Email Security

Contents

1 Introduction

Implement and configure `SPF` (Sender Policy Framework) and `DKIM` (DomainKeys Identified Mail) on a `sender` mail server and a `receiver` mail server.

2 Methodology

2.1 Installation

2.1.1 Server and IP Configuration

Ubuntu Server 18.04.2 LTS [1] was downloaded and installed in VirtualBox.

The first virtual machine (VM) was set up with the following information:

Hostname: `sender`
Username: user1
Password: password1

The second VM was set up with the following information:

Hostname:	`receiver`
Username:	user2
Password:	password2

OpenSSH server was installed on both VMs during set up for convenience.

When installation is complete, networking settings for both VMs were set to the following:

Adapter 1:	NAT
Adapter 2:	NAT Network

On the first network interface (NAT), add port forwarding rules:

Name:	Rule 1
Protocol:	TCP
Host IP:	127.0.0.1
Host Port:	2227 (2228 for `receiver`)
Guest IP:	10.0.2.15
Guest Port:	22

This allows the host to SSH into the VMs which can be started headless, instead of having to interact through VirtualBox windows:

```
ssh -vp2227 user1@127.0.0.1

ssh -vp2228 user2@127.0.0.1
```

Static IP addresses were configured for both `sender` and `receiver` mail servers. `sender` was assigned an IP address of `10.0.2.7` (seven for send), while `receiver` was assigned an IP address of `10.0.2.8` (change the appropriate line).

`/etc/netplan/01-netcfg.yaml`:

```
network:
    ethernets:
        enp0s3:
            dhcp: true
        enp0s8:
            dhcp4: false
            addresses: [10.0.2.7/24]
            gateway4: 10.0.2.1
            nameservers:
                addresses: [127.0.0.1]
            dhcp6: false
    version: 2
```

Test and apply the `netplan` settings:

```
sudo netplan try
```

NOTE FOR UBUNTU 18.04 HOSTS

A VirtualBox bug on hosts running Ubuntu 18.04 LTS prevents `NAT Network` from making proper connections to the Internet [2]. When `NAT` is enabled, `NAT Network` stops connecting to other guests.

If facing this problem, enable only `NAT` on `Adapter 1` while setting up the VMs, before switching it off and enabling `NAT Network` on `Adapter 2` when ready to proceed with testing.

Even then, `DKIM` queries will not work properly, and will time out instead. Use either Ubuntu 18.10 or a Windows 10 host to get proper query results.

2.1.2 bind9 DNS Setup

Install `bind9`:

```
sudo apt update && sudo apt install bind9
```

Change the DNS resolver daemon to use `localhost` instead of querying the router.

`/etc/systemd/resolved.conf`:

```
[Resolve]
DNS=127.0.0.1
```

Restart `systemd-resolved` to apply the settings:

```
sudo service systemd-resolved restart
```

Add forward zones to the local DNS settings.

`sudo vi /etc/bind/named.conf.local`:

```
zone "sender.com" {
        type master;
        file "db.sender.com";
};

zone "receiver.com" {
        type master;
        file "db.receiver.com";
};
```

Create the respective zone files.

`/var/cache/bind/db.sender.com`: db.sender.com

`/var/cache/bind/db.receiver.com`: db.receiver.com

The validity of the zone files can be checked:

```
sudo named-checkzone sender.com /var/cache/bind/db.sender.com

sudo named-checkzone receiver.com /var/cache/bind/db.receiver.com
```

Restart `bind9` to apply settings:

```
sudo service restart bind9
```

2.1.3 postfix Mail Transfer Agent Setup

Install `postfix`:

```
sudo apt install postfix
```

Select `Internet Site` for the type of mail configuration.

The mail name should be set respectively:

```
sender.com

receiver.com
```

2.1.4 SPF and DKIM Setup

Install `SPF daemon`, `OpenDKIM` and associated tools:

```
sudo apt install postfix-policyd-spf-python opendkim opendkim-tools python-dkim mailutils
```

Add the `postfix` user to the `opendkim` user group:

```
sudo usermod -aG opendkim postfix
```

Edit `postfix` master process configuration to start the `SPF daemon`.

`/etc/postfix/master.cf`:

```
# Postfix master process configuration file. ...
...

policyd-spf unix -       n       n       -       0       spawn
  user=policyd-spf argv=/usr/bin/policyd-spf
```

Configure `postfix` to reject mail that fails SPF, and add settings for `OpenDKIM`.

`/etc/postfix/main.cf`:

```
# See /usr/share/postfix/main.cf.dist for a commented...
...

policyd-spf_time_limit = 3600
smtpd_recipient_restrictions = permit_mynetworks permit_sasl_authenticated reject_unauth_destination check_policy_service unix:private/policyd-spf
# milter configuration for opendkim
milter_default_action = accept
milter_protocol = 6
smtpd_milters = local:/opendkim/opendkim.sock
non_smtpd_milters = $smtpd_milters
```

Edit the settings for `OpenDKIM`.

`/etc/opendkim.conf`:

```
...
# Commonly-used options; the commented-out versions show the defaults.
#Canonicalization       simple
#Mode                   sv
#SubDomains             no
Canonicalization        relaxed/simple
Mode                    sv
SubDomains              no
AutoRestart             yes
AutoRestartRate         10/1M
Background              yes
```

```
DNSTimeout              5
SignatureAlgorithm      rsa-sha256
```

Comment out the following socket configuration and add the following:

```
#Socket                 local:/var/run/opendkim/opendkim.sock
Socket                  local:/var/spool/postfix/opendkim/opendkim.sock
```

Append the following configuration at end of file:

```
...
# KeyTable and SigningTable required on sender.com only #
KeyTable                /etc/opendkim/key.table
SigningTable            refile:/etc/opendkim/signing.table
# ------------------------------------------------------ #
ExternalIgnoreList      /etc/opendkim/trusted.hosts
InternalHosts           /etc/opendkim/trusted.hosts
# to help with debugging
LogWhy                  Yes
SyslogSuccess           Yes
# important so that opendkim queries local dns
Nameservers             127.0.0.1
```

NOTE: Not adding the `Nameservers` setting will cause `OpenDKIM` to query actual DNS. The query will fail when it cannot find the DKIM public key.

Create `OpenDKIM` directories for `sender` and `receiver` respectively:

```
sudo mkdir -pv /etc/opendkim/keys/sender.com
sudo chown -R opendkim:opendkim /etc/opendkim
sudo chmod 711 /etc/opendkim/keys
sudo mkdir -pv /var/spool/postfix/opendkim
sudo chown opendkim:postfix !$
```

Create the file for trusted hosts for `sender` and `receiver` respectively.

`/etc/opendkim/trusted.hosts`:

```
127.0.0.1
localhost
*.sender.com
```

On `sender`, create the signing table and the key table:

`/etc/opendkim/signing.table`:

```
*@sender.com default._domainkey.sender
```

`/etc/opendkim/key.table`:

```
default._domainkey.sender sender.com:default:/etc/opendkim/keys/sender.com/default.private
```

2.1.5 OpenDKIM Public/Private Key Pair Generation

Generate the public/private key pair for `sender`, with a 2048-bit length as recommended by the Certified Senders Alliance in 2018 [3]:

```
sudo opendkim-genkey -b 2048 -d sender.com -D /etc/opendkim/keys/sender.com -s default -v
```

Change the owner of the private key to `opendkim`:

```
sudo chown opendkim:opendkim /etc/opendkim/keys/sender.com/default.private
```

Restart all services for the settings to apply:

```
sudo service bind9 restart && sudo service postfix restart && sudo service opendkim restart
```

3 Results

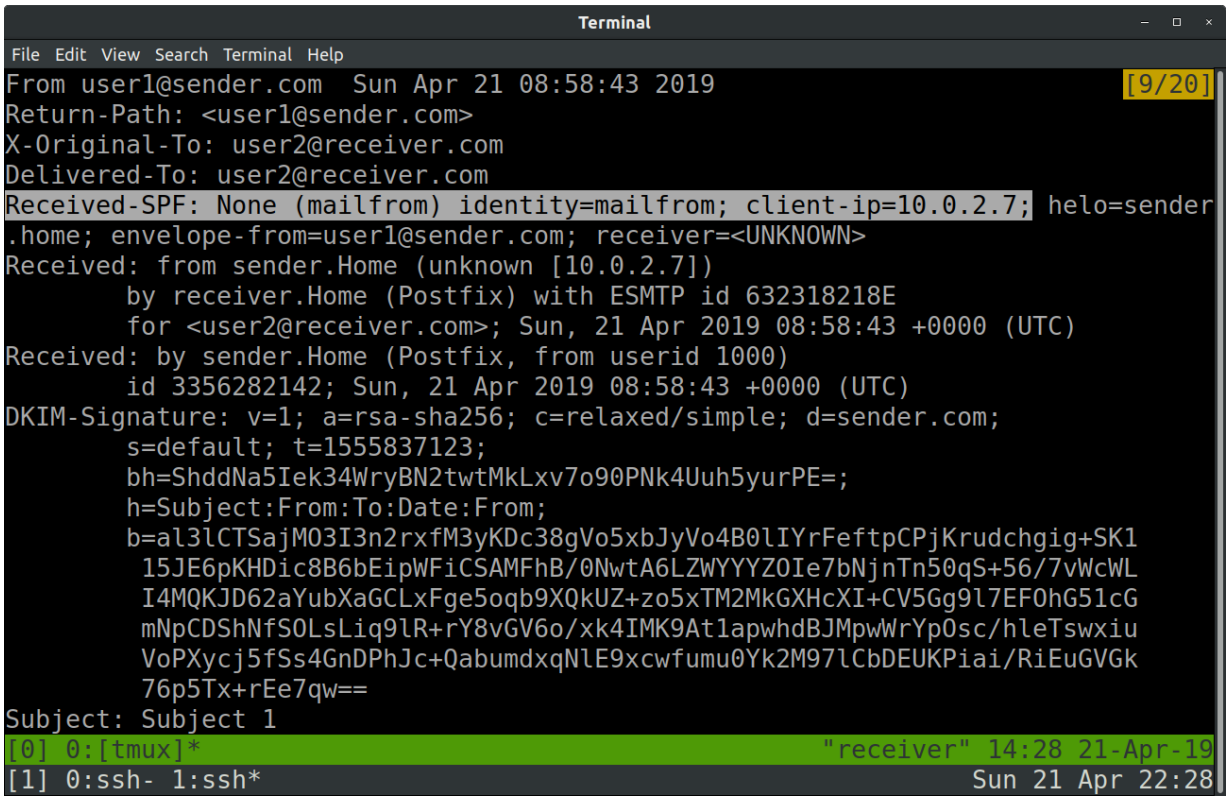
3.1 Sending Plain Email

A plain email was sent from `sender` to `receiver`:

```
echo "Hello from User 1" | mail -s "Subject 1" -a "From:user1@sender.com" user2@receiver.com
```

Email can be viewed on the `receiver` at `/var/mail/user2`.

When receiving email without `SPF` or `DKIM`, `Received-SPF` will have a value of `None`.



Received-SPF: None

The content of the email was saved to plain.eml.

3.2 Sending Spoofed Email

The IP address of `sender` was changed by editing the `netplan` configuration file.

`/etc/netplan/01-netcfg.yaml`:

```
network:
   ...
        enp0s8:
            ...
            addresses: [10.0.2.9/24]
            ...
```

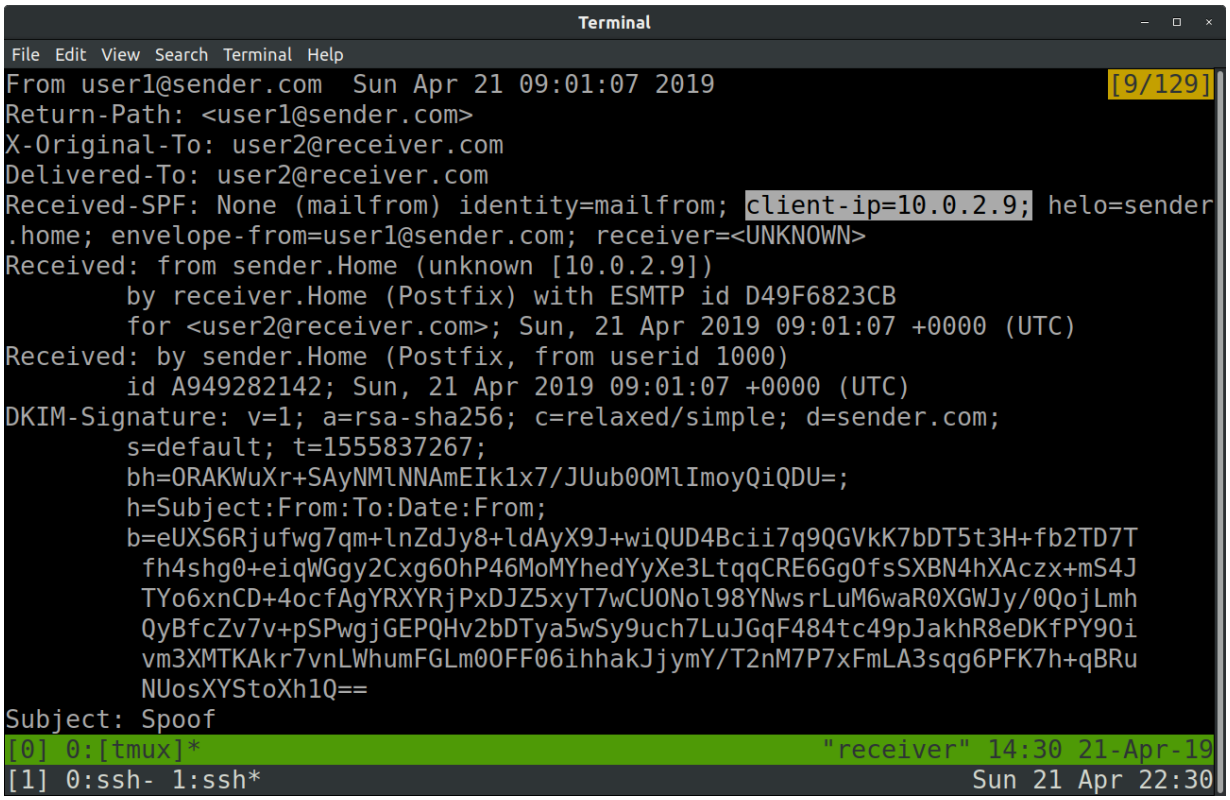
Settings were applied:

```
sudo netplan apply
```

An email from a "spoofed" IP address was sent from `sender` to `receiver`:

```
echo "Spoofed sender" | mail -s "Spoof" -a "From:user1@sender.com" user2@receiver.com
```

The email will still be received by `receiver`, but from the different IP address as expected.



client-ip=10.0.2.9

The content of the email was saved to spoofed.eml.

3.3 Enabling SPF

SPF was enabled by adding a `TXT` record to the local DNS zone file at `receiver`.

`/var/cache/bind/db.sender.com`:

```
...
@        IN       TXT      "v=spf1 ip4:10.0.2.7 -all"
```

The zone file is formatted in theh following manner:

name: `@` is a shortcut for the value of `$ORIGIN` (i.e. `example.com.`).

ttl: Leaving `ttl` blank defaults the time-to-live field to the value of `$TTL` (i.e. `1d`).

record class: `IN` refers to the Internet namespace.

record type: `TXT` records are used to store SPF configuration

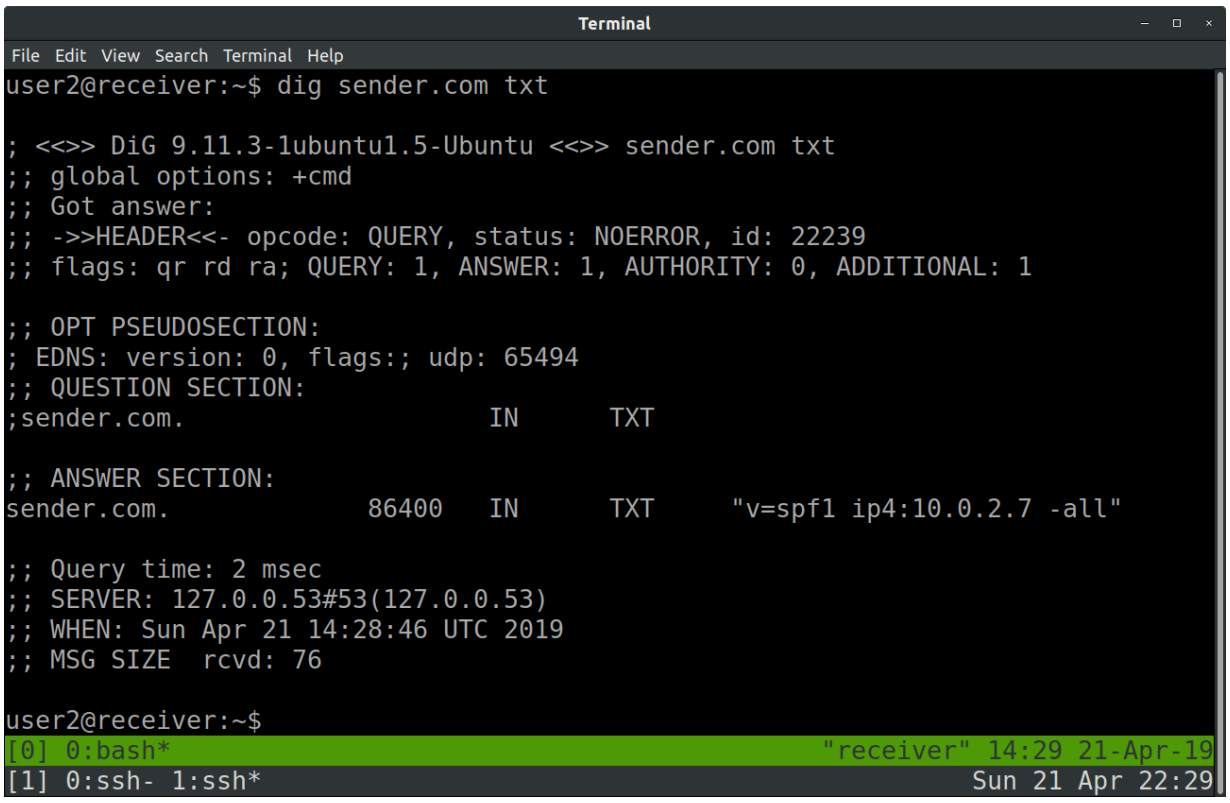
record data: `v=spf1` indicates version 1 of the Sender Policy Framework is used.

`ip4:x.x.x.x` directly lists the outgoing mail server's IP address to avoid additional DNS lookups, which is limited to a maximum of 10.

`-all` rejects all mail that do not match SPF records.

After restarting the DNS service (`sudo service bind9 restart`), the `TXT` record can be retrieved:

```
dig sender.com txt
```



"v=spf1 ip4:10.0.2.7 -all" in TXT record for sender.com

3.4 Sending Email with SPF

The IP address of `sender` was reset to `10.0.2.7` in `netplan` before proceeding.

An email was sent from `sender` to `receiver`:

```
echo "Hello from User 1 with SPF" | mail -s "Test SPF" -a "From:user1@sender.com" user2@receiver.com
```

The email was received with a `Received-SPF: Pass` header.

```
Terminal
File  Edit  View  Search  Terminal  Help
From user1@sender.com  Sun Apr 21 09:06:48 2019                    [9/85]
Return-Path: <user1@sender.com>
X-Original-To: user2@receiver.com
Delivered-To: user2@receiver.com
Received-SPF: Pass (mailfrom) identity=mailfrom; client-ip=10.0.2.7; helo=sender
.home; envelope-from=user1@sender.com; receiver=<UNKNOWN>
Received: from sender.Home (unknown [10.0.2.7])
        by receiver.Home (Postfix) with ESMTP id B065E823CD
        for <user2@receiver.com>; Sun, 21 Apr 2019 09:06:48 +0000 (UTC)
Received: by sender.Home (Postfix, from userid 1000)
        id 88AAD8214C; Sun, 21 Apr 2019 09:06:48 +0000 (UTC)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=sender.com;
        s=default; t=1555837608;
        bh=fm1Wb/rgulL2os+gDpV+R1AjQmcqYKSxaBrp0rSHDgM=;
        h=Subject:From:To:Date:From;
        b=EcTVUou6bLGX9O3SC7dWJVPQXA66hl8/IfWiX5vjylSJxhp9BXoZuWxtvnEQjtm5C
         maTa/gSIGShj+6Kchccih107+wwvJr5pleWMLHRooEqO/waFi7r8jwOuls218av/iR
         jT32wGpKZsxjud8qMGcJS7lrcmuDpHDaYwbD48Q9xIR6ZJvzK4NfIQxIXK/MRAV2iL
         pfekpGc4OrmUzOlu4B/bkzouMro/zUpGkGN9DwcSlGptOkX/monv7AZFOKGhrgIWbZ
         vcYA1+aUjk2+ImgIUVwFSufGax24Ca9R+3Pa7fZbxBdsJNvvb8MLSBrW3A4guRxROM
         su3hnCH1hlcIw==
Subject: Test SPF
[0] 0:[tmux]*                                   "receiver" 14:30 21-Apr-19
[1] 0:ssh- 1:ssh*                                        Sun 21 Apr 22:30
```

Received-SPF: Pass

The content of the email was saved to spf_pass.eml.

3.5 Sending Email with Invalid SPF

The IP address of `sender` was once again changed to `10.0.2.9` via `netplan`.

An email was sent from `sender` to `receiver`:

```
echo "Spoofed sender with SPF" | mail -s "Spoof with SPF" -a "From:user1@sender.com" user2@receiver.com
```

`receiver` did not receive any email as the SPF policy is set to hard fail invalid checks.

The spoofer (`sender`) however, received an email notification for failing the SPF check.

```
Terminal
File  Edit  View  Search  Terminal  Help
X-Postfix-Sender: rfc822; user1@sender.com                          [21/71]
Arrival-Date: Sun, 21 Apr 2019 09:04:03 +0000 (UTC)

Final-Recipient: rfc822; user2@receiver.com
Original-Recipient: rfc822;user2@receiver.com
Action: failed
Status: 5.7.23
Remote-MTA: dns; mail.receiver.com
Diagnostic-Code: smtp; 550 5.7.23 <user2@receiver.com>: Recipient address
    rejected: Message rejected due to: SPF fail - not authorized. Please see
    http://www.openspf.net/Why?s=mfrom;id=user1@sender.com;ip=10.0.2.9;r=<UNKNOW
N>

--C35B282142.1555837444/sender.Home
Content-Description: Undelivered Message
Content-Type: message/rfc822
Content-Transfer-Encoding: 8bit

Return-Path: <user1@sender.com>
Received: by sender.Home (Postfix, from userid 1000)
        id C35B282142; Sun, 21 Apr 2019 09:04:03 +0000 (UTC)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=sender.com;
[0] 0:[tmux]*                                      "sender" 14:26 21-Apr-19
[1] 0:ssh* 1:ssh-                                        Sun 21 Apr 22:26
```

Message rejected due to: SPF fail - not authorized.

The content of the email was saved to spf_fail.eml.

3.6 Enabling DKIM

The public key in `sender`'s `/etc/opendkim/keys/sender.com/default.txt` was added to the DNS zone records, particularly in `receiver`, for `DKIM` checks to succeed. Since the 2048-bit key we generated consists of over 256 characters, the entry for the TXT record has to be split into chunks of 255 octets or less due to limit specified in RFC 1035 [4]. Whitespace and splits are minimized to reduce overheads.

`/var/cache/bind/db.sender.com` at `receiver`:

```
default._domainkey IN TX ("v=DKIM1;h=sha256;k=rsa;p=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAwQcJVQIf/fUlfdIZ/EAjMGH6vb8ISZ0Cpuo0CIMLNbJwbZQYzCSo0bc14rPr1AuCrYTUrPHTIPcSLKA6UpMqx5+ZhzKjinf/3mlXMmKSuzZXAMjl7ZIdDpbrlgYwaicslLPrD0ksdK2vNbwPPKEZSvNHWHVDnQYJzDLzlbuBwzW" "uTZ3kPelPkzbIlVLxTPYKZ8KZCK1eYVbtk1n6l-iem9dBqplcbvdesdbm7mdzkBbpJbJ8QV77RgKBPJ8BLmZjrtbv-3-gR7BKDNBB8hAvqgpBKj1bK1PMbJbWX7HHLBMQEPSBw-BS/ieJ6JrVdb8qLdbQZDAQAB")
```

Restart all services for the settings to apply:

```
sudo service bind9 restart && sudo service postfix restart && sudo service opendkim restart
```

Test the `OpenDKIM` key, which should return `key OK`:

```
opendkim-testkey -d sender.com -s default -vvv
```

The domain key `TXT` record can then be retrieved:

```
dig default._domainkey.sender.com txt
```

```
Terminal
File  Edit  View  Search  Terminal  Help
user2@receiver:~$ dig default._domainkey.sender.com txt                [5/225]

; <<>> DiG 9.11.3-1ubuntu1.5-Ubuntu <<>> default._domainkey.sender.com txt
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 906
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;default._domainkey.sender.com. IN      TXT

;; ANSWER SECTION:
default._domainkey.sender.com. 86400 IN TXT     "v=DKIM1;h=sha256;k=rsa;p=MIIBI$
ANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAvWQL2VqOiT/7bI7nDIJG/IAjRb9bx603b249Cpxoo$
CIKLh6zWabZQYdCbo8kci4rPr1AoC7vTUnFHT1PcSLKA6UpRqe2+2hn9jiDf/3mlXMeKSxUZXAM9jlI$
1dDp0rlgYwavix1LPrd6VoxR2uhkDw2FKDZDxNH9BVDohQtbDb4zlSsBwz6" "ufZ1kPelFkohtiYLu$
TFKC662CKIaYVSnklXHhl+ie9n68qHlcXvd6ssbm7Am2k85p3aJGQVJ79gK9bFJdBiHZjrU5V+3+gB7$
RIKNG69sAv4ggEGKj1SNtPRULhNeXrH963MRQzP5Gw+8t/iejXoJYxE9dqodXGQIDAQAB"

;; Query time: 4 msec
[0] 0:[tmux]*                                   "receiver" 14:51 21-Apr-19
[1] 0:ssh- 1:ssh*                                          Sun 21 Apr 22:51
```

dig default._domainkey.sender.com txt

3.7 Sending Email with DKIM

The IP address of `sender` was reset to `10.0.2.7` in `netplan` before proceeding.

An email was sent from `sender` to `receiver`:

```
echo "Hello from User 1 with SPF and DKIM" | mail -s "Test SPF and DKIM" -a "From:user1@sender.com" user2@receiver.com
```

The email was received with a `dkim=pass` header.

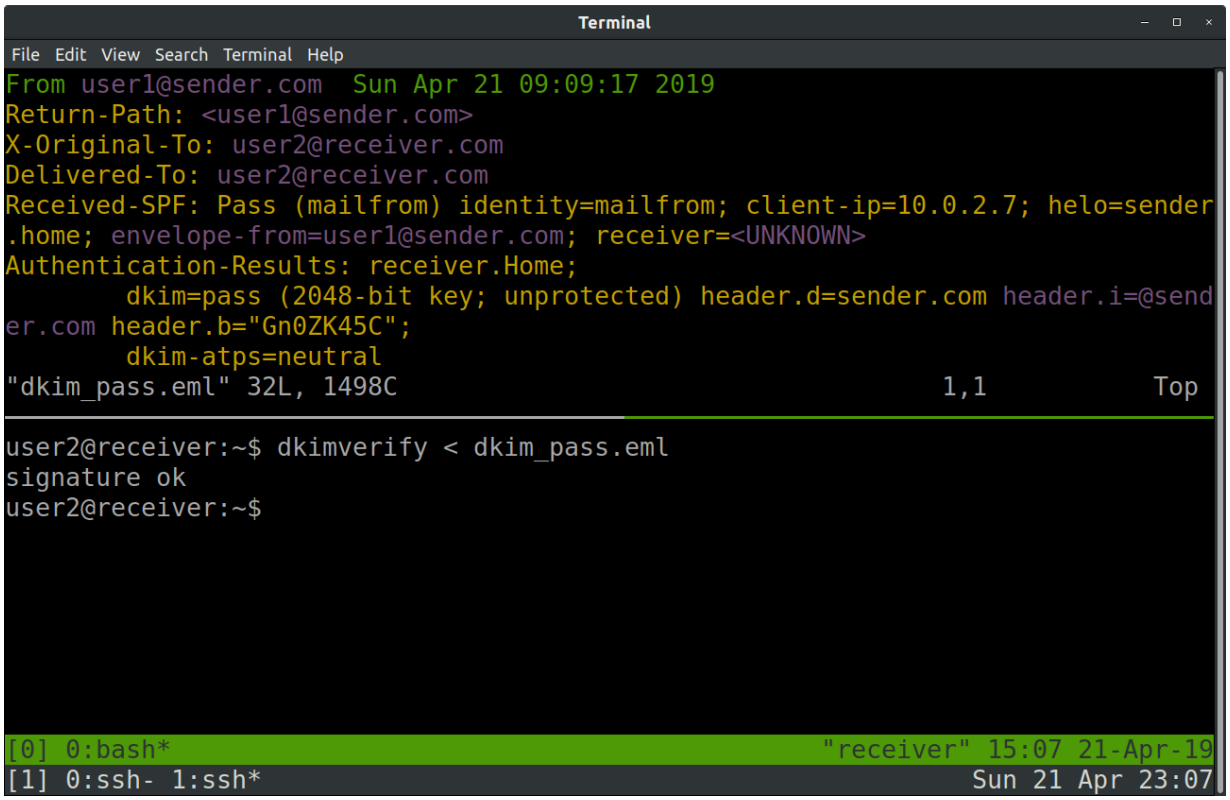
```
Terminal
File  Edit  View  Search  Terminal  Help
From user1@sender.com  Sun Apr 21 09:09:17 2019                          [13/274]
Return-Path: <user1@sender.com>
X-Original-To: user2@receiver.com
Delivered-To: user2@receiver.com
Received-SPF: Pass (mailfrom) identity=mailfrom; client-ip=10.0.2.7; helo=sende$
.home; envelope-from=user1@sender.com; receiver=<UNKNOWN>
Authentication-Results: receiver.Home;
        dkim=pass (2048-bit key; unprotected) header.d=sender.com header.i=@sen$
er.com header.b="Gn0ZK45C";
        dkim-atps=neutral
Received: from sender.Home (unknown [10.0.2.7])
        by receiver.Home (Postfix) with ESMTP id C90D8823CD
        for <user2@receiver.com>; Sun, 21 Apr 2019 09:09:16 +0000 (UTC)
Received: by sender.Home (Postfix, from userid 1000)
        id BEC948214C; Sun, 21 Apr 2019 09:09:16 +0000 (UTC)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=sender.com;
        s=default; t=1555837756;
        bh=9KQC2cIIfpJB4fesfFePkZ69SQp/gXE8KvmZEiRAAac=;
        h=Subject:From:To:Date:From;
        b=Gn0ZK45C+JAm/fzQ0W3G4UVyaeVakp6erlu+tg2MqBpdD4uINsebVJ0P5Rtv5s45y
         AwOyivF2XPSPgyJSuwPcFiQTuSPYDnsEvUoUQpAiXHpeA9Ym/5Dwxm9TcXYYl6jCaY
         cbYE3wZihz20gWkMv0Hf7rbPCIRUSMa+ouscbZC1+Fkv0xHNC9vd8iHhyOQLxzm73S
[0] 0:[tmux]*                                      "receiver" 14:59 21-Apr-19
[1] 0:ssh- 1:ssh*                                            Sun 21 Apr 22:59
```

dkim=pass (2048-bit key; unprotected)

The content of the email was saved to dkim_pass.eml.

The email was checked against the local DNS:

```
dkimverify < dkim_pass.eml
```



Verified DKIM signature

3.8 DKIM Verification Failure

To trigger a `DKIM` verification failure, the public key in `receiver`'s `/var/cache/bind/db.sender.com` was modified. Services were restarted as above.

An email was sent from `sender` to `receiver`:

```
echo "Hello from User 1 with failing DKIM" | mail -s "Test failing DKIM" -a "From:user1@sender.com" user2@receiver.com
```

The email was received with a `dkim=permerror` header, which indicates that the message could not be verified as the email signature did not match the public key retrieved from `default._domainkey.sender.com`.

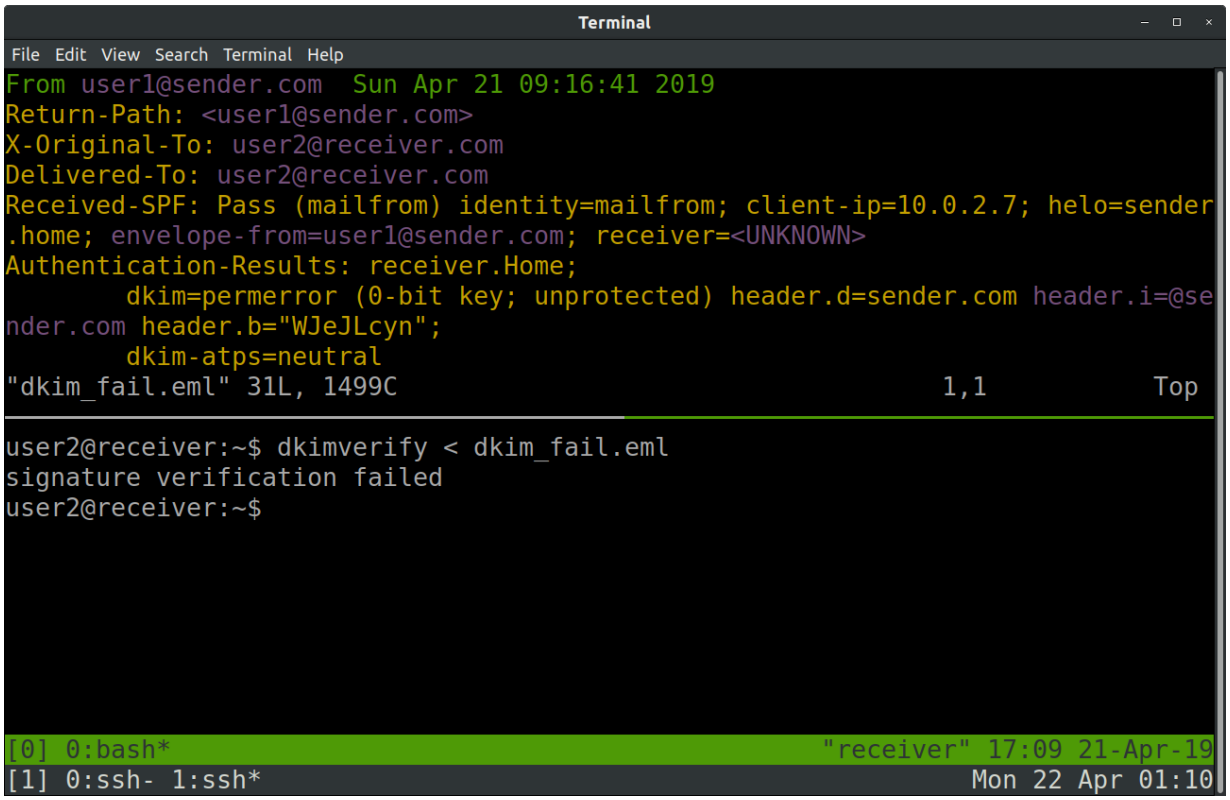
```
Terminal
File  Edit  View  Search  Terminal  Help
From user1@sender.com  Sun Apr 21 09:16:41 2019                          [12/51]
Return-Path: <user1@sender.com>
X-Original-To: user2@receiver.com
Delivered-To: user2@receiver.com
Received-SPF: Pass (mailfrom) identity=mailfrom; client-ip=10.0.2.7; helo=sende$
.home; envelope-from=user1@sender.com; receiver=<UNKNOWN>
Authentication-Results: receiver.Home;
        dkim=permerror (0-bit key; unprotected) header.d=sender.com header.i=@se
nder.com header.b="WJeJLcyn";
        dkim-atps=neutral
Received: from sender.Home (unknown [10.0.2.7])
        by receiver.Home (Postfix) with ESMTP id 5ADA1823CA
        for <user2@receiver.com>; Sun, 21 Apr 2019 09:16:39 +0000 (UTC)
Received: by sender.Home (Postfix, from userid 1000)
        id 5ACE78214C; Sun, 21 Apr 2019 09:16:39 +0000 (UTC)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=sender.com;
        s=default; t=1555838199;
        bh=8SFfPZEPsx+ZH1a2mUps0h1EZx8juWs3CmtSEzwva14=;
        h=Subject:From:To:Date:From;
        b=WJeJLcynNAwE6OUssl754Pdon+aYUoYr8ydBJVblirIl9V/xFMzaD2RM/jQ0p77tl
         wXY4UEb+3jSm/BEBWYTgjOIqz2jDmt9a2yD0YXyh5F/skQphnfMaw//MYhjn8y+pQo
         UH8NaM1D5b0qpFEWa3HxmKjpVclTAg2NB/PcC8StP4GSYgQtrZt1ghC3LS0OCSHxJu
[0] 0:bash- 1:[tmux]*                                "receiver" 17:05 21-Apr-19
[1] 0:ssh- 1:ssh*                                                Mon 22 Apr 01:05
```

dkim=permerror (0-bit key; unprotected)

The content of the email was saved to dkim_fail.eml.

The email was checked against the local DNS:

```
dkimverify < dkim_fail.eml
```



Invalid DKIM signature

4 Discussion

4.1 Digging TXT Records

Based on `DIG TXT` for SUTD, how many IP addresses are permitted to send email on behalf of `sutd.edu.sg`?

Retrieving and following all SPF rules on `sutd.edu.sg`:

```
sutd.edu.sg.            3599    IN      TXT     "v=spf1 ip4:103.24.77.20 ip4:202.94.70.20 include:spf.protection.outlook.com -all"
spf.protection.outlook.com. 181 IN      TXT     "v=spf1 ip4:207.46.100.0/24 ip4:207.46.163.0/24 ip4:65.55.169.0/24 ip4:157.56.110.0/23 ip4:157.55.234.0/24 ip4:213.199.154.0/24 ip4:213.199.180.128/26 ip4:52.100.0.0/14 include:spfa.protection.outlook.com -all"
spfa.protection.outlook.com. 482 IN     TXT     "v=spf1 ip4:157.56.112.0/24 ip4:207.46.51.64/26 ip4:64.4.22.64/26 ip4:40.92.0.0/15 ip4:40.107.0.0/16 ip4:134.170.140.0/24 include:spfb.protection.outlook.com ip6:2001:489a:2202::/48 -all"
spfb.protection.outlook.com. 260 IN     TXT     "v=spf1 ip6:2a01:111:f400::/48 ip4:23.103.128.0/19 ip4:23.103.198.0/23 ip4:65.55.88.0/24 ip4:104.47.0.0/17 ip4:23.103.200.0/21 ip4:23.103.208.0/21 ip4:23.103.191.0/24 ip4:216.32.180.0/23 ip4:94.245.120.64/26 -all"
```

The number of allowed IPv4 addresses permitted to send email for `sutd.edu.sg` is 507,906.

The number of allowed IPv6 addresses permitted to send email for `sutd.edu.sg` is over 2.4E24.

The breakdown as follows:

IPv4/6 Address	Address Count
103.24.77.20	1
202.94.70.20	1
207.46.100.0/24	256
207.46.163.0/24	256
65.55.169.0/24	256
157.56.110.0/23	512
157.55.234.0/24	256

213.199.154.0/24	256
213.199.180.128/26	64
52.100.0.0/14	262144
157.56.112.0/24	256
207.46.51.64/26	64
64.4.22.64/26	64
40.92.0.0/15	131072
40.107.0.0/16	65536
134.170.140.0/24	256
23.103.128.0/19	8192
23.103.198.0/23	512
65.55.88.0/24	256
104.47.0.0/17	32768
23.103.200.0/21	2048
23.103.208.0/21	2048
23.103.191.0/24	256
216.32.180.0/23	512
94.245.120.64/26	64
Sub-total	507906
2001:489a:2202::/48	1.20892581961463E+024
2a01:111:f400::/48	1.20892581961463E+024
Sub-total	2.41785163922926E+024
TOTAL	2.41785163922926E+024

4.2 DKIM TXT Entry Tags

Can you explain the significance of all the tags in your DKIM entry (v, a, c, d, s, t, bh, h, b)?

The definitions of the DKIM signature header field tags can be referenced from Section 3.5 of RFC 6376 [5], and are as follows:

- **v:** DKIM version; current implementation is "1"
- **a:** Signature algorithm; current recommendation is "rsa-sha256"
- **c:** Message canonicalization for header/body; "simple" is strict, "relaxed" is less strict
- **d:** Signing Domain Identifier (SDID) responsible for introducing the message; used in querying DNS for public key
- **s:** Selector; used in querying DNS for public key
- **t:** Timestamp of signature; measured in Unix Epoch time
- **bh:** Hash of canonicalized body as limited by body length count
- **h:** Signed header fields; contains complete list of header fields to be signed
- **b:** Signature data in base64; signed hash of the message hash

1 https://www.ubuntu.com/download/server/thank-you?version=18.04.2&architecture=amd64

2 https://bugs.launchpad.net/ubuntu/+source/virtualbox/+bug/1798813

3 https://certified-senders.org/wp-content/uploads/2018/06/DKIM-Recommendations-2018.pdf

4 https://tools.ietf.org/html/rfc1035#section-3.3.14

5 https://tools.ietf.org/html/rfc6376#section-3.5