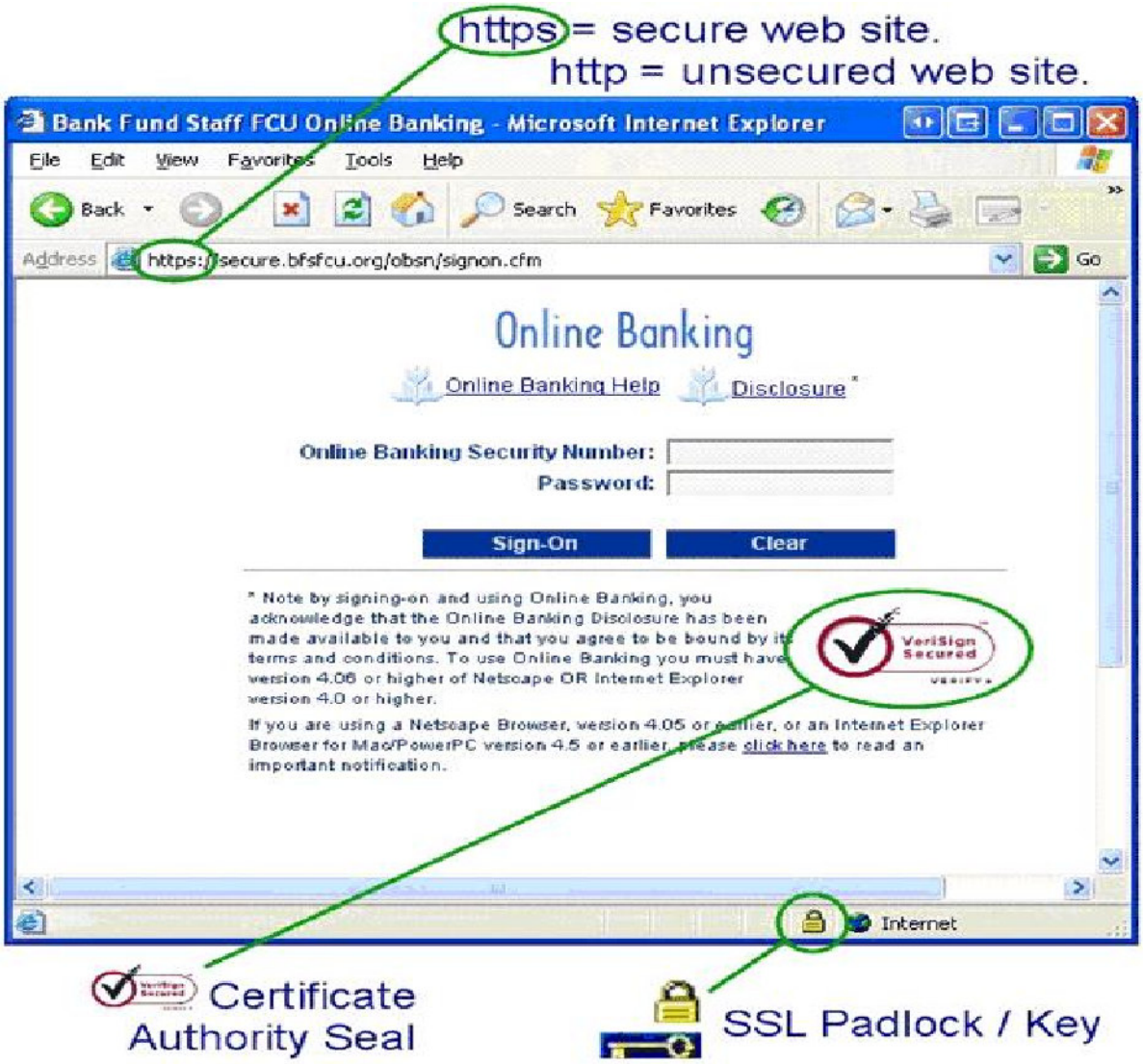
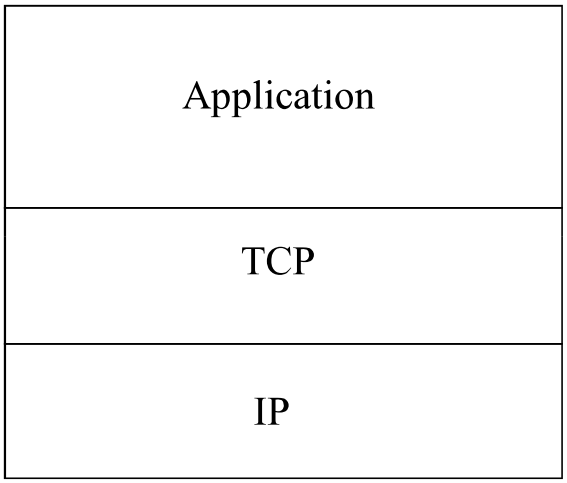
Secure Sockets Layer (SSL)

مقدمه

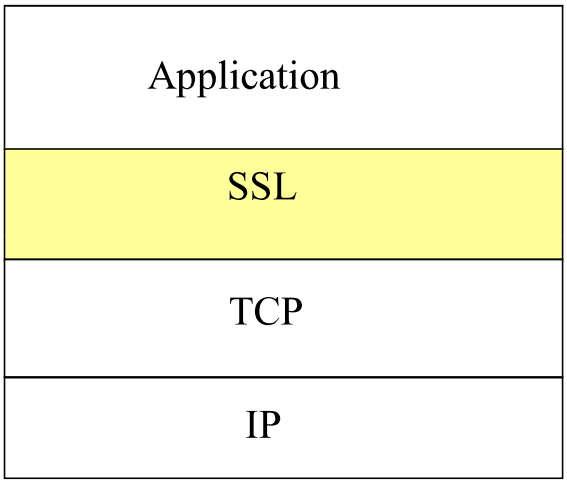
- طراحی شده توسط شرکت Netscape در سال ۱۹۹۳
- توسط اکثر مرورگرها پشتیبانی می شود.
- نمونه استاندارد شده: TLS (Transport Layer Security)
  - RFC 2246
- هدف
  - محرمانگی (Confidentiality)
  - جامعیت (Integrity)
  - تأیید هویت (Authentication)


https = secure web site.
http = unsecured web site.
Bank Fund Staff FCU Online Banking - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address https://secure.bfsfcu.org/obsn/signon.cfm
Go
Online Banking
Online Banking Help
Disclosure *
Online Banking Security Number:
Password:
Sign-On
Clear
* Note by signing-on and using Online Banking, you acknowledge that the Online Banking Disclosure has been made available to you and that you agree to be bound by its terms and conditions. To use Online Banking you must have version 4.06 or higher of Netscape OR Internet Explorer version 4.0 or higher.
If you are using a Netscape Browser, version 4.05 or earlier, or an Internet Explorer Browser for Mac/PowerPC version 4.5 or earlier, please click here to read an important notification.
VeriSign Secured
Internet
Certificate Authority Seal
SSL Padlock / Key

SSL و TCP/IP



Normal Application



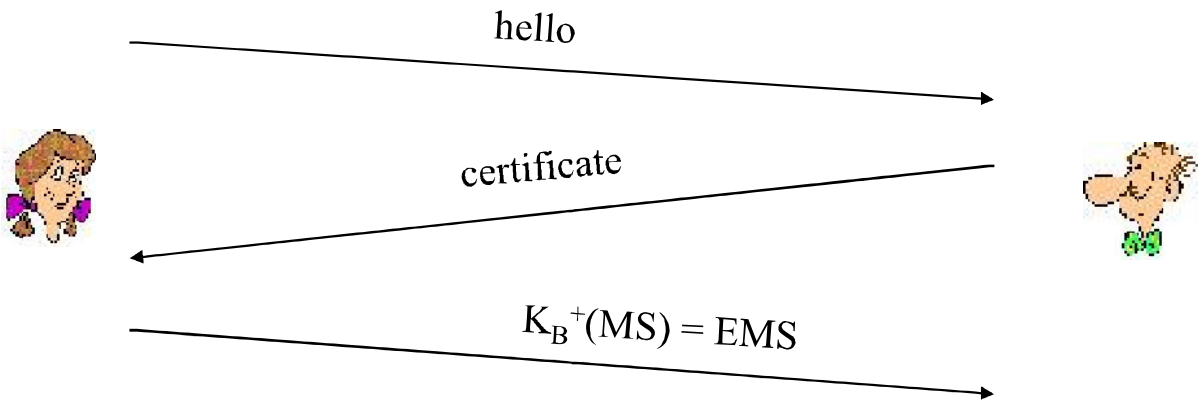
Application
with SSL

مراحل برقراری ارتباط امن

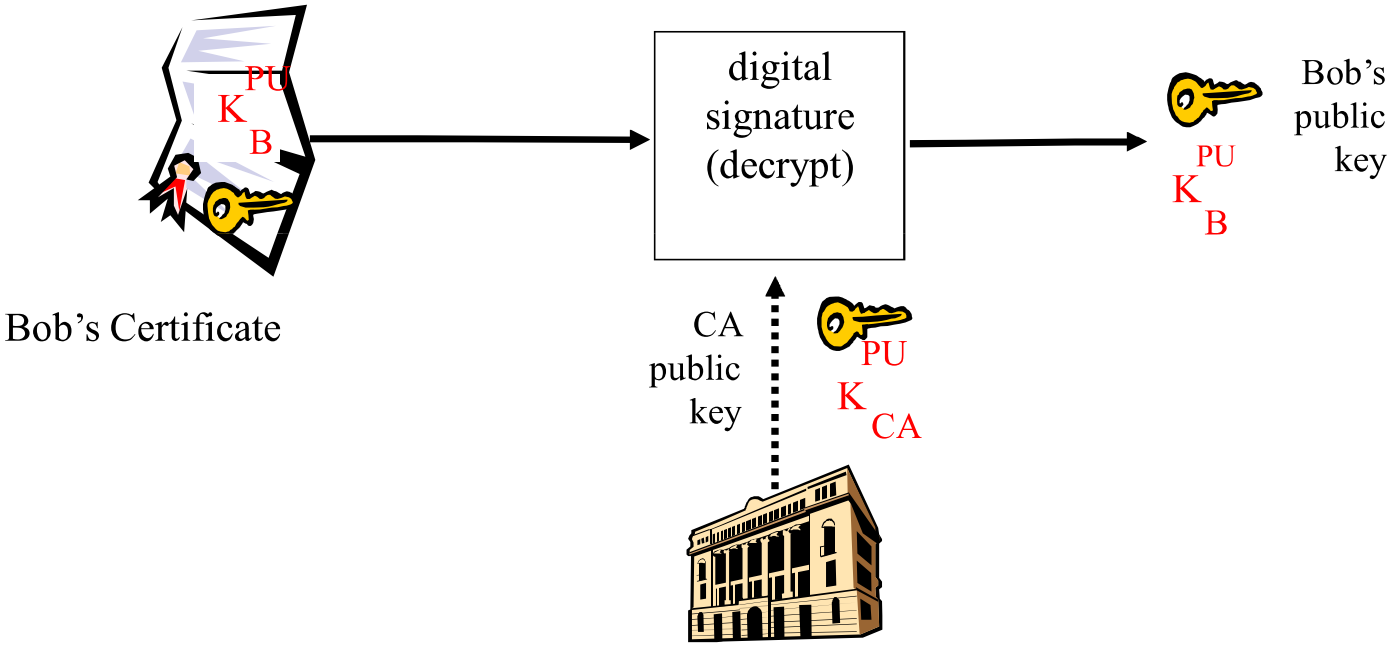
- **Handshake**: تأیید هویت طرفین ارتباط با استفاده از گواهی دیجیتال و تبادل کلید مشترک محرمانه
- **تولید کلید:** استفاده از کلید مشترک محرمانه برای تولید کلیدهای مورد نیاز
- **تبادل داده:** شکستن داده به تعدادی رکورد و ارسال رکوردها
- **بستن ارتباط:** تبادل پیام هایی برای بستن ارتباط به صورت امن

Handshake

- MS = Master Secret
- EMS = Encrypted Master Secret



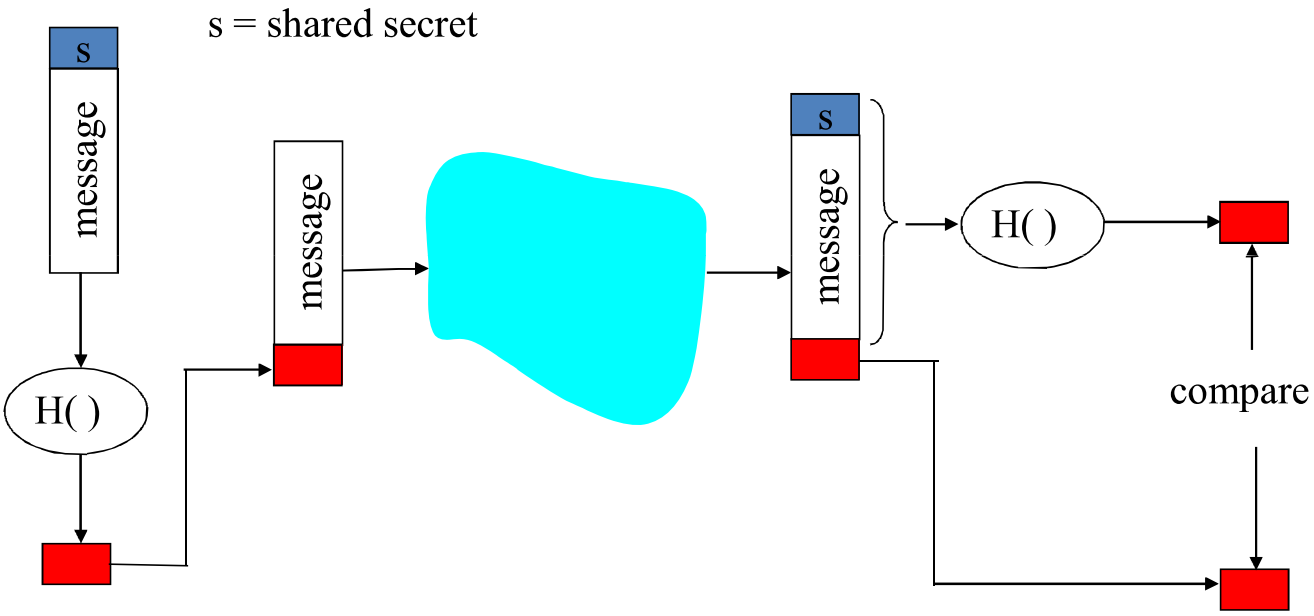
بدست آوردن کلید عمومی



تولید کلیدهای مورد نیاز

- استفاده از کلید یکسان برای عملیات مختلف موجب کاهش امنیت می شود.
- استفاده از کلیدهای مختلف برای رمزنگاری و MAC
  - K_c: کلید رمزنگاری برای ارسال داده از کلاینت بر سرور
  - M_c: کلید MAC برای ارسال داده از کلاینت بر سرور
  - K_s: کلید رمزنگاری برای ارسال داده از سرور به کلاینت
  - M_s: کلید MAC برای ارسال داده از سرور به کلاینت
- تولید کلیدهای فوق با کلید MS

MAC



رکورد داده

- داده به تعدادی رکورد شکسته می شود.
- هر رکورد MAC دارد.
- طول رکوردها می تواند متفاوت باشد.

شمارنده توالی: Sequence Number

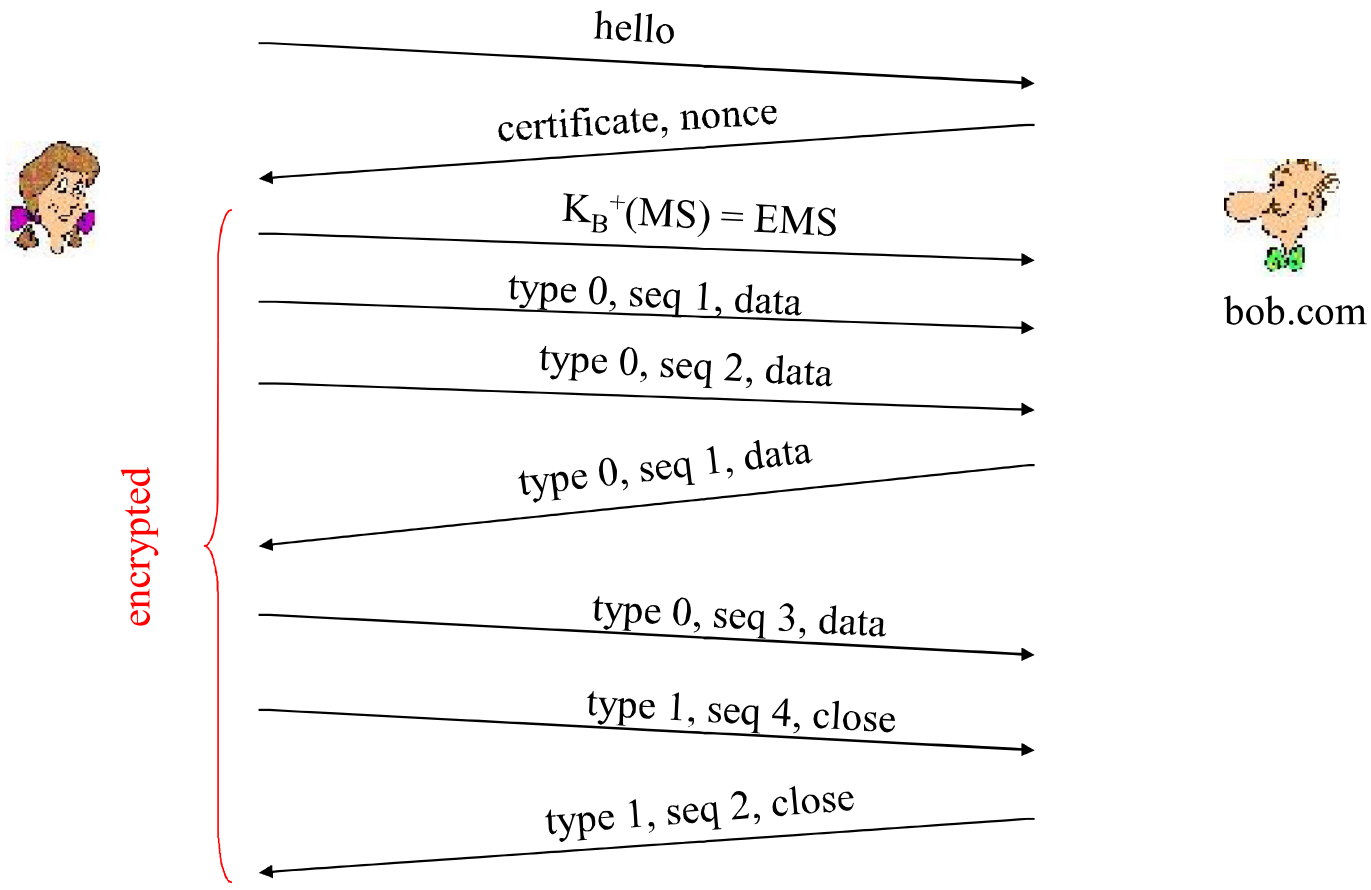
- مهاجم می تواند رکوردها را بدست آورد و دوباره ارسال کند یا ترتیب ارسال رکوردها را تغییر دهد.
- برای جلوگیری از این حمله، شمارنده توالی هر رکورد را در MAC قرار می دهیم.

  $MAC = MAC(M_x, seq\_num \,||\, data)$ –
- هیچ فیلدی برای شمارنده توالی در رکورد در نظر گرفته نشده و طرفین ارتباط شمارنده توالی را نگهداری می کنند.
- شمارنده توالی از کلاینت به سرور و بالعکس مستقل هستند.

اطلاعات کنترلی

- مهاجم می تواند بسته پایان ارتباط را جعل کند و ارتباط بین طرفین را ببند.
- برای جلوگیری از این حمله، برای رکوردها نوع مشخص کنیم.
  - نوع 0 برای رکورد داده
  - نوع 1 برای رکورد بستن ارتباط
- $MAC = MAC(M_x, seq\_num \| type \| data)$

خلاصه


hello
certificate, nonce
$K_B^+(MS) = EMS$
type 0, seq 1, data
type 0, seq 2, data
type 0, seq 1, data
type 0, seq 3, data
type 1, seq 4, close
type 1, seq 2, close
encrypted
bob.com

الگوریتم های رمزنگاری مورد استفاده SSL

- رمزنگاری متقارن
  - DES: Block Cipher
  - 3DES: Block Cipher
  - RC2: Block Cipher
  - RC4: Stream Cipher
- رمزنگاری کلید عمومی
  - RSA

SSL Cipher Suite

- Cipher Suite
  - الگوریتم رمزنگاری کلید عمومی
  - الگوریتم رمزنگاری کلید متقارن
  - الگوریتم MAC
- کلاینت و سرور روی Cipher Suite توافق می کنند.
- کلاینت Cipher Suiteهایی که پشتیبانی می کند را به پیشنهاد می کند و سرور یکی از آنها را انتخاب می کند.

SSL Handshake

1. کلاینت لیست الگوریتم هایی که پیشتیانی می کند را همراه با nonce به سرور ارسال می کند.
2. سرور الگوریتم انتخابی را همراه گواهی و nonce خود به کلاینت می فرستد.
3. کلاینت صحت گواهی را تأیید کرده و سپس کلید عمومی سرور را استخراج می کند و کلید MS تولید شده را با کلید عمومی سرور رمز می کند و به سرور ارسال می کند.
4. کلاینت و سرور با استفاده از کلید MS و nonceها، کلیدهای رمزنگاری و MAC را تولید می کنند.
5. کلاینت MAC تمام پیام های Handshake را به سرور می فرستد.
6. سرور MAC تمام پیام های Handshake را به کلاینت می فرستد.

مراحل ۵ و ۶؟

- کلاینت معمولاً لیستی از الگوریتم ها را ارائه می دهد که بعضی قوی ترند و بعضی ها ضعیف تر.
- مهاجم می تواند با استفاده از حمله Man-In-The-Middle الگوریتم های قوی تر را از لیست حذف کند.
- مراحل ۵ و ۶ از tampering جلوگیری می کنند.

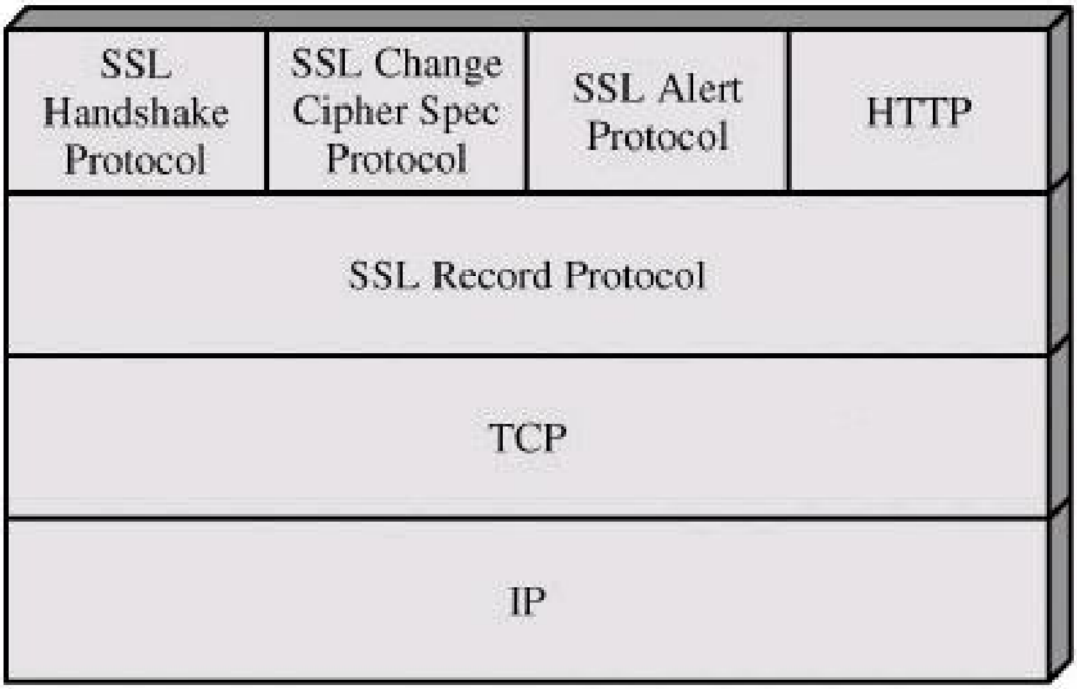
nonce تصادفی؟

- مهاجم می تواند تمام رکوردهای رد و بدل شده بین کلاینت و سرور را بگیرد.
- سپس با سرور ارتباط برقرار کرده و تمام رکوردها را با توالی درست به سرور بفرستد.
  - مثال: اجرای دستور روی سرور
- برای جلوگیری از این حمله، در هر ارتباط nonceهای تصادفی تولید می شود تا کلیدهای رمزنگاری و MAC تولید شده متفاوت باشند.

انواع پیام های Handshake

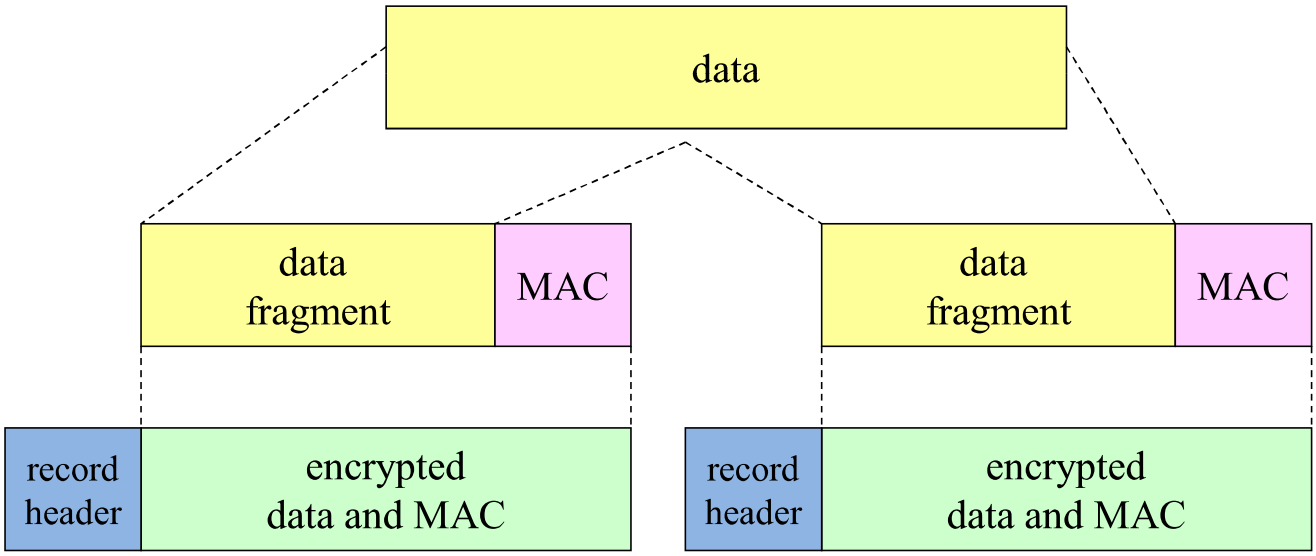
- هر پیام Handshake دارای یک فیلد ۱ بایتی نوع می باشد:
  - ClientHello
  - ServerHello
  - Certificate
  - ServerKeyExchange
  - CertificateRequest
  - ServerHelloDone
  - CertificateVerify
  - ClientKeyExchange
  - Finished

SSL Protocol Stack

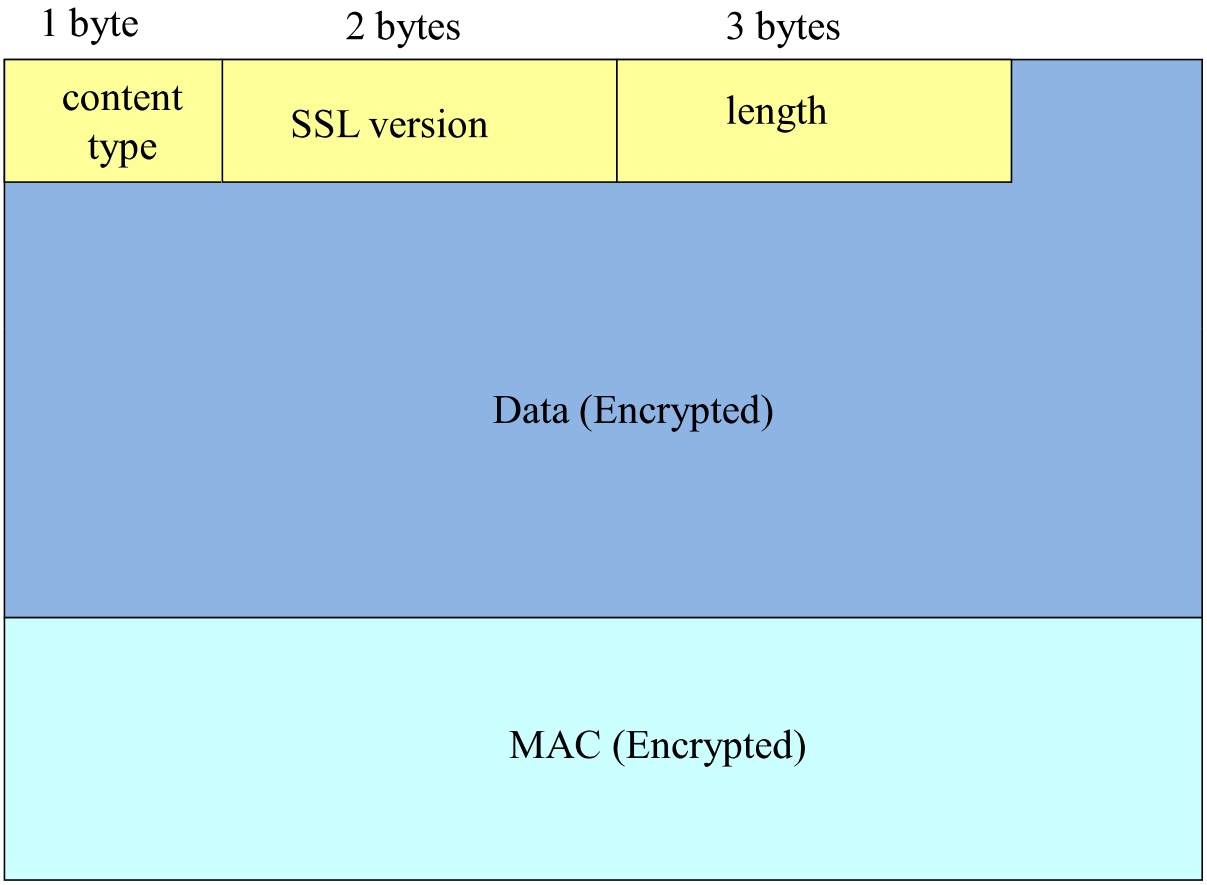


SSL Record Protocol

- Record Header: نوع محتوا، ورژن، طول
- هر Fragment حداکثر ۱۶ کیلو بایت است.

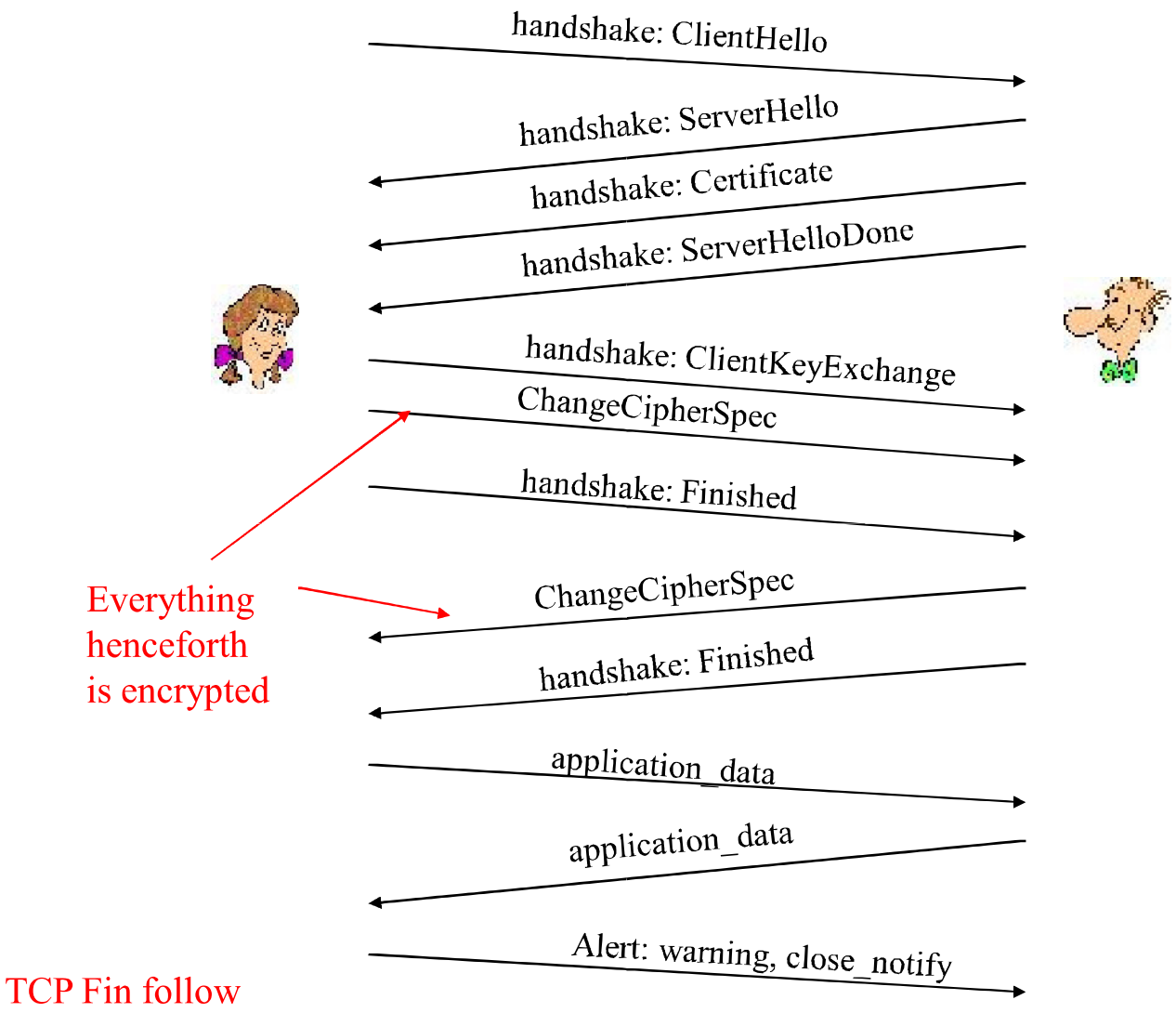


SSL Record Format

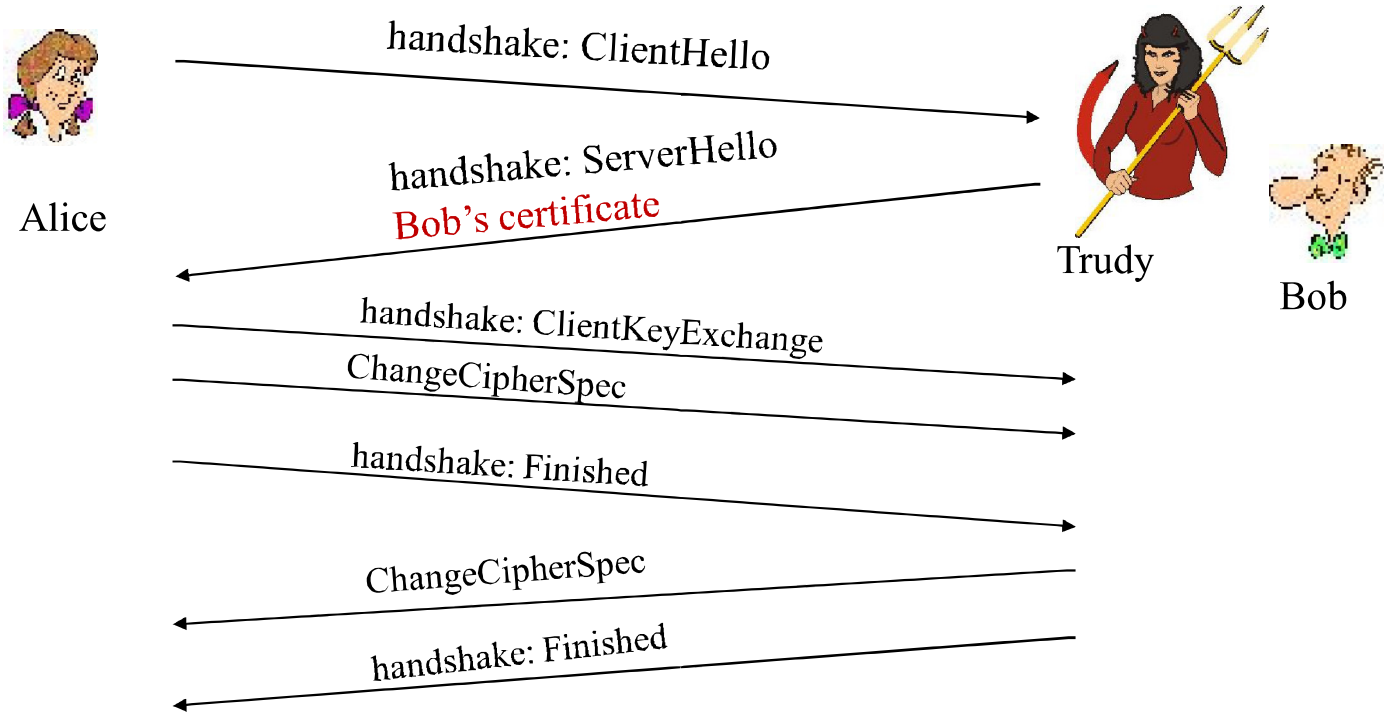


Content Type

- application_data (23)
- alert (21)
  - در صورت بروز خطا در طی Handshake
- handshake (22)
- change_cipher_spec (20)
  - درخواست برای تغییر الگوریتم های رمزنگاری و تأیید هویت


handshake: ClientHello
handshake: ServerHello
handshake: Certificate
handshake: ServerHelloDone
handshake: ClientKeyExchange
ChangeCipherSpec
handshake: Finished
Everything
henceforth
is encrypted
ChangeCipherSpec
handshake: Finished
application_data
application_data
Alert: warning, close_notify
TCP Fin follow

Man-In-The-Middle



تولید کلیدهای رمزنگاری و MAC

- با استفاده از nonce کلاینت، nonce سرور و کلید MS، یک عدد تصادفی تولید می کنیم.
- با استفاده از عدد تصادفی تولید شده و nonceها مقادیر زیر تولید می شوند:
  - کلیدهای MAC کلاینت و سرور
  - کلیدهای رمزنگاری کلاینت و سرور
  - Initialization Vector (IV) کلاینت و سرور

CBC (Cipher Block Chaining)

- نحوه رمز بلاک جاری به رمز بلاک قبلی وابسته است:
  - $C(i) = K_S(M(i) \oplus C(i-1))$
  - $M(i) = K_S(C(i)) \oplus C(i-1)$
- $C(0) = IV$
- تغییر IV برای هر پیام
  - پیام های یکسان در یک ارتباط دارای رمزهای متفاوتی خواهند بود.

تأیید هویت کلاینت

- SSL همچنین می تواند هویت کلاینت را تأیید کند.
- سرور پیام CertificateRequest را به کلاینت می فرستد.