TLS

(Transport Layer Security) TLS

- ✓ بر اساس تجربیات به دست آمده از SSL و PCT بنیان گذاشته شده است.
- ✓ بدنبال ایجاد یک نسخه استاندارد اینترنتی از SSL می باشد.
- ✓ این پروتکل در RFC2264 توسط IETF استاندارد شده است.
- ✓ بسیار شبیه پروتکل SSL نسخه ۳ است.
- ✓ پروتکل TLS از دو لایه تشکیل شده است:
  - TLS Record Protocol
  - TLS Handshake Protocol

(Transport Layer Security) TLS

پروتکل TLSدر حقیقت نسخه ای از SSL است که توسط IETF استاندارد و در سند RFC 2246 در سال ۱۹۹۹ معرفی شد.

تفاوت های TLS با SSL عبارتند از:

۱- در TLS بر خلاف SSL هرمشتری از سرویس دهنده مطالبه گواهی دیجیتالی کند ولی طرف مقابل آن را ارائه ننماید، نشست قابل ارائه نیست و فوراً قطع خواهد شد.

۲- روش تولید رشته های تصادفی (Nonce) تغییر کرده است.

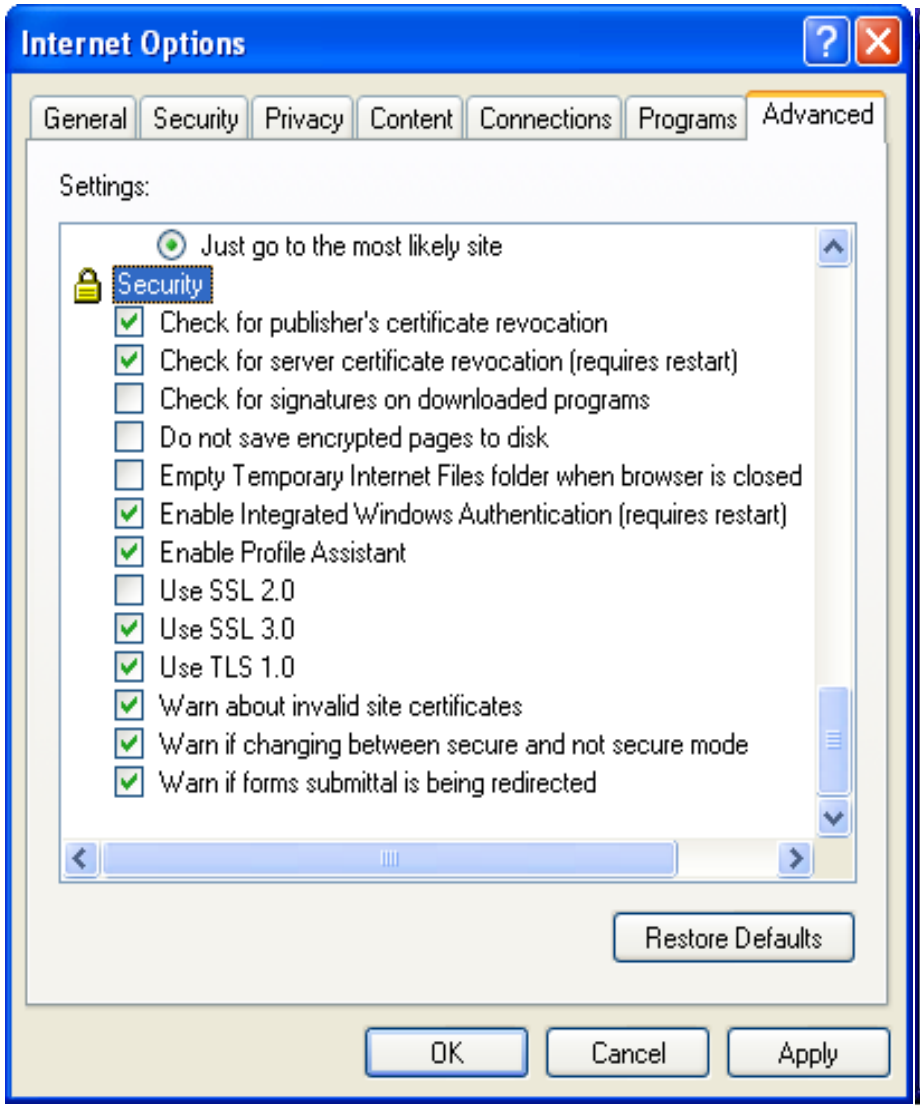
۳- روش تولید کد احراز هویت و اصالت هر پیام (MAC) تغییر کرده است.

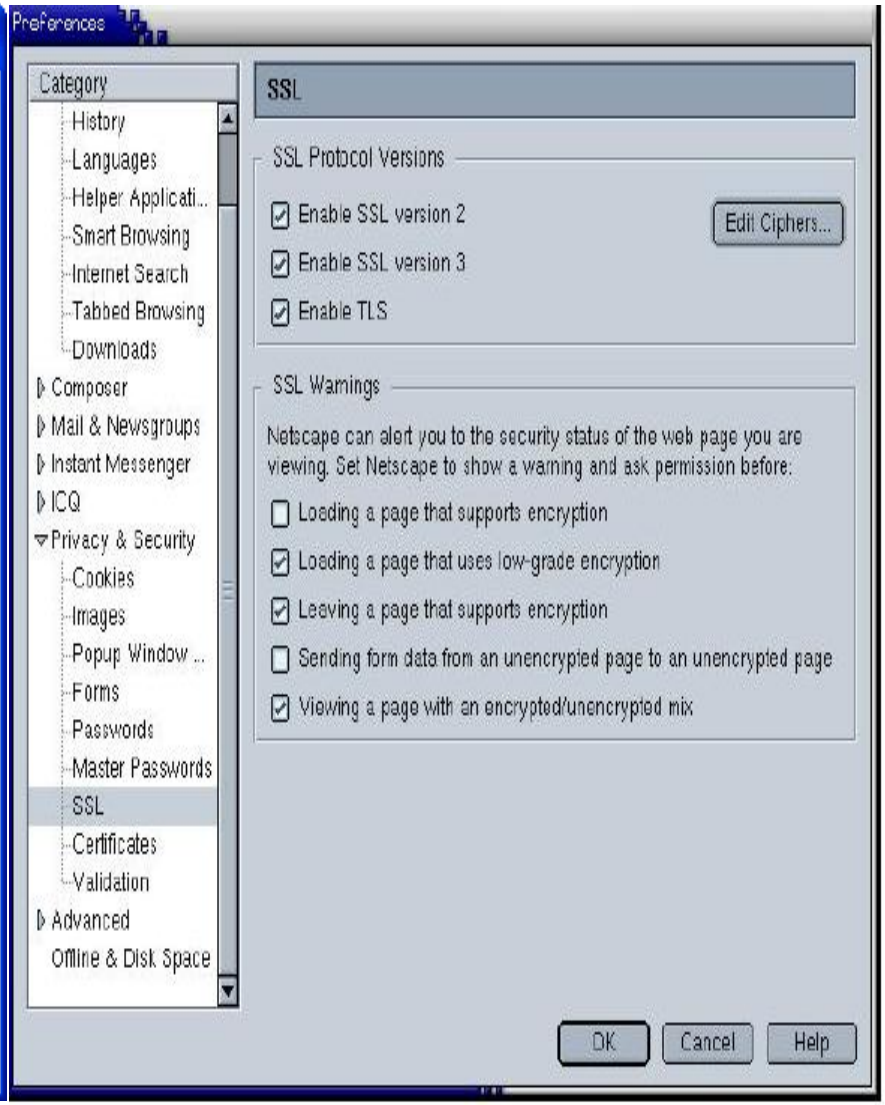
٤- روش های رمزنگاری متقارن جدیدی به آن اضافه شده و در عوض روش Fortezza که برای تعامل با کارت های هوشمند به کار می آم، حذف شده است.

TLS

- مشابه SSL با چند تفاوت اندک با آن، که خود موجب ناسازگاری آن با SSL است
- اگر مشتری از سرور گواهینامه او را مطالبه کند، و سرور نتواند گواهینامه ای را ارائه نماید تشکیل نشست لغو می شود
- روش تولید عدد تصادفی (Nonce) تغییر کرده است
- روش تولید کد احراز هویت (MAC) تغییر یافته است
- روش های رمزنگاری جدیدی اضافه شده است

SSL/TLS Configuration Options
Internet Explorer 6.0 & Netscape





الگوریتم HMAC

$$\text{HMAC} = H[(K \oplus \text{opad}) \| H[(K \oplus \text{ipad}) \| M]]$$

where

ipad = 00110110(0x36) repeated 64 times (512 bits)
opad = 01011100(0x5c) repeated 64 times (512 bits)
H = one-way hash function for TLS (either MD5 or SHA-1)
M = message input to HMAC
K = padded secret key equal to the block length of the hash code (512 bits for MD5 and SHA-1)

Pseudo Random Function

تولید اعداد تصادفی :

از یک تابع شبه تصادفی(PRF) استفاده می کند که از دو الگوریتم Hash تشکیل شده است :

```
P_hash(secret, seed) = HMAC_hash (secret, A(1)||seed) ||
                       HMAC_hash (secret, A(2)||seed) ||
                       HMAC_hash (secret, A(3)||seed) ||...
```

where A() is defined as:

A(0) = seed

A(i) = HMAC_hash(secret, A(i-1))

Pseudo Random Function

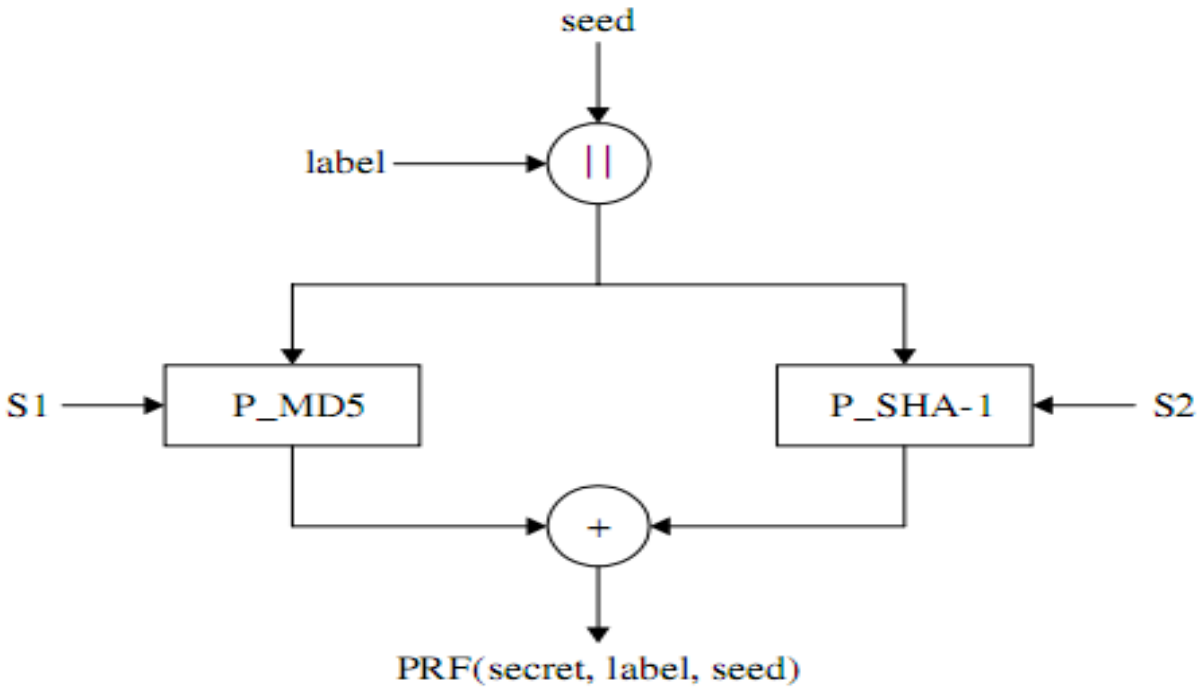
- ✓ برای ایجاد PRF نتیجه ی حاصل تابع گسترش به دو نیمه تقسیم می شود :
  - یک نیمه برای تولید داده با P_MD5
  - یک نیمه برای تولید داده با P_SHA-1
  - سپس دو نیمه برای تولید خروجی XOR می شوند.

(مقصود از P_MD5 یا P_SHA-1 این است که در P_Hash که در اول توضیح داده شد هر جا در HMAC نیاز به استفاده از الگوریتم HASH داشتیم از یکی از روش های MD5 یا SHA-1 استفاده می شود.)

$$PRF(secret, label, seed) = P\_MD5(S1, label \| seed) \oplus P\_SHA-1(S2, label \| seed)$$

Label : یک رشته ASCII معمولی است.

Pseudo-Random Function(PRF)…



S1: First half of the secret
S2: Second half of the secret

P_MD5: Data expansion function to expand a secret S1 and (seed| | secret) using MD5

P_SHA-1: Data expansion function to expand a secret S2 and (seed| | secret) using SHA-1

A pseudo-random function (PRF) generation scheme.

محاسبه master secret

- Pre-master-secret مشابه SSL محاسبه می شود اما محاسبه ی master-secret تفاوت دارد :

LABEL

```
master_secret = PRF(premaster_secret, ‘‘master secret’’,
                    ClientHello.random||ServerHello.random)
```

- و محاسبه key block ها :

```
key_block = PRF(master_secret, ‘‘key expansion’’,
            SecurityParameters.server_random||
                SecurityParameters.client_random)
```

TLS Handshake protocol

- ✓ Ciphers Supported
  - Block

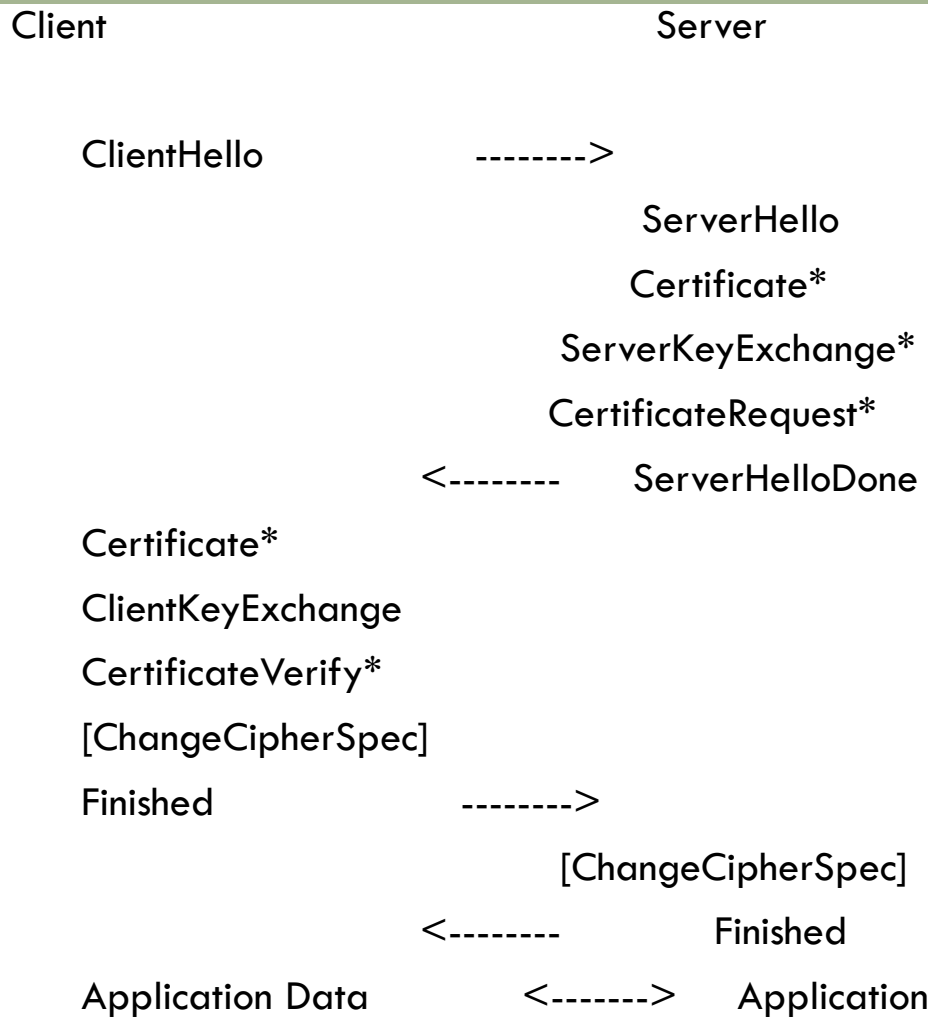
    DES, RC2, RSA, IDEA
  - Stream

    RC4
  - MAC

    SHA-1, MD5
  - Digital Signature

    DSS, RSA

TLS Handshake protocol



* مراحل اضافی نسبت به SSL Handshake

TLS

- مشابه SSL با چند تفاوت اندک با آن، که خود موجب ناسازگاری آن با SSL است
- اگر مشتری از سرور گواهینامه او را مطالبه کند، و سرور نتواند گواهینامه ای را ارائه نماید تشکیل نشست لغو می شود
- روش تولید عدد تصادفی (Nonce) تغییر کرده است
- روش تولید کد احراز هویت (MAC) تغییر یافته است
- روش های رمزنگاری جدیدی اضافه شده است

SSL/TLS Configuration Options
Internet Explorer 6.0 & Netscape