Getting DKIM, DMARC and SPF to work with Postfix, OpenDKIM and OpenDMARC

There are plenty of tutorials out there how to get DKIM, SPF, and DMARC working, but it still took me a couple of hours to get everything right, because each and every one of them lacks something to actually work.

There are plenty of tutorials out there how to get DKIM[^1], SPF[^2] and DMARC[^3] working, but it still took me a couple of hours to get everything right, because each and every one of them lacks something to actually work.

I also recommend reading another tutorial: https://www.skelleton.net/2015/03/21/how-to-eliminate-spam-and-protect-your-name-with-dmarc/

It's covering the topic a bit better than this entry of mine.

Note: mail servers are not hard, just complex

I've seen a lot of bragging these days that mail servers are hard; that the Big Services greylists and trust-lists are getting impenetrable.

That is not true, but you need to keep a few things in mind:

1. Do not try to run a massive newsletter server; that will always raise a warning flag. There are services out there for you to handle this.
2. **Make sure you can accept and send mails via TLS; this is crucial**. DKIM, SPF and DMARC are not yet a must, but TLS is.
3. Double check your mail server with external services; there are good and useful services out there, like:
    - mxtoolbox[^4]
    - DKIM & SPF[^5] validator
    - mail-tester[^6]
4. Forwarding mails is getting tricky due to the SPF and DMARC. Try to avoid that, if possible. If you need to do it anyway, make sure that at least you clear all DKIM headers from the mail, so you avoid bogus DKIM signatures.
5. DMARC and SPF can break mailing lists for the same reasons as forwarding. Keep that in mind.

Let's get started

The domain I'm setting this up for is `domain.com`; replace it with your own. To test the result, use https://www.mail-tester.com/.

WARNING If DKIM, SPF and DMARC are set in the DNS but not actually working, so mail is not signed, etc., it'll do more harm than good. Leave all the required DNS entries to the very end, after you're sure all mails are fine!

SPF

DNS

This is relatively easy and straightforward; add the following to your DNS record:

```apache
*.domain.com. 1800 IN TXT "v=spf1 mx ip4:YOUR_MX_IP -all"
domain.com. 1800 IN TXT "v=spf1 mx ip4:YOUR_MX_IP -all"
```

`YOUR_MX_IP` is the IP address of your mail server. If there is more, add more `ip4:IP` entries separated by spaces. `-all` means that mails should only be accepted from the IPs listed.

DKIM

DKIM key for your domain

You'll need to generate a DKIM private key. The fastest way to do this

```sh
opendkim-genkey -b 2048 -d domain.com -s domain.com.dkim
```

This will output 2 files:

- domain.com.dkim.private
- domain.com.dkim.txt

the `.private` is your key; keep it safe. The `.txt` contains the DNS entry you'll need.

OpenDKIM config

OpenDKIM[^7] is a useful software, but it's picky and it lacks proper error handling. This means that if you misconfigure it, or set `AutoRestart yes`, it will still act like it was fine, but it won't sign your mails.

So, my config (it's combined with postfix):

`/etc/opendkim.conf`

```apache
Socket local:/var/spool/postfix/private/opendkim
Syslog yes
UMask 002
UserID postfix

Selector mail
Mode sv
SubDomains yes
AutoRestart yes
Background yes
Canonicalization relaxed/relaxed
DNSTimeout 5
SignatureAlgorithm rsa-sha256
X-Header yes
```

```
Logwhy yes

InternalHosts /etc/internalhosts
KeyTable /etc/opendkim/keytable
SigningTable refile:/etc/opendkim/signtable

OversignHeaders From
```

The important part is the **Selector mail** - this will be what you need to set in the DNS and in the KeyTable.

`/etc/opendkim/signtable`

```apache
*@domain.com domain.com
```

`/etc/opendkim/keytable`

```apache
domain.com domain.com:mail:/path/to/
domain.com.dkim.private
```

`/etc/internalhosts`

```apache
your.domain.com
domain.com
192.168.0.0/255.255.255.0
```

DNS for DKIM

Remember the Selector entry in the opendkim.conf? You need that in from of `_domainkey.domain.com` to work; in our case, `mail`.

```apache
mail._domainkey.domain.com. 1800 IN TXT "what's in your
domain.com.dkim.txt file between the double quotes"
```

DMARC

DNS for DMARC

You'll then need to add another TXT record to your DNS to get DMARC working.

```apache
_dmarc.domain.com. 1800 IN TXT "v=DMARC1; p=none;
rua=mailto:postmaster@domain.com"
```

The `none` indicates that the remove server should not drop the mails, even if they are not coming from the servers listed in the SPF record. Once you're sure everything is fine, change the `none` to `reject`.

There is also the option to "cc" a service, by adding more `rua` entries. For example, you can include agari[^8], like this:

```apache
_dmarc.domain.com. 1800 IN TXT "v=DMARC1; p=reject;
rua=mailto:d@rua.agari.com,mailto:postmaster@domain.com;
ruf=mailto:d@ruf.agari.com,mailto:postmaster@domain.com"
```

This will not drop any mail, so this is only the initial setup; later change none to reject, but do be careful.

OpenDMARC config

Note: the Debian init script hardcodes the username/group; you have to change it there as well. Also, OpenDMARC seems to be buggy and somewhat unreliable; test it well before relying on it.

`/etc/opendmarc.conf`

```apache
AuthservID mail.domain.com
PidFile /var/run/opendmarc.pid
RejectFailures false
Syslog true
SyslogFacility mail
TrustedAuthservIDs mail.domain.com
IgnoreHosts /etc/opendmarc/ignore.hosts
```

```
UMask 002
UserID postfix:postfix
TemporaryDirectory /tmp
Socket local:/var/spool/postfix/private/opendmarc
FailureReportsSentBy postmaster@domain.com
FailureReportsBcc postmaster@domain.com
FailureReports false
AutoRestart true
PublicSuffixList /etc/effective_tld_names.dat
HistoryFile /var/log/opendmarc.log
```

/etc/opendmarc/ignore.hosts

```
localhost
127.0.0.0/8
192.168.0.0/24
```
apache

Postfix

This is not my complete postfix config, just the required lines for OpenDKIM and OpenDMARC to be effective.

Part of `/etc/postfix/main.cf`:

```apache
smtpd_milters = unix:private/opendkim unix:private/
opendmarc
non_smtpd_milters = unix:private/opendkim unix:private/
opendmarc
```

Working DNS example

This is how it should look like in the end, with a current, working example, for my `petermolnar.eu` domain:

```apache
$ORIGIN petermolnar.net.
$TTL 1800
petermolnar.net. IN SOA ns1.digitalocean.com.
hostmaster.petermolnar.net. 1471376771 10800 3600 604800 1800
petermolnar.net. 1800 IN NS ns1.digitalocean.com.
petermolnar.net. 1800 IN NS ns2.digitalocean.com.
petermolnar.net. 1800 IN NS ns3.digitalocean.com.
petermolnar.net. 1800 IN A 176.9.137.114
petermolnar.net. 1800 IN MX 10 mail.petermolnar.eu.
*.petermolnar.net. 1800 IN A 176.9.137.114
petermolnar.net. 1800 IN TXT "v=spf1 mx ip4:176.9.137.114
ip4:88.96.115.88/29 -all"
*.petermolnar.net. 1800 IN TXT "v=spf1 mx ip4:176.9.137.114
ip4:88.96.115.88/29 -all"
_dmarc.petermolnar.net. 1800 IN TXT "v=DMARC1; p=reject;
rua=mailto:d@rua.agari.com,mailto:postmaster@petermolnar.eu; ;
ruf=mailto:d@ruf.agari.com,mailto:postmaster@petermolnar.eu"
mail._domainkey.petermolnar.net. 1800 IN TXT "v=DKIM1; k=rsa;
p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDFBnAcF/
qUPAdpdPxYISnS0XrzS/GWIKa7r8Xh6lNTE4/
tBfSiFLFkHguOxoT6+JWJiKjjsvM9cXhLa2yKf1R5EjGuOoVQokcIqZJ2oeJRwJ
SRQBy6KX9cFuPD/
ZUYJiFFMPL1dqdD+G8FCnF1FjPddRaOgfokcT4KEB+JhbFuWwIDAQAB"
hello._pka.petermolnar.net. 1800 IN TXT
"v=pka1;fpr=AADDEF2263C9E5B52B4DE59C1E8898416C1F051F;uri=https:
//petermolnar.eu/pgp.asc/"
petermolnar.net. 1800 IN TXT dnslink="/ipns/
QmZhKarpdPZi5pgAsnoqQvX6dEAK4EDTcs96qGf4w38Td9"
```

Links

1. http://dkim.org/
2. https://en.wikipedia.org/wiki/Sender_Policy_Framework
3. https://dmarc.org/

4. https://mxtoolbox.com/NetworkTools.aspx
5. http://dkimvalidator.com/
6. https://www.mail-tester.com/
7. http://www.opendkim.org/
8. https://www.agari.com/

Created by Peter Molnar <mail@petermolnar.net>, published at 2015-10-24 15:52 UTC, last modified at 2021-06-21 18:07 UTC , to canonical URL https://petermolnar.net/article/howto-spf-dkim-dmarc-postfix/ , licensed under CC-BY-4.0 .