DNS Setup

DNS and Email Security

DNS infrastructure of the Internet plays a major role in email security today. Proper DNS setup for your email domain requires prior to setting up **Openemail** for your domain.

This section of the document guides you in setting up **recommended DNS records**. While some are mandatory for a mail server (A, MX), others are recommended to build a good reputation score (TXT/SPF) or used for auto-configuration of mail clients (SRV).

Setting up MX Records

Querying for Existing MX records

Let's first first find your MX records by running the `dig` command in your Linux system. Run the following query to view the MX records for an eample domain `openemail.io`. Simply change this domain to your domains to query MX for your domain.

```
dig openemail.io MX +short
```

You will get an output like below.

```
15 hasuna.openemail.io.
10 mail.openemail.io.
```

According to the above output the domain `openemal.io` has two `MX` records. The numbers `10` and `15` are respective `MX` priorities.

Sample MX Records

Use the below sample records to setup your `MX` Records

```
@   IN    MX  10  maila.yourdomain.tld
@   IN    MX  20  mailb.yourdomain.tld
```

The numbers 10 and 20 are the respective MX priorities which are used in delivering the mails. The MX records with lower priority will receive mails first.

Sample A Records for MX IPS

Next you need to setup A records for your MX. Below is a Sample

```
@   IN    A   maila     1.2.3.4
@   IN    A   mailb     4.5.6.7
```

Reverse DNS for Your MX IPs

Make sure that the PTR record of your IP matches the FQDN of your openemal host: `${OPENEMAIL_HOSTNAME}` [1]. This record is usually set at the provider you leased the IP (server) from.

To find the IP of your `MX` run:

Below is an example

```
dig mail.openemail.io +short
```

The above query should return an `IP` address like below.

```
68.183.186.231
```

To find your PTR record run:

```
host  68.183.186.231
```

You will get an output like Below

```
231.186.183.68.in-addr.arpa domain name pointer mail.openemail.io
```

Sample PTR Records

```
4.3.2.1.in-addr.arpa    IN    PTR   maila.yourdomain.tld.
7.6.5.4.in-addr.arpa.   IN    PTR   mailb.yourdomain.tld.
```

Auto Configuration of SSL certificates

For each domain that you will add to Openmail, it will try to resolve `autodiscover.ADDED_MAIL_DOMAIN` and `autoconfig.ADDED_MAIL_DOMAIN` to your servers IPv4

address. If it succeeds, these names will be added as SANs to the certificate request. The container `acme-openemail` is taking care of this process using Letsencrypt Certificate Authority. To learn more about this visit Firststeps SSL

Sample SSL Autoconfig and Autodiscover

Add these records to your DNS Zone as the per the sample below.

```
@   IN    A   autodiscover     1.2.3.4
@   IN    A   autodiscover     4.5.6.7
```

Setting up SPF Records

Haven't we all received emails that seem to be from our bank, our credit card company or even from ourselves but they were fake emails? The way emails are structured, spammers can and often do falsify the 'from email address' to send these types of spoof / spam emails. This is where SPF comes into place!

Sender Policy Framework (SPF) is a validation system that allows ISPs and webmail servers (Gmail, Yahoo, etc) to check if the incoming mail has been sent from an authorized server. Using the IP address of the sending server and the DNS records of your domain, ISPs can check if the sending server is authorized. If email is coming from an unauthorized sender, the emails will be marked as spam!

Set up SPF to prevent spammers from sending unauthorized emails from your domain. This type of spamming is called spoofing. Sender Policy Framework (SPF) is an email security method to prevent spoofing from your domain.

An SPF record is a TXT record that lists the mail servers that are allowed to send email from your domain. Messages sent from a server that isn't the SPF record might be marked as spam. Adding the TXT record doesn't affect your mail flow.

Sample SPF record

```
@   IN    TXT   "v=spf1 mx -all"
```

Using "v=spf1 mx -all" authorizes any IP that is also a MX for the sending domain. If you use `~all` ,it means softfail which allows mail whether or not it matches the parameters in the record.

Please refer to SPF Project for further reading.

Querying for SPF Records

` To find out your current SPF settings run:

```
$ dig TXT openemail.io +short
```

You will get an output like below.

```
"v=spf1  mx -all"
```

Setting UP DKIM Record

Use the DomainKeys Identified Mail (DKIM) standard to help prevent email spoofing on outgoing messages.

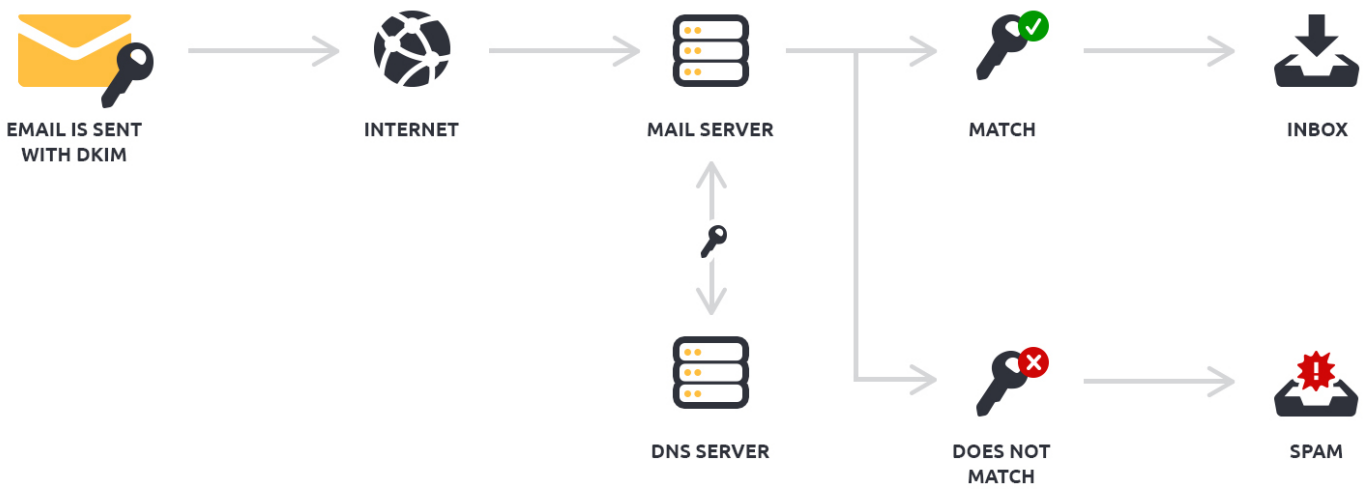
The acronym DKIM stands for “Domain Keys Identified Mail”. It is an encryption authentication method that is used by many ISPs to establish if the email originated from an authorized system and prevents spammers from stealing the identity of legitimate entities.

How does it work exactly?

DKIM allows for a unique signature to be added to the message header for each email you send. This signature is specific for your domain and is generated by a private key. The corresponding public key is added to a DNS record for your domain.

When an email server receives your email, it checks the public key to determine if your private key was used to generate the email signature. If your private key was not used, the email is considered to be a phishing or spam attempt.

The diagram below will make you understand about DKIM authentication process.



Creating a DKIM Key and DNS Record

It is highly recommended to create a **DKIM** TXT record in your **openemal UI## and set the corresponding TXT record in your DNS records. Please refer to OpenDKIM for further reading.

Creating DKIM key in Openemail UI

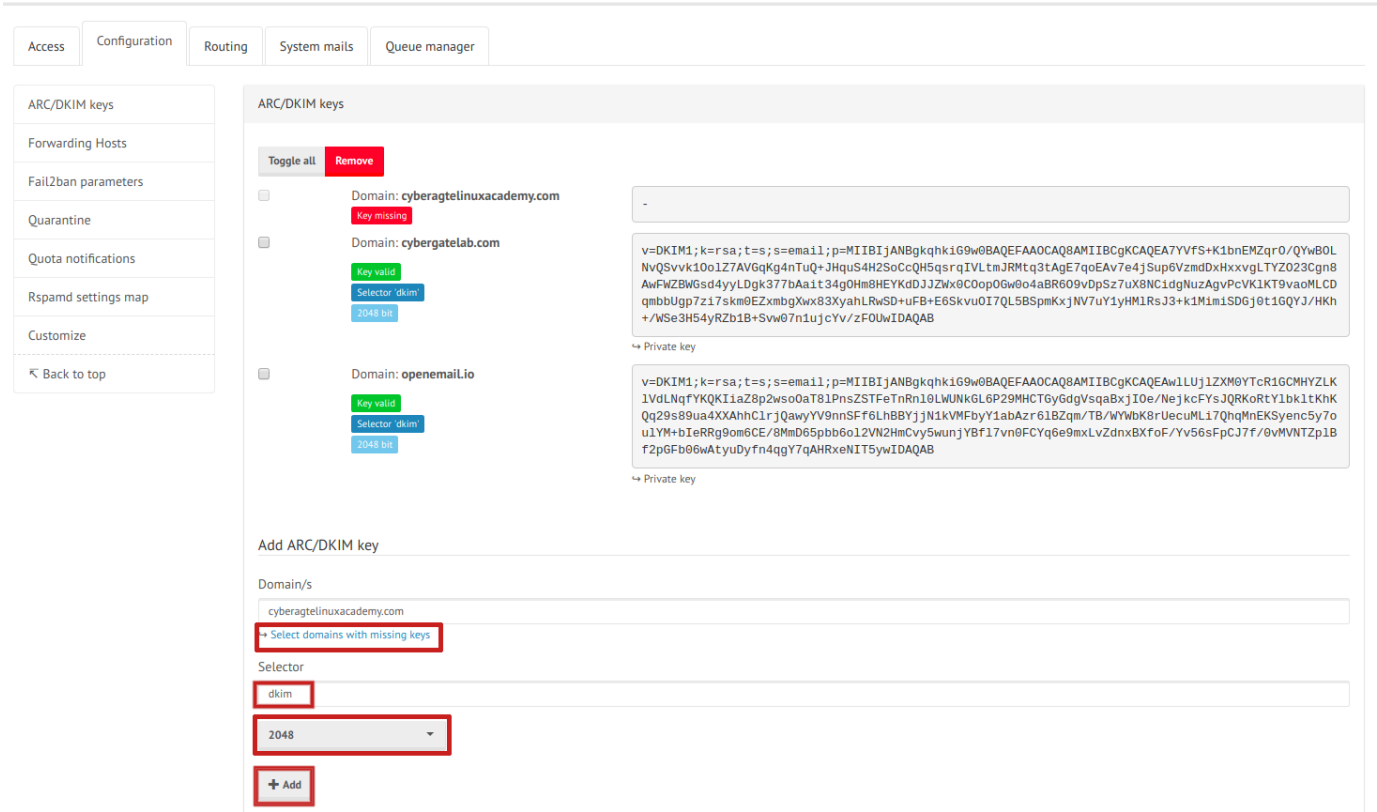
Do the following steps which are marked in red color in the configuration window screenshot of **openemail-UI** below.

1. Click on `Select domains with missing keys` to select your domain.

2. Set `dkim` key length as `2048`

3. Leave the DKIM selector as it is

4. Press `Add` button



Using OpenDKIM

The below steps can be used to create DKIM key in you Linux command line in case if you are using your own mail server like postfix. You do not need to execute commands below when you deploy **openemal**. We have listed those commands to understand how the postfix back-end has been configured for DKIM validations.

Install required tools

```
sudo apt-get install opendkim opendkim-tools
```

Create the _domainkey

Now create DKIM key pair using `opendkim-genkey` command line utility. For this guide we are using domain name `openemail.io`, Change this name with your actual domain name.

```
mkdir -p dkim-keys/openemail.io
cd dkim-keys/openemail.io
opendkim-genkey -t -s dkim -d openemail.io
cat dkim.txt
```

You will get an output like below. It is actually the TXT record for `bind` and compatible DNS servers.

```
dkim._domainkey IN  TXT ( "v=DKIM1; h=sha256; k=rsa; t=y; "

"p=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAtIfYQWd16SotLpDAIe+9CRda6+KmExTSM4lC
Dgq21/sYrERnnXYVvH5jSA44YBfiwrvuzopl/ekMp71WJMR73TmEQ1BTf6SZA6STvCJVj8d"
      "pAuLekL1EYQzsGsp1kcOFYOee9c9VjfgkuZnJkAkQmz94L3YTd/
08i6rZrKUSRXcx8dfitx7k12IrBsKCqlXgRIpWYWJW58gAM1Fif6gjDmugj4mAIQJ4oyWJdNDZPYpKEG+6V
----- DKIM key dkim for openemail.io
```

Setup DMARC Record

DMARC defines how your domain handles suspicious emails.

Spammers can forge the "From" address on email messages to make messages appear to come from someone in your domain. If spammers use your domain to send spam or junk email, your domain quality is negatively affected. People who get the forged emails can mark them as spam or junk, which can impact authentic messages sent from your domain.

Use DMARC to define the policy for how **openemail** handles spam emails that appear to be sent from your domain.

Learn more about DMARC.

Sample DMARC Record

You can setup a TXT record for your domains' DMARC using the example below.

```
_dmarc    IN    TXT   "v=DMARC1; p=quarantine; pct=5;
rua=mailto:postmaster@yourdomain.tld"
```

Following are the descriptions of the parameters used in the above record.

- p - Policy for organizational domain
- pct - Percentage of messages subjected to filtering
- rua - Reporting URI of aggregate reports

To query DNS for an existing entry you can run:

```
dig TXT _dmarc.openemail.io  +short
```

You will get an output like below.

```
"v=DMARC1; p=quarantine; pct=5; rua=mailto:postmaster@openemail.io"
```

The Advanced DNS Configuration

SRV records specify the server(s) for a specific protocol on your domain. If you want to explicitly announce a service as not provided, give "." as the target address (instead of "mail.yourdomain.tld."). Please refer to RFC 2782. for more details.

```
_imap._tcp          IN SRV     0 1 143   mail.yourdomain.tld.
_imaps._tcp         IN SRV     0 1 993   mail.yourdomain.tld.
_pop3._tcp          IN SRV     0 1 110   mail.yourdomain.tld.
_pop3s._tcp         IN SRV     0 1 995   mail.yourdomain.tld.
_submission._tcp    IN SRV     0 1 587   mail.yourdomain.tld.
_smtps._tcp         IN SRV     0 1 465   mail.yourdomain.tld.
_sieve._tcp         IN SRV     0 1 4190  mail.yourdomain.tld.
_autodiscover._tcp  IN SRV     0 1 443   mail.yourdomain.tld.
_carddavs._tcp      IN SRV     0 1 443   mail.yourdomain.tld.
_carddavs._tcp      IN TXT     "path=/SOGo/dav/"
_caldavs._tcp       IN SRV     0 1 443   mail.yourdomain.tld.
_caldavs._tcp       IN TXT     "path=/SOGo/dav/"
```

Testing for SPF, DKIM, and DMARC

Here are some tools you can use to verify your DNS configuration:

- MX Toolbox (DNS, SMTP, RBL)
- port25.com (DKIM, SPF)

- Mail-tester (DKIM, DMARC, SPF)
- DMARC Analyzer (DMARC, SPF)

Getting Additional Statistics

If you are interested in statistics, you can additionally register with the Postmaster Tool by Google and supply a **google-site-verification** TXT record, which will give you details about spam-classified mails by your domain. This is clearly optional. You can add a TXT records in your DNS server like the one below.

```
@                   IN TXT     "google-site-verification=..."
```

Additional References

- A good article covering all relevant topics: "3 DNS Records Every Email Marketer Must Know"
- Another great one, but Zimbra as an example platform: "Best Practices on Email Protection: SPF, DKIM and DMARC"
- An in-depth discussion of SPF, DKIM and DMARC: "How to eliminate spam and protect your name with DMARC"

---

1. A **Fully Qualified Domain Name** (**FQDN**) is the complete (absolute) domain name for a specific computer or host, on the Internet. The FQDN consists of at least three parts divided by a dot: the hostname (myhost), the domain name (mydomain) and the top level domain in short **tld** (com). In the example of `mx.openemail.io` the hostname would be `mx`, the domain name 'openemal' and the tld is `io`.