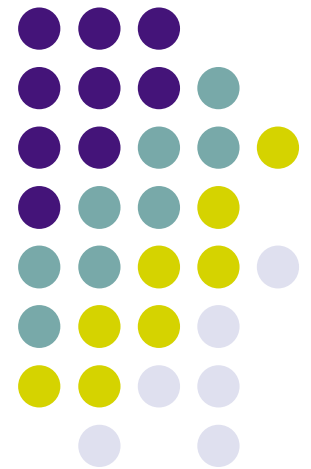
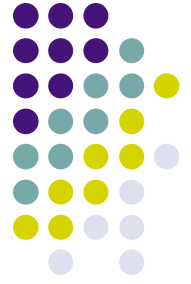


SSL/TLS

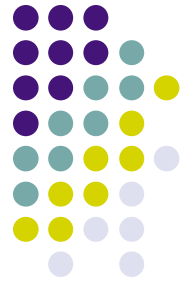
How to send your credit card number securely over the internet



The security provided by SSL



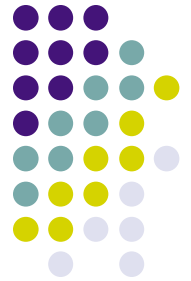
- SSL is implemented at level 4
 - The transport control layer
- In practice, SSL uses TCP sockets
 - The underlying TCP implementation handles robustness of communication, such as replay of lost packets, buffering packets to re-order them correctly, etc.
- SSL extends TCP interface for security



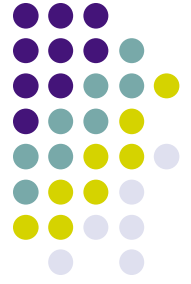
What does it entail?

- To use SSL, applications must change.
 - They have to use the SSL API (application programming interface) and use SSL calls instead of TCP calls. Applications' networking code must change
- SSL may be deployed without making changes to the underlying Operating System, because it does not alter the implementation of the TCP protocol.

The rogue packet problem (1)

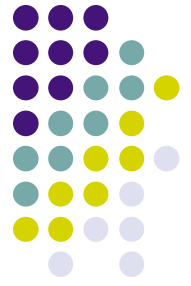


- TCP uses checksums to ensure correctness of data.
 - but this checksum prevents only against random errors.
- Suppose an attacker to SSL:
 - Forges the next TCP packet (in a TCP connection, packets are numbered).
 - Re-computes the TCP checksum
 - The TCP protocol accepts the corrupt packet, mark the packet number as delivered/received.



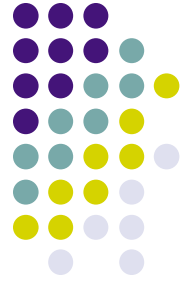
The rogue packet problem (2)

- TCP relays the corrupted packet to SSL
 - SSL checks its cryptographic checksum -- a message authentication code (MAC) -- and realizes that the packet has been forged
- TCP receives true packet from legitimate sender, sees that it has an already used number, and discards the packet as bad.
 - SSL cannot tell TCP to change its behavior, because it has not changed the TCP code. Only option for SSL is to hang up the connection.



SSL as software only

- Implementing SSL in hardware is unwieldy
 - It requires a TCP implementation to function
 - Therefore TCP has to be implemented in the same hardware
 - But TCP uses long buffers to ensure communication reliability. That means your hardware will require a lot of memory and be costly.
- If SSL worked at a lower level -- say IP (level 3), it could be coded in a net card.



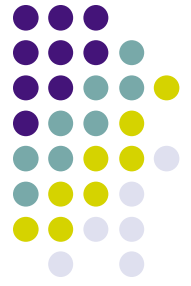
Advantages of SSL

- Allows for portable implementation, because it is an application-level process
 - Suitable for bundling with applications such as browsers, can be installed with user-privileges only, and minimum expertise in anything.
 - SSL can authenticate users (end-to-end authentication), not only machines or IP addresses (link-to-link authentication)

SSL/TLS: First Ingredients



- Ingredients:
 - SSL supports several “cipher suites”: algorithm sets for public key encryption, symmetric key encryption, and authentication (MACs). This flexibility was needed because of export restrictions. Client and Server must negotiate which algorithms are used in a session.
 - Client and server agree on a common secret, negotiated using public key cryptography, and incorporating challenges from both. From this common secret the symmetric keys are derived.



SSL/TLS: Ingredients (2)

- SSL uses symmetric keys asymmetrically:
 - After agreeing on common secret, client and server derive from it two IVs, *read* and *write* keys. The client read keys (server IV, server encryption key, server authentication key) equal the server write keys, and vice-versa. A total of six secrets are derived from the initial common secret.
 - The common secret is derived from cleartext randomizers from client and server, as well as a value (pre-master secret) chosen by client and transmitted under public key encryption to server.

SSL/TLS Basic Protocol

