DNS Security - Notes

EP2500/3200: Networked Systems Security

Panos Papadimitratos
Networked Systems Security Group
www.ee.kth.se/nss

2015-12-07

Contents

---

1 Intro (slides 2–21)

Questions about Domain Name System (DNS) vulnerability was raised as early as 1989 [1]. According to S. Bellovin (2004) [2],

> "The DNS remains a crucial weak spot in the Internet [3]."

DNS, as one of the main components of the functionality of Internet services, is used to translate the domain names to the numerical IP addresses. It has been assumed, often implicitly, that DNS query results are trusted; since the DNS results are used for other purposes, the impact of any existing vulnerability can be significant. The most apparent technical problem is that the DNS replies are not authenticated (Note: DNSSEC, which we will discuss later on, provides authenticated replies, but is not wildly used yet.). Moreover, the DNS Query ID field is only 16 bits (and in some older implementations it is not even randomized).

1.1 DNS Infrastructure (slides 3–6)

The DNS is a tree structured system, where you read the URL from right to left. This implies that the DNS requests are handled in a recursive manner. IETF RFC 1591 addressed that (slide 3):

> "In the Domain Name System (DNS) naming of computers there is a hierarchy of names. The root of system is unnamed. There are a set of what are called "top-level domain names" (TLDs). These are the generic TLDs (EDU, COM, NET, ORG, GOV, MIL, and INT), and the two letter country codes from ISO-3166."

The Internet Assigned Numbers Authority (IANA) is responsible for the overall coordination and management of the DNS, and especially the delegation of portions of the name space called top-level domains. The structure of authorities, including the second-level domains and sub-domain names, are managed by organizations or individuals.

The DNS zones are jurisdictions delegated to a DNS server. As shown in slide 5, when a new zone (KTH) for a sub-domain is created, delegation from the parent zone (SE) is needed. In this way, the .se Name Server (NS) trusts the .kth.se NS to resolve *.kth.se. ([4] among the recommended textbooks has sections on DNS.)

1.2 A resolution example (slides 7–11)

This name tree is traversed from the closest node upwards to the root (if necessary) when a query for an URL is sent. The images are mostly from unixwiz.net[1], where details of the example can be found.

1.3 A DNS query (slides 12–16)

Details can again be found on the web.[2] The important part is that the DNS Data only contains a 16-bit Query ID that links DNS queries to DNS responses.

"Referrals" contain the next DNS server that might know what we are looking for.

[1] `http://unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html#simple`
[2] `http://unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html#packet`

1.4 Checking the response (slides 17–21)

For the DNS client to verify the response, it checks several fields/sections that:

- The response is sent to the same UDP port from which it was sent out it (more on that later in Sec. 2.2).
- The Query ID matches a pending query; otherwise, it is simply dropped.
- The Question section is a duplicate of what we asked.
- The response comes from the same domain as the query. This is sometimes called *bailiwick checking*.

The checks imply that a response that fails any of them can be fraudulent.

The DNS Records contain a TTL parameter telling the DNS cache how long the response should be stored. This can be exploited, as shown later (see section 2.3), to mount an attack.

2 Attacks (slides 22–38)

A recommended reading is the IETF RFC3833 "Threat Analysis of the Domain Name System (DNS)" [5].

Most PKI structures rely on DNS to verify if a user has a given domain name; this is the case for certificates used for SSL. "... many Certificate Authorities validate a user's control over a domain by sending email"[3].

2.1 MitM (slides 24–27)

It is clear that plain DNS is vulnerable to so-called man-in-the-middle attacks. A 2007 easy-to-read overview paper [6] explains this.

DNS is commonly used as the means for lawful blocking of web pages. Tech-savvy users could use of course an alternative DNS service[4] or obtain the IP addresses in another way. But for the large volume of users, controlling the DNS entries puts a stop to a large number of requests and thus accesses to sites (that are to be blocked).

In Sweden, several ISPs voluntarily implemented a DNS block on a list provided by the police, which they say contains child pornography[5]. There have been some controversy concerning the process and the possible misuse of censorship.

In January of 2012, a self-imposed day of blackout was used by Wikipedia and other sites to protest upcoming legislation in the US[6] that would enforce DNS blocking of sites accused of copyright infringement.

2.2 DNS Cache Poisoning and countermeasyre (slides 28–33)

The Kaminsky[7] DNS vulnerability allows the attacker to basically brute force DNS replies to provoked queries. DNSSEC can thwart such attacks. While waiting for it

[3] `http://unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html#summary`
[4] Like `www.opendns.com`.
[5] `http://www.polisen.se/sv/Om-polisen/Sa-arbetar-Polisen/Olika-typer-av-brott/Sexuella-overgrepp-mot-barn-och-barnpornografi/`
[6] `http://en.wikipedia.org/wiki/Protests_against_SOPA_and_PIPA`
[7] `http://unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html#poisoning`

to be very broadly deployed, a way to mitigate the attack is to randomize the query ID and source port, i.e., having 27-bit long random entries (rather than 16-bit long ones.)

The goal of the attack is to intercept a legitimate DNS query. The attacker could either form this query while being on the network she wants to attack, or have a machine in the targeted network run some code. Either way, the attacker could be made aware of the ongoing query. Once the attacker knows that there is a query from a local DNS server (with an associated cache), she tries to brute force responses to it, each with a different query ID. The ID space is just 16 bits, so it is likely to craft a faulty response with a matching ID, especially if the attacker has a topological advantage. If the attack fails, the attacker can simply try again with a different host (host name). The Wikipedia article for DNS Cache Poisoning[8] links to a video[9] that explains the attack.

The way to mitigate such attacks is to use a larger ID space, notably using additional available space in the DNS packets. An additional 11 bits (Slide 33) can increase the size of the ID space to 2^{27} possible IDs.

2.3 DNS Rebinding Attack (slides 34–38)

The DNS Rebinding Attack[10] [7] poses another threat to DNS. A fraudulent DNS server can serve query replies with a short time to live, helping a client-side java script (or similar piece of software) to circumvent the same-origin policy.[11] Essentially, the attacker creates an own web server, it attracts victims to visit, and when the DNS resolution takes place, the attacker gives an authoritative yet short-lived (low TTL) response. It also feeds the unsuspected client with a malicious piece of software (e.g., a java script). Upon expiration, and a new request from the script, now the attacker gives an IP that is a machine inside the victim's network as they both correspond to the same origin (name-wise). But within the same origin, scripts can read write content and network resources using HTTP. What does this mean? If there is firewall that prevents access to a machine, the attacker can now, thanks to the rebinding, access it even though it is outside the firewall, leveraging the client's browser. Alternatively, the re-binding can lead to IP address hijacking and then perpetrate other attacks, e.g., click fraud. We will revisit these aspects in the web security module. Question: Does authentication (DNSSEC, coming next) thwart rebinding attacks?

3 DNSSEC (slides 39–44)

The *Domain Name System Security Extensions* (DNSSEC) try to address the most pressing DNS problem by signing all response data fields. DNSSEC provides only authentication, not confidentiality. It also inherits issues/problems that may emerge with the reliance on a PKI. For example, if an attacker can manipulate the clock of a host, she can make the victim use revoked certificates.

3.1 Zone Walking (slide 41–43)

Zone walking (aka zone enumeration) is a side effect of DNSSEC. Although the main idea of DNSSEC is to enhance the security of DNS, it allows exposure of information

[8] `http://en.wikipedia.org/wiki/DNS_cache_poisoning`
[9] `http://www.checkpoint.com/defense/advisories/public/dnsvideo/index.html`
[10] `http://crypto.stanford.edu/dns/`
[11] `http://en.wikipedia.org/wiki/Same-origin_policy`

that would otherwise not be necessarily accessible.

A relatively newly introduced feature the "next secure" (NSEC) record, including all Resource Records (RRs) in the zone, allows any resolver to control for/verify non-existent RRs. In such do-not-exist responses, the next owner name is provided. But this information can be used by an aggressive querier to enumerate/list all resources within a zone/system. An alternative scheme that provides denial of existence is NSEC3 [8]: the main difference from NSEC is that names are hashed, salted hashes (note: the salt cannot be unknown, it can be obtained), which keep the names hidden. An aggressive querier could still get lists of hashed names, but it would need to spend a significant effort to obtain the clear-text names.

3.2 Deployment (slide 44)

At one of the ICANN meetings, many engineers from some of the early-adopting registries gathered for their regular face-to-face discussion about how to break the "chicken or egg" problems of secure domain name deployment:

> "While the global roll out of Domain Name System Security Extensions (DNSSEC) continues at the domain name registry level — with more than 25% of top-level domains now signed — the industry continues to focus on the problem of registrar, ISP and ultimately end-user adoption."[12]

The deployment continues, maps are available at:
`http://www.internetsociety.org/deploy360/dnssec/maps/`

4 Extra reading

- RFC3833: Threat Analysis of the Domain Name System (DNS)
- RFC4033: DNS Security Introduction and Requirements
- RFC4034: Resource Records for the DNS Security Extension
- An Illustrated Guide to the Kaminsky DNS Vulnerability[13]

References

[1] S. M. Bellovin, "Security problems in the TCP/IP protocol suite," *Computer Communications Review*, vol. 19, no. 2, pp. 32–48, April 1989. [Online]. Available: https://www.cs.columbia.edu/~smb/papers/ipext.pdf

[2] ——, "A look back at "Security problems in the TCP/IP protocol suite"," in *Annual Computer Security Applications Conference*, December 2004, invited paper. [Online]. Available: https://www.cs.columbia.edu/~smb/papers/acsac-ipext.pdf

[3] F. B. Schneider, *Trust in Cyberspace*. The National Academies Press, 1999.

[4] W. Cheswick, S. Bellovin, and A. Rubin, *Firewalls and Internet security: repelling the wily hacker*, ser. Addison-Wesley professional computing series. Addison-Wesley, 2003.

[12] `http://www.circleid.com/posts/20111130_dnssec_update_from_icann_42_in_dakar/`
[13] `http://unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html`

[5] D. Atkins and R. Austein, "Threat Analysis of the Domain Name System (DNS)," RFC 3833 (Informational), Internet Engineering Task Force, Aug. 2004. [Online]. Available: http://www.ietf.org/rfc/rfc3833.txt

[6] S. Ariyapperuma and C. Mitchell, "Security vulnerabilities in dns and dnssec," in *Availability, Reliability and Security, 2007. ARES 2007. The Second International Conference on*, 2007, pp. 335–342.

[7] C. Jackson, A. Barth, A. Bortz, W. Shao, and D. Boneh, "Protecting browsers from dns rebinding attacks," *ACM Transactions on the Web (TWEB)*, vol. 3, no. 1, p. 2, 2009.

[8] J. Bau and J. C. Mitchell, "A security evaluation of dnssec with nsec3," in *Network and Distributed Systems Security (NDSS) Symposium. The Internet Society*, 2010.