

dns-guide-answers

Practice Domain: christian.io

Lab/Homework Exercise Answers and Explanations

Commands needed:

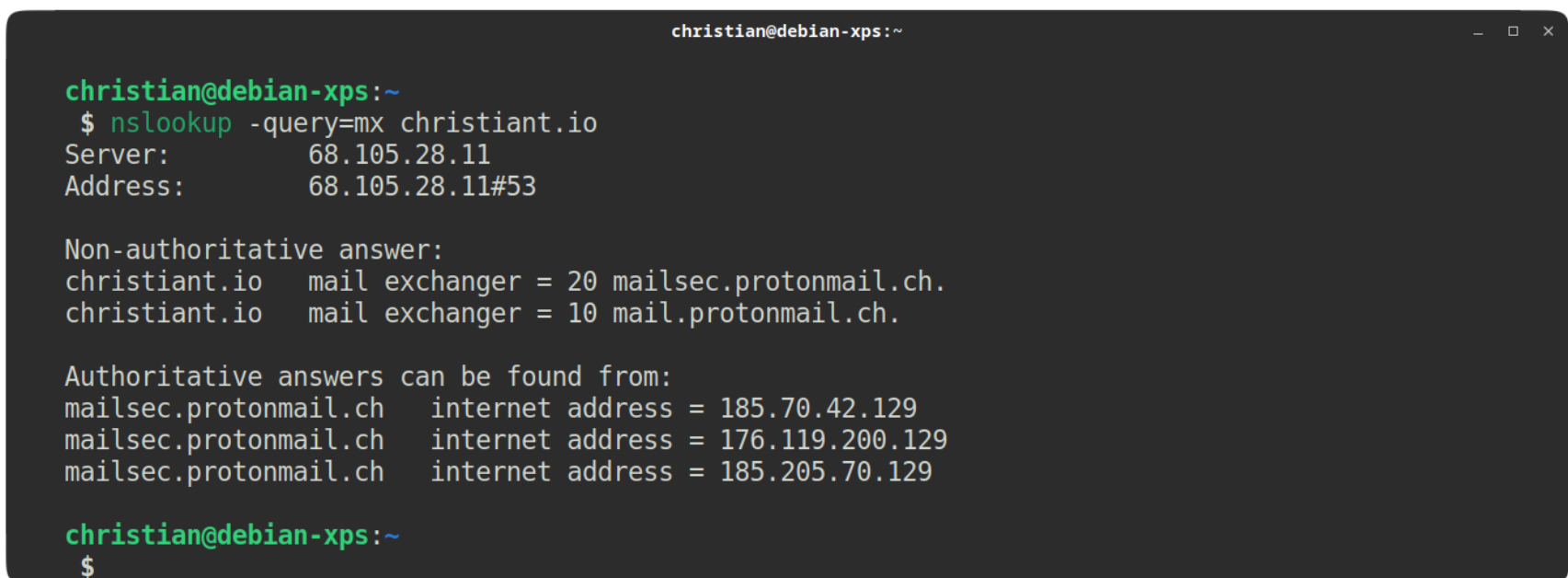
1. nslookup -type=mx christian.io
2. nslookup -type=mx caffeinatedsheep.com
3. nslookup -type=txt christian.io
4. nslookup -type=txt caffeinatedsheep.com
5. Google / OSINT
6. nslookup protonmail._domainkey.christian.io
nslookup protonmail._domainkey.caffeinatedsheep.com
7. nslookup -type=txt _dmarc.christian.io
nslookup -type=txt _dmarc.caffeinatedsheep.com

1. What service does christian.io use for mail exchange?

To answer this question we are looking for the Mail Exchange record. This will point to where mail for this domain should be sent. Often the records provide the name of the service.

Note: If only an IP was provided or records that do not indicate service, passive scanning tools like shodan.io or active scanning may be required to determine email service.

```
~$ nslookup -query=mx christian.io
```



```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -query=mx christian.io  
Server:        68.105.28.11  
Address:      68.105.28.11#53  
  
Non-authoritative answer:  
christian.io  mail exchanger = 20 mailsec.protonmail.ch.  
christian.io  mail exchanger = 10 mail.protonmail.ch.  
  
Authoritative answers can be found from:  
mailsec.protonmail.ch  internet address = 185.70.42.129  
mailsec.protonmail.ch  internet address = 176.119.200.129  
mailsec.protonmail.ch  internet address = 185.205.70.129  
  
christian@debian-xps:~  
$
```

In this case, we see `Protonmail.ch` is providing email exchange services. Googling this will tell us even more about Proton. But we have the information we need to move on.

ANSWER: PROTONMAIL

2. Which domain has two mail exchange records, christian.io or caffeinatedsheep.com?

We can clearly see from our output that christian.io records two Mail Exchange servers: mailsec.protonmail.ch and the higher priority mail.protonmail.ch. Lets check the mx record for caffeinatedsheep.com to be certain.

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -type=mx caffeinatedsheep.com  
Server:          68.105.28.11  
Address:         68.105.28.11#53  
  
Non-authoritative answer:  
caffeinatedsheep.com mail exchanger = 10 mail.protonmail.ch.  
  
Authoritative answers can be found from:  
mail.protonmail.ch internet address = 185.70.42.128  
mail.protonmail.ch internet address = 185.205.70.128  
mail.protonmail.ch internet address = 176.119.200.128  
  
christian@debian-xps:~  
$ _
```

The domain christian.io is the only domain with record of two servers.

ANSWER: CHRISTIAN . IO

3. According to SPF, who is allowed to send mail for christian.io?

As the question reminds us, SPF or Sender Policy Framework records tell us who is allowed to send mail for a domain. We can find the `spf` record in the `txt` record for a domain.

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -query=txt christian.io  
Server:          68.105.28.11  
Address:         68.105.28.11#53  
  
Non-authoritative answer:  
christian.io    text = "v=spf1 include:_spf.protonmail.ch mx ~all"  
christian.io    text = "protonmail-verification=07059ca1b33d3d1994b  
22621d13009c60f4c801d"  
  
Authoritative answers can be found from:  
  
christian@debian-xps:~  
$ _
```

Here you can see protonmail.ch is included in the record and is therefore permitted to send email on behalf of this domain.

Answer: `protonmail.ch` or Proton Mail

4. According to SPF, who is allowed to send mail for caffeinatedsheep.com?

We can determine this by checking the `spf` record for `caffeinatedsheep.com`.

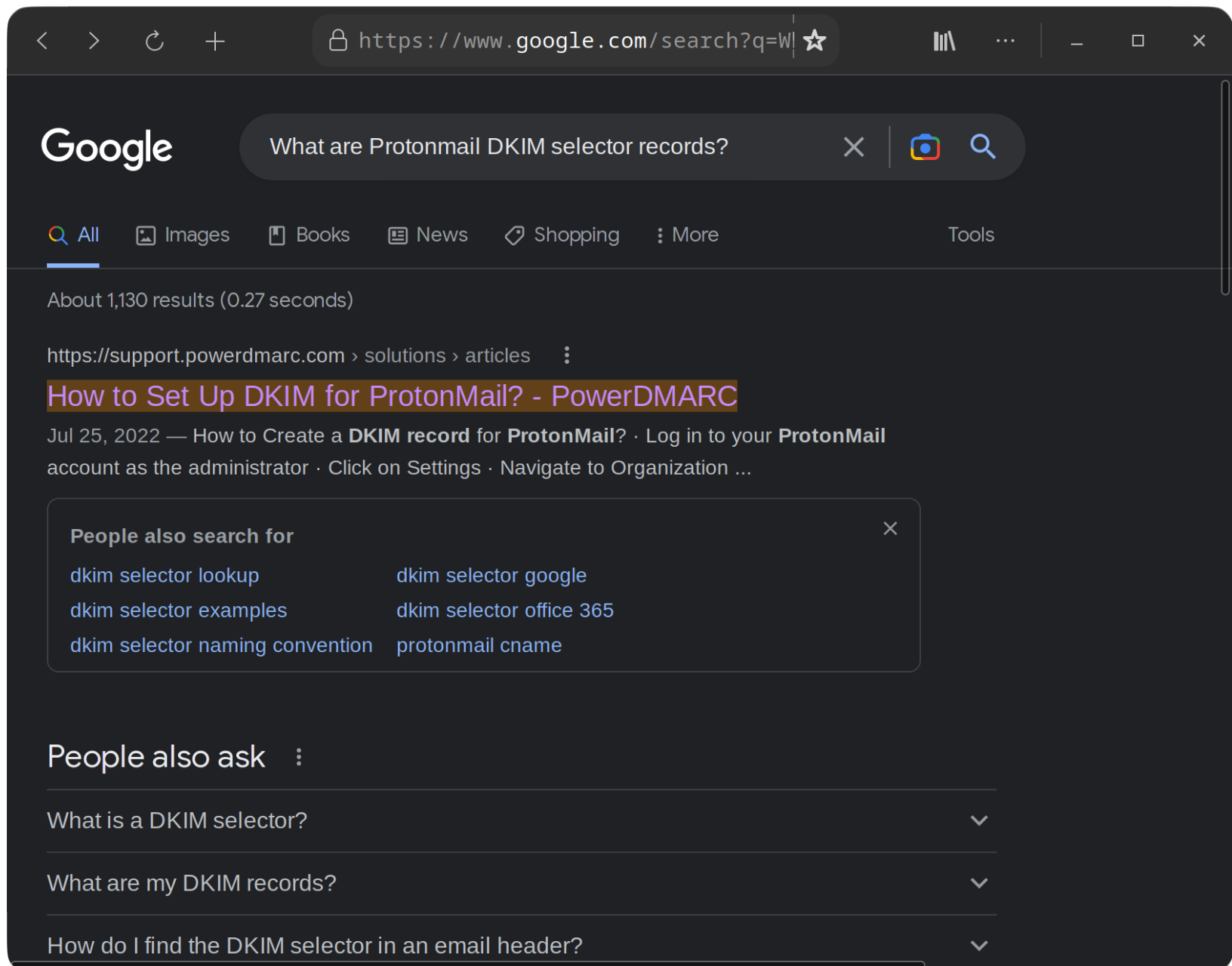
```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -type=txt caffeinatedsheep.com  
Server:          68.105.28.11  
Address:         68.105.28.11#53  
  
Non-authoritative answer:  
caffeinatedsheep.com    text = "v=spf1 include:_spf.prot  
onmail.ch mx ~all"  
  
Authoritative answers can be found from:  
  
christian@debian-xps:~  
$ _
```

ANSWER: ALSO `PROTONMAIL.CH` OR `PROTON MAIL`







5. What are Protonmail's DKIM selector records?

As discussed and can be found on Google, DKIM (or Domain Keys Identified Mail) is a method to sign an email to verify authenticity. This allows recipients to detect forged (spoofed) sender addresses, a common technique involved in phishing. This key signs each outgoing email with a digital signature that is added to the header of an email. A public DNS record hosts a key that can be used by recipients to verify the included DKIM signature in the header of an email.

To find the selector record that Proton uses, let us do a bit of research and Google the question. Many sites may record the DKIM selector records (which are CNAME records) that are used by Proton. We will just look at the first site first



Please add all 3 of the following CNAME records. Note, DNS records can take several hours to update.

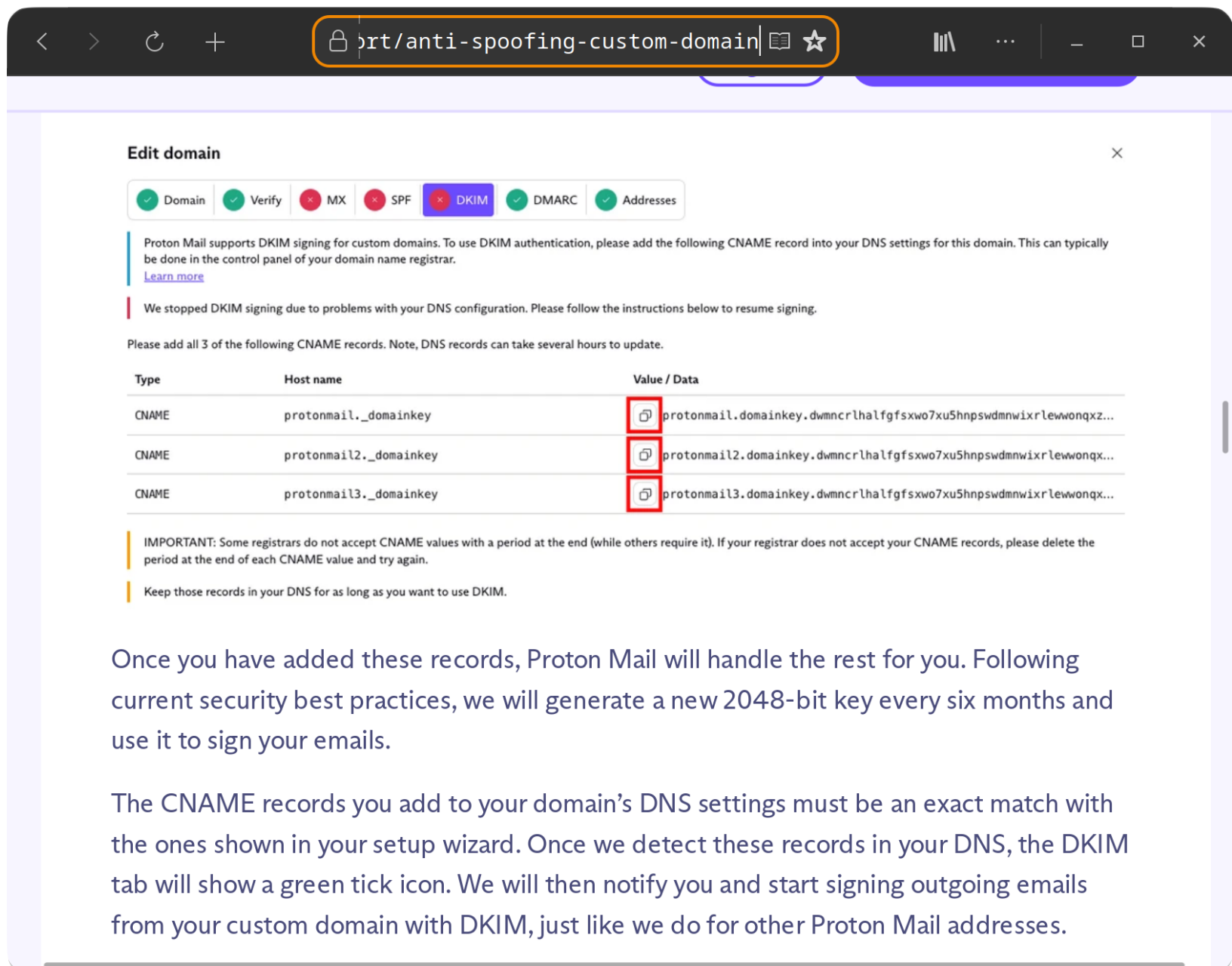
TYPE	HOST NAME	VALUE / DATA
CNAME	protonmail._domainkey	 
CNAME	protonmail2._domainkey	 
CNAME	protonmail3._domainkey	 

Note: The records shown above are merely an example. I generated by ProtonMail for your domain.

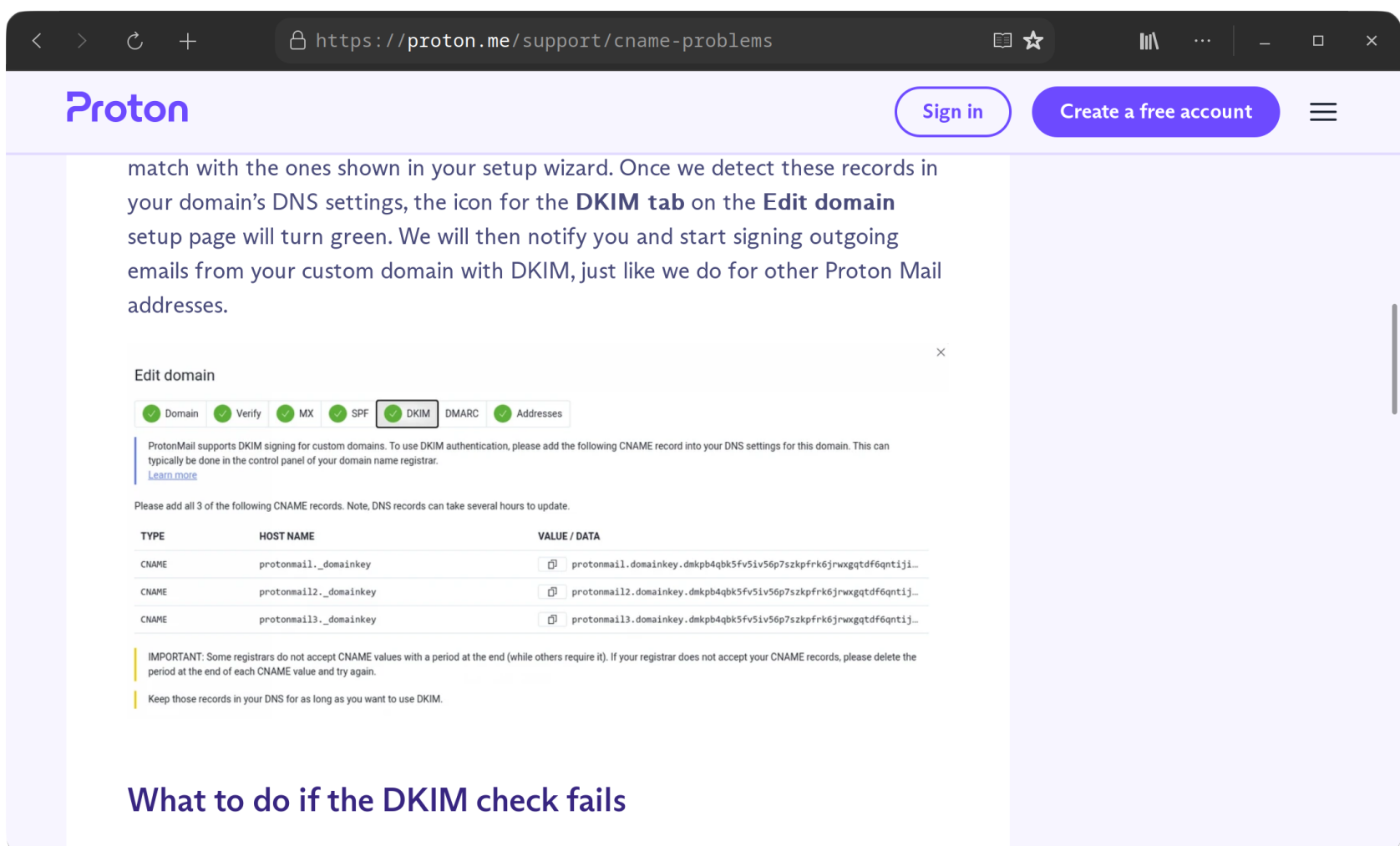
To check your DKIM record, use our [DKIM record lookup t](#)

At this site, we can see that Protonmail's selector's for DKIM are: `protonmail._domainkey`, `protonmail._domainkey2`, and `protonmail._domainkey3`.

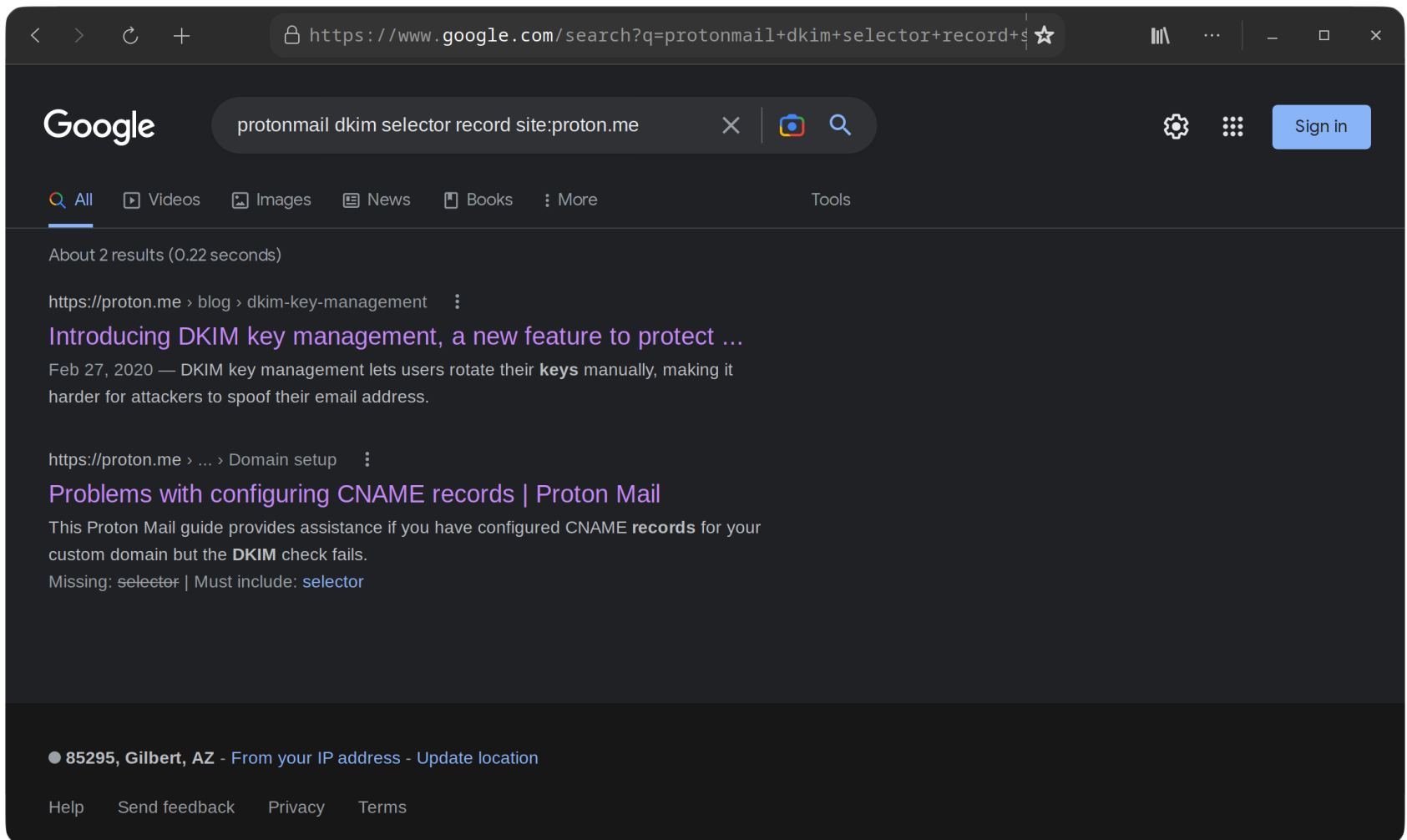
During the lecture portion, we mentioned that the email service documentation will include the selectors. If we search for "protonmail dkim" in DuckDuckGo or Google, we can find Proton's support page on [anti-spoofing-domain-configuration](#).



We can also search something like "protonmail dkim selector record" in Google we can find the following guide [CNAME Records for DKIM](#).



Pro-tip: We can use 'site:proton.me' in our Google search to limit the search to just Proton's documentation.



ANSWER: ALL OF THESE SITES PROVIDE THE SAME ANSWER: PROTONMAIL._DOMAINKEY, PROTONMAIL._DOMAINKEY2, PROTONMAIL._DOMAINKEY3.

6. Does chrtistian.io or caffeinatedsheep.com use DKIM signing?

At this point we know both christian.io and caffeinatedsheep.com use ProtonMail for mail exchange. We can check this again though using the `mx` records.

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -type=mx christian.io  
Server:          68.105.28.11  
Address:         68.105.28.11#53  
  
Non-authoritative answer:  
christian.io    mail exchanger = 10 mail.protonmail.ch.  
christian.io    mail exchanger = 20 mailsec.protonmail.ch.  
  
Authoritative answers can be found from:  
mail.protonmail.ch    internet address = 185.205.70.128  
mail.protonmail.ch    internet address = 185.70.42.128  
mail.protonmail.ch    internet address = 176.119.200.128  
  
christian@debian-xps:~  
$ nslookup -type=mx caffeinatedsheep.com  
Server:          68.105.28.11  
Address:         68.105.28.11#53  
  
Non-authoritative answer:  
caffeinatedsheep.com  mail exchanger = 10 mail.protonmai  
l.ch.  
  
Authoritative answers can be found from:  
mail.protonmail.ch    internet address = 185.205.70.128  
mail.protonmail.ch    internet address = 185.70.42.128  
mail.protonmail.ch    internet address = 176.119.200.128  
  
christian@debian-xps:~  
$
```

We also know that Proton Mail keeps DKIM records at `protonmail._domainkey`, `protonmail._domainkey2`, `protonmail._domainkey3` to be used for verifying DKIM signatures. Query the record for one or all of these records for each domain (one is sufficient to verify if it

used or not).

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup protonmail._domainkey.caffeinatedsheep.com  
Server:          68.105.28.11  
Address:         68.105.28.11#53  
  
** server can't find protonmail._domainkey.caffeinatedsheep.com: NXDOMAIN  
  
christian@debian-xps:~  
$ nslookup protonmail._domainkey.christiant.io  
Server:          68.105.28.11  
Address:         68.105.28.11#53  
  
Non-authoritative answer:  
protonmail._domainkey.christiant.io      canonical name = protonmail.domainkey.dbrmn6rjajwbu5rnepocowlvbchyyfyr5atcxuvgtvvl2cddbtsq.domains.proton.ch.  
  
christian@debian-xps:~  
$ _
```

ANSWER: CHRISTIANT.IO HAS RECORDS FOR DKIM. THE CAFFEINATEDSHEEP.COM DOMAIN HAS NO SUCH RECORDS.

Pro-tip: The CNAME points to where the key is hosted. If you query the txt record of the canonical name you can view the key used to verify the signature. This isn't used for this exercise; however.

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -type=txt protonmail.domainkey.dbrmn6rjajwbu5rnepocowlvbchyyfyr5atcxuvgtvvl2cddbtsq.domains.proton.ch  
;; Truncated, retrying in TCP mode.  
Server:          68.105.28.11  
Address:         68.105.28.11#53  
  
Non-authoritative answer:  
protonmail.domainkey.dbrmn6rjajwbu5rnepocowlvbchyyfyr5atcxuvgtvvl2cddbtsq.domains.proton.ch      text = "v=DKIM1;k=rsa;p=MIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEA62vxwue0FXs2B8ij9rE00mvM/PjdX8XEYz++LOSBA2u/4twCaEb0I3ennQI0qgIuHTjeuRCMLyMSYh9GU/zDnfv/8EKufZsntiU6I/bK83cvmV9IRFYhlqAkh0y0iYAoiwTGN4Yo3klUdJfa3L7tSm+WjrPVX/NgGPYWSCrsT+N6yrV9FaEMvl0indCAj/AaTa" "bfJJysLJt6mQn+3910qRj1FFMACSDk+0ESaUDGwV/5RNuED0q8kWm4DxMH2/PJAktmQ05boscfm6f7Jg3ATXyeLkWNHaNre9cr+aUAGAoCTGUHIPDi00nLMHDDhXRUHxGATo7fG7LyaefGRd2wIDAQAB;"  
  
Authoritative answers can be found from:  
  
christian@debian-xps:~  
$ _
```

7. If a victim/target company's MTA was configured to follow the DMARC policy, which domain would be harder to spoof in a phishing attack?

As we did in the demonstration on redhat.com, we can check the DMARC policy by looking at the `txt` record of `_dmarc.(DOMAIN)` for example `_dmarc.google.com`.

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -type=txt _dmarc.christiant.io  
Server:        68.105.28.11  
Address:       68.105.28.11#53  
  
Non-authoritative answer:  
_dmarc.christiant.io    text = "v=DMARC1; p=quarantine"  
  
Authoritative answers can be found from:  
  
christian@debian-xps:~  
$ nslookup -type=txt _dmarc.caffeinatedsheep.com  
Server:        68.105.28.11  
Address:       68.105.28.11#53  
  
Non-authoritative answer:  
_dmarc.caffeinatedsheep.com    text = "v=DMARC1; p=none"  
  
Authoritative answers can be found from:  
  
christian@debian-xps:~  
$
```

Here we can see that `christiant.io` is telling recipient's MTA to quarantine emails that fail SPF or DKIM verification. We can also see that even if `caffeinatedsheep.com` had the proper SPF and DKIM records, they, the administrators of the `caffeinatedsheep.com` domain, aren't telling recipients to check for it when they receive an email from the `caffeinatedsheep.com`.

ANSWER: THE CHRISTIANT.IO DOMAIN HAS THE STRONGER DEFENSES. THE CAFFEINATEDSHEEP.COM DOMAIN IS THE WEAKER VICTIM AND EASIER TO SPOOF TARGET.

NS Lookup Guide

Perform an nslookup

```
~$ nslookup christiant.io
```

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup christiant.io  
Server:        68.105.28.11  
Address:      68.105.28.11#53  
  
Non-authoritative answer:  
Name:   christiant.io  
Address: 185.199.109.153  
Name:   christiant.io  
Address: 185.199.111.153  
Name:   christiant.io  
Address: 185.199.108.153  
Name:   christiant.io  
Address: 185.199.110.153  
Name:   christiant.io  
Address: 2606:50c0:8003::153  
Name:   christiant.io  
Address: 2606:50c0:8000::153  
Name:   christiant.io  
Address: 2606:50c0:8001::153  
Name:   christiant.io  
Address: 2606:50c0:8002::153  
  
christian@debian-xps:~  
$ _
```

Lets look at root txt record.

```
~$ nslookup -query=txt christiant.io
```

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -query=txt christiant.io  
Server:        68.105.28.11  
Address:      68.105.28.11#53  
  
Non-authoritative answer:  
christiant.io  text = "v=spf1 include:_spf.protonmail.ch mx ~all"  
christiant.io  text = "protonmail-verification=07059ca1b33d3d1994b  
22621d13009c60f4c801d"  
  
Authoritative answers can be found from:  
  
christian@debian-xps:~  
$ _
```

These are non-authoritative settings (cached). Lets find what servers are authoritative for this device. This call tell us about the DNS Registrar as well.

```
~$ nslookup -type=ns christiant.io
```

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -type=ns christiant.io  
Server:        68.105.28.11  
Address:       68.105.28.11#53  
  
Non-authoritative answer:  
christiant.io  nameserver = ns-cloud-b4.googledomains.com.  
christiant.io  nameserver = ns-cloud-b1.googledomains.com.  
christiant.io  nameserver = ns-cloud-b3.googledomains.com.  
christiant.io  nameserver = ns-cloud-b2.googledomains.com.  
  
Authoritative answers can be found from:  
ns-cloud-b1.googledomains.com  internet address = 216.239.32.107  
ns-cloud-b1.googledomains.com  has AAAA address 2001:4860:4802:32::6b  
ns-cloud-b3.googledomains.com  internet address = 216.239.36.107  
ns-cloud-b3.googledomains.com  has AAAA address 2001:4860:4802:36::6b  
ns-cloud-b2.googledomains.com  internet address = 216.239.34.107  
ns-cloud-b2.googledomains.com  has AAAA address 2001:4860:4802:34::6b  
ns-cloud-b4.googledomains.com  internet address = 216.239.38.107  
ns-cloud-b4.googledomains.com  has AAAA address 2001:4860:4802:38::6b  
  
christian@debian-xps:~  
$
```

Non-authoritative records may be *behind* the authoritative server. Some records can take hours or days to properly propagate. When making changes or checking for updates consider checking the records from the authoritative server.

```
~$ nslookup -type=txt www.christiant.io ns-cloud-b3.googledomains.com
```

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -type=txt www.christiant.io ns-cloud-b3.googledomains.com  
Server:        ns-cloud-b3.googledomains.com  
Address:       2001:4860:4802:36::6b#53  
  
www.christiant.io      canonical name = christian-tailon.github.io.  
  
christian@debian-xps:~  
$
```

SOA record can provide information about domain, email, and site maintainers. Lets see what this domain has.

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -type=soa christiant.io  
Server:        68.105.28.11  
Address:       68.105.28.11#53  
  
Non-authoritative answer:  
christiant.io  
    origin = ns-cloud-b1.googledomains.com  
    mail addr = cloud-dns-hostmaster.google.com  
    serial = 15  
    refresh = 21600  
    retry = 3600  
    expire = 259200  
    minimum = 300  
  
Authoritative answers can be found from:  
  
christian@debian-xps:~  
$
```

Lets check the `_dmarc` file setting.

```
~$ nslookup -query=txt _dmarc.christiant.io
```

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -query=txt _dmarc.christiant.io  
*** Invalid option: query=txt  
Server:        68.105.28.11  
Address:       68.105.28.11#53  
  
Non-authoritative answer:  
*** Can't find _dmarc.christiant.io: No answer  
  
christian@debian-xps:~  
$ _
```

Lets check the mx record.

```
~$ nslookup -query=mx christiant.io
```

```
christian@debian-xps:~  
  
christian@debian-xps:~  
$ nslookup -query=mx christiant.io  
Server:        68.105.28.11  
Address:       68.105.28.11#53  
  
Non-authoritative answer:  
christiant.io  mail exchanger = 20 mailsec.protonmail.ch.  
christiant.io  mail exchanger = 10 mail.protonmail.ch.  
  
Authoritative answers can be found from:  
mailsec.protonmail.ch  internet address = 185.70.42.129  
mailsec.protonmail.ch  internet address = 176.119.200.129  
mailsec.protonmail.ch  internet address = 185.205.70.129  
  
christian@debian-xps:~  
$ _
```

Dig (Domain Information Groper)

Perform an dig

```
~$ dig christiant.io
```

```
christian@debian-xps:~  
christian@debian-xps:~  
$ dig christiant.io  
  
; <<> DiG 9.18.10-2-Debian <<> christiant.io  
;; global options: +cmd  
;; Got answer:  
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 37229  
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 1  
  
;; OPT PSEUDOSECTION:  
; EDNS: version: 0, flags:; udp: 512  
;; QUESTION SECTION:  
christiant.io.                IN      A  
  
;; ANSWER SECTION:  
christiant.io.                2879   IN      A      185.199.108.153  
christiant.io.                2879   IN      A      185.199.111.153  
christiant.io.                2879   IN      A      185.199.110.153  
christiant.io.                2879   IN      A      185.199.109.153  
  
;; Query time: 7 msec  
;; SERVER: 68.105.28.11#53(68.105.28.11) (UDP)  
;; WHEN: Mon Jan 23 19:47:19 MST 2023  
;; MSG SIZE rcvd: 106  
  
christian@debian-xps:~  
$
```

Lets cleanup the output with the `+short` flag.

```
~$ dig +short christiant.io
```

```
christian@debian-xps:~  
christian@debian-xps:~  
$ dig +short christiant.io  
185.199.108.153  
185.199.110.153  
185.199.109.153  
185.199.111.153  
christian@debian-xps:~  
$
```

These are non-authoritative settings (cached). Lets find what servers are authoritative for this device. This call tell us about the DNS Registrar as well.

```
~$ dig +short christiant.io +nssearch
```

```
christian@debian-xps:~  
christian@debian-xps:~  
$ dig +short christiant.io +nssearch  
SOA ns-cloud-bl.googledomains.com. cloud-dns-hostmaster.google.com. 15 21600 3600 259200 300 from server 216.239.36.107 in 79 ms.  
SOA ns-cloud-bl.googledomains.com. cloud-dns-hostmaster.google.com. 15 21600 3600 259200 300 from server 216.239.34.107 in 79 ms.  
SOA ns-cloud-bl.googledomains.com. cloud-dns-hostmaster.google.com. 15 21600 3600 259200 300 from server 216.239.32.107 in 79 ms.  
SOA ns-cloud-bl.googledomains.com. cloud-dns-hostmaster.google.com. 15 21600 3600 259200 300 from server 216.239.38.107 in 83 ms.  
SOA ns-cloud-bl.googledomains.com. cloud-dns-hostmaster.google.com. 15 21600 3600 259200 300 from server 2001:4860:4802:36::6b in 83 ms.  
SOA ns-cloud-bl.googledomains.com. cloud-dns-hostmaster.google.com. 15 21600 3600 259200 300 from server 2001:4860:4802:34::6b in 83 ms.  
SOA ns-cloud-bl.googledomains.com. cloud-dns-hostmaster.google.com. 15 21600 3600 259200 300 from server 2001:4860:4802:32::6b in 83 ms.  
SOA ns-cloud-bl.googledomains.com. cloud-dns-hostmaster.google.com. 15 21600 3600 259200 300 from server 2001:4860:4802:38::6b in 83 ms.  
christian@debian-xps:~  
$
```

Non-authoritative records may be *behind* the authoritative server. Some records can take hours or days to properly propagate. When making changes or checking for updates consider checking the records from the authoritative server.

Lets query `www.christiant.io` with an authoritative source.

```
~$ dig +short www.christiant.io @ns-cloud-bl.googledomains.com
```

```
christian@debian-xps:~  
christian@debian-xps:~  
$ dig +short www.christiant.io @ns-cloud-b1.googledomains.com  
christian-taillon.github.io.  
christian@debian-xps:~  
$ _
```

SOA record can provide information about domain, email, and site maintainers. Lets see what this domain has.

~\$ dig SOA christiant.io

```
christian@debian-xps:~  
$ dig SOA christiant.io  
  
; <<> DiG 9.18.10-2-Debian <<> SOA christiant.io  
;; global options: +cmd  
;; Got answer:  
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 65006  
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1  
  
;; OPT PSEUDOSECTION:  
; EDNS: version: 0, flags:; udp: 512  
;; QUESTION SECTION:  
christiant.io. IN SOA  
  
;; ANSWER SECTION:  
christiant.io. 21600 IN SOA ns-cloud-b1.googledomains.com. cloud-dns-hostmaster.google.com. 15 21600 3600 25920  
0 300  
  
;; Query time: 80 msec  
;; SERVER: 68.105.28.11#53(68.105.28.11) (UDP)  
;; WHEN: Mon Jan 23 19:53:31 MST 2023  
;; MSG SIZE rcvd: 135  
  
christian@debian-xps:~  
$ _
```

Dig Bonus:

Some Dig returns are not as pretty to read. You can beautify the outputs by showing the results as YAML.

~\$ dig SOA christiant.io +yaml

```
christian@debian-xps:~  
$ dig SOA christiant.io +yaml  
-  
  type: MESSAGE  
  message:  
    type: RECURSIVE RESPONSE  
    query_time: !!timestamp 2023-01-24T03:01:58.078Z  
    response_time: !!timestamp 2023-01-24T03:01:58.158Z  
    message_size: 135b  
    socket_family: INET  
    socket_protocol: UDP  
    response_address: "68.105.28.11"  
    response_port: 53  
    query_address: "0.0.0.0"  
    query_port: 0  
    response_message_data:  
      opcode: QUERY  
      status: NOERROR  
      id: 51625  
      flags: qr rd ra  
      QUESTION: 1  
      ANSWER: 1  
      AUTHORITY: 0  
      ADDITIONAL: 1  
      OPT_PSEUDOSECTION:  
        EDNS:  
          version: 0  
          flags:  
            udp: 512  
      QUESTION_SECTION:  
        - christiant.io. IN SOA  
      ANSWER_SECTION:  
        - christiant.io. 21600 IN SOA ns-cloud-b1.googledomains.com. cloud-dns-hostmaster.google.com. 15 21600 3600 259200 300  
christian@debian-xps:~  
$ _
```


Lets check the `_dmarc` file setting.

```
~$ dig -t txt _dmarc.christiant.io +short
```

```
christian@debian-xps:~  
christian@debian-xps:~  
$ dig -t txt _dmarc.christiant.io +short  
"v=DMARC1; p=quarantine"  
christian@debian-xps:~  
$ _
```

Lets check the `mx` record.

```
~$ dig +short -t mx christiant.io
```

```
christian@debian-xps:~  
christian@debian-xps:~  
$ dig +short -t mx christiant.io  
20 mailsec.protonmail.ch.  
10 mail.protonmail.ch.  
christian@debian-xps:~  
$ _
```

Lets check the `txt` record which also contains an `spf` record.

```
~$ dig +short -t txt christiant.io
```

```
christian@debian-xps:~  
christian@debian-xps:~  
$ dig +short -t txt christiant.io 8.8.8.8  
"v=spf1 include:_spf.protonmail.ch mx ~all"  
"protonmail-verification=07059ca1b33d3d1994b22621d13009c60f4c801d"  
christian@debian-xps:~  
$ _
```