

## TLS, SSL and HTTP

#####  
The article was obtained at the following URL: <http://www.kfwebs.net/articles/article/35>  
The article might be distributed further as long as it is provided as it is, with the credits stated.  
The Article was written and first published by KF Webs, at <http://www.kfwebs.net>  
#####

*Both TLS and SSL provide the same service. It enables you to encrypt the communication and to verify the host against a central authority. There are however some differences in implementations that especially come in effect when you are a hosting provider.*

*Added: 2006-03-13 22:48:56 - Modified: 2006-03-31 08:48:43 - Level: Intermediate*

## Table of Contents

1. [Introduction](#)
  - ◆ [Apache 1.3.x and SSL](#)
2. [Why should HTTP be able to use TLS](#)
3. [How to work around the current limitations](#)
4. [The road ahead](#)

## Introduction

Both TLS and SSL provide the same service. It enables you to encrypt the communication and to verify that the host is verified by a central authority (CA) such as Verisign.

The ability to use encrypted content is very important in order to safely transfer personal information across the Internet. This can range from your social security number, your internet bank log in credentials or your password on your web host.

This is something that should be considered whenever communicating, especially across the Internet, including the use of email.



The major difference between the SSL and TLS implementations will however only come to effect if you are a hosting provider.

TLS is an abbreviation for Transport Layer Security. TLS is commonly used for amongst other things sending emails using the Simple Mail Transport Protocol (SMTP). The Internet Engineering Task Force has the current website for the [TLS Charter](#).

Currently websites that want to encrypt the communication use the Secure Socket Layer (SSL) protocol. This is what is in effect whenever you see an address starting with https:// and a lock icon appears.

## Apache 1.3.x and SSL

Using SSL on a virtual host in Apache 1.3.x seem to disable the use of Apache's "graceful" restart. You will have to entirely stop and start Apache or it will simply die without an error message. One administrator stated:

*I finally connected "adding a new client" with "server stopping that weekend" and from there worked out that adding new SSL vhosts caused graceful to kill the server instead of restart it*

## Why should HTTP be able to use TLS

The main difference between SSL and TLS is how the connection is established. Whereby SSL negotiate an SSL connection before connecting, TLS connect first then negotiate for TLS. Using SSL non-secured data is ordinarily connected on port 80, while secured (SSL) data is on port 443. As opposed the current implementation for TLS and sending emails can be both secured and un-secured on port 25.

This difference in connection become important if you host more than one website. As of today you require two IP-addresses if you wish to enable SSL for both hostA.com and hostB.com. This is because both would require to be bound to port 443, unless you wish to use a non-standard port that might be firewalled and possible be inconvenient as you have to specify the port in the address.

Where SSL will only enable you to host one SSL-enabled website on the same IP, using TLS would enable you to host multiple secure websites.

## How to work around the current limitations

One way to get around the current limitations with regards to SSL would be to use the Mass Virtual Hosting approach discussed priorly in order to create https://SUBDOMAIN.our-secure-host.com and using a wildcard SSL Certificate for \*.our-secure-host.com.

KF Webs recently published a story about [Mass Virtual Hosting using Apache](#) and it was a discussion about that that led to this article.

This approach, however, introduce some issues. For one thing users will be redirected to a website that has a different host name, which could open up the possibility for [Phishing](#)

## The road ahead

HTTP over TLS is currently discussed in the [IETF Request for Comment 2818](#). Hopefully it will be implemented in the major browsers and be ready to start using within a reasonable time frame as it would improve security for many users.



KF Webs is not the only entity that feel that TLS should be used. This can be found in [RFC 2817: Upgrading to TLS Within HTTP/1.1](#):

#### Abstract

This memo explains how to use the Upgrade mechanism in HTTP/1.1 to initiate Transport Layer Security (TLS) over an existing TCP connection. This allows unsecured and secured HTTP traffic to share the same well known port (in this case, http: at 80 rather than https: at 443). It also enables "virtual hosting", so a single HTTP + TLS server can disambiguate traffic intended for several hostnames at a single IP address.

Although RFC1817 was published in May 2000 and state that *At the Washington DC IETF meeting in December 1997, the Applications Area Directors and the IESG reaffirmed that the practice of issuing parallel "secure" port numbers should be deprecated. The HTTP/1.1 Upgrade mechanism can apply Transport Layer Security [6] to an open HTTP connection.* this still has not happened 9 years later, which is one of the reasons KF Webs would like to put focus on the matter.

## Related articles: