首先是文献介绍，讲得很详细。讲了“是什么”“为什么”“怎么做”这三个问题。特别是为什么要搞一个STARTTLS这个东西。

SSL vs TLS vs STARTTLS

There's often quite a bit of confusion around the different terms **SSL**, **TLS** and **STARTTLS**.

[SSL and TLS](http://hello.k7mm.com/5i5.php?u=T2k4dlpXNHVkMmxyYVhCbFpHbGhMbTl5Wnk5M2FXdHBMMVJ5WVc1emNHOXlkRjlNWVhsbGNsOVRaV04xY21sMGVRPT0%3D&b=13) both provide a way to encrypt a communication channel between two computers (e.g. your computer and our server). TLS is the successor to SSL and the terms SSL and TLS are used interchangeably unless you're referring to a specific version of the protocol.

[STARTTLS](http://hello.k7mm.com/5i5.php?u=T2k4dlpXNHVkMmxyYVhCbFpHbGhMbTl5Wnk5M2FXdHBMMU4wWVhKMGRHeHo%3D&b=13) is a way to take an existing insecure connection and upgrade it to a secure connection using SSL/TLS. Note that despite having TLS in the name, STARTTLS doesn't mean you have to use TLS, you can use SSL.

SSL/TLS version numbers

Version numbering is inconsistent between SSL and TLS versions. When TLS took over from SSL as the preferred protocol name, it began a new version number, and also began using sub-versions. So the ordering of protocols in terms of oldest to newest is: SSL v2, SSL v3, TLS v1.0, TLS v1.1, TLS v1.2.

When you connect to an SSL/TLS encrypted port, or use STARTTLS to upgrade an existing connection, both sides will negotiate which protocol and which version to use based on what has been configured in the software and what each side supports.

Support for SSL/TLS is virtually universal these days, however which versions are supported is variable. Pretty much everything supports SSL v3 (except a few very old Palm Treo devices [as we discovered](http://hello.k7mm.com/5i5.php?u=T2k4dllteHZaeTVtWVhOMGJXRnBiQzVqYjIwdk1qQXdPUzh4TVM4eE9DOXpjR1ZqYVdGc0xYTnpiQzFvYjNOMGJtRnRaUzFtYjNJdGIyeGtMV05zYVdWdWRITXY%3D&b=13)). Most things support TLS v1.0. As at May 2012, [support for TLS v1.1 and TLS v1.2 is more limited](http://hello.k7mm.com/5i5.php?u=T2k4dlpXNHVkMmxyYVhCbFpHbGhMbTl5Wnk5M2FXdHBMMVJ5WVc1emNHOXlkRjlNWVhsbGNsOVRaV04xY21sMGVRPT0%3D&b=13#Browser_implementations).

TLS vs STARTTLS naming problem

One significant complicating factor is that some email software **incorrectly** uses the term TLS when they should have used STARTTLS. Older versions of Thunderbird in particular used "TLS" to mean "enforce use of STARTTLS to upgrade the connection, and fail if STARTTLS is not supported" and "TLS, if available" to mean "use STARTTLS to upgrade the connection if the server advertises support for it, otherwise just use an insecure connection".

SSL/TLS vs plaintext/STARTTLS port numbers

The above is particularly problematic when combined with having to configure a port number for each protocol.

To add security to some existing protocols (e.g. IMAP, POP, etc.), it was decided to just add SSL/TLS encryption as a layer underneath the existing protocol. However, to distinguish that software should talk the SSL/TLS encrypted version of the protocol rather than the plaintext one, a different port number was used for each protocol. So you have:

* IMAP uses port 143, but SSL/TLS encrypted IMAP uses port 993.
* POP uses port 110, but SSL/TLS encrypted POP uses port 995.
* SMTP uses port 25, but SSL/TLS encrypted SMTP uses port 465.

At some point, it was decided that having 2 ports for every protocol was wasteful, and instead you should have 1 port that starts off as plaintext, but the client can upgrade the connection to an SSL/TLS encrypted one. This is what STARTTLS was created to do.

There were a few problems with this though. There was already existing software that used the alternate port numbers with pure SSL/TLS connections. Client software can be very long lived, so you can't just disable the encrypted ports until all software has been upgraded.

Mechanisms were added to each protocol to tell clients that the plaintext protocol supported upgrading to SSL/TLS (i.e. STARTTLS), and that they should not attempt to log in without doing the STARTTLS upgrade. This created two unfortunate situations:

1. Some software just ignored the "login disabled until upgraded"announcement and just tried to log in anyway, sending the username and password over plaintext. Even if the server then rejected the login, the details had already been sent over the Internet in plaintext.
2. Other software saw the "login disabled until upgraded" announcement, but then wouldn't upgrade the connection automatically, and thus reported login errors back to the user, which caused confusion about what was wrong.

Both of these problems resulted in significant compatibility issues with existing clients, and so most system administrators continued to just use plaintext connections on one port number, and encrypted connections on a separate port number.

This has now basically become the de facto standard that everyone uses. IMAP SSL/TLS encrypted over port 993 or POP SSL/TLS encrypted over port 995. Many sites (including FastMail) now disable plain IMAP (port 143) and plain POP (port 110) altogether so people **must** use an SSL/TLS encrypted connection. By disabling ports 143 and 110, this removes completely STARTTLS as even an option for IMAP/POP connections.

SMTP STARTTLS as an exception

The one real exception to the above is SMTP. However that's for a different reason again. Most email software used SMTP on port 25 to submit messages to the email server for onward transmission to the destination. However, SMTP was originally designed for transfer, not submission. So yet another port (587) was [defined for message submission](http://hello.k7mm.com/5i5.php?u=T2k4dmRHOXZiSE11YVdWMFppNXZjbWN2YUhSdGJDOXlabU15TkRjMg%3D%3D&b=13). Although port 587 doesn't mandate requiring STARTTLS, the use of port 587 became popular around the same time as the realisation that SSL/TLS encryption of communications between clients and servers was an important security and privacy issue.

The result is that in most cases, systems that offer message submission over port 587 **require**clients to use STARTLS to upgrade the connection and also require a username and password to authenticate. There has been an added benefit to this approach as well. By moving users away from using port 25 for email submission, ISPs are now able to block outgoing port 25 connections from users' computers, which were a significant source of spam due to infection with spam-sending viruses.

Currently, things seem relatively randomly split between people using SMTP SSL/TLS encrypted over port 465, and people using SMTP with STARTTLS upgrading over port 587.

从“When you connect to an SSL/TLS encrypted port, or use STARTTLS to upgrade an existing connection, both sides will…”可以知道，有两种方式进行安全的传输，一个是另开一个完全用ssl加密的端口（比如，IMAP 明文是143端口，ssl加密的是993端口），还有一种是在原来的明文端口上进行“升级”，通过 STARTTLS命令（或者说是协议，里面涉及到tls handshake）。如果大家都用第一种方法的话，每个明文协议端口都需要另外配备一个加密的端口用作加密传输，比如上面所说的：

* IMAP uses port 143, but SSL/TLS encrypted IMAP uses port 993.
* POP uses port 110, but SSL/TLS encrypted POP uses port 995.
* SMTP uses port 25, but SSL/TLS encrypted SMTP uses port 465.

这样的话岂不是很浪费？

因此，第二种方法才是比较合理的，即不开辟任何新的端口，在原端口的基础上，通过STARTTLS进行加密升级，升级到SSL加密传输，而我们用的还是明文的端口！

这就是STARTTLS的由来，但是由于各种软件兼容性的问题，大部分还是用的方法一，只有在SMTP中STARTTLS得到了广泛的应用。为什么只有SMTP可以用STARTTLS呢？上面也说的很清楚了。以下就是gmail->帮助 中的一个例子也应征了这个问题：

# 以其他地址发送：选择端口号和身份验证类型

如果您想使用其他域的 SMTP 服务器并以不同的电子邮件地址来发送邮件，则需要选择相应的邮件提供商支持的端口和身份验证协议。在某些情况下，邮件提供商不支持这些推荐的设置，此时您可以尝试使用其他的端口和协议。如果所有安全的选项都不可用，您可以考虑选择不安全的选项。

以下是有关不同的身份验证协议的详细信息：

### SSL 和 TLS 有什么不同？

#### SSL

SSL 代表安全套接字层 (Secure Sockets Layer)。当使用 SMTP 时，这种 SSL 模式通常是指在连接邮件服务器时要求使用安全连接。与 SSL 相关联的端口通常是 465，但也可能使用其他端口。我们建议您最好与相应的邮件提供商确认，以获取详细信息。

#### TLS（较为正式的名称是 STARTTLS）

TLS 代表传输层安全 (Transport Layer Security)。当使用 SMTP 时，这种类型的身份验证协议会先与服务器建立一个不安全的连接，再执行 STARTTLS 命令，然后在实际传输数据时升级为安全连接。与 TLS 相关联的端口通常是 25 和 587，但也可能使用其他端口。我们建议您最好与相应的邮件提供商确认，以获取详细信息。

请注意，SSL 和 TLS 都是有效的安全协议，可确保您的数据（如用户名、密码和邮件）会进行加密。也就是说，正常的语言会变为代码，因此任何人都无法读取您发送的内容。当您添加其他发件人地址时，更重要的是知道相应的邮件提供商支持的端口和身份验证协议。

### 如果我的邮件提供商不支持通过 SSL 或 TLS 连接到其邮件服务器，这意味着什么？

如果您的邮件提供商不支持通过安全方式连接到其邮件服务器，我们支持使用端口 25 的不安全连接。使用不安全的连接时，不会对您的信息进行编码，这意味着有人可能会在您连接时看到您的登录凭据和邮件数据。

### 使用不安全的连接情况会很糟吗？

建议您不要使用，但每个邮件提供商和 ISP 是不同的，他们会支持不同类型的连接。如果您的邮件提供商或 ISP 不支持安全连接，您可能需要考虑通过 Gmail 的服务器并使用自定义的“收件人”来收发邮件。

请注意，如果您选择使用备用地址并通过 Gmail 的服务器来发送邮件，您的 Gmail 地址仍会包含在电子邮件标头的“发件人”字段中，以防您的邮件被标记为垃圾邮件。大多数电子邮件客户端不显示“发件人”字段，但某些版本的 Microsoft Outlook 可能会显示“发件人：<用户名>@gmail.com，代表 <自定义地址>@<域名>.com”。

最后，看看维基百科是怎么介绍STARTTLS的（现在看来确实有一语中的之感）：

**STARTTLS** is an extension to plain text communication protocols, which offers a way to upgrade a plain text connection to an encrypted ([TLS](http://hello.k7mm.com/5i5.php?u=T2k4dlpXNHVkMmxyYVhCbFpHbGhMbTl5Wnk5M2FXdHBMMVJ5WVc1emNHOXlkRjlNWVhsbGNsOVRaV04xY21sMGVRPT0%3D&b=13) or [SSL](http://hello.k7mm.com/5i5.php?u=T2k4dlpXNHVkMmxyYVhCbFpHbGhMbTl5Wnk5M2FXdHBMMU5sWTNWeVpWOVRiMk5yWlhSZlRHRjVaWEk9&b=13)) connection instead of using a separate port for encrypted communication.