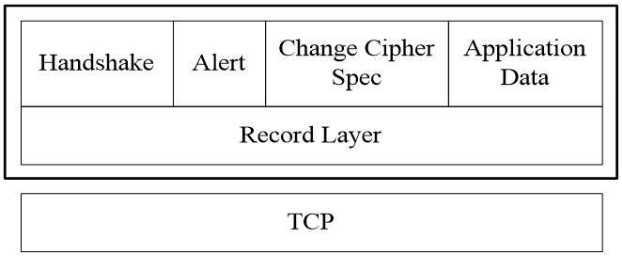
传输层安全协议

TLS（Transport Layer Security），前身是Netscape于1994年为因特网的安全传输所开发的SSL（Security Socket Layer）。

TLS分为两层[1]：

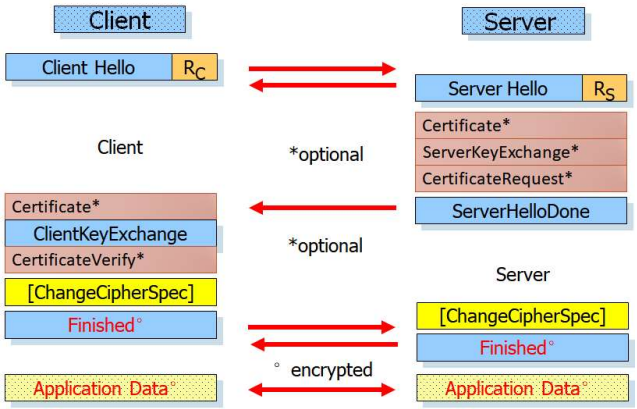
- 封装上层协议数据的Record Layer，保证数据的保密性与完整性
- 位于Record Layer之上的协议
  - 做身份认证及密钥交换的Handshaking协议：Handshake、Alert、Change Cipher Spec
  - 一系列应用层协议，比如HTTP、SMTP



【图：层次关系】

TLS的握手过程无非：

- cipher suit的参数协商。使用时间戳与随机数防止重放攻击
- 进行身份认证。一种方式是使用公钥证书体系
- 进行密钥协商。一种是采用DH算法进行密钥交换，通过随机产生的pre-master key获取master key，再获取session key
- 协商完毕，开始传输数据



【图：协议的整体步骤】

Note：当然也可以重用之前协商出的Session。

只关心TLS中涉及密码学的部分，主要有如下几类[3]：

- **密钥交换**

- **身份认证**
- **数据加密**
- **消息认证**

Note：数据加密与消息认证可合并成***认证加密***（Authenticated Encryption，AE），或者称为***用于关联数据的认证加密***（Authenticated Encryption with Associated Data，AEAD，AE的变种），即先加密再认证（Encrypt-then-MAC，EtM）。

将这些部分进行组合，协商出一种cipher suit，`openssl ciphers -v`输出的一部分cipher suit如下：

```
$ openssl ciphers -v
ECDHE-ECDSA-AES256-GCM-SHA384 TLSv1.2 Kx=ECDH     Au=ECDSA Enc=AESGCM(256) Mac=AEAD
DHE-RSA-AES128-SHA256   TLSv1.2 Kx=DH       Au=RSA  Enc=AES(128)  Mac=SHA256
RSA-PSK-AES256-CBC-SHA384 TLSv1 Kx=RSAPSK   Au=RSA  Enc=AES(256)  Mac=SHA384
DHE-PSK-AES256-CBC-SHA384 TLSv1 Kx=DHEPSK   Au=PSK  Enc=AES(256)  Mac=SHA384
ECDHE-PSK-AES128-CBC-SHA TLSv1 Kx=ECDHEPSK Au=PSK  Enc=AES(128)  Mac=SHA1
PSK-AES128-CBC-SHA      SSLv3 Kx=PSK      Au=PSK  Enc=AES(128)  Mac=SHA1
```

Note:

- Kx代表密钥交换，Au代表身份认证，Enc代表加密，Mac代表消息认证

TLS通过迪菲-赫尔曼密钥交换（Diffie–Hellman key exchange，DH）提供前向安全性（Forward Secrecy，FS，or Perfect Forward Secrecy，PFS）。它可以通过不安全的公共信道交换一个信息，从而创建一个可以用于在公共信道上安全通信的密钥。基于椭圆曲线密码学（ECC）的DH称为ECDH。

> Note：
>
> 1. 前向安全性：长期使用的主密钥泄露不会导致过去的会话密钥泄露。
> 2. PSK：Pre Share Key，预共享密钥。
> 3. 密钥分发有三种方式：
>   - 对称密钥分发：通过预共享的Master Key加密Session Key，分发Session Key
>   - 非对称密钥分发
>   - DHE
>   - 公钥加密，私钥解密
> 4. DH/ECDH vs DHE/ECDHE[7]：E代表Ephemeral（短暂的）。
>   对于DH，在配置Web Server时，除了公钥证书之外，还需要*dhparam.pem*——这是一个静态的DH公钥，在协商密钥的时候使用，此时只有客户端随机选择一个数。

> ECDH并不使用*dhparam.pem*，但在服务的也无需产生随机数，这个数字来源于ECC证书。[5]
> 而DHE/ECDHE，双方都会在协商时随机选择一个数。
> 有地方[5]称，ECDH这种不具备前向安全性。

DHE易受到中间人攻击，因此搭配公钥基础设施（Public Key Infrastructure，PKI）使用，通过使用公钥证书（一种格式是X.509）进行身份认证。X.509当前存在RSA与ECC两种证书格式，一种基于合数的素因子分解难题，另一种是椭圆曲线上离散对数难题。

> Note：
> 消息认证通过消息认证码（Message Authentication Code，MAC）的方式实现，但它不能处理通信双方自身发生的攻击，因此一种更好的方案是使用数字签名。
> DSA（Digital Signature Algorithm）是一种数字签名算法。

第三部分保证了数据的保密性，所需要的密钥来源于之前的密钥交换。AES（Advanced Encryption Standard）是一种对称加密标准。区别于流密码，CBC、GCM这种是分组密码的工作模式名称。其中GCM在工作的过程中对每一块数据先加密再认证。
第四部分保证了数据的完整性。消息认证的的一种实现基于Hash函数，称之为HMAC，另一种基于分组密码。

除此外，TLS还涉及伪随机数产生以及密钥扩展的问题，这些通过使用**伪随机函数**（Pseudo Random Function，PRF）来实现。它可以是基于特别设计的算法，也可以使用Hash函数和MAC产生伪随机数。

> Note：
> 密钥扩展：TLS握手生成的master secret只有48字节：两组encryption key、MAC key、IV加起来一般会超过48字节，所以需要一个PRF，把48字节拓展到需要的长度。

[1]: https://blog.helong.info/blog/2015/09/07/tls-protocol-analysis-and-crypto-protocol-design/
[2]: https://wiki.mozilla.org/Security/Server_Side_TLS
[3]: https://en.wikipedia.org/wiki/Cipher_suite
[4]: https://www.smwenku.com/a/5bd8a2f02b71774cdc227ec9
[5]: https://codertw.com/%E7%A8%8B%E5%BC%8F%E8%AA%9E%E8%A8%80/496424/
[6]: https://en.wikipedia.org/wiki/Forward_secrecy
[7]: https://crypto.stackexchange.com/questions/39985/whats-the-difference-between-dh-and-dhe
[8]: https://github.com/fi3ework/blog/issues/17
[9]: https://www.iana.org/assignments/tls-parameters/tls-parameters.xhtml

---

by river[river@vvl.me]

2019.0227: initialization