

## Email Stuff

# 1 Manual Email Server Interaction

How to send emails manually. This is essential for troubleshooting automated email systems.

## 2 Basics

```
openssl s_client -debug -starttls smtp -crlf -connect smtp.emailprovider.com:465
then continue with
EHLO hostname
```

```
AUTH
etc...
```

Here's an example:

```
Compression: NONE
Expansion: NONE
No ALPN negotiated
SSL-Session:
  Protocol   : TLSv1.2
  Cipher     : ECDHE-RSA-AES128-GCM-SHA256
  Session-ID: 480E968D62206D936801683C3BB12679DCB5CEDB0B20F4C3C2EB9DC97A54FAF4
  Session-ID-ctx:
  Master-Key: BAB3C9AD7E209D2A979A621731BCB6A0B4368EEE03E2B99CF884D2671E2D62A1A365E
  Key-Arg    : None
  PSK identity: None
  PSK identity hint: None
  SRP username: None
  TLS session ticket lifetime hint: 300 (seconds)
  TLS session ticket:
0000 - 26 e3 0c d9 a7 0e b1 88-52 ce 35 aa a0 8d bd 40   &.....R.5....@
0010 - 0c 39 e4 60 04 59 06 bb-b2 7e 35 f9 56 36 d1 46   .9.'Y...~5.V6.F
0020 - d2 8c 48 29 44 fa 9f 2b-cd af fc 9a 68 b0 b8 63   ..H)D..+....h..c
0030 - 35 06 cd ba 67 57 bf c4-84 19 5a 05 fd a7 62 51   5...gW....Z...bQ
0040 - 1c 63 d7 b0 b4 ce 62 9f-66 13 7b 4c 74 54 86 fa   .c....b.f.{LtT..
```

```

0050 - 3b 31 e1 e1 bd 19 29 e8-76 c2 6c 45 db 1d 32 50 ;1....).v.lE..2P
0060 - 60 63 ec 38 a3 e5 de 26-d6 27 c6 26 60 09 1a 59 'c.8...&.'.&'..Y
0070 - 59 e9 97 b4 07 9f 6c fc-46 cb 9e 0b 40 57 8f 71 Y.....l.F...@W.q
0080 - 20 98 54 06 38 42 47 c1-e6 fd 74 92 dc 11 a5 97 .T.8BG...t.....
0090 - 5d 81 7e 0b 66 b7 8e aa-c0 61 48 0c 90 84 ac 2d ]..~.f....aH....-

```

```

Start Time: 1590893833
Timeout : 300 (sec)
Verify return code: 0 (ok)

```

---

250 PIPELINING

EHLO localhost

write to 0x1c0dd20 [0x1c17853] (45 bytes => 45 (0x2D))

```

0000 - 17 03 03 00 28 2a 65 16-62 f5 2c 8c e5 d9 f2 ef ....(*e.b.,.....
0010 - bb 8a ac 3a 6a dc db 32-79 56 7f 86 90 2e ed 44 ...:j..2yV.....D
0020 - 6e 6a ea 12 64 bb f8 1e-95 7e 63 16 c5 nj..d....~c..

```

read from 0x1c0dd20 [0x1c13303] (5 bytes => 5 (0x5))

```

0000 - 17 03 03 00 a7 .....

```

read from 0x1c0dd20 [0x1c13308] (167 bytes => 167 (0xA7))

```

0000 - 80 ae b0 b8 00 a2 d6 6b-47 42 2c 5d 30 1c 04 bd .....kGB,]0...
0010 - 5c bb 84 61 20 5a 2e 0d-c5 79 28 a9 21 d6 ad 1b \.a Z...y(!...
0020 - 99 e5 fc da a1 fb da bc-70 9e 9b ac 5a e1 0f 8e .....p...Z...
0030 - bb c8 6d cb 2e 27 af 9b-4a 6a e8 17 69 31 63 14 ..m..'..Jj..i1c.
0040 - 24 68 d7 6b cc a7 dc 24-7c d6 4b fa 2a 3e 38 61 $h.k...$|.K.*>8a
0050 - 58 57 4d e8 1c e7 66 3c-22 0f 07 62 42 14 e7 78 XWM...f<"..bB..x
0060 - 86 8a 6c 6f 8f 79 06 f6-dc 33 b5 e9 ec db b8 b4 ..lo.y...3.....
0070 - 0e 7d d7 08 17 cd 4d b3-ac 16 79 08 87 df 31 95 .}....M...y...1.
0080 - 03 8d 9e 8c 25 09 ca 38-d0 be ee 45 38 d8 c4 a5 ....%..8...E8...
0090 - c3 90 44 23 5d e7 9b 1b-c0 40 d2 cc 40 c5 4a e4 ..D#]....@..@.J.
00a0 - 25 38 8b 05 2f b2 33 %8../.3

```

250-www192.vfemail.net

250-8BITMIME

250-AUTH PLAIN

250-BURL imap

250-CHUNKING

250-ENHANCEDSTATUSCODES

250-SIZE 78643200

250 PIPELINING

Here you might enter at the prompt

## AUTH LOGIN

But that will fail:

### AUTH LOGIN

```
write to 0x1c0dd20 [0x1c17853] (41 bytes => 41 (0x29))
0000 - 17 03 03 00 24 2a 65 16-62 f5 2c 8c e6 59 01 74    ....$*e.b.,..Y.t
0010 - 10 99 f5 d2 e6 75 88 d3-b9 58 0e 92 78 fc f1 ad    .....u...X..x...
0020 - c6 aa 50 2a 6c f4 cf e5-dd                        ..P*1....
read from 0x1c0dd20 [0x1c13303] (5 bytes => 5 (0x5))
0000 - 17 03 03 00 3a                                    ....:
read from 0x1c0dd20 [0x1c13308] (58 bytes => 58 (0x3A))
0000 - 80 ae b0 b8 00 a2 d6 6c-bf e5 f8 3d 5a 39 93 2b    .....l...=Z9.+
0010 - f8 a2 12 03 0f 0f 5c a3-95 0a 6c b6 d9 fa 3d 09    .....\.l...=.
0020 - 73 f7 5c 26 8c 41 aa cf-87 0f 65 61 52 7d ec f7    s.\&.A....eaR}..
0030 - 4a fe cd a4 f8 3e be 48-a3 92                    J....>.H..
504 5.5.4 Authentication failed.
```

If you look at the EHLO, it responds with what auth it supports. In this case, AUTH PLAIN. So it accepts plain text passwords (over an encrypted session of course).

Unfortunately, ssmtp (which I'm using) doesn't support plain. from ssmtp manual:

Specifies mechanism for SMTP authentication. (Only LOGIN and CRAM-MD5)

If you instead type AUTH PLAIN or auth plain, it will respond with 334. It is then waiting for credentials.

ref: [https://wiki.zoneminder.com/SMS\\_Notifications](https://wiki.zoneminder.com/SMS_Notifications)

You can also compare logs in /var/log/mail.log Here's a working example:

```
Jun 16 01:01:02 server sSMTP[19958]: Creating SSL connection to host
Jun 16 01:01:02 server sSMTP[19958]: SSL connection using DHE_RSA_AES_256_CBC_SHA1
Jun 16 01:01:04 server sSMTP[19958]: Sent mail for .com (221 2.0.0 Bye) uid=0 usernam
Jun 16 01:01:04 server sSMTP[19978]: Creating SSL connection to host
Jun 16 01:01:05 server sSMTP[19978]: SSL connection using DHE_RSA_AES_256_CBC_SHA1
Jun 16 01:01:06 server sSMTP[19978]: Sent mail for .com (221 2.0.0 Bye) uid=0 usernam
```

```

Jun 18 13:30:31 server sSMTP[18220]: Creating SSL connection to host
Jun 18 13:30:32 server sSMTP[18220]: 220 smtp.mxes.net ESMTP Postfix 03
Jun 18 13:30:32 server sSMTP[18220]: EHLO s.com
Jun 18 13:30:32 server sSMTP[18220]: 250 CHUNKING
Jun 18 13:30:32 server sSMTP[18220]: STARTTLS
Jun 18 13:30:32 server sSMTP[18220]: 220 2.0.0 Ready to start TLS
Jun 18 13:30:32 server sSMTP[18220]: SSL connection using DHE_RSA_AES_256_CBC_SHA1
Jun 18 13:30:32 server sSMTP[18220]: EHLO s.com
Jun 18 13:30:32 server sSMTP[18220]: 250 CHUNKING
Jun 18 13:30:32 server sSMTP[18220]: AUTH LOGIN
Jun 18 13:30:32 server sSMTP[18220]: 334 ls!sls!sls
Jun 18 13:30:32 server sSMTP[18220]: somethuing
Jun 18 13:30:32 server sSMTP[18220]: 334 something else6
Jun 18 13:30:32 server sSMTP[18220]: somehashedstuff
Jun 18 13:30:32 server sSMTP[18220]: 235 2.7.0 Authentication successful
Jun 18 13:30:32 server sSMTP[18220]: MAIL FROM:<person@some.com>
Jun 18 13:30:32 server sSMTP[18220]: 250 2.1.0 Ok
Jun 18 13:30:32 server sSMTP[18220]: RCPT TO:<fgdfg>
Jun 18 13:30:33 server sSMTP[18220]: 250 2.1.5 Ok
Jun 18 13:30:33 server sSMTP[18220]: DATA
Jun 18 13:30:33 server sSMTP[18220]: 354 End data with <CR><LF>.<CR><LF>
Jun 18 13:30:33 server sSMTP[18220]: Received: by .com (sSMTP sendmail emulation); Th
Jun 18 13:30:33 server sSMTP[18220]: From: "C" <c@com>
Jun 18 13:30:33 server sSMTP[18220]: Date: Thu, 18 Jun 2020 13:30:30 -0400
Jun 18 13:30:33 server sSMTP[18220]: To: j
Jun 18 13:30:33 server sSMTP[18220]: Subject: My email check
Jun 18 13:30:33 server sSMTP[18220]:
Jun 18 13:30:33 server sSMTP[18220]: Hello, World
Jun 18 13:30:34 server sSMTP[18220]: .
Jun 18 13:30:34 server sSMTP[18220]: 250 2.0.0 Ok: queued as 3B1D6759AF
Jun 18 13:30:34 server sSMTP[18220]: QUIT
Jun 18 13:30:34 server sSMTP[18220]: 221 2.0.0 Bye
Jun 18 13:30:34 server sSMTP[18220]: Sent mail for c (221 2.0.0 Bye) uid=1000 usernam

```

Here is another non working one.

```

Jun 18 17:35:46 zmctankhome sSMTP[22488]: Set MailHub="smtp.mxes.net"
Jun 18 17:35:46 zmctankhome sSMTP[22488]: via SMTP Port Number="587"
Jun 18 17:35:47 zmctankhome sSMTP[22488]: Creating SSL connection to host

```

```
Jun 18 17:35:47 zmctankhome sSMTP[22488]: 220 smtp.mxes.net ESMTP Postfix 03
Jun 18 17:35:47 zmctankhome sSMTP[22488]: EHLO .com
Jun 18 17:35:47 zmctankhome sSMTP[22488]: 250 CHUNKING
Jun 18 17:35:47 zmctankhome sSMTP[22488]: STARTTLS
Jun 18 17:35:47 zmctankhome sSMTP[22488]: 220 2.0.0 Ready to start TLS
Jun 18 17:35:48 zmctankhome sSMTP[22488]: SSL connection using ECDHE_RSA_AES_256_GCM_
Jun 18 17:35:48 zmctankhome sSMTP[22488]: EHLO .com
Jun 18 17:35:48 zmctankhome sSMTP[22488]:
Jun 18 17:35:48 zmctankhome sSMTP[22488]: (.com)
```

Notice the TLS / SSL encryption is different. That can be a deal breaker.  
Ouch. Otherwise both are Debian (one is 9, the other 10).