

CPNI VIEWPOINT 01/2008

DOMAIN NAME SYSTEM (DNS)

MAY 2008

Abstract

This Viewpoint considers some of the security considerations of the Domain Name System and makes some observations regarding how organisations can begin to reduce their risk.

CPNI Viewpoints

CPNI Viewpoints are intended to provide a management level overview of emerging risks. A Viewpoint may not necessarily offer mitigation advice; other CPNI products are available for this purpose.

Disclaimer

Reference to any specific commercial product, process or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement, recommendation, or favouring by CPNI. The views and opinions of authors expressed within this document shall not be used for advertising or product endorsement purposes.

To the fullest extent permitted by law, CPNI accepts no liability for any loss or damage (whether direct, indirect or consequential and including, but not limited to, loss of profits or anticipated profits, loss of data, business or goodwill) incurred by any person and howsoever caused arising from or connected with any error or omission in this document or from any person acting, omitting to act or refraining from acting upon, or otherwise using, the information contained in this document or its references. You should make your own judgement as regards use of this document and seek independent professional advice on your particular circumstances.

CONTENTS

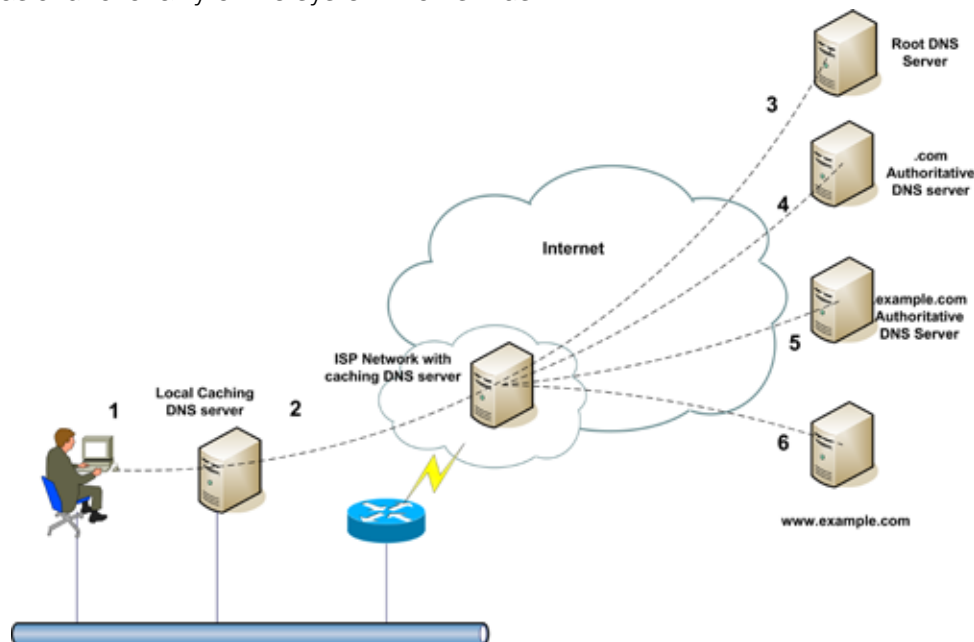
Key points	3
Introduction.....	4
Security considerations	5
Denial of service (dos) attacks against dns	5
Dos attacks against networks delivered via dns.....	6
Dns cache poisoning.....	7
Dns covert channels.....	7
Tunnelling.....	7
Rebinding	8
Application level attacks	8
Reducing vulnerability in dns	9
Securing the server and operating system.....	9
Physical security must come first:	9
Separation of functionality:	9
Operating system updates:	9
Securing the application.....	10
Defending against dos.....	10
Protecting the user of dns services.....	11
Conclusions	11
References / further reading	12

KEY POINTS

- DNS is essential for the proper functioning of both private networks and the internet
- The original protocol design does not take account of security issues
- DNS can be exploited to compromise the confidentiality, integrity and availability of data
- Various attacks have been documented which exploit features of DNS in an effort to manipulate the service or else to simply deny service to legitimate users
- A certain amount of effort is required to maintain secure configurations of devices which run DNS software
- Software must be updated regularly to maintain security
- The successful deployment of Next Generation Networks will also depend heavily on DNS services
- Operators/owners of Critical National Infrastructure should review their DNS infrastructure to minimise the risks from network attack and loss of confidential data/documents

INTRODUCTION

1. The Domain Name System (DNS) is a distributed internet directory service. DNS is primarily used to translate domain names to IP addresses and vice-versa, and to control email delivery. Most internet services rely on DNS to work. If DNS fails or is too slow then web sites cannot be reached and email delivery fails.
2. Domain naming makes it much easier for users to be able to locate resources on the network, as it is much more intuitive to browse to `www.mywebserver.com` rather than having to remember the actual 32-bit network address of the server in the form `192.168.100.101`, for example. DNS takes care of this translation in the background and remains largely transparent to end users/applications.
3. The basic functionality of the system works thus:



- (1) The user opens a web browser and requests a page from **www.example.com**; the PC doesn't know where to find this page so refers to the local DNS server in order to find the right IP address.
- (2) The LOCAL DNS SERVER might have the address in cache already if another user has recently requested information from the same site; if not it then refers upstream to the ISP's DNS SERVERS.
- (3) The ISP DNS server asks the ROOT DNS SERVER for the location of `www.example.com` (assuming no record already cached) and gets back a referral to the .com AUTHORITATIVE NAME SERVER.
- (4) The ISP DNS server then asks the .com name server for the location of `www.example.com` and gets back a referral to the .example.com authoritative name server.
- (5) ISP DNS server now talks to the name server which is in charge of the .example.com zone and requests the address for `www.example.com`; this address is passed back via the caching servers to the remote user.

- (6) User makes a connection to the IP address of www.example.com and the web server returns the page to the user's browser. The user's PC will also invoke a caching function so that within certain time-limits no further DNS lookups will be required to find the same resource.
4. The DNS is basically made up of three components: DNS Resolvers, DNS Servers and DNS data in the form of Resource Records (RR). The internet as a whole contains millions of RRs, which are divided up into zones. Each zone has a set of authoritative Name Servers which look after that zone.
5. There are 13 clusters of special DNS servers around the globe which are known as the ROOT SERVERS; these are identified using the letters A through to M. The root servers essentially answer queries regarding the Top Level Domains (TLDs): com, org, mil, gov, org, uk, eu and others. All DNS servers should contain information about the root servers within their configuration files, so that in the event of no useful cached information being available a query can start from the internet root servers.

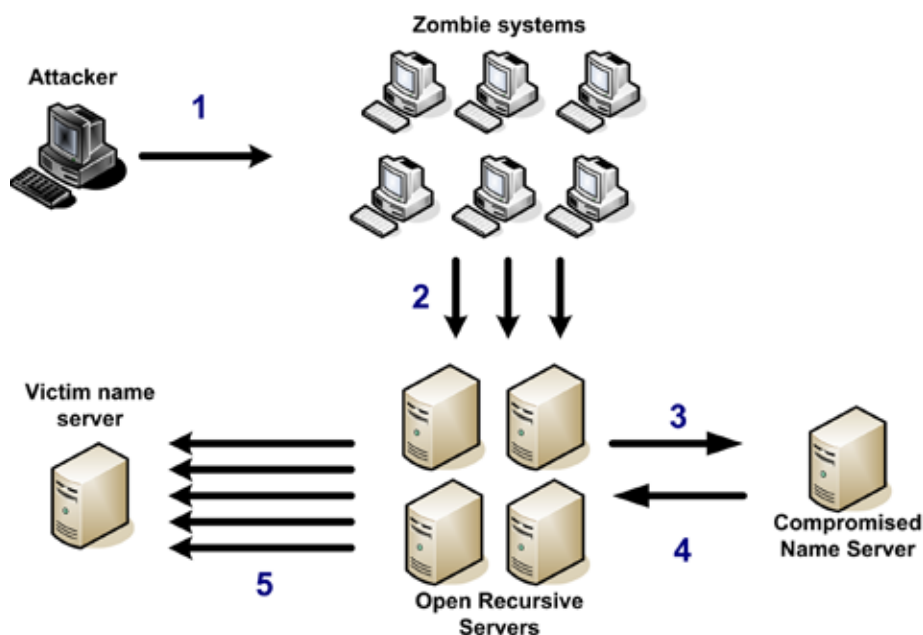
SECURITY CONSIDERATIONS

Denial of service (Dos) attacks against dns

6. Denial of Service (DoS) is an incident during which an individual/group/organisation cannot access a service which it would normally expect to be available. "Service" could be almost anything, but in this context it refers to a particular application, the DNS server software. DoS attacks involving DNS are of two distinct forms; the first is where the attack is directed against the DNS servers specifically, and the second is where the DNS mechanism is itself used as the vehicle to deliver a DoS attack against a third party. In this section we will look at the first kind of attack.
7. A DoS attack generally involves sending more traffic than the destination can handle at any one time. In the context of DNS this means that web browsing and email will become unavailable for many users, as they are unable to resolve hostnames. Attack traffic is usually sent from a spoofed (false) IP address, so that the source is difficult to identify.
8. To make the attack even more effective, the attacker could build himself a robot network (botnet) in order to deliver the attack traffic as a Distributed Denial of Service (DDoS). A botnet is a collection of systems which have been compromised and are subsequently controlled by a malicious attacker (usually known as a bot herder) as a single 'network'; these are most often the PCs of home-users with broadband connections, who remain largely oblivious to the compromise.
9. DDoS attack using a botnet can involve many thousands of compromised systems (zombies) and is an extremely potent weapon. Botnets can even be rented for a short time specifically for this kind of malicious activity.
10. There have been several recorded incidents of attempted DoS attacks against the internet root servers. These have had varying degrees of success in achieving their aim, but none has resulted in establishing internet-wide DoS against the DNS.

DoS Attacks against networks delivered via DNS

11. The architecture of DNS can be exploited by a malicious attacker to deliver DoS attacks against other servers and networks. This style of attack depends upon some advance preparation by the attacker, and on his ability to find a large number of open recursive DNS servers which he can leverage to deliver the attack. Open recursive servers are those which will answer a particular type of DNS query called a recursive query, but which will answer for anyone who sends the query (rather than recommended best practice which is to reply only to a trusted set of clients).
12. Finding these open recursive servers is extremely easy as there are websites available which will provide search facilities for them. The attacker delivers his DoS to the victim using a method similar to that shown below:

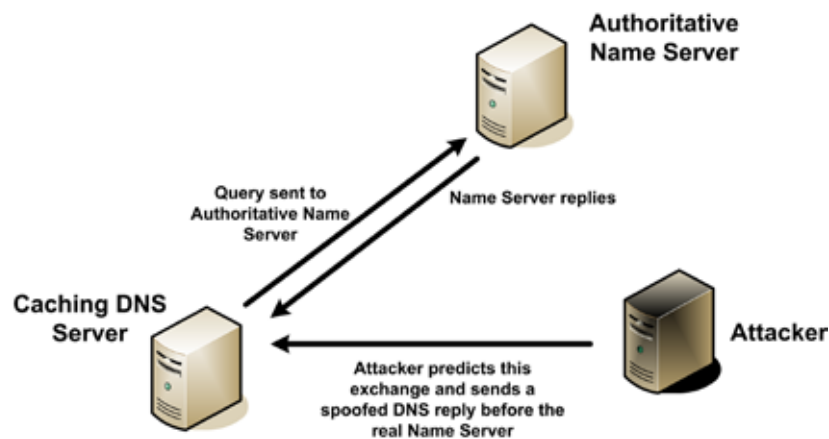


- (1) The attacker directs his botnet to begin sending DNS queries to the open recursive servers.
 - (2) These queries are sent using the (spoofed) source IP address of the intended victim name server and a source port of 53/UDP.
 - (3) The open recursive servers then issue their own DNS query to the authoritative name server for the DNS zone in question, in this case the attacker's previously compromised name server.
 - (4) The compromised name server sends a DNS response back.
 - (5) The open recursive servers send their own DNS response - and this goes to the spoofed IP address of the victim name server.
13. What makes this attack especially effective is that the attacker can create an initial DNS request which is very small, but which results in a much larger reply, thus creating an amplification effect. The victim server receives a flood of traffic which it did not request, and which has no distinct source, so is difficult to filter or block.

14. DNS amplification attacks have been used against internet root servers, top level domain operators and many other networks. It is also worth noting that many of the open recursive servers on the internet are home routers which are seldom patched (even if the manufacturer has released updated software) and likely to be open to abuse for many years to come.

DNS cache poisoning

15. The basic concept of cache poisoning:
 - (1) A client needs to make a DNS query so sends this to an upstream DNS server in the normal way and awaits a response.
 - (2) A malicious third party then sends a spoofed reply to the client before the real DNS server does.
 - (3) The client accepts this response as though it was genuine and caches it because there is no way for it to know that this didn't come from the expected source. In this way the malicious third party can 'poison' the DNS cache of the client with false information.



16. In order for an attacker to successfully deliver this attack he would need to be able to deduce some pieces of information about the interaction between the client and server: source and destination IP addresses, source and destination ports, and the DNS transaction ID being used. There are several papers available on the internet which describes how the necessary information can be acquired; it is not difficult, so a moderately skilled attacker could perform this attack with ease.

DNS covert channels

Tunnelling

17. DNS tunnelling has been known for several years; the basic concept is to move data in and out of a network by placing it inside DNS Resource Records. This can be achieved quite easily, albeit with some limitations on the packet sizes and transfer rates achievable.
18. The fact that DNS is relied upon in just about every network, coupled with the fact that it is an ideal vehicle in which to tunnel data, makes it a good choice for this activity. Network operators

and system administrators are not surprised by seeing many thousands of DNS queries for hundreds of different domain names on a daily basis, so if a small percentage of this total were actually being used to move sensitive documents out of a network it would be extremely difficult to spot.

19. The network administrator would need to enforce a policy which prevented DNS queries from being sent to any destination other than the local DNS server at which point the traffic could be inspected for anomalous queries (messages are larger and more frequent than usual) in order to mitigate the attack. Another approach would be to prevent the internal clients from performing their own queries by forcing them to use a proxy server; this would prevent DNS replies being returned to the clients. Even so, a stealthy attacker could still have a high probability of success moving data out of a network whilst evading detection.

Rebinding

20. This attack is not as well known as tunnelling, but can be just as effective at giving a remote attacker unauthorised access to resources on the internal segments of an organisation's network. The attack works by exploiting a weakness in web-browsers, or more specifically, in browser plugins (additional software components usually installed to provide a browser with added functionality).
21. Rebinding works by tricking the browser into thinking that a server on the outside also exists on the internal network, and turning the browser into a proxy between the two (this is achieved by manipulating the mapping between IP address and domain name). Connections can then be made to internal servers from outside the organisation and documents can be exfiltrated using those unauthorised connections.
22. Rebinding attacks against browsers were originally documented as long ago as 1996, and mitigation was introduced in the form of a feature called DNS pinning. For many years since, it has been assumed that the vulnerability which allows this attack was no longer exploitable. However, new vulnerabilities have been discovered which make the DNS rebinding attack viable unless mitigation is put in place to prevent it.

Application level attacks

23. Vulnerabilities within DNS software itself have been discovered regularly over a period of many years. The most dangerous type of vulnerability is one which results in a remotely exploitable bug known as a stack-based buffer overflow.
24. The vulnerability arises from the application developer not properly defining and filtering the input which a program expects to receive (in this case a DNS server application). This allows an attacker to send specially crafted data into the program which will result in an overflow of allocated memory in a special area called 'the stack'. When this occurs, the attacker can redirect the operation of the program to point at his own code, which then executes whatever instructions the attacker chooses. At this point the attacker could create his own 'backdoor' on the server, upload malicious programs, download username and account details from the server, delete logs or data, or many other unauthorised activities.

25. These vulnerabilities have been discovered in many different vendors' products. DNS software is present in many more locations than just the obvious DNS servers operated by enterprises and network operators. For example, almost without exception, wireless access points, routers, switches, firewalls and cable modems are all supplied with DNS software pre-installed. This means that network devices from the home user through to the largest enterprise can share many of the same vulnerabilities when it comes to DNS software.

REDUCING VULNERABILITY IN DNS

26. The following sections describe some of the basic elements of securing DNS. These ideas are not fully developed here, but will be expanded upon in other CPNI publications. The reader is also advised to explore the material under References/Further Reading.

Securing the server and operating system

27. Attacks against the DNS server and/or operating system on which it runs could lead to disruption or manipulation of the DNS application. It is therefore vital that the server, operating system and environment are deployed in a secure configuration. It is beyond the scope of this viewpoint to examine all the specifics of how this can be accomplished, and configurations will vary depending upon the choice of hardware and operating system. However, some general guidelines should be applied in all deployments:

Physical security must come first:

28. Server hardware should be secured against unauthorised access. Logical security mechanisms (firewalls/antivirus etc) cannot make up for poor physical security; if a malicious third party can get physical access to the hardware then exploitation or manipulation of the system will be possible.

Separation of functionality:

29. A DNS server is of such importance to the network that it should always be available. A good network design therefore separates this function onto specific servers which are responsible for providing DNS services only. Conversely, DNS services should be disabled or removed from systems which do not need them.

Operating system updates:

30. Whatever the operating system there will be possible issues; secure configuration and timely patching schedule are therefore essential. Service packs and security fixes should be applied swiftly at all times, but only after proper testing in a lab environment; never deploy software updates directly to the live production environment.
31. These measures are far from being comprehensive; network owners should look to the hardware and operating system vendors for specific advice regarding security of their products. Consider the use of application layer firewalls and Intrusion Detection Systems (IDS). Also, RFC

2827 (Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing) is a useful reference when deploying measures to protect DNS servers.

Securing the application

32. Application vulnerabilities (such as buffer overflows) in DNS software itself may also result in compromise of the system, there are many well known vulnerabilities and no doubt more will be discovered over time. Therefore always run the current version of software and ensure that all vendor supplied updates and security patches are applied. The following principles should be adopted:
- Adopt the principle of 'least privilege' when installing software. Do not run the DNS service with root or system privilege; if a compromise occurs the attacker will gain the same rights as the compromised process – in this case root/system.
 - Design the network so that DNS is split to handle internal and external queries separately. This avoids having to have the organisation's internal namespace held in a location which is accessible to hosts on the outside.
 - Disable open recursive queries.
 - Only allow zone transfer requests from trusted hosts.
 - Implement cryptographic techniques for authentication and integrity checking wherever possible e.g. DNSSEC or TSIG (see next section).
 - Check with software vendors for secure installation templates /recommendations.

Defending Against DoS

33. Most of the operators of the internet root servers have now deployed these critical servers using a technology called Anycast. The idea behind Anycast is that rather than just deploying a single physical server an operator can deploy many. These servers all run an 'instance' of the root DNS nameserver which they represent, and which will answer DNS queries originating in that geographical locality.
34. Anycast technology effectively means that there are now hundreds of internet root servers distributed around the globe, providing a high degree of resilience to attempted DoS attacks.
35. Other network owners also need to protect themselves from DoS attack. This can be accomplished by implementing network traffic filtering solutions at network borders, and by working with upstream providers to develop mitigations further away. A good working relationship with the upstream network provider is essential for responding to incidents and attacks when they occur.

Protecting the user of DNS services

36. A user of DNS services can be an organisation or an individual. The most important factor for users is that the information contained within DNS is accurate and trustworthy. In a security context this means mitigating cache poisoning attacks.
37. Cache poisoning attacks can be mitigated by the use of Domain Name System Security Extensions (DNSSEC), which adds data origin authentication and data integrity to the DNS.

DNSSEC employs new types of Resource Records together with public key cryptography to provide a much greater level of assurance to the user.

38. For enterprise users there are other solutions which can be implemented to help increase assurance in DNS services, such as Transaction Signature (TSIG). Similar to DNSSEC, TSIG also uses an extra RR to implement a cryptographic solution. The main difference from DNSSEC is that TSIG uses a symmetric encryption key, whereas DNSSEC implements public/private keys, so DNSSEC does not have to contend with the problem of how to distribute keys securely. For this reason, TSIG is most commonly seen on systems which share a common administrative control, its use being primarily to secure zone-transfers between DNS servers.
39. Organisations should also make sure that **all** their network equipment capable of providing DNS service is both correctly configured and updated with latest software fixes. In many cases this will simply mean securing weak default configurations and turning off services which are not required.
40. Similarly for home users, it is important to apply all the latest software updates from the vendor of the home router or access point.

CONCLUSIONS

41. DNS is fundamental to the normal operation of the internet as well as communications within private networks; therefore attacks against DNS (and DNS servers) are of concern to CPNI. The most critical attacks can disable DNS functionality effectively 'breaking' the network, or else sabotaging the data within DNS, manipulating it to the advantage of a malicious third party.
42. Work done recently appears to have largely been successful in mitigating DDoS attacks against the DNS infrastructure. The increasing numbers of root servers, combined with anycast technology, means that it would be a staggeringly difficult task to mount a successful DoS against the root servers directly. However, the threat of DDoS attack hangs over all network owners, so operators of DNS servers within the CNI should take steps to manage the impact that DDoS attack could have on their networks; this may need to be carried out in conjunction with service providers.
43. The large number of open recursive DNS servers available on the internet facilitates many of the DDoS attacks that are carried out. There is generally no requirement for these servers to be configured in this way, so the potential for DDoS could be vastly reduced if these servers were deployed in a more security conscious manner. This scenario is similar to the way in which a large population of unwitting home broadband users have failed to keep their PCs secure and unknowingly become part of the botnet problem, delivering spam email and phishing attacks to the wider world. Similarly, operators of insecure DNS servers may be contributing to the next DDoS event with their insecure configurations. User education combined with vendors supplying 'secure by default' software (and home routers) will be necessary to improve this situation.
44. There is good work being done with regard to mitigating the cache poisoning attacks which are possible at the moment. However, more support for DNSSEC is required from the internet community at large before this class of attack has a genuine, widely deployed defence. CPNI supports this effort and encourages zone operators to deploy DNSSEC as soon as they are able.
45. Finally, the imminent arrival of Next Generation Networks (NGNs) places even more emphasis on the need to deploy and configure network services in a secure manner. NGNs enable connectivity from any IP-enabled device to any other, including PCs, mobile phones, PDAs, Smartphones and other devices. The network itself will effectively be transparent to end users, who will simply become consumers of data. This development in network technology relies heavily on DNS, so a robust service is critical to the future success of NGNs.

REFERENCES / FURTHER READING

1. P. Mockapetris RFC1034, RFC1035
www.ietf.org/rfc/rfc1035.txt
2. ISC BIND
www.isc.org/sw/bind/
3. ICANN Report on DDoS
www.icann.org/committees/security/dns-ddos-advisory-31mar06.pdf
4. RFC3258:Distributing Authoritative Name Servers via Shared Unicast Addresses
<http://tools.ietf.org/html/rfc3258>
5. BIND 9 DNS Cache Poisoning – Amit Klein
www.trusteer.com/docs/bind9dns.html
6. RFC 2845: Secret Key Transaction Authentication for DNS (TSIG)
<ftp://ftp.rfc-editor.org/in-notes/rfc2845.txt>
7. RFC 2930: Secret Key Establishment for DNS (TKEY RR)
www.ietf.org/rfc/rfc2930.txt
8. DNS Security Extensions
www.dnssec.net/
9. RFC 4033: DNS Security Introduction and Requirements
www.ietf.org/rfc/rfc4033.txt?number=4033
10. CPNI: Botnets - the threat to Critical National Infrastructure
www.cpni.gov.uk/Docs/botnet_11a.pdf
11. Protecting Browsers from Rebinding Attacks
<http://crypto.stanford.edu/dns/dns-rebinding.pdf>
12. ICANN DNSSEC survey results
<http://ccnso.icann.org/surveys/dnssec-survey-report-2007.pdf>