Sicurezza delle reti[1]

Mattia Monga

Dip. di Informatica
Università degli Studi di Milano, Italia
mattia.monga@unimi.it

a.a. 2012/13

Lezione VI: TLS/SSL

TLS/SSL

Un'altra possibilità è introdurre misure di sicurezza sopra il livello di trasporto TCP.

- 1993–1995, Netscape rilascia un Secure Socket Layer SSL (2.0) pensato per proteggere la navigazione web.
- SSL 3.0, standardizzato da IETF come TLS Transport Layer Security

Obiettivi di sicurezza

- cifratura end-to-end
- protezione dell'integrità
- autenticazione del server (il client rimane anonimo)
- efficienza adeguata alle connessioni HTTP, brevi e stateless

sessione TLS

I nodi mantengono lo stato della sessione per gestire la cifratura del traffico.

- TLS handshake protocol
- TLS record layer
- una sessione può gestire piú connessioni per ridurre l'overhead

TLS handshake

1. C richiede la connessione, elencando quali cipher suite (CS) conosce
2. S sceglie CS compatibile e spedisce un digital certificate (DC) firmato da una CA
3. C controlla DC e manda criptata una chiave di sessione (K) random

Integrazione con le applicazioni

Tre strategie:

1. Creare un nuovo servizio (es. SSH2)
2. Aggiungere TLS ad un servizio noto (es. HTTPS)
3. Estendere un servizio noto affinché usi TLS (es. ESMTP)

Riassumendo

- TLS permette cifratura e autenticazione dei server (tramite CA) a livello di trasporto
- La gestione delle sessioni è progettata per essere efficiente in presenza di connessioni ripetute
- Molto diffuso perché facile da integrare nelle applicazioni

Livello di trasporto

IPsec e TLS possono essere piuttosto penalizzanti dal punto di vista delle prestazioni (Dal punto di vista delle performance del server, TLS può arrivare ad essere fino a 82 volte piú lento di una connessione TCP).
tcpcrypt è una proposta recente (2010) piú efficiente (3 volte piú lento di TCP)

Altri limiti di TLS

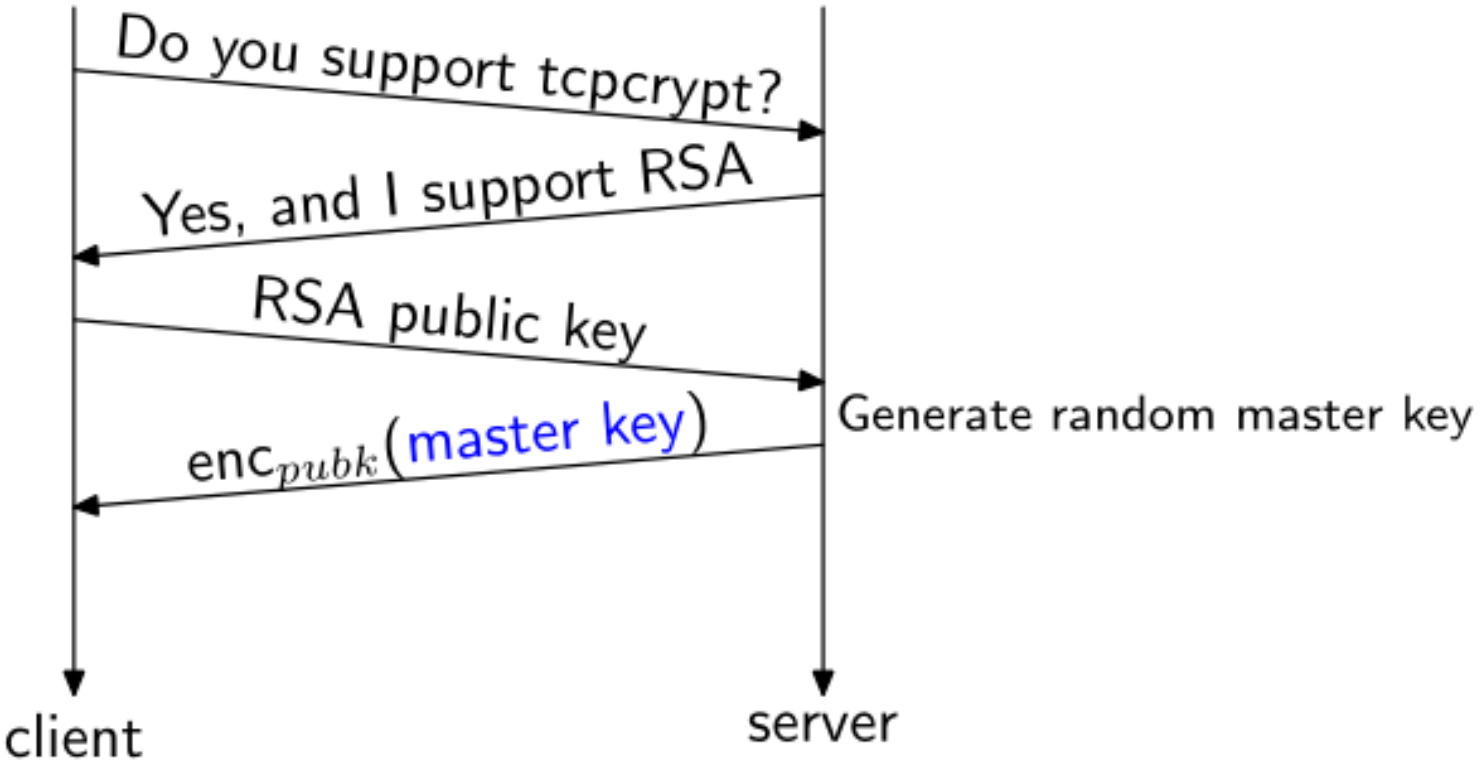
La cifratura dipende dall'autenticazione del server, a sua volta garantita dall'autorità certificatrice.

- Se l'autenticazione è falsa, la cifratura non è molto utile (ma l'overhead rimane)

tcpcrypt

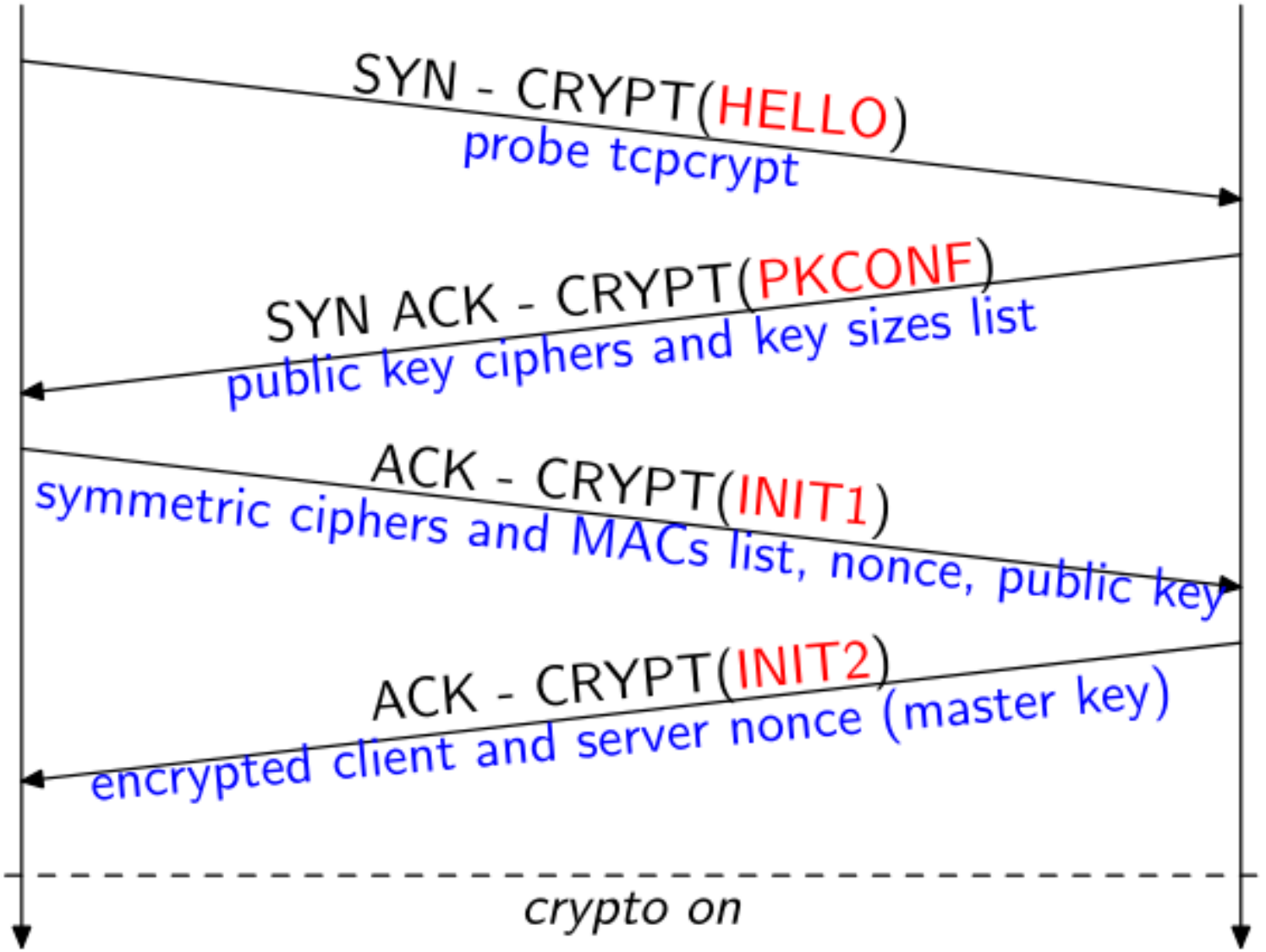
- Estensione di TCP
- Il carico computazionale crittografico è per lo piú spostato sui client
- Opportunistic encryption: attiva solo se supportata da entrambi (attenzione agli attacchi attivi!)

tcpcrypt handshake



36 volte piú veloce di TLS

tcpcrypt handshake



Autenticazione

Non c'è autenticazione del server con CA come nel caso di TLS, ma un session ID probabilisticamente unico (anche quando uno dei nodi è malevolo).
Un segreto condiviso k può essere usato cosí

1. $C \rightarrow S : HASH(k, C|SessionID)$
2. $S \rightarrow C : HASH(k, S|SessionID)$

Se anche S è malevolo (e k non generabile da un dizionario), non potrà riusare k (non estraibile da $HASH(k, C|SessionID)$) né $HASH(k, C|SessionID)$ perché il $SessionID$ sarà diverso.

Riassumendo

- tcpcrypt è un'estensione di TCP, che permette di cifrare il livello di trasporti
- è molto piú efficiente di TLS perché il carico crittografico è per lo piú spostato sui client
- Il Session ID permette di costruire protocolli di autenticazione a livello applicativo