Oggi parliamo di un altro protocollo, utilizzato a livello di rete locale (a livello di Data Link), che è il protocollo 802.11, anche chiamato “Wi-fi”. Il protocollo è apparentemente semplice, ma in realtà ha una serie di complicazioni. Questo protocollo è imparentato col protocollo 802.3, che è il protocollo Ethernet. Infatti, il Wi-fi è sostanzialmente la versione wireless del protocollo ethernet.

Le cose uguali tra i due protocolli sono gli indirizzi MAC (il formato è condiviso tra 802.3 e 802.15). Essi sono rappresentati sempre su 6 byte univoci della rete locale. Di Ethernet abbiamo visto due versioni principali (La HUB e la Switch): Se nella versione Hub si usa il protocollo CSMA/CD (per risolvere eventuali collisioni), nella versione Switch si usa la tecnica dello store and forward (che trasforma la connessione da broadcast a punto punto). Quindi abbiamo una comunicazione punto punto bidirezionale tra Host e Switch, tramite canali full-duplex, nella versione Switch.

Il Wi-fi è una via di mezzo tra i due tipi di approccio. L’idea è che un host deve essere dotato di un’antenna, attraverso la quale possa ricevere segnali elettromagnetici di tipo radio: un host potrebbe collegarsi ad un altro usando appunto le onde radio. Dal punto di vista informatico questo tipo di comunicazione (tra host e host via canale radio) viene chiamata “comunicazione Peer to Peer (P2P)”. Esempi di queste comunicazioni sono le ricetrasmittenti, che infatti funzionano solo entro una certa distanza. Una soluzione di questo genere viene utilizzata nel protocollo bluetooth (che permette la gestione di una connessione che avviene tra host a breve distanza).   
La qualità di questa connessione dipende dalla distanza, dalla frequenza e dalla presenza di ostacoli tra peers. In assenza di ostacoli, nel caso di bluetooth si possono raggiungere anche diversi, anche decine di, metri.

Sulla potenza utilizzabile per i segnali radio ci sono una serie di normative (onde evitare comunicazione via microonde).

L’alternativa alla connessione peer-to-peer, che viene adottata dal protocollo Wi-Fi, è la “Infrastructure”. Ciò funziona in maniera vagamente simile al caso dell’ethernet di tipo switch: abbiamo i nostri host, dotati di antenna, che si connettono anziché tra di loro a un altro dispositivo chiamato “access point”. L’access point di solito è un dispositivo fisso (mentre gli altri possono essere mobili): l’idea è che quando i dispositivi mobili si avvicinano abbastanza all’access point, entrando nel suo raggio di copertura, viene loro permessa la connessione.

Nell’infrastruttura wi-fi, se abbiamo gli host h1 e h2, h1 per parlare con h2 deve connettersi all’access point a cui è connesso h2 e inviare lì il frame wi-fi (header + footer, con dentro il payload), aspettandosi che l’access point recapiti il messaggio. I due host devono essere abbastanza vicini da connettersi via radio all’access point, ma non devono necessariamente essere vicini TRA DI LORO (potrebbero essere a 38 metri di distanza tra di loro, ma se l’access point si trova in mezzo non ci sono problemi): L’access point funziona anche da ripetitore.

Il protocollo wi-fi dunque, usa dei messaggi di tipo broadcast ma sfruttando il meccanismo degli switch di store and forward. Poiché c’è un unico canale condiviso, è necessario utilizzare un protocollo che tenga conto della possibilità di collisione. Il protocollo usato da 802.11 è quindi chiamato CSMA/CA (Carrier Sense, Multiple Access/ Collision Avoidance).

Perché collision avoidance e non detection? Per un problema pratico: non è possibile realizzare un trasmettitore via radio che possa trasmettere e ricevere contemporaneamente, contrariamente a quello che accade su fili elettrici.

Non possiamo avere ricezione e trasmissione simultanea perché la propagazione avviene nello spazio a 3 dimensioni e man mano che ci allontaniamo cambia la quantità di energia che si riesce a catturare con un’antenna (l’energia catturata equivale all’area dell’antenna diviso l’area del fronte d’onda complessivo, cioè della sfera che si sta allargando e che rappresenta l’onda), dunque la quantità di energia catturata è l’inverso del quadrato della distanza percorsa dal segnale. Questo crea grandi sproporzioni tra segnali inviati e segnali ricevuti (un segnale che ha inviato qualcun altro sarà ben diverso, forse addirittura impercettibile, da quello che ho inviato io). Dunque su un canale radio non si può realizzare un collision detection sul lato del trasmittente. L’unico che può rendersi conto di una collisione è l’access point, in caso riceva contemporaneamente più segnali da host diversi.

Questo problema viene risolto col collision avoidance, che implicherebbe che la collisione viene evitata, ma non è così. Se dal punto di vista del mittente non è possibile accorgersi della collisione, questo continuerà a trasmettere imperterrito. Ad accorgersi del problema sarà l’access point che informerà il client se l’operazione è andata a buon fine oppure no. Per questo motivo, è introdotto un acknowledgement che viene inviato dall’access point all’host mittente se l’invio del messaggio va a buon fine. Se ci sono collisioni, l’acknowledgement non può essere inviato (perché l’access point non ci capisce nulla); nel momento in cui gli host non si vedono arrivare l’acknowledgement dopo un certo timeout gli host assumono il peggio.

Se non arrivano gli acknowledgement nei tempi stabiliti gli host fanno il back-off, come nel caso di ethernet: ossia si aspetta per un tempo casuale, scelto su un intervallo di valori progressivamente più lungo. Prima o poi i due o piu host in collisione si differenzieranno e riusciranno a inviare i propri messaggi.

Il CSMA/CA prevede due tempi base: il tempo “corto”, quello necessario per ottenere l’acknowledgement se le cose sono andate a buon fine, e un tempo “lungo” che indica l’intervallo di vulnerabilità, sul quale si basa il back-off in caso di collisione.

Supponiamo che h1 e h2 siano molto vicini. In questo caso h2 può sentire la trasmissione di h1 (anche meglio dell’access point probabilmente): cosa deve fare in questo caso? Non deve assolutamente riceverlo, ma aspettare che l’access point lo rinvii. Questo porta a un ritardo nei segnali rispetto al protocollo CSAM/CD di ethernet, tuttavia, è necessario per assicurare il funzionamento della rete sia che i dispositivi siano vicini che lontani.

Come fa h2 a distinguere tra i due segnali (quello di h1 e quello forwardato dall’access point) per scegliere di ascoltare solo il secondo: La scelta che è stata fatta è stata di dare un indirizzo MAC anche all’access point. Quindi per far funzionare la connessione servono indirizzo MAC del mittente, del destinatario e dell’access point (l’indirizzo di destinazione è quindi in realtà un indirizzo di “inoltro” nella prima fase del messaggio, dal momento che tra una fase e l’altra l’access point passa da “falso” destinatario a “falso” mittente). L’utilizzo di un indirizzo mac in più (3 anziché 2) rende le cose più complicate rispetto al protocollo ethernet, infatti, lo switch non aveva bisogno di alcun indirizzo mac, visto che le sue porte lo connettevano a tutta la rete.

Vediamo quindi la struttura dei frame wi-fi (802.11). Abbiamo una parte header, costituita da 2 byte che specificano il tipo di messaggio, poi abbiamo 3 indirizzi MAC su 6 byte ciascuno (indirizzo mittente, destinazione e di forward). Poi abbiamo altre informazioni che fanno parte dell’header, poi abbiamo il payload e infine abbiamo un footer di 4 byte che fanno controllo di ridondanza. Il controllo di ridondanza permette di stabilire con un certo grado di confidenza il fatto che ci siano stati degli errori o meno durante la trasmissione del messaggio, attraverso un algoritmo (di consistenza?). Si usa il contenuto del messaggio per calcolare un valore da inserire nel CRC (footer) e ogni ricevente fa il controllo con questo valore e il risultato dell’algoritmo applicato al messaggio che ha ricevuto.   
Il messaggio di acknowledgement viene inviato dall’access point dopo che ha verificato il controllo del CRC.

Su rete fissa è molto improbabile che fallisca il controllo sul CRC, mentre su rete wireless via radio è molto più probabile che accada, per via delle collisioni.

Esistono degli altri algoritmi che vengono utilizzati per migliorare il canale di comunicazione, però questi vengono usati a livello di codifica dei segnali, quindi fanno parte puramente del livello fisico. A livello fisico viene quindi utilizzato un codice a correzione di errore (Forward Error Correction = FEC). Già a livello di codifica dei bit viene usato questo codice ridondante, inviando più bit di quelli strettamente necessari e usando quelli aggiuntivi per verificare la presenza di eventuali errori di connessione, provvedendo a correggere questi errori se si presentano su pochi bit.

Esempio terra-terra di un possibile funzionamento: Se io volessi mandare 4 bit di dati, potrei aggiungere un quinto bit di parità, che indichi una qualche condizione che devono rispettare i quattro bit precedenti. Se volessi inviare i bit 1100 mi aspetto che al bit di parità ci sia 0 e 1 nel caso di 1000 (il bit di parità usa la funzione XOR). Questo permetterebbe anche di correggere eventuali errori in caso di situazioni non gravi. La cosa è scalabile per qualsiasi k bit. Potremmo usare 2 bit per il controllo FEC per ogni 4 bit di dati, oppure 3. Questo rallenta la trasmissione sul canale ma lo rende più affidabile.

I tipi di frame definiti nell’802.11 sono 4 (identificati da due bit all’inizio: 00, 01, 10, 11).

Abbiamo i frame di gestione (00), i frame di controllo (01), i frame di dati (10) e i frame di estensione (11).

Dopo la suddivisione su due bit, abbiamo un ulteriore suddivisione per sottotipo su 4 bit.  
Tra i messaggi di estensione ne abbiamo alcuni che vengono chiamati “beacon”, in cui l’access point di presenta agli host con il suo nome: tale nome è chiamato SSID. Questi frame sono inviati con cadenza regolare.

Tra i sottotipi del tipo dati, ce ne sono una quantità molto elevata che non sono utilizzati, quello utile nel 99% dei casi è il tipo 0 (cioè il tipo data, che contiene i dati in maniera pura e semplice).

Tra i sottotipi del tipo di controllo c’è l’acknowledgement, importante per confermare l’assenza di collisioni (si noti che nel protocollo TCP non c’è un datagramma particolare per l’acknowledgement, si fa piggy-backing su altri datagrammi che contengono dati, mentre nel wi-fi si usano datagrammi specializzati).  
L’acknowledgement è codificato col numero 13 (1101).  
Altri frame di controllo degni di nota sono gli RTS e i CTS.

RTS sta per Request To Send e CTS per Clear To Send. Questo permette di creare un dialogo tra host e access point (l’host funziona da client e l’access point funziona da client). Tipicamente prima di inviare i dati, un host invierà un Request To Send, un messaggio molto piccolo che permetterebbe di perdere poco tempo in caso di collisione. L’access point, se il canale è libero, risponderà un con un Clear To Send (ovviamente un solo host per volta potrà ricevere l’ok dal messaggio di tipo CTS). Il Clear To Send però, è inviato in broadcast a tutti gli host, specificando qual è l’unico host che ha il diritto di inviare messaggi per una certa quantità di tempo. In questo modo, ogni host riconoscerà i messaggi CTS e guardando qual è l’indirizzo destinatario, può capire se lui ha il diritto di mandare messaggi in quel momento o no.

Questo meccanismo è complementare al CSMA/CA, che viene utilizzato quando non c’è coordinazione, quando c’è coordinazione si può usare il meccanismo dell’RTS e del CTS per evitare le collisioni.

Nota: Nel frame ci sono anche due byte dedicati al tempo dopo il quale si pensa si possa considerare terminato l’inivio del messaggio (e ciò viene utilizzato per stabilire quanto tempo dare a un host nel messaggio CTS e fa anche da timeout per gli host).

Preferibilmente si usa il CSMA/CA per inviare il RTS, e poi si usa la “protezione” del CTS per inviare i datagrammi con frame di tipo data.

Il Tipo Management serve per gestire le interazioni tra host e access point o tra access point e access point nel caso ce ne siano più di uno nelle immediate vicinanze. Per quanto riguarda la comunicazione tra host e access point la cosa funziona in questo modo: ogni tanto l’access point si pubblicizza col suo beacon, a questo punto un host può inviare la sua richiesta all’access point in cui gli chiede di essere associato a lui (e che gli venga assegnato un indirizzo IP). Questo tipo di richiesta di associazione è inviata con un frame di Management, così come il messaggio di risposta da parte dell’access point.  
Altri messaggi di questo tipo sono i Keep Alive e i messaggi di dissociazione (in cui l’host notifica che si sta staccando dalla rete locale) e altri messaggi necessari per le operazioni di cifratura.