

Dig for dummies

Explaining an highly useful network tool

Willem L. Middelkoop

May 24, 2019



```
willem@base:~$ dig willem.com

;<<<> DiG 9.11.5-P4-1-Debian <<<> willem.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 28667
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;willem.com.                IN
;; ANSWER SECTION:
willem.com.                1245  IN  A      87.253.135.162
;; AUTHORITY SECTION:
willem.com.                1808  IN  NS     ns1.lemmid.com.
willem.com.                1808  IN  NS     ns2.lemmid.com.

;; Query time: 2 msec
;; SERVER: 80.69.66.67#53(80.69.66.67)
;; WHEN: Fri May 24 12:25:44 CEST 2019
;; MSG SIZE rcvd: 98

willem@base:~$ dig +short willem.com
87.253.135.162
willem@base:~$ dig +short willem.com MX
0 online.lemmid.com.
willem@base:~$ dig +short willem.com NS
ns2.lemmid.com.
ns1.lemmid.com.
willem@base:~$ dig +short willem.com TXT
"v=spf1 a:online.lemmid.com a:transfer.lemmid.com -all"
willem@base:~$
```

[willem] 0:dig for dummies* "base" 12:26 24-May-19

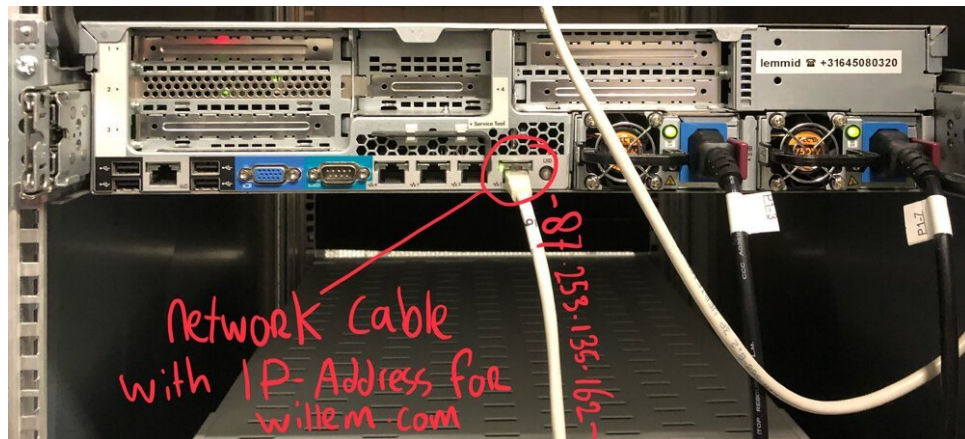
When you're building websites, apps or email services you may run into domain names and their configurations. When everything is working as it should, most of this is invisible. But when troubleshooting a domain name configuration, it may be necessary to dig a little deeper... read along to learn how!

Domain Name System (DNS)

The internet works using numerical internet protocol (IP) addresses as means to locate and to identify online computers, (cloud)services and devices. Instead of memorising all these IP-addresses, the domain name system has become the essential world wide directory service, linking names to numbers.

You're reading this on **willem.com**, but in reality this domain name is simply a pointer for a physical machine, connected to the internet (with wires, really!). One of

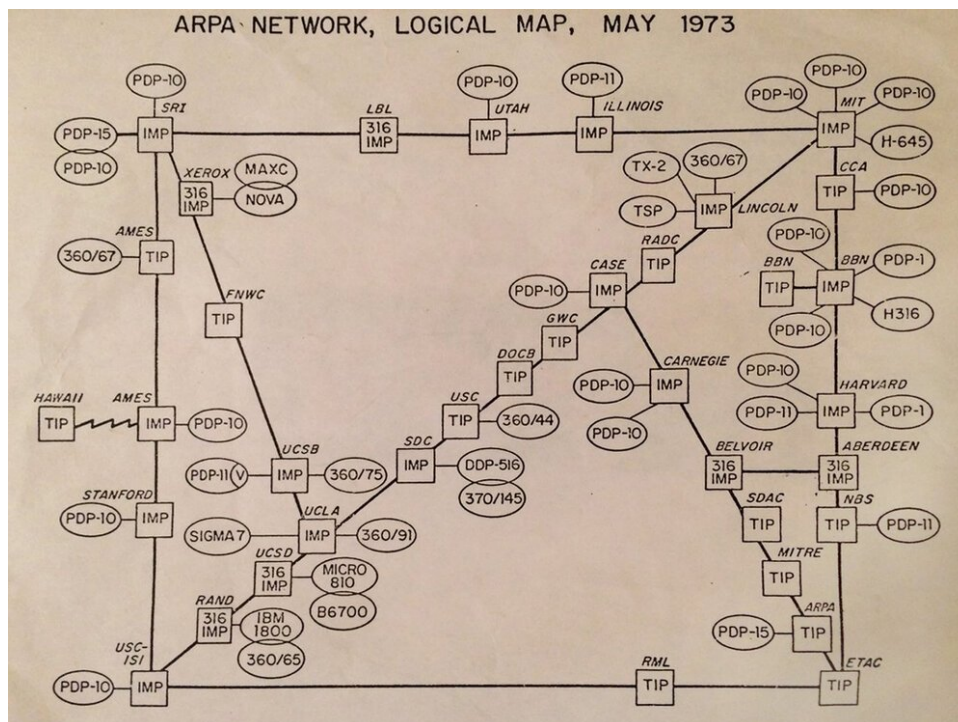
those wires is linked to IP-address **87.253.135.162**. It's the responsibility of the DNS server to keep the correct (IP-address) numbers matching to the (domain) names.



The Domain Name System (DNS) links domain names to IP-addresses, which in turn are routed through cables. You're reading this - seriously! - through the cable connected to the server shown on this photo!

Distributed and decentralised

When the Americans designed their ARPANET in the early seventies, one of their military design requirements was that the network would survive attacks (by enemies). They accomplished this by implementing packet switching, a way to share and reuse network cables by multiple people in multiple ways. If one part of the network gets damaged, packets of information are rerouted over other parts of the network.



A map of the ARPANET in 1973... imaging mapping the internet today! (Public domain)

To achieve maximum survivability, there is no single root DNS server. Instead, the domain name system delegates responsibility of assigning domain names and mappings to those names by designating an authoritative name server for each domain name.

This is why there are many different DNS-servers and configurations. With so many ways and places that things can go wrong, you need a tool to dig around.

dig (domain information groper)

The 'dig'-command is a tool for questioning DNS nameservers for information about IP-addresses, hostnames, mail servers and other kinds of network settings. The dig-command is available on Unix, macOS, GNU/Linux and Windows.

Using dig to query domain name servers

You can use dig to get information about a given domain name by simply typing: 'dig willem.com'. See the following screenshot with explanation:

```
willem@base:~$ dig willem.com
;<<>> Dig 9.11.5-P4-1-Debian <<>> willem.com
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 28667
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:;; udp: 4096
;; QUESTION SECTION:
;willem.com.                IN      A
;; ANSWER SECTION:
willem.com.                 1245    IN      A      87.253.135.162
;; AUTHORITY SECTION:
willem.com.                 1808    IN      NS      ns1.lemmid.com.
willem.com.                 1808    IN      NS      ns2.lemmid.com.

;; Query time: 2 msec
;; SERVER: 80.69.66.67#53(80.69.66.67)
;; WHEN: Fri May 24 12:25:44 CEST 2019
;; MSG SIZE rcvd: 98

willem@base:~$ dig +short willem.com
87.253.135.162
willem@base:~$ dig +short willem.com MX
0 online.lemmid.com.
willem@base:~$ dig +short willem.com NS
ns2.lemmid.com.
ns1.lemmid.com.
willem@base:~$ dig +short willem.com TXT
"v=spf1 a:online.lemmid.com a:transfer.lemmid.com -all"
willem@base:~$
```

Using dig to get information about a domain name, willem.com

- 1) **dig willem.com** will return all information about the question, answer, authority (the one that answered the question) and statistics of the actual query (like how long it took to get an answer).
- 2) **dig +short willem.com** omits all the extra information and only returns the answer. Here you can see that **willem.com** points to IP-address **87.253.135.162**.
- 3) **dig +short willem.com MX** will show you the mailer exchange (MX) records. MX-records are crucial for email to arrive at the right server. Basically it's the designation of the post office responsible for email linked to the domain name. For **willem.com** this is a mail services called **online.lemmid.com**. Other popular mail services are Office365 and Google Gmail. Using dig you can find out what mail service somebody is using.

- 4) **dig +short willem.com NS** will show you the name servers that are responsible for handling queries for this domain name. Usually these are the servers operated by the domain name registry, your domain name provider. Using dig you can find what kind of domain name provider somebody is using (in the case of **willem.com**, that's **Lemmid**).
- 5) **dig +short willem.com TXT** will return the text record linked to the domain name, this is where you'll find the so called SPF-record. The SPF-record is another crucial part of email, its useful to dig a little deeper into SPF.

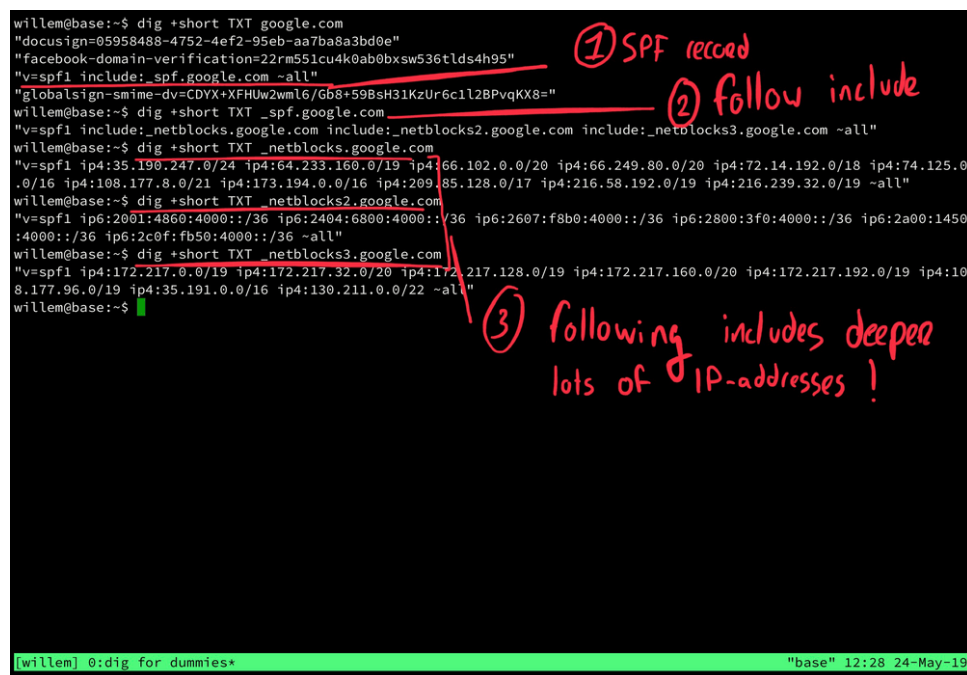
Sender Policy Framework (SPF)

To detect forged sender addresses in emails (so called email spoofing), the SPF standard was defined in 2004. Originally called "Sender Permitted From", SPF is a way to check if somebody is allowed to send email on behalf of a given domain name. SPF allows the owner of a domain name to specify which computers are authorised to send email with FROM addresses of that domain.

If a domain name owner publishes a SPF-record, spammers and phishers are less likely to (ab)use the domain name to send forged emails, pretending to be from that domain. An SPF-protected domain is therefore less likely to be blacklisted by spam filters, making it more likely to allow legitimate email from the domain to get through. A misconfigured SPF-record can however disrupt email delivery.

Using dig to query SPF-records

Using the dig-command you can query a SPF-record and see what the exact sender policy is. See the following screenshot with explanation:



```
willem@base:~$ dig +short TXT google.com
"docusign=05958488-4752-4ef2-95eb-aa7ba8a3bd0e"
"facebook-domain-verification=22rm551cu4k0ab0bxsxw536tlds4h95"
"v=spf1 include:_spf.google.com ~all"
"globalsign-smime-dv=CDYX+XFHUw2wml6/Gb8+598sH31KzUr6c1l2BPvqKX8="
willem@base:~$ dig +short TXT _spf.google.com
"v=spf1 include:_netblocks.google.com include:_netblocks2.google.com include:_netblocks3.google.com ~all"
willem@base:~$ dig +short TXT _netblocks.google.com
"v=spf1 ip4:35.190.247.0/24 ip4:64.233.160.0/19 ip4:66.102.0.0/20 ip4:66.249.80.0/20 ip4:72.14.192.0/18 ip4:74.125.0.0/16 ip4:108.177.8.0/21 ip4:173.194.0.0/16 ip4:209.85.128.0/17 ip4:216.58.192.0/19 ip4:216.239.32.0/19 ~all"
willem@base:~$ dig +short TXT _netblocks2.google.com
"v=spf1 ip6:2001:4860:4000::/36 ip6:2404:6800:4000::/36 ip6:2607:f8b0:4000::/36 ip6:2800:3f0:4000::/36 ip6:2a00:1450:4000::/36 ip6:2c0f:fb50:4000::/36 ~all"
willem@base:~$ dig +short TXT _netblocks3.google.com
"v=spf1 ip4:172.217.0.0/19 ip4:172.217.32.0/20 ip4:172.217.128.0/19 ip4:172.217.160.0/20 ip4:172.217.192.0/19 ip4:108.177.96.0/19 ip4:35.191.0.0/16 ip4:130.211.0.0/22 ~all"
willem@base:~$
```

① SPF record

② follow include

③ following includes deeper lots of IP-addresses!

[willem] 0:dig for dummies*

"base" 12:28 24-May-19

Using dig to get information about the SPF-records for a domain name, google.com

- 1) **dig +short TXT google.com** will show us the TXT-record linked configured for google.com, the place where you can find the SPF-record (among other things).

The SPF-record for **google.com** is "**v=spf1 include:_spf.google.com ~all**". The **include:-**part means that for this domain name external SPF values should be loaded, too.

- 2) **dig +short TXT _spf.google.com** shows what those external SPF values for **google.com** actually are. In this case it reveals another three includes: "**_netblocks.google.com**", "**_netblocks2.google.com**" and (how original..) "**_netblocks3.google.com**".
- 3) **dig +short TXT _netblocks.google.com** (and the 2 + 3 variants) specify the actual IP-addresses that are allowed to send email on behalf of **google.com**. As Google is a big internet company there are many IP-addresses in these lists. If you look carefully, you'll notice that some of these addresses have a "/"-suffix, like "/24". This is the way to notate an entire block (e.g. street) of IP-addresses.

IP blocks explained

The easiest way to understand IP-address block notation is by the following examples:

- **/8 = 255.0.0.0**
- **/16 = 255.255.0.0**
- **/24 = 255.255.255.0**
- **/32 = 255.255.255.255**

Where you'll see '0' in the addresses above, other values are included. So, for instance, the 173.194.0.0/16 actually means "all the addresses starting with '173.194' ". Given 255 possible positions on each octet, that is 255x255=65025 different addresses!

Conclusion

To solve problems you need to know where the problem is. Using the dig-command you can learn about a given network and domain configuration.

Answers from one command to another, enable you to dig deep into the innards of the amazing interwebs. Good luck!