

# Linux

- [How to verify that SSL for IMAP/SMTP works and a proper SSL certificate is in use](#)
- [Get Numeric File/Directory Permissions](#)
- [Mount SMB Share in fstab Ubuntu](#)
- [Find out what's taking up all the space in your linux install](#)
- [Extract Certificate Information with OpenSSL](#)
- [Run fstrim manually](#)

# How to verify that SSL for IMAP/SMTP works and a proper SSL certificate is in use

Credit [Robert Asibekov](#)

## IMAP via SSL using port 993

- connect to a mail server using openssl:

```
openssl s_client -showcerts -connect mail.example.com:993 -servername mail.example.com
```

- Check output and make sure that a valid certificate is shown:

Server certificate

subject=/OU=Domain Control Validated/OU=PositiveSSL/CN=mail.example.com

issuer=/C=GB/ST=Greater Manchester/L=Salford/O=COMODO CA Limited/CN=PositiveSSL CA 2

- Make sure that you received IMAP server response:

```
* OK [CAPABILITY IMAP4rev1 UIDPLUS CHILDREN NAMESPACE THREAD=ORDEREDSUBJECT  
THREAD=REFERENCES SORT QUOTA AUTH=CRAM-MD5 AUTH=PLAIN IDLE ACL ACL2=UNION] Courier-IMAP  
ready. Copyright 1998-2004 Double Precision, Inc. See COPYING for distribution information.
```

## SMTP via TLS/StartTLS using port 25 or 587

- Connect to a mail server using openssl:

```
openssl s_client -starttls smtp -showcerts -connect mail.example.com:25 -servername mail.example.com
```

- Check output and make sure that a valid certificate is shown:

#### Server certificate

subject=/OU=Domain Control Validated/OU=PositiveSSL/CN=mail.example.com

issuer=/C=GB/ST=Greater Manchester/L=Salford/O=COMODO CA Limited/CN=PositiveSSL CA 2

- Make sure that you received SMTP server response:

250 DSN

# Get Numeric File/Directory Permissions

- Run the following command as root on a file/directory of your choice:

```
stat -c %a /etc/hosts
```

- You should get the numeric permissions of the file/directory output:

```
644
```

# Mount SMB Share in fstab

## Ubuntu

### Username/Password in connect string

- Edit /etc/fstab:

```
vi /etc/fstab
```

- Enter the following:

```
#theShare
//theServer/theShare /mnt/theMount cifs
vers=3.0,username=theUsername,password=thePassword,uid=1000,gid=104,iocharset=utf8,file_mode=0777,
dir_mode=0777,nounix 0 0
```

### Username/Password in credentials file

- Create a credentials file to a directory of your choice:

```
vi /home/username/.smbcredentials
```

- Enter the following entries and adjust the username, password and/or domain to your environment:

```
#username=MyUsername
#password=MyPassword

# OR:
username=MyUsername
password=MyPassword
domain=MYDOMAIN
```

- Save and set the permissions to the .smbcredentials file to 660:

```
chmod 660 /home/username/.smbcredentials
```

- Edit /etc/fstab:

```
vi /etc/fstab
```

- Enter the following:

```
#theShare
//theServer/theShare /mnt/theMount cifs
vers=3.0,credentials=/home/username/.smbcredentials,uid=1000,gid=104,icharset=utf8,file_mode=0777,dir_
mode=0777,nounix 0 0
```

# Find out what's taking up all the space in your linux install

Example for / partition:

```
du -h / | grep '[0-9\.]\++G'
```

# Extract Certificate Information with OpenSSL

## Fetch the X.509 Public Key Certificate File

```
openssl s_client -connect google.com:443 -showcerts </dev/null | openssl x509 -outform pem > googlecert.pem
```

## Decode the Entire Certificate

```
openssl x509 -in googlecert.pem -noout -text
```

## Extract Specific Information from the Certificate

### Extract the Subject

```
openssl x509 -in googlecert.pem -noout -subject subject=CN = *.google.com
```

### Extract the Issuer

```
$ openssl x509 -in googlecert.pem -noout -issuer issuer=C = US, O = Google Trust Services LLC, CN = GTS CA 1C3
```

## Extracting the Extension Fields

```
$ openssl x509 -in googlecert.pem -noout -ext subjectAltName X509v3 Subject Alternative Name:  
DNS:*.google.com, DNS:*.appengine.google.com, DNS:*.bdn.dev, DNS:*.cloud.google.com,  
DNS:*.crowdsourcing.google.com, DNS:*.datacompute.google.com, ...(truncated)
```





# Run fstrim manually

```
fstrim / -v
```