Reti teoria

sabato 17 dicembre 2022 14:16

Strati

Applicativo

Elaborazione dei dati. Origine dell'informazione
Comprende: URL, HTTP (e altri web), FTP, TELNET, SMTP (e altri di posta elettronica), DNS

Presentazione

Unificazione dei dati, preparazione del pacchetto

Sessione

Dialogo tra host

Trasporto

Offre servizi allo strato applicativo
Collega processi in host diversi, trasferisce dati tra host, realizza dialogo end to end
Comprende: TCP, UDP

Rete

Realizza una connessione logica tra host diversi: interconnette reti.
Offre servizi al livello di trasporto
Instradamento del traffico
Comprende: IP, ARP

Datalink

Consegna il frame tra le interfacce
Comprende: Ethernet

Fisico

Segnale elettrico
Comprende: bit su file

Grassetto parte di TCP/IP

Servizi di applicazione

HTTP

Protocollo generico, stateless e object oriented
Porta 80
Modello client-server
Request/response: connessione viene iniziata dal client. Ogni coppia è indipendente.
Utilizza connessione TCP.
HTTP 1.0: connessione separata per ogni url. Aumenta il carico, congestioni. Non persistente.
1.1: Persistente. Prevede un meccanismo di chiusura della connessione
2: maggiore flessibilità lato server, trasmissione con priorità, oggetti divisi in frame. Supporta multiplexing. Problema: Head Of Line blocking - perdita provoca lo stallo
3: aggiunge sicurezza, controllo di errori e congestione su UDP. Basato su QUIC, UDP connection oriented

Metodi

Safe

Non hanno effetti collaterali
GET, HEAD, OPTIONS, TRACE

Idempotenti

Non hanno effetti ulteriori se fatti n>1 volte
GET, HEAD, PUT, DELETE, OPTIONS, TRACE

Telnet

Permette accesso remoto, multiplo a un computer
Coppie client server applicative non specializzate
Servizio trasparente. Usa TCP
Passa tutto in chiaro -> SSH (usa servizi di cifratura, molto più potente)

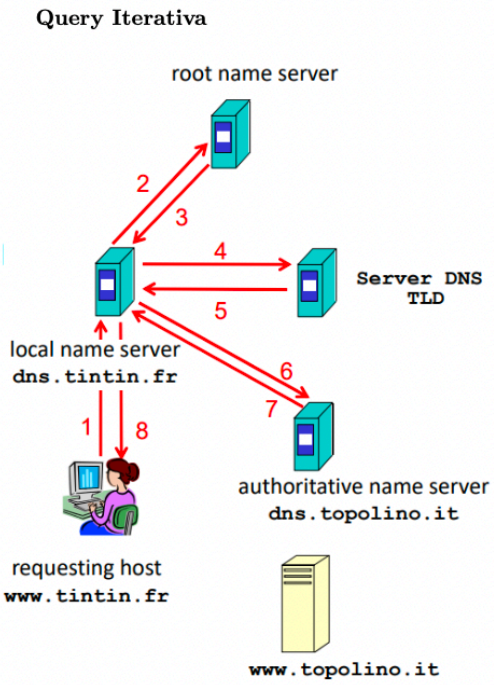
FTP

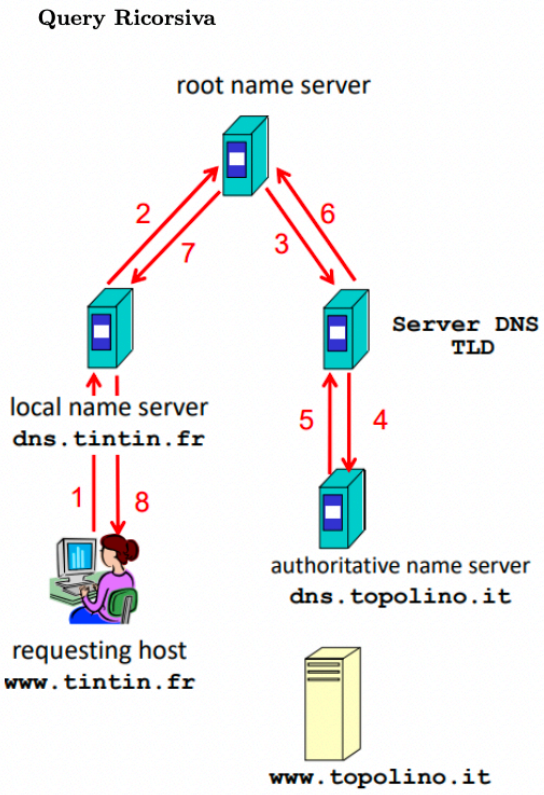
Trasferimento di file, standard.
Stateful
Controllo: Porta 21. Connessione TCP, persistente, basata su telnet
Dati: TCP

Active: una connessione per ciascun trasferimento che poi viene chiuso. Porta specifica lato client
Passive: Porta 20, aspetta una richiesta di connessione

DNS

Impossibile gestire IP statici: sono troppi. Domain Name System
Struttura gerarchica, ad albero, punto punto
+ veloce, flessibile, aggiornabile
Specifica la sintassi, consente la conversione
Costituito da schema di assegnazione dei nomi, database e protocollo per la distribuzione tra i name server
Name server: programma che gestisce la conversione da nome di dominio a indirizzo IP.
Zona: regione di cui è responsabile un name server
Gerarchia: root servers in testa, collegati a top level domain s, collegati a altri server primari (mantengono file di zona) e/o secondari (offrono anche traduzione)
Risoluzione dei nomi salendo per albero, o ricorsiva o iterativa (risposte restituite direttamente a client)
Header messaggi 12 byte
Usa UDP su porta 53





SMTP

Protocollo di posta elettronica, trasferimento dei messaggi di
TCP, porta 25

TCP, porta 25
Protocollo di tipo push

Solo messaggi di testo, problema con caratteri non ASCII e altri contenuti. Soluz. MIME
    Codifica ASCII Base64, quoted printable encoding per pochi char non ascii
    Obsoleto, ormai vengono trasmessi dati in 8bit, in caso fallisce si usa ASCII

POP3

Protocollo di tipo pull, legge messaggi di posta elettronica
TCP, porta 110
IMAP più complesso, manipola messaggi e permette di estrarre solo alcune componenti

Servizi di trasporto

Multiplexing: entità riceve info da più di una sorgente
    Provvede allo smistamento pacchetti tra rete e applicazioni
Demultiplexing: entità trasmette info a più destinatari
    Accompagnamento dei flussi dati dai processi verso la rete
Mux su mittente e Demux su destinatario

TCP

Connection oriented, affidabile, con controllo del flusso e congestione
Multiplexing assegna connessione a un processo
Usato in posta elettronica, accesso a terminali remoti, web, trasferimento file
Servizio a stream, flusso di byte ordinati ma non strutturati
Trasferimento bidirezionale indipendente
Controllo della connessione: handshake
Numera i byte invece dei segmenti
    Nr di sequenza: nr del primo byte del segmento
    Nr di riscontro (ACK): 1+ nr dell'ultimo byte ricevuto. Aspetto il byte y
Intestazione: da 20 a 60 byte
Three way handshake:

1. Client invia richiesta di connessione
2. Server estrae segmento, alloca buffer e variabili, risponde con segmento di connessione garantita (SYNACK)
3. Client alloca buffer e variabili e manda riscontro positivo. Inizia lo scambio

Chiusura: dopo l'ack finale è immediata.
Half close: manda un messaggio I have no more data to send. Termina con Time Wait

Half close: manda un messaggio I have no more data to send. Termina con Time Wait
Attesa di 2MSL(massimo tempo pacchetto IP vive sulla rete)
Implementa la terminazione in entrambe le connessioni
Consente eliminazione di duplicati

Trasferimento dati affidabile: comunicazione ack da parte di destinatario
Lato mittente:
Timeout, no ack ricevuto: ritrasmetto segmento, riavvio timer
3 ack duplicati: segmento andato perso, ritrasmetto rapidamente prima che scada timer
Destinatario:
Segmento in ordine: ritarda ack di 500ms || arriva un altro segmento
Segmento atteso ma non il precedente: invia immediatamente ack
Seg. Fuori sequenza, mancante, duplicato: invia immediatamente Ack con il prossimo atteso

Calcolo del timeout
RTO: retransmission timeout, deve essere maggiore di RTT
RTO = est. RTT + varianzaRTT*4
RTT: tempo trascorso da quando si invia un messggio a quando si riceve il riscontro
Controllo di flusso: capacità di evitare di saturare buffer receiver. Mette in relazione la frequenza di invio con quella di lettura
Finestra di trasmissione
Per il controllo del flusso. Si sovrappone alla sequenza da trasmettere, avanza alla ricezione di un ACK
Host imposta buffer invio e ricezione. Dest legge da buf ricezione.
Receive window: variabile mantenuta dal mittente, dice quanto spazio è ancora disponibile. Comunicato da dest nell'header a ogni messaggio
Se window = 0 vengono mandati segmenti di sonda di 1 byte per ricevere aggiornamenti

Controllo della congestione: incremento additivo, decremento moltiplicativo
Slow start: cWnd partenza a 1 mss, incremento 1mss a ogni ack: crescita esponenziale. Quando ERR, dimezza
Soglia: valore oltre cui comincia congestion avoidance con AIMD
AIMD: crescita lineare
Politica Reno
Fast recovery: soglia a cWnd/2, cWnd a soglia +3
Time out: soglia a cWnd/2, cWnd a 1mss (slow start)
Tahoe: entrambi in slow start

UDP

Connectionless, non affidabile, nessun controllo
Meno complesso, semplice, basso throughput
Consente controllo completo della temporizzazione: più veloce
Datagrammi indipendenti tra loro, servizio a messaggi
Usato nei servizi di streaming multimediale e telefonia -> possono tollerare perdite
Intestazione: 8 bit

Servizi di rete

IP

Connectionless, non affidabile, senza garanzie su tempo di consegna e controllo di flusso
Intestazione 20-60 byte, 20 standard
Lunghezza dei datagrammi IP dettata da MTU, standard 1500 byte (20 di intestazione). Se più lunghi, frammentazione
    Uso byte di identificazione (id del datagramma in tempo adeguato), offset e flag(0: reserved 0 , 1: do not fragment if 1, 2: no more fragments if 0)
Indirizzi lunghi 32 bit (IPv4)
Classful addressing: rigido, poco pratico.
Classless: byte.byte.byte.byte/n, n bit più a sx sono network ID
Subnet mask: distingue quale parte identifica la rete e quale l'host. Messa in AND con IP permette di trovare la rete

DHCP:

assegna IP temporanei
UDP incapsulato in IP incapsulato in Ethernet
Forwarding
    Diretto: destinatario sulla propria rete, invio a destinatario direttamente
    Indiretto: destinatario su un'altra rete, delego invio a un router

NAT:

permette di trasmettere traffico su internet proveniente da sottoreti private. Router ha unico indirizzo IP pubblico e tutto il traffico in entrata e uscita ha quell'IP
    Il router ha in memoria una tabella di traduzione NAT

ICMP

ICMP

Usato da host e router per scambiarsi messaggi di errore
Incapsulati in datagrammi IP ma parte dello strato di rete
Instradati prima dei pacchetti ordinari
Per pacchetti frammentati, rif solo a offset 0
Mai inviati a IP che non rappresenta un unico host
Mai inviati a messaggi di errore ICMP, possibile risposta a interrogazione
Ping: Si basa su request response echo di ICMP. Verifica se host è attivo.
    Può misurare grossolanamente affidabilità e congestione
traceroute: individua percorso da sorgente a destinazione. Standard max 30 salti.
IPv6: Indirizzi v4 si stanno esaurendo. Spazio da 128 bit, frammentazione eseguita alla sorgente, classi di traffico e etichette di flusso
Double stack: necessario far convivere le due versioni
Tunneling: header v4 aggiunto davanti header v6 se router di destinazione non supporta v6

Router

accetta pacchetti in entrata, usa una tabella d'inoltro per trovare la porta e invia il pacchetto. Diviso in due parti:
    Data plane: analizza e instrada i pacchetti in entrata. Attraversano il router.
    Control plane: Interni al router. Servono per aggiornare le proprie informazioni
Routing statico: entry configurate manualmente, si prevedono tutti i percorsi possibili. Reti di piccole dimensioni
Routing dinamico: Necessari protocolli specifici per inserimento in tab, reti medio grandi e a topologia variabile
Operano solo su frame il cui indirizzo di destinazione è quello dell'interfaccia su cui arrivano

Algoritmi

Link state: Tutti i nodi hanno informazioni, c'è una tabella con tutte le distanze dei nodi memorizzata. Calcola cammino di costo minimo con algoritmo di Dijkstra globale
    O(nE) messaggi, algoritmo O(n2). Oscillazioni di velocità, poco robusto perché router può comunicare costo sbagliato
Distance vector: Algoritmo di Bellman Ford, memorizza i distance vectors iniziali dei nodi
decentralizzato
    Tempo di convergenza varia, cicli di instradamento, count to infinity problem, un calcolo errato si può diffondere per l'intera rete, la tabella può essere usata da altri

33.4.5 Link-State vs Distance Vector

Complessità dei messaggi

Link-State: con n nodi, E collegamenti, si inviano O(nE) messaggi.

Distance Vector: richiede scambi tra nodi adiacenti. Il tempo di convergenza può variare

Velocità di convergenza

Link-State: l'algoritmo $O(n^2)$ richiede O(nE) messaggi. Ci possono essere oscillazioni di velocità

Distance Vector: può convergere lentamente. Può presentare cicli d'instradamento. Può presentare il Count-To-Infinity problem

Robustezza

Link-State: un router può comunicare via broadcast un costo sbagliato per uno dei suoi collegamenti connessi, ma non per altri. I nodi si occupano di calcolare soltanto le proprie tabelle.

Distance Vector: un nodo può comunicare cammini a costo minimo errati a tutte le destinazioni. La tabella di ciascun nodo può essere usata dagli altri. Un **calcolo errato si può diffondere per l'intera rete**.

Struttura di internet

Insieme di router organizzati in sistemi autonomi

Un gruppo connesso di una o più reti

Decisione autonoma di protocolli e politiche di routing interne

- Stub: AS collegato solo a un altro AS
- Multihomed: collegato a più AS, trasporta solo traffico di cui è origine o destinazione
- Transito: tutti gli altri

Protocolli

RIP

- INTRA-AS
- Metrica: nr sottoreti attraversate (max 15)
- Distance vector con poisoned reverse

OSPF

- INTRA AS, link state
- Metrica decisa da admin per i cammini (1 per closest)
- Manda a tutti i router in caso di aggiornamento
- Elevato nr di messaggi, rischio flooding -> divido in aree collegate a una backbone

BGP

- INTER AS, unico usato
- Coppie di router si scambiano info su tcp semi permanenti
- Distance Vector, ha route interne (iBGP) e esterne (eBGP), sono i router che comunicano e non gli AS
- Aggregazione degli indirizzi, distribuzione delle info di raggiungibilità, politiche di importazione
- scelta delle rotte, regole

1) Preferenza locale
2) AS_path più breve
3) Next_HOP più vicino (hot potato)

Livello Data Link

Frame: unità di dati scambiate a livello link

Muove i datagrammi da un nodo a un altro adiacente lungo un singolo canale di comunicazione

Tipologie di collegamento

- Punto a punto: dedicato a due dispositivi
- Broadcast: condiviso tra più dispositivi, diffonde a tutti i canali collegati

Framing: separa i vari messaggi durante la trasmissione

Affidabilità non necessaria su collegamenti con basso numero di errori

Errori causati da attenuazione del segnale e rumore elettromagnetico

Protocolli ad accesso multiplo

Se un nodo riceve 2 o + segnali: collisione. Da evitare

A suddivisione del canale

- Dividono il canale in pezzi più piccoli
- Risorse allocate in modo esclusivo
- TDMA: accesso a intervalli di tempo, time slot di lunghezza fissa
- FDMA: canale diviso in bande di frequenza. Ogni stazione 1 banda fissa

Random access

- Canale condiviso, possono esserci collisioni
- Recupero da collisione
- Slotted ALOHA
  - time slot uguali e fissi, trasmissione a inizio slot. Se canale libero pog, se occupato ritrasmetto con probabilità p nello slot successivo fino a successo.
  - Con grandi numeri di nodi, low success rate

ALOHA puro (unslotted)

Niente sincronizzazioni, dati trasmessi immediatamente. Probabilità collisioni maggiore

CSMA

Se il canale è libero trasmette, occupato ritarda. Non si interrompe chi parla

Ritardo di propagazione può causare collisioni

CSMA/CD

Trasmissioni che collidono vengono abortite

Difficile collision detection in wireless

Rotazione

Rotazione tra i nodi

Polling: master invita nodi slave a trasmettere a rotazione. Overhead, latenza, single point of failure

Token passing: token passato da nodo a nodo. Soliti svantaggi

Altri protocolli: bluetooth

Indirizzo MAC

Noto anche come indirizzo LAN o indirizzo fisico

Associato alla scheda di rete, permanente. Lungo 6 byte, espresso in hex.

Primi 24 bit assegnati da IEEE, altri 24 lasciati alle aziende

Tutte FF è broadcast nella LAN

ARP

Una macchina conosce il proprio MAC ma non sa chi ha attorno: Address Resolution Protocol

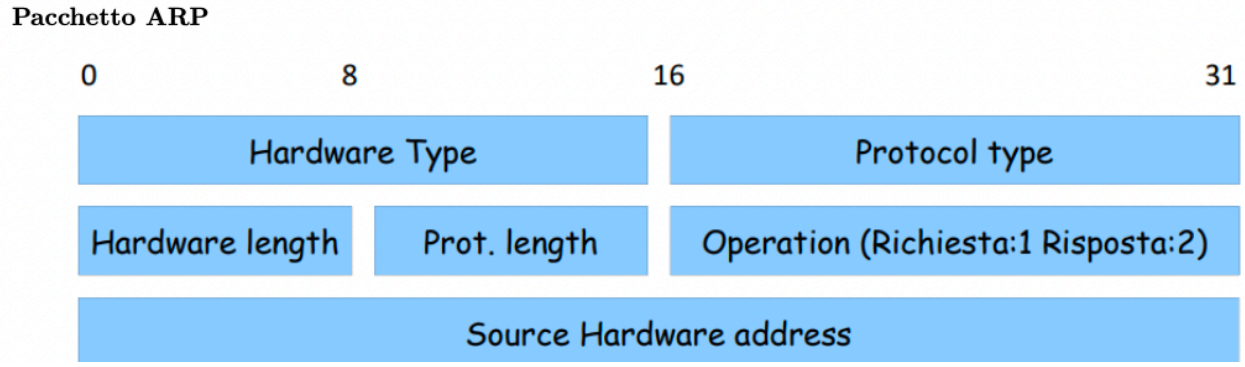
Richieste ARP fatte in broadcast, non conosco i MAC vicini

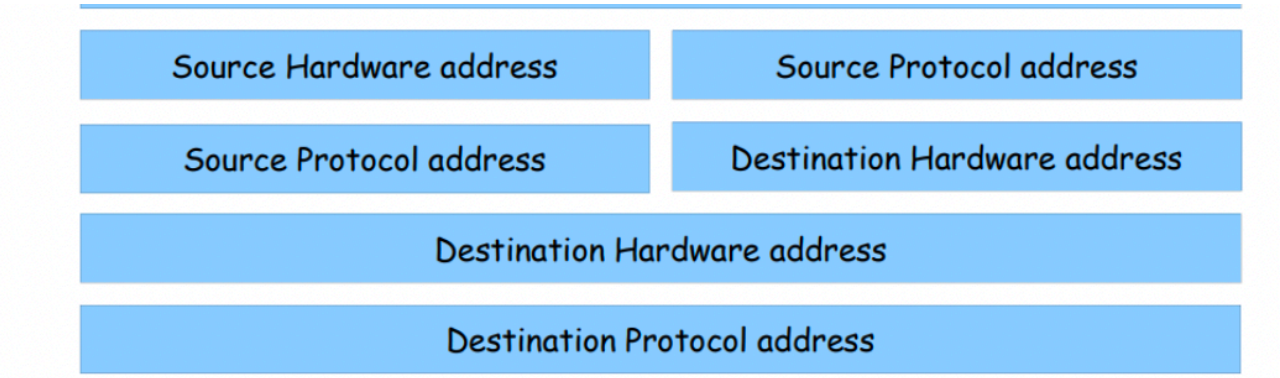
Ogni nodo IP ha una tabella ARP. Tabelle ARP contengono corrispondenze IP-MAC per i nodi della stessa sottorete, non necessariamente tutti

Protocollo di rete

Ogni nodo della rete riceve e elabora pacchetto richiesta ARP, se riconosce il proprio IP restituisce pacchetto di risposta con IP e MAC in unicast, altrimenti scarta.

La tabella ARP si costruisce automaticamente, non deve essere configurata





Hardware Type Protocollo di livello data link (es. Ethernet 1)

Protocol Type Protocollo di livello superiore (es. IP)

Source Hardware Address
Source Protocol Address Indirizzi del nodo mittente a livello link e superiore. Lunghezza variabile.

Destination Hardware Address Vuoto nelle richieste
Destination Protocol Address

Ethernet

LAN ad alta velocità. Dominante perché semplice e up-to-date
Topologia a stella, host collegati a hub/switch.
I datagrammi IP vengono incapsulati in pacchetti ethernet

Dispositivi di interconnessione

Repeater: solo livello fisico (segnale)
Hub: repeater multiporta. Tutto viene mandato a tutti i dispositivi collegati
Switch: livello fisico e link (verifica MAC nel frame). Tabella di filtraggio, non modificano MAC
Router: Liv fisico, link e rete (verifica IP). Ogni router ha 1 ip e 1 MAC per ogni interfaccia. Cambiano indirizzi link su cui operano

VLAN

Dominio broadcast di un gruppo di terminali non vincolati fisicamente che possono comunicare come se fossero sulla stessa LAN
Consentono di disperdere fisicamente le pozioni di rete senza perdere l'identità di gruppo
Concetto a livello di data link, implementata negli switch
Letteralmente LAN virtuali
Può attraversare più switch interconnessi, necessario stabilire chi appartiene alla VLAN

- Per gruppo di porta: una porta per ogni specifica VLAN, facile da configurare

ma va riconfigurata se un terminale cambia porta
Per MAC: l'appartenenza va assegnata inizialmente ma è persistente rispetto a spostamenti fisici
Per informazioni sul protocollo: flessibile, switch deve analizzare porzioni del MAC a livelli superiori->prestazioni --

Frame tagging: ogni frame ha un'intestazione relativa alla VLAN di cui fa parte

P2P

Tutti gli host agiscono sia da client che da server (peer)
Di base tutti i nodi hanno la stessa importanza. Sono indipendenti, autonomi e localizzati ai bordi di internet.
No controllo centralizzato
Sistemi altamente distribuiti, con anche O(100k) nodi
Nodi dinamici e autonomi (possono entrare e uscire dalla rete in qualsiasi momento)
Nella pratica sono presenti server centralizzati o nodi con funzionalità diverse (supernodi)
Peer possono essere inattivi e cambiare IP: come tenere traccia e trovarli?

Directory centralizzata: client-server, bottleneck
Reti strutturate: vincoli su grafo e posizionamento dei nodi. Principi rigidi, aggiunta e rimozione sono operazioni costose. Obiettivo localizzazione risorse

DHT: ogni peer ha un ID e conosce un certo nr di peer. Risorse condivise hanno ID basato su hash

Reti non strutturate: grafo casuale, no vincoli su posizionamento. Localizzazione difficoltosa. Aggiunta e rimozione semplice, gestisce nodi transient facilmente. Prone a query flooding.

Copertura gerarchica: no server centralizzati, group leaders potenti in banda o risorse. Group leader tiene traccia dei figli.

BitTorrent

Idea: dividere file in blocchi da 256 byte e far distribuire a ogni peer i dati ricevuti
Riduce carico sorgente, riduce dipendenza da distributore originale, fornisce ridondanza
Swarm: insieme di peer che partecipa alla distribuzione di un file scambiandosi parti (chunks)
Strategie

Tit for Tat: si inviano dati a peer che inviano dati, priorità a frequenza maggiore

Vicini classificati in choked e interested, lista aggiornata ogni 10s
Ogni 30s un choked a caso viene promosso a interested

Rarest chunks first

Sicurezza

Cifratura simmetrica: Chiave segreta condivisa per cifrare e decifrare. Come concordare la chiave? Segretezza a rischio. Ex. DES, AES

Asimmetrica: No chiavi segrete condivise, ognuno ha una chiave pubblica (nota) e una privata (segreta). Problema: efficienza. Ex. RSA.

Message digest: non è importante la segretezza ma l'integrità. Funzione hash.

Invio di M e MAC(=f(M)), dest ricalcola MAC e confronta.

Firma digitale per firmare digest

Ipsec

Mod. trasporto: protegge dati passati da livello trasporto a livello rete. Intestazione non protetta.

Mod. tunnel: protegge intero pacchetto IP

ESP: per autenticare la sorgente

Aggiunto un trailer ESP, payload e ESP crittografati, intestazione ESP, payloac e trailer ESP usati per generare dati di autenticazione, aggiungo intestazione IP