Análise das versões do TLS para o PHP (VERSÃO PÚBLICA)

Sumário

1.TLS

TLS é o acrônimo de *Transport Layer Security* (Segurança em Camada de Transporte), um protocolo publicado em 1999 na RFC 2246. O objetivo principal do protocolo TLS é fornecer privacidade e integridade de dados entre dois aplicativos em comunicação. O protocolo é composto por duas camadas: o protocolo de registro TLS e o protocolo de handshake TLS. No nível mais baixo, sobreposto a algum protocolo de transporte confiável (por exemplo, TCP[TCP]), está o protocolo de registro TLS. A Figura 1 mostra como o TLS é implementado na camada de Sessão do modelo Open systems interconnection (OSI)[1].

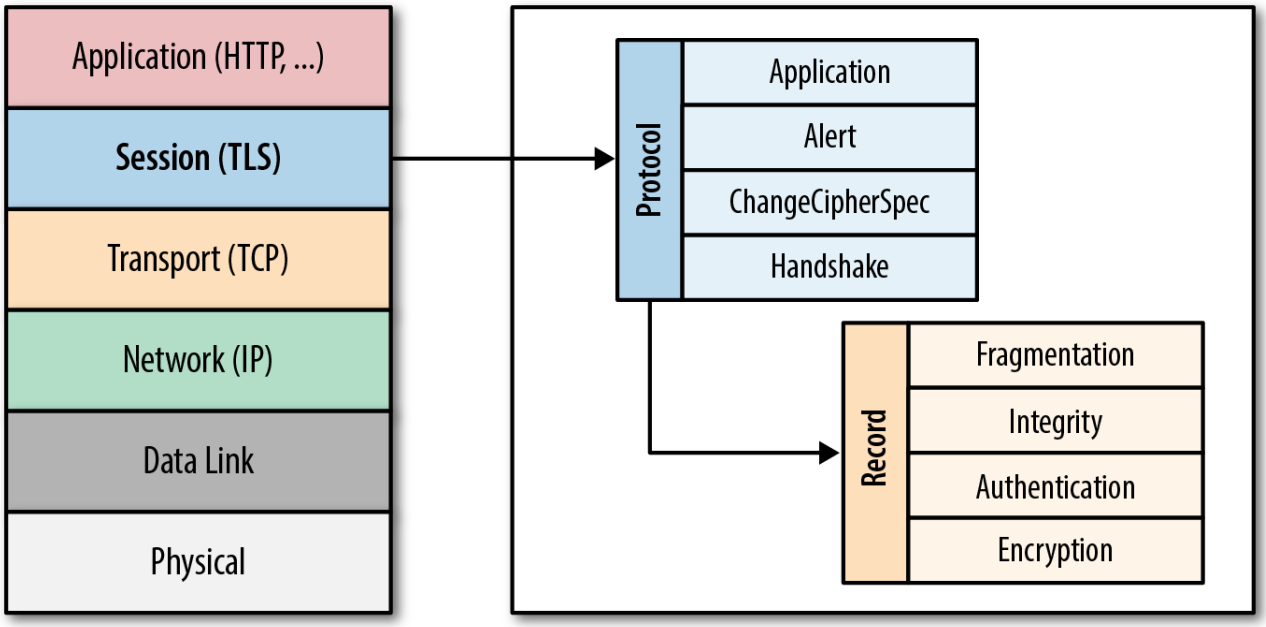

Figura 1: Arquitetura do protocolo Transport Layer Security (TLS)

1.1. Versões do TLS

O protocolo tem mais de uma versão, e assim, mais de uma implementação disponível em diversas linguagens de programação. O uso da versão mais recente é recomendável porque ela implementa o estado da arte da segurança em comunicação.

Tabela 1: Versões do TLS

Versão	RFC	Definida em
1.0	2246	1999
1.1	4346	2006
1.2	5246	2008
1.3	8446	2018

1 A especificação OSI está disponível em: https://www.iso.org/ics/35.100/x/

Há uma implementação de TLS em código livre e aberto, o OpenSSL. No lugar de implementar diretamente o TLS, as linguagens de programação podem usar o OpenSSL. A versão estável mais recente do OpenSSL é a série 3.2 com suporte até 23/11/2025. Também está disponível a série 3.1 com suporte até 14/03/2025, e a série 3.0 que é uma versão de suporte de longo prazo (LTS) com suporte até 07/09/2026. Todas as versões mais antigas versões (incluindo 1.1.1, 1.1.0, 1.0.2, 1.0.0 e 0.9.8) agora estão sem suporte e não devem ser usadas. Os usuários dessas versões mais antigas são incentivados a atualizar para 3.2 ou 3.0 o mais rápido possível. O sítio do OpenSSL contém a lista de vulnerabilidades da implementação e quais são as versões que as corrigem. A Tabela 2 mostra em quais versões de OpenSSL foi adicionado suporte às versões de TLS.

Tabela 2: Versões do OpenSSL e suporte a TLS

OpenSSL	Lançamento	Adicionou suporte a TLS
0.9.0	Não houve	1.0
0.9.1	Não houve	1.1
1.0.1	14/03/2012	1.2
3.0	07/09/2021	1.3

SSL é o acrônimo de *Secure Sockets Layer*, um protocolo publicado na versão 2.0 por uma equipe da Netscape liderada pelo criptógrafo egípcio Taher Elgamal, em 1995. O protocolo TLS foi lançado em 1999 como sucessor do SSL 3.0. O TLS é o sucessor direto do SSL, e todas as versões do SSL agora estão obsoletas. No entanto, é comum encontrar o termo *SSL* descrevendo uma conexão TLS (como no caso de funções e argumentos da linguagem PHP). Na maioria dos casos, os termos *SSL* e *SSL/TLS* referem-se ao protocolo TLS e aos certificados TLS. Assim, o OpenSSL deveria se chamar OpenTLS, pois já nasceu após a obsolescência do SSL.

1.2. Versões do PHP

A versão mais recente do PHP é a 8.3, com suporte ativo até 23/11/2025 e suporte de segurança até 23/11/2026. É vital que a versão de PHP em produção tenha pelo menos o suporte de segurança. Recomenda-se, entretanto, a atualização para a versão com suporte ativo para evitar possíveis explorações de vulnerabilidades pela demora na migração de versões.

Todas as versões com suporte de segurança válido até novembro de 2024 suportam OpenSSL 3.0. Na Tabela 3 mostramos qual a versão mínima de OpenSSL requerida por cada versão de PHP.

Tabela 3: Versões do PHP e versões do TLS

PHP	Suporte Ativo	Suporte de Segurança	OpenSSL mínimo requerido
8.1	25/11/2023	25/11/2024	>= 1.0.2
8.2	08/12/2024	05/12/2025	>= 1.0.2
8.3	23/11/2025	23/11/2026	>= 1.0.2

Para saber se o PHP foi instalado com a extensão openssl, basta executar o seguinte comando:

```
php -m
```

Para ter uma informação completa sobre a versão de OpenSSL suportada por uma instalação de PHP e a configuração da extensão, basta executar o seguinte comando:

```
php -i | grep -i openssl
```

A primeira linha de resultado já informará a versão de OpenSSL.

Nas Tabelas 4, 5, e 6, podemos ver, respectivamente, *features*, obsolescências e incompatibilidades relacionadas a TLS nas versões de PHP com suporte de segurança ativo.

Tabela 4: Versões do PHP e features de TLS

PHP	**Obsolescência**
8.1	Esta versão introduziu as opções **CURLOPT_PROXY_SSLCERT_BLOB** e **CURLOPT_SSLCERT_BLOB** para a extensão curl.
8.2	Esta versão adicionou suporte AEAD[i] para o algoritmo chacha20-poly1305[ii].
8.3	Nenhuma

Tabela 5: Versões do PHP e obsolescências de TLS

PHP	**Obsolescência**
8.1	A opção *ssl_method* de *SoapClient::__construct()* foi descontinuada em favor das opções de contexto de fluxo SSL.
8.2	Nenhuma
8.3	Nenhuma

Tabela 6: Versões do PHP e incompatibilidades de TLS

PHP	**Incompatibilidade**
8.1	As chaves privadas EC passaram a ser exportadas no formato PKCS#8 em vez do formato tradicional, assim como todas as outras chaves. *openssl_pkcs7_encrypt()* e *openssl_cms_encrypt()* passaram a ser padronizados usando AES-128-CBC[iii] em vez de RC2-40[iv]. A cifra RC2-40 é considerada insegura e não habilitada por padrão pelo OpenSSL 3.
8.2	Nenhuma
8.3	Nenhuma

Não há registro de vulnerabilidades do PHP relacionadas a TLS. Com relação a vulnerabilidades relacionadas a SSL (termo praticamente sinônimo), há vários registros sobre aplicações implementadas em PHP, o que é falha da aplicação e não da linguagem.

Especificamente sobre a linguagem, há uma vulnerabilidade de versões inferiores a 5.6 e duas de versões inferiores a 5.5[2][3]. Essas versões de PHP estão obsoletas e sem suporte.

2 https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-4721
3 https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-4248

2.Análise

Conforme relatamos no capítulo anterior, todas as versões de PHP com suporte de segurança válido até novembro de 2024 suportam OpenSSL 3.0, que implementa TLS 1.3. A recomendação é que todas as aplicações PHP em produção no Serpro utilizem preferencialmente a versão com suporte ativo. No mínimo, deve ser usada uma versão com suporte de segurança. O uso de versões de PHP sem suporte de segurança é um risco para a empresa e para seus clientes.
Considerando que o suporte de segurança da versão 8.1 de PHP termina em novembro de 2024, devem ser tomadas providências para migrar as aplicações pelo menos para a versão 8.2 até o final de 2024.

A existência de versões de PHP anteriores à 8.1 é uma situação temerosa, com gravidade inversamente proporcional ao número da versão. Se existem aplicações em produção nessa situação, elas devem ser migradas com máxima urgência. Caso não seja possível uma migração direta para a versão 8.2 por causa do acúmulo de incompatibilidades, a migração pode ser planejada para ser feita mais de uma etapa (por exemplo, migrando para a versão 7 e depois para a versão 8).

3.Conclusão

Para um uso funcional e seguro de TLS com PHP, é recomendável o uso mínimo da versão 8.2, considerando que o suporte de segurança da 8.1 será encerrado em menos de um ano.
A recomendação para versões de PHP inferiores a 8.1 é apenas uma: migração imediata para a versão ativa mais recente possível.

Este documento foi finalizado em 29 de fevereiro de 2024.

i AEAD é o acrônimo de *Authenticated encryption with associated data*, uma variante do esquema de criptografia **Authenticated Encryption** (AE). Esse esquema é especificado pela RFC 5116.

ii Os algoritmos ChaCha20 e Poly1305 são definidos na RFC 7539.

iii AES é o acrônimo de *Advanced Encryption Standard* e CBC é o acrônimo de *Cipher Block Chaining*. O uso do algoritmo AES-CBS é tratado na RFC 3602.

iv RC2-CBC é um mecanismo para criptografia e descriptografia de partes únicas e múltiplas. O mecanismo usado para gerar uma chave secreta RC2 de 40 bits é o SHA-1-PBE.