SSL Offloading \ SSL Termination:

<https://www.f5.com/services/resources/glossary/ssl-offloading>

SSL offloading is the process of removing the SSL-based encryption from incoming traffic to relieve a web server of the processing burden of decrypting and/or encrypting traffic sent via SSL. The processing is offloaded to a separate device designed specifically for [SSL acceleration](https://www.f5.com/services/resources/glossary/ssl-acceleration) or [SSL termination](https://www.f5.com/services/resources/glossary/ssl-termination).

SSL termination is particularly useful when used with clusters of [SSL VPN](https://www.f5.com/services/resources/glossary/ssl-vpn)s, because it greatly increases the number of connections a cluster can handle.

SSL termination refers to the process of decrypting encrypted traffic before passing it along to a web server.

**What is SSL termination?**

Approximately 90% of web pages are now encrypted with the SSL (Secure Sockets Layer) protocol and its modern, more secure replacement TLS (Transport Layer Security). This is a positive development in terms of security because it prevents attackers from stealing or tampering with data exchanged between a web browser and a web or application server. But, decrypting all that encrypted traffic takes a lot of computational power—and the more encrypted pages your server needs to decrypt, the larger the burden.

SSL termination (or [SSL offloading](https://www.f5.com/services/resources/glossary/ssl-offloading)) is the process of decrypting this encrypted traffic. Instead of relying upon the web server to do this computationally intensive work, you can use SSL termination to reduce the load on your servers, speed up the process, and allow the web server to focus on its core responsibility of delivering web content.

**How does SSL termination work?**

SSL termination works by intercepting the encrypted traffic before it hits your servers, then decrypting and analyzing that traffic on an [Application Delivery Controller](https://devcentral.f5.com/s/articles/what-is-an-application-delivery-controller-part-1-24742) (ADC) or dedicated SSL termination device instead of the app server. An ADC is much better equipped to handle the demanding task of decrypting multiple SSL connections, leaving the server free to work on application processing.

**Why is SSL termination important?**

Many security inspection devices have trouble scaling to handle the tidal wave of malicious traffic, much less decrypting, inspecting, and then re-encrypting it again. Using an ADC or dedicated SSL termination device to decrypt encrypted traffic ensures that your security devices can focus on the work they were built to do.

In addition, by using SSL termination, you can empower your web or app servers to manage many connections at one time, while simplifying complexity and eliminating performance degradation. SSL termination is particularly useful when used with clusters of SSL VPNs, because it greatly increases the number of connections a cluster can handle.

Offloading SSL or TLS traffic to an ADC or dedicated device enables you to boost the performance of your web applications while ensuring that encrypted traffic remains secure.

<https://www.thesslstore.com/blog/ssl-offloading-bridging-termination/>

Before TLS 1.3, even before TLS 1.2, frankly, SSL/TLS used to legitimately add latency to connections. That’s what lent itself to the perception that SSL/TLS slowed down websites. Ten years ago, that was the knock on SSL certificates. “Oh they slow down your site.” And that was true at the time. It’s not today, but in the past SSL/TLS was considered a bit resource hungry. For starters, you have the SSL/TLS handshake. It’s been refined to where it’s now a single roundtrip in TLS 1.3, but before that it took several roundtrips. Then, following the handshake, additional processing power had to be exerted to encrypt and decrypt the data being transmitted. As the additional load from SSL/TLS increases on the server, it’s no longer able to process at full capacity. Again, a lot of this has been cleaned up in TLS 1.3, and HTTP/2 – which requires the use of SSL/TLS – helps to increase performance even more, but even with all of those improvements, SSL/TLS can still add latency with higher volumes of traffic. So, what is SSL offloading? Well, to help offset the extra burden SSL/TLS adds, you can spin up separate Application-Specific Integrated Circuit (ASIC) processers that are limited to just performing the functions required for SSL/TLS, namely the handshake and the encryption/decryption. This frees up processing power for the intended application or website. That’s SSL offloading in a nutshell. Sometimes it’s also called load balancing. You may hear the term load balancer tossed around. A load balancer is any device that helps improve the distribution of workloads across multiple resources, for instance distributing the SSL/TLS workload to ASIC processors. What are the advantages of SSL offloading? SSL offloading has several benefits: It offloads additional tasks from your application servers so they can focus on their primary functions. It saves resources on those application servers. And, depending on what load balancer you’re using, it can also help with HTTPS inspection, reverse-proxying, cookie persistence, traffic regulation, etc. That last one is one of the most important: that in some cases SSL offloading can assist with traffic inspection. As important as encryption is, it has one major drawback: attackers can hide in your encrypted traffic. There’s no shortage of high-profile exploits that have occurred as a result of attackers hiding in HTTPS traffic, recently Magecart has been using HTTPS traffic to obfuscate the PCI it’s been exfiltrating from various payment pages. Being able to inspect HTTPS traffic becomes almost compulsory once your organization reaches a certain size, and one of the best ways to do that is to offload your SSL/TLS processes. How does SSL offloading work? When we talk about SSL offloading there are two different ways to accomplish it: SSL Termination SSL Bridging Let’s start with SSL termination first because it’s a little bit simpler. Essentially it works this way, the proxy server or load balancer you use for the SSL offloading acts as the SSL terminator, which also acts as an edge device. When a client attempts to connect to a website, the client connects to the SSL terminator—that connection is HTTPS. But the connection between the SSL terminator and the application server is via HTTP. Now, you may be asking how that doesn’t cause problems with the browser, it’s because the HTTP connection is taking place behind the scenes – on the internal network, protected by firewalls – the client still has a secure connection with the SSL terminator, which is acting as a pass-through. Here’s a visualization of SSL Termination: SSL Bridging is extremely similar conceptually, except rather than sending the traffic and requests on via HTTP, it re-encrypts everything before sending it to the application server. Here’s a visualization of SSL Bridging: Both allow you to perform traffic inspection and can help tremendously when you’re dealing with high volumes of traffic on larger networks.  
  
Read more at: https://www.thesslstore.com/blog/ssl-offloading-bridging-termination/