Esame di Reti di Calcolatori e Sicurezza

17 luglio 2006

1. (a) Quali scelte sono state fatte nello standard 802.3u (Fast Ethernet) per raggiungere i 100Mbps?

   (b) Sempre in 802.3u, quale codifica viene usata sul doppino Cat.5, e perché?

   (c) Quale importante differenza presenta lo standard 802.3z (Gbit Ethernet)? Cosa comporta questa scelta nel caso di connessioni lente come telnet o ssh?

   Risposta:

   (a) (2pt) Stesso frame di 802.3, ma tempo di propagazione massimo 10 volte inferiore a 802.3, ossia 5 μsec.

   (b) (2pt) la 4B5B: 4 bit codificati con 5 bit, quindi solo 16 combinazioni su 32 sono significative. La velocità di segnalazione è 125Mbaud. Non si può usare Manchester perché Cat.5 non sostiene 200MHz, e la 4B5B garantisce sufficiente variazione del segnale per sincronizzare le stazioni.

   (c) (2pt) Il frame è lungo almeno 4096 bit=512 byte (trasmesso su fibra o su doppino con opportune codifiche). Connessioni con datarate basso producono frame molto corti (pochi byte, spesso 1-2), che richiedono un congruo padding e quindi bassa efficienza.

2. (a) Si disegni il diagramma temporale di una transazione CSMA/CA con RTS/CTS, indicando con precisione gli intervalli DIFS e SIFS tra i vari frame.

   (b) Si ricordi che in 802.11b è DIFS=50 μsec, SIFS=10 μsec; il frame RTS è lungo 20 byte, mentre i frame CTS e ACK sono lunghi 14 byte. Quale sarebbe l'efficienza del canale se il payload fosse sempre di 1500 byte e in assenza di collisioni e disturbi?

   (c) Tuttavia, se ogni frame trasmesso viene riconosciuto con un altro frame di 40 byte (contenente l'ACK del protocollo TCP), qual è il transfer rate massimo ottenibile?

   Risposta:

   (a) (2pt) Il diagramma è quello in Figura 4.27 sul Tanenbaum, dove l'intervallo prima di RTS è DIFS, mentre tra RTS e CTS, tra CTS e Dati, e tra Dati e ACK è SIFS.

   (b) (3pt) Il frame dati 802.11 ha un overhead di 34 byte, quindi in una transazione complessivamente vengono trasmessi 20+14+1534+14=1582 byte, che a 11Mbps prendono $1582*8/11 = 1150\mu$sec. A questo bisogna aggiungere un DIFS e 3 SIFS che prendono 50+3*10=80 μsec. Complessivamente la trasmissione prende $1150 + 80 = 1230\mu$sec. Di questi, sono utili solo quelli occupati dai 1500 byte del payload, ossia $1500 * 8/11 = 1091\mu$sec. L'efficienza è quindi $1091/1230 = 88,7\%$.

   (c) (3pt) In questa situazione, per ogni frame da 1500 byte viene trasmesso (nella direzione opposta) anche un frame da 40 byte, anch'esso incapsulato in un frame dati 802.11. Analogamente al punto precedente, la trasmissione di ritorno contiene effettivamente $20 + 14 + 74 + 14 = 122$ byte che prendono $122 * 8/11 = 88,7\mu$sec, più 80 μsec di intervalli DIFS e SIFS. Quindi la trasmissione di 1500 byte utili impiega $1230 + 88,7 + 80 = 1398,7\mu$sec. Il transfer rate è quindi $1500/1398,7 = 1,07$ Mbyte/sec = 8,58 Mbps

3. Alcune piattaforme adottano il *code signing* per vari tipi di codice mobile (p.e., driver, controlli ActiveX,...). Un esempio di protocollo code signing è il seguente:

   1) Alice chiede a Bob, distributore di software, un certo codice eseguibile.

   2) Bob spedisce a Alice il messaggio "X, N_C, S", dove X è il codice richiesto (o presunto tale), N_C è il nome del programmatore, Carol, che avrebbe prodotto tale software, e S è la presunta firma digitale.

   3) Alice non si fida di Bob e richiede ad una certification autority CA (di cui si fida) la chiave pubblica KP_C di Carol

4) Alice verifica che $D_{KP_C}(S) = H(X)$, dove H è una funzione di hash prefissata. In caso affermativo, il codice X viene eseguito da Alice; altrimenti viene rigettato.

In caso di successo, quali delle seguenti proprietà vengono garantite da questo protocollo?

(a) X è stato "veramente" (=con altissima probabilità) scritto da Carol. (Perché?)

(b) X è stato ottenuto veramente da Bob e non da qualche intruder che si sostituisce a Bob.

(c) X è esente da errori e malfunzionamenti.

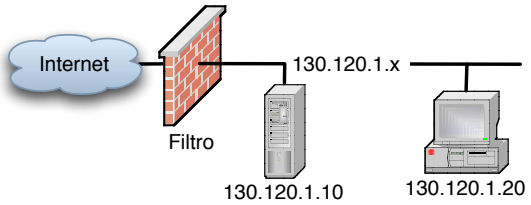
Risposta:

(a) (3pt) Sì: affinché il test abbia successo deve essere $S = E_{KS_C}(H(X))$, dove KS_C è la chiave segreta di Carol e quindi solo Carol può aver prodotto S. Rimane tuttavia la possibilità di un attacco a compleanno: Bob o un intruder potrebbe aver sostituito X con un X' con lo stesso hash, ossia $H(X) = H(X')$. Questa possibilità viene minimizzata scegliendo una funzione di hash sufficientemente robusta.

(b) (2pt) No: il codice può essere ottenuto anche da un intruder, ma questo non cambia la certificazione del codice di cui sopra.

(c) (2pt) No: lo schema proposto non garantisce nessuna proprietà del codice, tranne la sua origine. Il codice potrebbe anche essere `rm -rf /`.

4. Si consideri la seguente rete con un filtro di pacchetti stateful le cui regole sono date sotto:



IP:Port sorg	IP:Port dest	Proto	Azione
:	130.120.1.10:*	*	accept
130.120.1.10:*	*:*	*	accept
:	*:*	*	reject

(a) Cosa accade ad un pacchetto TCP originato da 158.110.1.7:32753 per 130.120.1.10:25? Ed ad un pacchetto da 130.120.1.20:45323 per 158.110.1.7:25?

(b) L'host 130.120.1.20 può utilizzare un server SMTP (porta 25) esterno alla rete? Se no, come devono essere cambiate le regole per consentirlo?

(c) È possibile che un pacchetto dall'esterno raggiunga 130.120.1.20, senza che sia una risposta ad una connessione?

Risposta:

(a) (2pt) Viene accettato. Viene rifiutato.

(b) (2pt) Non può. Bisogna aggiungere in penultima posizione la regola seguente:

IP:Port sorg	dest	Proto	Azione
130.120.1.0:*	*:25	TCP	accept

(c) (2pt) Sì: è sufficiente sostituire artificiosamente l'indirizzo IP sorgente con 130.120.1.10 ("IP spoofing").

5. [Da sviluppare su un foglio a parte]

Si descriva la fase di Fingerprinting della metodologia di attacco; in particolare, si descrivano gli obiettivi di tale fase e le tecniche di banner grabbing, OS fingerprinting (attivo e passivo) e Service fingerprinting.