have used the categories defined by Australian Government standards and regulations described in the Commonwealth Protective Security Manual (Attorney-General's Department) which is shown in the current practices, which uses the formula of likelihood x consequence = risk severity.

DNS Flooding

There are two types of DNS flooding (Bennie, p.65). The first is aim at the DNS server itself, the second is aimed at a victim's machine which may be on an internal network or external network.

In the first case the DNS server is attacked by generating a large number of DNS queries and sending them to the DNS server, causing the DNS server to run out of resources and crash, or slow it down to an unusable state. See figures 2, 3 and 4.

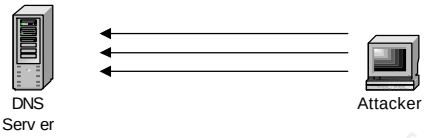

Figure 2 – Single machine sending multiple requests

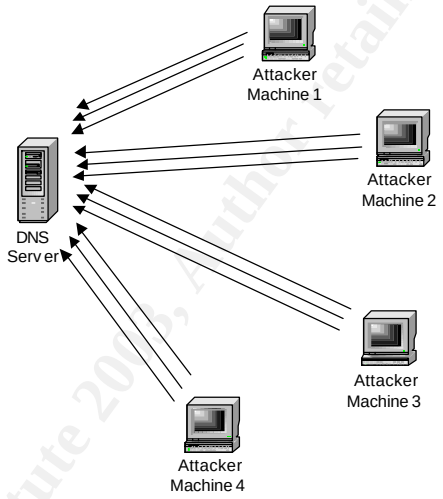

Figure 3 – multiple machines sending many requests

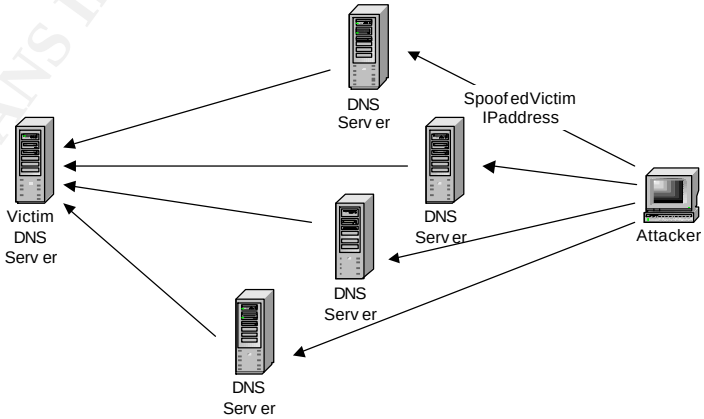

Figure 4 – Attacker sends spoofs DNS request to multiple DNS server from victims DNS server

In the second case, DNS queries are sent to the DNS server from a spoofed address of the victim by an attacker, the attacker may use more than one DNS server for the attack. This second from is aimed at a bandwidth denial of service (Bennie, p.65) (Caloyannides, p.39). See figure 5.

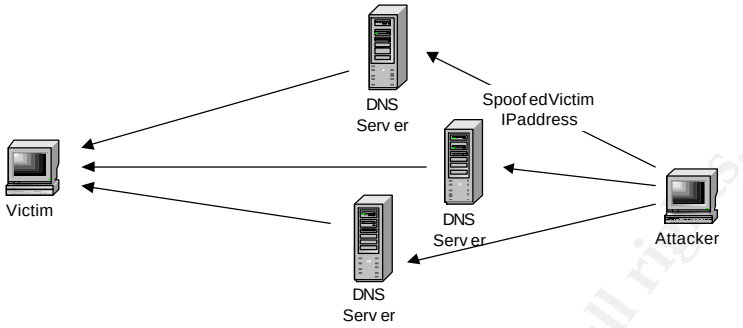

Figure 5 – Attacker sends spoofs requests to one or more DNS servers – similar to Figure 4, except victim could be any host, not just a DNS server.

Both forms of DNS flood are aimed at a denial of service.

Likelihood of DNS flooding

In the first case the likelihood or chances of a DNS flood aimed at our DNS server is a real possibility and can be considered as “moderate”. If quality of service is enabled on the router to rate limit packets, or if the DNS server itself has rate limiting enabled then while the likelihood of the attack occurring will remain the same, the likelihood of effecting the DNS server is reduced. Having a secondary slave server means that’s if only one server is attacked, the DNS services will still be provided so the likelihood is reduced to “unlikely”.

In the second case, the likelihood of a DNS flood from an internal machine to an external DNS server is not possible as anti-spoofing rules are turned on at the router. However an internal machine could use the internal DNS services on the local network (the primary and secondary DNS servers) to send a DNS flood to a spoofed address as the packets are not going through the router. Due to the router having stateful filtering turned on, the likelihood of an internal machine attacking an external machine via a DNS flood is “rare”.

The likelihood of an external attacker performing a DNS flood on the other hand is “likely” as spoofing an IP address for queries from an external site will not get blocked via the router as it would appear genuine. Rate limiting on the DNS server and the router however will reduce this likelihood to “moderate”. Also restricting queries to only internal clients or authorised hosts would make the likelihood to “unlikely”.

Consequences of DNS Flooding.

Both types of DNS flooding attacks attempt to cause a denial of service. In the first case, our DNS server could be brought down. If our secondary slave server did not exist the consequences would be twofold. Firstly external

parties may not be able to access our systems (however with a slave DNS server in other regions this likelihood is reduced). However the internal servers which need to do reverse authentication would not be able to, as the DNS servers they know, about would not be up to send queries to. Secondly since the DNS provides name resolution to our internal network, our internal clients would not be able to access corporate systems and some internal services. The reason for this is the reverse authentication not being available, and also the DNS servers not being available for a resolver client to ask an IP lookup from it for hosts or services – this would also mean clients could not access external sites or clients.

What does a denial of service in the first case of DNS flooding mean to the organisation – cost in time and wages of employees who are unable to work as some services become available, such as email, NIS access (means can't access home directory or use samba as they are configured for reverse IP authentication) , and web access. The consequences of the first type of DNS flood are "high" to "severe".

In the second form of DNS flooding, where the DNS server is used to attack a victim, the consequences of the attack also include a denial of services against an external party's network. With respect to network availability and service access to the division and organisation, these will not be greatly affected. There are however other consequences which could effect the organisation. If such an attack were possible to be allowed, there may be legal action against the organisation and the divisions and organisations credibility by the victim and others may be reduced. In this case the consequences can be considered "medium" to "high".

Overall risk of DNS flooding

The risk associated with DNS flood aimed at the DNS server is "significant". (likelihood => unlikely x consequence => high = risk => significant)

The risk of an internal host performing a DNS flood against an external victim is "low".
(likelihood => rare x consequence => medium = risk => low)

This risk of an external host performing a DNS flood using our DNS server to attack a victim is "Significant".
(likelihood => moderate x consequence => medium = risk => significant).

DNS Cache Poisoning

There are several different ways cache poisoning can occur (Detoisien), (Hanely), (Stewart). DNS cache poisoning is aimed at confusing or injecting incorrect information or data into a DNS servers database. By providing incorrect DNS information, when a host requests an IP address to be looked up or a the IP address of a service to be looked up, instead of getting the real or true IP address of the machine, a different IP address is returned. The DNS server (depending on how it is configured) will cache this information and next

time the IP address needs to be looked up, it will return the information returned in it's cache. The two most common ways of poisoning a DNS cache are by effecting the data returned in a response and by effecting the information returned from an non-authoritative server (Erdfelt), (Liu). In the first case, the final IP address returned from the DNS server has wrong IP address. Caching is only performed on authoritative answers in this type of cache poisoning. See Figure 6.

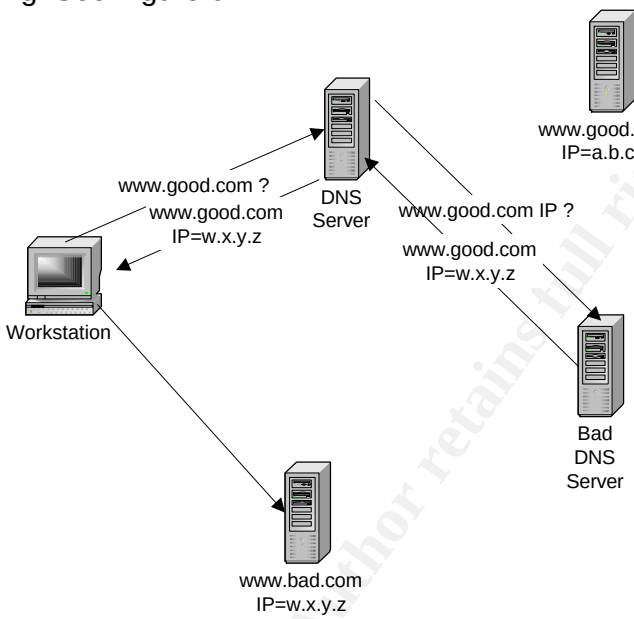

Figure 6 – Incorrect Data returned to DNS Server

In the second case, when a DNS server is not authoritative or does not know the IP address, it will send the resolver or DNS server querying it the IP addresses and names of other DNS servers to lookup (Albitz & Liu, p.211). An attacker may use this by specifying the name of the real DNS server to lookup, but specify a fake IP address. In this second case the returning of names of other servers and IP addresses is often referred to as fetching the glue (Liu) (Householder, King & Silva). See Figure 7.

There is also another way of effecting the DNS servers cache and that is a residual effect if the server is configured to cache data, when DNS ID spoofing occurs (discussed later).

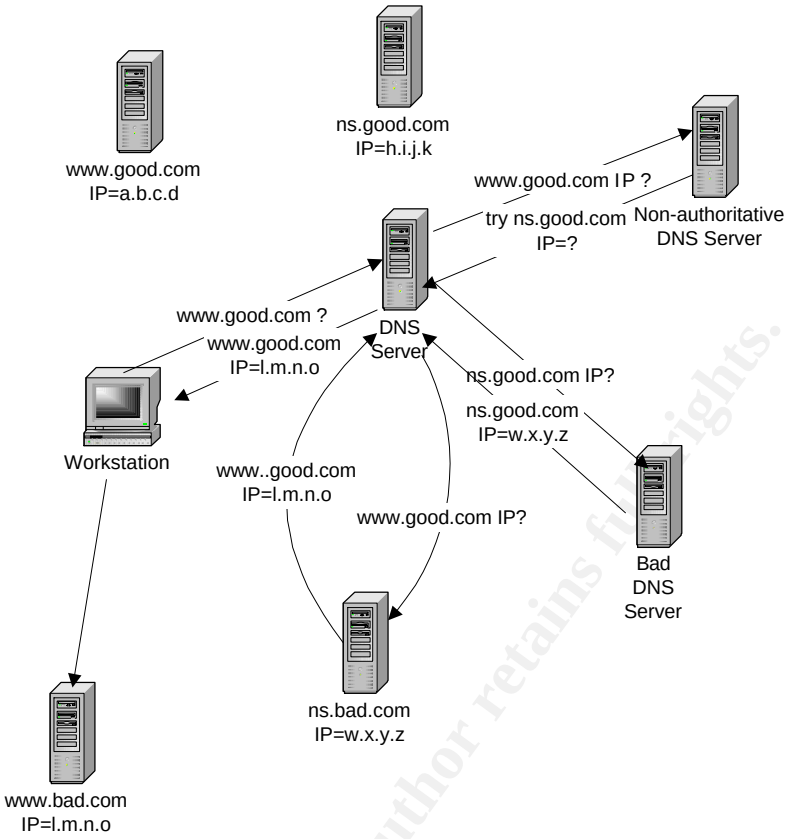

Figure 7 – DNS Server needs to get IP address of nex t DNS s erver to contact (fetch -glue)
Both invalid name server and invalid host addresses get cached.

While there are several ways of poisoning a cache, there are also two types of cache which can be poisoned (Albitz & Liu, p.34). There is the normal cache where information about names and IP addresses are stored, then there is the negative cache. The negative cache stores the names or IP addresses of hosts which cannot be resolved (Gregory, pp.198-200). Negative caching in BIND has the option of making the DNS server authoritative when responding to DNS queries that it has found information for in it's negative cache (Albitz & Liu, pp.296-297,327). Being authoritative means that other DNS server may also then cache the negative cache information of the DNS server.

Likelihood of DNS cache poisoning.

DNS cache poisoning is probably the most common form of attack on a DNS server. In the first case where a incorrect data about the hostname being queried for is returned. For caching to occur however the DNS server being attacked must have recursion turned on. Having recursion means that the DNS server will perform the search for the IP address rather than just referring the query to another DNS server (Albitz & Liu, pp.28-30). It also requires an attacker to have a malicious DNS server setup (Detoisien), and requires a query to the malicious DNS server to occur. In general the attacker often sends a request to our DNS server to lookup an address hosted by their malicious DNS server (Liu). This will cache the information about the

attackers site. By the attacker sending a query to cause the DNS server to do a search can be considered as an active attack. By an internal client seeking an IP address hosted by the attacker is a passive attack as the attacker is not actively poisoning the cache. This form of cache poisoning though would require generally require you wanting information from the attackers site in order to be effective. Unless of course the IP address returned from the attacker is for a high volume site that people would go to, but with another hostname of the attackers domain associated with it. Eg. Attacker has domain evil.com but returns IP address for host in it's site which actually is the IP address of www.goodguys.com so now goodguys.com IP is associated with evil.com domain in the cache. The likelihood of DNS cache poisoning depends on whether recursion is turned on or off, and if it is on whether it is restricted or not. In our case examination which is shown in the audit shows that recursion is turned off for external DNS queries which are not from the division or organisation. As a result the likelihood of this type of cache poisoning is "unlikely", however if it was not turned off or restricted the likelihood would be "almost certain".

In the case for DNS cache poisoning in relation to a DNS response containing the IP addresses and names of authoritative servers, this also relates to recursion and fetching the "glue". There is an option in BIND 8 to turn off the "glue" fetching option (Albitz & Liu, p.248). As stated in the DNS cache poisoning related to false information on the host being queried, this second form of DNS cache poison is "rare" if recursion is turned off or restricted. However if it is not turned off or restricted, the likelihood of this attack would be "almost certain".

As for negative caching, this attack it worsened by the fact that a DNS can respond as authoritative for information in it's cache. In this instance the attacker does not need to own an authoritative name server, to return an authoritative answer. By returning an answer as authoritative other DNS servers will cache the negative information.

Consequences of DNS cache poisoning.

The main consequences of DNS cache poisoning is a denial of service attack. By providing a response to an IP address of a machine that is unreachable or does not exist. As a result when the host which asked for the IP to be resolved gets a response it tries to connect to a machine that does not exist. The other alternative to this is the IP address for the machine exists, but the machine does not host the service that the host was trying to connect to. A second form of denial of service which DNS cache poisoning can produce is when the IP address returned for a query points to a machine with the service running, but with the wrong IP address (Fiore & Francois). For example someone wants to lookup www.research.gov.au, but ends up going to a gambling site. In extreme cases where the attacker may be malicious or fraudulent, the IP address points to a service which the hacker is running which tries to imitate a real site. For example, the hacker is running a web server which looks like a banks web, or an on-line store (Giovanni). A user fills in their bank details, or credit card details and the hacker now has a copy of

his information which they may now use of sell. The 2003 Australian Computer Crime and Security Survey (Attorney-General's Department, p.23) describes a recent case where a well known large bank in Australia had their web pages imitated an user accounts and passwords were retrieved from customers. This type of DNS cache poisoning relates to DNS hijacking (Caloyannides, p.37).

There are several consequences to DNS cache poisoning as it can effect the internal network as well as external clients. As mentioned above, cache poisoning can leat to a denial of service. In the instance where an internal host is directed to a machine that does not exist or not running a service, the impact depends on the how many clients attempt to request the IP address of a machine. If the number of machines requesting the IP address is few, then the impact to the division is minimal. On the other hand if it is a service that a lot of machines want to connect to then impact will be greater. Generally the DNS cache poisoning effects access to external services and so the internal network should continue without major incident. As a result, DNS cache poisoning in relation to a denial of service to a non-existent host or service is "low" to "medium". The reason it is not "rare" is that the DNS server will perform recursion for internal clients queries if asked. As a result, a passive DNS cache poisoning attack may occur.

In the instance where an external host is redirected to a service which is not the organisations or the divisions, such as web page of an unreputable site, it can be an embarrassment and also loss of reputation depending on the type of web site pointed to. In this type of cache poisoning incident however, since we are the authoritative server for our own division and site, we should not get effected by cache poisoning of our own web host or other services. In this instance it would not be our DNS server which is poisoned but an external DNS server. In such a case we have no control over it, and it will be an external party referencing the external DNS which will be effected by a denial of service. The only possible way our DNS server may be effected is if a corporate DNS server response is spoofed via DNS ID spoofing when an internal client requests the IP of a corporate server and we get some invalid data. However this would be out of our control. While external DNS servers may be effected, in terms of our division, the local network would not be effected. As a result, the consequences of misredirection of a web page would be "low" to "medium".

To detect DNS cache poisoning, secondary slave servers can be queried to see if the information returned is consistent (Anonymous, 1999, p.284).

Overall risk of DNS Cache Poisoning

DNS cache poisoning effecting sites internal users visit when recursion is turned off or restricted is "significant".
(likelihood => moderate x consequence => medium = risk => significant)

DNS cache poisoning effecting sites internal users visit when recursion is turned and on not restricted is "high".

(likelihood => almost certain x consequence => medium = risk => high).

DNS cache poisoning effecting external users visiting our site when referencing our DNS server is “low”.
(likelihood => unlikely x consequence => medium = risk => low).

DNS ID spoofing

DNS ID’s are used to authenticate queries and responses between DNS servers and clients (Stewart).

DNS ID spoofing (Detoisien) is where the DNS packet ID and IP address are spoofed, fooling a DNS server or client into believing it has received a DNS response from the real DNS server (ADM Crew). This type of attack consists of two stages (Detoisien), the first is monitoring the response from a DNS server to see whether the DNS ID’s are predictable. For this to occur the attacker usually has their own DNS server it asks the target DNS server to query. See figure 8.

Figure 8 – Stage 1 – monitor DNS ID

Secondly, if the target DNS ID’s are predictable then the attacker attempts to send a request to the target DNS server they want to affect by asking it to resolve an IP address for a victim’s host. This request may be from a spoofed IP address. Because the DNS ID is predicable (Caloyannides, p.35), the attacker then forges replies (Medvedovsky) from the victim using IP and DNS ID spoofing to the target DNS, so that the target DNS will receive the response from the attacker before the real victim’s DNS responds. See figure 9.

This type of attack has several victims, one is the target DNS, the other is the victims host, in terms of external sources may be redirected to another IP address. This attack requires the target DNS server to cache responses.

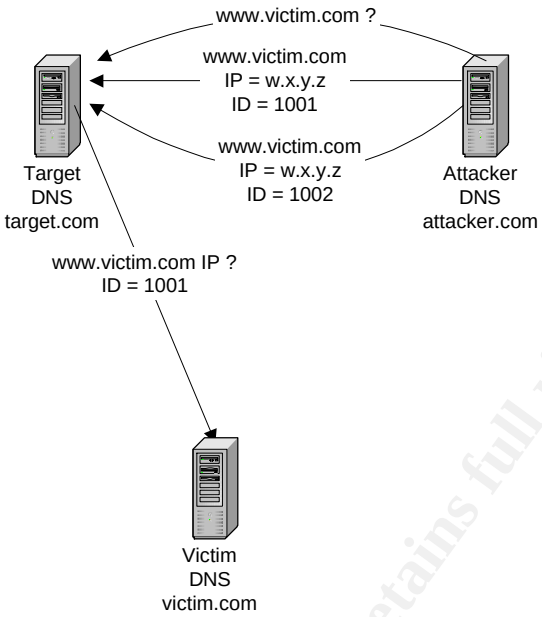

Figure 9 – Stage 2: spoof replies to target DNS before the victim's DNS responds

A new form of DNS ID spoofing was found by Vagner Sacramento (Sacramento) and is known as the Birthday Attack (Stewart). The Birthday Attack is works on the fact that if a DNS receives n number of queries, then n number of lookups and responses will be returned (Stewart). This effectively means that there is a much greater likelihood of the DNS ID being spoofed with the more queries to the same address sent. Effectively this does not rely on the DNS ID being as predictable as in the first DNS ID spoofing case. See figure 10.

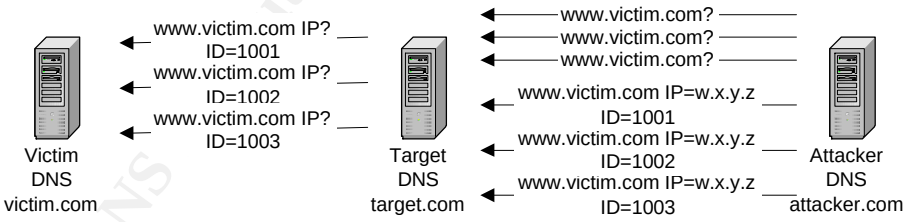

Figure 10 – DNS ID spoofing with Birthday Attack (Stew art)

Likelihood of DNS ID Spoofing

The likelihood of the first instance of DNS ID spoofing will be "almost certain" if the DNS ID's are predictable, and if the DNS server does not restrict who it answers queries for or who it performs recursive queries for. If on the other hand DNS ID's are randomized and recursive queries are restricted to authorised hosts, then the likelihood of this form of DNS ID spoofing is "unlikely".

The likelihood of the Birthday Attack is more so than the standard DNS ID spoofing case. It is not as dependent on predictability, but more the fact that

the more requests one sends and the more responses one fakes, then the more likely a fake response will have the correct DNS ID. As a result the likelihood of this attack, with recursion unrestricted is “almost certain”. With recursion disabled for unauthorised hosts, the likelihood of the attack is “unlikely”.

Consequences of DNS ID Spoofing

If the DNS server performs recursion and therefore caches the responses, the consequences are “medium”. Both these forms of DNS ID spoofing allow an attacker to cause a cache poisoning of a DNS server to provide the wrong IP to the host asking for a name to be resolved. The consequences of DNS ID spoofing are the same as DNS cache poisoning. If caching is restricted and anti-spoofing IP address rules are implemented on the router or firewall, then the consequences are “low”.

Overall risk of DNS ID Spoofing

The risk of DNS ID spoofing with recursion not restricted and DNS ID’s predictable is “high”.
(likelihood => almost certain x consequence => medium = risk => high).

The risk of DNS ID spoofing with recursion not restricted and DNS ID’s randomised is “significant”.
(likelihood => moderate x consequence => medium = risk => significant).

The risk of DNS ID spoofing via the Birthday Attack and recursion not restricted is “high”.
(likelihood => almost certain x consequence => medium = risk => high).

The risk of any form of DNS ID spoofing with recursion turned off or restricted is “low”.
(likelihood => unlikely x consequence => low = risk => low).

Software vulnerabilities

In general there are two types of software vulnerabilities. The first is related to the application code which may contain software bugs which can be exploited by an attacker. The second type of software vulnerability is misconfiguration of the software, which may lead to lax security which an attacker can exploit.

There are generally two main types of software coding bugs. The first are vulnerabilities which allow an attacker to exploit some bug which causes the DNS server to crash or fail. The second type are vulnerabilities which allow an attacker to gain root access to a DNS server (E-Mind). An example of the first type of software vulnerability is described in CERT Advisory CA-2002-15 (see *SANS/FBI Top 20 List*) (see Men & Mice’s *BIND Vulnerability*) in relation to the latest versions of BIND, BIND 9. The server is sent a DNS packet that it is unable to handle properly causing the DNS server to shutdown. Another example is the server being unable to cope with multiple resource records for

the same resource in the one packet caused the DNS server to crash. An example of the second type of software exploit is the BIND NXT buffer overflow which was discovered in earlier versions of BIND 8.2 (Athanasiou), (see *CERT Advisory CA-1999-14),* (McClure, Scambray & Kurtz, p.347).

In relation to the misconfiguration of the BIND software, this may lead to an attacker gaining access to information about the system or network (McClure, Scambray & Kurtz, p.19) in the one case to being able to modify the DNS data in the worst case. Men & Mice surveys (see *Survey Results: Domain health survey for .COM*) show 68.4% of ".COM" zones are misconfigured and 33% or a third of DNS servers are spoofable (see *DNS Survey*) due to misconfiguration.

Likelihood of Software Vulnerabilities

In researching vulnerabilities, it appears that BIND 8.2.7 is the latest secure version of BIND 8.2. As a result there are no known BIND 8.2.7 bugs which could allow root access or cause a DNS server to crash. That being said however, there is always the possibility that vulnerability such as a root exploit will be found in the future. The only known vulnerability for BIND 8.2.7 is the birthday attack, but this is affecting all DNS servers running bind. As a result, the likelihood of being attacked through a software vulnerability is "rare" at present.

In terms of software vulnerabilities to gain root access. BIND 8,2,7 is considered the most secure version of bind 8.2, as a result the likelihood of an attack to gain root access it also "rare", based on current vulnerabilities list.

In terms of misconfiguration – this may vary based on what is misconfigured. For example if zone transfers aren't restricted, or recursion is not turned off or restricted or dynamic updates are not restricted (Sahlin), then the likelihood will be "almost certain". In fact it will be "almost certain" that an attacker will attempt to test a DNS server to check if it is misconfigured in some way, that they may be able to use as an attack.

Consequences of Software Vulnerabilities

If an attacker can exploit vulnerability in the software the DNS server could be brought down to an unusable state causing a denial of service attack. The same consequences exist here as for what the denial of service attack from DNS flooding attack aimed at a server. Due to not only internal services but external service being affected, the consequences are "medium" to "high".

On the other hand an exploit that allows a user to gain remote access would mean that the attacker had free range on the DNS server. Meaning they could directly modify the zone information on the computer and any other piece of information on the DNS server they wanted to. The attacker would no need to use cache poisoning to modify records as they could modify them directly. In this case the consequences are "high" to "extreme".

With misconfiguration the consequences may vary. If zone transfer restrictions is the only misconfiguration, then the consequence is that an attacker may be able to gain information about the network and the hosts on it. This would not cause any operational problems to the DNS server itself, but may provide reconnaissance information for an attacker to plan an attack. In this case the consequence may vary based on what information about the network and hosts was available, but as the actual zone transfer did not cause any disruption to services, the consequence may be considered "low" to "medium". On the other end of the spectrum, if recursion and dynamic updates were not restricted, then an attacker could do DNS cache poisoning, or make changes to the DNS data itself, respectively. In this case the consequences would be "medium" to "very high".

Overall Risk of Software Vulnerabilities
The risk of a software bug leading to a denial of service exploit is "moderate". (likelihood => rare x consequence => high = risk => moderate)

The risk of a software bug leading to a root exploit is "significant".
(likelihood => rare x consequence => extreme = risk => significant).

The risk of software misconfiguration leading to information leakage is "significant".
(likelihood => almost certain x consequence => low = risk => significant)

The risk of software misconfiguration leading to cache poisoning is "major". (likelihood => almost certain x consequence =>medium = risk => major).

The risk of software misconfiguration leading to DNS data being modified is "severe".
(likelihood => almost certain x consequence => high = risk => severe).

DNS Hijacking

DNS hijacking, also known as domain hijacking (Stewart), employs cache poisoning and DNS ID spoofing with the intent to redirect traffic to a fake site or DNS server (see *2003 Australia Computer Crime and Security Survey*, p.23). Rather than cause a denial of service, the basis behind the attack is aimed at fraudulent activity, although denial of service may be a repercussion as traffic is redirected elsewhere instead of at the real site. Denial of Service aimed at disabling a target DNS server, is sometimes involved in DNS hijacking however. If the real DNS server is up it may be able to answer some queries, whereas if it can't be found and DNS servers trying to resolve a host may try other hosts listed in the non-authoritative list of servers they receive, which may be servers which have their cache poisoned, as an authoritative DNS server can't get cache poisoning for the zone data it owns. See figure 11.

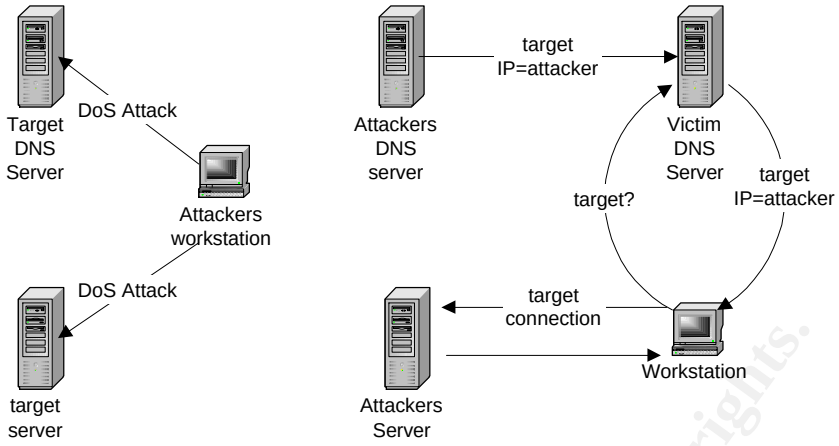

Figure 11 – DNS/Domain hijacking

Likelihood of DNS Hijacking

The aim of DNS hijacking is not to affect internal hosts accessing internal services on a local area network, but is aimed at hosts accessing external services. DNS hijacking is outside the control of the local DNS server, and as a result is likely to occur. There are two aspects to DNS hijacking to consider – the likelihood of our domain to be hijacked, the likelihood our DNS server could be poisoned to allow hijacking of an external domain.

The likelihood of our domain being hijacked is possible but would involve external DNS servers being cache poisoned – we have no control over this. However as our top level domain acme.research.gov.au has a number of servers that act as slaves for the domain there are chances that all client queries to the domain get redirected are less likely, especially if the clients are in Australia. However if the cache poisoning occurred overseas on, the likelihood of cache poisoning may be higher as a responses from the DNS servers in Australia may take longer than fake responses – but it all relies on vulnerable DNS servers outside of our control. As a result the likelihood of this attack could be considered “likely”.

The likelihood of our DNS servers being used in an attack against another domain would rely on our DNS servers being recursive without restriction. If recursion was not restricted, cache poisoning would “almost certainly” occur. However as our DNS servers restrict recursive queries, the likelihood of our DNS server being used to cache poison is reduced, however it is not completely eliminated, as recursion is allowed for internal clients. If an attacker could get an internal machine to query their malicious DNS server, then the likelihood is higher. As anti-spoofing rules are in place on the router, an attacker could not spoof a query from an internal host, so as a result likelihood of our DNS servers being used in an attack against another domain is “unlikely”.

Consequences of DNS Hijacking

DNS hijacking will not affect a local network from accessing local services, so the local network on which our primary DNS server is located, would not be effected.

If our domain was hijacked however, the consequences could be quite damaging. If acme.research.gov.au was hijacked, an attacker could pose as an internal host and gain access to corporate services such as the internal web services and the HR database (McClure, Scambray & Kurtz, p.513). Similarly if they posed as an internal host, they may be able to perform a recursive query if the other DNS servers in the organisation allowed recursion to internal hosts. There is also the case if the attacker could hijack or spoof the corporate DNS server. In this case, they might be able to send bogus zone transfer information to our DNS server.

As a result of domain hijacking of our domain, there would be an implicit denial of service resulting from clients not being able to get our site, however the denial of service would only be for external clients. The consequences of our domain being hijacked can be considered as "very high" in the case where an attacker could pose as a valid host to a corporate server. In the case where clients are sent to another site, the consequences are less, and may be considered "medium".

If our DNS server was poisoned, the consequences would be that hosts wanting to go to a certain domain would be redirected to another site. If our corporate services were spoofed such as the corporate web or other server where telnet or ssh is required, a user may enter their username and password which an attacker could capture and then use to gain access to our internal network or services. In this scenario the consequences are "very high".

The overall risk of DNS hijacking

The risk of our domain being hijacked is "high".
(likelihood => likely x consequence => high" = risk => high).

The risk of our DNS server being used in an attack against another domain is "significant".
(likelihood => unlikely x consequence => very high = risk => significant).

Man in the Middle Attack

There are several forms of man in the middle attack. There are those on a local area network where arp spoofing is used, and there are those which happen across the Internet and these are the ones we are concerned about as it uses the DNS to implement the attack. A man in the middle attack is similar to DNS hijacking (Stewart). It relies on the ability to perform DNS cache poisoning. However unlike a DNS hijack where the real DNS server may be brought down with a denial of service attack to stop responding for a

domain, a man in the middle attack may rely on the target DNS server to be up. The reason for this is to poison the target DNS as well as the victims, so that traffic can be intercepted. This leads to two types of man in the middle attacks – a transparent (see figure 13) and non-transparent (see figure 12). In the first case, both the target DNS and victim's DNS are poisoned, so that they both redirect traffic to the attacker. In the second case, only the victim's DNS server is poisoned to point to the attacker. The attacker then acts like a proxy server and established a connection with the target site and re-sends the victim's information, so the target site sees the attacker's machine, not the victims.

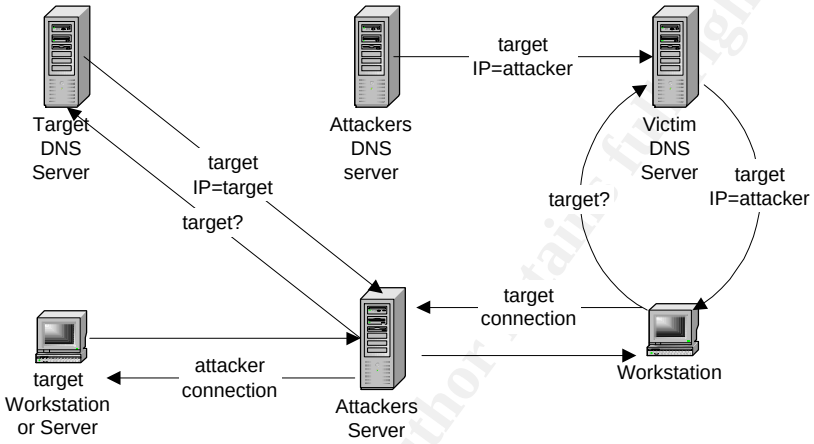

Figure 12 – Non-transparent man-in-middle attack

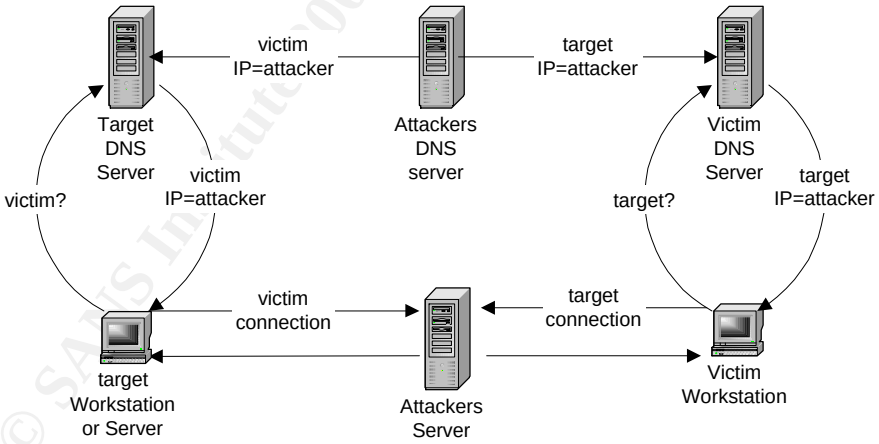

Figure 13 – Transparent man-in-middle

With a man in the middle attack the aim of the attacker is to intercept traffic between two hosts. The attackers host acts as a relay or transparent proxy between the two depending on the type of man in the middle attack. There are a couple of different types of man in the middle attacks. In one instance the aim is just to monitor the traffic. In another the attacker has set up a service which a victim users believing it is the real machine, the attacker then sets up a session with the real machine and sends the victims inputs and vice-versa with the real machines input to the victim (Giovanni). In the first instance it is

just eaves dropping, in the second instance it is intercepting the data and may involve manipulating it.

An attacker relies on DNS cache poisoning to poison the cache of a DNS that the victim uses or one which the victims DNS will query to get to a target host. The victim will get poisoned information pointing them to the attackers DNS server and host. The attacker then establishes the session to a target host, whose true IP address will be returned from the target hosts DNS server. The target DNS server may be poisoned with the IP address of the viaticums machine also pointing to the attacker's host. A man in the middle attack is focused on two endpoints, as a result only the DNS server of the victim and target may be poisoned.

The Likelihood of a man in the middle attack

The man in the middle attack, like DNS hijacking relies on cache poisoning to be effective (Detoisien) (Liu). If the DNS servers involved do not have recursion restricted or glue fetching turned off, then it is "likely" that those DNS servers will get poisoned caches. DNS ID spoofing is probably the most effective way to poison a DNS cache, if the DNS ID is randomised the likelihood is lessened, but still does not prevent cache poisoning occurring.

If recursion is turned off, and glue fetching turned off then the likelihood is lessened. If DNSSEC is employed – which authenticates hosts and responses (Irving), (Lierley, pp.482-495), (Hinshelwood), (Tunnissen), then the likelihood of this type of attack would be "rare", as DNSSEC provides a mechanism for the DNS servers to authenticate each other.

The consequences of a man in the middle attack

The types of consequences resulting from a man in the middle attack could be that private and confidential information is read by an attacker, or manipulated. It depends on whether our DNS server is the victim's DNS or the targets DNS.

If our DNS is the victims, then it is likely the attack is aimed at our employees. In this case it may be that the data being sent by our employees or hosts could get manipulated to provide incorrect information. For example, in the case of a banking web site, the victim asks to transfer $x from one account to another. The attacker could modify this to specify $y being transferred from the account to the attackers account. Basically the same consequences exist for a man in the middle attack as for DNS hijacking in relation to a service being hijacked. DNS hijacking can used for gaining information which can be used at a later stage, such as obtaining user names and passwords – the attacker can use them later on to get into the target site. A man in the middle attack is more for acting on a live connection rather than storing information for later use. In the case of a corporate DNS being spoofed, user names and passwords may be captured leading to a system compromise or if the HR database information is intercepted, user details could be captured or modified. In these cases the consequences are "high" to "extreme".

If our DNS is the target DNS, then it is a service we provide which is being affected, such as a web page where the client's information such as user name and password to certain pages may be captured. This would also lead to "high" to "extreme" consequences.

The overall risk of a main in the middle attack.

The risk of a man in the middle attack where our DNS server is the target and recursion not restricted and fetch glue is on, is "high".
(likelihood=>likely x consequence=>high = risk=> high).

The risk of a man in the middle attack were our DNS server is the target and recursion and fetch glue are restricted and turned off respectively is "moderate".
(likelihood=rare x consequence=high => risk=moderate)

The risk of a man in the middle attack where our DNS server is the victim and recursion not restricted and fetch glue is on, is "high".
(likelihood=>likely x consequence=>high = risk=> high).

The risk of a man in the middle attack were our DNS server is the victim and recursion and fetch glue are restricted and turned off respectively is "moderate".
(likelihood=rare x consequence=high => risk=moderate)

What is the current state of practice, if any?

While there is a lot of information and a number of tools for auditing the data stored within a DNS server and ensuring the data's consistency ensuring there is no conflicting or erroneous data, there does not appear to be any security audit checklists for a DNS server. There are however a number of great sources dealing with configuring BIND with security in mind. While these are not audit checklists, the recommendations they make can be turned into checklist items.

There are several companies such as Men and Mice and Lantest that will run a remote test and audit of the DNS and provide a report, but they do not provide the audit checklist they use on the web site. Men and Mice though provide lots of white papers, surveys and documents on various aspects of DNS - helping give back to the IT security community.

There are several good papers which can be found in the SANS reading room on DNS security and BIND, these are *How Secure are the Root DNS Servers?* by Susan Baranowski (Baranowski), *Why is Securing DNS zone transfer necessary?* By Steve Lau (Lau), *Defense in Depth for DNS* by Cheng C. Teoh. (Teoh), *DNS Security Considerations and the Alternatives to BIND*" by Lim Seng Chor (Seng Chor), and *The Achilles Heal of DNS* by Christopher Irving (Irving).

I found the most useful references were *Security and Internet Name Server* by Cricket Liu (Liu) , and the O'Reilly book *DNS and BIND* – 3rd Edition by Paul Albitz and Cricket Lui (Albitz & Liu). The *DNS and BIND* book is the one of the best source of technical details and looks at all of the options of BIND in great detail and while some of the options were not discussed in terms of security, how they affect the DNS servers operation can be security related. Another good source on best practices on naming standards within an organization is *Best Practices – Naming* – version 1.1 by Tom Jackiewicz (Jackiewicz) which gives recommendations on how BIND should be setup, among setting up other services. Other papers I found useful were *Hardening the BIND DNS Server* by Sean Boran (Boran) and *Securing an Internet Name Server* by Allen Householder, Brian King and Ken Silva release by CERT (Householder, King & Silva) and based upon Cricket Liu's paper (Liu). Another good source for looking in detail on how DNS works is *TCP/IP Illustrated Volume1 – The Protocols* by W. Richard Stevens (Stevens). This book is an excellent reference on TCP/IP protocols and looks in depth at how DNS works and examines DNS packet structure and standard responses.

I also found another good source has been the bind vulnerabilities from such sources as the Internet Software Consortium (ISC) who write BIND, CERT and AusCERT advisories, Internet Security Systems (ISS), and the SANS vulnerability lists – some of these vulnerabilities are in the list of references associated with these sites. Not only do they state the problems with BIND, they often have recommendations on how to fix the vulnerabilities, whether it be upgrades or configuration changes. Other books like *Maximum Security* (Anonymous, 2001), *Maximum Linux Security* (Anonymous, 1999) examine exploits of DNS server, and provide recommendations on how to combat the exploits.

Other sites like Security Focus (http://www.securityfocus.com) provide alerts along with other papers on security. One such paper from the Security Focus web site is *DNS Cache Poisoning – The Next Generation* by Joe Stewart (Stewart). This paper can also be found on the SANS site, and is an excellent paper on cache poisoning and the implementation of a Birthday Attack on a DNS server, and provides a perl script to test if the a DNS server is vulnerable. The email from Ramon Izaguire on BugTraq entitled *An Implementation of a Birthday Attack in a DNS Spoofing* (Izaguire) builds upon Joe Stewarts code, by actually implementing the Birthday Attack – the tool created is a good program to use to perform audit testing on the DNS server.

In looking at best practices for IT organizations, there are sites such as ISACA where the international standards, such as that for auditing like the COBIT standards (see *Control Objectives, Framework, Executive Summary, Management Guidelines* and *Implementation Tool Set*) can be found. COBIT provides control objective information for auditing information systems. Being a government organization, there are several standards which need to be adhered to. These include the *Commonwealth Protective Security Manual* (Attorney-General's Department), the *Australian Communications-Electronic Security Instruction 33 (ACSI 33)* by the Defense Signals Directorate (see *Australian Communications-Electronic Security Instruction 33 (ACSI 33)*), the

NOIE Security standards (NOIE), the Defense Signals Directorate's *Evaluated Products List (EPL)* (see *Evaluated Products List (EPL)*), the *Gateway Certification Guide* (see *Gateway Certification Guide*) and the Office of Information Technology's *Information Security Guidelines* (see *Information Security Guidelines for NSW Government Agencies September 2001*).

The Commonwealth Protective Security Manual (PSM) sets out such guidelines on risk assessment and how to measure risk, and the requirements that government organizations must apply by in relation to Information Systems, it also defines the security classification of data. The PSM provides a risk assessment matrix which can be used to evaluate risk in terms of threats, and is shown in the following table.

	Consequences				
Likelihood	Extreme	Very high	Medium	Low	Negligible
Almost certain	Severe	Severe	High	Major	Significant
Likely	Severe	High	Major	Significant	Moderate
Moderate	High	Major	Significant	Moderate	Low
Unlikely	Major	Significant	Moderate	Low	Trivial
Rare	Significant	Moderate	Low	Trivial	Trivial

Table 1 – Risk Analysis Matrix – From
Commonwealth Protective Security Manual (Attorney -General's Department, p. B -26)

The ASCI 33 regulations is series of handbooks on security for an organization, from physical security to IT security, it provides a similar risk analysis matrix to the PSM. This can be found at http://www.dsd.gov.au/infosec/acsi33/HB3.html. The Evaluated Products List is used for when a government department may wish to make a purchase of equipment, it provides a list of products which have been tested and meet the specifications required by Australian Government Organizations. The Gateway Certification Guide is from the Defense Signals Directorate and examines the need perimeter (or gateway) security and has been created as a guide for government offices to meet industry best practice. It also provides a risk assessment method and risk assessment matrix similar to that of the Commonwealth Protective Security Manual.

There are other Australian standards that should be applied to IT organizations, these are the AS/NZ 7799 (formerly AS/NZ 4444) and AS/NZ 17799 standards which is equivalent to the British 7799 standard and the international 17799 standards respectively. The AS/NZ 7799 standards are composed of two main parts which government organizations need to adhere to – AS/NZ 7799.1:2003 (see Standards Australia, *AS/NZ 7799.1:2003*) which defines the code of practice, and AS/NZ 7799.2:2003 (see Standards Australia, *AS/NZ 7799.2:2003*) which defines the specification of information security management systems. In examining the standards used by most IT organizations in Australia, the 2003 Australian Computer Crime and Security Survey (see *2003 Australian Computer Crime and Security Survey*, pp.4-5) showed that of those surveyed, the Australian Standard AS/NZ 17799:2001

and AS/NZ 7799.2:2003 were the most commonly used Information Security Standard, followed by the Australian Communications-Electronic Security Instruction 33 (ACSI 33) standard. The percentages of each were AS/NZ 17799 91%, the AS/NZ 7799 55% and the ACSI 33 33%.

The Office of Information technology's Information Security Guidelines also covers risk assessment and security in reference to Australian standards. It also provides some good examples of threats and vulnerabilities (see *Information Security Guidelines Part 2 – Threats and Vulnerabilities*) and gives a good guide and example on using the AS/NZ 7799 standards. Further standards for Information Security can be found at AusCert under *Information Security Standards* at http://www.auscert.org.au/render.html?it=2248&cid=1920.

A useful checklist which I found in my research is from Interpol. It has a very good high level checklist for Companies in regard to Information Technology crime prevention which covers policies and procedures which should be in place in organizations. The Interpol *Company Checklist* can be found at http://www.interpol.int/Public/TechnologyCrime/CrimePrev/companyChecklist.asp. In conjunction with this there is also the "IT Security and crime prevention methods" from Interpol, which can be found at http://www.interpol.int/Public/TechnologyCrime/CrimePrev/ITSecurity.asp.

The BIND security recommendations and the Australian government regulations and standards were used in building the audit checklist for the technical and procedural audit items respectively.

It needs to be noted that BIND is an application, it sits on top of an operating system and hardware. As a result a full audit of a DNS server can be broken down into an audit of the operating system and the BIND application itself. There are numerous audit checklists for auditing Solaris and Unix systems as well as several books on the subject. A search on the Internet and references from papers point to CISECURITY as maintaining one of the most comprehensive checklists for the Solaris operating system. CISECURITY maintain a number of checklists for other items, such as Cisco Routers and Windows 2000 operating systems. I have suggested the CISECURITY Solaris checklist (CISECURITY) for auditing the base operating system that the BIND DNS software resides.

Apart from the software vulnerabilities of BIND; cache poisoning, DNS ID spoofing, Denial of service attacks and misconfiguration are the other main vulnerabilities. One aspect of the current state of practice of BIND and DNS relates, in particular, to the architecture of the network and the placement of DNS servers to reduce of these vulnerabilities. In general, security principles maintain that internal networks should be separate from an external network (Liu), and should not be visible outside on the Internet. As a result the idea is to have at least two DNS servers - one on the internal network and one on the Demilitarized Zone (DMZ) or Screened Subnet. The internal DNS server is setup to forward requests from the internal clients to the DNS server on the DMZ (Albitz & Liu, pp.383-386). In general the DNS server on the DMZ should

only be recursive for the internal DNS server. This means at least the internal DNS server will not be susceptible to cache poisoning, so that the internal network maintains to function.

Other recommendations by Cricket Liu (Liu) and others (Householder, King & Silva) to prevent Denial of Service and system outages of the name server function are to:

- do not have all DNS servers on a single subnet
- do not have the DNS servers behind a single firewall
- run an offsite DNS slave

The other current practice is to run Split DNS (Albitz & Liu, pp.394-398) – one DNS server for the internal network, and one DNS server for public services. The information on the external DNS server has only information about services you want the public to know about, it does not provide information about the internal network. Split DNS can be used in conjunction with the network architecture for DNS servers mentioned above.

Another important current practice it to have the firewall filter all the traffic to the DNS servers. (Liu)

In terms of the configuration of a DNS server, the main points of current practice which Cricket Liu (Liu) and Doug Sax (Sax) in his paper "DNS Spoofing (Malicious Cache Poisoning)" point out are:

- Using the latest version of BIND
- Restricting and Authenticating Zone Transfers
- Restrict Dynamic Updates
- Turning off recursion or restrict recursive queries
- Running the DNS server under a non-root/administrator account
- Configure Split Name Servers (Split-DNS)

While it may not be possible to always match the current practice model, such as in the design of the network architecture, if the underlying options and principles are followed, then there should be a reduced risk and reduced likelihood of the DNS server being a point of vulnerability.

Assignment 2 – Create an Audit Checklist

Introduction

The auditing of the DNS server has been broken down into two main sections – and audit of the operating system on which the DNS server runs, and an audit of the DNS application itself. The main aspect of this assignment has been the focus of the DNS portion of the server. As a result most of the research for the audit has been concentrating on auditing the BIND application. There are plenty of sources for auditing the Solaris operating system, one of the best is that provided by CISECURITY (CISECURITY) which I suggest basing the operating system audit on. The scope of this audit is on the BIND DNS application.

Audit Checklist for Operating System

The checklist for auditing the Solaris operating system on which the DNS server runs, has been taken from CISECURITY. This appeared to be a comprehensive checklist covering the major aspects of Solaris security. While the audit of the operating system is a necessary step in auditing the DNS server, it only concentrates on the operating system itself. I have not included the CISECURITY checklist as it would only be taking the focus off the DNS audit, and is beyond the scope of this audit.

Audit Checklist for BIND DNS server

The DNS servers operation is critical during general working hours as it is used for authentication of hosts to access certain services that users rely on. For instance XDMCP clients (such as Win32 X window emulators), corporate web service access, corporate database access, as well as general internal services such as NIS, NFS, and some email systems. If the DNS was brought down during the day, there would be loss of service to users. Similarly nightly backups of the servers require the DNS to be operational. If the DNS server was down during this time, backup failures may occur. As a result our window for the audit must occur outside of these time windows. In general working hours are from 8:30 am to 5:00 pm weekdays, and the nightly incremental backups start around 8:00pm and lasts for a couple of hours. The full backup occurs once a month and takes several days to complete due to the amount of data being backed up.

This leaves the general time frame from 5:00 pm to 8:00 pm of a night, or from around 10:00pm to 8:30am during the night (apart from when a full backup is occurring), or weekends. Noting this however, many of the audit checklist items will not cause a system outage, and as a result they can be performed at any time. However for those items which may cause a system failure or outage, the audit items will be performed during the time window described.

When auditing the DNS server, we must also take into account the relationships the DNS server has with other hosts on the internal and external

networks. This will for instance provide us with the IP addresses and names of hosts we should perform tests against. The policy of the organization and government regulations also affect the checklist.

In auditing the DNS server, some traffic may need to be captured between the DNS server and other hosts. In order to do this, the test machine must be placed on the same physical segment as the DNS server. However, as all machine on the local network are connected via a switch, in order to capture traffic, we will place a hub on the link between the DNS server and the switch. In this was we can connect our test machine to the hub to capture the packets. See figure (Figure 14). Alternatively we could configure the switch to send the data sent to the DNS server to the port of the test machine. The Primary DNS server being tested is dnsserver1.nsw.acme.research.gov.au and it's IP address is 171.93.20.11.

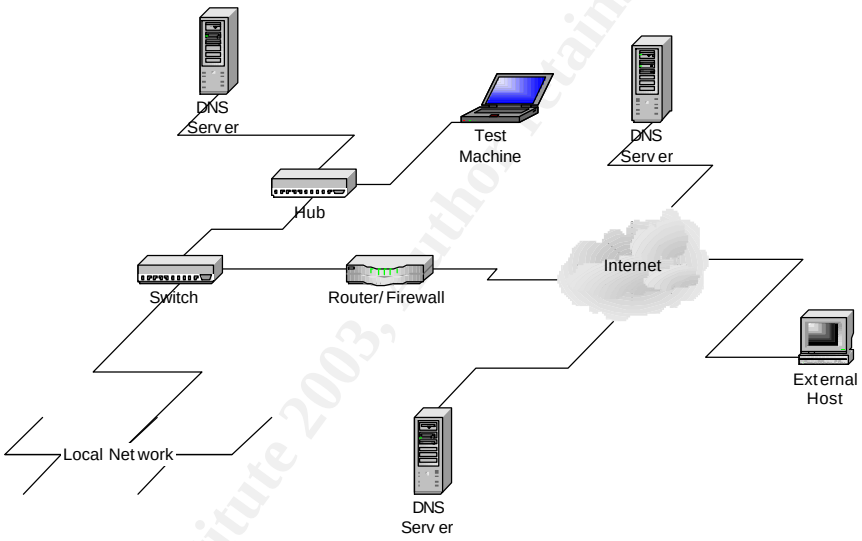

Figure 14 – Setup of testing equipment

Item 1 - Logging is turned on for DNS

References

Information of logging configuration and suggestions can be found in *DNS and BIND* (3rd Ed) by Paul Albitz and Cricket Liu (Albitz & Liu, pp.147-156).

The government regulations described in the Commonwealth Protective Security Manual - section C 7.35 (Attorney-General's Department. p.C-51) and the ACSI33 standard (see *Australian Communications-Electronic Security Instruction 33 (ACSI 33)*, pp.13-4–13-6,13-11–13-12) required logging for monitoring and audit purposes. The AS/NZ 7799 standards – section 4.6.1.3 (see Standards Australia, *AS/NZ 7799.2:2003*, p.11) and section 12.1.7 (see Standards Australia, *AS/NZ 7799.1:2003,* p.75) also recommend logging be used. Government regulations define that audit trails must be kept for a minimum of 1 year (Attorney-General's Department)

Control Objective

Logging is turned on. There are various amounts of information that may be logged. In our case we want logging is set to record successful and unsuccessful zone transfer attempts, all query attempts and any unexpected incidents (Albitz & Liu, p.322).

Risk

There are several risks involved in not logging information. If someone tries to attack the server, if there is no log information we may not detect the attack. Logging provides a way of monitoring the level of activity on our server. It is also a legal responsibility of the organisation to comply with government regulations in regard to monitoring and providing an audit trail. An example of insufficient or no logging can be seen in a recent case study from the 2003 Australian Computer Crime and Security Survey (see *2003 Australian Computer Crime and Security Survey*, p.23). One company had their computer systems compromised, as they had minimal and insufficient logging, the company was unable to reconstruct the attack as no logs were available. As a result of this, investigations by the police were impeded. This case shows the importance of logging not only in terms of monitoring, but also in terms of forming an audit trail when legal investigations are pending.

Another aspect of logging of the DNS server is that if logging is turned on misconfigurations of the DNS server will be reported. For example, if we see a valid zone transfer to a host which we don't know, then there is a problem with the configuration of the DNS server. Errors such as incorrect serial numbers on zone files between servers may also show up.

There are various other logs that can be recorded, however these are the main items we would like to see as they correspond to the main areas where attack may occur.

Compliance

While checking if logging is enabled or not is a binary value (either yes or no), the extent to which logging is enabled can vary. In this case we want to check

that most items are logged. Specifically zone transfer success/failure, queries and unexpected responses. As a result the testing was broken down into sub components of logging. These individual tests will produce binary values relating to each to the main logging concerns.

Testing

As there are different logging levels available, and various types of events that can be logged, we will be testing for several important items. The named.conf file can be checked for location the location of the log file, and also for the configuration of the level of logging. The logging levels can be checked by generating queries/zone transfers and other types of stimulus and the logs examined.

- **authorized and unauthorized zone transfers are logged**

  To test logging we will attempt to do a zone transfer from an internal host, and external host, an trusted slave server. To do this we will logon to each of these machines and run the following commands:

```
% nslookup
> server 171.93.20.11
> ls –d nsw.acme.research.gov.au
```

  The expected response we want to observe is that the internal and external hosts have an unauthorized attempt recorded, and the slave DNS server has an authorized attempt recorded in the logs.

- **all unauthorized query attempts are logged**

  It is possible to have logging enabled for all queries - allowed and denied. However while this information is useful, this may fill the logs with an unnecessary level of information and will cause the logs to fill quickly. In general the main information we would like is unauthorized

  To test this we will attempt to perform several DNS queries from internal hosts and external hosts. To test this we will logon to each of the machines and run the following commands:

```
% nslookup
> server 171.93.20.11
> www.microsoft.com
> nsw.acme.research.gov.au
```

  alternatively, we could run another application, such as ping, telnet or start up a web browser and connect to a machine. In fact any application which allows a user to specify an Internet host to contact would be able to cause a query to occur – if the DNS server address was configured for the host performing the test.

  The results of this test are dependent on whether queries are restricted or

not. If no queries are restricted then there will not be any unauthorized queries logged.

- **unexpected DNS responses are logged**

  There are a couple of ways to test this, one would be to craft a DNS response packet, the other which is much simpler, is to spoof the DNS servers IP address in a query to another DNS server. In the second method, if we are testing from an internal host, we need to take the DNS server we are testing out of our DNS server search list (on UNIX the /etc/resolv.conf file, in windows under the TCP/IP properties of the network interface). If we are testing from an external host, the primary DNS server will not be set as the one we are testing. Due to access lists on the router which prevent spoofing of source addresses outside of the internal address range, it is easier to test the server from an internal host.

  In general there are two different types of unexpected responses. A response to the DNS server for a query it did not initiate, and a response to a query from the DNS server, but coming from a host or DNS server it was not expecting the response from.

  To test this I have modified Ramon Izaguire's (Izaguire) script several ways – the main section of code is as follows:

```
# build query packets skeleton...
my $packet_q = Net::DNS::Packet ->new($domain);
my $restpacket_q = substr($packet_q ->data,2);
my $udp_q = new Net::RawIP({ip=> {daddr=>$daddr}, udp=>{dest=>$dport}});

# build response packets skeleton and more...
my $packet_r = Net: :DNS::Packet->new($domain);
$packet r->push("pre",rr add($domain . " A " . $fakeip));
$packet_r->header->qr(1);
my $restpacket r = substr($packet r ->data,2);
my $udp_r = new Net::RawIP({ip=> {saddr=>$saddr, daddr=>$daddr},
udp=>{source=>$dport, dest=>$spor t}});

for (0..($m - 1))
 { $anyip[$_] =
sprintf("%d.%d.%d.%d",171,93,20,int(rand(256)));
   $anyport[$_] = sprintf("%d", int(rand($port_range)+1024));
   $anyid[$_] =   pack ("H*", sprintf("%.4x", int(rand($id_range))));
 }
```

  To try several different unexpected responses, the first way has been to change the response packet so that the domain name is different to the query packet:

```
my $packet_r = Net::DNS::Packet ->new($otherdomain);
$packet_r->push("pre",rr_add($otherdomain . " A " . $fakeip));
```

  Then try a combination of *$otherdomain* and *$domain* in these locations.

I have also made some changes so for when running an internal query an internal IP address appears in the request. The reason for this is that if recursion is denied for external hosts, none of the other queries will get processed.

I have also modified Ramon Izaguires code for the birthday attack, and commented out the sending out of query packets, just leaving the response packets being sent. I have called the script testdns.pl.

```
./testdns.pl 203.23.194.11 171.93.20.11 6321
www.microsoft.com.au 203.23.194.11 5000
```

Ramon Izaguires program is used to spoof queries and responses. In this case we are just spoofing responses. The name server and domain are real, usually you would give the wrong IP address if you were trying to poison the cache. However we are not trying to poison the cache so we are using the real IP and real domain. This test will appear as responses from www.microsoft.com.au is sending responses. However, in general the DNS server usually logs for an unexpected response to a query, rather than to just unexpected unsolicited response packages, and in general occurs when a reverse address lookup is performed and the hostname is returned from the wrong address.

As a result to test this we need to perform a reverse lookup on a host for which we know the details of the hostname are recorded in the wrong reverse IP address lookup table.

The expected response from these tests is that we see the unexpected responses recorded in the log file.

- **unauthorized attempts to update the DNS are recorded (this also tests dynamic updates)**

  If dynamic DNS were enabled on the server, we want to stop unauthorized hosts from updating the DNS. If the DNS server does not support dynamic DNS, an attempt to update the DNS should still be recorded. To test this, a Microsoft Windows 2000 or Windows XP host can be used. Under the TCP/IP properties of the interface there is a checkbox stating whether or not the machine should register itself with the DNS. By turning this on, the machine will try and send dynamic DNS updates to the DNS server if the DNS server is set as the Primary DNS server in the DNS properties.
  For other tests that can be used see Audit Item on Dynamic Update Restriction.

  The expected observation is to see a record in the logfile showing an unauthorized attempt to update the DNS occurred.

- **Log details are kept for a minimum of 1 year.**

  Government regulations state that all audit information must be kept for a

minimum period of 1 year. Computer systems generally do not log for this long. In general logs are rotated regularly every couple of weeks. As a result, the only way to ensure log details are kept for a minimum of 1 year is to check the backup of the system. This audit item relates to the procedural requirements. Tests can be found in audit Item 2.

Objective/Subjective

The above tests are objective as they apply to any DNS server, can be easily reproduced and have a binary outcome of either meeting logging requirements or not.

Item 2 - Backup of Data and the System

References

The Commonwealth Protective Security Manual - section C 7.35 (Attorney-General's Department, p.C-51), the ASCI33 standards (see *Australian Communications-Electronic Security Instruction 33 (ACSI 33)*, pp.12-4,12-5) and the AS/NZ 7799 standard – section 4.6.4.1 (see Standards Australia, *AS/NZ 7799.2:2003,* p.12). These standards and government regulations define practices that the organisation should observe in relation to disaster recovery, as well as audit trail information.

Control Objective

This objective is examining whether the organisation complies with government regulations regarding disaster recovery and audit trail retention period. This objective could be considered more an operating system audit item, however as we have examined logging in relation to the DNS server and providing an audit trail, it seems appropriate to include this item.

Government regulations (Attorney General's Department) stipulate that audit trails must be kept for a minimum of 1 year. As a result we are testing two things – that a backup of the server is occurring, and that the retention period of the backup data is at least 1 year.

Risk

The risk of not backing up information has several implications. There is the fact that if the data is not backed up, then if the system fails, there may be no way to recover from it, or recovery will take longer than if a backup was available. Secondly there is a legal and government regulation for keeping log and audit trail information a minimum of 1 year. If this audit trail information is not kept, and an attacker is able to compromise your system and goes undetected for several months, and you want to prosecute the attacker in a legal court, then you will need to have a backup of the audit information, so that the crime can be reconstructed. If you don't have this, then there is the likelihood that the prosecution will fail as there is not enough evidence. In respect to audit trails as well, if an attacker gains access to your system, and uses it to attack another organisation, it may appear as if your organisation has perpetrated the attack. If you don't have audit trails to fall back on to discover the attack was really from an outside party, the organisation may be held liable for damages.

Compliance

The audit test is testing for binary responses. Firstly, that the DNS server is either being backed up or not, and secondly that the backup information is held for 12 months or more.

Testing

To test whether a backup is occurring and how long the data is being held for, we can examine the backup logs and the configuration of the backup server.

The backup server is running Legato Networker. Backup reports are emailed

to the system administrators after each incremental and full backup with a status report. The backup software also allows interrogation of the backup information. To determine whether a backup is occurring for the DNS server, we can examine the email, or interrogate the backup server. We will do the later test. To interrogate the backup server the following command can be used:

```
% mminfo –c dnsserver1.nsw.acme.research.gov.au
```

This command will print out all the information regarding the backup of the DNS server.

To test for the retention period, the configuration of the backup server for the DNS server can be examined. However to test this, we can examine the contents of the tapes which were backup up 12 months earlier. This can be done via examining the information on the tape volumes in the database for tapes which have not been recycled:

```
% mminfo –v –q 'level=full,savetime <= 52 weeks ago'
–c dnsserver1.nsw.acme.research.gov.au –o t
```

Objective/Subjective

The result of this audit step is an objective one, tests can be regularly performed to come up with an outcome that will either comply or not comply with the audit control objective.

Item 3 - Statistics Enabled and Dumped Regularly

References
Information regarding the enabling of DNS server statistics can be found in *DNS and BIND* by Paul Albitz and Cricket Liu (Albitz & Liu, pp.163-172).
While the book highlights the use of statistics in terms more for performance, from my reading of the book, I found that statistics of the DNS server when enabled can provide a lot of useful information in terms of monitoring the server and in terms of auditing the server. By having statistics enabled, a baseline over time can be made of normal DNS activity.

Control Objective
The objective of this audit item is to determine whether statistics are enabled on the server, and if so what level and to ensure that they are dumped regularly.

Risk
The risk of not having statistics turned on is low. There is no necessity to have it turned on in terms of providing security, however it does provide a good monitoring and audit mechanism as it provides statistics on the number of queries and zone transfers it has, and provides details about hosts contacted. Given this, the use of statistics on the DNS server means that over time a baseline of standard DNS activity could be defined, so that thresholds could be set and events triggered if the threshold was significantly exceeded, in a similar manner to that of how traffic analysis may have a baseline.

Compliance
The DNS server will be compliant if the statistics option is found to be turned on and dumped regularly. There are several levels of statistics dumping. If the minimum or default level is turned on then the DNS server will be compliant. The regularity may vary, however at minimum once a day would be considered compliant, although every hour, which is the default option is more useful.

Testing
To test whether statistics is enabled and turned on, the system logs can be examined. If statistics is enabled, a dump of statistical information will occur at regular intervals. These intervals are defined by the configuration file (/etc/named.conf) for the BIND DNS server. In some cases BIND may not have been compiled to allow for statistics.

We can also examine if statistics is possible by running the following command (assuming the default location of named.pid has not been changed):

```
kill -ILL `cat /etc/named.pid`
```

If statistics is enabled, then the output should be dumped to the file “named.stats” should be created in the working directory of bind. The working directory of BIND can be found by examining the “directory” value under

"options". i.e.

```
options {
          directory "/var/bind";
        };
```

To test if the file has been created use the following command:

```
ls -al /var/bind/named.stats
```

To examine if regular dumps are occurring the log files should be examined to see how often the statistics are dumped. The log files will also show the level of statistics enabled.. The level of statistics can also be found by examining the contents of the /var/bind/named.stats file. If full statistics is enabled, then there will be host information with statistics for each host. If the default level of statistics is used, then only the statistics for the DNS server itself will be shown in the file.

Objective/Subjective

As the compliance of this test shows, the test indicates a yes or no value that can be tested. As a result this test is objective.

Item 4 - Bind Version Number

References
There are several references about the bind version number such as *Maximum Security* (Anonymous, 2001, pp.484-487), *Maximum Linux Security* (Anonymous, 1999, pp.283-285) and *Hardening the BIND DNS Server* (Boran). The reason for this audit item is to determine the version of bind running, and to determine what vulnerabilities may exist for the version (Faure).

Control Objective
To determine what version of BIND is running in order to research the vulnerabilities associated with the version. This also allows us to find out what other people on the Internet can see about our version. The main aim is to avoid people finding version, so the version details should be obfuscated.

Risk
There is a risk if the version information about the BIND server is available to external parties or anyone on the Internet. This means that if there was a malicious user, they could examine the version number and then search the Internet for vulnerabilities associated with it. Using the version number and the vulnerabilities they can plan an attack. The risk associated with the bind version number is related to information reconnaissance.

As the version number does not cause any denial of service or exploit directly, the risk of allowing the version number out is low.

Compliance
The DNS server will be compliant if the chaos txt bind version query returns no useful information about the BIND version running.

Testing
To test for the bind version, there are several tests, the first is to use dig to query the version number. The second alternative is to run the named program with the option to print the version. We will do both tests.

```
% nslookup –q=txt –class=chaos version.bind
  acme.research.gov.au

% dig @acme.research.gov.au version.bind txt chaos
```

This first test is a test anyone outside on the Internet can perform where as the second test is one which only a user logged onto the computer can perform.

```
% named –v
```

Generally, the expected observation from these is the version of bind returned. However there are options for returning the bind version as some unmeaningful string. For instance there is an option in BIND to change the

version details returned via a dig or nslookup command. To stop information leakage, the version number should not provide any meaningful data an attacker can use against the DNS server. (Boran) From bind 8.1.2 the "version" option was included (ISC:d).

```
options {
     version "you don't need to know";
};
```

Objective/Subjective

This audit item is objective as it is returning just the version details.

Item 5 - Recursion is turned off by default or is restricted to authorised hosts only

Reference
Many CERT vulnerabilities provide recommendations for turning off recursion as a work around to some BIND vulnerabilities. Recursion is also a major factor in DNS cache poisoning. If recursion is turned off, the query answer is not cached on the DNS server. Other references are Cricket Liu's *Securing an Internet Name Server* (Liu), *DNS and BIND* (Albitz & Liu, pp.246-248,255-256), and Cheng C.Teoh's *Defense in Depth for DNS* (Teoh).

In general whether recursion is allowed depends on the location of the DNS server – whether it is internal behind the firewall or outside the on the DMZ. This will be subjective in terms of network architecture, organisational policy and requirements, and the number of DNS servers in the network.

In terms of best practice, there are usually two DNS server, on the internal network, and another on the DMZ. The internal DNS server doesn't do recursion, but forwards queries to the external DNS server on the DMZ (Albitz & Liu, pp.383-386), or if only one DNS server exists, then it should be recursive for internal hosts, but not external hosts.

As there is no external DNS, recursion may be turned on, but it needs to be restricted to internal clients only.

Control Objective
To determine whether recursion is turned on or not, and if it is, then to determine whether recursion is restricted to authorised hosts.

Risk
Allowing recursion means that DNS cache poisoning can occur. This is a high risk. By restricting recursion to internal clients and authorised hosts, this reduces the risk of cache poisoning occurring.

All external DNS queries should not be recursive. This way an attacker can not actively poison the DNS cache by sending a DNS query to your DNS server which asks the attackers DNS server. The only way an attacker can poison the cache is passively, if an internal client asks the DNS server to resolve an IP address and asks the attackers DNS server.

Compliance
This audit item determines recursion is on or off, and if it is on, verifies that the DNS server will only do recursion with authorised hosts. The DNS server will be compliant if recursion is found to be off, or if recursion or recursive queries are found to be restricted to authorised hosts (generally hosts on the internal network).

Testing
By default recursion should be turned off for all hosts. This can be done via the configuration file, which should have the following:

```
options {
     recursion no;
     };
```

If recursion is required, then it should be limited to a set of hosts. This can be done via acls in BIND 8, and associating recursion with a host list.

```
acl internal_hosts {
     network/mask;
     };

options {
     recursion { internal_hosts; };
     }
```

To test recursion, we can use several tools, we will use both nslookup and sam spade as two different tools.

Several different tests need to be used. Firstly we will test the DNS server by sending a query with recursion turned on from an external host, then from an internal host. For the external host recursion should be denied, for the internal host, recursion should be allowed.

```
% nslookup
> set recurse
> server 171.93.20.11
> www.microsoft.com.
```

The responses from nslookup should show whether recursion was allowed or not. Similarly we should see some entries in the logfiles showing the same if logging is enabled.

A tcpdump of the query will also show the DNS flags which indicate whether recursion is used or not.

Sam Spade can also be used to test the DNS server for a recursive query. It also provides a window with response messages.

Objective/Subjective
This test is objective as the results of the test are distinct and reproducible.

Item 6 - Zone Transfers Restricted

Reference
Most of the papers which discuss security of a DNS server recommend restricting zone transfers to authorised hosts. References include *DNS and BIND* (Albitz & Liu, pp.252-253), *Securing an Internet Name Server* (Liu), *Defense in Depth for DNS* (Teoh), *Counter Hack* (Skoudis, p.171), *Maximum Security* (Anonymous, 2001, pp.61,487), *Hacking Linux Exposed* (Hatch, Lee & Kurtz, pp. 81-86), *Why is securing DNS zone transfer necessary?* (Lau), *DNS Security* (Holland) and *Softpanorama DNS Security Page* (Softpanorama).

Control Objective
The objective of this audit item is to ensure that zone transfers are restricted to trusted hosts.

Risk
Allowing zone transfers to unrestricted hosts means that information about the computers on the network and the network itself can be send to unauthorised machines. This information can be used to plan a coordinated attack against the network. This risk is medium. While allowing zone transfers to unauthorised hosts is a security risk, in terms of damage to the system or denial of service via system outage, in general the worst that normally happens is that information about the network is leaked. However, there are some attacks which can cause a denial of service attack by having zone transfer access. One case is where the an attacker may cause a large number of zone transfer queries to occur, causing the DNS server resources and network bandwidth to be used up. One point to be noted is that the restrictions to hosts is also done via an IP address for host authentication. If an attacker can spoof an IP address and DNS messages and route replies through their machines, then they can effectively become trusted. The firewall/router can be used to stop source routing to reduce the likelihood a reply packet will go past the attackers machine.

Compliance
The results of this audit item will show that either zone transfer are allowed to unauthorised hosts or they are not allowed. The DNS server will be compliant if the zone transfers are only restricted to other trusted DNS servers.

Testing
Zone transfers are restricted in the /etc/named.conf file by specifying what hosts are allowed to do the transfer via the allow-transfer option in the zone configuration.
For example,

```
zone "test.acme.research.gov.au" {
        type master;
        file "test.db";
        allow-transfer { trusted ip; };
};
```

To test restricted zone transfers, we will attempt to do a zone transfer from several hosts

- a zone transfer from an internal host but which is not trusted for zone transfers
- a zone transfer from an external host which is not trusted for zone transfers
- a zone transfer from a trusted host.

To perform a zone transfer there are several tools which can be used – nslookup, dig, sam spade and named-xfer.

```
% nslookup
> server 171.93.20.11
> ls –d nsw.acme.research.gov.au.

% dig @ nsw.acme.research.gov.au . axfr
```

named-xfer is the program which BIND or in.named calls to perform zone transfers. This can be used to manually perform a zone transfer. (Albitz & Liu, p.301). This test however can only be performed on a DNS server with the software installed.

```
% named-xfer –z nsw.acme.research.gov.au –f
/etc/bind/db.nsw.acme.research.gov.au –s 0 dnsserver1
```

via sam spade, the zone transfer option can be chosen.

Objective/Subjective

This test is objective as the outcome of whether the DNS server is compliant or not, is clearly visible from the results, which can be easily reproduced.

Item 7 - Zone Transfers are authenticated with transaction signatures

Reference
Securing an Internet Name Server (Liu), *Security Complete* (Lierley, p.482-495), *Defense in Depth for DNS* (Teoh), *Why is securing DNS zone transfer necessary?* (Lau), *Hardening the BIND DNS Server* (Boran) and *Softpanorama DNS Security Page* (Softpanorama) are among the many references which recommend transaction signatures for authorising DNS server communication between two DNS servers in order to authenticate each endpoint.

Control Objective
This audit test is to examine whether before zone transfers occur, the hosts are authenticated.

Risk
If zone transfers aren't authenticated via checking the host asking for the zone is who they say they are, then we could possibly be transferring zone information to an authorised host who is appearing to be a valid host. This risk associated with sending out our zone information is that some unauthorised user will gain knowledge of our computer network and hosts which are on it. On the other hand, if our primary DNS server is a also a slave for the corporate network zone information, then if the attacker can spoof the corporate DNS server and send us bogus information about corporate servers, they can either cause a denial of service attack, or send hosts to a malicious server to capture usernames and passwords for instance. A main in the middle attack could also occur capturing the corporate information. In these instances the risk is higher. Not having transaction signatures for authorisation for internal DNS hosts is less risk as the firewall should prevent spoofing of internal hosts coming in. The risk of host authentication is more associated with zone transfers across the Internet, as a result the two hosts should have authentication turned on.

The likelihood of this of one of the slave DNS servers or one of the corporate DNS servers being spoofed and requesting or sending zone transfer information is low.

Compliance
The outcome of this audit test item is a binary response as we are looking for whether hosts performing zone transfers authenticate or not. However there is the case where we may have some zone transfers being authenticated and others not. This depends on the design and architecture of the network. However in general authentication should be turned on at least for those zone transfers which occur externally to the local area network over the Internet. To be compliant the DNS server should use authentication of hosts which are not on the internal or local network before a zone transfer is performed.

Testing
To test whether zone transfers are authenticated with transaction signatures we need to look at the DNS traffic for a DNS zone transfer. To do this test of

zone transfer authentication a tcpdump of the traffic between two DNS server when a zone transfer is performed can be done. If the two DNS servers are using transaction signatures, we should see this in the tcpdump output, there should be content in the packets indicating the key algorithm being used, before the zone transfer occurs.

Step 1: start tcpdump on packet sniffing host.

```
% tcpdump –w /tmp/zonetransfer.out –n host dnsserver1
```

We use the –n option so that tcpdump does not try to resolve names, as this would just cause more traffic for the DNS server that we would capture. A lookup of the names can be done when we read the file later on in step 3.

Step 2: On trusted host, such as the slave server, perform a zone transfer.

To perform a zone transfer, if notify is turned on rather than polling, then by updating a DNS record's serial number and reloading the file should cause a notify message to be sent to the slave to do a zone transfer of the file. Otherwise you may need to wait for the slave to poll the master server. You can check when the zone transfer has occurred by looking at the time stamp for the zone file. When it changes you can assume a zone transfer has occurred to update the file.

Alternatively, rather than waiting for polling if notifying is not used, then the named-xfer file can be used to do the zone transfer after the serial numbers of the zone file have been updated.

```
% named-xfer –z nsw.acme.research.gov.au –f
/etc/bind/db.nsw.acme.research.gov.au –s 0 dnsserver1
```

Step 3: examine output of tcpdump, and repeat the steps for all the hosts which the DNS server performs zone transfers. Due to a DNS server generally getting lots of traffic, you can filter the output file. The –X option allows us to view the packets in ACSII as well as hex. This allows us to search for strings such as HMAC or MD5 which will indicate authentication of the zone transfer.

```
%tcpdump –X –r /tmp/zonetransfer.out src host dnsserver1
&& dst host dnsserver2 || src host dnsserver2 && dst host
dnsserver1
```

Objective/Subjective

The test of zone transfer authentication is objective as the outcomes are either meet or fail the control objective.

Item 8- Limit the number of zone transfers that can occur at any one time

Reference
While Paul Albitz and Cricket Liu in their book *DNS and BIND* (Albitz & Liu, p.233-236) refer to limiting the number of zone transfers for performance issues of the server. It can also be looked at in terms of preventing denial of service attacks causing the DNS server to run out of resources.

Control Objective
To check that the number of zone transfers that can occur at any one time are limited.

Risk
If there is no restriction on zone transfers, then the risk of a denial of service attack against the DNS server via a flood of zone transfer requests is high. However if the zone transfers are restricted to known authorised hosts, then the risk of a denial of service attack is reduced significantly. However a spoofing attempt if transaction signatures are not used can be used to perform such an attack. This however is also dependent on being able to predict DNS ID's. Therefore the overall risk, while dependent on the results of other audit items can be considered low, as the likelihood of an occurrence is low, if the other audit items are met.

Compliance
As limiting the number of zone transfers is related to the performance of the DNS server, it can vary depending on the type of hardware and operating system the DNS server is running on. In order to ensure the correct limit is set, the DNS server should be monitored via the use of the built-in statistics of BIND and a baseline created in order to determine a threshold between where the DNS server responds well and where queries and zone transfers time out. While there is no exact or correct number that can be specified for every DNS server, a small number or limit such a single digit should ensure the server is not over-loaded at any one time. If the DNS server performance is not affected too much and the DNS server maintains operation, then the DNS server will be compliant.

Testing
In testing this audit item as this test may effect the DNS servers usage, we must ensure that we observe the time constraints in testing so that minimum disruption to the users and the network is caused.

This test can only be performed with a slave or master server which is allowed a zone transfer with the DNS server under examination. The hosts used in this test are dependent on the results of the audit item on restricting zone transfers.

On a slave DNS server, the following simple perl script can be run:

```
#!/bin/perl
foreach i (0..5000)
{
```

```
        system("dig \@171\.93\.20\.11 acme\.research\.gov\.au afxr \&");
}
```

To make this test more effective it would be good to run this simultaneously on multiple DNS servers.

The resources on the DNS server being tested should be monitored to see how well it copes. To test the resource usage on the DNS server, we can run system tools such as “top” to examine memory and CPU resource usage and “ntop” for network resources. If statistics is enabled, we can perform a statistical dump of the server resources. We can also examine the resource usage by using “dig” to perform queries and examine the time taken to respond to the queries. Unlike dig, nslookup does not provide information on the time it takes for the DNS server to respond to a query.

Objective/Subjective

This audit item should be considered subjective as it depends on many factors, such as the network speed, and the hardware configuration of the DNS server.

Item 9 - Limit queries of non-public information to internal clients or trusted hosts

References
Most of the documentation on the security of a DNS server - such as *DNS and BIND* (Albitz & Liu, pp. 394-398), *Counter Hack* (Skoudis, pp.165-172) recommend separating internal private information from that which is available to the public. Generally this is done through having multiple DNS servers set up with what is termed Split-DNS. However BIND 8 allows for queries to zones to be restricted to certain hosts or networks.

Control Objective
To test that internal information is not accessible to the public.

Risk
The risk associated with this test is dependent on the network architecture and design. In our case the DNS server is serving public and internal hosts. As the hosts in the network all have their own IP address which is not in a private address range the information that an attacker would get is the same as if they had access to perform a zone transfer, however it would take them longer to create as they would have to examine each IP address and hostname individually via nslookup or other tool.

The risk is also dependent on the type of DNS information being held. For instance if the internal information provided HINFO, then this would be more risky than if no HINFO records were associated with the DNS data maintained for the internal network. Generally, the risk associated with this is that information about hosts is leaked out. This means an attacker could plan an attack against the network, but no actual denial of service, exploit or disruption will occur if queries are not restricted. The likelihood of getting information is high, but the consequences are low, so over all it is a low risk.

Compliance
This audit item has a binary response in terms of restricting or not restricting queries. However the responses will vary and be determined by what zones may be restricted. In general if the DNS server does not allow information about the internal network out, then the DNS server is compliant with the audit item.

Testing
To test this item we need to try and get information about internal hosts from externally to show that information is not accessible, and then from an internal host to show that only internal hosts have access.

In this particular case, we know that the most of the internal network information is under nsw.acme.research.gov.au, while the publicly available services are under acme.research.gov.au – such as www.acme.research.gov.au. As a result the two zones are separate. So we should restrict queries to nsw.acme.research.gov.au to the NSW network and the other divisional networks and corporate networks.

To perform this test, we will perform queries from internal and trusted hosts, and from external hosts. If the audit item is met, then internal and trusted hosts will be able to perform a query on nsw.acme.research.gov.au as well as acme.research.gov.au and we would expect external hosts not be able to query nsw.acme.research.gov.au, but only acme.research.gov.au. Trusted hosts would include other DNS servers doing a query for a client, this must be taken into account when restricting queries.

From an internal host and trusted host, then from an external host:

```
% nslookup
> server 171.93.20.11
> nsw.acme.research.gov.au.
> acme.research.gov.au.
```

Objective/Subjective

While the test for limiting queries is objective, the actual results depend on whether the DNS server has been setup to contain non-public information. If it does then this test is objective, if it is not then the results of the test are subjective as to whether the information being handed our is private or public. It is a decision for System Administrators and IT management to make in regard to what is public and what is private information and the level of information which is stored in the DNS, and as a result this test is subjective.

Item 10 - DNS Server runs under a non-root account

References
There are several references on the reasons for running BIND under a non-root account, these include *DNS and BIND* (Albitz & Liu, pp. 253-255), *Maximum Security* (Anonymous, 2001, p.486), *Quick guide to BIND 8* (Faure) and *ISS Security Alert Summary* (see *ISS Security Alert Summary – Vol4 No. 9*).

Control Objective
To ensure that the DNS server is not running under the root account.

Risk
The risk of running the DNS server under the root account, is that if the DNS server has a vulnerability, such as a buffer overflow, that allows an attacker to run arbitrary programs, the attacker will be able to run the programs under whatever privileges the DNS server runs at. So if the DNS server is running as root, the attacker has root access. The risk of getting root access is high if a vulnerability exists. The consequences are the attacker controls the DNS server, and does not need to do cache poisoning to effect DNS data. The DNS server not only is a good source of information about the network and hosts, but as all hosts must talk to the DNS server, it is a good way of spying on the type of information people in the organisation are working on, and if the organisation has business partners, analysis of the traffic and which hosts external to the local network used the DNS server the most could indicate who the business partners are and may lead to a backdoor into another organisation. Having root access to the DNS server or any server or host for that matter means an attacker has a launching pad for attacking the local network, or directing attacks to other external sites. If the attacker has installed password sniffing programs, etc., they may gain usernames and accounts of other users which they can use to get into other systems. Alternatively the attacker could modify the DNS configuration file and allow zone transfers to their own sites and create or delete zones.

As a result it is a good idea not to have other services like NIS on the DNS server as this may allow access to other users home directories and if misconfigured allow writable access by root to the disks.

The likelihood of finding a vulnerability that has exploits allowing root access depends on the version of BIND being run. In our case it is version 8.2.7, which many people regard as a safe BIND version as it has bug fixes for previous vulnerabilities involving root exploits, but it may be anytime new vulnerabilities are found. However in the present situation, BIND 8.2.7 is considered to be safe from root exploits, and so the risk of obtaining root access is low.

Compliance
The DNS server will be compliant if it is running under an account other than "root".

Testing

To test if the DNS server is running as root or not is quite simple. We can test this by listing the processes running and see who owns them. Secondly we can support this by examining the configuration file which starts the DNS server.

In examining the configuration file on Solaris 8, we examine /etc/init.d/inetsvc and check the line used to call the DNS server and see if the options of running under a different user account are specified. For example:

```
in.named -u named
```

Secondly, to ensure what we find in the configuration file is correct, and an administrator has not just started it under root from the command line, we can examine the processes running. The BIND server is started vi a program called “named” in some cases it is “in.named”. We need to check that this service is running and under what account.

```
% ps –ef | grep named
```

This will show us whether the “named” or “in.named” process is running and under what account. If this audit item is met, we should expect that the account “root” will not be shown.

Objective/Subjective

This test is objective, either the DNS server process is running under root privileges or not.

Item 11 - Dynamic Updates are restricted if turned on

References
Paul Albitz and Cricket Liu discuss the importance of restricting dynamic DNS updates in their book *DNS and BIND* (Albitz & Liu. pp.231-233) as well as Cricket Liu in his paper *Securing an Internet Name Server* (Liu). In general from my experience BIND DNS servers are not usually configured for dynamic updates.

Control Objective
Dynamic updates should be disabled unless specifically required. In the case were it is allowed, it must be restricted to specific hosts. This test is to ensure that if dynamic DNS is used, it is restricted to trusted hosts.

Risk
Dynamic update allows DNS records to be modified, added or deleted. If an attacker can spoof a DNS IP and appear to be from a zone delegated for dynamic updates, the attacker can basically delete all the records causing a denial of service, or could inert their own records or modify existing records such as the MX record to point to a new mail server.

Due to the nature of dynamic DNS allowing so many changes to the DNS database, it is imperative that if dynamic DNS is used, it is restricted, allowing anyone to do dynamic updates is asking for trouble. The hosts for which dynamic updates are allowed should have their security audited regularly.

While the possibility probably exists for the IP address of a host or DNS server which is allowed to perform dynamic DNS is allowed, being spoofed and malicious changes made, the likelihood is low. Firewall policies should perform ingress and egress filtering to stop IP spoofing and as dynamic DNS is usually involved with the DHCP and Windows machines on the local network, the risk is low.

Compliance
The result of the test will either indicate dynamic dns is on or off is a binary response. Similarly if dynamic DNS is on, the other tests will show whether the hosts tested are trusted or not trusted to perform dynamic updates. For the DNS to be compliant, it must have Dynamic DNS disabled, or must restrict the DNS updates.

Testing
The configuration file should be check to see if dynamic updates are allowed. This can be found by examining the /etc/named.conf file and search for “allow-update” which is followed by a host list. We should see something like:

```
allow-update {
                restricted-ip ;
              };
```

If allow-update is discovered it means the zone for which it references allows dynamic updates.

Another test for dynamic updates is to obtain a Windows 2000/XP machine and under the TCP/IP properties of the network interface, there is an option for registering the computer in the DNS. If this option is turned on, the windows computer will try and register itself by sending an update to the DNS server. If the DNS server is not configured to support dynamic DNS an error message will appear in the logfile. If the DNS server supports dynamic updates but the host is not allowed to update the DNS the same or similar error message will be displayed in the logfile.

Alternatively, the command nsupdate can be used to try and update the DNS server.

```
% nsupdate
> server 171.93.20.11
> zone nsw.acme.research.gov.au
> update add newmachine.nsw.acme.research.gov.au 60
  A 171.93.25.253
> send
```

This test can be run from an untrusted host and a trusted host for dynamic updates. For a trusted host, the DNS entry should be added to the DNS database, so doing an nslookup should return the information.

```
% nslookup
> server 171.93.20.11
> newmachine.nsw.acme.research.gov.au.
```

If dynamic DNS is working, the search should show newmachine.nsw.acme. research.gov.au.

Objective/Subjective
This audit item is objective as the tests can be repeated to achieve the same answers.

Item 12 - Ensure root server information updates regularly

References
Paul Albitz and Cricket Liu (Albitz & Liu, pp.141,332-333).recommend that the cache of root server information be updated regularly to ensure that you always have the correct root server information.

Control Objective
To ensure that the root server information is updated regularly.

Risk
The risk of not updating the root server information is that if the root servers change IP address, and you still have an old IP address for a root server, you could be asking a host of an attacker running a malicious DNS server. They can then poison your cache with wrong information or send you to a fake authoritative server for a domain. The likelihood of this occurring is very low, and as mentioned, the root name servers rarely change addresses.

Compliance
This test checks that the root server information is updated regularly, definition of regular may vary, but in general if the timestamp on the file is greater than one or two months, we can say that the updates are not regular. Paul Albitz and Cricket Lui suggest every month or two or less is acceptable. The DNS server will be compliant if the root server information is updated according to these time frames.

Testing
The root server information is usually kept in the file “db.cache”. To check the file, we can look at the timestamp on the db.cache file. Depending on the time stamp we can say whether it has been updated. The time between updates is subjective and depends on how often the root server IP addresses change. This is not very often, but updating it at least once a month or once week is a good start to being a regular update.

To test the time stamp, we can type the following command in the directory where the db.cache file is located, in our case /etc/bind/PRIMARY.

```
% ls –al db.cache
```

The crontab files can be check to see whether it is an automated task to update the files.

The db.cache file can be updated with the following command (Albitz & Liu, p.141):

```
% dig @a.root-servers.net . ns > db.cache
```

Objective/Subjective

While the time frame for what is considered a regular update may vary, this audit item is determining the fact that updates do occur and on a regular basis. As a result this audit item is objective.

Item 13 - Authoritative Negative caching should be turned off

References
Peter H. Gregory in *Solaris Security* (Gregory, p.197-200) discusses name server caching and some negative aspects of it.

Paul Albitz and Cricket Liu discuss negative caching in their book DNS and BIND in relation to troubleshooting why other name servers don't cache negative responses. (Albitz & Liu. p.327). One aspect to negative caching that is important to note is that by default the option "auth-nxdomain" is turned on in BIND. What this means is that any negative cache information is authoritative, whether you are the authoritative domain for the information or not. It tells other DNS servers to treat your negative responses as if they were from an authoritative server. This provides a way an attacker can cause a denial of service attack via sending negative cache information for other domains. – negative cache poisoning. Paul Albitz & Cricket Liu suggest leaving it turned on as some old name servers may rely on it to work, however since new versions of BIND don't rely on this being turned on, I suggest that it is disabled.

If an attacker forges a replies for a query saying that a host doesn't exist, this information gets stored in the negative cache. Potentially a denial of service attack could be instigated by filling the negative cache. This is similar to cache poisoning, however, rather than poisoning the normal cache we are poisoning the negative cache.

Control Objective
To test that authoritative negative caching is turned off.

Risk
The risk of negative caching is a possible denial of service attack. The important aspect is that negative cache messages are marked as authoritative even if they are not. This causes a flow on effect that other DNS servers will cache the negative responses, because they appear authoritative. The fact that the messages are authoritative means an attacker does not have to be an authoritative DNS running for a domain to have the negative response cached by the resolver program or querying DNS server.

By disabling authoritative negative caching, other DNS servers may not cache the negative cache of the DNS server. This does not stop negative caching all together, but just stops other DNS servers caching the negative cache information.

The likelihood of negative caching attacks occurring is generally low.

Compliance
The result of the test is a binary response that negative caching is on or off. To be compliant the DNS server must have negative caching turned off.

Testing

To test this we can ask the DNS server a for information on an non existent domain via nslookup or similar tool asking that the DNS server do recursion. This should leave a negative answer in the DNS cache. If we try again to look up the same domain, we should get a quicker response as the server has already cached the negative answer, and that answer should appear authoritative if auth-nxdomain is turned on, otherwise it should not be authoritative.

We can also check the /etc/named.conf file to see that the following exists:

```
option {
          auth-nxdomain no;
       };
```

Programs such as nslookup or dig can provide information about the responses returned from the DNS server such as whether it is authoritative or non-authoritative. If negative authoritative caching is turned on, then we should expect to see an authoritative answer in the results of an nslookup or dig. There are several steps to testing whether negative caching is authoritative.

Step 1: From a client query the DNS server for a non existent host using noone.fromnowhere.com as an example – but any hostname could be put here.

```
% nslookup
> server 171.93.20.11
> noone.fromnowhere.com.
```

This first lookup will cause the DNS server to try and resolve the name to an IP address. Once not found, the hostname will be cached in the negative cache.

Step 2: From a client query the DNS server again for the same non existent host.

```
% nslookup
> server 171.93.20.11
> noone.fromnowehere.com.
```

In this second lookup the DNS should respond more quickly as it will be referring to it's negative cache for the answer. Based on the results of this answer if negative caching is authoritative, nslookup will show that the DNS server's answer was authoritative.

As an alternative or augmentable step, we could also turn debugging on in the DNS server which would show how the DNS server answers the query. To do this, in BIND 8, we send a signal to the *named* process:

```
% kill -USR1 `cat /etc/named.pid`
```

This will set the DNS server to debug level 1. Further USR1 signals will cause the DNS server to move to higher debug levels before cycling back to debug level 1. To turn debugging off, send a USR2 signal to the server process.

```
% kill -USR2 `cat /etc/named.pid`
```

A tcpdump of the query should also show whether the DNS server responded with an authoritative answer.

Objective/Subjective

This audit test is objectively testing whether negative caching is turned on or off.

Item 14 - The firewall or router filters traffic to DNS server

Reference
Paul Albitz and Cricket Liu (Albitz & Liu, pp.379-381) discuss how DNS may be configured with a firewall. Householder, King and Silva in *Securing an Internet Name Server* (Householder, King & Silva, pp.5-6) also discuss the need for firewall filtering.

Control Objective
To check that access from external hosts on the Internet to the DNS server is restricted. Specifically the firewall or router should block traffic to the host apart from udp 53 for external host queries, but allow tcp 53 with authorised slaves and masters with which we transfer zone information.

Risk
If the DNS server is hardened correctly, then the risk of not having a firewall or router blocking traffic is low. However, in general it is good practice that packets are filtered before coming in the network. If the firewall did not block other traffic, then an attacker may be able to exploit other vulnerabilities which may exist on other services which the machine may provide or exploits in the operating system the DNS server runs on. The risk of not filtering the DNS server packets is high.

If an attacker can exploit other services, they may be able to gain root access to the machine, or at least cause a denial of service attack against the machine.

Compliance
The expected outcomes from this test are binary responses in terms of a tcp or udp port 53 being open to trusted and non-trusted hosts. We should only see udp 53 open for the external untrusted hosts. Whether other ports are open, such as port 22 (ssh) will also show up as well however. What services are open will depend on the system policies. This however will alert us that the firewall is not filtering the correct information. For this audit item to be compliant, the firewall must only allow tcp and udp port 53 traffic to the DNS server.

Testing
To test which ports are open to the DNS server, we can run a network scan on the host using a tool such as nmap. We will need to run this tool from an external host, and from a trusted DNS server, such as a slave from another site.

From the external host and trusted host, we need to type the following command to check for tcp connections:

```
% nmap 171.93.20.11
```

To check for udp connections, we need to run the following command:

```
% nmap –sU 171.93.20.11
```

This test should be applied to all servers for which access is meant to be allowed.

Objective/Subjective
This audit item is objective, as it is easily repeated and reproducible with the same results.

Item 15 - Fetch Glue is turned off

References
Cricket Liu in *Securing an Internet Name Server* (Liu) and Eric Detoisien in *System Administration: External Attacks* (Detoisien) mention that glue fetching is another way that DNS cache poisoning can occur. Glue records are the records in the DNS that point a domain to a name server. Glue fetching is when records are returned by a DNS server to a requesting DNS server, telling the requesting DNS server what other name servers to contact. In some cases a DNS server does not specify the IP address, as a result the requesting DNS server performing the query will try to get the IP address of the other name server to contact.

Control Objective
This audit item tests that glue fetching for the DNS server is turned off.

Risk
The risk involved with having fetch glue turned on is that an attacker could poison the DNS server's cache. This could lead to a denial of service attack, DNS hijacking or a man in the middle attack.

Compliance
The result of this audit test is a binary outcome. For the DNS server to be compliant, glue fetching must not be enabled.

Testing
In the configuration file an in the options section should exist specifying that the glue fetching is turned off:

```
option {
          fetch_glue no;
       };
```

To test that fetch glue is not enabled, we need to perform a query where a name server is going to return the name of another DNS server, but not the IP address.

A tcpdump of some DNS traffic when the query occurs can be referred to in order to see which DNS servers respond with name server information without an associated address record. This may involve capturing the entire packet with tcpdump so that the information about the name servers can be extracted.

Once the query is finished, if fetch glue is turned on, then for the name server pointed to (without an address) which we have extracted, we should see traffic from our DNS server trying to resolve the IP address. If fetch glue is not turned on, we should not see traffic for a query for the IP address of the name server whose information we extracted.

To ensure the test is repeatable it is best if a test DNS server with an authoritative domain be setup. This way you can specify a lookup to the domain and know what information is being returned. In this domain place the name of the name server in the NS records for the domain, but do not provide an IP address for it.

A sample named.conf extract and zone file to do this are as follows:

```
zone "test.acme.research.gov.au" {
        type master;
        file "/etc/bind/db.test";
        allow-query { any; };
}
```

```
;
$TTL    3600
@    IN    SOA    test.acme.research.gov.au.
hostmaster.acme.research.gov.au.
 (
                        2003070801      ; serial n umber, in date form
                        10800           ; refresh 4 minutes
                        3600            ; retry interval 2 minutes
                        604800          ; expire
                        3600            ; default ttl
)

;NS
@                       IN      NS      ns1.peterhost.ru.
@                       IN      NS      ns1.pchome.org.
@                       IN      NS      ns1.test.acme.research.gov.au.
@                       IN      NS      dnsserver1.nsw.acme.research.g  ov.au.
@                       IN      A       171.93.20.17

@                       IN      MX      10 mail
www                     IN      A       171.93.20.17
```

Turning debugging on in the DNS server, should also show us what queries are attempted for the lookup of the address.

Objective/Subjective

The result of this audit item is objective, it is a clear distinction as to whether glue fetching is turned on or not, there are no variances in the outcome.

Item 16 - Disable Name Server Caching Daemon (nscd)

References

There have been some problems with the nscd caching daemon, as a result Peter H. Gregory in his book *Solaris Security* (Gregory, p.98-200) recommends that the nscd cache be disabled. If the nscd caching daemon is enabled, it will answer hosts queries before the DNS server does. This is a separate caching daemon to the DNS cache that BIND uses. The nscd also performs negative caching as well which can effect the DNS responses.

Control Objective

This audit item examines whether the Name Server Caching Daemon is turned off.

Risk

The risk of having the Name Server Caching Daemon enabled is that it can interfere with the way a DNS server handles a DNS query. For instance in terms of negative caching it can result in a denial of service to a DNS servers own zone information. For example if a host to be added to the DNS is queried and does not exist, but is later added to the DNS, the name server caching daemon may respond to a query still stating the host does not exist. If the name server caching daemon was disabled, the DNS would have known the new host was added after the zone file was reloaded and responded with the correct response that the host existed. (Gregory, p.199). While the likelihood of such an incident occurring is rare or unlikely, the fact that it can cause a denial of service means that the consequences are high.

Compliance

The outcome of the test is a binary response relating to whether the Name Server Caching Daemon is on or off. To be compliant, the expected result is that the caching daemon is disabled for DNS.

Testing

To test whether the Name Server Caching Daemon is running is a simple matter of performing a process listing.

```
% ps –ef | grep nscd
```

If the process is running, a check of the configuration file can be done to determine wether DNS caching is performed.

In order to test the fact that DNS caching is not performed, a trace on the process can be used:

```
# truss -p `cat /etc/nscd.pid`
```

Then a lookup to a host which is known to be cached should be performed. To ensure the host information is cached, a lookup of the same host can be performed twice.

Or an easier method is to get nscd to print the settings it is running, run nscd with the –g option:

```
# nscd -g
```

Objective/Subjective

This audit item is objective as it is a test which can be repeated to gain the same results over again, and the result of the test is a clear binary value of either meeting the expected response or not.

Item 17 - Known Vulnerabilities – Birthday Attack

References
Vagner Sacramento (Saramento) found a vulnerability – CERT Vulnerability Note VU#457875 (see CERT/CC*, Vulnerability Note VU#457875: Various DNS service implementations generate multiple simultaneous queries for the same resource record*) in which when multiple requests were sent to a DNS server, there multiple replies from the DNS. This vulnerability is more in the design of the DNS protocol than the software. As a result DNS ID spoofing can be made easier in what is known as the Birthday Attack. Joe Stewart (Stewart) and Ramon Izaguire (Izaguire) have written some tests to prove this theory.

Control Objective
To determine if the DNS server is vulnerable to the Birthday Attack.

Risks
The risk of being vulnerable to the Birthday Attack means that cache poisoning of the DNS server can occur, even if the DNS ID's are randomised. The reason for this is there is a limit to the number of DNS ID's that a DNS server can produce as DNS ID's are 16 bit numbers. As a result, the more requests sent, the more DNS ID's are used, and the mode likely a DNS response with a spoofed DNS ID will succeed.

If successful the consequence that can occur is DNS cache poisoning, which could be used for a Denial of Service of access to an external network for the local network, DNS hijacking or a man in the middle attack. (stewart)

The likelihood of it occurring depends on caching. If recursion is on then caching will occur. By restricting recursion and glue-fetching, the likelihood is reduced to "unlikely". If recursion is not restricted the likelihood is "almost certain". As a result the risk will be low and high respectively, depending on caching being disabled or enabled.

Compliance
The DNS server will be compliant if the Birthday Attack is unsuccessful.

Testing
There are two tools that can be used to see if the DNS server is vulnerable to this attack. The first is that of Joe Stewart (Stewart) who has provided a perl program to test if the DNS server would be vulnerable. This program is more a proof of concept and will not actually perform cache poisoning. To run this test, use the following commands:

```
# spooftest.pl 171.93.20.11 5000
```

Ramon Izaguire's (Izaguire) script on the other hand takes Joe Stewart's one step further and actually implements the attack and will allow cache poisoning to occur. To run it type the following:

```
# poison.pl 203.23.194.11 171.93.20.11 6321
  www.microsoft.com 198.161.118.107 5000
```

These tests can be run on both internal and external machines. Depending on the configuration of the DNS server, it may be necessary to modify the scripts to spoof internal IP addresses (when run from an internal machine) in the case of where recursion is disabled for external hosts. In our case the following line in Ramon Izaguire's script can be modified as:

```
              . . .
              . . .
$anyip[$_] = sprintf("%d.%d.%d.%d",171,93,20,int(rand(256)));
              . . .
              . . .
```

A tcpdump of the tests should also reveal whether the tests were successful. Generally if the DNS ID's don't match an icmp port unreachable is sent in response from the DNS server. If the DNS ID's do match, then this message will not occur, rather the data will be accepted from the DNS server.

Similarly, a lookup of the host name being spoofed should return the incorrect address.

Objective/Subjective

This test is objective as it is easily repeatable and the results are definitive in determining whether the DNS server is vulnerable or not to the Birthday Attack.

Item 18 - Restrict HINFO and TXT usage on publicly accessible DNS servers

References
Counter Hack by Ed Skoudis (Skoudis, pp.169-172) describes how attackers can use the information in HINFO queries to gain information about computers on a network.

Control Objective
To ensure that HINFO and TXT queries to a DNS server do not provide information that an attacker can use for reconnaissance in planning an attack against the network.

Risk
The risk of allowing HINFO and some TXT information is that an attacker may gain knowledge about the computers on the network. For instance the HINFO query will return a string with the operating system and CPU type. If this information is kept up to date by an administrator, then the attacker could use this information to determine if there are any vulnerabilities on a host.
The risk of giving the information out in relation to the DNS server vulnerability is low, as this does not cause a denial of service or system failure. The likelihood of HINFO occurring may be high if HINFO resource records exist.
A better option than allowing HINFO information to be given out is to place a comment on the HINFO line before the information. This way the information is readable by an administrator modifying the text file, but is not visible to people querying the DNS server for the information.

Compliance
The results of this test will show what information is available to an attacker from the HINFO and TXT queries. Whether this test is met or not is a binary response, of either HINFO and TXT information was available or not. For the DNS server to be compliant no non-public information should be returned from the tests.

Testing
To test what information is visible to external hosts a few simple nslookup or dig queries can be made.

```
% nslookup
> set query=hinfo
> acme.research.gov.au.
> nsw.acme.research.gov.au.

% nslookup
> set query=txt
> acme.research.gov.au.
> nsw.acme.research.gov.au.
```

Objective/Subjective

This test is objective as it is aimed at working out whether HINFO records are able to be retrieved from the DNS server. It is however dependent on the administrator or manager to define which is allowed to be public and what is not.

Item 19 - DNS Data is consistent and up to date

References
Paul Blitz and Cricket Liu in *DNS and BIND* (Albitz & Liu, pp.229-337) discuss problems that can occur when DNS data is not kept up to date. Artur Romano in *RFC 1713 – Tools for DNS Debugging* (Romano) discusses DNS problems and tools that can be used to examine the problems. Inconsistent data can lead to problems when trying to resolve information and possibly lead to an attack.

Control Objective
To ensure that the data of the DNS server is consistent and up-to-date.

Risk
The risk associated with not keeping up to date information is that the IP addresses for old machines may still be registered with DNS as certain resource records. For example the DNS may have several name servers for a domain. If one of the domain controllers is turned off or removed, but the NS resource record is still pointing to the IP address or the configuration file still points to this address for allowing zone transfers, then if an attacker could determine this information, they could for instance spoof DNS requests for zone transfers, or set up a malicious DNS server at this IP address. The likelihood this could occur would happen more with DNS servers which are pointed to are external controlled, but which had a trust relationship with our DNS server.

Compliance
To be compliant with this item, the data on the DNS server should not have any errors or warnings as a result of running the testing tools.

Testing
There are several programs that can be used to automate the consistency checking of data within the DNS. For example, dlint (also known as dns_lint), dnswalk, and DOC (Domain Obscenity Control). These should be run against the nsw.acme.research.gov.au and acme.research.gov.au domains.

```
# dlint nsw.acme.research.gov.au
# dlint acme.research.gov.au

# dnswalk -r nsw.acme.research.gov.au
# dnswalk -r acme.research.gov.au

# doc nsw.acme.research.gov.au
# doc acme.research.gov.au
```

Generally these commands should be run on a DNS server which has zone transfer capability with the zones being audited. It is useful however to try these tests on other machines, as they may point out some misconfigurations that would not otherwise be noticeable.

Objective/Subjective

This audit item is objective in terms that a tool can be run to determine inconsistencies in the DNS. Whether these inconsistencies are cause for concern however will depend on the type of inconsistency which occurred.

Item 20 - DNS ID is Randomized

References
Allen Householder, Brian King and Ken Silva in *Securing an Internet Name Server* (Householder, King & Silva) make reference to DNS ID and randomisation, Joe Stewart in *DNS Cache Poisoning – The Next Generation* (Stewart) and Ramon Izaguire (Izaguire) show how DNS ID randomisation is important to prevent DNS ID spoofing.

Control Objective
To test that DNS transaction ID's are randomised.

Risk
The risk of not having DNS ID's randomised, is that an attacker would be able to predict the DNS ID of the DNS server, and using this information spoof DNS replies to other DNS servers, as well as being able to spoof responses to the DNS server from other servers, as is the case with the Birthday Attack. The consequence of a DNS ID being spoofed is that the DNS servers cache may be poisoned or other DNS servers expecting a response may have their caches poisoned. This may lead to a DNS denial of service, or DNS hijacking or man in the middle attack. The likelihood of these attacks depends on the randomness or predictability of the DNS ID. If it is hard to predict a DNS ID, then the likelihood of the attack is low. On the other hand if the DNS ID is predictable, then the likelihood of an attack is high. However, the effectiveness of the attack would depend upon the DNS server having caching turned on and not having recursion restricted. In such a case the risk is high.

Compliance
The results of this audit test is binary as it is a test to see whether the DNS IDs of packets generated by the DNS server are predictable or not. The DNS server will be compliant if the DNS ID's it generates are not predictable.

Testing
Some versions of BIND did not randomise the DNS transaction ID of packets. This has been a known vulnerability and has been fixed in the latest versions. Implicitly BIND should randomise transaction ID's. There is however a specific option that can be used to explicitly specify using random transaction ID's. This can be done via (Householder, King & Silva, p.14):

```
option  {
          use_id_pool yes;
        };
```

In order to verify that DNS ID's are random we need to monitor the traffic from the DNS server. This can be done with tcpdump by capturing packets and examining the ID's shown to see if they occur in a predictable sequence. This consists of several steps.

Step1: Capture the traffic from the DNS server

```
% tcpdump -n host dnsserver1 -w /tmp/dnsid.out
```

We are actually mainly looking for the DNS ID's produced by the server. In this case the traffic could be filtered further so that we capture only the source host. However, this can also be done when reading the data captured. Steps 1 and 2 could be merged to show the data without writing it to disk, but it is much easier to go through when you can refer back to older packets that have been captured. Note we are using the –n option of tcpdump to tell tcpdump not to do name resolution on the captured data as this would only cause more traffic. If data is saved to a file, the name resolution can be done when the data is read back in.

Step 2: Examine the traffic captured and filter the results.

```
% tcpdump –r /tmp/dnsid.out src host dnsserver1
```

In examining the output, we need to see if there is a pattern in the DNS ID's that can be perceived from the traffic data collected. Earlier versions of BIND used to just increment the DNS ID by one, however this should not be the case here, but there may be some level of predictability in the DNS ID's produced.

We can also use a tool such as *zodiac* which can be used to test whether DNS ID spoofing is possible for a desired DNS server. Zodiac can be found at: http://www.packetfactory.net/projects/zodiac/.

Objective/Subjective
This test is objective as the outcome of the audit check is based on tests which are repeatable to obtain the same result.

Item 21 - Vulnerability alerts monitored and patch ad upgrade procedures in place

References
The 2003 Australian Computer Crime and Security Survey (see *2003 Australian Computer Crime and Security Survey*, p.24), notes that a major cause of incidents is the lack of patched systems. It is also Government regulation that these be in place – see *Commonwealth Protective Security Manual* – section C 5.13 (Attorney-General's Department, p.C-23).

Control Objective
To test that administrators are notified of vulnerability alerts to software such as BIND, and have procedures in place to patch or upgrade the software when patches become available.

Risk
If vulnerability alerts are not sent to administrators and they do not actively look up vulnerability alerts, then computer systems may go unpatched and remain vulnerable to exploits. Similarly, if vulnerability alerts are received but there are not procedures in place, then similarly machines may remain unpatched or not upgraded. Both alerting and procedures must be in place for ensuring systems are secure. The risk of not upgrading or patching is leaving the network and in relation to BIND, the DNS server vulnerable to attacks. These attacks may be root exploits or denial of service attacks. As a result the organisation may get a bad reputation if it becomes public knowledge that a breaking or denial of service has occurred, or legal action may be taken if the machines were used to attack other external sites. The risk varies from low to high. Alerting and procedures may be in place, but it is up to the system administration team and IT manager to follow up on it. If no action is done by the administrators, the risk is high, if the administrators are reliable and conscientious and take action, then the risk is low.

Compliance
The compliance to this is binary in terms of receiving notifications and procedures being in place for upgrading and patching. However the types and number of vulnerability notifications they receive depend on the lists they are subscribed to.

Testing
Testing of this control objective can be done via examining the email administrators receive. In general many groups of administrators get the email sent to a group list, which is then exploded to individuals. If it is not possible to view the administrators email, then a check of the mail system configuration may show up some type of notification list.

To ensure procedures are in place, the documentation for handling patching and updates should be sighted.

If change control procedures are in place, the administrators should be able to show you the change control forms when the systems were patched or upgraded.

Objective/Subjective

This audit item is subjective, as the risk may vary based on the actions of the system administration team. Similarly the test is subjective in terms that apart from asking an administrator to show you how the patching and update procedures are done you only have their word and the sighting of a procedural document that action is taken when vulnerabilities are found, unless of course change control documentation is available, which can be referred to.

Item 22 - Incident Handling and Response Procedures in Place

References
The ASCI33 (see *Australian Communications-Electronic Security Instruction 33 (ACSI 33) – Version 1.0*, pp.4-3,13-7,13-8) states that incident handling mechanism and response plans must be in place. The *Commonwealth Protective Security Manual* – section G (Attorney General's Department, pp.G-3–G-37) also has guidelines on reporting incidents. The AS/NZ 7799.1:2003 - sections 6.3 and 8.1.3 (see Standards Australia, *AS/NZ 7799.1:2003,* pp.17,27) and AS/NZ 7799.2:2003 – sections 4.4.3.1, 4.4.3.4 and 4.6.1.3 (see Standards Australia, *AS/NZ 7799.2:2003,* pp.9, 11) standards also specify the requirement for incident handling and response procedures

Control Objective
To ensure that incident handling and Response procedures in place.

Risk
It is a government regulation for the organisation to have incident handling procedures and response plans. If an attack occurs and the system is compromised and there are incident procedures and response plans, then there are steps to follow. For example, a root exploit has occurred and the DNS server has had a root kit installed on it. The machine logs are taken and recorded for audit trail purposes and evidence in case of legal proceedings. An alternative machine is built and data restored from backup to maintain services, or the slave DNS server is made primary while the compromised server is repaired.

If there are no incident handling and response procedures in place, then the organisation may not follow the correct procedures in handling evidence of an attack, and may not have contingency plans in place to keep the service running.

At the risk of not having an incident handling and response procedure, the organisation may find themselves in a legal predicament as it is a government regulation that one is in place. As a result the risk of not having the procedures is high. The likelihood of not having a plan is low, as it is a requirement for an organisation. However the ability to follow it is dependent on the training of the system administration staff to follow it. If this has not been done, then the risk is even higher, as there is a procedure in place which has not been followed – making the organisation possibly liable if any damage is done to external parties from the organisations computing facilities.

Compliance
This audit item test is binary in response to there being procedure and plans, but as to how they are carried out may vary, as the readers may interpret things differently.

Testing

To test this control objective, we must ensure that relevant documentation for handling incidents and contingencies are in place. This requires sighting of the document for incident handling and the document containing the contingency plans. A good test would be to ask an administrator to demonstrate how they would go about handling an incident and what procedures they would follow and what contingencies they would put in place. This should match the incident handling procedures and response plan documentation.

Objective/Subjective

This test is subjective as it depends on people being able to follow procedures, which each may interpret differently. It also depends on a test which may have different results each time, in terms of asking the user to demonstrate the procedures.

Item 23 - Disaster Recovery plans are in place

References
The *Commonwealth Protective Security Manual* – section C 5.13 (Attorney-General's Dep., p.C-23) and the ASCI 33 (see *Australian Communications-Electronic Security Instruction 33 (ACSI 33) – Version 1.0*, p.4-5) standards, along with the AS/NZ 7799.1:2003 – sections 8.1.3, 8.1.4 and 11.1 (see Standards Australia, *AS/NZ 7799.1:2003*, pp.27,32,68-71) and AS/NZ 7799.2:2003 – sections 4.6.4.1 and 4.9 (see Standards Australia, *AS/NZ 7799.2:2003*, pp.12,20) standards recommend that disaster recovery plans be in place. Other references like the Interpol (see Interpol, *Company Checklist*) *Company Checklist* also require disaster recovery plans are in place.

Control Objective
To ensure that disaster recovery plans are in place.

Risk
The risk of not having a disaster recovery plan and procedure is high. The likelihood of a disaster occurring is high. This could be in terms of hardware failure, such as a hard disk failing, or in terms of an attack on a system. Without disaster recovery procedures and plans the recovery of a system to a working state may take a long time. For example, in the 2003 Australian Computer Crime and Security Survey (see *2003 Australian Computer Crime and Security Survey*, p.21), of the companies or organisations surveyed, 39% of incidents took 1 to 7 days, and 10% took longer than 8 days to get systems back up. In terms of the DNS server, while the likelihood of failure may be medium to high, if a disaster recovery plan is in place, which it should be due to government regulations and standards, the consequences of a failure are reduced by the ability to get the service backup and running via the disaster recovery plan. In this instance, the secondary slave DNS servers provide a backup for the primary DNS server while it is recovered.

Compliance
That disaster recovery plans exist is a yes or no value. That people understand the procedures and plans and can implement them is dependent on the training and skills of the staff. To test that data is being backed up a look at the backup logs and database will reveal what is backed up, the level of data that is backed up may vary however.

Testing
To test that disaster recovery plans are in place, sighting of the disaster recovery plans and procedures should be performed. For testing the procedures a test could be to ask the system administration team to assume that a server had gone down, and ask them to perform a disaster recovery on a test machine. Disaster recovery also involves backup procedures and offsite storage of backup data as a general standard.

A test to ensure the backup is done, is to examine the backup server and database and generate reports, or look at the log files to ensure that machines are being backed up. The test procedure for ensuring the backup of

the DNS data in an earlier audit item can be performed to test this item. Backup is only one part of disaster recovery, as a result, a test to ensure a machine can be recovered, should be in place. This would involve performing a restore from the backup system, and recording the time take to perform the restore and how services were maintained while a machine was being restored.

Objective/Subjective

This audit test is subjective as disaster recovery plans may be different for different organisation and may provide different levels of detail. In terms of checking a disaster recovery plan exists, this may be objective, but knowing that procedures are followed is subjective and may be different depending on the level of skills of the employees and their interpretation of the procedures and plans.

Categorisation of Audit Items

There are 3 main categories where each of the audit items fall, these are for preventative controls, detective controls and corrective controls.

Preventative Audit Items

Most of the audit items in the checklist are for preventative security measures – that is, security by preventing a threat. The following checklist items fall into this category:

- BIND version number obfuscated
- Run BIND as non-root account
- Recursion disabled or restricted
- Zone Transfers restricted
- Zone Transfers authenticated
- Dynamic DNS updates disabled or restricted
- Ensure root servers updated regularly
- Limit or restrict queries for internal network information
- Limit number of zone transfers at one time
- Restrict HINFO and TXT usage
- Firewall filters traffic to DNS server
- Authoritative Negative Caching disabled
- DNS ID Randomisation
- Fetch Glue Disabled
- NSCD DNS cache disabled
- Vulnerability alerts monitored and patch procedures in place

Detective Audit Items

The detective items are those items which can be used to monitor or alert when an unexpected or malicious incident occurs. The following checklist items fall into this category:

- Logging is turned on for BIND
- Statistics enabled and dumped regularly

Corrective Audit Items

The corrective items are those items which can be used to correct a problem with security, and also include responses to security incidents. Some people like to separate corrective and responsive items, however a responsive action could be considered as a corrective action The following audit checklist items fall into this category:

- Backup of Data and System – this may also be considered at preventative measure, but it exists in order to respond as a corrective measure.
- Incident handling and response procedures in place
- Disaster recovery plans in place

Baselining a DNS server

A baseline of a DNS server in relation to the amount of general traffic, the types of queries, the number of queries, the number of zone transfers and the hosts involved can be gained from the BIND server in order to create a baseline or standard operating configuration and level of service.

There are two main parts to obtaining information in building a DNS server baseline. Using the DNS server statistics and using the log files. For the first case, the server needs to be compiled to support statistics. The DNS server statistics is an important part. Server statistics can be dumped in any time frame – they could be daily, weekly, monthly, hourly or via the minute.

There are also several tools for compiling the and understanding the statistics. Paul Albitz and Cricket Lui's *DNS and BIND* (Albitz & Liu, pp. 163-172) book provides a useful interpretation of the information which is dumped from the statistics. There is also software which can be obtain to analyze the statistical dumps, a very good one is bindstat by NRG (http://nrg.help.wisc.edu/bindstat.html). This provides a graphical representation of the BIND data for comparison, such as number of queries and responses and the response times. The output is similar to that of the mrtg network tools.

Another tool which can be used to analyze the BIND data is dnsstats.(http://fresh.t-systems-sfr.com/unix/src/misc/dns/dnsstats). It can be used to examine the log files. This is useful for obtaining statistics, if statistics is not compiled into the BIND application and the only information you have is in the BIND logs. This is a shell script which reads the information in the logs and produces some statistics.

Assignment 3 – Audit Evidence

Introduction

Some of the tests are very similar or the same for testing to different aspects of DNS security. For instance in examining dynamic DNS restriction, the same test could be used for testing that the logs record denied dynamic updates. The test for the backup of the DNS server also provides evidence that the logging audit trails are kept for at least a minimum period. As a result I have included in this section some additional checklist audit items to show 10 distinct different tests.

Audit Item 5 - Recursion is turned off, or if on is restricted to authorised clients

Test 1: Test for recursion from an internal client:

```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

O:\>nslookup
Default Server:  dnsserver2.nsw.acme.research.gov.AU
Address:  171.93.20.14

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

> set recurse
> www.microsoft.com.
Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

------------
Got answer:
    HEADER:
        opcode = QUERY, id = 11, rcode = NOERROR
        header flags:  response, want recursion, recursion avail.
        questions = 1,  answers = 7,  authority records = 8,  additional = 8

    QUESTIONS:
        www.microsoft.com, type = A, class = IN
    ANSWERS:
    ->  www.microsoft.com
        canonical name = www.microsoft.akadns.net
        ttl = 7062 (1 hour 57 mins 42 secs)
    ->  www.microsoft.akadns.net
        Internet address = 207.46.249.222
        ttl = 163 (2 mins 43 secs)
    ->  www.microsoft.akadns.net
        Internet address = 207 .46.134.155
        ttl = 163 (2 mins 43 secs)
    ->  www.microsoft.akadns.net
        Internet address = 207.46.249.27
        ttl = 163 (2 mins 43 secs)
    ->  www.microsoft.akadns.net
        Internet address = 207.46.249.190
        ttl = 163 (2 mins  43 secs)
    ->  www.microsoft.akadns.net
        Internet address = 207.46.134.222
        ttl = 163 (2 mins 43 secs)
    ->  www.microsoft.akadns.net
```

```
        Internet address = 207.46.134.190
        ttl = 163 (2 mins 43 secs)
    AUTHORITY RECORDS:
    ->  akadns.net
        nameserver = zc.akadns.net
        ttl = 162978 (1 day 21 hours 16 mins 18 secs)
    ->  akadns.net
        nameserver = zf.akadns.net
        ttl = 162978 (1 day 21 hours 16 mins 18 secs)
    ->  akadns.net
        nameserver = ns1-93.akam.net
        ttl = 162978 (1 day 21 hours 16 mins 18 secs)
    ->  akadns.net
        nameserver = ns1-159.akam.net
        ttl = 162978 (1 day 21 hours 16 mins 18 secs)
    ->  akadns.net
        nameserver = use2.akam.net
        ttl = 162978 (1  day 21 hours 16 mins 18 secs)
    ->  akadns.net
        nameserver = usw5.akam.net
        ttl = 162978 (1 day 21 hours 16 mins 18 secs)
    ->  akadns.net
        nameserver = use4.akam.net
        ttl = 162978 (1 day 21 hours 16 mins 18 secs)
    ->  akadns.net
        nameserver = asia3.akam.net
        ttl = 162978 (1 day 21 hours 16 mins 18 secs)
    ADDITIONAL RECORDS:
    ->  zc.akadns.net
        Internet address = 63.241.199.50
        ttl = 145820 (1 day 16 hours 30 mins 20 secs)
    ->  zf.akadns.net
        Internet address = 63.215.198.79
        ttl = 145820 (1 day 16 hours 30 mins 20 secs)
    ->  ns1-93.akam.net
        Internet address = 193.108.91.93
        ttl = 145820 (1 day 16 hours 30 mins 20 secs)
    ->  ns1-159.akam.net
        Internet address = 193.108.91.159
        ttl = 145820 (1 day 16 hours 30 mins 20 secs)
    ->  use2.akam.net
        Internet address = 63.209.170.136
        ttl = 145820 (1 day 16 hours 30 mins 20 secs)
    ->  usw5.akam.net
        Internet address = 63.241.73.214
        ttl = 145820 (1 day 16 hours 30 mins 20 secs)
    ->  use4.akam.net
        Internet address = 80.67.67.182
        ttl = 145820 (1 day 16 hours 30 mins 20 secs)
    ->  asia3.akam.net
        Internet address = 193.108.154.9
        ttl = 145820 (1 day 16 hours 30 mins 20 secs)

------------
Non-authoritative answer:
Name:    www.microsoft.akadns.net
Addresses:  207.46.249.222, 207.46.134.155, 207.46.249.27, 207.46.249.190
          207.46.134.222, 207.46.134.190
Aliases:  www.microsoft.com

>
```

The output of the nslookup show that recursion was requested, and recursion was available in the answer. As a result we can see that recursion is turned on and is allowed for internal clients.

Test 2: Test of a recursive query from an external host:

```
# nslookup
Default Server:  externalhost.telstra.bigpond.com.au
Address:  139.162.101.102

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.au
Address:  171.93.20.11

> set debug
> set recurse
> www.microsoft.com.
Server:  dnsserver1.nsw.acme.research.gov.au
Address:  171.93.20.11

;; res_nmkquery(QUERY, www.microsoft.com, IN, A)
------------
Got answer:
    HEADER:
        opcode = QUERY, id = 26862, rcode = REFUSED
        header flags:  response, want recursion
        questions = 1,  answers = 0,  authority records = 0,  additional = 0

    QUESTIONS:
        www.microsoft.com, type = A, class = IN

------------
*** dnsserver1.nsw.acme.research.gov.au can't find www.microsoft.com.: Query
refused
>
```

The output of the nslookup shows that the query was refused.

Test 3: Test of a recursive query from an external host for something on acme.research.gov.au domain.

```
# nslookup
Default Server:  externalhost.telstra.bigpond.com.au
Address:  139.162.101.102

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.au
Address:  171.93.20.11

> set recurse
> set debug
> www.acme.research.gov.au
Server:  dnsserver1.nsw.acme.research.gov.au
Address:  171.93.20.11
;; res_nmkquery(QUERY, www.acme.research.gov.au, IN, A)
------------
Got answer:
    HEADER:
        opcode = QUERY, id = 61151, rcode = NOERROR
        header flags:  response, auth. answer, want recursion
        questions = 1,  answers = 2,  authority records = 6,  additional = 6

    QUESTIONS:
        www.acme.research.gov.au, type = A, class = IN
```

```
    ANSWERS:
    ->  www.acme.research.gov.au
        Internet address = 181.52.31.10
        ttl = 600 (10M)
    ->  www.acme.research.gov.au
        Internet address = 171.93.30.1
        ttl = 600 (10M)
    AUTHORITY RECORDS:
    ->  acme.research.gov.au
        nameserver = dnsserver1.nsw.acme.research.gov.au
        ttl = 10800 (3H)
    ->  acme.research.gov.au
        nameserver = vicdns1.vic.acme.research.gov.au
        ttl = 10800 (3H)
    ->  acme.research.gov .au
        nameserver = vicdns2.vic.acme.research.gov.au
        ttl = 10800 (3H)
    ->  acme.research.gov.au
        nameserver = wadns1.wa.acme.research.gov.au
        ttl = 10800 (3H)
    ->  acme.research.gov.au
        nameserver = sadns1.sa.acme.re search.gov.au
        ttl = 10800 (3H)
    ->  acme.research.gov.au
        nameserver = actdns1.act.acme.research.gov.au
        ttl = 10800 (3H)
    ADDITIONAL RECORDS:
    ->  dnsserver1.nsw.acme.research.gov.au
        Internet address = 171.93.20.11
        ttl = 86400 (1D)
    ->  vicdns1.vic.acme.research.gov.au
        Internet address = 173.14.56.2
        ttl = 86400 (1D)
    ->  vicdns2.vic.acme.research.gov.au
        Internet address = 173.14.56.30
        ttl = 86400 (1D)
    ->  wadns1.wa.acme.research.gov.au
        Internet address = 174.92.51.128
        ttl = 86400 (1D)
    ->  sadns1.sa.acme.research.gov.au
        Internet address = 175.39.42.2
        ttl = 600 (10M)
    ->  actdns1.act.acme.research.gov.au
        Internet address = 18 1.52.33.1
        ttl = 86400 (1D)

------------
Name:    www.acme.research.gov.au
Addresses:  181.52.31.10, 171.93.22.1

>
```

The output of this test shows that the domain can be queried. However even though recursion was requested, recursion does not happen as the server is authoritative for the domain and controls the zone files. There is no “recursion available” message in the header section of the nslookup debug.

From tests 2 and 3 we can see that recursion is not possible from an external host. Queries from an external host about other external domains are refused. Test 3 shows that only queries about the domain itself can be made.

Test 1 however shows that recursion is on and can be performed by the internal hosts.

Compliance with the Audit Control Objective

The results of the tests show that the DNS server has met the criteria of restricting recursion to authorised hosts. However the way it is restricted is different from the way we were expecting to depict it.

It was not possible to test a recursive query from an external host to an external site, as queries were restricted for external hosts. So the question remains is it the just the query restriction rule, or if the query restriction was turned off, would there be a recursive restriction rule.

A look at the configuration file reveals the following:

```
options {
        check-names response warn;
        check-names slave ignore;
        directory "/var/bind";
        fake-iquery yes;
        auth-nxdomain no;
        allow-transfer { none; };
        allow-query { acme_clients; };
        allow-recursion { acme_clients; };
};
```

It is fair to assume that if query restrictions were removed, recursion would only be allowed for the trusted internal hosts.

Audit Item 6 - Zone transfers Restricted

Test 1: Zone transfer from an unauthorised internal host

```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

O:\>nslookup
Default Server:  dnsserver2.nsw.acme.research.gov.AU
Address:  171.93.20.14

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

> ls -d acme.research.gov.au
[dnsserver1.nsw.acme.research.gov.AU]
*** Can't list domain acme.research.gov.au: Query refused
> ls -d nsw.acme.research.gov.au
[dnsserver1.nsw.acme.research.g ov.AU]
*** Can't list domain nsw.acme.research.gov.au: Query refused
>
```

From the output of the nslookup we can see that a general internal host is not authorised to perform a zone transfer.

If we examine the logs for the DNS server regarding this host, the following is shown:

```
# grep 171.93.20.92 /var/adm/messages
```

```
Jul  5 19:58:55 dnsserver1.nsw.acme.research.gov.AU named[22620]: denied
AXFR from [171.93.20.92].2962 for "acme.research.gov.au" IN (acl)
Jul  5 19:59:02 dnsserver1.nsw.acme.research.gov.AU name d[22620]: denied
AXFR from [171.93.20.92].2963 for "nsw.acme.research.gov.au" IN (acl)
#
```

The results of this first test show that the internal hosts if they are not authorised cannot perform a zone transfer.

Test 2: Zone transfer from an external untrusted host

```
# nslookup
Default Server:  externalhost.telstra.bigpond.au
Address:  139.162.101.102

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

> ls -d nsw.acme.research.gov.au
[dnsserver1.nsw.acme.resear ch.gov.AU]
*** Can't list domain nsw.acme.research.gov.au: Unspecified error
> ls -d acme.research.gov.au
[dnsserver1.nsw.acme.research.gov.AU]
*** Can't list domain acme.research.gov.au: Unspecified error
> exit
```

From the output of the nslookup, we can see that we have received an error message. It does not relate to refusal that we can see from this output.

We should also examine the logs

```
# grep 139.162.101.102 /var/adm/messages
Jul  5 19:50:12 dnsserver1.nsw.acme.research.gov.AU named[22620]: denied
AXFR from [139.162.101.102].61458 for "nsw.acme.research.gov.au" IN (acl)
Jul  5 19:50:34 dnsserver1.nsw.acme.research.gov.AU named[22620]: denied
AXFR from [139.162.101.102].61460 for "acme.research.gov.au" IN (acl)
#
```

We can see from the logs that the DNS server has denied the zone transfer for the unauthorised host.

Test 3: Zone transfer from a trusted host

```
Script started on Sat 05 Jul 2003 08:06:47 PM EST

dnsserver2.nsw.acme.research.gov.AU:/
35 >nslookup
Default Server:  dnsserver2.nsw.acme.research.g ov.AU
Address:  171.93.20.14

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

> ls -d acme.research.gov.au
[dnsserver1.nsw.acme.research.gov.AU]
$ORIGIN acme.research.gov.au.
@                   3H IN SOA     dnsserver1.nsw h ostmaster.nsw (
                                  2003051400    ; serial
                                  1H            ; refresh
```

```
                                20M          ; retry
                                4W           ; expiry
                                3H )         ; minimum

                    3H IN NS    dnsserver1.nsw
dnsserver1.nsw      1D IN A     171.93.20.11
@                   3H IN NS    vicdns1.vic
vicdns1.vic         1D IN A     173.14.56.2
@                   3H IN NS    vicdns2.vic
vicdns2.vic         1D IN A     173.14.56.30
@                   3H IN NS    wadns1.wa
wadns1.wa           1D IN A     174.92.51.128
@                   3H IN NS    sadns1.sa
sadns1.sa           10M IN A    175.39.42.2
@                   3H IN NS    actdns1.act
actdns1.act         1D IN A     181.52.33.1
@                   3H IN MX    0 nswmail.nsw
                    3H IN MX    20 vicmail.vic
                    3H IN TXT   "ACME Research Organisation"
www1                10M IN A    171.93.22.1
www2                10M IN A    181.52.31.10
                . . .
                . . .
                . . .
act                 1D IN NS    actdns1.act
actdns1.act         1D IN A     181.52.33.1
act                 1D IN NS    wadns1.wa
wadns1.wa           1D IN A     174.92.51.128
act                 1D IN NS    sadns1.sa
sadns1.sa           10M IN A    175.39.42.2
act                 1D IN NS    dnsserver1.nsw
dnsserver1.nsw      1D IN A     171.93.20.11
act                 1D IN NS    vicdns1.vic
vicdns1.vic         1D IN A     173.14.56.2
act                 1D IN NS    vicdns2.vic
vicdns2.vic         1D IN A     173.14.56.30
Internet1           3H IN A     171.93.22.1
stereo              3H IN CNAME homan.act
Internet2           3H IN A     181.52.31.10
@                   3H IN SOA   dnsserver1.nsw hostmaster.nsw (
                                2003051400   ; serial
                                1H           ; refresh
                                20M          ; retry
                                4W           ; expiry
                                3H )         ; minimum

> ls -d nsw.acme.research.gov.au
[dnsserver1.nsw.acme.research.gov.AU]
$ORIGIN nsw.acme.research.gov.au.
@                   1D IN SOA   dnsserver1 hostmaster.acme.research.gov.au.
(
                                2003070303   ; serial
                                3H           ; refresh
                                20M          ; retry
                                4W           ; expiry
                                1D )         ; minimum

                    1D IN NS    dnsserver1
                    1D IN NS    actdns1.act.acme.research.gov.au.
                    1D IN NS    wadns1.wa.acme.research.gov.au.
                    1D IN NS    sadns1.sa.acme.research.gov.au.
                    1D IN NS    vicdns2.vic.acme.research.gov.au.
                    1D IN NS    vicdns1.vic.acme .research.gov.au.
                    1D IN MX    10 vicmail.vic.acme.research.gov.au.
                    1D IN MX    0 nswmail
                    1D IN TXT   "wp-noop://"
nswcol1             1D IN MX    0 nswmail
                    1D IN A     171.93.23.5
```

```
nsrhost              1D IN CNAME   dnsserver2
                  . . .
                  . . .
                   . . .

somersby-nh          1D IN MX      0 nswmail
                     1D IN A       171.93.26.41
panania-nh           1D IN MX      0 nswmail
                     1D IN A       171.93.25.131
lotus                1D IN MX      0 nswmail
                     1D IN A       171.93.20.159
@                    1D IN SOA     dnsserver1 hostmaster.acme.research.gov.au.
(
                                   2003070303    ; serial
                                   3H            ; refresh
                                   20M           ; retry
                                   4W            ; expiry
                                   1D )          ; minimum

> exit

dnsserver2.nsw.acme.research.gov.AU:/
36 >^D__
script done on Sat 05 Jul 2003 08:07:11 PM EST
```

From the output, it can be seen that the zone transfer requests to both zones was allowed. The output has been condensed as there was over 30 pages of information returned by the zone transfer requests. An examination of the log files for the DNS server reveal that all zone transfers are logged.

Compliance with the Audit Control Objective

From the results we can conclude that zone transfers are restricted to only authorised hosts. As a result the DNS server meets this audit item.

Audit Item 10 - BIND DNS server is running as non-root

Test. See what user the name server is running under:

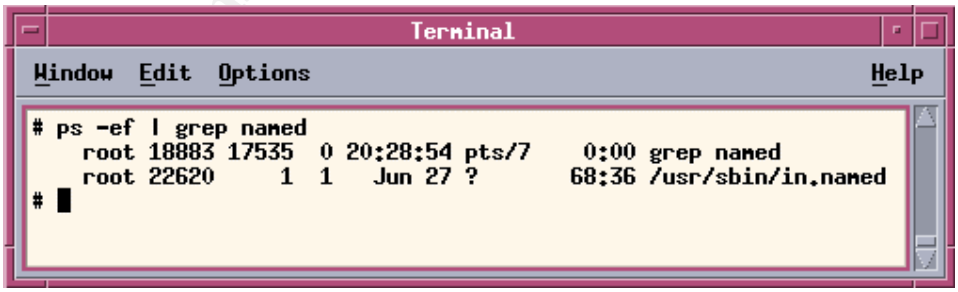

Figure 15 – Screenshot of ps listing

From the output we can see that the DNS server is running under the “root” user account.

Compliance with the Audit Control Objective

The DNS server does not meet the control objective of running a DNS server as an account other than root.

Audit Item 7 - Zone transfers are authenticated with transaction signatures

Test. Monitor traffic of a zone transfer between the primary DNS server and the secondary slave.

```
16:05:53.270913 171.93. 20.11.47607 > 171.93.20.14.53:  58823 notify
[b2&3=0x2400
] SOA? nsw.acme.research.gov.au. (35) (DF)
0x0000   4500 003f 03c1 4000 ff11 52b1 ab5d 140b        E..?..@...R.....
0x0010   ab5d 140e b9f7 0035 002b 96b2 e5c7 2400        .......5.+....$.
0x0020   0001 0000 0000 0000 036e 7377 0461 636d        .......nsw.acme.
0x0030   6508 7265 7365 6102 4155 0000 0600 01          research.gov.au.
16:05:53.271052 171.93.20.11.47607 > 181.52.2.201.53:  65390 notify
[b2&3=0x2400] SOA? nsw.acme.research.gov.au. (35) ( DF)
0x0000   4500 003f 001a 4000 ff11 4ddb ab5d 140b        E..?..@...M.....
0x0010   b534 02c9 b9f7 0035 002b 748e ff6e 2400        .S.....5.+t..n$.
0x0020   0001 0000 0000 0000 036e 7377 0461 636d        .......nsw.acme.
0x0030   6508 7265 7365 6102 4155  0000 0600 01          research.gov.au.
16:05:53.271792 171.93.20.14.43875 > 171.93.20.11.53:  38232 SOA?
nsw.research.gov.AU. (35) (DF)
0x0000   4500 003f 00fe 4000 ff11 5574 ab5d 140e        E..?..@...Ut....
0x0010   ab5d 140b ab63 0035 002b 19b6 9558 00 00        .....c.5.+...X..
0x0020   0001 0000 0000 0000 036e 7377 0461 636d        .......nsw.acme.
0x0030   6508 7265 7365 6102 4155 0000 0600 01          research.gov.au.
16:05:53.271853 171.93.20.14.53 > 171.93.20.11.47607:  58823 notify* 0/0/0
(35) (DF)
0x0000   4500 003f 00ff 4000 ff11 5573 ab5d 140e        E..?..@...Us....
0x0010   ab5d 140b 0035 b9f7 002b 1632 e5c7 a480        .....5...+.2....
0x0020   0001 0000 0000 0000 036e 7377 0461 636d        .......nsw.acme.
0x0030   6508 7265 7365 6102 4155 0 000 0600 01          research.gov.au.
16:05:53.272744 171.93.20.11.53 > 171.93.20.14.43875:  38232* 1/6/6
SOA[|domain] (DF)
0x0000   4500 015b 03c2 4000 ff11 5194 ab5d 140b        E..[..@...Q.....
0x0010   ab5d 140e 0035 ab63 0147 ed01 9558 8480        ... ..5.c.G...X..
0x0020   0001 0001 0006 0006 036e 7377 0461 636d        .......nsw.acme.
0x0030   6508 7265 7365 6102 4155 0000 0600 01c0        research.gov.au.
0x0040   0c00 0600 0100 0151 8000 2a06 646e 7373        .......Q..*.dnss
0x0050   6572
16:05:53.292864 171.93.20.14.34558 > 171.93.20.11.53: S
3463187864:3463187864(0) win 8760 <mss 1460> (DF)
0x0000   4500 002c 0101 4000 ff06 558f ab5d 140e        E..,..@...U.....
0x0010   ab5d 140b 86fe 0035 ce6c 0d98 0000 0000        .......5.l......
0x0020   6002 2238 ed7a 0000 0204 05b4 5555             `."8.z......UU
16:05:53.293146 171.93.20.11.53 > 171.93.20.14.34558: S
781605821:781605821(0) ack 3463187865 win 24820 <mss 1460> (DF)
```

```
0x0000   4500 002c 03c3 4000 4006 11ce ab5d 140b        E..,..@.@.......
0x0010   ab5d 140e 0035 86fe 2e96 5bbd ce6c 0d99        .....5....[..l..
0x0020   6012 60f4 245a 0000 0204 05b4 5555              `.`.$Z......UU
16:05:53.293277 171.93.20.14.34558 > 171.93.20.11.53: . ack 1 win 8760 (DF)
0x0000   4500 0028 0102 4000 ff06 55 92 ab5d 140e        E..(..@...U.....
0x0010   ab5d 140b 86fe 0035 ce6c 0d99 2e96 5bbe        .......5.l....[.
0x0020   5010 2238 7ad3 0000 5555 5555 5555              P."8z...UUUUUU
16:05:53.293363 171.93.20.14.34558 > 171.93.20.11.53: P 1:38(37) ack 1 win
8760 39604 SOA? nsw.acme.research.gov.au. (35) (DF)
0x0000   4500 004d 0103 4000 ff06 556c ab5d 140e        E..M..@...Ul....
0x0010   ab5d 140b 86fe 0035 ce6c 0d99 2e96 5bbe        .......5.l....[.
0x0020   5018 2238 6019 0000 0023 9ab4 0000 0001        P. "8`....#......
0x0030   0000 0000 0000 036e 7377 0461 636d 6508        .....nsw.acme.re
0x0040   7265 7365 6102 4155 0000 0600 01                search.gov.au.
16:05:53.293458 171.93.20.11.53 > 171.93.20.14.34558: . ack 38 win 24783
(DF)
0x0000   4500 0028  03c4 4000 4006 11d1 ab5d 140b        E..(..@.@.......
0x0010   ab5d 140e 0035 86fe 2e96 5bbe ce6c 0dbe        .....5....[..l..
0x0020   5010 60cf 3c17 0000 5555 5555 5555              P.`.<...UUUUUU
16:05:53.295536 171.93.20.11.53 > 171.93.20.14.34558: P 1: 322(321) ack 38
win 24820 39604* 1/6/6 (319) (DF)
0x0000   4500 0169 03c5 4000 4006 108f ab5d 140b        E..i..@.@.......
0x0010   ab5d 140e 0035 86fe 2e96 5bbe ce6c 0dbe        .....5....[..l..
0x0020   5018 60f4 f483 0000 013f 9ab4 8480 0001        P.`. .....?......
0x0030   0001 0006 0006 036e 7377 0461 636d 6508        .....nsw.acme.re
0x0040   7265 7365 6102 4155 0000 0600 01c0 0c00        search.gov.au...
0x0050   0600                                          ..
16:05:53.295702 171.93.20.14.34558 > 1 71.93.20.11.53: . ack 322 win 8760
(DF)
0x0000   4500 0028 0104 4000 ff06 5590 ab5d 140e        E..(..@...U.....
0x0010   ab5d 140b 86fe 0035 ce6c 0dbe 2e96 5cff        .......5.l.... \.
0x0020   5010 2238 796d 0000 5555 5555 5555              P."8ym..UUUUUU
16:05:53.297179 171.93.20.14.34558 > 171.93.20.11.53: P 38:75(37) ack 322
win 8760 39605 AXFR? nsw.acme.research.gov.au. (35) (DF)
0x0000   4500 004d 0105 4000 ff06 556a ab5d 140e        E..M..@...Uj....
0x0010   ab5d 140b 86fe 0035 ce6c 0dbe 2e96 5cff         .......5.l....\.
0x0020   5010 2238 796d 0000 5555 5555 5555              P."8ym..UUUUUU
16:05:53.297179 171.93.20.14.34558 > 171.93.20.11.53: P 38:75(37) ack 322
win 8760 39605 AXFR? nsw.acme.research.gov.au. (35) (DF)
0x0000   4500 004d 0105 4000 ff0 6 556a ab5d 140e        E..M..@...Uj....
0x0010   ab5d 140b 86fe 0035 ce6c 0dbe 2e96 5cff        .......5.l.... \.
0x0020   5018 2238 68b1 0000 0023 9ab5 0000 0001        P."8h....#......
0x0030   0000 0000 0000 036e 7377 0461 636d 6508        .....nsw.acme  .re
0x0040   7265 7365 6102 4155 0000 fc00 01                search.gov.au.
16:05:53.297292 171.93.20.14.40119 > 171.93.20.11.514: udp 72 (DF)
0x0000   4500 0064 0106 4000 ff11 5547 ab5d 140e        E..d..@...UG....
0x0010   ab5d 140b 9cb7 0202 0050 729c 3c 33 303e        .........Pr.<30>
0x0020   4a75 6c20 2033 2031 363a 3039 3a31 3220        Jul..3.16:09:12.
0x0030   6e61 6d65 642d 7866 6572 5b31 3235 3931        named -xfer[12591
0x0040   5d3a 2073 656e 6420 4158 4652 2071 7565        ]:.send.AXFR.que
0x0050   7279
16:05:53.314018 171.93.20.11.53 > 171.93.20.14.34558: . 322:1782(1460) ack
75 win 24820 39605* - 1/0/0 (1458) (DF)
0x0000   4500 05dc 03c6 4000 4006 0c1b ab5d 140b        E.....@.@.......
0x0010   ab5d 140e 0035 86fe 2 e96 5cff ce6c 0de3        .....5.... \..l..
0x0020   5010 60f4 ba1a 0000 0059 9ab5 8400 0001        P.`......Y......
0x0030   0001 0000 0000 036e 7377 0461 636d 6508        .....nsw.acme.re
0x0040   7265 7365 6102 4155 0000 fc00 01c0 0c00        resarch.gov .au..
0x0050   0600                                          ..
16:05:53.314141 171.93.20.11.53 > 171.93.20.14.34558: P 1782:3242(1460) ack
75 win 24820 21071 [b2&3=0x241] [1a] [21760q] [1n] [1au][|domain] (DF)
0x0000   4500 05dc 03c7 4000 4006 0c1a ab5d  140b        E.....@.@.......
0x0010   ab5d 140e 0035 86fe 2e96 62b3 ce6c 0de3        .....5....b..l..
0x0020   5018 60f4 f2b6 0000 5349 524f 0241 5500        P.`......gov.au.
0x0030   0001 0001 0001 5180 0004 ab5d 1720 003b        ......Q........;
0x0040   9ab5 8000 0000 0001 0000 0000 0661 6c74        .............alt
0x0050   6d61                                           ma
```

```
16:05:53.314266 171.93.20.11.53 > 171.93.20.14.34558: . 3242:4702(1460) ack
75 win 24820 32768 inv_q+ [b2&3=0xb00] [6q] [28021a] [29 298n] [24953au]
Type51 (Class 39605)? ^@^@^@^@^A^@^@^@^@^Epptp7^Cnsw^Dc[|domain] (DF)
0x0000   4500 05dc 03c8 4000 4006 0c19 ab5d 140b        E.....@.@.......
0x0010   ab5d 140e 0035 86fe 2e96 6867 ce6c 0de3        .....5....hg.l..
0x0020   5010 60f4 009d  0000 0151 8000 0b00 0006        P.`......Q......
0x0030   6e73 776d 6169 c012 0033 9ab5 8000 0000        mailhost.3......
0x0040   0001 0000 0000 0570 7074 7037 036e 7377        .......pptp7.nsw
0x0050   0463                                           .c
16:05:53.314267 171.93.20.14.34558 > 171.93.20.11.53: . ack 1782 win 8760
(DF)
0x0000   4500 0028 0107 4000 ff06 558d ab5d 140e        E..(..@...U.....
0x0010   ab5d 140b 86fe 0035 ce6c 0de3 2e96 62b3        .......5.l....b.
0x0020   5010 2238 7394 0000 5555  5555 5555             P."8s...UUUUUU
.....
.....
16:05:53.394034 171.93.20.11.53 > 171.93.20.14.34558: P 76242:76962(720) ack
75 win 24820 21071 [b2&3=0x241] [1a] [21760q] [1n] [1au][|domain] (DF)
0x0000   4500 02f8 03fa 4000 4006 0ecb ab5d 140b        E. ....@.@.......
0x0010   ab5d 140e 0035 86fe 2e97 858f ce6c 0de3        .....5.......l..
0x0020   5018 60f4 ed47 0000 5349 524f 0241 5500        P.`..G…gov.au.
0x0030   0001 0001 0001 5180 0004 ab5d 1a35 0040        ......Q......5.@
0x0040   9ab5 8000 0000  0001 0000 0000 0b63 7265        .............cre
0x0050   6d6f                                           mo
16:05:53.398463 171.93.20.14.34558 > 171.93.20.11.53: . ack 74782 win 8760
(DF)
0x0000   4500 0028 0118 4000 ff06 557c ab5d 140e        E..(..@...U| ....
0x0010   ab5d 140b 86fe 0035 ce6c 0de3 2e97 7fdb        .......5.l......
0x0020   5010 2238 566b 0000 5555 5555 5555             P."8Vk..UUUUUU
16:05:53.402614 171.93.20.14.34558 > 171.93.20.11.53: F 75:75(0) ack 76962
win 8760 (DF)
0x0000   4500 0028  0119 4000 ff06 557b ab5d 140e        E..(..@...U{....
0x0010   ab5d 140b 86fe 0035 ce6c 0de3 2e97 885f        .......5.l.....
0x0020   5011 2238 4de6 0000 5555 5555 5555             P."8M...UUUUUU
16:05:53.402730 171.93.20.11.53 > 171.93.20.14.34558: . a ck 76 win 24820
(DF)
0x0000   4500 0028 03fb 4000 4006 119a ab5d 140b        E..(..@.@.......
0x0010   ab5d 140e 0035 86fe 2e97 885f ce6c 0de4        .....5..... .l..
0x0020   5010 60f4 0f2a 0000 5555 5555 5555             P.`..*..UUUUUU
....
....
16:05:53.398463 171.93.20.14.34558 > 171.93.20.11.53: . ack 74782 win 8760
(DF)
0x0000   4500 0028 0118 4000 ff06 557c ab5d 140e        E..(..@...U|....
0x0010   ab5d 140b 86fe 0035 ce6c 0de3 2e97 7fdb        .......5.l......
0x0020   5010 2238 566b 0000 5555 5555  5555             P."8Vk..UUUUUU
16:05:53.402614 171.93.20.14.34558 > 171.93.20.11.53: F 75:75(0) ack 76962
win 8760 (DF)
0x0000   4500 0028 0119 4000 ff06 557b ab5d 140e        E..(..@...U{....
0x0010   ab5d 140b 86fe 0035 ce6c 0de3 2e97 885f        ..... ..5.l....._
0x0020   5011 2238 4de6 0000 5555 5555 5555             P."8M...UUUUUU
16:05:53.402730 171.93.20.11.53 > 171.93.20.14.34558: . ack 76 win 24820
(DF)
0x0000   4500 0028 03fb 4000 4006 119a ab5d 140b        E..(..@.@.......
0x0010   ab5d 140e 003 5 86fe 2e97 885f ce6c 0de4        .....5..... .l..
0x0020   5010 60f4 0f2a 0000 5555 5555 5555             P.`..*..UUUUUU
```

From the output we can see that the primary DNS server sends a notify message to the slave DNS server for the nsw.acme.research.gov.au domain. The slave responds to the DNS server with a notify to say it has received the notify from the primary DNS server. The DNS zone transfer then beings. There is no indication inside the packets that signatures are being used. If there were we would see a string containing the algorithm type.

As we can see from the output there is not authentication traffic between the primary DNS server and the slave before the zone transfer begins.

Compliance with the Audit Control Objective

From the results of the test, there is no transaction signatures used to authenticate zone transfers. As a result, the DNS server fails to meet the requirements for compliance with the audit item.

Audit Item 1 - Logging is turned on for the DNS

1. Authorized and unauthorised zone transfers are logged

Test 1: Zone transfer from an unauthorised host.

```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

O:\>nslookup
Default Server:  dnsserver2.nsw.acme.research.gov.AU
Address:  171.93.20.14

> server 1 71.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

> ls -d nsw.acme.research.gov.au
[dnsserver1.nsw.acme.research.gov.AU]
*** Can't list domain nsw.acme.research.gov.au: Query refused
>
```

We can see that a zone transfer is denied.

We need to check the logs:

```
# grep 171.93.20.92 /var/adm/messages
Jul  5 21:22:04 dnsserver1.nsw.acme.research.gov.AU named[22620]: denied
AXFR from [171.93.20.92].3032 for "nsw.acme.research.gov.au" IN (acl)
#
```

We can see from the output that the logs are recording the unauthorised zone transfer attempts.

Test 2: Zone transfer from an authorised host.

```
dnsserver2.nsw.acme.research.gov.AU:/home/norrieb
35 >nslookup
Default Server:  dnsserver2.nsw.acme.research.gov.AU
Address:  171.93.20.14

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

> ls -d acme.research.gov.au
[dnsserver1.nsw.acme.research.gov.AU]
$ORIGIN acme.research.gov.au.
@                   3H IN SOA     dnsserver1.nsw hostmaster.nsw (
                                  2003051400    ; serial
                                  1H            ; refresh
                                  20M           ; retry
                                  4W            ; expiry
                                  3H )          ; minimum
                            . . .
                            . . .
```

From the above output we can see that a zone transfer has been successful. We now need to look at the log file:

```
# grep 171.93.20.14 /var/adm/messages
# grep dnsserver2 /var/adm/messages
Jul  5 21:25:29 dnsserver2.nsw.acme.research.gov.AU named[23754]: Zone
"hq.research.gov.au" (IN) SOA serial# (2001281524) rcvd from [171.93.99.207]
is < ours (2001281527): skipping
Jul  5 21:25:30 dnsserver2.nsw.acme.research.gov.AU named[23754]: owner name
"Merb_pc65.hq.research.gov.au" IN (secondary) is invalid  - proceeding anyway
Jul  5 21:25:30 dnsserver2.nsw.acme.research.gov.AU named[23754]: owner name
"Merb_PC89.hq.research.gov.au" IN (secondary) is invalid  - proceeding anyway
Jul  5 21:25:30 dnsserver2.nsw.acme.research.gov.AU named[23754]: owner name
"protein_coating.hq.research.gov.au" IN (secondary) is invalid  - proceeding
anyway
#
```

From the output of the log files, it is clear that there is no log entry for the successful zone transfer.

2. All unauthorised query attempts are logged

```
# nslookup
Default Server:  externalhost.telstra.bigpond.au
Address:  161.192.101.102

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

> www.google.com.au
Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

*** dnsserver1.nsw.acme.research.gov.AU can't find www.google.com.au: Query
refused
>
```

From the output we see we have a denied query. We will now examine the logs to see if it is recorded.

```
# grep 161.192.101.111 /var/adm/messages
Jul  5 21:36:40 dnsserver1.nsw.acme.research.gov.AU named[22620]: denied
query from [161.192.101.111].45391 for "www.google.com .au" IN
Jul  5 21:36:40 dnsserver1.nsw.acme.research.gov.AU named[22620]: denied
query from [161.192.101.111].45396 for "www.google.com.au.research.gov.au"
IN
#
```

From the output of the log file, we can see that indeed the DNS server logs unauthorised queries.

3. Unexpected DNS responses are logged.

Step 1. Start tcpdump in order to show unexpected responses

```
testmachine:/# tcpdump  -n -w queries.out
tcpdump: listening on eth0

5589 packets received by filter
0 packets dropped by kernel
testmachine:/#
```

Step 2. Run Script

```
testmachine:~# ./testdns.pl 203.23.194.11 171.93.20.11 6321
www.microsoft.com.au 203.23.194.11 5000

Generating unique transaction id numbers...
Sending the packets...
Done.
testmachine:~#
```

Step 3. Examine tcpdump of traffic

```
21:39:33.356632 ns2.codify.com.domain >
dnsserver1.nsw.acme.research.gov.au.6321:  61451 - 1/0/0 A ns2.codify.com
(54) (DF) [tos 0x10]
21:39:33.356920 dnsserver1.nsw.acme.research.gov.au > ns2.codify.com: icmp:
dnsserver1.nsw.acme.research.gov.au udp port 6321 unre achable (DF)
21:39:33.357216 ns2.codify.com.domain >
dnsserver1.nsw.acme.research.gov.au.6321:  22198 - 1/0/0 A ns2.codify.com
(54) (DF) [tos 0x10]
21:39:33.357386 dnsserver1.nsw.acme.research.gov.au > ns2.codify.com: icmp:
dnsserver1.nsw.acme.research.gov .au udp port 6321 unreachable (DF)
21:39:33.357705 ns2.codify.com.domain >
dnsserver1.nsw.acme.research.gov.au.6321:  41146 - 1/0/0 A ns2.codify.com
(54) (DF) [tos 0x10]
21:39:33.357870 dnsserver1.nsw.acme.research.gov.au > ns2.codify.com: icmp:
dnsserver1 .nsw.acme.research.gov.au udp port 6321 unreachable (DF)
21:39:33.358184 ns2.codify.com.domain >
dnsserver1.nsw.acme.research.gov.au.6321:  53917 - 1/0/0 A ns2.codify.com
(54) (DF) [tos 0x10]
21:39:33.358350 dnsserver1.nsw.acme.research.gov.au > ns2.codify .com: icmp:
dnsserver1.nsw.acme.research.gov.au udp port 6321 unreachable (DF)
21:39:33.358657 ns2.codify.com.domain >
dnsserver1.nsw.acme.research.gov.au.6321:  61268 - 1/0/0 A ns2.codify.com
(54) (DF) [tos 0x10]
21:39:33.358831 dnsserver1.nsw.acme.resear ch.gov.au > ns2.codify.com: icmp:
dnsserver1.nsw.acme.research.gov.au udp port 6321 unreachable (DF)
21:39:33.359134 ns2.codify.com.domain >
dnsserver1.nsw.acme.research.gov.au.6321:  4641 - 1/0/0 A ns2.codify.com (54)
(DF) [tos 0x10]
21:39:33.359298 dnsse rver1.nsw.acme.research.gov.au > ns2.codify.com: icmp:
dnsserver1.nsw.acme.research.gov.au udp port 6321 unreachable (DF)
21:39:33.359612 ns2.codify.com.domain >
dnsserver1.nsw.acme.research.gov.au.6321:  39035 - 1/0/0 A ns2.codify.com
(54) (DF) [tos 0x10]
21:39:33.359781 dnsserver1.nsw.acme.research.gov.au > ns2.codify.com: icmp:
dnsserver1.nsw.acme.research.gov.au udp port 6321 unreachable (DF)
21:39:33.360079 ns2.codify.com.domain >
dnsserver1.nsw.acme.research.gov.au.6321:  42716 - 1/0/0 A ns2.codify.com
(54) (DF) [tos 0x10]
21:39:33.360244 dnsserver1.nsw.acme.research.gov.au > ns2.codify.com: icmp:
dnsserver1.nsw.acme.research.gov.au udp port 6321 unreachable (DF)
21:39:33.360539 ns2.codify.com.domain >
dnsserver1.nsw.acme.research.gov.au.6321:  2777 - 1/0/0 A ns2.codify.com (54)
(DF) [tos 0x10]
21:39:33.360701 dnsserver1.nsw.acme.research.gov.au > ns2.codify.com: icmp:
dnsserver1.nsw.acme.research.gov.au udp port 6321 unreachable (DF)
21:39:33.361009 ns2.codify.com.domain >
dnsserver1.nsw.acme.research. gov.au.6321:  54299 - 1/0/0 A ns2.codify.com
(54) (DF) [tos 0x10]
```

```
21:39:33.361170 dnsserver1.nsw.acme.research.gov.au > ns2.codify.com: icmp:
dnsserver1.nsw.acme.research.gov.au udp port 6321 unreachable (DF)
```

Step 4. Examine log file

```
# grep –i codify /var/adm/messages
# grep 203.23.194.11 /var/adm/messages
# grep –i unexp /var/log/debug
Jul  7 10:14:03 dnsserver1.nsw.acme.research.gov.au named[22620]: Response
from unexpected source ([130.95.128.2].53) for query "56.128.95.130.in –
addr.arpa IN PTR"
#
```

From the output of each step in, but specifically in steps 3 and 4, we can see that DNS responses have been sent to the DNS server, however no information from the DNS server has been recorded in the log file for the spoofing test we performed. Several other tests performed with the modified script as mentioned in the checklist audit item 1, did not produce any results either. The reason for this may have been that the scripts were unsuccessful in sending the correct spoofed DNS ID response packet, or the information may have already been cached on the DNS server.

There is however a message in the log relating to an unexpected response, so while although the tests did not show up any messages, we can see from the logs that unexpected response packets are being recorded.

4. Unauthorised attempts to update the DNS are recorded.

Step 1. Configure the Windows 2000 client to register itself with the DNS server.

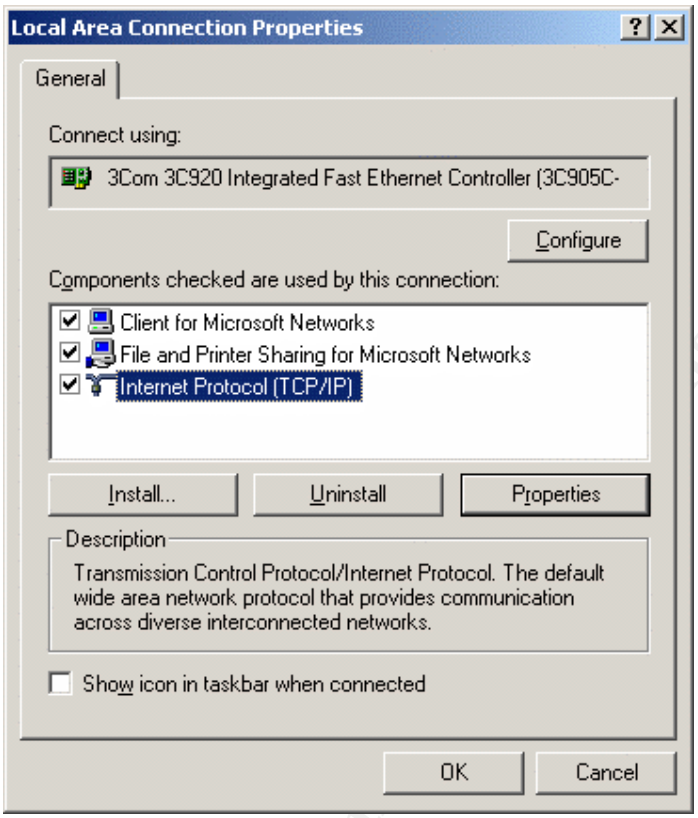

Figure 16 – Network Properti es Window

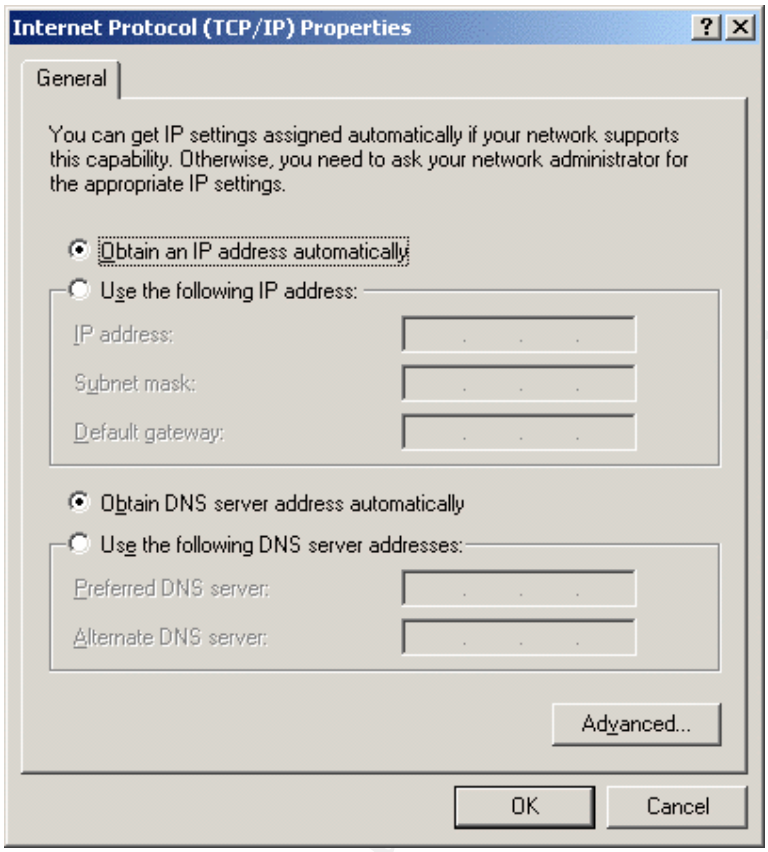

Figure 17 – TCP/IP Properties Window

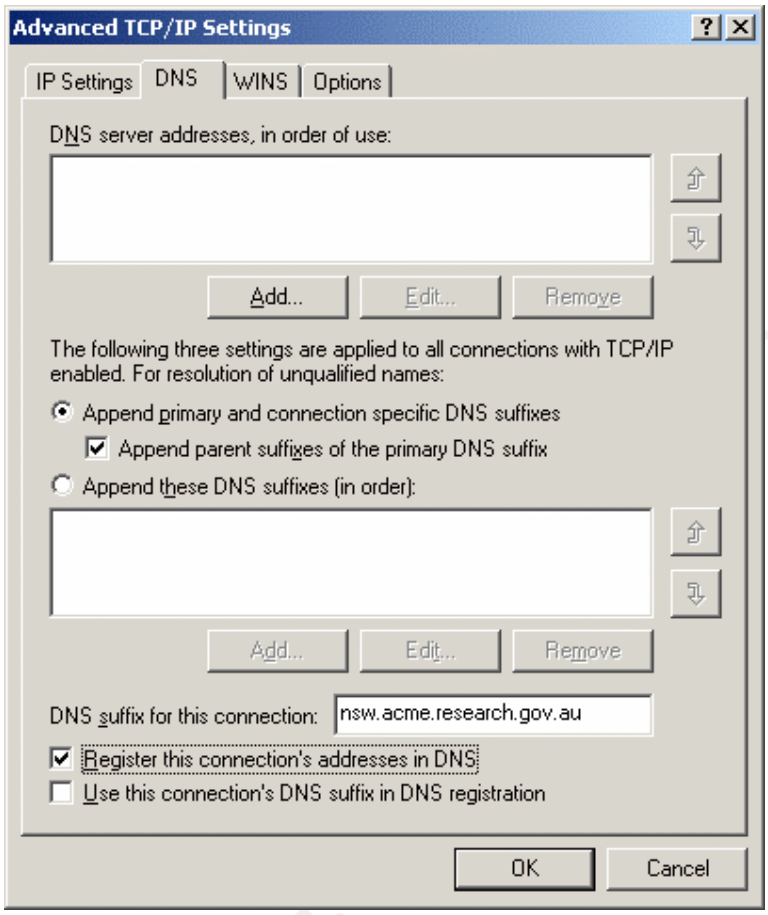

Figure 18 – TCP/IP DNS Settings

Once the windows 2000 client is setup to register the connection in the DNS, we can release and renew the IP address, this will force the client to update the information in the DNS.

Step2. Examine the log files.

```
# grep -i update | grep 171.93.20.92
Jul  5 07:58:58 dnsserver1.nsw.acme.research.gov.AU named[22620]: denied
update from [171.93.20.92].3014 for "25.93.171.in -addr.arpa" IN
#
```

The output of the log file shows that an update was recorded and it was denied.

5. Log details are kept for a minimum of 1 year.

The Audit results for Audit Item 2 show that the DNS server is backed up regularly and that there is a retention period of more than 1 year for all data backed up.

Compliance with Audit Control Objective

From the results of the tests we can see that the DNS server meets all the logging requirements, making it compliant with this audit objective.

Audit Item 11 - Dynamic Updates are restricted or turned off.

Test. Try an update on the DNS server from a host on the network

```
testmachine:~# nsupdate
> server 171.93.20.11
> zone nsw.acme.research.gov.au
> update add newmachine.nsw.acme.research.gov.au 60 A 171.93.25.253
> send
>
```

Now try to perform a lookup on the hostname.

```
testmachine:~# nslookup
Note:  nslookup is deprecated and m ay be removed from future releases.
Consider using the `dig' or `host' programs instead.  Run nslookup with
the `-sil[ent]' option to prevent this message from appearing.
> server 171.93.20.11
Default server: 171.93.20.11
Address: 171.93.20.11#53
> newmachine.nsw.acme.research.gov.au
Server:         171.93.20.11
Address:        171.93.20.11#53

** server can't find newmachine.nsw.acme.research.gov.au: NXDOMAIN
>
```

The lookup for the hostname from the nsupdate has failed.

We need to examine the logs to see what has happened.

```
# grep 171.93.20.111 /var/adm/messages
Jul  5 22:44:06 dnsserver1.nsw.acme.research.gov.AU named[22620]: denied
update from [171.93.20.111].2076 for "nsw.acme.research.gov.au" IN
#
```

From the output of the log file, we can see that the dynamic update was denied.

The outcome of the tests show that dynamic updates are restricted. But from the message we cannot determine whether the message is from an update restriction, or dynamic updates are turned off. In order to determine this we can actually stop and restart the DNS server by stopping and restarting the named or in.named process.

Compliance with the Audit Control Objective

As the DNS restricts the Dynamic Updates it is compliant with the audit item.

Audit Item 20 - DNS ID's are Randomised.

Test. Traffic is captured for the DNS server, and the DNS ID's are examined to see if there is a pattern between them.

```
testmachine:/# tcpdump  -r traffic.out  -n src host 171.93.20.11 | grep  -v
171.93.20.11.domain | grep ".53 " | grep  -v 171.93.20.11.53 | grep  -v icmp |
grep -v "arp reply" | grep  -v "udp" | more
16:54:49.632953 171.93.20.11.47607 > 171.93.194.32.53:  30717 SOA?
110.93.171.IN-ADDR.ARPA. (42) (DF)
16:55:49.665554 171.93.20.11.58939 > 171.93.66.10.53: P 0:43(43) ack 1 win
24820 54453 SOA? 66.93.171.IN-ADDR.ARPA. (41) (DF)
16:56:01.547053 171.93.20.11.47607 > 192.42.93.30.53:  54568 A?
dns4.fasttelco.net. (36) (DF)
16:56:35.625853 171.93.20.11.47607 > 195.86.134.127.53:  37855 A?
37.230.215.62.proxy.relays.osirusoft.com. (58) (DF)
16:56:48.467953 171.93.20.11.47607 > 205.231.29.244.53:  43333 TXT?
39.169.242.218.list.dsbl.org. (46) (DF)
16:57:57.625253 171.93.20.11.47607 > 171.93.181.1.53:  50067 notify
[b2&3=0x2400] SOA? 192.194.138.IN-ADDR.ARPA. (42) (DF)
16:57:57.631553 171.93.20 .11.47607 > 140.253.123.16.53:  30993 notify
[b2&3=0x2400] SOA? 192.194.138.IN-ADDR.ARPA. (42) (DF)
16:58:04.639553 171.93.20.11.47607 > 181.52.30.17.53:  8460 SOA?
hq.Research.gov.au. (32) (DF)
16:58:04.682058 171.93.20.11.58941 > 181.52.30.17.53: P 0:34( 34) ack 1 win
24840 33553 SOA? hq.Research.gov.au. (32) (DF)
16:58:05.625953 171.93.20.11.47607 > 140.253.81.16.53:  22882 notify
[b2&3=0x2400] SOA? 192.194.138.IN-ADDR.ARPA. (42) (DF)
17:00:14.294792 171.93.20.11.47607 > 194.109.6.152.53:  60953 A?
104.133.228.66.list.dsbl.org. (46) (DF)
17:00:39.689510 171.93.20.11.47607 > 140.186.128.222.53:  25953 A?
80.32.108.80.opm.blitzed.org. (46) (DF)
17:00:49.667653 171.93.20.11.58942 > 171.93.66.10.53: . ack 168 win 24820
(DF)
17:01:03.946853 171.93.20.11.47607 >  200.179.192.15.53:  32418 A?
16023933.rjo.virtua.com.br. (44) (DF)
17:01:21.768419 171.93.20.11.58943 > 181.52.30.17.53: . ack 36453 win 24840
(DF)
17:01:46.028353 171.93.20.11.47607 > 203.16.167.1.53:  30577 A?
67.249.8.218.spamsources.relays.osirusoft.c om. (63) (DF)
17:02:26.681329 171.93.20.11.47607 > 198.161.156.1.53:  6053 A?
athq356sy38wf.bc.hsia.telus.net. (49) (DF)
17:03:15.617084 171.93.20.11.47607 > 171.93.37.20.53:  27153 SOA?
39.93.171.IN-ADDR.ARPA. (41) (DF)
17:03:32.914155 171.93.20.11.47607  > 171.93.14.16.53:  16253 notify
[b2&3=0x2400] SOA? _msdcs.Research.gov.au. (33) (DF)
17:03:32.915324 171.93.20.11.47607 > 181.52.231.1.53:  7753 notify
[b2&3=0x2400] SOA? _msdcs.Research.gov.au. (33) (DF)
17:03:36.617755 171.93.20.11.47607 > 181.52.231.1. 53:  7753 notify
[b2&3=0x2400] SOA? _msdcs.Research.gov.au. (33) (DF)
17:03:36.618600 171.93.20.11.47607 > 171.93.14.16.53:  16253 notify
[b2&3=0x2400] SOA? _msdcs.Research.gov.au. (33) (DF)
17:03:44.623353 171.93.20.11.47607 > 140.253.72.81.53:  40258 not ify
[b2&3=0x2400] SOA? _msdcs.Research.gov.au. (33) (DF)
17:03:44.630022 171.93.20.11.47607 > 171.93.14.16.53:  16253 notify
[b2&3=0x2400] SOA? _msdcs.Research.gov.au. (33) (DF)
17:03:44.630834 171.93.20.11.47607 > 181.52.231.1.53:  7753 notify
[b2&3=0x2400] SOA? _msdcs.Research.gov.au. (33) (DF)
17:04:45.054953 171.93.20.11.123 > 171.93.31.255.123:  v3 bcast strat 2 poll
6 prec -15 (DF) [ttl 1]
17:06:18.774153 171.93.20.11.47607 > 203.202.150.20.53:  43863 A?
jasper.cqu.EDU.au. (35) (DF)
17:06:50.592581 171.93.20.11.47607 > 200.23.1.1.53:  53853 A?
nsmty2.uninet.net.mx. (38) (DF)
```

```
17:07:31.611239 171.93.20.11.47607 > 138.194.60.2.53:  1053 SOA?
forprod.Research.gov.au. (34) (DF)
17:08:04.606353 171.93.20.11.47607 > 171.93.194.161.53:  59019 SOA?
hq.Research.gov.au. (32) (DF)
17:08:04.979753 171.93.20.11.58946 > 181.52.30.17.53: . ack 157617 win 15574
(DF)
17:08:05.000153 171.93.20.11.58946 > 181.52.30.17.53: . ack 164308 win 8883
(DF)
testmachine:/#
```

From this output it can be seen that the DNS ID's appear to be random. I.e. 54568, 37885, 43333, etc.

Running zodiac will also sniff the DNS packets and inform us of hosts which are using sequential DNS ID's.

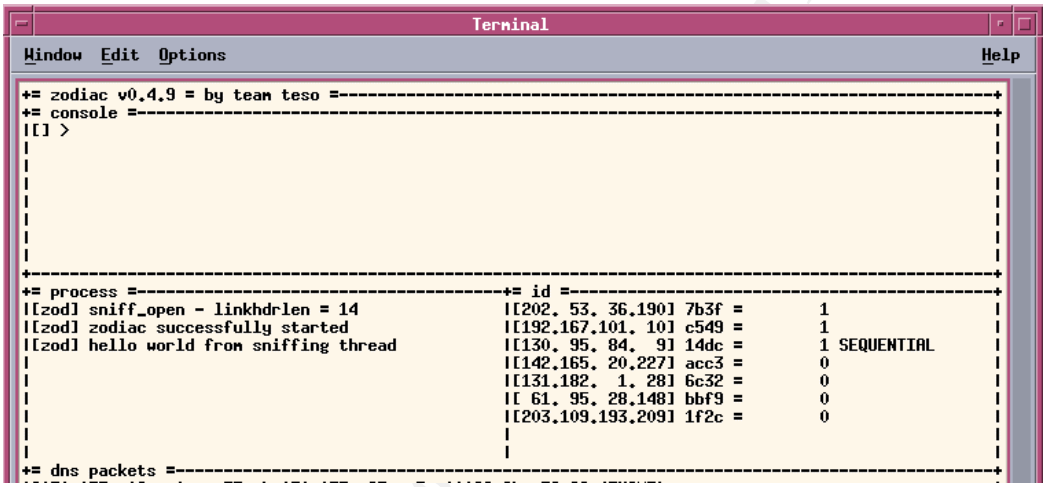

Figure 19 – Zodiac window showing sequential DNS ID's detected for a host

Using zodiac to try and spoof the DNS:

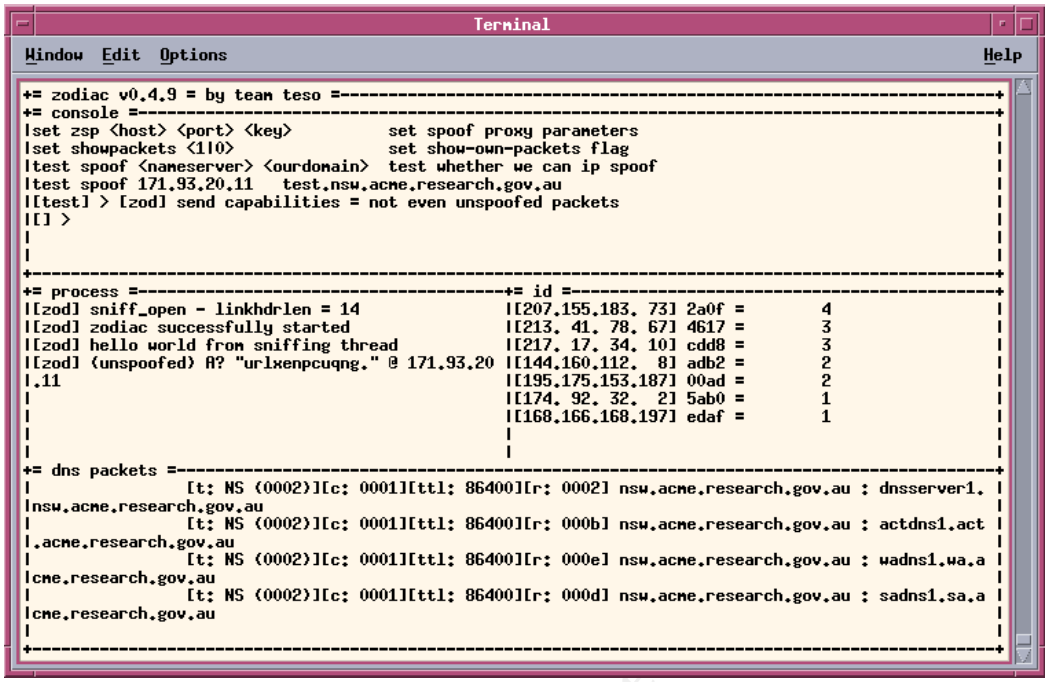

Figure 20 – Trying to spoof DNS server – fails to work

As can be seen from the above zodiac was unable to predict the DNS ID's for the DNS server.

Compliance with the Audit Item Control Objective

From the results of the tests, it can be seen that the DNS ID's are random, and not predictable. As a result the DNS server is compliant with this audit item.

Audit Item 15 - Fetch Glue Turned off

For this test we used the first method of testing which involved listening to traffic of the DNS server.

Step 1: Capture packets going to DNS server.

```
testmachine:/# tcpdump  -n -s 2000  -w traffic.out
tcpdump: listening on eth0
```

Step 2: Convert the raw capture file to human readable form, adding also a hexadecimal dump of the packet and an ASCII readable form (using the –X option).

```
testmachine:/# tcpdump  -n -X -s 2000  -r traffic.out > testdump.txt
```

Step 3: See if there are any response packets which do not have an A pointer (IP address) for the name server in the list in the additional records section. The tcpdump output shows the number of answers returned, the number of authoritative name servers returned and the number of additional records

which should contain the IP addresses for the authoritative name servers returned, this is represented by "#/#/#' in the output of tcpdump. For this we search for DNS packets where the list of name servers and the number of additional records do not match. Preferably if we find a response packet to the DNS server with "#/#/0' then this means that the IP addresses are not in the response. Note this response packet should not have an authoritative answer as this would mean that the final packet has been received the answer to the query sent, therefore we do not want a "*" in the tcpdump packet record.

In looking at the tcpdump output, there are a number of records with "0/1/0" these are generally authoritative responses from the authoritative server with an answer to the query which has the IP address of the host being queried or a response such as host doesn't exist (NXDOMAIN). As a result to find response packets which the server may then use to look up the tcpdump output can be filtered.

```
testmachine:/# grep "/0 " testdump.txt | grep " 0/" | grep  -v "0/1/0" | grep
-v "0/0/0" | more
```

This produces the following results:

```
17:10:16.123718 ns.ripe.net.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  62852 - 0/2/0 (92) (DF)
17:10:20.649804 b.gtld -servers.net.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  61829 - 0/3/0 (109)
17:10:59.477897 munnari.OZ.au.doma in >
dnsserver1.nsw.acme.research.gov.au.47607:  28003 - 0/3/0 (100)
17:10:59.678765 munnari.OZ.au.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  53526 - 0/3/0 (100)
17:11:15.252586 arrowroot.arin.net.domain >
dnsserver1.nsw.acme.research.gov.au.47607:   31365- 0/3/0 (107) (D F)
17:11:39.621504 munnari.OZ.au.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  43563 - 0/3/0 (100)
17:11:54.948587 a3.NSTLD.COM.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  33988 - 0/4/0 (134) (D F)
17:12:21.820813 munnar i.OZ.au.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  42501 - 0/3/0 (100)
17:12:37.428711 munnari.OZ.au.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  30971 - 0/3/0 (100)
17:13:54.187815 ns.ripe.net.domain >
dnsserver1.nsw.acme.research.gov.au.4 7607:  51592- 0/5/0 (185) (D F)
17:13:57.870959 munnari.OZ.au.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  42063 - 0/3/0 (100)
17:14:05.351859 h.gtld -servers.net.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  13665 - 0/3/0 (109) (D F)
17:14:21.23 5343 ns.ripe.net.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  20405 - 0/4/0 (142) (D F)
```

There was a large number of records returned, so the list shown is only an excerpt from the whole data, but it gives us enough information to continue.
Notice also from the results that none of the responses are authoritative, there is no "*". This means the DNS server needs to lookup the rest of the information if it want's to resolve the query. As there are no address pointer records in the responses, if the DNS server tries to lookup the address of a name server, in the response packet, then fetch glue is turned on.

Looking at the first packet in the above list:

```
17:10:16.123718 ns.ripe.net.domain >
dnsserver1.nsw.acme.research.gov.au.47607:   62852- 0/2/0 (92) (DF)
0x0000   4500 0078 0000 4000 2f11 f717 c100 00c1        E..x..@./.......
0x0010   ab5d 140b 0035 b9f7 0064 ce3b f584 8000        .....5...d.;....
0x0020   0001 0000 0002 0000 0231 3903 3130 3103        .........19.101.
0x0030   3231 3802 3830 0769 6e2 d 6164 6472 0461        218.80.in -addr.a
0x0040   7270 6100 000c 0001 c00f 0002 0001 0001        rpa.............
0x0050   5180 0012 03 6e 7331 0863 6162 6c65 636f        Q.... ns1.cableco
0x0060   6d03 6e65 7400 c00f 0002 0001 0001 5180        m.net.........Q.
0x0070   0006 036e 7332 c03c                            ...ns2.<
```

From the hex and acsii output, we can see that a reverse lookup is actually being performed, looking for the hostname for the IP address 80.218.101.19. The name server specified is ns1.cablecom.net.

If fetch glue is turned on, then we should find a query from our DNS server to for the address of ns1.cablecom.net and then a query to it.

```
17:10:16.124471 dnsserver1.nsw.acme.research.gov.au.47607 >
ns1.cablecom.net.domain:  16049 PTR? 19.10 1.218.80.in-addr.arpa. (44) (DF)
0x0000   4500 0048 b71f 4000 ff11 d3e1 ab5d 140b        E..H..@.........
0x0010   3e02 2005 b9f7 0035 0034 c2ec 3eb1 0000        >......5.4..>...
0x0020   0001 0000 0000 0000 0231 3903 3130 3103        .........19.101.
0x0030   3231 3802 3830 0769 6e2d 6164 6472 0461        218.80.in  -addr.a
0x0040   7270 6100 000c 0001                            rpa.....
```

From the above output we can see that the DNS server has contacted ns1.cablecom.net. However the output of the tcpdump did not show any address query. Usually you would expect something of the following form:

```
17:10:20.473030 dnsserver1.nsw.acme.research.gov.au.47607 >
ns.ripe.net.domain:  61829 A? ns1.cablecom.net. (37) (DF)
```

There may be several reasons for this. Firstly as the DNS server is very busy the packet may not have been captured via tcpdump. Secondly the DNS server may have had the address cached.

A look at other records in the tcpdump output show other examples. A search based on " A?" pattern will show the request for an address of a machine.

Performing a grep shows up some interesting results:

```
17:09:45.012273 dnsserver1.nsw.acme.research.gov.au.47607 > ns.bacs  -
net.hu.domain:  22307 A? ns.inc.hu. (27) (DF)
. . .
17:10:20.473030 dnsserver1.nsw.acme.research.gov .au.47607 > b.gtld-
servers.net.domain:  61829 A? ns3.melbourneit.com. (37) (DF)
17:14:05.041007 dnsserver1.nsw.acme.research.gov.au.47607 > h.gtld  -
servers.net.domain:  13665 A? ns3.melbourneit.com. (37) (DF)
```

In examining the traffic related to these queries we find the following:

```
17:10:20.649804 b.gtld -servers.net.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  61829 - 0/3/0 (109)
0x0000   4500 0089 07cf 0000 2b11 26ba c021 0e1e        E.......+.&..!..
0x0010   ab5d 140b 0035 b9f7 0075 6b93 f185 8000         .....5...uk.....
0x0020   0001 0000 0003 0000 036e 7333 0b6d 656c        .........ns3.mel
```

```
0x0030   626f 7572 6e65 6974 0363 6f6d 0000 0100        bourneit.com....
0x0040   01c0 1000 0200 0100 02a3 0000 1803 4e53        ..............NS
0x0050   310b 4d45 4c42 4f55 524e 4549 5403 434f        1.MELBOURNEIT.CO
0x0060   4d02 4155 00c0 1000 0200 0100 02a3 0000        M.au............
0x0070   0603 4e53 32c0 35c0 1000 0200 0100 02a3        ..NS2.5.........
0x0080   0000 0603 4e53 34c0 35                          ....NS4.5
                . . .
17:14:05.039555 nswmail.nsw.acme.research.gov.au.65099 >
dnsserver1.nsw.acme.research.gov.au.domain: 64794+ MX? sesahs.nsw.gov.au.
(35) (DF)
0x0000   4500 003f 370e 4000 ff11 1f65 ab5d 1403        E..?7.@....e....
0x0010   ab5d 140b fe4b 0035 002b 6910 fd1a 0100        .....K.5.+i.....
0x0020   0001 0000 0000 0000 0673 6573 6168 7303        .........sesahs.
0x0030   6e73 7703 676f 7602 6175 0000 0f00 01          nsw.gov.au.....
17:14:05.040402 dnsserver1.nsw.acme.resea rch.gov.au.47607 >
box2.aunic.net.domain:  42183 A? gw2.sesahs.nsw.gov.au. (39) (DF)
0x0000   4500 0043 36d2 4000 ff11 505c ab5d 140b        E..C6.@...P \....
0x0010   cbca 9614 b9f7 0035 002f cad5 a4c7 0000        .......5./......
0x0020   0001 0000 0000 0 000 0367 7732 0673 6573        .........gw2.ses
0x0030   6168 7303 6e73 7703 676f 7602 6175 0000        ahs.nsw.gov.au..
0x0040   0100 01                                        ...
17:14:05.041007 dnsserver1.nsw.acme.research.gov.au.47607 > h.gtld -
servers.net.domain:  13665 A? ns3.melbourneit.com. (37) (DF)
0x0000   4500 0041 542f 4000 ff11 648b ab5d 140b        E..AT/@...d.....
0x0010   c036 701e b9f7 0035 002d 410d 3561 0000        .6p....5. -A.5a..
0x0020   0001 0000 0000 0000 03 6e 7333 0b6d 656c        .........ns3.mel
0x0030   626f 7572 6e65 6974 0363 6f6d  0000 0100        bourneit.com....
0x0040   01                                             .
17:14:05.041275 dnsserver1.nsw.acme.research.gov.au.domain >
nswmail.nsw.acme.research.gov.au.65099: 64794 1/ 9/8 MX
gw2.sesahs.nsw.gov.au. 10 (413) (DF)
0x0000   4500 01b9 0d02 4000 ff11 47f7 ab5d 140b        E.....@...G.....
0x0010   ab5d 1403 0035 fe4b 01a5 c049 fd1a 8180        .....5.K...I....
0x0020   0001 0001 0009 0008 0673 6573 6168 7303        .........s esahs.
0x0030   6e73 7703 676f 7602 6175 0000 0f00 01c0        nsw.gov.au......
0x0040   0c00 0f00 0100 001f 5800 0800 0a03 6777        ........X.....gw
0x0050   32c0 0cc0 1700 0200 0100 010b 3000 1503        2...........0...
0x0060   6e73 310b 6175 7372 6 567 6973 7472 7903        ns1.ausregistry.
0x0070   6e65 7400 c017 0002 0001 0001 0b30 0006        net..........0..
0x0080   036e 7332 c047 c017 0002 0001 0001 0b30        .ns2.G.........0
0x0090   0006 036e 7333 c047 c017 0002 0001 0001        ...ns3.G... .....
0x00a0   0b30 0015 03 6e 7333 0b6d 656c 626f 7572        .0...ns3.melbour
0x00b0   6e65 6974 0363 6f6d  00c0 1700 0200 0100        neit.com........
0x00c0   010b 3000 0603 6e73 34c0 47c0 1700 0200        ..0...ns4.G.....
0x00d0   0100 010b 3000 0d04 62 6f 7832 0561 756e        ....0...box2.aun
0x00e0   6963 c053 c017 0002 0001 0001 0b30 000f        ic.S.........0..
0x00f0   0464 6e73 3107 7465 6c73 7472 61c0 53c0        .dns1.telstra.S.
0x0100   1700 0200 0100 010b 3000 0e05 6175 326c        ........0... au2l
0x0110   6405 7265 7365 6fc0 1bc0 1700 0200 0100        d...............
0x0120   010b 3000 1407 6175 646e 7330 3103 7379        ..0...audns01.sy
0x0130   6405 6f70 7475 73c0 53c0 4300 0100 0100        d.optus.S.C.....
0x0140   0250 ae00 04cb 1238 29c 0 6400 0100 0100        .P.....8).d.....
0x0150   0250 ae00 04cb 1238 2ac0 7600 0100 0100        .P.....8*.v.....
0x0160   0250 ae00 04cb 1238 2bc0 a900 0100 0100        .P.....8+.......
0x0170   0250 ae00 04d2 080f fdc0 bb00 0100 0100        .P..........  ...
0x0180   0250 a400 04cb ca96 14c0 d400 0100 0100        .P..............
0x0190   0250 ae00 04cb 3205 c8c0 ef00 0100 0100        .P....2.........
0x01a0   0151 8000 0482 7402 15c1 0900 0100 0100        .Q....t.........
0x01b0   00eb e800 04d2 3114 d0                          ......1..
```

From the output above it appears that the DNS server is looking up the address of ns3.melbourneit.com. As can be seen in the later section, it is returning the address in it's response to mailserver.nsw.acme.research. gov.au when searching for the address of seasahs.nsw.gov.au. It is clear from here that the DNS server is asking for the IP address of the nameserver ns3.melbourneit.com in response to another query. However again there

appears to be packets missing – obviously there is a lot of traffic going past the DNS server and tcpdump has not managed to capture all the packets.

To verify this however we can try an nslookup for ns3.melbourneit.com. The following test was performed a couple of days later. This way we know the cache of the data will have timed out.

```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

O:\>nslookup
Default Server:  dnsserver2.nsw.acme.research.gov.AU
Address:  171.93.20.14

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

> ns3.melbourneit.com.
Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

Non-authoritative answer:
Name:    ns3.melbourneit.com
Address:  203.27.227.10

>
```

A tcpdump of this query is as follows:

```
13:37:45.687644 testmachine.nsw.acme.research.gov.au.4931 >
dnsserver1.nsw.acme.research.gov.au.domain:
 3+ A? ns3.melbourneit.com. (37)
0x0000   4500 0041 7315 0000 8011 9903 ab5d 195c        E..As.........  \
0x0010   ab5d 140b 134 3 0035 002d b07d 0003 0100        .....C.5. -.}....
0x0020   0001 0000 0000 0000 036e 7333 0b6d 656c        .........ns3.mel
0x0030   626f 7572 6e65 6974 0363 6f6d 0000 0100        bourneit.com....
0x0040   01                                             .
13:37:45.694404 dnsserver1.nsw.acme.research.gov.au.47607 > aus139537 -
1.gw.connect.com.au.domain:
 46902 A? NS1.MELBOURNEIT.COM.au. (40) (DF)
0x0000   4500 0044 2777 4000 ff11 df8f ab5d 140b        E..D'w@.........
0x0010   d208 0ffd b9f7 0035 0030 9960 b73 6 0000        .......5.0.`.6..
0x0020   0001 0000 0000 0000 034e 5331 0b4d 454c        .........NS1.MEL
0x0030   424f 5552 4e45 4954 0343 4f4d 0241 5500        BOURNEIT.COM.au.
0x0040   0001 0001                                      ....
13:37:45.700065 dn sserver1.nsw.acme.research.gov.au.47607 > aus139537 -
1.gw.connect.com.au.domain:
 1565 A? NS2.MELBOURNEIT.COM.au. (40) (DF)
0x0000   4500 0044 2778 4000 ff11 df8e ab5d 140b        E..D'x@.........
0x0010   d208 0ffd b9f7 0035 0030 4a79 061d 0000        .... ...5.0Jy....
0x0020   0001 0000 0000 0000 034e 5332 0b4d 454c        .........NS2.MEL
0x0030   424f 5552 4e45 4954 0343 4f4d 0241 5500        BOURNEIT.COM.au.
0x0040   0001 0001                                      ....
13:37:45.705520 dnsserver1.nsw.acme. research.gov.au.47607 > aus139537 -
1.gw.connect.com.au.domain:
 51057 A? NS4.MELBOURNEIT.COM.au. (40) (DF)
0x0000   4500 0044 2779 4000 ff11 df8d ab5d 140b        E..D'y@.........
0x0010   d208 0ffd b9f7 0035 0030 8922 c771 0000        .......5.0.".q..
0x0020   0001 0000 0000 0000 034e 5334 0b4d 454c        .........NS4.MEL
```

```
0x0030   424f 5552 4e45 4954 0343 4f4d 0241 5500        BOURNEIT.COM.au.
0x0040   0001 0001                                      ....
13:37:45.714429 dnsserver1.nsw.acme.research.gov.au.4 7607 > b.gtld-
servers.net.domain:  7002 A? ns
3.melbourneit.com. (37) (DF)
0x0000   4500 0041 3526 4000 ff11 e5a9 ab5d 140b        E..A5&@.........
0x0010   c021 0e1e b9f7 0035 002d bd29 1b5a 0000        .!.....5. -.).Z..
0x0020   0001 0000 0000 0000 036e 7 333 0b6d 656c        .........ns3.mel
0x0030   626f 7572 6e65 6974 0363 6f6d 0000 0100        bourneit.com....
0x0040   01                                             .
13:37:45.719649 aus139537 -1.gw.connect.com.au.domain >
dnsserver1.nsw.acme.research.gov .au.47607:
 46902- 0/3/3 (138) (DF)
0x0000   4500 00a6 0000 4000 3311 d2a5 d208 0ffd        E.....@.3.......
0x0010   ab5d 140b 0035 b9f7 0092 42ea b736 8000        .....5....B..6..
0x0020   0001 0000 0003 0003 034e 5331 0b4d 454c        .........NS1.MEL
0x0030   424f 5552 4e45 4954 0343 4f4d 0241 5500        BOURNEIT.COM.au.
0x0040   0001 0001 c010 0002 0001 0000 0e10 0002        ................
0x0050   c00c c010 0002 0001 0000 0e10 0006 036e        ...............n
0x0060   7332 c010 c010 0002 0001 0000  0e10 0006        s2..............
0x0070   036e 7334 c010 c00c 0001 0001 0000 0e10        .ns4............
0x0080   0004 cb1f c6c4 c042 0001 0001 0000 0e10        .......B........
0x0090   0004 cb1b e332 c054 0001 0001 0000 0e10        .....2.T........
0x00a0   0004 cb1f c787                                 ......
13:37:45.723908 aus139537 -1.gw.connect.com.au.domain >
dnsserver1.nsw.acme.research.gov.au.47607:
 1565- 0/3/3 (138) (DF)
0x0000   4500 00a6 0000 4000 3311 d2a5 d208 0ffd        E.....@.3.......
0x0010   ab5d 140b 0035 b9f7 0092 f403 061d 8000        .....5..........
0x0020   0001 0000 0003 0003 034e 5332 0b4d 454c        .........NS2.MEL
0x0030   424f 5552 4e45 4954 0343 4f4d 0241 5500        BOURNEIT.COM.au.
0x0040   0001 0001 c010 0002 0001 000 0 0e10 0002        ................
0x0050   c00c c010 0002 0001 0000 0e10 0006 036e        ...............n
0x0060   7334 c010 c010 0002 0001 0000 0e10 0006        s4..............
0x0070   036e 7331 c010 c054 0001 0001 0000 0e10        .ns1...T........
0x0080   0004 cb1f c6c4 c00c 0001 0001 0000 0e10        ................
0x0090   0004 cb1b e332 c042 0001 0001 0000 0e10        .....2.B........
0x00a0   0004 cb1f c787                                 ......
13:37:45.727840 aus139537 -1.gw.connect.com.au.do main >
dnsserver1.nsw.acme.research.gov.au.47607:
 51057- 0/3/3 (138) (DF)
0x0000   4500 00a6 0000 4000 3311 d2a5 d208 0ffd        E.....@.3.......
0x0010   ab5d 140b 0035 b9f7 0092 32bd c771 8000        .....5....2..q..
0x0020   0001 0000 0003 0003 034e 5 334 0b4d 454c        .........NS4.MEL
0x0030   424f 5552 4e45 4954 0343 4f4d 0241 5500        BOURNEIT.COM.au.
0x0040   0001 0001 c010 0002 0001 0000 0e10 0006        ................
0x0050   036e 7332 c010 c010 0002 0001 0000 0e10        .ns2............
0x0060   0002 c00c c010 0002 0001 0000 0e10 0006        ................
0x0070   036e 7331 c010 c054 0001 0001 0000 0e10        .ns1...T........
0x0080   0004 cb1f c6c4 c034 0001 0001 0000 0e10        .......4........
0x0090   0004 cb1b e332 c00c 0001 00 01 0000 0e10        .....2..........
0x00a0   0004 cb1f c787                                 ......
13:37:45.893641 b.gtld -servers.net.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  7002 - 0/3/
0 (109)
0x0000   4500 0089 5050 0000 2b11 de38 c021 0e1e         E...PP..+..8.!..
0x0010   ab5d 140b 0035 b9f7 0075 41bf 1b5a 8000        .....5...uA..Z..
0x0020   0001 0000 0003 0000 036e 7333 0b6d 656c        .........ns3.mel
0x0030   626f 7572 6e65 6974 0363 6f6d 0000 0100        bourneit.com....
0x0040   01c0  1000 0200 0100 02a3 0000 1803 4e53        ..............NS
0x0050   310b 4d45 4c42 4f55 524e 4549 5403 434f        1.MELBOURNEIT.CO
0x0060   4d02 4155 00c0 1000 0200 0100 02a3 0000        M.au............
0x0070   0603 4e53 32c0 35c0 1000 0200 0100 02a3         ..NS2.5.........
0x0080   0000 0603 4e53 34c0 35                         ....NS4.5
13:37:45.900191 dnsserver1.nsw.acme.research.gov.au.47607 >
ns4.melbourneit.com.au.domain:  43966
A? ns3.melbourneit.com. (37) (DF)
0x0000   4500 0041 0e53 4000 ff11 4 815 ab5d 140b        E..A.S@...H.....
```

```
0x0010   cb1f c787 b9f7 0035 002d 685d abbe 0000        .......5. -h]....
0x0020   0001 0000 0000 0000 036e 7333 0b6d 656c        .........ns3.mel
0x0030   626f 7572 6e65 6974 0363 6f6d 0000 0100        bourneit.com....
0x0040   01                                             .
13:37:46.035867 ns4.melbourneit.com.au.domain >
dnsserver1.nsw.acme.research.gov.au.47607:  43966
1/3/3 A ns3.melbourneit.com (173) (DF)
0x0000   4500 00c9 0000 4000 3311 21e1 cb1f c787        E... ..@.3.!.....
0x0010   ab5d 140b 0035 b9f7 00b5 fcf0 abbe 8080        .....5..........
0x0020   0001 0001 0003 0003 036e 7333 0b6d 656c        .........ns3.mel
0x0030   626f 7572 6e65 6974 0363 6f6d 0000 0100        bourneit.com....
0x0040   01c0 0c00 0100  0100 0006 4800 04cb 1be3        ..........H.....
0x0050   0ac0 1000 0200 0100 0006 4800 1803 6e73        ..........H...ns
0x0060   320b 4d65 6c62 6f75 726e 6549 5403 636f        2.MelbourneIT.co
0x0070   6d02 6175 00c0 1000 0200 0100 0006 4800        m.au. .........H.
0x0080   0603 6e73 34c0 45c0 1000 0200 0100 0006        ..ns4.E.........
0x0090   4800 0603 6e73 31c0 45c0 7700 0100 0100        H...ns1.E.w.....
0x00a0   000e 1000 04cb 1fc6 c4c0 4100 0100 0100        ..........A.....
0x00b0   000e 1000 04cb 1 be3 32c0 6500 0100 0100        ........2.e.....
0x00c0   000e 1000 04cb 1fc7 87                         .........
13:37:46.052934 dnsserver1.nsw.acme.research.gov.au.domain >
testmachine.nsw.acme.research.gov.au.4931:
 3 1/3/3 A ns3.melbourneit.com (192) ( DF)
0x0000   4500 00dc 90f5 4000 ff11 bb87 ab5d 140b        E.....@.........
0x0010   ab5d 195c 0035 1343 00c8 a333 0003 8180        ... \.5.C...3....
0x0020   0001 0001 0003 0003 036e 7333 0b6d 656c        .........ns3.mel
0x0030   626f 7572 6e65 6974 0363  6f6d 0000 0100        bourneit.com....
0x0040   0103 6e73 330b 6d65 6c62 6f75 726e 6569        ..ns3.melbournei
0x0050   7403 636f 6d00 0001 0001 0000 0648 0004        t.com........H..
0x0060   cb1b e30a c029 0002 0001 0002 8c63 0018        .....).......c ..
0x0070   034e 5331 0b4d 454c 424f 5552 4e45 4954        .NS1.MELBOURNEIT
0x0080   0343 4f4d 0241 5500 c029 0002 0001 0002        .COM.au..)......
0x0090   8c63 0006 034e 5332 c058 c029 0002 0001        .c...NS2.X.)....
0x00a0   0002 8c63 0006 034e 5334  c058 c054 0001        ...c...NS4.X.T..
0x00b0   0001 0000 0d5c 0004 cb1f c6c4 c078 0001        ..... \.......x..
0x00c0   0001 0000 0d5c 0004 cb1b e332 c08a 0001        ..... \.....2....
0x00d0   0001 0000 0d5c 0004 cb1f c787                  ..... \......
```

The results of the query to ns3.melbourneit.com show that three other name server addresses are looked up and then one of the name servers ns4.melbourneit.com is looked up to obtain the IP address of ns3. melbourneit.com.

The previous tests may be hard to reproduce as they have just been captures of DNS traffic. In order to reproduce a test, we need to be able to know a DNS server will return just the names of names servers and not the IP addresses. In order to do this, we have created our own sub-domain called test.acme.research.gov.au, this has another sub domain called domain.test. acme.research.gov.au. The name servers for test.acme.research.gov.au point to primary DNS server dnsserver1, and our test DNS server, as well as 2 other addresses – ns1.perterhost.ru and ns1.pchome.org. The primary DNS server knows the IP addresses of itself and the test DNS server, however it does not know the IP addresses of the other 2 name servers. When a nslookup on the domain.test.acme.research.gov.au is performed and fetch glue is turned on, the DNS server will send a request to find the IP addresses of ns1.peterhost.ru and ns1.pchome.org. The TTL for the records is 3600 seconds, so that the cache times out quickly for further testing.

The output of the nslookup command for the domain.test.acme.research. gov.au is as follows:

```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

O:\>nslookup
Default Server:  dnsserver2.nsw.acme.research.gov.au
Address:  171.93.20.14

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.au
Address:  171.93.20.11

> set debug
> domain.test.acme.research.gov.au.
Server:  dnsserver1.nsw.acme.research.gov.au
Address:  171.93.20.11

------------
Got answer:
    HEADER:
        opcode = QUERY, id =  5, rcode = NOERROR
        header flags:  response, auth. answer, want recursion, recursion
avail.
        questions = 1,  answers = 1,  authority records = 4,  additional = 2

    QUESTIONS:
        domain.test.acme.research.gov.au, type = A, class = IN
    ANSWERS:
    ->  domain.test.acme.research.gov.au
        Internet address = 171.93.20.17
        ttl = 3600 (1 hour)
    AUTHORITY RECORDS:
    ->  test.acme.research.gov.au
        nameserver = ns1.pchome.org
        ttl = 3600 (1 hour)
    ->  test.acme.research.gov.au
        nameserver = ns1.peterhost.ru
        ttl = 3600 (1 hour)
    ->  test.acme.research.gov.au
        nameserver = dnsserver1.nsw.acme.research.gov.au
        ttl = 3600 (1 hour)
    ->  test.acme.research.gov.au
        nameserv er = test.acme.research.gov.au
        ttl = 3600 (1 hour)
    ADDITIONAL RECORDS:
    ->  dnsserver1.nsw.acme.research.gov.au
        Internet address = 171.93.20.11
        ttl = 86400 (1 day)
    ->  test.acme.research.gov.au
        Internet address =  171.93.20.17
        ttl = 3600 (1 hour)

------------
Name:    domain.test.acme.research.gov.au
Address:  171.93.20.17

>
```

From the output we cannot actually see that fetch glue was done. This may be due to the time for the external name servers to respond with the IP addresses.

The tcpdump of the query will be able to show what actually occurred:

```
19:16:37.799343 171.93.20.92.2460 > 171.93.20.11.53:  5+ A?
domain.test.acme.research.gov.au. (45)
0x0000   4500 0049 f984 0000 8011 128c ab5d 195c        E..I ..........\
0x0010   ab5d 140b 099c 0035 0035 be6a 0005 0100        .......5.5.j....
0x0020   0001 0000 0000 0000 0664 6f6d 6169 6e06        .........domain.
0x0030   7465 7374 6060 0461 636d 6508 7265 7365        test.acme.resear
0x0040   7202 6175 6175  6175 6175 0000 0100 01          ch.gov.au......
19:16:37.800225 171.93.20.11.51119 > 192.41.162.36.53:  45063 A?
ns1.pchome.org. (32) (DF)
0x0000   4500 003c 2fcd 4000 ff11 56f9 ab5d 140b        E..</.@...V.....
0x0010   c029 a224 c7af 0035 0028 0769 b007  0000        .).$...5.(.i....
0x0020   0001 0000 0000 0000 036e 7331 0670 6368        .........ns1.pch
0x0030   6f6d 6503 6f72 6700 0001 0001                   ome.org.....
19:16:37.800665 171.93.20.11.51119 > 192.112.36.4.53:  43678 A?
ns1.peterhost.ru. (34 ) (DF)
0x0000   4500 003e 71cc 4000 ff11 92d1 ab5d 140b        E..>q.@.........
0x0010   c070 2404 c7af 0035 002a 77b6 aa9e 0000        .p$....5.*w.....
0x0020   0001 0000 0000 0000 036e 7331 0970 6574        .........ns1.pet
0x0030   6572 686f 7374 0272 7 500 0001 0001             erhost.ru.....
19:16:37.800871 171.93.20.11.53 > 171.93.20.92.2460:  5* 1/4/2 A
171.93.20.17 (197) (DF)
0x0000   4500 00e1 f3d8 4000 ff11 589f ab5d 140b        E.....@...X.....
0x0010   ab5d 195c 0035 099c 00cd 743e 0005 8580          ...\.5....t>....
0x0020   0001 0001 0004 0002 0664 6f6d 6169 6e06        .........domain.
0x0030   7465 7374 6060 0461 636d 6508 7265 7365        test.acme.resear
0x0040   7202 6175 0000 0100 01c0 0c00 0100 0100        ch.gov.au.......
0x0050   000e 100 0 0482 9b10 0bc0 1300 0200 0100        ................
0x0060   000e 1000 1003 6e73 3106 7063 686f 6d65        ......ns1.pchome
0x0070   036f 7267 00c0 1300 0200 0100 000e 1000        .org............
0x0080   1203 6e73 3109 7065 7465 7268 6f73 7402          ..ns1.peterhost.
0x0090   7275 00c0 1300 0200 0100 000e 1000 0d06        ru..............
0x00a0   646e 7373 6572 036e 7377 c01a c013 0002        dnsserver1.nsw..
0x00b0   0001 0000 0e10 0009 0674 6573 7460 60c0        .........test...
0x00c0   13c0 8300  0100 0100 0151 8000 0482 9b10        .........Q......
0x00d0   01c0 9c00 0100 0100 000e 1000 0482 9b10        ................
0x00e0   0b                                              .
```

From the output, it can be seen that the DNS server did queries for ns1.peterhost.ru and ns1.pchome.org. This is shown via the “A?” tcpdump lines. This shows fetch glue must be enabled.

Compliance with the Audit Control Objective

This audit item was to test that fetch glue was turned off to prevent cache poisoning. As can be seen from the results it does not appear that fetch glue is turned off.

A quick search in the /etc/named.conf file for fetch glue will show the configuration option:

Figure 21 – search for fetch-glue option in configuration file.

As can be seen no fetch-glue option was found in the configuration file. The default option if fetch-glue is not specified is to be enabled. As a result, this confirms our findings from the testing.

Audit Item 2 - The DNS server is backed up

As the DNS server is backed up on a Legato Networker server, we need to run the mminfo command on the server:

```
# mminfo -c dnsserver1.nsw.acme.research.gov.au
volume        client       date      size   level  name
NC0192  dnsserver1.nsw.acme.research.gov.au 03/21/03 237 MB full
/export/home
NC0192  dnsserver1.nsw.acme.research.gov.au 03/21/03 79 MB full /usr/local
NC0192  dnsserver1.nsw.acme.research.gov.au 03/21/03 922 MB full /var/log
NC0192  dnsserver1.nsw.acme.research.gov.au 03/21/03 490 MB full /var
NC0192  dnsserver1.nsw.a cme.research.gov.au 03/21/03 948  B full /tmp
NC0192  dnsserver1.nsw.acme.research.gov.au 03/21/03 958 MB full /
               . . .
               . . .
NI0576  dnsserver1.nsw.acme.research.gov.au 07/02/03 96 KB incr /export/home
NI0576  dnsserver1.ns w.acme.research.gov.au 07/02/03 61 MB incr /var
NI0576  dnsserver1.nsw.acme.research.gov.au 07/02/03 948  B incr /tmp
NI0576  dnsserver1.nsw.acme.research.gov.au 07/02/03 921 KB incr /
NI0576  dnsserver1.nsw.acme.research.gov.au 07/03/03 15 KB incr /usr/lo  cal
NI0577  dnsserver1.nsw.acme.research.gov.au 07/04/0 3 61 MB 5 /export/home
NI0577  dnsserver1.nsw.acme.research.gov.au 07/04/03 650 MB 5 /var/log
NI0577  dnsserver1.nsw.acme.research.gov.au 07/04/03 443 MB 5 /var
NI0577  dnsserver1.nsw.acme.research.gov .au 07/04/03 948  B 5 /tmp
NI0577  dnsserver1.nsw.acme.research.gov.au 07/04/0 3 1094 KB 5 /
NI0578  dnsserver1.nsw.acme.research.gov.au 07/04/03 151 KB 5 /usr/local
NI0584  dnsserver1.nsw.acme.research.gov.au 07/05/03 5232 KB incr
/export/home
NI0584  dnss erver1.nsw.acme.research.gov.au 07/05/03 981 KB incr /
NI0585  dnsserver1.nsw.acme.research.gov.au 07/05/03 15 KB incr /usr/local
NI0585  dnsserver1.nsw.acme.research.gov.au 07/05/03 391 MB incr /var/log
NI0585  dnsserver1.nsw.acme.research.gov.au 07/05/03  246 MB incr /var
NI0585  dnsserver1.nsw.acme.research.gov.au 07/05/03 948  B incr /tmp
#
```

From this extract of the information on the backup of the primary DNS server, we can see that the system is being backed up.

From the client configuration within legato we can check the backup retention period:

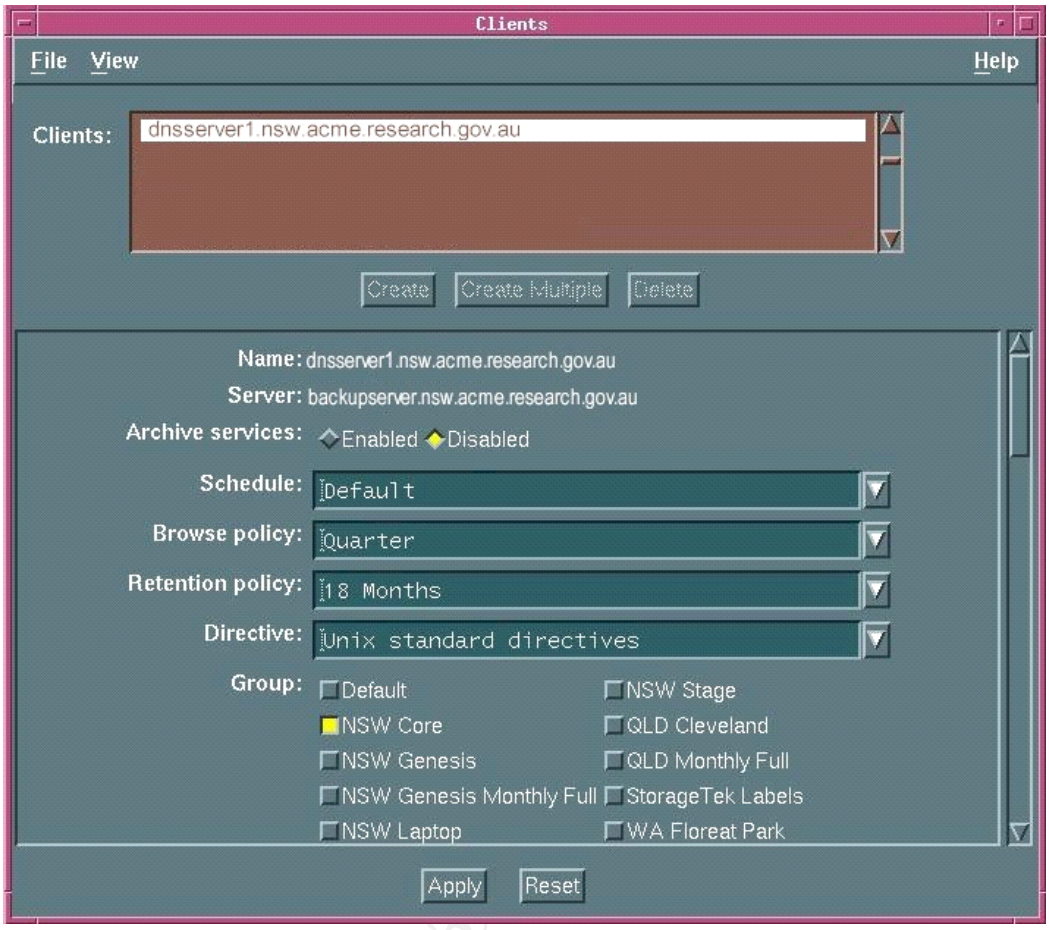

Figure 22 – Screenshot of backup retention schedule

To show that the backup does still have the files for more than 12 months, we need to run the mminfo command:

```
# mminfo -v -q 'level=full,sametime <= 52  weeks ago' -c
dnsserver1.nsw.acme.research.gov.au  -o t

volume        client         date     time       size ssid      fl   lvl
name
NC0366         dnsserver1.nsw.acme.research.gov.au 07/20/01 20:41:33 110 KB
1477116161 cE full /var/spool/mail
NC0364         dnsserver1.nsw.acme.research.gov.au 07/20/01 20:59:01 247 MB
1477380865 cE full /var/spool/ftp
NC0364         dnsserver1.nsw.acme.research.gov.au 07/20/01 21:35:07 114 MB
1477932801 cE full /usr/local
NC0364         dnsserver1.nsw.acme.research.gov.au  07/20/01 22:03:09 276 MB
1478362881 cE full /var/log
NC0364         dnsserver1.nsw.acme.research.gov.au 07/20/01 22:28:19 231 MB
1478757121 cE full /home
NC0364         dnsserver1.nsw.acme.research.gov.au 07/20/01 22:54:13 168 MB
1479147265 hE full /usr
NC0366         dnsserver1.nsw.acme.research.gov.au 07/20/01 22:54:13 43 MB
1479147265 tE full /usr
NC0364         dnsserver1.nsw.acme.research.gov.au 07/20/01 22:54:13 168 MB
1479147265 hE full /usr
NC0366         dnsserver1.nsw.acme.research.gov.au 07/20/0  1 22:54:13 43 MB
1479147265 tE full /usr
NC0364         dnsserver1.nsw.acme.research.gov.au 07/20/01 22:54:13 168 MB
1479147265 hE full /usr
```

```
NC0368         dnsserver1.nsw.acme.research.gov.au 07/20/01 23:14:55 107 MB
1479474433 cE full /var
NC0347         dnsserver1.nsw.acme.research.gov.au 07/20/01 23:26:21 9023 KB
1465725441 cE full /
NC0385         dnsserver1.nsw.acme.research.gov.au 08/17/01 21:25:02 112 KB
2097098497 cE full /var/spool/mail
NC0385         dnsserver1.nsw.acme.research.gov.au 08/17/01 21 :27:16 244 MB
2097125121 cE full /var/spool/ftp
NC0382         dnsserver1.nsw.acme.research.gov.au 08/17/01 22:01:08 114 MB
2097646337 cE full /usr/local
NC0382         dnsserver1.nsw.acme.research.gov.au 08/17/01 22:13:42 339 MB
2097842945 cE full /var/lo g
NC0382         dnsserver1.nsw.acme.research.gov.au 08/17/01 22:51:48 236 MB
2098432513 cE full /home
NC0382         dnsserver1.nsw.acme.research.gov.au 08/17/01 23:18:48 211 MB
2098844929 cE full /usr
NC0382         dnsserver1.nsw.acme.research.gov.au 08 /17/01 23:42:16 97 MB
2099248641 cE full /var
NC0382         dnsserver1.nsw.acme.research.gov.au 08/17/01 23:54:48 9024 KB
2099417857 cE full /
NC0406         dnsserver1.nsw.acme.research.gov.au 09/21/01 21:05:09 113 KB
2870930946 cE full /var/spool/mail
NC0407         dnsserver1.nsw.acme.research.gov.au 09/21/01 21:07:20 244 MB
2870966273 cE full /var/spool/ftp
                              . . . .
                              . . . .
```

As can be seen from the results, the backup has been kept for nearly two years. We know this because the entries will be removed when the tapes are re-written to.

Compliance with Audit Item Control Objectives

The results of the test show that the DNS server is backed up regularly and has a retention period of 18 months, and backups exist for even longer. As a result this audit item is compliant.

Audit Item 19 - DNS Data is consistent and up-to-date

Step 1: Using Dlint to check zone header information is correct – for acme.research.gov.au and nsw.acme.research.gov.au:

```
testmachine:/# dlint acme.research.gov.au
;; dlint version 1.4.0, Copyright (C) 1998 Paul A. Balyoz <pab@domtools.com>
;;     Dlint comes with ABSOLUTELY NO WARRANTY.
;;     This is free software, and you are welcome to redistribute it
;;     under certain co nditions.  Type 'man dlint' for details.
;; command line: /usr/bin/dlint acme.research.gov.au
;; flags: normal -domain recursive.
;; using dig version 9.2.2
;; run starting: Mon Jul  7 16:51:58 EST 2003
;; ================================================= =========
;; Now linting acme.research.gov.au
;; Checking serial numbers per nameserver
;;     2003051400 vicdns1.vic.acme.research.gov.au.
;;     2003051400 vicdns2.vic.acme.research.gov.au.
;;     2003051400 wadns1.wa.acme.research.gov.au.
;;     2003051 400 sadns1.sa.acme.research.gov.au.
;;     2003051400 actdns1.act.acme.research.gov.au.
;;     2003051400 dnsserver1.nsw.acme.research.gov.au.
;; All nameservers agree on the serial number.
;; Now caching whole zone (this could take a minute)
;; trying nam eserver sadns1.sa.acme.research.gov.au.
```

```
ERROR: no A records found.
;; no subzones found below acme.research.gov.au, so no recursion will take
place.
;; ============================================================
;; dlint of acme.research.gov.au run ending  with errors.
;; run ending: Mon Jul  7 16:51:59 EST 2003

testmachine:/# dlint nsw.acme.research.gov.au
;; dlint version 1.4.0, Copyright (C) 1998 Paul A. Balyoz <pab@domtools.com>
;;     Dlint comes with ABSOLUTELY NO WARRANTY.
;;     This is free softwa re, and you are welcome to redistribute it
;;     under certain conditions.  Type 'man dlint' for details.
;; command line: /usr/bin/dlint nsw.acme.research.gov.au
;; flags: normal -domain recursive.
;; using dig version 9.2.2
;; run starting: Mon Jul  7 16 :53:02 EST 2003
;; ============================================================
;; Now linting nsw.acme.research.gov.au
;; Checking serial numbers per nameserver
;;     2003070707 vicdns1.vic.acme.research.gov.au.
;;     2003070707 dnsserver1.nsw.acme.rese arch.gov.au.
;;     2003070707 actdns1.act.acme.research.gov.au.
;;     2003070707 wadns1.wa.acme.research.gov.au.
;;     2003070707 sadns1.sa.acme.research.gov.au.
;;     2003070707 vicdns2.vic.acme.research.gov.au.
;; All nameservers agree on the serial  number.
;; Now caching whole zone (this could take a minute)
;; trying nameserver sadns1.sa.acme.research.gov.au.
ERROR: no A records found.
;; no subzones found below nsw.acme.research.gov.au, so no recursion will
take place.
;; ========================== ==================================
;; dlint of nsw.acme.research.gov.au run ending with errors.
;; run ending: Mon Jul  7 16:53:03 EST 2003
```

Step 2: Using dnswalk to check DNS data information:

```
testmachine:/# dnswalk  -r acme.research.gov.au.
Checking acme .research.gov.au.
Getting zone transfer of acme.research.gov.au. from
dnsserver1.nsw.acme.research.gov.au...failed
FAIL: Zone transfer of acme.research.gov.au. from
dnsserver1.nsw.acme.research.gov.a u failed: truncated zone transfer
Getting zone transfer o f acme.research.gov.au. from
vicdns1.vic.acme.research.gov.au...failed
FAIL: Zone transfer of acme.research.gov.au. from
vicdns1.vic.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of acme.research.gov.au. from
vicdns2.vic.acme.r esearch.gov.au...failed
FAIL: Zone transfer of acme.research.gov.au. from
vicdns2.vic.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of acme.research.gov.au. from
wadns1.wa.acme.research.gov.au...failed
FAIL: Zone transfer of ac me.research.gov.au. from
wadns1.wa.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of acme.research.gov.au. from
sadns1.sa.acme.research.gov.au...failed
FAIL: Zone transfer of acme.research.gov.au. from
sadns1.sa.acme.research.go v.au failed: truncated zone transfer
Getting zone transfer of acme.research.gov.au. from
actdns1.act.acme.research.gov.au...done.
SOA=dnsserver1.nsw.acme.research.gov.AU
contact=hostmaster.nsw.acme.research.gov.AU
WARN: ils.acme.research.gov.AU CNAME il s.act.acme.research.gov.AU: CNAME (to
viper-bt.act.acme.research.gov.AU)
WARN: imp.acme.research.gov.AU CNAME imp.act.acme.research.gov.AU: CNAME (to
mira.act.acme.research.gov.AU)
```

```
WARN: experimental.acme.research.gov.AU CNAME
experimental.act.acme.researc h.gov.AU: CNAME (to
homan.act.acme.research.gov.AU)
WARN: sis.acme.research.gov.AU CNAME sis.act.acme.research.gov.AU: CNAME (to
uromys.act.acme.research.gov.AU)
WARN: obp.acme.research.gov.AU CNAME obp.act.acme.research.gov.AU: CNAME (to
mira.act.acme.res earch.gov.AU)
WARN: carlton.acme.research.gov.AU MX stranger.vic.acme.research.gov.AU:
unknown host
Checking act.acme.research.gov.au
Getting zone transfer of act.acme.research.gov.au from
actdns1.act.acme.research.gov.au...done.
SOA=actdns1.act.acme.resea rch.gov.AU
contact=hostmaster.mailhost.act.acme.research.gov.AU
WARN: nsr.act.acme.research.gov.AU CNAME nsrhost.act.acme.research.gov.AU:
CNAME (to mimosa.act.acme.research.gov.AU)
WARN: nym.act.acme.research.gov.AU A 181.52.96.254: no PTR record
WARN: libra.act.acme.research.gov.AU MX mail -bt.act.acme.research.gov.AU:
CNAME (to saab -bt.act.acme.research.gov.AU)
WARN: lemans -bt.act.acme.research.gov.AU A 181.52.88.2: no PTR record
WARN: crux.act.acme.research.gov.AU MX mail -bt.act.acme.research.gov.AU :
CNAME (to saab -bt.act.acme.research.gov.AU)
WARN: collab.act.acme.research.gov.AU A 181.52.89.100: no PTR record
WARN: polaris.act.acme.research.gov.AU A 181.52.88.15: no PTR record
WARN: coolum.act.acme.research.gov.AU MX traal.act.acme.research.gov.AU:
unknown host
WARN: oxford -bt.act.acme.research.gov.AU A 181.52.91.8: no PTR record
WARN: austin -bt.act.acme.research.gov.AU A 181.52.91.7: no PTR record
WARN: gw91.act.acme.research.gov.AU A 181.52.91.177: no PTR record
WARN: gw98.act.acme.research.gov.AU  A 181.52.98.254: no PTR record
WARN: gw100.act.acme.research.gov.AU A 181.52.100.254: no PTR record
WARN: benetton -bt.act.acme.research.gov.AU A 181.52.88.158: no PTR record
WARN: vauxhall -bt.act.acme.research.gov.AU A 181.52.91.4: no PTR record
WARN: pho enix.act.acme.research.gov.AU MX mail -bt.act.acme.research.gov.AU:
CNAME (to saab -bt.act.acme.research.gov.AU)
WARN: humber -bt.act.acme.research.gov.AU A 181.52.91.3: no PTR record
WARN: news.act.acme.research.gov.AU CNAME newshost.act.acme.research.gov.AU  :
CNAME (to perseus.act.acme.research.gov.AU)
WARN: aeon.act.acme.research.gov.AU CNAME alfalpha.act.acme.research.gov.AU:
unknown host
WARN: tekcolor.act.acme.research.gov.AU CNAME lp8.act.acme.research.gov.AU:
unknown host
Checking gforge.acme.research.g ov.au
Getting zone transfer of gforge.acme.research.gov.au from
gforge.gforge.acme.research.gov.au...failed
FAIL: Zone transfer of gforge.acme.research.gov.au from
gforge.gforge.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of  gforge.acme.research.gov.au from
dnsserver1.nsw.acme.research.gov.au...failed
FAIL: Zone transfer of gforge.acme.research.gov.au from
dnsserver1.nsw.acme.research.gov.a u failed: truncated zone transfer
BAD: All zone transfer attempts of gforge.acme.researc h.gov.au failed!
Checking gw.acme.research.gov.au
Getting zone transfer of gw.acme.research.gov.au from
dnsserver1.nsw.acme.research.gov.au...failed
FAIL: Zone transfer of gw.acme.research.gov.au from
dnsserver1.nsw.acme.research.gov.au failed: truncated z one transfer
Getting zone transfer of gw.acme.research.gov.au from
dnsserver2.nsw.acme.research.gov.au...failed
FAIL: Zone transfer of gw.acme.research.gov.au from
dnsserver2.nsw.acme.research.gov.a u failed: truncated zone transfer
BAD: All zone transfer a ttempts of gw.acme.research.gov.au failed!
Checking nsw.acme.research.gov.au
Getting zone transfer of nsw.acme.research.gov.au from
dnsserver1.nsw.acme.research.gov.au...failed
FAIL: Zone transfer of nsw.acme.research.gov.au from
dnsserver1.nsw.acme.resear ch.gov.au failed: truncated zone transfer
```

```
Getting zone transfer of nsw.acme.research.gov.au from
wadns1.wa.acme.research.gov.au...failed
FAIL: Zone transfer of nsw.acme.research.gov.au from
wadns1.wa.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of nsw.acme.research.gov.au from
sadns1.sa.acme.research.gov.au...failed
FAIL: Zone transfer of nsw.acme.research.gov.au from
sadns1.sa.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of nsw.acme.research.gov.a u from
vicdns2.vic.acme.research.gov.au...failed
FAIL: Zone transfer of nsw.acme.research.gov.au from
vicdns2.vic.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of nsw.acme.research.gov.au from
vicdns1.vic.acme.research.gov.au.. .failed
FAIL: Zone transfer of nsw.acme.research.gov.au from
vicdns1.vic.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of nsw.acme.research.gov.au from
actdns1.act.acme.research.gov.au...done.
SOA=dnsserver1.nsw.acme.research.g ov.AU
contact=hostmaster.acme.research.gov.AU
WARN: maroubra -nh.nsw.acme.research.gov.AU A 171.93.25.223: no PTR record
WARN: ntp.nsw.acme.research.gov.AU CNAME tictoc.tip.research.gov.au: unknown
host
WARN: spock.nsw.acme.research.gov.AU A 171.93.18.11 : no PTR record
WARN: adamstown -nh.nsw.acme.research.gov.AU A 171.93.26.38: no PTR record
WARN: grumpy.nsw.acme.research.gov.AU A 171.93.25.6: no PTR record
WARN: vintage.nsw.acme.research.gov.AU A 171.93.18.9: no PTR record
WARN: homes -bt.nsw.acme.researc h.gov.AU CNAME homes -
bt.act.acme.research.gov.AU: CNAME (to saab -bt.act.acme.research.gov.AU)
WARN: townhall -nh.nsw.acme.research.gov.AU A 171.93.25.2: no PTR record
WARN: hq -br.nsw.acme.research.gov.AU A 171.93.18.6: no PTR record
WARN: sax.nsw.acme.resea rch.gov.AU A 171.93.18.2: no PTR record
WARN: whip.nsw.acme.research.gov.AU A 171.93.18.3: no PTR record
WARN: hunterhills -nh.nsw.acme.research.gov.AU A 171.93.25.5: no PTR record
WARN: mail -bt.nsw.acme.research.gov.AU CNAME mail -
bt.act.acme.research.gov.A U: CNAME (to saab -bt.act.acme.research.gov.AU)
WARN: kirk.nsw.acme.research.gov.AU A 171.93.18.10: no PTR record
WARN: blacksmiths -nh.nsw.acme.research.gov.AU A 171.93.26.37: no PTR record
WARN: lisarow.nsw.acme.research.gov.AU A 171.93.23.44: no PTR recor d
WARN: files -bt.nsw.acme.research.gov.AU CNAME files -
bt.act.acme.research.gov.AU: CNAME (to saab -bt.act.acme.research.gov.AU)
WARN: dock83 -06-02.nsw.acme.research.gov.AU MX
nswmail.nsw.acme.research.gov.au.nsw.acme.research. gov.AU: unknown host
Checking q ld.acme.research.gov.au
Getting zone transfer of qld.acme.research.gov.au from
dnsserver1.nsw.acme.research.gov.au...failed
FAIL: Zone transfer of qld.acme.research.gov.au from
dnsserver1.nsw.acme.research.gov.a u failed: truncated zone transfer
Getting zon e transfer of qld.acme.research.gov.au from
actdns1.act.acme.research.gov.au...done.
SOA=dnsserver1.nsw.acme.research.gov.AU
contact=hostmaster.acme.research.gov.AU
WARN: bmh.qld.acme.research.gov.AU A 176.153.76.80: no PTR record
Checking sa.acme.resea rch.gov.au
Getting zone transfer of sa.acme.research.gov.au from
sadns1.sa.acme.research.gov.au...failed
FAIL: Zone transfer of sa.acme.research.gov.au from
sadns1.sa.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of sa.acme.res earch.gov.au from
vicdns1.vic.acme.research.gov.au...failed
FAIL: Zone transfer of sa.acme.research.gov.au from
vicdns1.vic.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of sa.acme.research.gov.au from
vicdns2.vic.acme.research .gov.au...failed
FAIL: Zone transfer of sa.acme.research.gov.au from
vicdns2.vic.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of sa.acme.research.gov.au from
wadns1.wa.acme.research.gov.au...failed
```

```
FAIL: Zone transfer of sa.ac me.research.gov.au from
wadns1.wa.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of sa.acme.research.gov.au from
dnsserver1.nsw.acme.research.gov.au...failed
FAIL: Zone transfer of sa.acme.research.gov.au from
dnsserver1.nsw.acm e.research.gov.au failed: truncated zone transfer
BAD: All zone transfer attempts of sa.acme.research.gov.au failed!
Checking vic.acme.research.gov.au
Getting zone transfer of vic.acme.research.gov.au from
vicdns1.vic.acme.research.gov.au...failed
FAIL: Zone transfer of vic.acme.research.gov.au from
vicdns1.vic.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of vic.acme.research.gov.au from
sadns1.sa.acme.research.gov.au...failed
FAIL: Zone transfer of vic.acme.research.gov.au fro m
sadns1.sa.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of vic.acme.research.gov.au from
wadns1.wa.acme.research.gov.au...failed
FAIL: Zone transfer of vic.acme.research.gov.au from
wadns1.wa.acme.research.gov.au failed: trun cated zone transfer
Getting zone transfer of vic.acme.research.gov.au from
vicdns2.vic.acme.research.gov.au...failed
FAIL: Zone transfer of vic.acme.research.gov.au from
vicdns2.vic.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer  of vic.acme.research.gov.au from
dnsserver1.nsw.acme.research.gov.au...failed
FAIL: Zone transfer of vic.acme.research.gov.au from
dnsserver1.nsw.acme.research.gov.a u failed: truncated zone transfer
BAD: All zone transfer attempts of vic.acme.research.gov .au failed!
Checking wa.acme.research.gov.au
Getting zone transfer of wa.acme.research.gov.au from
wadns1.wa.acme.research.gov.au...failed
FAIL: Zone transfer of wa.acme.research.gov.au from
wadns1.wa.acme.research.gov.au failed: truncated zone transfer
Getting zone transfer of wa.acme.research.gov.au from
actdns1.act.acme.research.gov.au...done.
SOA=wadns1.wa.acme.research.gov.AU
contact=hostmaster.wa.acme.research.gov.AU
WARN: nsrhost.wa.acme.research.gov.AU CNAME nsrmaster -
nsw.acme.research.gov.AU: CNA ME (to backupserver.nsw.acme.research.gov.au)
WARN: floreat -fs.wa.acme.research.gov.AU A 174.92.51.92: no PTR record
WARN: admin.wa.acme.research.gov.AU A 174.92.51.39: no PTR record
26 failures, 48 warnings, 4 errors.
testmachine:/#
```

This test was run on a machine which should not have zone transfer capabilities. Dnswalk showed that it could perform a zone transfer with the DNS server in ACT. This shows that there is a misconfiguration on this DNS server. Further tests found that zone transfers were restricted to internal clients. However other DNS servers within the acme.research.gov.au division only allow other DNS servers to perform zone transfers.

Now test on a DNS server which is authorised for zone transfers.

Test 3: dnswalk test on nsw.acme.research.gov.au

```
67 >./dnswalk nsw.acme.research.gov.au.
Checking nsw.acme.research.gov.au.
Getting zone transfer of nsw.acme.research.gov.au. from
dnsserver1.nsw.acme.research.gov.au...done.
SOA=dnsserver1.nsw.acme.research.gov.AU
contact=hostmaster.acme.res earch.gov.AU
WARN: maroubra -nh.nsw.acme.research.gov.AU A 171.93.25.223: no PTR record
```

```
WARN: ntp.nsw.acme.research.gov.AU CNAME tictoc.tip.research.gov.au: unknown
host
WARN: spock.nsw.acme.research.gov.AU A 171.93.18.11: no PTR record
WARN: adamstown-nh.nsw.acme.research.gov.AU A 171.93.26.38: no PTR record
WARN: grumpy.nsw.acme.research.gov.AU A 171.93.25.6: no PTR record
WARN: vintage.nsw.acme.research.gov.AU A 171.93.18.9: no PTR record
WARN: homes-bt.nsw.acme.research.gov.AU CNAME homes-
bt.act.acme.research.gov.AU: CNAME (to saab-bt.act.acme.research.gov.AU)
WARN: townhall-nh.nsw.acme.research.gov.AU A 171.93.25.2: no PTR record
WARN: hq-br.nsw.acme.research.gov.AU A 171.93.18.6: no PTR record
WARN: sax.nsw.acme.research.gov.AU A 171.93.18.2: no PTR rec ord
WARN: whip.nsw.acme.research.gov.AU A 171.93.18.3: no PTR record
WARN: hunterhills-nh.nsw.acme.research.gov.AU A 171.93.25.5: no PTR record
WARN: mail-bt.nsw.acme.research.gov.AU CNAME mail-
bt.act.acme.research.gov.AU: CNAME (to saab-bt.act.acme.research.gov.AU)
WARN: alibon-nh.nsw.acme.research.gov.AU A 171.93.25.120: no PTR record
WARN: sunshine.nsw.acme.research.gov.AU A 171.93.25.250: no PTR record
WARN: kirk.nsw.acme.research.gov.AU A 171.93.18.10: no PTR record
WARN: blacksmiths-nh.nsw.acme.research.gov.AU A 171.93.26.37: no PTR record
WARN: fatback.nsw.acme.research.gov.AU A 171.93.25.80: no PTR record
WARN: lisarow.nsw.acme.research.gov.AU A 171.93.23.44: no PTR record
WARN: files-bt.nsw.acme.research.gov.AU CNAME files-
bt.act.acme.research.gov.AU: CNAME (to saab-bt.act.acme.research.gov.AU)
WARN: dock83-06-02.nsw.acme.research.gov.AU MX
nswmail.nsw.acme.research.gov.au.nsw.acme.research. gov.AU: unknown host
0 failures, 21 warnings, 0 errors.
```

Test 4: dnswalk test on acme.research.gov.au

```
70 >./dnswalk acme.research.gov.au.
Checking acme.research.gov.au.
Getting zone transfer of acme.research.gov.au. from
dnsserver1.nsw.acme.research.gov.au...done.
SOA=dnsserver1.nsw.acme.research.gov.AU
contact=hostmaster.nsw.acme.research.gov.AU
WARN: ils.acme.research.gov.AU CNAME ils.act.acme.research.gov.AU: CNAME (to
viper-bt.act.acme.research.gov.AU)
WARN: imp.acme.research.gov.AU CNAME imp.act.acme.research.gov.AU: CNAME (to
mira.act.acme.research.gov.AU)
WARN: experimental.acme.research.gov.AU CNAME
experimental.act.acme.research.gov.AU: CNAME (to
homan.act.acme.research.gov.AU)
WARN: itsg2.acme.research.gov.AU A 181.52.31.40: no PTR record
WARN: intra.acme.research.gov.AU A 181.52.31.40: no PTR record
WARN: intra2.acme.research.gov.AU A 181.52.31.40: no  PTR record
WARN: itsg.acme.research.gov.AU A 181.52.31.40: no PTR record
WARN: sis.acme.research.gov.AU CNAME sis.act.acme.research.gov.AU: CNAME (to
uromys.act.acme.research.gov.AU)
WARN: obp.acme.research.gov.AU CNAME obp.act.acme.research.gov.AU: CNAME   (to
mira.act.acme.research.gov.AU)
WARN: carlton.acme.research.gov.AU MX stranger.vic.acme.research.gov.AU:
unknown host
0 failures, 10 warnings, 0 errors.
```

Test 5: Using DOC for nsw.acme.research.gov.au

```
56 >./doc nsw.acme.research.gov.au.
Doc-2.1.4: doc nsw.acme.research.gov.au.
Doc-2.1.4: Starting test of nsw.acme.research.gov.au.   parent is
acme.research.gov.au.
Doc-2.1.4: Test date  - Mon Jul  7 17:19:05 EST 2003
Summary:
   No errors or warnings issued for nsw.acme.research.gov.au.
Done testing nsw.acme.research.gov.au.  Mon Jul  7 17:19:08 EST 2003
```

Test 6: Using DOC for acme.research.gov.au

```
57 >./doc acme.research.gov.au.
Doc-2.1.4: doc acme.research.gov.au.
Doc-2.1.4: Starting test of acme.research.gov.au.   parent is
research.gov.au.
Doc-2.1.4: Test date - Mon Jul  7 17:19:19 EST 2003
;; res_nsend to server nxact1-yf.hq.research.gov.au.  181.52.231.1:
Connection timed out
DIGERR (UNKNOWN): dig @nxact1-yf.hq.research.gov.au. for SOA of parent
(research.gov.au.) failed
; Bad server: nxnsw1-pr.hq.research.gov.au.  -- using default server and
timer opts
; Bad server: nxnsw3-cj.hq.research.gov.au.  -- using default server and
timer opts
;; res nsend to server nxvic1-mb.hq.research.gov.au.  175.39.48.204:
Connection timed out
DIGERR (UNKNOWN): dig @nxvic1-mb.hq.research.gov.au. for SOA of parent
(research.gov.au.) failed
; Bad server: nxvic1-mh.hq.research.gov.au.  -- using default server and
timer opts
; Bad server: nxwa1-mr.hq.research.gov.au.  -- using default server and timer
opts
; Bad server: nxnsw1-pr.hq.research.gov.au.  -- using default server and
timer opts
; Bad server: nxnsw3-cj.hq.research.gov.au.  -- using default server and
timer opts
; Bad server: nxvic1-mh.hq.research.gov.au.  -- using default server and
timer opts
; Bad server: nxwa1-mr.hq.research.gov.au. -- using default server and timer
opts
Summary:
   WARNINGS issued for acme.research.gov.au. (count: 5)
Done testing acme.research.gov.au.  Mon Jul  7 17:19:52 EST 2003
```

As can be seen from the results, specifically tests 3 and 4, there are some errors in the DNS data.

Compliance with Audit Item Control Objective

The results indicate the data for the DNS server is inconsistent. The DNS server therefore does not comply with the audit requirements for this item.

Audit Item 14 - The Firewall or router filters traffic to the DNS server.

Step 1. Perform a tcp scan of the DNS server.

The firewall filters icmp echo responses, as a result we need to perform a scan without a ping being sent first. As a result the –P0 option for nmap is used.

```
(16:26)-54 [~] % nmap -P0 -sT 171.93.20.11

Starting nmap V. 2.53 by fyodor@insecure.org (  www.insecure.org/nmap/  )
WARNING:  your system apparently does not offer snprintf().  Reverting to
less secure version Unable to find nmap-services!  Resorting to
/etc/services Interesting ports on dnsserver1.nsw.acme.research.gov.AU
(171.93.20.11): (The 1030 ports scanned but not shown below are in state:
filtered)
Port       State       Service
53/tcp     open        domain
```

```
Nmap run completed -- 1 IP address (1 host up) scanned in 568 seconds
(16:37)-55 [~] %
```

From the above output it can be seen that only tcp 53 can be seen from an external host.

Step 2: Perform a udp scan of the DNS server.

Again as with step 1, as icmp echo replies are blocked by the firewall, we need to use the –P0 option for nmap.

```
% nmap -P0 -sU 171.93.20.11
Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Interesting ports on dnsserver1.nsw.acme.research.gov.AU (171.93.20.11):
(The 1016 ports scanned but n ot shown below are in state: filtered)
Port       State       Service
53/udp     open        domain
228/udp    open        unknown
374/udp    open        unknown
400/udp    open        unknown
550/udp    open        new -rwho
748/udp    open        unknown
806/udp    open        unknown
808/udp    open        unknown
834/udp    open        unknown
972/udp    open        unknown

Nmap run completed -- 1 IP address (1 host up) scanned in 5932 seconds
```

From the above output it appears that there are a lot of ports open on the DNS server. However the results of this test may be due to the firewall rules and the blocking of icmp error messages. As a result we need to do a scan on the host from an internal client to check what services are running on it. If there are ports matching the scan through the firewall which match the scan from an internal host then these will be the udp services accessible through the firewall.

A scan from an internal machine:

```
testmachine:/# nmap  -sU 171.93.20.11

Starting nmap 3.27 ( w ww.insecure.org/nmap/ ) at 2003 -07-09 15:59 EST
Interesting ports on dnsserver1.nsw.acme.research.gov.au (171.93.20.11):
(The 1459 ports scanned but not shown below are in state: closed)
Port       State       Service
53/udp     open        domain
111/udp    open        sunrpc
123/udp    open        ntp
514/udp    open        syslog

Nmap run completed -- 1 IP address (1 host up)  scanned in 156.766 seconds
testmachine:/#
```

From the results of the scan from an internal client and that of an external client, we can see that only udp port 53 match in both scans. As a result we can conclude that the firewall only allows port udp 53 and tcp 53.

Compliance with Audit Item Control Objective

The results of the scans show that the firewall meets with the audit control objectives and as a result is compliant.

Audit Item 13 - Authoritative Negative Cache turned off

Using nslookup:

Step 1: Start tcpdump

Step 2: Perform an nslookup of a non existent host

```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

O:\>nslookup
Default Server:  dnsserver2.nsw.acme.research.gov.au
Address:  171.93.20.14

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.au
Address:  171.93.20.11

> set debug
> noiphost.nepean.uws.edu .au.
Server:  dnsserver1.nsw.acme.research.gov.au
Address:  171.93.20.11

------------
Got answer:
    HEADER:
        opcode = QUERY, id = 3, rcode = NXDOMAIN
        header flags:  response, auth. answer, want recursion, recursion
avail.
        question s = 1,  answers = 0,  authority records = 1,  additional = 0

    QUESTIONS:
        noiphost.nepean.uws.edu.au, type = A, class = IN
    AUTHORITY RECORDS:
    ->  uws.EDU.AU
        ttl = 86400 (1 day)
        primary name server = ob1.uws.EDU.AU
        responsible mail addr = hostmaster.cooper.uws.EDU.AU
        serial  = 553
        refresh = 10800 (3 hours)
        retry   = 3600 (1 hour)
        expire  = 3600000 (41 days 16 hours)
        default TTL = 86400 (1 day)

------------
*** dnsserver1.nsw. acme.research.gov.au can't find
noiphost.nepean.uws.edu.au.: Non -existent domain
```

Step 3: Perform an nslookup for the same non-existent host.

```
> noiphost.nepean.uws.edu.au.
Server:  dnsserver1.nsw.acme.research.gov.AU
Address:  171.93.20.11

------------
Got answer:
    HEADER:
        opcode = QUERY, id = 4, rcode = NXDOMAIN
        header flags:  response, want recursion, recursion avail.
        questions = 1,  answers = 0,  authority records = 1,  additional = 0

    QUESTIONS:
        noiphost.nepean.u ws.edu.au, type = A, class = IN
    AUTHORITY RECORDS:
```

```
    ->  uws.edu.au
        ttl = 10790 (2 hours 59 mins 50 secs)
        primary name server = ob1.uws.edu.au
        responsible mail addr = hostmaster.cooper.uws.edu.au
        serial  = 553
        refresh = 10800 (3 hours)
        retry   = 3600 (1 hour)
        expire  = 3600000 (41 days 16 hours)
        default TTL = 86400 (1 day)

------------
*** dnsserver1.nsw.acme.research.gov.AU can't find
noiphost.nepean.uws.edu.au.: Non -existent domain
>
```

Step 4: Examine the tcpdump output

```
17:51:35.068069 171.93.20.92.1276 > 171.93.20.11.53:  3+ A?
noiphost.nepean.uws.edu.au. (44)
0x0000   4500 0048 8d6f 0000 8011 7ea2 ab5d 195c        E..H.o....~....  \
0x0010   ab5d 140b 04fc 0035 0034 64a4 0003 0100         .......5.4d.....
0x0020   0001 0000 0000 0000 086e 6f69 7068 6f73        .........noiphos
0x0030   7406 6e65 7065 616e 0375 7773 0365 6475        t.nepean.uws.edu
0x0040   0261 7500 0001 0001                            .au.....
17:51:35.071606 171.93.20. 11.47607 > 137.154.16.20.53:  43234 A?
noiphost.nepean.uws.edu.au. (44) (DF)
0x0000   4500 0048 70a3 4000 ff11 deb6 ab5d 140b        E..Hp.@.........
0x0010   899a 1014 b9f7 0035 0034 0a12 a8e2 0000        .......5.4......
0x0020   0001 0000 0000 0000 086e  6f69 7068 6f73        .........noiphos
0x0030   7406 6e65 7065 616e 0375 7773 0365 6475        t.nepean.uws.edu
0x0040   0261 7500 0001 0001                            .au.....
17:51:35.075942 171.93.16.55.1022 > 171.93.20.11.513: . ack 3015 win 8760
(DF)
0x0000   4500 0028 2a04 4000 ff06 2c5d ab5d 1437        E..(*.@...,]...7
0x0010   ab5d 140b 03fe 0201 0fd0 5442 e372 b505        ..........TB.r..
0x0020   5010 2238 65a4 0000 5555 5555 5555             P."8e...UUUUUU
17:51:35.076590 137.154.16.20.53 > 171 .93.20.11.47607:  43234 NXDomain*
0/1/0 (112) (DF)
0x0000   4500 008c 273d 4000 f811 2ed9 899a 1014        E...'=@.........
0x0010   ab5d 140b 0035 b9f7 0078 4a75 a8e2 8483        .....5...xJu....
0x0020   0001 0000 0001 0000 086e 6f69 7068 6f73        ... ......noiphos
0x0030   7406 6e65 7065 616e 0375 7773 0365 6475        t.nepean.uws.edu
0x0040   0261 7500 0001 0001 0375 7773 0345 4455        .au......uws.EDU
0x0050   0241 5500 0006 0001 0001 5180 002e 036f        .au.......Q....o
0x0060   6231 c02c 0a68  6f73 746d 6173 7465 7206        b1.,.hostmaster.
0x0070   636f 6f70 6572 c02c 0000 0229 0000 2a30        cooper.,...)..*0
0x0080   0000 0e10 0036 ee80 0001 5180                  .....6....Q.
17:51:35.079318 171.93.20.11.53 > 171.93.20.92.1276:  3 NXDomain  * 0/1/0
(112) (DF)
0x0000   4500 008c 93d6 4000 ff11 b8f6 ab5d 140b        E.....@.........
0x0010   ab5d 195c 0035 04fc 0078 a507 0003 8583        ... \.5...x......
0x0020   0001 0000 0001 0000 086e 6f69 7068 6f73        .........noiphos
0x0030   7406 6e65  7065 616e 0375 7773 0365 6475        t.nepean.uws.edu
0x0040   0261 7500 0001 0001 0375 7773 0345 4455        .au......uws.EDU
0x0050   0241 5500 0006 0001 0001 5180 002e 036f        .au.......Q....o
0x0060   6231 c02c 0a68 6f73 746d 6173 7465 7206         b1.,.hostmaster.
0x0070   636f 6f70 6572 c02c 0000 0229 0000 2a30        cooper.,...)..*0
0x0080   0000 0e10 0036 ee80 0001 5180                  .....6....Q.
                . . . . .
                . . . . .
                . . . . .

17:51:45.556048 171.93.20.92.1277 > 171.93.20.11.53:  4+ A?
noiphost.nepean.uws.edu.au. (44)
0x0000   4500 0048 8d9b 0000 8011 7e76 ab5d 195c        E..H......~v...  \
0x0010   ab5d 140b 04fd 0035 0034 64a2 0004 0100        .......5.4d.....
0x0020   0001 0000 0000 0000 08 6e 6f69 7068 6f73        .........noiphos
```

```
0x0030   7406 6e65 7065 616e 0375 7773 0365 6475        t.nepean.uws.edu
0x0040   0261 7500 0001 0001                            .au.....
17:51:45.570039 171.93.20.11.53 > 171.93.20.92.1277 :  4 NXDomain 0/1/0 (102)
(DF)
0x0000   4500 0082 93d7 4000 ff11 b8ff ab5d 140b        E.....@.........
0x0010   ab5d 195c 0035 04fd 006e 2a3c 0004 8183        ... \.5...n*<....
0x0020   0001 0000 0001 0000 086e 6f69 7068 6f73        .........noiphos
0x0030   7406 6e65 7065 616e 0375 7773 0365 6475        t.nepean.uws.edu
0x0040   0261 7500 0001 0001 c01c 0006 0001 0000        .au.............
0x0050   2a26 002e 036f 6231 c01c 0a68 6f73 746d        *&...ob1...hostm
0x0060   6173 7465 7206 636f 6f70 6572 c01c 0000        aster.cooper  ....
0x0070   0229 0000 2a30 0000 0e10 0036 ee80 0001        .)..*0.....6....
0x0080   5180                                           Q.
```

The first test (Test 1) shows the first lookup to noiphost.nepean.uws.edu.au. The second test (Test 2) shows the second lookup to noiphost.nepean.uws. edu.au. There is not much difference between the two tests, however you can see from the first test it is an authoritative answer. The second test is not. We assume this is because it is coming from the DNS cache. The third test (Test 3) shows the tcpdump output. As can be seen from this the first lookup goes to the Internet to get the details, the second lookup is answered directly via the DNS server. As the DNS server does not answer authoritatively we can see that authoritative negative caching is not enabled.

Using Dig - A better picture is given with the tool Dig as this shows the timing of queries:

Step 1: Start tcpdump

Step 2: Perform dig on non existent host

```
37 >dig @171.93.20.11 nohost.nepean.uws.edu.au.

; <<>> DiG 9.2.1 <<>> @171.93.20.11 nohost.nepean.uws.edu.au. A
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 51150
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;nohost.nepean.uws.edu.au.      IN      A

;; AUTHORITY SECTION:
uws.EDU.au.             86400   IN      SOA     ob1.uws.EDU.au.
hostmaster.cooper.uws.EDU.au. 553 10800 3600 3600000 86400

;; Query time: 23 msec
;; SERVER: 171.93.20.11#53(171.93.20.11)
;; WHEN: Mon Jul  7 17:57:42 2003
;; MSG SIZE  rcvd: 110
```

The output of the first dig command shows “aa” authoritative answer and the time to perform the query was 23 milliseconds.

Step 3: perform dig on same non existent host

```
38 >dig @171.93.20.11 nohost.ne pean.uws.edu.au.

; <<>> DiG 9.2.1 <<>> @171.93.20.11 nohost.nepean.uws.edu.au. A
;; global options:  printcmd
;; Got answer:
```

```
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 26424
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;nohost.nepean.uws.edu.au.      IN      A

;; AUTHORITY SECTION:
uws.edu.au.             10794   IN      SOA     ob1.uws.edu.au.
hostmaster.cooper.uws.edu.au. 553 10800 3600 3600000 86400

;; Query time: 2 msec
;; SERVER: 171.93.20.11# 53(171.93.20.11)
;; WHEN: Mon Jul  7 17:57:48 2003
;; MSG SIZE  rcvd: 100
```

From the output of the second query, we see that this response is not authoritative as there is no “aa” flag shown. If we note the time, it has taken 2 milliseconds to respond. Obviously this is coming from the cache of the DNS server.

Step 4: Examine the tcpdump output

```
17:54:13.539057 171.93.20.15.32771 > 171.93.20.11.53:  51150+ A?
nohost.nepean.uws.edu.au. (42) (DF)
0x0000   4500 0046 0000 4000 4011 0b66 ab5d 1a0a        E..F..@ .@..f....
0x0010   ab5d 140b 8003 0035 0032 9390 c7ce 0100        .......5.2......
0x0020   0001 0000 0000 0000 066e 6f68 6f73 7406        .........nohost.
0x0030   6e65 7065 616e 0375 7773 0365 6475 0261        nepean.uws.edu.a
0x0040   7500 0001 0001                                 u.....
17:54:13.556030 171.93.20.11.47607 > 137.154.16.20.53:  13753 A?
nohost.nepean.uws.edu.au. (42) (DF)
0x0000   4500 0046 70a4 4000 ff11 deb7 ab5d 140b        E..Fp.@.........
0x0010   899a 1014 b9f7 0035 0032 efa8 35b9 00 00        .......5.2..5...
0x0020   0001 0000 0000 0000 066e 6f68 6f73 7406        .........nohost.
0x0030   6e65 7065 616e 0375 7773 0365 6475 0261        nepean.uws.edu.a
0x0040   7500 0001 0001                                u.....
17:54:13.559615 137. 154.16.20.53 > 171.93.20.11.47607:  13753 NXDomain*
0/1/0 (110) (DF)
0x0000   4500 008a 924d 4000 f811 c3ca 899a 1014        E....M@.........
0x0010   ab5d 140b 0035 b9f7 0076 3010 35b9 8483        .....5...v0.5...
0x0020   0001 0000 0001 0000 066e 6f68 6f 73 7406        .........nohost.
0x0030   6e65 7065 616e 0375 7773 0365 6475 0261        nepean.uws.edu.a
0x0040   7500 0001 0001 0375 7773 0345 4455 0241        u......uws.EDU.A
0x0050   5500 0006 0001 0001 5180 002e 036f 6231        U.......Q....ob1
0x0060   c02a 0a68 6f73 746d 6173 7465 7206 636f        .*.hostmaster.co
0x0070   6f70 6572 c02a 0000 0229 0000 2a30 0000        oper.*...)..*0..
0x0080   0e10 0036 ee80 0001 5180                       ...6....Q.
17:54:13.561517 171.93.20.11.53 > 171.93.20.15.3 2771:  51150 NXDomain* 0/1/0
(110) (DF)
0x0000   4500 008a 9fed 4000 ff11 ac33 ab5d 140b        E.....@....3....
0x0010   ab5d 1a0a 0035 8003 0076 d3f7 c7ce 8583        .....5...v......
0x0020   0001 0000 0001 0000 066e 6f68 6f73 7406        .........nohos  t.
0x0030   6e65 7065 616e 0375 7773 0365 6475 0261        nepean.uws.edu.a
0x0040   7500 0001 0001 0375 7773 0345 4455 0241        u......uws.EDU.A
0x0050   5500 0006 0001 0001 5180 002e 036f 6231        U.......Q....ob1
0x0060   c02a 0a68 6f73 746d 6173  7465 7206 636f        .*.hostmaster.co
0x0070   6f70 6572 c02a 0000 0229 0000 2a30 0000        oper.*...)..*0..
0x0080   0e10 0036 ee80 0001 5180                       ...6....Q.
                . . . . .
                . . . . .
17:54:19.421303 171.93. 20.15.32771 > 171.93.20.11.53:  26424+ A?
nohost.nepean.uws.edu.au. (42) (DF)
0x0000   4500 0046 0000 4000 4011 0b66 ab5d 1a0a        E..F..@.@..f....
0x0010   ab5d 140b 8003 0035 0032 f426 6738 0100        .......5.2.&g8..
0x0020   0001 0000 0000 0000 066 e 6f68 6f73 7406        .........nohost.
0x0030   6e65 7065 616e 0375 7773 0365 6475 0261        nepean.uws.edu.a
```

```
0x0040   7500 0001 0001                                    u.....
17:54:19.422933 171.93.20.11.53 > 171.93.20.15.32771:  26424 NXDomain 0/1/0
(100) (DF)
0x0000   4500 0080 9fee 4000 ff11 ac3c ab5d 140b        E.....@....<....
0x0010   ab5d 1a0a 0035 8003 006c b9c2 6738 8183        .....5...l..g8..
0x0020   0001 0000 0001 0000 066e 6f68 6f73 7406        .........nohost.
0x0030   6e65 7065 616e 0375  7773 0365 6475 0261        nepean.uws.edu.a
0x0040   7500 0001 0001 c01a 0006 0001 0000 2a2a        u.............**
0x0050   002e 036f 6231 c01a 0a68 6f73 746d 6173        ...ob1...hostmas
0x0060   7465 7206 636f 6f70 6572 c01a 0000 0229        ter.cooper .....)
0x0070   0000 2a30 0000 0e10 0036 ee80 0001 5180        ..*0.....6....Q.
```

Again the output from the tcpdump shows that in the first query the DNS server has performed a recursive query asking the name server for nepean.uws.edu.au for the host “nohost”. The name server responds to the DNS with a negative answer (NXDOMAIN) saying that the host does not exist. In the second query the DNS server responds directly as the information is cached – thus the quick response time in the second query with Dig.

Compliance with the Audit Item Control Objective

As can be seen from the tests, the DNS server does not perform negative authoritative caching, which is compliant with the audit control objective.

Measure Residual Risk

From the checklist and audit results, we can see that there are a number of items which were not compliant, some which were partially compliant and other which met the expectations of the tests.

Non-Compliant Audit Items

BIND Version number obfuscation

The results of this test showed that the BIND version number was not hidden from people on the Internet.

The consequences of people obtaining this information is low, in terms of directly effecting the operation of the DNS server or network. As a result the residual risk associated with not obfuscating the BIND version number is low.

Exposure => low – controls => none = residual risk => low.

Recommendations – BIND version be changed in configuration file.

Fetch Glue Disabled

Allowing Fetch Glue to be enabled means that the DNS server is open to DNS cache poisoning, as a result the exposure of the system is high. However, as the test for logging unauthorised queries showed, the DNS queries to domains other than the DNS servers own domain are restricted, as a result this mitigates much of the risk of having fetch glue turned on.

This however does not mean the server is free from cache poisoning, as if an internal client makes a query, the DNS server will try to recursively solve it, and in the process may make a fetch glue query.

The residual risk of not disabling fetch glue is low to medium, as cache poisoning is not ruled out via the DNS servers configuration, however it is drastically reduced via restricting recursive queries to internal hosts and disabling queries for any other domain except the DNS's own domain for external hosts.

exposure => high – controls => medium = residual risk => low/medium.

Recommendations – turn off fetch_glue. This may mean that some DNS requests will not get answered, however it reduces the risk of cache poisoning.

Run as non-root

While BIND version 8.2.7 is considered a safe BIND release as it has fixes for all the previous root exploits found in other earlier releases, it does not mean vulnerabilities will not be found in the future. The consequences of running BIND as root mean that an attacker if a vulnerability is found in BIND may gain root privileges. As a result of this, the exposure of the system is high.

The residual risk associated with not running BIND as a non-root user as also high. This is due to the fact that there are no controls in place to mitigate a root compromise if a BIND vulnerability is found.

Exposure => high – controls => none = residual risk => high

Recommendations. There are two recommendations or controls that can be put in place to reduce this level of risk. The first is to run BIND as a non-root user. This would involve changing the ownership of the files used by BIND, but should not cause any inconvenience or break other software. The second alternative is to run BIND in a chrooted environment. In this way the root process is jailed into a set of directories. If a root exploit is found, then an attacker would only have access to the BIND directories and nothing else on the system. This second recommendation would also involve changing file system locations, but should not effect the operation of the DNS server or break other software. There are several documents and books which show how to configure BIND for a chrooted environment (Wunsch).

At a minimum, running BIND as a non-root user should be undertaken.

Zone Transfers Authenticated

The audit test of the DNS zone transfer shows that transfers are not authenticated with transaction signatures. However as the DNS server is configured to use a NOTIFY message. This is used to tell slaves that a zone transfer should take place, the slaves must then send a corresponding

NOTIFY message to the DNS server in order to update the information in a zone file. As a result there is some level of zone transfer authentication. If polling was used instead of a notify message, then there would be even less authentication.

General zone transfers via dig or nslookup don't update the slave DNS, but rather are just used to obtain zone information.

While having notify messages provides some level of authentication, the DNS servers are still susceptible to spoofing, and as a result there is still some risk.

The consequences of a zone transfer being spoofed are high. There is some control being provided by the use of notify messages between a DNS master and slave to provide some level of authentication, but this is not totally secure. As a result the use of notify messages to provide some control over mitigating the risk of a zone transfer being spoofed is low to medium.

With notify messages in place, the residual risk is medium.

Exposure => high – control => low = residual risk => medium.

Recommendations – the use of transaction signatures when performing zone transfers is highly recommended to provide a higher level of authentication.

Limit queries of non-public information to trusted hosts

As a result of all internal hosts having public IP addresses, it was possible to perform a reverse lookup query on each host individually. As a result information about the internal hosts on the network can be found.
This exposure is low/medium no harm as a direct result of this can occur. However the information can be used to map the internal network by an attacker.

The residual risk is medium as there are no controls in place to restrict this information. The residual risk is that an external party has map of network. However there are controls in place to stop attacks, such as a the firewall access lists and machines patches, as a result the consequences are limited.

Exposure => medium - control => none = residual risk => medium

Recommendations – the internal network be segregated from the public DNS server. This could be done via a split DNS setup.

Estimated costs in doing a split DNS.

Cost of second DNS server, cost of modifying the network architecture, an upgrade of the router for another interface, cost of modifying the router access lists. Also cost in moving Web services and mail services to external DMZ. The cost of performing a split DNS and modifying the network to suit it is not

worth the benefit it provides. The business needs of the organisation also mean that some of the research machines must be open to provide services.

Residual risk is that an external party has map of network. However if the firewall access lists are secure and machines are patched, then the consequences are limited.

DNS Data is consistent and up-to-date.

The results of the audit to check whether the DNS data is correct, show that the information maintained by the primary DNS server being audited and slave DNS server at the same site is inconsistent. There were some anomalies found in the data and configuration of the other DNS servers after running the consistency checking tools.

The consistency check of the data showed that there were pointers to machine names that did not have IP addresses associated with them. The consistency check of the data also showed that there was still an old MX record for a mail server that no longer existed in the Victorian DNS server (vicdns1). Running the checking tools on a host other than a DNS server also showed that the DNS server in Canberra (actdns1) did not restrict zone transfers to specific hosts, but allowed any internal host to perform a zone transfer. However as these DNS servers are out of the control of the administrators in the NSW site, and the scope of the audit was the primary DNS server, this audit item is compliant. However as the primary DNS server does answer for the other zones, this inconsistency should be noted.

The exposure of inconsistent data and data which is not up to date is medium to high if an attacker can spoof old IP addresses of machines which no longer exist, but still remain the DNS. However there are controls in place to prevent spoofing such as the firewall anti-spoofing rules, however these only stop spoofing at a local site, not the IP addresses of a remote site. By providing logging which should pick up some of this information, a control is in place to reduce the risk, however the control logging provides is low in this case. As a result the residual risk is medium for DNS Data Consistency and being up to date.

Exposure => high – control => low = residual risk => medium

Recommendations – In regard to the current DNS data inconsistencies. It is recommended that the DNS records be fixed. For the Victorian and Canberra DNS servers, the old MX records and zone transfer problems should be fixed – this information should be sent to the administrators in those regions.
It is also recommended that a better alerting process be put in place. This would involve defining better procedures for monitoring inconsistent data and should involve the automation of the process - via the DNS data checking tools available, and an alerting mechanism such as email to administrators of the DNS servers.

Known Vulnerability – Birthday Attack

The results of the audit showed that the DNS server was vulnerable to this attack. The exposure of this vulnerability is high. However there are controls to mitigate the likelihood of this attack. As queries to external hosts from external hosts are denied, the attack cannot be performed externally. The only way an a Birthday Attack can occur is from an internal host, which means the DNS does recursion and will query for external hosts.

The residual risk is low as there are mitigating controls in place.

Exposure => high – control => high = residual risk => low.

Recommendations – As the birthday attack can only occur from within the internal network itself, it is recommended that a procedure be put in place to monitor internal clients for suspicious activity and compromise.

Partially Compliant Items

Statistics Enabled

The results of the audit showed that DNS server is compliant with the audit control objective – that the BIND server was capable of providing statistical information, and statistics were regularly dumped. However full statistics were not enabled, meaning that some information such as host query statistics were not available.

A point to note however is that while statistical information was being dumped, there were no processes in place to use the information, apart from having it as a reference in the log files. Not having any monitoring of the information is just as useful as not having statistics turned on – that being said, it is better to have the information than not have it just as a reference in the case where the system resources may go up – such as in the case of a denial of service attack. The information can also be used to see whether the DNS server requires more resources.

The consequences of not having this service enabled is low. Having statistics enabled does not provide any more security directly to the system. It does however provide an auditor or administrator the ability to make a baseline of a DNS server in terms of standard operating traffic patterns, and can provide some useful information in terms of supporting log file findings if full statistics are enabled.

The residual risk of not enabling statistics is low to none, as there is no risk in not having statistics turned off, but there is a benefit if it is turned on.

exposure => low – controls => none = residual risk => low.

Recommendations – a process for monitoring the statistics should be put in place. Full statistics, including host statistics be recorded.

Compliant Items

Limit zone transfers that can occur at any one time

The residual risk associated with the number of zone transfer that can occur at any one time is low. The tests performed showed that the DNS server was quite capable of still answering queries relatively well. A performance hit on the server was noticeable in some cases. The difference in performance was about 10 to 11 times slower than when the transfers were not taking place. However this was still quick, instead of taking around 400 ms the query took around 5000 ms to complete. By controlling risk by restricting zone transfers the exposure to a zone transfer flood is reduced.

Exposure => high – control => high = residual risk => low.

Ensure root server information updated regularly

There were controls in place from the audit results to ensure that the root servers were updated regularly.

The residual risk associated with this is low as the root servers do not change very often, but could be a change in-between the regular updates.

Exposure => low/medium – control => medium = residual risk => low.

NSCD cache disabled for host lookup

The results of the audit test for examining whether NSCD cache was disabled for DNS were compliant with the Control Objective.

As a result of being compliant, there is no residual risk for the problems associated with the NSCD cache being enabled for host lookups. The exposure with NSCD cache being enabled is medium, and the control in place of the NSCD cache being disabled is high.

Exposure => medium – control => high = residual risk => none

Firewall filters traffic to DNS Server

The results of the tests show that filters were in place restricting traffic to just port 53 tcp and udp to the DNS server.
The residual risk associated with allowing access to any listening ports to the server is low, as filters are in place. However the risk is not none due to the fact that DNS may be exploited, however blocking all the ports would defeat the purpose of the DNS server as it is a business requirement that the ports be open for external queries.

Exposure => high – control => high = residual risk => low/none

Backup of Data and the System

There is no residual risk associated with the backup, as backups are in place and performed regularly.

Exposure => medium/high – control => high = residual risk => none

Dynamic updates restricted

There is no residual risk as dynamic updates are disabled.

Exposure => high – control => high = residual risk => none

Authoritative negative caching disabled

There is no residual risk associated with Authoritative negative caching in terms of security as it is disabled.

Exposure => high – control => high = residual risk => none

Vulnerability alerts monitored and patch procedures in place

There are no residual risks associated with this as alerting and patch procedures are in place, and are understood by staff who have been trained. There were change control documents to show that patching actually occurred.

Incident handling and response procedures in place

There are no residual risks associated with this as incident handling and response procedures are in place, and are understood by staff who have been trained.

Restrict HINFO and TXT usage on public DNS servers

There is no residual risk associated with HINFO and TXT queries to the DNS server as this information is not defined in the DNS data for hosts.

Exposure => medium/low – control => high = residual risk => none

Recursion is turned off or restricted

The audit results show that recursion was restricted to internal clients, and all external queries were not recursive, or were denied. The means that for cache poisoning to occur a query from an internal host is required. As a result cache poisoning cannot be ruled out totally. In terms of the DNS server however it has been configured securely to meet the business needs, as a result the residual risk is low/medium. Monitoring procedures of internal clients are another area for examination.

Exposure => high – control => high = residual risk => low.

Logging is turned on for BIND

There is no residual risk with logging. It was found from the audit that logging was in place to log the most important areas of DNS security.

Residual risk => medium/low – control => high = residual risk => none

DNS ID's randomised

The residual risk associated with DNS ID randomisation is low. Even with the vulnerability of the Birthday Attack, it relies on the attack being performed from an internal client, as it relies on recursion being used to poison the DNS servers cache. The monitoring of the network and the monitoring of clients is not part of the role of the DNS server. As recursion is disabled for external hosts, there is a high degree of control in place to mitigate an attack.

Exposure => high – control => high = residual risk => low

Recommendations – It is recommended that network monitoring be put in place to check internal hosts compromises.

Is the System Auditable ?

The BIND DNS server can be audited as can be seen from the audit results. There were however a few aspects where validation was not possible in terms of representation of actual objective tests, as these test were by inquiry.

Apart from the backup and disaster recovery procedures which can be objectively tested, other procedural aspects such as incident handling and response could not be properly validated.

The limiting of zone transfers has some objective tests. However determining the actual number of zone transfers which the DNS server is limited to is hard to perform. By looking at the aspect of performance instead of the actual number of zone transfers at one time, a determination relating to the audit test can be found.

It was also not possible to verify specifically recursion being restricted for external hosts, due to the fact that queries were restricted for external hosts. This basically meant queries where denied for any external host querying a host not in the domain zones owned by the DNS server. By restricting queries however, recursion is also restricted.

Assignment 4 – Audit Report or Risk Assessment.

Summary

The results of the audit indicate that the DNS server 'dnsserver1.nsw.acme.research.gov.au' has met the compliance criteria for most of the audit checklist items. However of those which were not compliant, there were several which posed a medium to high level of risk to the organisation. These items need to be addressed to ensure the DNS is secure.

The DNS server allows information about itself and the network to be obtained. While this is not a threat in itself, it can lead to attacks from outside.

A major concern was the lack of authentication between hosts that exchange DNS data.

Another concern was the small threat of the possibility of cache poisoning that existed, which was increased by the lack of disabling a feature known as "fetch glue" which does a query for the IP address of a name server in a search list, whose IP is not specified.

There was also the fact that the DNS application was run with root privileges, which is a concern if a BIND exploit is found in the future.

It was also found that the root DNS server information was not updated regularly.

Lastly the system was prone to a DNS cache poisoning attack from internal hosts, either via a recursive query to a malicious DNS server, or an internal host performing a Birthday Attack. Basically the internal network is the main area where any attacks to the DNS server via cache poisoning will successfully occur.

Background

There were several items which were non-compliant and partially compliant from the results of the audit on the DNS server running BIND 8.2.7. Of the checklist items the following were non compliant:

Non Compliant Items

Audit Item 15 – Fetch Glue Disabled

By allowing fetch glue to be enabled the likelihood of DNS cache poisoning is increased. If a malicious DNS server returns a name server but no ip address for that name server, the DNS server will do a query for that IP address, and then make a query to the name server. This information may be then cached. Unlike cache poisoning of the IP address of a general host, this information is the IP address of a name server for a domain. As a result, future queries to a domain may be referred to this name server address. The risk to the

organisation is that this form of cache poisoning may lead to the DNS hijacking of an external domain or a man in the middle attack.

For the acme division itself, there is not much harm in relation to accessing information about it's own hosts, as the DNS server contains the zone information for all of acme as well as the top level zone of the organisation, so the DNS server does not need to perform a recursive query for the acme.research.gov.au domain. As the DNS server has the top level domain information, it would also have a list of name servers for other divisions within the organisation. The risk associated with fetch glue is that internal hosts may be redirected to external malicious sites, or to non existent servers. As a result user information may be captured or a denial of service in attempting to access external information may occur.

External hosts cannot exploit this vulnerability as queries for other domains are denied. Only queries for zone information which the DNS manages are possible from external hosts, but as zone information is contained locally, no fetching of data from other servers is required.

This exploit can only occur if an internal host performs a query which causes the DNS server to contact a misconfigured or malicious DNS server which returns invalid data.

Audit Item 10 – BIND run as non-root

By allowing the DNS server to run as root, the BIND application effectively has root permissions. If a software exploit for BIND is found in the future for BIND 8.2.7 that allows an attacker to remotely run an application or shell, then they will have root access to the machine. The consequences of an attacker gaining root access is severe. They would have full control of the DNS server, and would not need to perform cache poisoning attacks to corrupt the data. The attacker could simply modify the DNS files. Root access would also mean an attacker could install a network sniffer and sniff traffic and possibly usernames and passwords to gain access to other parts of the network.

The risk to the organisation is that root access may allow an attacker to use the organisations computers to form an attack against another external organisation, and could jeopardise the reputation of the organisation.

Audit Item 7 – Zone Transfers Authenticated

While there is a form of authentication provided via the use of notify messages between a master DNS server and slave, there is no accurate way of authenticating the two hosts involved in a zone transfer if there is no authentication enabled. As a result there is the possibility of a master DNS server for which our DNS server 'dnsserver1' is a slave, such as the corporate DNS server or regional DNS servers in other states, to be spoofed. Without authentication the DNS server may receive malicious information for a zone. In terms of the corporate DNS zone, for example, malicious name server addresses may be given for the research.gov.au domain and other sub

domains corresponding to the other divisions within the organisation. This may be used for a man in the middle attack or DNS hijacking of the domain. The result of DNS hijacking for example may allow an attacker to masquerade there machine as a valid organisational host and gain access to internal services which are authenticated via reverse lookups for a host.
To the organisation this may mean that an attacker could gain access to confidential information in a form of corporate espionage and cause a loss of reputation and revenue.

Audit Item 4 – BIND Version obfuscated

Reporting the version of BIND running to people, would allow an attacker to use this information to do research on the vulnerabilities associated with the version of BIND and plan an attack. If there were known vulnerabilities, an attacker could use these vulnerabilities to exploit the server. For instance in the case of a root exploit and attacker could gain root access to the system if BIND was running as root. Similarly, if there are software exploits which allow an attacker to cause the DNS server to crash, and therefore cause a denial of service attack. The later would have little effect on the acme division as the DNS servers allow for redundancy, although it could used in conjunction with other types of attack, such as domain hijacking.

While in itself the leakage of this information is not harmful to the DNS server, the use that the information gets put to can be.

Audit Item 21 – Known Vulnerabilities – Birthday Attack

Some degree of risk needs to be accepted for this vulnerability. It effects all versions of BIND and there is no fix as yet for it. The only way to reduce or remove the risk is to remove caching. However this is not feasible as it would mean removing recursion from the DNS server for internal clients. As internal clients are fairly secure this risk is acceptable. The results of this attack lead to DNS cache poisoning which can be used for a denial of service attack, DNS hijacking or man in the middle attack, the last two being is the main one which would effect the organisation, if users were redirected to malicious hosts.

Audit Item 9 – Internal information not available to public

By allowing information about the hosts on the network to be accessed externally, an attacker can create a map of the internal network. If the information such as HINFO or TXT is available about the hosts, then this information can be used to pinpoint a particular host to attack. In our case HINFO and TXT information is not used as was found by the audit to be restricted (Audit Item 18)

As with the BIND version number, this information by itself is not harmful to the DNS server or organisation, but what it gets used for can be.

Audit Item 19 – DNS data consistent

The audit tests showed that the data for the NSW DNS zones had some inconsistencies. For the Sydney (NSW) region, there were several hosts entries with A records, but no reverse lookup PTR records. In examining the list, these are for machines which no longer exist. The other zone data that the DNS server acts a slave for was inconsistent. For the Melbourne (VIC) DNS server there was an error in the data maintained, and there was a configuration problem on the Canberra (ACT) DNS server that showed up when performing the consistency checking which allowed zone transfers from any internal host.

Inconsistent data, such as leaving the MX record for an old mail server which no longer exists could allow an attacker to use this to intercept mail for the organisation. In this case the MX record pointed to a DNS name not an IP address as a result if the attacker could poison a DNS cache on external machine, mail from an external site could be pointed to their host.

Allowing zone transfers to any host internally means that if an internal machine is compromised, an attacker could obtain information on all of the acme division as zone data for all regions is kept on all DNS servers.

Partially Compliant Items

The following item is classed as partially compliant. While the tests showed it was compliant, there are improvements to make it compliant above a minimum standard.

Audit Item 3 – Statistics Enabled

While the audit tests showed that this item was compliant, in terms of having statistics enabled and dumped at regular intervals, full statistics recording was not enabled. While this was not a requirement for compliance, it is useful to enable.

The current statistics recording level is good as it provides resource usage and the level of activity of the DNS server. However it does not provide statistics on hosts that connect and perform queries or zone transfers.

There is no risk to the organisation even if statistics was not enabled at all, however full statistics provides an audit trail and a mechanism for monitoring hosts connecting to the DNS server.

System Changes and Further Testing

From the non-compliant items there are several which can be rectified. There are also some items which cannot. Those items which can be rectified are as follows:

Audit Item 15 – Fetch Glue Disabled

It is recommended that fetch glue be disabled on the DNS server. This can easily be done by adding the following to the options section of the BIND configuration file /etc/named.conf (assuming default file location).

```
option {
            fetch-glue no;
        };
```

The DNS server daemon needs to be restarted, or prompted to reread the configuration file. This can be done in either of two ways:

To restart the daemon:

```
# ndc restart
```

Or a simpler method is to tell the DNS server to re-read the configuration file:

```
# kill -HUP `cat /etc/named.pid`
```

To retest the system, we will perform the test on the sub domain running on our test machine.

Step 1: Start tcpdump

Step 2: Query the domain domain.acme.research.gov.au

```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

O:\>nslookup
Default Server:  dnsserver2.nsw.acme.research.gov.au
Address:  171.93.20.14

> server 171.93.20.11
Default Server:  dnsserver1.nsw.acme.research.gov.au
Address:  171.93.20.11

> set debug
> domain.test.acme.research.gov.au.
Server:  dnsserver1.nsw.acme.research.gov.au
Address:  171.93.20.11

------------
Got answer:
    HEADER:
        opcode = QUERY, id = 3, rcode = NOERROR
        header flags:  response,  auth. answer, want recursion, recursion
avail.
        questions = 1,  answers = 1,  authority records = 4,  additional = 2
```

```
    QUESTIONS:
        domain.test.acme.research.gov.au., type = A, class = IN
    ANSWERS:
    ->  domain.test.acme.research.gov. au
        Internet address = 171.93.20.17
        ttl = 3600 (1 hour)
    AUTHORITY RECORDS:
    ->  test.acme.research.gov.au
        nameserver = ns1.pchome.org
        ttl = 3600 (1 hour)
    ->  test.acme.research.gov.au
        nameserver = ns1.peter host.ru
        ttl = 3600 (1 hour)
    ->  test.acme.research.gov.au
        nameserver = dnsserver1.nsw.acme.research.gov.au
        ttl = 3600 (1 hour)
    ->  test.acme.research.gov.au
        nameserver = ns1.test.acme.research.gov.au
        ttl = 36 00 (1 hour)
    ADDITIONAL RECORDS:
    ->  dnsserver1.nsw.acme.research.gov.au
        Internet address = 171.93.20.11
        ttl = 86400 (1 day)
    ->  ns1.test.acme.research.gov.au
        Internet address = 171.93.20.17
        ttl = 3600 (1 hour)

------------
Name:    domain.test.acme.research.gov.au
Address:  171.93.20.17

>
```

Step 3: Examine the tcpdump output

```
7:30:00.079800 171.93.20.92.2371 > 171.93.20.11.53:  3+ A?
domain.test.acme.research.gov.au. (45)
0x0000   4500 0049 8643 0000 8011 85cd ab 5d 195c        E..I.C..... \
0x0010   ab5d 140b 0943 0035 0035 bec5 0003 0100        .....C.5.5......
0x0020   0001 0000 0000 0000 0664 6f6d 6169 6e06        .........domain.
0x0030   7465 7374 6060 0461 636d 6508 7265 7365        test.acme.resear
0x0040   7202 6175 6175 6175 6175 0000 0100 01          ch.gov.au......
17:30:00.080350 171.93.20.11.53 > 171.93.20.92.2371:  3* 1/4/2 A
171.93.20.17 (197) (DF)
0x0000   4500 00e1 5aad 4000 ff11 f1ca ab5d 140b        E...Z.@.........
0x0010   ab5d 195c 0035 094 3 00cd 7499 0003 8580        ... \.5.C..t.....
0x0020   0001 0001 0004 0002 0664 6f6d 6169 6e06        .........domain.
0x0030   7465 7374 6060 0461 636d 6508 7265 7365        test.acme.resear
0x0040   7202 6175 0000 0100 01c0 0c00 0100 0100        ch.gov.a u.......
0x0050   000e 1000 0482 9b10 0bc0 1300 0200 0100        ................
0x0060   000e 1000 1003 6e73 3106 7063 686f 6d65        ......ns1.pchome
0x0070   036f 7267 00c0 1300 0200 0100 000e 1000        .org............
0x0080   1203 6e73 3109 7065  7465 7268 6f73 7402        ..ns1.peterhost.
0x0090   7275 00c0 1300 0200 0100 000e 1000 0d06        ru..............
0x00a0   646e 7373 6572 036e 7377 c01a c013 0002        dnsserver1.nsw..
0x00b0   0001 0000 0e10 0009 0674 6573 7460 60c0        ......... test...
0x00c0   13c0 8300 0100 0100 0151 8000 0482 9b10        .........Q......
0x00d0   01c0 9c00 0100 0100 000e 1000 0482 9b10        ................
0x00e0   0b                                             .
```

As can be seen from the results, the DNS server is not trying to look up the IP information for the servers ns1.pchome.org or ns1.peterhost.ru. As a result it returns 4 name server names, but only 2 IP addresses, which relate to the local DNS servers. The DNS server is now compliant with this audit item.

Audit item 4 – BIND version obfuscated

To hide the BIND version number from external queries we can modify the configuration file /etc/named.conf for BIND and add to the options section:

```
option {
            version "unknown";
        };
```

To get BIND to re-read the configuration file, we can send a HUP signal to the process:

```
# kill -HUP `cat /etc/named.pid`
```

To retest the version number we run the following command:

```
# dig @dnsserver1.acme.research.gov.au version.bind txt chaos
```

The output of this command is as follows:

The results show that the BIND version number is now hidden or obfuscated from external queries.

```
testmachine:~$ dig @dnsserver1.nsw.acme.research.gov.au version.bind txt
chaos

; <<>> DiG 9.2.2 <<>> @ dnsserver1.nsw.acme.research.gov.au ve  rsion.bind txt
chaos
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52658
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;version.bind.                  CH       TXT

;; ANSWER SECTION:
VERSION.BIND.           0       CH      TXT     "unknown"

;; Query time: 1 msec
;; SERVER: 171.93.20.11#53(dnsserver1.nsw.acme.research.gov.au)
;; WHEN: Fri Jul 11 15:02:31 2003
;; MSG SIZE  rcvd: 62
```

As can be seen from the results, BIND now returns the value of "unknown" and so the DNS server is now compliant with this audit item.

Audit Item 10 – BIND as non-root

Modify the /etc/init.d/inetsvc start-up file, add the –u option to specify user to start named as:

```
#
# If this machine is configured to be an Internet Domain Name System (DNS)
# server, run the name daemon.  Start named prior to: route add net host,
```

```
# to avoid dns gethostbyname timout delay for nameserver during boot.
#
if [ -f /usr/sbin/in.named  -u bind -a -f /etc/named.conf ]; then
        echo 'starting Internet domain name server.'
        /usr/sbin/in.named  -u bind &
fi
```

Modify password files /etc/passwd & /etc/shadow to create new user and group for bind.

Give group bind permissions on files used by BIND.

Test who named or in.named is running as:

```
# ps -ef | grep named
    bind   354     1  1   Jul 08 ?       25:58 /usr/sbin/in.named  -u bind
    root 26976 26975  0 15:57:52 pts/9    0:00 grep named
#
```

From the re-test, the DNS server is now compliant with Audit Item 10.

Audit Item 19 – DNS data consistent and up to date

To resolve the inconsistencies in the data, edit the forward and reverse zone files and either edit or remove those entries with which warnings and errors were associated. Then re-run the consistency checking tool. We are only running dnswalk as this appears to provide the most useful information.

```
107 >./dnswalk nsw.acme.research.gov.au.
Checking nsw.acme.research.gov.au.
Getting zone transfer of nsw.acme.research.gov.au.
Dnsserver1.nsw.acme.resea rch.gov.au....done.
SOA=dnsserver1.nsw.acme.research.gov.au
contact=hostmaster.acme.research.gov.au
0 failures, 0 warnings, 0 errors.
```

From the results it can be seen that the DNS data is now consistent, and meets the compliance requirements for the local NSW region.

Audit Item 7 – Zone transfers are authenticated

Generate the shared key to use between DNS servers.

```
# dnskeygen  -H 128 -z -n nsw.acme.research.gov.au.
Generating 128 bit HMAC -MD5 Key for nsw.acme.research.gov.au.

Generated 128 bit Key for  nsw.acme.research.gov.au. id=0 alg=157 flags=257

# more Knsw*
::::::::::::::
Knsw.acme.research.gov.au.+157+00000.key
::::::::::::::
nsw.acme.research.gov.au. IN KEY 257 3 157 Nl4FQQrSQHI54MnIhAjXuA==
::::::::::::::
Knsw.acme.research.gov.au.+157+00000.pr ivate
::::::::::::::
Private-key-format: v1.2
Algorithm: 157 (HMAC)
```

```
Key: Nl4FQQrSQHI54MnIhAjXuA==
#
```

Using the key, add an entry on DNS server dnsserver1 for dnsserver2:

```
key "nsw.acme.research.gov.au." {
        algorithm hmac-md5;
        secret "Nl4 FQQrSQHI54MnIhAjXuA==";
};

server 171.93.20.14 {
        keys { "nsw.acme.research.gov.au."; };
};
```

Now add the reciprocal – an entry on DNS server dnserver2 for dnsserver1:

```
key "nsw.acme.research.gov.au." {
        algorithm hmac-md5;
        secret "Nl 4FQQrSQHI54MnIhAjXuA==";
};

server 171.93.20.11 {
        keys { "nsw.acme.research.gov.au."; };
};
```

Modify a zone file on dnsserver1, then capture the traffic with tcpdump:

```
14:54:36.226802 171.93.20.11.62644 > 171.93.20.14.53:  32779 notify
[b2&3=0x2400] [1au] SOA? nsw.acme.research.gov.au. (122) (DF)
0x0000   4500 0096 55a0 4000 ff11 007b ab5d 140b        E...U.@....{....
0x0010   ab5d 140e f4b4 0035 0082 0ac9 800b 2400        .......5......$.
0x0020   0001 0000 0000 0001 036e 7377 0461 636d        .... ....nsw.acme
0x0030   6508 7265 7365 6102 4155 0000 0600 0103        .research.gov.au
0x0040   6e73 7704 6163 6d65 0572 6573 6561 0261        nsw.acme.researc
0x0050   7500 00fa 00ff 0000 0000 003a 0848 4d41        h.gov.au...:.HMA
0x0060   432d 4d44 3507  5349 472d 414c 4703 5245        C -MD5.SIG-ALG.RE
0x0070   4703 494e 5400 0000 3f12 386d 012c 0010        G.INT...?.8m.,..
0x0080   7d53 d05e 1727 1a47 c4ce 80f8 61c6 0b39        }S.^.'.G....a..9
0x0090   800b 0000 0000                                .....  .
14:54:36.227831 171.93.20.14.53 > 171.93.20.11.62644:  32779 notify* 0/0/1
(122) (DF)
0x0000   4500 0096 645d 4000 ff11 f1bd ab5d 140e        E...d]@.........
0x0010   ab5d 140b 0035 f4b4 0082 7c48 800b a480        .....5....|H....
0x0020   0001 0000 000 0  0001 036e 7377 0461 636d        ........nsw.acme
0x0030   6508 7265 7365 6102 4155 0000 0600 0103        .research.gov.au.
0x0040   6e73 7704 6163 6d65 0572 6573 6561 0261        nsw.acme.researc
0x0050   7500 00fa 00ff 0000 0000 003a 0848 4d41        h.  gov.au...:.HMA
0x0060   432d 4d44 3507 5349 472d 414c 4703 5245        C -MD5.SIG-ALG.RE
0x0070   4703 494e 5400 0000 3f12 386d 012c 0010        G.INT...?.8m.,..
0x0080   12e2 a652 8a5e 67fd 246d d057 4d7d 5214        ...R.^g.$m.WM}R.
0x0090   800b 0000 000 0                                 ......
14:54:36.229393 171.93.20.11.53 > 171.93.20.14.36267:  11778* 1/6/7 SOA
(406) (DF)
0x0000   4500 01b2 55a1 4000 ff11 ff5d ab5d 140b        E...U.@....]....
0x0010   ab5d 140e 0035 8dab 019e a556 2e02 8480        ... ..5.....V....
0x0020   0001 0001 0006 0007 036e 7377 0461 636d        ........nsw.acme
0x0030   6508 7265 7365 6102 4155 0000 0600 01c0        .research.gov.au
0x0040   0c00 0600 0100 0151 8000 2a06 646e 7373        .......Q..*dnsse
0x0050   6572 c00c 0a68  6f73 746d 6173 7465 72c0        rver1.hostmaster
0x0060   1077 6471 a800 002a 3000 0004 b000 24ea        .wdq...*0.....$.
0x0070   0000 0151 80c0 0c00 0200 0100 0151 8000        ...Q.........Q..
                      . . . .
```

```
                    . . . .
0x0130   0100 0002 5800 0490 6e20 02c0 bd00 0100        ....X...n.......
0x0140   0100 0151 8000 048a c21e 1ec0 d500 0100        ...Q............
0x0150   0100 0151 8000 048a c21e 0203 6e73 7704        ...Q......nsw.ac
0x0160   6163 6d65 0572 6573 6561 0 261 7500 00fa        me.research.gov.
0x0170   00ff 0000 0000 003a 0848 4d41 432d 4d44        au.....:.HMAC -MD
0x0180   3507 5349 472d 414c 4703 5245 4703 494e        5.SIG -ALG.REG.IN
0x0190   5400 0000 3f12 386d 012c 0010 72bc 3c59        T...?.8m.,..r.<Y
0x01a0   e6b0 fc92 6f73 daff 8679 ed76 2e02 0000        ....os...y.v....
0x01b0   0000                                           ..
14:54:36.306125 171.93.20.14.56001 > 171.93.20.11.53: S
476416560:476416560(0) win 8760 <mss 1460> (DF)
0x0000   4500 002c  645f 4000 ff06 f230 ab5d 140e        E..,d_@....0....
0x0010   ab5d 140b dac1 0035 1c65 8a30 0000 0000        .......5.e.0....
0x0020   6002 2238 cf26 0000 0204 05b4 5555             `."8.&......UU
14:54:36.306303 171.93.20.11.53 > 171.93.20.14.56001: S
2545488858:2545488858(0) ack 476416561 win 24820 <mss 1460> (DF)
0x0000   4500 002c 55a2 4000 4006 bfee ab5d 140b        E..,U.@.@.......
0x0010   ab5d 140e 0035 dac1 97b9 13da 1c65 8a31        .....5.......e.1
0x0020   6012 60f4 e4c5 0000 0204 05b4 5555              `.`.........UU
14:54:36.306433 171.93.20.14.56001 > 171.93.20.11.53: . ack 1 win 8760 (DF)
0x0000   4500 0028 6460 4000 ff06 f233 ab5d 140e        E..(d`@....3....
0x0010   ab5d 140b dac1 0035 1c65 8a31 97b9 13db        .......5.e.1....
0x0020   5 010 2238 3b3f 0000 5555 5555 5555             P."8;?..UUUUUU
14:54:36.306704 171.93.20.14.56001 > 171.93.20.11.53: P 1:125(124) ack 1 win
8760 34844 [1au] SOA? nsw.acme.research.gov.au. (122) (DF)
0x0000   4500 00a4 6461 4000 ff06 f1b6 ab5d 140e        E.. .da@.........
0x0010   ab5d 140b dac1 0035 1c65 8a31 97b9 13db        .......5.e.1....
0x0020   5018 2238 c0f0 0000 007a 881c 0000 0001        P."8.....z......
0x0030   0000 0000 0001 036e 7377 0461 636d 6508        ......nsw.acme.r
0x0040   7265 7365 6102  4155 0000 0600 0103 6e73        esearch.gov.au..
0x0050   7704 6163 6d65 0572 6573 6561 0261 7500        nsw.acme.researc
0x0060   00fa 00ff 0000 0000 003a 0848 4d41 432d        h.gov.au.:.HMAC -
0x0070   4d44 3507 5349 472d 414c 4703 5245 4703        MD5. SIG-ALG.REG.
0x0080   494e 5400 0000 3f12 386d 012c 0010 3879        INT...?.8m.,..8y
0x0090   4c63 663a 56f5 73c8 c53f 8634 e3d0 881c        Lcf:V.s..?.4....
0x00a0   0000 0000                                      ....
14:54:36.306921 171.93.20.11.53 > 17 1.93.20.14.56001: . ack 125 win 24820
(DF)
0x0000   4500 0028 55a3 4000 4006 bff1 ab5d 140b        E..(U.@.@.......
0x0010   ab5d 140e 0035 dac1 97b9 13db 1c65 8aad        .....5.......e..
0x0020   5010 60f4 fc06 0000 5555 5555 5555             P.`.....UUU  UUU
14:54:36.308472 171.93.20.11.53 > 171.93.20.14.56001: P 1:409(408) ack 125
win 24820 34844* 1/6/7 SOA (406) (DF)
0x0000   4500 01c0 55a4 4000 4006 be58 ab5d 140b        E...U.@.@..X....
0x0010   ab5d 140e 0035 dac1 97b9 13db 1c65 8aad        .....5....  ...e..
0x0020   5018 60f4 1ee7 0000 0196 881c 8480 0001        P.`.............
0x0030   0001 0006 0007 036e 7377 0461 636d 6508        .......nsw.acme.
0x0040   7265 7365 6102 4155 0000 0600 01c0 0c00        research.gov.au.
0x0050   0600 0100 0151 8000 2 a06 646e 7373 6572        .....Q..*.dnsser
                    . . . .
                    . . . .
0x0140   0002 5800 0490 6e20 02c0 bd00 0100 0100        ..X...n.........
0x0150   0151 8000 048a c21e 1ec0 d500 0100 0100        .Q..............
0x0160   0151 8000 048a c21e 0203 6e73 7704 636d        .Q........nsw.ac
0x0170   6973 0572 6573 6561 0261 7500 00fa 00ff        me.research.gov.
0x0180   0000 0000 003a 0848 4d41 432d 4d44 3507        au...:.HMAC -MD5.
0x0190   5349 472d 414c 4703 5245 4703 494 e 5400        SIG -ALG.REG.INT.
0x01a0   0000 3f12 386d 012c 0010 5c8b cbf8 918f        ..?.8m.,.. \.....
0x01b0   8743 fdfe 9166 9f0a 626b 881c 0000 0000        .C...f..bk......
14:54:36.308624 171.93.20.14.56001 > 171.93.20.11.53: . ack 409 win 8760
(DF)
0x0000   4500 0028 6462 4000 ff06 f231 ab5d 140e        E..(db@....1....
0x0010   ab5d 140b dac1 0035 1c65 8aad 97b9 1573        .......5.e.....s
0x0020   5010 2238 392b 0000 5555 5555 5555             P."89+..UUUUUU
14:54:36.310352 171.93.20.14.56001 > 171 .93.20.11.53: P 125:249(124) ack 409
win 8760 34845 [1au] AXFR? nsw.acme.research.gov.au. (122) (DF)
0x0000   4500 00a4 6463 4000 ff06 f1b4 ab5d 140e        E...dc@.........
```

```
0x0010   ab5d 140b dac1 0035 1c65 8aad 97b9 1573        .......5.e.....s
0x0020   5018 2238 9f2e 0000 007a 881d 0000 0001        P."8.....z......
0x0030   0000 0000 0001 036e 7377 0461 636d 6508        .......nsw.acme.
0x0040   7265 7365 6102 4155 0000 fc00 0103 6e73        research.gov.au.
0x0050   7704 6163 6d65 0572 6573 6561 0261 75 00        nsw.acme.researc
0x0060   00fa 00ff 0000 0000 003a 0848 4d41 432d        h.gov.au.:.HMAC -
0x0070   4d44 3507 5349 472d 414c 4703 5245 4703        MD5.SIG -ALG.REG.
0x0080   494e 5400 0000 3f12 386d 012c 0010 9fa0        INT...?.8m.,....
0x0090   3290 5d28 e9c9 eb54 bb51 a303 abf7 881d        2.](...T.Q......
0x00a0   0000 0000                                     ....
14:54:36.337472 171.93.20.11.53 > 171.93.20.14.56001: . 409:1869(1460) ack
249 win 24820 34845*- 1/0/1 SOA (1458) (DF)
0x0000   4500 05dc 55a5 4000 4006 ba3b ab5d 140b        E...U.@.@..;....
0x0010   ab5d 140e 0035 dac1 97b9 1573 1c65 8b29        .....5.....s.e.)
0x0020   5010 60f4 0ec2 0000 00b0 881d 8400 0001        P.`.............
0x0030   0001 0000 0001 036e 7377 0461 636d 6508         .......nsw.acme.
0x0040   7265 7365 6102 4155 0000 fc00 01c0 0c00        research.gov.au.
0x0050   0600 0100 0151 8000 2a06 646e 7373 6572        .....Q..*.dnsser
0x0060   c00c 0a68 6f73 746d 6173 7465 72c0 1077        ...hostmaster..w
0x0070   6471 a800 002a 3000 0004 b000 24ea 0000        dq...*0.....$...
0x0080   0151 8003 6e73 7704 6163 6d65 0563 7369        .Q..nsw.acme.res
0x0090   726f 0261 7500 00fa 00ff 0000 0000 003a        earch.gov.au...:
0x00a0   0848 4d41 432d 4d44 3507 5349 472d 414c         .HMAC-MD5.SIG-AL
0x00b0   4703 5245 4703 494e 5400 0000 3f12 386d        G.REG.INT...?.8m
0x00c0   012c 0010 dbe8 5e0f d55f 0573 7637 b82c        .,....^.._.sv7.,
0x00d0   9116 d441 881d 0000 0000 0032 881d 8000        ...A.......2....
0x00e0   0000 0001 0000 0000 036e 7377 0461 636d        .........nsw.acm
0x00f0   6508 7265 7365 6102 4155 0000 0200 0100        e.research.gov.a
. . . .
                         . . .
0x05a0   3388 1d80 0000 0000 0100 0000 0005 7070        3.......  ......pp
0x05b0   7470 3103 6e73 7704 6163 6d65 0543 5349        tp1.nsw.acme.res
0x05c0   524f 0241 5500 0001 0001 0001 5180 0004        earch.gov.auQ...
0x05d0   ab5d 19c8 003c 881d 8000 0000                  .....<......
14:54:36.337595 171.93.20.11.53  > 171.93.20.14.56001: P 1869:3329(1460) ack
249 win 24820 0 [29559a] [1902q] [25455n] [27699au] MX?
nsw.acme.research.gov.au. Type337 (Class 32768)? . Type13704 (Class 7552)?
^@^@^FmailhostM-@^T. Type0? . Type0 (Class 7)? .[|domain] (DF)
0x0000   4500 05dc  55a6 4000 4006 ba3a ab5d 140b        E...U.@.@..:....
0x0010   ab5d 140e 0035 dac1 97b9 1b27 1c65 8b29        .....5.....'.e.)
0x0020   5018 60f4 e4ba 0000 0001 0000 0000 076e        P.`.............n
0x0030   7377 636f 6c33 036e 7377 0461 636d 6508         swcol3.nsw.acme.
0x0040   7265 7365 6102 4155 0000 0f00 0100 0151        research.gov.au.
0x0050   8000 0b00 0006 6e73 776d 6169 c014 0035        Q.....mailhost.5
0x0060   881d 8000 0000 0001 0000 0000 076e 7377        .............nsw
0x0070   636f 6c33  036e 7377 0461 636d 6508 4353        col3.nsw.acme.re
0x0080   4952 4f02 4155 0000 0100 0100 0151 8000        search.gov.au.O.
                       . . . .
                       . . . .

14:54:36.441972 171.93.20.14.56001 > 171.93.20.11.53: F 249:249(0)  ack 78388
win 8760 (DF)
0x0000   4500 0028 6477 4000 ff06 f21c ab5d 140e        E..(dw@.........
0x0010   ab5d 140b dac1 0035 1c65 8b29 97ba 460e        .......5.e.)..F.
0x0020   5011 2238 0812 0000 5555 5555 5555             P."8....UUUUUU
14:54:36.442064 171.93.20.11.53 > 171.93.20.14.56001: . ack 250 win 24820
(DF)
0x0000   4500 0028 55db 4000 4006 bfb9 ab5d 140b        E..(U.@.@.......
0x0010   ab5d 140e 0035 dac1 97ba 460e 1c65 8b2a        .....5....F..e.*
0x0020   5010 60f4 c955 0000 5555 5555 5555              P.`..U..UUUUUU
14:54:36.442312 171.93.20.11.53 > 171.93.20.14.56001: F 78388:78388(0) ack
250 win 24820 (DF)
0x0000   4500 0028 55dc 4000 4006 bfb8 ab5d 140b        E..(U.@.@.......
0x0010   ab5d 140e 0035 dac1 97ba 460e 1c65 8b2a        .....5.  ...F..e.*
0x0020   5011 60f4 c954 0000 5555 5555 5555             P.`..T..UUUUUU
14:54:36.442391 171.93.20.14.56001 > 171.93.20.11.53: . ack 78389 win 8760
(DF)
0x0000   4500 0028 6478 4000 ff06 f21b ab5d 140e        E..(dx@.........
```

```
0x0010   ab5d 140b dac1 0035 1c65 8b2a 97ba 460f        .....5.e.*..F.
0x0020   5010 2238 0811 0000 5555 5555 5555             P."8....UUUUUU
```

As can be seen from the results, the transaction signature is being sent when a zone transfer is occurring. The transaction signatures appear until the start of authority (SOA) has been sent by the DNS server. The transaction signatures do not appear to be used after this. The DNS server is now compliant with this audit item.

System Justification

For those items which were not-compliant or partially compliant but were not changed, the reasons for not changing the system are given as follows:

Audit Item 3 – Statistics Enabled

Enabling statistics is not critical for the security of the DNS server, however it is a useful tool to have. By providing statistics a baseline of the DNS servers operation can be taken. While statistics were enabled for the DNS server itself, no statistics were recorded for those hosts which contacted the DNS server. Paul Albitz and Cricket Liu state that there is a trade-off between enabling full statistics on hosts and the DNS servers performance. As a result if full statistics were enabled, the CPU and memory usage of the DNS server would be very intensive and may possibly cause more problems than not having full statistics enabled. For full statistics a large amount of disk space would also be required. As the information on hosts connecting to the DNS server for which queries are denied or zone transfers occur can be found in the logs, this information could be correlated with the standard statistics of the DNS server. As a result full statistics has not been enabled.

Audit Item 9 – Internal Information not available to public

Due to business needs and some services which scientists are required to access, the internal workstations are visible from outside as they directly connect to external hosts rather than go through a proxy. The main reason therefore that the IP address space is visible is for the reverse-IP lookup which is required by some services. The reason for this is that all machines within the internal network use publicly addressable IP addresses, rather than private IP addresses. There are many services such as mail and web which require reverse IP lookup to authentication a host before the service will work. There are compensating controls in place to restrict access to internal hosts. The router which is acting as a firewall blocks all requests from external hosts to internal hosts to service ports and also uses stateful or reflexive access lists to block communications to internal hosts. This means that only an internal client can initiate a connection to an external host, no external hosts can initiate any communication with any internal hosts apart from those defined servers such as the web, mail and DNS servers.

Audit Item 19 – DNS data consistent and up to date

The changes to some of the DNS zones for which data is stored on the DNS server was not updated. The reason for this is that the zone data is retrieved from regional DNS servers which were outside the control of the local administrators to modify. To resolve the situation the administrators in the other regions should be notified.

Recommendations

From the results of the audit on the DNS server, and after making changes to bring some of the audit items in line with compliancy, there are still some recommendations for further securing the DNS server.

Monitoring of Internal Hosts

The weakness of networks setup in regard to the DNS server are the internal hosts. The internal hosts are allowed to perform recursive queries which can lead to cache poisoning. The internal hosts are also the only hosts which are allowed to perform DNS queries to external sites. As a result of this, the internal hosts are the only platform from which a DNS ID spoofing attack on the DNS server could occur as this requires caching and recursion. Due to this the recent Birthday Attack vulnerability can only be performed from an internal client.

The Internal hosts require monitoring for compromises, and for monitoring of patch levels and vulnerabilities as these would be the access points by which an attacker could cause cache poisoning to the DNS server. As a result there is a recommendation for some form of intrusion detection system to be applied to the internal network if other forms of prevention are not implemented such as split DNS. This is both a detective and preventative solution, as hosts are monitored for compromises and as such detecting a compromise may prevent damage to the DNS server from occurring.

Proxy Server in place for external communications

There is some risk associated with allowing internal hosts to be looked up from external hosts. While not a risk in itself, it may lead to an internal host being compromised. It is necessary to have this open at present as reverse lookup of the IP addresses is required for the authentication of some services.

By implementing a proxy server for internal hosts to communicate to external hosts however, the need for providing internal IP address and host name information can be reduced or eliminated.

Monitor DNS data and alert when problems exist

The audit showed that some of the DNS data held by the DNS server was inconsistent. While it was consistent for the local region, it was not for the remote regions. It was only noticed that there was an inconsistency when the consistency checking tools were run. This data had been incorrect for some time, but was never noticed. The same is also true of the misconfiguration of one of the remote DNS servers. As a result it is a recommendation that procedures are put in place to monitor the DNS data, such as a regular running of the data consistency tools, to provide alerting to administrators when problems exist.

Monitor DNS statistics

There is regular dumping of DNS statistics, however this information is not put to any use. There was no baseline of the DNS server activity, so there was no way to determine what the regular load the DNS server usually is. By regular monitoring of the statistics a view of the regular operation of the DNS server can be put in place, and then alerting procedures could be put in place when the statistics show there is an irregularity in the DNS servers operation or load. It is recommended that procedures, preferably automated, be put in place to monitor the DNS server statistics and alert when significant changes occur.

List of References

ADM Crew. “DNS ID Hacking Presentation”. w00w00. 2002.
URL:http://spias.act.uji.es/spi/docs/redes_doc/dnsit.txt (23 Jun. 2003).

Albitz, Paul. And Liu, Cricket. DNS and BIND. 3rd ed. Sebastopol, CA: O’Rielly & Associates, Inc., 1998.

Anonymous. Maximum Linux Security. Indianapolis, Indiana: SAMS Publishing, 1999. pp. 283-285.

Anonymous. Maximum Security. 3rd ed. Indianapolis, Indiana: SAMS Publishing, 2001. pp. 484-487.

Athanasiou, Ken. “DNS Remote Root Exploit – ADM Named 8.2/8.2.1 NXT Remote Overflow”. SANS GSEC Practicals. 17 Apr. 2000.
URL:http://www.giac.org/pratical/Ken_Athanasiou_GSEC.doc (2 Jul. 2003).

Attorney-General’s Department. Commonwealth Protective Security Manual. Barton, ACT: Commonwealth of Australia. 2000.

AusCERT. “Information Security Standards”. AusCERT. 27 May 2002.
URL:http://www.auscert.org.au/render.html?it=2248&cid=1920 (28 May 2003).

Australian Federal Police, South Australian Police, Western Australia Police and AusCERT. “2003 Australian Computer Crime and Security Survery”. Mar. 2003.
URL:http://www.auscert.org.au/donwload.html?f=65/ and
URL:http://www.auscert.org.au/render?cid=1920 (28 May 2003).

Baranowski, Susan. "How Secure are the Root DNS Servers?". SANS Reading Room. 6 May 2003.
URL:http://www.sans.org/rr/catindex.php?cat_id=991 (14 Jul. 2003).

Barr. “dnswalk – a DNS database debugger”.
URL:http://www.visi.com/~barr/dnswalk (27 May 2003).

Bellovin, Steven M. “Using the Domain Name System for System Break-ins”. AT&T Bell Laboratories. Proceedings of the Fifth Usenix UNIX Security Symposium. Salt Lake City, UT. 1995.
URL:ftp://ftp.research.aat.com/dist/smb/dnshack.ps (28 May 2003).

Bennie, N. “GIAC Training & Certification - Level Two Firewalls, Perimeter Protection, and VPN's - GCFW Practical Assignment”. SANS GCFW Practicals. 2001.
URL:http://www.giac.org/practical/Norrie_Bennie_gcfw.doc (28 May 2003).

Bernstein, D.J. “Notes on the Domain Name System”.
URL:http://cr.yp.to/djbdns.html (28 May 2003).

Bernstein, D.J. “DNS Forgery”.
URL:http://cr.yp.to/djbdns/forgery.html (28 May 2003).

Boran, Sean. “Hardening the BIND DNS server”. 5 Dec 2000.
URL:http://networking.earthweb.com/netsecur/print.php/625781 (28 May 2003).

Bytefusion. “DNS/IP spoofing”. Bytefusion. 2003.
URL:http://www.bytefusion.com/products/en/secex/dnsipsoofing.htm (23 Jun. 2003).

Caloyannides, Michael. A. “Potentially Catastrophic Vulnerabilities of the Internet and Proposed Remedies”. Falls Church, VA: Mitretek Systems. Dec. 2002.
URL:http://www.ists.dartmouth.edu/ISTS/ists_docs/intvuln.pdf (7 Jul. 2003).

Carli, Florent. "Security Issues with DNS". SANS Reading Room. 2 Jun. 2003.
URL:http://www.sans.org/rr/catindex.php?cat_id=1069 (14 Jul. 2003).

CERT/CC. “Vulnerability Note VU#457875: Various DNS service implementations generate multiple simultaneous queries for the same resource record”. Cert Coordination Center. 30 May 2003.
URL:http://www.kb.cert.org/vuls/id/457875 (17 Jun. 2003).

CERT/CC. “CERT Incident Note IN-2000-04: Denial of Service Attacks using Nameserver”. Cert Coordination Center. 15 Jan 2001.
URL:http://www.cert.org/incident_notes/IN-2000-04.html (17 Jun. 2003).

CERT/CC. “CERT Advisory CA-2002-19 Buffer Overflows in Multiple DNS Resolver Libraries”. Cert Coordination Center. 9 Sep. 2002.
URL:http://www.cert.org/advisories/CA-2002-19.html (3 Jun. 2003).

CERT/CC. “CERT Incident Note IN-2001-03: Exploitation of BIND Vulnerabilities”. Cert Coordination Center. 30 Mar. 2001.
URL:http://www.cert.org/incident_notes/IN-2001-03.html (17 Jun. 2003).

CERT/CC. “CERT Advisory CA-2001-02 Multiple Vulnerabilities in BIND”. Cert Coordination Center. 7 Aug. 2001.
URL:http://www.cert.org/advisories/CA-2001-02.html (3 Jun. 2003).

CERT/CC. “Cert Advisory CA-2000-03 Continuing Compromises of DNS Servers. Cert Coordination Center. 26 Apr. 2000.
URL:http://www.cert.org/advisories/CA-2000-03.html (3 Jun. 2003).

CERT/CC. “CERT Advisory CA-1999-14 Multiple Vulnerabilities in BIND”. Cert Coordination Center. 25 Apr. 2000.
URL:http://www.cert.org/advisories/CA-1999-14.html (3 Jun 2003).

CERT/CC. “CERT* Advisory CA-98.05 Multiple Vulnerabilities in BIND”. Cert Coordination Center. 16 Nov. 1998.
URL:http://www.cert.org/advisories/CA-98.05.bind_problems.html (3 Jun. 2003).

CISECURITY. “Solaris Benchmark v1.2.0”. The Center for Internet Security. 19 Feb. 2003.
URL:https://www.cisecurity.org/tools2/solaris/SolarisBenchmark.pdf (23 Apr 2003).

COBIT.”Management Guidelines”. COBIT.
URL:http://www.isaca.org/mg.pdf (17 Jun. 2003).

COBIT. “Executive Summary”. COBIT.
URL:http://www.isaca.org/execsum.pdf (17 Jun. 2003).

COBIT. “Framework”. COBIT.
URL:http://www.isaca.org/framewrk.pdf (17 Jun. 2003).

COBIT. “Control Objectives”. COBIT.
URL:http://www.isaca.org/control.pdf (17 Jun. 2003).

COBIT. “Implementation Tool Set”. COBIT.
URL:http://www.isaca.org/cobitits.pdf (17 Jun. 2003).

Cohen, Beth. “DNSSEC: Security for Essential Network Services”.
URL:http://networking.earthweb.com/netsecur/print/php/2204801 (17 Jun. 2003).

DAG. “BIND Dynamic DNS (DDNS) updates using nsupdate”. DAG. 14 Feb. 2003.
URL:http://dag.wieers.com/howto/bits/bind_ddns.php (3 Jul. 2003).

Department of the Prime Minister and Cabinet. “Security in the Government Sector”. Department of the Prime Minister and Cabinet. 22 Jul. 2002.
URL:http://www.security.govt.nz/sigs/index.html (28 May 2003).

Detoisien. Eric. “System Administration: External Attacks”. Linux Focus.
URL:http://en.tldp.org/linuxfocus/English/March2003/article282.shtml (28 May 2003).

DSD Customer Service Team. “Australian Communications-Electronic Security Instruction 33 (ACSI 33) – Version 1.0”. Defence Signals Directorate.
URL:http://www.dsd.gov.au/infosec/publications/asci33.html (28 May 2003).

DSD Information Security Group. “Internet Systems – Security and Authentication Issues – version 1”. Defence Signals Directorate. 2001
URL:http://www.dsd.gov.au/infosec/publications/intsystems.html (28 May 2003).

DSD. “Gateway Certitfication Guide – Version 2.1”. Defence Signals Directorate. Feb. 2001.
URL:http://www.dsd.dov.au/infosec/Publications/gcg.html (26 Mar. 2003).

E-Mind. “BIND 8.2-8.2.2 * Remote root How-To exploit *”.
URL:http://www.securent-2000.com/articile.php?sid=332 (2 Jul. 2003).

Erdfelt, Johannes. “Everything you ever wanted to know about DNS spoofing”. 25 Jul. 1997.
URL:http://www.the-project.org/admins/0797/msg00070.html (23 Jun. 2003).

Extralan. ‘Simple DNS Plus”. Extralan. 2003.
URL:http://www.extralan.co,uk/products/Diagnostic-Tools/SDNSPlus/SDNSPlus.htm (24 Jun. 2003).
- DNS tool for managing Microsoft Windows DNS

Faure, F. “Quick Guide to BIND 8”.
URL:http://perso.club-Internet.fr/ffaure/dns.html (26 Jun. 2003).

Fiore, Frank. And Francois, Jean. “Unwitting Collaborators, Part 11: DNS Poisoning and Domain Hijacking”. 30 Aug. 2002.
URL:http://www.informit.com/isapi/product_id~{DIC81609-A591-423D-ACAF-3A562B8B96A5}/content/index.asp (2 Jul. 2003).

Fresh T-Systems SfR. “Fresh – the T-Systems SfR Freeware/Shareware Archive”. Fresh T-Systems SfR. 22 Apr. 2003.
URL:http://fresh.t-systems-srf.com/unix/src/misc/dns (27 May 2003).

Giovanni, Coretez. “Bypassing Secure Web Transactions via DNS corruption: A man-in-the-middle attack”. Endevour Systems. 27 Apr. 1999.
URL:http://packetstormsecurity.nl/papers/general/middleman.pdf (17 Jun. 2003).

Gregory, Peter. H. Solaris Security. Upper Saddle River, New Jersey: Sun Microsystems Press/Prentice-Hall, Inc. 2000. pp. 181-189, 198-200.

Groothius, Edwin..“DNSTRACER – Exploring the DNS Infrastructure”. SAGE Advice 2003. Volume 9 Number 1. SAGE-AU.

Hanely, Sinead. “DNS Overview with a discussion of DNS Spoofing”. 6 Nov. 2002.
URL:http://www.bgc.com.au/dos/win9x/dns_spoofing.html (24 Jun. 2003).

Hatch, Brian., Lee, James., & Kurtz, George. Hacking Linux Exposed – Linux Security Secrets & Solutions. Berkely, California: Osbourne/McGraw-Hill. 2001.pp. 81-88, 216-217, 226-227.

Hinshelwood, David. "DNS, DNSSEC and the Future". SANS Reading Room. 30 May 2003.
URL:http://www.sans.org/rr/catindex.php?cat_id=1054 (14 Jul. 2003).

Holland, Jeff. “DNS Security”. 23 Jul. 2000.
URL:http://www.whitehats.ca/main/members/Jeff/jeff_dns_security/jeff_dns_security.htm (30 Jun. 2003).

Householder, Allen., King, Brian. And Silva, Ken. “Securing an Internet Name Server”. Cert Coordination Center. Aug 2002.
URL:http://www.cert.org/archive/pdf/dns.pdf (17 Jun. 2003).

iDefence. “iDefence Security Advisories”. IDefense.
URL:http://www.idefense.com/current.html (28 May 2003).

Internet Security Systems. “Top 10 Security Risks”. ISS Security Alert.
URL:http://www.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20337 (3 Jun. 2003).

Internet Security Systems. “Mulitple Remote Vulnerabilities in BIND 4 and BIND 8”. Internet Security Systems. 12 Nov. 2002.
URL:http://bvlive01.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=21469 (2 Jun. 2003).

Internet Security Systems. “Remote Denial of Service Vulnerability in ISC BIND”. ISS Security Alert. 4 Jun. 2002.
URL:http://www.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20210 (3 Jun. 2003).

Internet Security Systems. “Internet Security Systems Security Alert Summary AS01-09”. ISS Security Alert. 3 Dec. 2001.
URL:http://www.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20346 (3 Jun. 2003).

Internet Security Systems. “ISS Security Alert Summary”. Vol 6, No. 8. ISS Security Alert. 9 Jul. 2001.
URL:http://www.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20427 (3 Jun. 2003).

Internet Security Systems. “ISS Security Alert Summary”. Vol 6, No. 3. ISS Security Alert. 6 Feb. 2001.
URL:http://www.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20422 (3 Jun. 2003).

Internet Security Systems. “Remote Vulnerabilities in BIND versions 4 and 8”. ISS Security Alert. 29 Jan 2001.
URL:http://www.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20313 (3 Jun. 2003).

Internet Security Systems. “bind inverse query disclosure”. Internet Security Systems. 29 Jan. 2001.
URL:http://xforce.iss.net/xforce/xfdb/6018 (3 Jun. 2003).

Internet Security Systems. “ISS Security Alert Summary”. Vol 6, No. 1. ISS Security Alert. 4 Dec. 2000.
URL:http://www.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20418 (3 Jun. 2003).

Internet Security Systems. “bind-zxfr-dos”. Internet Security Systems. 7 Nov 2000.
URL:http://xforce.iss.net/xforce/xfdb/5540 (3 Jun. 2003).

Internet Security Systems. “isc-bind-axfr-bo”. Internet Security Systems. 1 Nov 2000.
URL:http://xforce.iss.net/xforce/xfdb/5462 (3 Jun. 2003).

Internet Security Systems. “ISS Security Alert Summary”. Vol4, No. 9. ISS Security Alert. 15 Nov. 1999.
URL:http://www.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20407 (3 Jun. 2003).

Internet Security Systems. “ISS Security Alert Summary”. Vol 2, No. 5. ISS Security Alert. 24 Apr. 1998.
URL:http://www.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20383 (3 Jun. 2003).

Internet Software Consortium. “BIND 8.2T5B – BIND users ML Archive”. Internet Software Consortium. 29 Jan. 1999.
URL:http://www.isc.org/ml-archives/bind-users/1999/01/msg00136.html (3 Jun. 2003).

Internet Software Consortium. “Other BIND and DNS Resources”. Internet Software Consortium. 2003.
URL:http://www.isc.org/products/BIND/contributions.html (3 Jun. 2003)

Internet Software Consortium. “BIND Vulnerabilities”. Internet Software Consortium. 2003.
URL:http://www.isc.org/products/BIND/bind-security.html (3 Jun. 2003)

Internet Software Consortium. “ISC BIND”. Internet Software Consortium. 2003.
URL:http://www.isc.org/products/BIND (3 Jun. 2003)

Interpol. “Company Checklist”. Interpol.
URL:http://www.interpol.int/Public/TechnologyCrime/CrimePrev/companyChecklist.asp (7 Jul. 2003).

Interpol. “IT Security and crime prevention methods”. Interpol.
URL:http://www.interpol.int/Public/TechnologyCrime/CrimePrev/ITSecurity.asp (7 Jul. 2003).

Irving, Christopher. "The Achilles Heal of DNS". SANS Reading Room. 2 Aug. 2001.
URL:http://www.sans.org/rr/catindex.php?cat_id=565 (14 Jul. 2003).

Izaguire, Ramon. “An Implementation of a Birthday Attack in a DNS Spoofing”. Bugtraq. 24 Apr. 2003.
URL:http://www.securityfocus.com/archive/1/319622 (28 May 2003).
– *includes source code for attack.*

Jackiewicz, Tom. “Best Practices – Naming Version 1.1”. 23 May 2001. pp. 36-60.
URL:http://www.sun4c.net/papers/taos-naming.pdf (17 Jun. 2003).

Jones, Jim. “Coding with the DNS protocol v2”. 2002.
URL:http://packetstormsecurity.nl/programming-tutorials/coding-with-the-dns-protocol.txt (26 May 2003).

Klemencic, Joe. "The Flat Footed Hacker". SANS Reading Room. 17 Sep. 2001.
URL:http://www.sans.org/rr/catindex.php?cat_id=566 (14 July 2003).

Krauz, Pavel. “Pavel Krauz’s Home Page.”
URL:http://lin.fsid.cvut.cz/~kra/index.html (23 Jun. 2003).
- *Hunt Project*

L0T3K. “Spoofing”. L0T3K.
URL:http://www.l0t3k.org/security/docs/spoofing/en (23 Jun. 2003).

Langfeldt, Nicolai., Norrish, Jamie. & others. “DNS HOWTO”. 2001.
URL:http://langfeldt.net/DNS-HOWTO/ (28 May 2003).

Lau, Steve. "Why is securing DNS zone transfer necessary ?". SANS Reading Room. 31 Mar. 2003.
URL:http://www.sans.org/rr/catindex.php?cat_id=868 (14 Jul. 2003).

Lierley, Mark (Editor). Security Complete. Alameda, California: Sybex, Inc., 2001. pp. 451-495.

Linux-Sec. “DNS Server Hardening”. Linux-Consulting. 22 May 2003.
URL:http://www.linux-sec.net/DNS (27 May 2003).

Liu, Cricket. “Securing an Internet Name Server”. Verisign.
URL:http://www.linuxsecurity.com/resource_files/server_security/securing_an_Internet_name_server.pdf (28 May. 2003).

Lugo, Dave “Dual chrooted BIND/DNS servers”. 2000.
URL:http://www.etherboy.com/dns/chrootdns.html (3 Jun. 2003).

McClure, Stuart., Scambray, Joel., Kurtz, George. Hacking Exposed: Network Security Secrets & Solutions. 3rd ed. Berkley, California: Osbourne/McGraw Hill, 2001. pp. 19-24, 347-351, 506-507, 283.

Medvedovsky, Ilya. “Fake DNS Servers in the Internet (DNS Spoofing)”. Bugtraq (Russian). 26 Nov. 2002.
URL:http://bugtraq.ru/library/security/dns.html (24 Jun. 2003).

Men & Mice. ”What is DNS spoofing?”. Men & Mice.
URL:http://www.menandmice.com/9000/9211_dns_spoofing.html (17 Jun. 2003).

Men & Mice. ”DNS Survey”. Men & Mice.
URL:http://www.menandmice.com/6000/6000_domain_health.html (17 Jun. 2003).

Men & Mice. ”Survey Results: Domain Health Survey for .COM”. Men & Mice. Feb. 2003.
URL:http://www.menandmice.com/6000/61_recent_survey.html (17 Jun. 2003).

Men & Mice. ”BIND Vulnerability”. Men & Mice.
URL:http://www.menandmice.com/6000/6200_bind_research.html (17 Jun. 2003).

Microsoft Corporation. “Domain Name System (DNS) Center”. Microsoft Corporation.
URL:http://www.microsoft.com/windows2000/technologies/communications/dns/default.asp (7 Jul. 2003).

Mixter. “A brief programming tutorial in C for raw sockets:.
URL:http://security.royans.net/info/rawip/code/rawip1.shtml (7 Jul. 2003).

Mockapetris, P. “RFC 1035: Domain Names – Implementation and Specification”. Nov. 1987.
URL:http://www.faqs.org/rfcs/rfc1035.html (30 May 2003).

Nemo. “DNS Abuser”: Deepzone.
URL:http://www.deepzone.org/advisories/adv06.htm (26 Jun. 2003).

Neophasis. “BIND Announce – Update to BIND Vulnerabilities”. Neophasis Archives. 31 Jul 2002.
URL:http://archives.neohapsis.com/archives/bind/2002/0015.html (24 Jun. 2003).

Neosoft. dig man page. Neosoft.
URL:http://www.neosoft.com/neosoft/man/dig1.html (27 May 2003).

Nessus Project. “Download the stable version of the Nessus Security Scanner for Unix-Compatible systems”. Nessus Project.
URL:http://www.nessus.org/nessus_2_0.html (28 May 2003).

NOIE. Index Web Page. National Office for the Information Economy.
URL:http://www.govonline.gov.au/projects/index.htm (16 May 2003).

Nominum. “How to Measure the Performance of a Caching DNS Server”. Nominum, Inc.
URL:http://www.nominum.com/content/documents/CNS_WP.pdf (27 May 2003).

Nominum. “Nominum DNS Response Validator”. Nominum.
URL:http://www.nominum.com/product.php?id=DRV (24 Jun. 2003).

Nominum. “BIND FAQs”. Nominum.
URL:http://www.nominum.org/getOpenSourceResource.php?id=6 (30 May 2003).

Office of Information Technology. “Information Management Audit Guideline”. Office of Information Technology.
URL:http://www.oit.nsw.gov.au/pages/4.3.12-IM-Audit.htm (16 May. 2003).

Office of Information Technology. “Information Security Guidelines for NSW Government Agencies September 2001”. Office of Information Technology.
URL:http://www.oit.nsw.gov.au/pages/12.2.2.Information-Security.htm (16 May. 2003).

Office of Information Technology. “Information Security Guidelines Part 1 – Risk Management - Summary”. Office of Information Technology.
URL:http://www.oit.nsw.gov.au/pages/4.3.16-Security-Pt1.htm (16 May. 2003).

Office of Information Technology. “Information Security Guidelines Part 2 – Threats and Vulnerabilities - Summary”. Office of Information Technology.
URL:http://www.oit.nsw.gov.au/pages/4.3.17-Security-Pt2.htm (16 May. 2003).

Office of Information Technology. “Information Security Guidelines Part 3 – Baseline Controls - Summary”. Office of Information Technology. 2003.
URL:http://www.oit.nsw.gov.au/pages/4.3.18-Security-Pt3.htm (16 May. 2003).

Romao.Artur. “RFC 1713 – Tools for DNS Debugging”. Nov 1994.
URL:http://www.faqs.org/rfcs/rfc1713.html (27 May 2003).

Sacramento, Vagner. “Vulnerability in the sending requests control of Bind version 4 and 8 allows DNS spoofing”. Ccais/RNP and DIMAp/UFRN. 2002. URL:http://www.rnp.br/cais/alertas/2002/cais-ALR-19112002a.html (28 May 2003).

Sahlin, Bengt. “New Challenges to the Domain Name SystemL extensions for security, dynamic updates and IPv6”. Helsinki University of Technology. 29 Nov. 1999.
URL:http://www.tml.hut.fi/~bos/dns.html (3 Jul. 2003).

SANS. “SANS/FBI Top 20 List”. SANS. 29 May 2003.
URL:http://www.sans.org/top20/ (18 Jun. 2003).

SANS. “SANS Critical Vulnerability Analysis (CVA) “ Vol 1, Issue 19. SANS. 1 Dec. 2002.
URL:http://www.sans.org/newsletters/cva/cva1_19.php (24 Jun. 2003).

SANS. “Lion Worm”. SANS. 18 Apr. 2001.
URL:http://www.sans.org/y2k/lion.htm (17 Jun. 2003).

Sax, Doug. “DNS Spoofing (Malicius Cache Poisoning)”. SANS. 12 Nov. 2000.
URL:http://www.sans.org/infosecFAQ/firewall/DNS_Spoof.htm (28 May 2003).

SearchSecurity. “Glossay – Definition – Hijacking”. SearchSecurity. 7 Feb.2003.
URL:http://searchsecurity.techtarget.com/gDefinition/0,294236,sid14_gci519370,00.html (7 Jul. 2003).

SecuriTeam. “SecuriTeam.com (Archive) – Exploits”.SecuriTeam.
URL:http://www.securiteam.com/exploits/archive.html (7 Jul. 2003).

SecuriTeam. “[UNIX] Vulnerability in Requests Control of BIND Versions 4 and 8 Allows DNS Spoofing”. Securiteam. 2002.
URL:http://www.derkeiler.com/Mailling-Lists/Securiteam/2002-11/0095.html (28 May 2003).

SecurityFocus. “2002-006: buffer overrun in libc DNS resolver”. SecurityFocus. 27 Jun. 2002.
URL:http://www.securityfocus.com/advisories/4243 (17 Jun. 2003).

SecurityFocus. “AL-99.004: Denial of Service (Dos) attacks using the Domain Name System (DNS)”. SecurityFocus.13 Aug. 1999.
URL:http://www.securityfocus.com/advisories/1705 (17 Jun. 2003).

Send Chor, Lim. "DNS Security Considerations and the Alternatives to BIND". SANS Reading Room. 2 Oct. 2001.
URL:http://www.sans.org/rr/catindex.php?cat_id=567 (14 Jul. 2003).

Skoudis, Ed. Counter Hack.Upper Saddle River, New Jersey: Prentice-Hall. 2002.

Soderstrom, Gustav. “Man in the middle Attacks – A practical approach in a switched environment”. KTH IMIT, Royal Institute of Technology, Stockholm Sweden. 2002.
URL:http://www.e.kth.se/~e97_gso (24 Jun. 2003).

Softpanorama. “Softpanorama DNS Security Page”. Softpanorama. 4 Jan. 2003.
URL:http://www.softpanorama.org/security/dns_security.shtml (30 Jun. 2003).

Sorenson, Holt. “Secure Installation of BIND”. SecurityFocus. 8 Feb. 2000.
URL:http://www.securityfocus.com/infocus/1361 (29 May 2003).

Standards Australia. AS/NZ 7799.1:2003 – Information Security Management – Part 1. Strathfield, NSW: Standards Australia International Ltd., 2003.

Standards Australia. AS/NZ 7799.1:2003 – Information Security Management – Part 2. Strathfield, NSW: Standards Australia International Ltd., 2003.

StartX. “DNS Spoofing – Basic”. 2001.
URL:http://www.under-host.com/hosts/ugw/tutorials/dnsspoofing.txt (23 Jun. 2003).

Stevens, Richard. W. TCP/IP Illustrated, Volume 1. Reading, Massachusetts: Addison Wesley Longman, 1994. pp.187-208.

Stewart, Joe. “DNS Cache Poisoning – The Next Generation”. 27 Jan 2003.
URL:http://www.securityfocus.com/guest/17905 (28 May 2003).
– *includes source code for spoof test.*

Sun Microsystems. In.named(1M) man page. Solaris 2.7. 24 Feb. 1998.

Sweetman, James. "Current Issues in DNS Security: ICANN's November 2001 Annual Meeting". SANS Reading Room. 28 Nov. 2001.
URL:http://www.sans.org/rr/catindex.php?cat_id=568 (14 Jul. 2003).

Teoh, Cheng. "Defense in Depth for DNS". SANS Reading Room. 13 Feb. 2003.
URL:http://www.sans.org/rr/catindex.php?cat_id=867 (14 Jul. 2003).

Thomas, Rob. “Secure BIND Template Version 4.0”. 8 Apr. 2003.
URL:http://www.cymru.com/Documents/secure-bind-template.html (17 Jun. 2003).

Tunnissen, Jacco. “Securing the Domain Name System with DNSSEC (DNS Security Extensions)”.
URL:http://www.dnssec.net (18 Jun. 2003).

Valgasu. “WinDNSSpoof”.SecuriteInfo. 2003.
URL:http://www.securiteinfo.com/outils/WinDNSSpoof.shtml (3 Jun. 2003).

Van Hauser. “Placing Backdoors Through Firewalls – v1.5”.
URL:http://www.thc.org/papers/fw-backd.htm (18 Jun. 2003).

Verten, Dani. “Update: Groups warn of Server Internet Security Hole”
Computerworld. 29 Jan. 2001.
URL:http://www.computerworld.com/securitytopics/security/story/0,10801,57079,00.html (4 Jun. 2003).

Wunsch, Scott.“Chroot-BIND8 HOWTO”. 2001.
URL:http://www.locurs.org/docs/howto/Chroot-BIND.html (3 Jun. 2003).

URLs for Tools

Adig
URL:http://nscan.hypermart.net/cgi-bin/link.pl?link=dig041 (12 Jul. 2003).
URL:http://packetstormsecurity.nl/Win/Adig02f.zip (3 Jul. 2003).

bindstat
URL:http://nrg.help.wisc.edu/bindstat.html (7 Jul. 2003).

denver
URL:http://www.ilionsecurity.ch/denver/ (7 Jul. 2003).

Dlint
URL:http://www.domtools.com/dlint/ (28 Jun. 2003).

DNSabuser
URL:http://www.deepzone.org/advisories/adv06.htm (10 Jul. 2003).

dnssend.c
URL: http://packetstorm.linuxsecurity.com/groups/w00w00/misc/dnssend.c
(18 Jun. 2003).

dnsstats
URL:http://fresh.t-systems-sfr.com/unix/src/misc/dns/dnsstats (29 Jun. 2003).

dnswalk
URL: http://www.visi.com/~barr/dnswalk (17 Jun. 2003).

DOC
URL:http://fresh.t-systems-sfr.com/unix/src/misc/dns/doc-2.2.3.tar.gz (29 Jun. 2003).

Forgeron
URL:http://www.iiens.net/pilon/forgeron (18 Jun. 2003).

Host
URL:http://ejo.univ-lyon1.fr/unix/network/tcpip/dns/host.tar.gz (7 Jul. 2003).

Hunt v1.5
URL:http://lin.fsid.cvut.cz/~kra/index.html (7 Jul. 2003).

nmap
URL:http://www.insecure.org/nmap/nmap_download.html (28 May 2003).

Packit 0.5.0
URL: http://packetstormsecurity.nl/UNIX/misc/packit-0.5.0.tgz (7 Jul. 2003).

poison.pl (Ramon Izaguire's script).
URL:http://www.securityfocus.com/archive/1/319622 (24 Apr. 2003).

Simple DNS Plus
URL:http://www.extralan.co.uk/products/Diagnostic-Tools/SDNDPlus.htm (7 Jul. 2003).

snoof
URL:http://packetstormsecurity.nl/spoof/unix-spoof-code/snoof.tar.gz (18 Jun. 2003).

spoof.c
URL:http://packetstorm.linuxsecurity.com/groups/w00w00/misc/spoof.c (18 Jun. 2003).

Spooftest.pl (Joe Stewarts test program)
URL:http://www.securityfocus.com/guest/17905 (28 May 2003).

WinDNSSpoof
URL:http://www.securiteinfo.com/outils/WinDNSSpoof.shtml (18 Jun. 2003).

Zodiac
URL:http://www.packetfactory.net/projects/zodiac/ (18 Jun. 2003).

There are a couple of good sites I found useful for getting tools for DNS. These are:

URL:http://ejo.univ-lyon1.fr/unix/network/tcpip/dns/ (10. Jul. 2003).
URL:http://fresh.t-systems-srf.com/unix/src/misc/dns (27 May 2003).