




9.1 Computer and Network Security

INFO1112

Bob Kummerfeld

For network connections we wish to provide:

- 
- **Confidentiality**
 - Protecting information from interception
 - Ensure information remains confidential if intercepted
 - **Integrity**
 - Detecting whether information has been tampered with
 - **Authenticity**
 - Ensuring we know who the sender is
 - Non-repudiation

Security when browsing the web

We need confidentiality when browsing the web, otherwise an attacker may be able to monitor our traffic and get access to usernames and passwords for services such as banking.

The HTTP web protocol has a secure version called HTTPS. In fact it is the same protocol but the data on the transport layer connection (TCP/IP) is encrypted.

Layering

Using the principle of layering we can provide security for existing protocols such as HTTP.



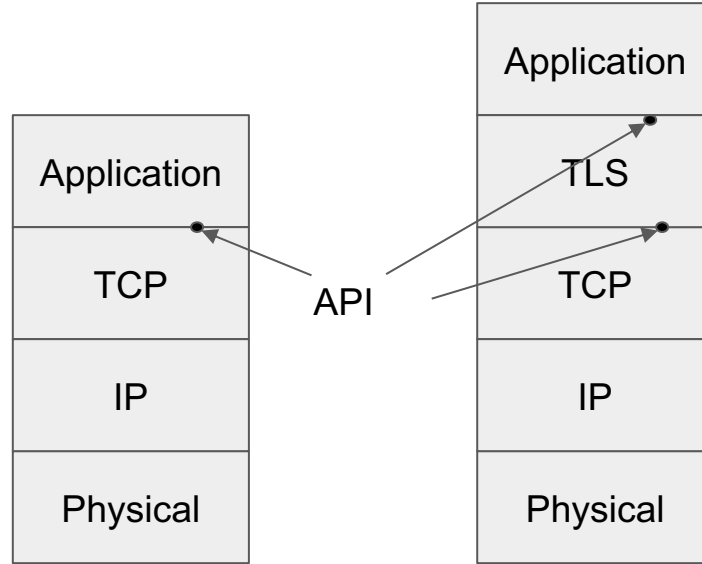
Encryption

In the encryption layer we scramble the data in such a way that it can't be understood by an attacker but the intended recipient can un-scramble it to get the original data.

We do the "scrambling" using *cryptology*.

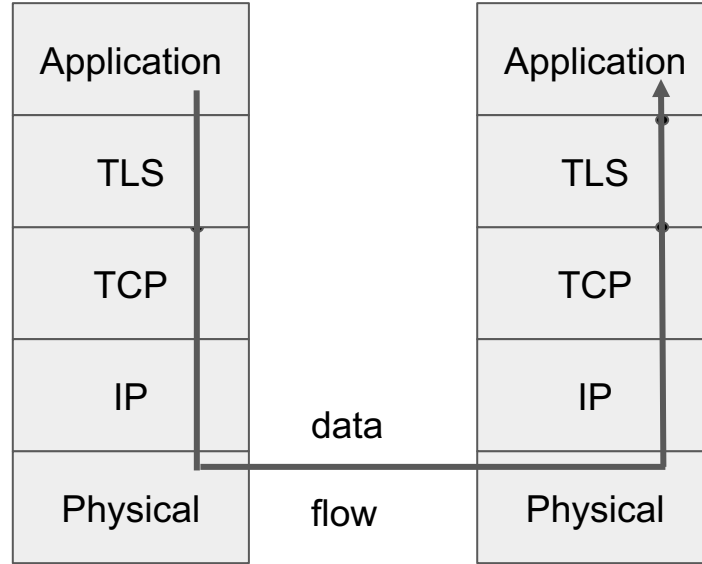
Layering





Application

Application using TLS



Cryptography

- Cryptography is used to guard against interception of communication
- A message m is to be sent from Alice to Bob
 - Carol may be able to observe the communication channel
- Alice converts m to an encrypted version using an encryption function $E(m)$; Alice sends $E(m)$ on the channel; Bob receives it and applies a decryption function D , to calculate $D(E(m))$; the result is the original message, m
- m is called the *plaintext*
- $E(m)$ is called the *ciphertext*
- The intention is that Carol may see $E(m)$, but can't find m

Alice

The quick brown fox jumped over the lazy dog

plaintext

$E(\text{plaintext})$

apply the function E to the plaintext message

header

7d38b5cd25a2baf85ad3bb5b9311383e67....

ciphertext

$D(\text{ciphertext})$

apply the function D to the ciphertext message

Bob

The quick brown fox jumped over the lazy dog

plaintext



Carol

Codes and Keys

- Usually encoding is done by an algorithm with a parameter k , called a key (or password, or pass-phrase)
 - It is difficult to keep algorithms secret, easier to keep keys secret.
 - The key k is a **shared secret**, known by Alice and Bob but not known by Carol
 - Can easily be changed
 - We have to assume that Carol knows the algorithms
- Trivial example algorithm: shift all letters to the left by n
 - The key is the amount of the shift, n
- Cipher text is $E(m, k)$
 - Decoder is $D(c, k)$ where c is ciphertext
- The main attack against this is try to find out the key!

Alice

The quick brown fox jumped over the lazy dog

plaintext

algorithms are public

keys are private

$E(\text{plaintext}, \text{key})$

apply the function E to the plaintext message

header

7d38b5cd25a2baf85ad3bb5b9311383e67....

ciphertext

$D(\text{ciphertext}, \text{key})$

apply the function D to the ciphertext message

Bob

The quick brown fox jumped over the lazy dog

plaintext



Carol

Using Encryption

To protect a web transaction, eg sending your banking ID and password, we can encrypt the data stream with a key that the user knows and the server knows.

The problem with this is that it is difficult to set up a secret key for a casual transaction that both the user and the server know.

We can't send the key over the connection without first encrypting it and therefore we need a secret key.....

