Autore: Prof. Gabriel Rovesti

1. INTRODUZIONE ALLE RETI

1.1 Ripasso Topologie e Modelli ISO/OSI - TCP/IP

Le topologie di rete definiscono la disposizione fisica o logica dei dispositivi di rete:

- **Stella**: tutti i nodi sono collegati a un nodo centrale
- **Anello**: ogni nodo è collegato a due nodi adiacenti formando un circuito chiuso
- **Bus**: tutti i nodi sono collegati a un unico canale di comunicazione
- **Maglia (mesh)**: ogni nodo è collegato a tutti gli altri (completa) o ad alcuni (parziale)
- **Albero**: struttura gerarchica con nodi che si diramano da un nodo radice

1.2 Definizione di Rete e Modelli

Una rete di computer è un insieme di dispositivi interconnessi che condividono risorse e comunicano tra loro.

Modello ISO/OSI (International Organization for Standardization/Open Systems Interconnection):

1. **Livello fisico**: trasmissione di bit grezzi tramite il mezzo fisico
2. **Livello data link**: framing e controllo degli errori
3. **Livello rete**: routing e indirizzamento dei pacchetti
4. **Livello trasporto**: trasferimento dati affidabile end-to-end
5. **Livello sessione**: gestione delle sessioni di comunicazione
6. **Livello presentazione**: rappresentazione e crittografia dei dati
7. **Livello applicazione**: interfaccia con le applicazioni utente

Modello TCP/IP (Transmission Control Protocol/Internet Protocol):

1. **Livello accesso alla rete**: corrisponde ai livelli 1 e 2 del modello OSI
2. **Livello internet**: corrisponde al livello 3 del modello OSI
3. **Livello trasporto**: corrisponde al livello 4 del modello OSI
4. **Livello applicazione**: corrisponde ai livelli 5, 6 e 7 del modello OSI

1.3 Differenze tra i Modelli

Caratteristica	ISO/OSI	TCP/IP
Numero di livelli	7	4

Caratteristica	ISO/OSI	TCP/IP
Approccio	Teorico	Pratico
Sviluppo	Prima il modello, poi i protocolli	Prima i protocolli, poi il modello
Flessibilità	Più rigido	Più flessibile
Adozione	Principalmente teorica	Standard de facto di Internet
Separazione	Chiara distinzione tra servizi, interfacce e protocolli	Meno distinzione

1.4 Enti di Standardizzazione

- **ISO** (International Organization for Standardization): sviluppa e pubblica standard internazionali
- **IEEE** (Institute of Electrical and Electronics Engineers): definisce standard per reti locali e metropolitane (802.x)
- **IETF** (Internet Engineering Task Force): sviluppa e promuove standard Internet, principalmente TCP/IP

1.5 Architetture di Rete

- **Client/Server**:
  - Server centralizzati forniscono servizi ai client
  - Facile gestione e sicurezza
  - Possibile collo di bottiglia e single point of failure
  - Esempi: web, email, database
- **Peer-to-Peer (P2P)**:
  - Ogni nodo può fungere sia da client che da server
  - Decentralizzato, più resiliente
  - Più difficile da gestire e proteggere
  - Esempi: BitTorrent, blockchain, alcune VoIP

1.6 Introduzione al Physical Layer

Il livello fisico è il livello più basso del modello OSI, responsabile della trasmissione di bit grezzi.

- **LLC** (Logical Link Control):
  - Sottolivello superiore del livello data link
  - Fornisce interfaccia al livello rete
  - Indipendente dal tipo di rete fisica

    - Controllo di flusso, rilevamento errori
- **MAC** (Media Access Control):
    - Sottolivello inferiore del livello data link
    - Gestisce l'accesso al mezzo condiviso
    - Indirizzamento hardware (MAC address)
    - Protocolli specifici per ogni tipo di rete (Ethernet, WiFi, ecc.)

2. LIVELLO FISICO

2.1 Livello LLC, HDLC, MAC ed Ethernet

- **LLC** (Logical Link Control):
    - IEEE 802.2
    - Multiplazione dei protocolli di livello superiore
    - Tipi di servizio: connectionless e connection-oriented
    - Frame: DSAP, SSAP, Control
- **HDLC** (High-level Data Link Control):
    - Protocollo di livello data link orientato ai bit
    - Modalità: NRM (Normal Response Mode), ARM (Asynchronous Response Mode), ABM (Asynchronous Balanced Mode)
    - Tipi di frame: Information, Supervisory, Unnumbered
    - Controllo di flusso e rilevamento errori
- **MAC** (Media Access Control):
    - Controllo accesso al canale condiviso
    - Indirizzo MAC: 48 bit (6 byte), univoco globalmente
    - Formato: OUI (Organizationally Unique Identifier, 24 bit) + NIC (Network Interface Controller, 24 bit)
- **Ethernet** (IEEE 802.3):
    - Standard dominante per LAN
    - Velocità: da 10 Mbps a 400 Gbps
    - CSMA/CD (Carrier Sense Multiple Access with Collision Detection)
    - Frame: Preambolo, SFD, MAC dest/src, Lunghezza/Tipo, Dati, FCS

2.2 Tipologie di Cavo

- **Cavi in rame**:
    - **Doppino intrecciato (Twisted Pair)**:
        - UTP (Unshielded Twisted Pair): senza schermatura
        - STP (Shielded Twisted Pair): con schermatura
        - Categorie: Cat5e (1 Gbps), Cat6 (10 Gbps), Cat7 (100 Gbps)
        - Vantaggio: economico, facile da installare

        - Svantaggio: sensibile a interferenze, distanza limitata
    - **Cavo coassiale**:
        - Conduttore centrale, isolante, schermatura, guaina
        - Tipi: thick Ethernet (10Base5), thin Ethernet (10Base2)
        - Vantaggio: buona immunità alle interferenze
        - Svantaggio: meno flessibile, più costoso
- **Cavi in silicio**:
    - Utilizzati principalmente in circuiti integrati
    - Altissima velocità di trasmissione
    - Distanze estremamente ridotte
- **Cavi ottici**:
    - **Fibra ottica**:
        - Monomodale: core piccolo (8-10μm), lunga distanza
        - Multimodale: core grande (50-62.5μm), distanze più brevi
        - Materiali: silice, plastica
        - Vantaggio: alta velocità, immune a interferenze, lunga distanza
        - Svantaggio: costo, fragilità, installazione complessa

2.3 Mezzi Trasmissivi, Caratteristiche e Segnali

- **Mezzi trasmissivi**:
    - Guidati (via cavo): rame, fibra ottica
    - Non guidati (wireless): radio, microonde, infrarossi, laser
- **Caratteristiche dei segnali**:
    - **Ampiezza**: intensità del segnale
    - **Frequenza**: numero di cicli al secondo (Hz)
    - **Fase**: spostamento relativo della forma d'onda
    - **Larghezza di banda**: intervallo di frequenze utilizzabili
    - **Attenuazione**: perdita di potenza del segnale
    - **Distorsione**: alterazione della forma del segnale
    - **Rumore**: interferenze elettromagnetiche

2.4 Modulazioni e Multiplexing

- **Modulazioni**:
    - **Analogiche**:
        - AM (Amplitude Modulation): varia l'ampiezza
        - FM (Frequency Modulation): varia la frequenza
        - PM (Phase Modulation): varia la fase
    - **Digitali**:
        - ASK (Amplitude Shift Keying): varia l'ampiezza

        - FSK (Frequency Shift Keying): varia la frequenza
        - PSK (Phase Shift Keying): varia la fase
        - QAM (Quadrature Amplitude Modulation): combina ampiezza e fase
- **Multiplexing**:
    - **FDM** (Frequency Division Multiplexing): suddivisione in frequenza
    - **TDM** (Time Division Multiplexing): suddivisione temporale
    - **WDM** (Wavelength Division Multiplexing): per fibra ottica, multiple lunghezze d'onda
    - **CDM** (Code Division Multiplexing): codici unici per ogni canale
    - **SDM** (Space Division Multiplexing): canali fisicamente separati

2.5 Continuazione Tipologie di Cavo e Conclusione Multiplexing

- **Doppini**:
    - Coppie di fili intrecciati per ridurre le interferenze
    - Categorie in base alla velocità supportata
    - RJ-45: connettore standard per doppini
    - Standard di cablaggio: T568A e T568B
- **Fibra ottica**:
    - Principio: riflessione totale interna
    - Componenti: core (nucleo), cladding (mantello), buffer (rivestimento)
    - Connettori: SC, LC, ST, FC
    - FTTH (Fiber To The Home): fibra fino all'abitazione
    - Vantaggi: alta larghezza di banda, bassa attenuazione, sicurezza

2.6 Codici di Correzione Errore

- **Codifiche di linea**:
    - **NRZ** (Non-Return-to-Zero):
        - Usa due livelli di tensione
        - Problemi con lunghe sequenze dello stesso bit
    - **RZ** (Return-to-Zero):
        - Ritorna a zero dopo ogni bit
        - Auto-sincronizzazione migliore
        - Richiede maggiore larghezza di banda
    - **Manchester**:
        - Transizione a metà bit: alto-basso per 0, basso-alto per 1
        - Auto-sincronizzazione
        - Utilizzato in Ethernet 10Base-T
- **CRC** (Cyclic Redundancy Check):

    - Algoritmo di rilevamento errori
    - Polinomio generatore
    - Divisione in modulo-2
    - Standard: CRC-16, CRC-32
    - Efficace per burst error

2.7 Tipi di Trasmissione

- **Simplex**:
    - Comunicazione unidirezionale
    - Es: radio, televisione
- **Half-duplex**:
    - Comunicazione bidirezionale alternata
    - Un dispositivo alla volta può trasmettere
    - Es: walkie-talkie
- **Full-duplex**:
    - Comunicazione bidirezionale simultanea
    - Canali separati per trasmissione e ricezione
    - Es: telefono, Ethernet moderno

2.8 Dispositivi di Rete

- **Hub**:
    - Dispositivo di livello 1 (fisico)
    - Ripete il segnale su tutte le porte
    - Crea un unico dominio di collisione
    - Obsoleto, sostituito da switch
- **Switch**:
    - Dispositivo di livello 2 (data link)
    - Inoltra frame basandosi su indirizzi MAC
    - Crea domini di collisione separati
    - Mantiene tabella MAC-porta
- **Router**:
    - Dispositivo di livello 3 (rete)
    - Inoltra pacchetti tra reti diverse
    - Determina il percorso migliore (routing)
    - Separa domini di broadcast

3. LIVELLO DI RETE IP

3.1 Topologie e Introduzione al Livello IP

- **Topologie fisiche**:
  - A stella, ad anello, a bus, a maglia, ad albero
- **Topologie logiche**:
  - Broadcast: tutti i nodi ricevono tutti i messaggi
  - Token passing: trasmissione gestita da un token
- **IP** (Internet Protocol):
  - Protocollo di livello 3
  - Connectionless: ogni pacchetto indipendente
  - Best-effort: nessuna garanzia di consegna
  - Versioni: IPv4 (32 bit), IPv6 (128 bit)
- **Caratteristiche del frame IP**:
  - Header: informazioni di controllo
  - Payload: dati utente
  - Non include informazioni di controllo fisico (preambolo, FCS)

3.2 Classi di Indirizzi IP e Subnetting

- **Indirizzo IPv4**: 32 bit (4 byte), notazione decimale puntata (es: 192.168.1.1)
- **Classi di indirizzi**:
  - **Classe A**: 0.0.0.0 - 127.255.255.255 (8 bit rete, 24 bit host)
  - **Classe B**: 128.0.0.0 - 191.255.255.255 (16 bit rete, 16 bit host)
  - **Classe C**: 192.0.0.0 - 223.255.255.255 (24 bit rete, 8 bit host)
  - **Classe D**: 224.0.0.0 - 239.255.255.255 (multicast)
  - **Classe E**: 240.0.0.0 - 255.255.255.255 (riservata/sperimentale)
- **Indirizzi speciali**:
  - Rete: tutti i bit host a 0 (es: 192.168.1.0)
  - Broadcast: tutti i bit host a 1 (es: 192.168.1.255)
  - Loopback: 127.0.0.1
  - Private:
    - 10.0.0.0/8 (Classe A)
    - 172.16.0.0/12 (Classe B)
    - 192.168.0.0/16 (Classe C)
- **Subnetting**:
  - Divisione di una rete in sottoreti
  - Utilizzo di una subnet mask
  - Notazione CIDR (Classless Inter-Domain Routing): /n (n = numero di bit di rete)
  - Esempio: 192.168.1.0/24 (subnet mask 255.255.255.0)

3.3 Esempi Pratici di Subnetting

- **Calcolo subnet**:

1. Identificare la classe e la subnet mask
2. Determinare il numero di sottoreti necessarie
3. Calcolare la subnet mask estesa
4. Calcolare indirizzi di rete, host validi e broadcast

- **Esempio**:
  - Rete: 192.168.1.0/24 (Classe C)
  - Necessità: 4 subnet
  - Bit necessari: 2 ($2^2 = 4$)
  - Nuova subnet mask: 255.255.255.192 (/26)
  - Sottoreti:
    - 192.168.1.0/26 (host: 192.168.1.1 - 192.168.1.62, broadcast: 192.168.1.63)
    - 192.168.1.64/26 (host: 192.168.1.65 - 192.168.1.126, broadcast: 192.168.1.127)
    - 192.168.1.128/26 (host: 192.168.1.129 - 192.168.1.190, broadcast: 192.168.1.191)
    - 192.168.1.192/26 (host: 192.168.1.193 - 192.168.1.254, broadcast: 192.168.1.255)

3.4 Routing e Tipi

- **Concetto di routing**:
  - Processo di determinazione del percorso migliore per i pacchetti
  - Basato su tabelle di routing
  - Metriche: hop count, latenza, larghezza di banda, costo
- **Tipi di routing**:
  - **Routing statico**:
    - Configurato manualmente dall'amministratore
    - Non si adatta ai cambiamenti topologici
    - Basso overhead, alta sicurezza
    - Adatto a reti piccole e stabili
  - **Routing dinamico**:
    - Protocolli automatici che aggiornano le tabelle
    - Si adatta ai cambiamenti topologici
    - Maggiore overhead, ma più flessibile
    - Tipi:
      - Distance vector: RIP, EIGRP
      - Link state: OSPF, IS-IS
      - Path vector: BGP
- **Esempi di codice per routing statico** (in Cisco IOS):

```
Router(config)# ip route 192.168.2.0 255.255.255.0 192.168.1.2
```

3.5 Tipi di Indirizzamento Avanzati

- **VLSM** (Variable Length Subnet Mask):
  - Permette subnet di dimensioni diverse
  - Assegna subnet in base alle reali necessità
  - Riduce lo spreco di indirizzi
  - Esempio:
    - Rete: 192.168.0.0/24
    - Subnet 1 (100 host): 192.168.0.0/25 (126 host utilizzabili)
    - Subnet 2 (50 host): 192.168.0.128/26 (62 host utilizzabili)
    - Subnet 3 (20 host): 192.168.0.192/27 (30 host utilizzabili)
- **CIDR** (Classless Inter-Domain Routing):
  - Supera i limiti delle classi
  - Aggregazione di rotte (route summarization)
  - Notazione /n per la subnet mask
  - Esempio:
    - 192.168.0.0/23 include 192.168.0.0/24 e 192.168.1.0/24
    - 172.16.0.0/12 include tutte le reti da 172.16.0.0 a 172.31.255.0

3.6 Algoritmi di Routing

- **Bellman-Ford** (Distance Vector):
  - Ogni router condivide la propria tabella con i vicini
  - Calcola il percorso più breve basandosi sulla distanza
  - Problemi: count-to-infinity, convergenza lenta
  - Utilizzato in RIP
- **Dijkstra** (Link State):
  - Ogni router costruisce una mappa completa della rete
  - Calcola il percorso più breve basandosi sul costo
  - Vantaggi: convergenza rapida, no count-to-infinity
  - Utilizzato in OSPF

3.7 Traffic Shaping

- **Leaky Bucket**:
  - Regola il flusso come un secchio che perde
  - Rata di uscita costante
  - Traffico in eccesso viene scartato o accodato

    - Riduce burst di traffico
- **Token Bucket**:
    - Genera token a velocità costante
    - Un pacchetto può essere trasmesso solo con un token
    - Consente burst controllati
    - Più flessibile del leaky bucket
- **Choke Packet**:
    - Il router congestionato invia pacchetti di "strozzamento"
    - Le sorgenti riducono la velocità di trasmissione
    - Permette una risposta rapida alla congestione

3.8 Problemi MAC

- **Stazione nascosta** (Hidden Terminal):
    - Due stazioni non si "sentono" ma interferiscono su un terzo nodo
    - Problema tipico delle reti wireless
    - Soluzione: RTS/CTS (Request to Send/Clear to Send)
- **Stazione esposta** (Exposed Terminal):
    - Una stazione si astiene dal trasmettere perché "sente" un'altra trasmissione
    - La trasmissione non interferirebbe con il ricevitore reale
    - Causa inefficienza nell'uso del canale

3.9 Conclusione Routing e Sicurezza

- **CNLS** (Connectionless Network Service):
    - Ogni pacchetto indipendente
    - Nessuna connessione preliminare
    - Usato in IP
- **CONS** (Connection-Oriented Network Service):
    - Stabilisce una connessione prima della trasmissione
    - Mantiene lo stato della connessione
    - Usato in X.25, Frame Relay
- **Accenni a sicurezza e crittografia**:
    - Minacce: intercettazione, modifica, denial of service
    - Meccanismi di difesa: autenticazione, crittografia, firewall
    - Principi crittografici: confidenzialità, integrità, autenticità

3.10 Algoritmi di Contesa

- **CSMA** (Carrier Sense Multiple Access):
    - Ascolta prima di trasmettere

    - Varianti:
        - 1-persistente: trasmette immediatamente se canale libero
        - Non-persistente: attende un tempo casuale e riprova
        - p-persistente: trasmette con probabilità p se canale libero
- **CSMA/CD** (CSMA with Collision Detection):
    - Rileva le collisioni durante la trasmissione
    - In caso di collisione: abort, jam signal, backoff
    - Utilizzato in Ethernet tradizionale

4. PROTOCOLLI E APPLICAZIONI

4.1 Tecniche di Accesso Multiplo al Canale

- **TDMA** (Time Division Multiple Access):
    - Suddivide il tempo in slot
    - Ogni utente ha slot dedicati
    - Utilizzato in GSM
- **FDMA** (Frequency Division Multiple Access):
    - Suddivide lo spettro in canali
    - Ogni utente ha frequenze dedicate
    - Utilizzato in radio AM/FM
- **CDMA** (Code Division Multiple Access):
    - Utenti condividono frequenza e tempo
    - Differenziati da codici unici
    - Maggiore capacità e sicurezza
    - Utilizzato in 3G

4.2 Problemi dell'Accesso Multiplo

- **Collisioni**:
    - Due o più stazioni trasmettono contemporaneamente
    - Segnali si corrompono a vicenda
    - Necessità di meccanismi di rilevamento e risoluzione
- **Fairness** (equità):
    - Garantire a tutte le stazioni opportunità equivalenti
    - Evitare starvation (fame) di alcuni nodi
- **Overhead**:
    - Costo di gestione del protocollo
    - Bilanciamento tra efficienza e robustezza

4.3 ALOHA e Varianti

- **ALOHA puro**:
  - Trasmissione immediata quando ci sono dati
  - In caso di collisione: ritrasmissione dopo tempo casuale
  - Throughput massimo: 18.4%
- **Slotted ALOHA**:
  - Tempo diviso in slot discreti
  - Trasmissione solo all'inizio di uno slot
  - Riduce probabilità di collisione
  - Throughput massimo: 36.8%

4.4 ARP e ICMP

- **ARP** (Address Resolution Protocol):
  - Mappa indirizzi IP in indirizzi MAC
  - Broadcast: "Chi ha l'IP x.x.x.x?"
  - Risposta unicast: "Io ho x.x.x.x, il mio MAC è xx:xx:xx:xx:xx:xx"
  - Cache ARP per ridurre traffico broadcast
  - Vulnerabilità: ARP poisoning/spoofing
- **ICMP** (Internet Control Message Protocol):
  - Protocollo di controllo per IP
  - Funzioni:
    - Echo request/reply (ping)
    - Destination unreachable
    - Time exceeded
    - Redirect
    - Router advertisement/solicitation

4.5 Esercizi di Subnetting

- **Processo di subnetting**:
  1. Identificare classe e subnet mask iniziale
  2. Determinare requisiti (n° subnet, host per subnet)
  3. Calcolare bit necessari per subnet e host
  4. Determinare nuova subnet mask
  5. Calcolare indirizzi di rete, range di host, broadcast
- **Esempio**:
  - Data rete 172.16.0.0/16, creare 14 subnet
  - Bit necessari: 4 ($2^4=16 > 14$)
  - Nuova subnet mask: 255.255.240.0 (/20)
  - Prima subnet: 172.16.0.0/20 (host: 172.16.0.1 - 172.16.15.254)
  - Seconda subnet: 172.16.16.0/20 (host: 172.16.16.1 - 172.16.31.254)

- E così via...

5. LIVELLO DI TRASPORTO

5.1 Ripasso Indirizzi IP e Routing

- **Struttura indirizzo IP**:
  - 32 bit (IPv4) divisi in porzione rete e host
  - Notazione decimale puntata (es: 192.168.1.1)
  - Subnet mask per identificare la porzione di rete
- **Funzioni livello rete**:
  - Indirizzamento logico
  - Routing
  - Frammentazione e riassemblaggio
  - Controllo congestione
- **Funzioni livello data link**:
  - Framing
  - Controllo errori
  - Controllo flusso
  - Accesso al mezzo

5.2 Introduzione a TCP e UDP

- **TCP** (Transmission Control Protocol):
  - Orientato alla connessione
  - Affidabile: ordinamento, rilevamento errori, ritrasmissione
  - Controllo di flusso e congestione
  - Comunicazione stream-based
  - Applicazioni: web (HTTP), email (SMTP), file transfer (FTP)
- **UDP** (User Datagram Protocol):
  - Non orientato alla connessione
  - Non affidabile: no garanzie di consegna o ordine
  - Nessun controllo di flusso o congestione
  - Comunicazione message-based
  - Basso overhead, alta velocità
  - Applicazioni: DNS, streaming, VoIP, online gaming

5.3 Quality of Service (QoS)

- **Definizione**: capacità di fornire diversi livelli di servizio a diversi tipi di traffico
- **Parametri QoS**:
  - **Bandwidth** (larghezza di banda): quantità di dati trasmissibili per unità di tempo

    - **Delay** (ritardo): tempo necessario ai pacchetti per attraversare la rete
    - **Jitter**: variazione del ritardo
    - **Packet loss** (perdita di pacchetti): percentuale di pacchetti persi
    - **Throughput**: quantità di dati effettivamente trasmessi per unità di tempo
- **Meccanismi QoS**:
    - **Classificazione traffico**: identificazione e categorizzazione
    - **Marking**: etichettatura pacchetti con priorità
    - **Policing/Shaping**: controllo velocità traffico
    - **Queuing**: gestione code in base a priorità
    - **Congestion avoidance**: prevenzione congestione

5.4 Struttura Pacchetti TCP e UDP

- **Header TCP** (20-60 byte):
    - Source Port (16 bit)
    - Destination Port (16 bit)
    - Sequence Number (32 bit)
    - Acknowledgment Number (32 bit)
    - Data Offset (4 bit)
    - Reserved (6 bit)
    - Control Flags (6 bit): URG, ACK, PSH, RST, SYN, FIN
    - Window Size (16 bit)
    - Checksum (16 bit)
    - Urgent Pointer (16 bit)
    - Options (variabile)
- **Header UDP** (8 byte):
    - Source Port (16 bit)
    - Destination Port (16 bit)
    - Length (16 bit)
    - Checksum (16 bit)

5.5 Meccanismi di Trasmissione TCP

- **Three-way handshake** (apertura connessione):
    1. Client → Server: SYN
    2. Server → Client: SYN+ACK
    3. Client → Server: ACK
- **Four-way handshake** (chiusura connessione):
    1. Client → Server: FIN
    2. Server → Client: ACK
    3. Server → Client: FIN

4. Client → Server: ACK

- **Parametri di connessione**:
  - **RTT** (Round Trip Time): tempo di andata e ritorno
  - **RTO** (Retransmission Timeout): tempo prima di ritrasmettere
  - **MSS** (Maximum Segment Size): dimensione massima segmento
  - **Window Size**: numero di byte che possono essere inviati senza ACK
- **Fairness**: equa distribuzione della banda tra flussi concorrenti

5.6 Problemi TCP

- **Slow Start**:
  - All'inizio la finestra di congestione è piccola
  - Aumenta esponenzialmente fino alla soglia
  - Poi aumenta linearmente (congestion avoidance)
- **Slow Start**:
  - All'inizio la finestra di congestione è piccola
  - Aumenta esponenzialmente fino alla soglia
  - Poi aumenta linearmente (congestion avoidance)
- **Fast Retransmit**:
  - Non attende il timeout per ritrasmettere
  - Se riceve 3 ACK duplicati, ritrasmette immediatamente
  - Migliora significativamente le prestazioni

5.7 Tecniche di Controllo di Flusso

- **Stop-and-wait**:
  - Sender invia un pacchetto e attende ACK
  - Semplice ma inefficiente
  - Utilizzo basso della banda
- **Sliding Window**:
  - Permette l'invio di più pacchetti prima di ricevere ACK
  - Dimensione finestra determina quanti pacchetti possono essere in transito
  - Varianti:
    - Go-Back-N: in caso di errore ritrasmette tutti i pacchetti da N in poi
    - Selective Repeat: ritrasmette solo i pacchetti persi

5.8 Altri Protocolli di Livello Trasporto

- **DHCP** (Dynamic Host Configuration Protocol):
  - Assegna dinamicamente indirizzi IP
  - Processo: Discover → Offer → Request → Acknowledge

    - Fornisce anche subnet mask, gateway, DNS
    - Usa porte UDP 67/68
- **ARP** (Address Resolution Protocol):
    - Mappa indirizzi IP in indirizzi MAC
    - Essenziale per la comunicazione in LAN
    - Cache ARP per memorizzare mappature recenti
    - Vulnerabilità: ARP spoofing

6. SICUREZZA NELLE RETI

6.1 Introduzione alla Crittografia

- **Definizione**: trasformazione di dati per renderli incomprensibili senza apposita chiave
- **Obiettivi**:
    - **Confidenzialità**: protezione da accessi non autorizzati
    - **Integrità**: garanzia che i dati non siano alterati
    - **Autenticità**: verifica dell'identità del mittente
    - **Non ripudio**: impossibilità di negare azioni compiute
- **Tipi di crittografia**:
    - **Simmetrica**: stessa chiave per cifrare e decifrare
    - **Asimmetrica**: coppia di chiavi correlate (pubblica e privata)
    - **Ibrida**: combina entrambe le tecniche

6.2 Crittografia Simmetrica

- **Caratteristiche**:
    - Una sola chiave per cifrare e decifrare
    - Veloce ed efficiente
    - Problema della distribuzione sicura delle chiavi
- **Algoritmi**:
    - **DES** (Data Encryption Standard):
        - Sviluppato negli anni '70
        - Blocchi di 64 bit, chiave di 56 bit
        - Considerato insicuro oggi (brute force possibile)
    - **3DES** (Triple DES):
        - Applica DES tre volte con chiavi diverse
        - Blocchi di 64 bit, chiave effettiva 112/168 bit
        - Più sicuro ma più lento di DES
    - **AES** (Advanced Encryption Standard):
        - Standard attuale
        - Blocchi di 128 bit, chiavi di 128/192/256 bit

        - Sicuro e relativamente veloce
    - **Altri**: Blowfish, Twofish, RC4, ChaCha20

6.3 Crittografia Asimmetrica

- **Caratteristiche**:
    - Coppia di chiavi: pubblica (cifratura) e privata (decifratura)
    - Più lenta della simmetrica
    - Risolve il problema della distribuzione delle chiavi
- **Algoritmi**:
    - **RSA** (Rivest-Shamir-Adleman):
        - Basato sulla difficoltà di fattorizzare numeri grandi
        - Ampiamente utilizzato
        - Chiavi tipicamente 2048-4096 bit
        - Esempio di funzionamento:
            1. Generazione chiavi: scelta di p, q primi; n = p×q; φ(n) = (p-1)(q-1)
            2. Scelta e (coprimo con φ(n)); calcolo d (inverso moltiplicativo di e modulo φ(n))
            3. Chiave pubblica: (n, e); chiave privata: (n, d)
            4. Cifratura: c = m^e mod n
            5. Decifratura: m = c^d mod n
    - **Altri**: ElGamal, ECC (Elliptic Curve Cryptography), DSA

6.4 Trasposizione e Firma Digitale

- **Cifrari a trasposizione**:
    - **Cesare**: sostituzione con shift fisso dell'alfabeto
        - Es: shift 3: A→D, B→E, C→F, ecc.
        - Facilmente decifrabile (26 possibilità)
    - **Vigenère**: sostituzione polialfabetica con chiave
        - Usa una tabella e una parola chiave
        - Più sicuro di Cesare ma comunque vulnerabile
- **Firma digitale**:
    - Garantisce autenticità e non ripudio
    - Processo:
        1. Hash del documento
        2. Cifratura dell'hash con chiave privata del mittente
        3. Verifica con chiave pubblica del mittente
    - Applicazioni: PEC, documenti XML, certificati digitali

6.5 Funzioni di Hash

- **Caratteristiche**:
  - Trasformano input di lunghezza arbitraria in output di lunghezza fissa
  - Idealmente: piccoli cambi nell'input creano grandi cambi nell'output
  - Unidirezionali: impossibile risalire all'input dall'output
  - Resistenti alle collisioni: difficile trovare due input con stesso output
- **Algoritmi**:
  - **MD5** (Message Digest 5):
    - Output di 128 bit
    - Considerato insicuro (collisioni trovate)
    - Ancora usato per checksum (non per sicurezza)
  - **SHA** (Secure Hash Algorithm):
    - SHA-1: output di 160 bit (vulnerabile)
    - SHA-2: include SHA-256, SHA-384, SHA-512
    - SHA-3: nuovo standard, approccio diverso
    - Ampiamente utilizzati per sicurezza

6.6 HTTPS (Livello 7)

- **HTTP Secure**:
  - HTTP su connessione crittografata (SSL/TLS)
  - Garantisce confidenzialità e integrità
  - Autenticazione del server mediante certificati
- **Funzionamento**:
  1. Client richiede connessione sicura
  2. Server invia certificato con chiave pubblica
  3. Client verifica certificato con CA (Certificate Authority)
  4. Client genera chiave di sessione e la cifra con chiave pubblica del server
  5. Server decifra la chiave di sessione
  6. Comunicazione crittografata con chiave di sessione (simmetrica)

6.7 Attacchi di Sicurezza

- **Man in the Middle**:
  - L'attaccante si interpone tra due comunicanti
  - Può intercettare, modificare, iniettare messaggi
  - Contromisure: autenticazione forte, crittografia, certificati
- **DoS** (Denial of Service):
  - Sovraccarico di un servizio per renderlo indisponibile
  - Tecniche: SYN flood, ICMP flood, UDP flood, amplification
  - Contromisure: filtraggio, rate limiting, ridondanza
- **DDoS** (Distributed Denial of Service):

    - DoS da molteplici sorgenti (botnet)
    - Più difficile da contrastare
    - Contromisure: CDN, servizi anti-DDoS, traffic scrubbing

6.8 Bluetooth

- **Caratteristiche generali**:
    - Tecnologia wireless a corto raggio (PAN)
    - Frequenza: 2.4 GHz ISM
    - Versioni: da 1.0 a 5.3, con miglioramenti in velocità, range, consumo
    - Sicurezza: pairing, crittografia
- **Architettura**:
    - **Beacon**: segnali periodici per sincronizzazione e discovery
    - **Piconet**: rete di dispositivi Bluetooth (1 master + fino a 7 slave)
    - **Scatternet**: interconnessione di più piconet
- **Profili**: specificano come usare Bluetooth per specifiche applicazioni (A2DP, HFP, OBEX, ecc.)

6.9 VPN e Firewall

- **VPN** (Virtual Private Network):
    - Estende rete privata su rete pubblica
    - Sicurezza: crittografia, autenticazione, tunneling
    - Tipi:
        - Site-to-site: collega intere reti
        - Remote access: collega utenti singoli a rete
    - Protocolli: IPsec, SSL/TLS, OpenVPN, WireGuard
- **Tunneling**:
    - Incapsulamento di un protocollo in un altro
    - Permette trasporto attraverso reti con restrizioni
    - Nasconde dettagli del traffico interno
- **Firewall**:
    - Sistema di sicurezza che monitora e filtra traffico di rete
    - Tipi:
        - **Packet filter**: filtro basato su header
        - **Stateful inspection**: tiene traccia dello stato delle connessioni
        - **Application layer**: analizza il traffico a livello applicativo
        - **Next-gen**: include IDS/IPS, antivirus, DLP
    - Posizionamento: perimetrale, interno, host-based

6.10 Politiche di Accesso e Sicurezza

- **DAC** (Discretionary Access Control):
  - Il proprietario della risorsa decide chi può accedervi
  - Flessibile ma potenzialmente meno sicuro
  - Esempio: permessi file in sistemi operativi desktop
- **MAC** (Mandatory Access Control):
  - Il sistema impone regole di accesso basate su policy
  - Più rigido ma più sicuro
  - Esempio: SELinux, sistemi militari
- **HTTPS e SSL/TLS**:
  - SSL (Secure Sockets Layer): predecessore di TLS
  - TLS (Transport Layer Security): versioni 1.0-1.3
  - Handshake: scambio di chiavi e parametri
  - Record protocol: trasferimento dati crittografati
  - Certificati X.509 per autenticazione
- **IPsec** (IP Security):
  - Suite di protocolli per sicurezza a livello IP
  - Componenti:
    - AH (Authentication Header): integrità e autenticazione
    - ESP (Encapsulating Security Payload): confidenzialità, integrità, autenticazione
    - IKE (Internet Key Exchange): gestione chiavi

7. LIVELLO APPLICATIVO

7.1 Sicurezza Wireless

- **WEP** (Wired Equivalent Privacy):
  - Primo standard di sicurezza 802.11
  - Cifrario RC4 con chiavi statiche
  - Gravemente vulnerabile, non utilizzare
  - Problemi: vettori di inizializzazione deboli, gestione chiavi, integrità
- **WPA** (Wi-Fi Protected Access):
  - Sostituto temporaneo di WEP
  - TKIP (Temporal Key Integrity Protocol)
  - Più sicuro di WEP ma comunque vulnerabile
  - Autenticazione: PSK o 802.1X/EAP
- **WPA2**:
  - Standard dal 2004
  - Cifrario AES-CCMP
  - Sicurezza significativamente migliore

    - Vulnerabilità: KRACK (Key Reinstallation Attack)
- **WPA3**:
    - Standard più recente (2018)
    - Miglioramenti: Simultaneous Authentication of Equals, forward secrecy
    - Protezione contro attacchi offline, improved handshake
    - Modalità personale (SAE) e enterprise (802.1X)

7.2 Protocolli di Livello Applicativo

- **DNS** (Domain Name System):
    - Risolve nomi di dominio in indirizzi IP
    - Struttura gerarchica (root, TLD, domain, subdomain)
    - Record: A, AAAA, MX, CNAME, TXT, NS, ecc.
    - Porte: UDP/TCP 53
    - Vulnerabilità: cache poisoning, DDoS, tunneling
- **HTTPS**:
    - HTTP su TLS/SSL
    - Porte: TCP 443
    - Certificati: X.509, validati da CA
    - HSTS: forza connessioni HTTPS
    - HTTP/2, HTTP/3: miglioramenti prestazioni

7.3 Architetture di Rete e Problemi

- **Client/Server**:
    - Server centralizzati forniscono servizi
    - Client richiedono servizi
    - Vantaggi: gestione centralizzata, controllo
    - Svantaggi: single point of failure, scalabilità
- **Peer-to-Peer (P2P)**:
    - Nodi fungono sia da client che da server
    - Decentralizzato, distribuito
    - Vantaggi: resilienza, scalabilità
    - Svantaggi: gestione complessa, sicurezza
- **Microservizi**:
    - Applicazioni come suite di servizi indipendenti
    - Ogni servizio è un processo distinto
    - Comunicazione via API (spesso REST)
    - Vantaggi: scalabilità, resilienza, sviluppo agile
    - Svantaggi: complessità, overhead comunicazione

7.4 Protocolli di Posta Elettronica

- **SMTP** (Simple Mail Transfer Protocol):
  - Per invio email
  - Porta: TCP 25 (non sicura), 587 (TLS), 465 (SSL)
  - Comandi: HELO/EHLO, MAIL FROM, RCPT TO, DATA, QUIT
  - Estensioni: ESMTP (autenticazione, crittografia)
- **POP3** (Post Office Protocol v3):
  - Per scaricamento email
  - Porta: TCP 110 (non sicura), 995 (SSL)
  - Semplice, scarica e-mail sul client
  - Comandi: USER, PASS, LIST, RETR, DELE
  - Limiti: non sincronizzazione multi-dispositivo
- **IMAP** (Internet Message Access Protocol):
  - Per gestione email sul server
  - Porta: TCP 143 (non sicura), 993 (SSL)
  - Mantiene email sul server, sincronizzazione
  - Supporta cartelle, flag, ricerca
  - Vantaggi: multi-dispositivo, accesso parziale

7.5 Connessione Remota

- **SSH** (Secure Shell):
  - Protocollo per connessione sicura
  - Porta: TCP 22
  - Autenticazione: password, chiavi pubbliche/private
  - Tunneling: port forwarding, SOCKS proxy
  - Utilizzi: terminale remoto, SCP, SFTP, X11 forwarding
- **Telnet**:
  - Predecessore di SSH, non sicuro
  - Trasmissione in chiaro (incluse credenziali)
  - Porta: TCP 23
  - Da evitare, preferire SSH
- **API e Microservizi**:
  - **API** (Application Programming Interface):
    - Interfaccia per interazione tra componenti software
    - Tipi: SOAP, REST, GraphQL, gRPC
  - **Microservizi**:
    - Architettura con servizi indipendenti
    - Comunicazione via API

        - Scalabilità individuale dei componenti
        - Container e orchestrazione (Docker, Kubernetes)

7.6 API REST e HTTP

- **REST** (Representational State Transfer):
    - Architettura per sistemi distribuiti
    - Principi:
        - Stateless: ogni richiesta è indipendente
        - Resource-based: URI identificano risorse
        - Rappresentazioni: JSON, XML, ecc.
        - Interfaccia uniforme: metodi HTTP standard
- **HTTP** (Hypertext Transfer Protocol):
    - Protocollo application layer per il web
    - Metodi: GET, POST, PUT, DELETE, PATCH, ecc.
    - Codici stato: 1xx (info), 2xx (successo), 3xx (redirect), 4xx (client error), 5xx (server error)
    - Header: content-type, authorization, cache-control, ecc.
    - HTTP/1.1, HTTP/2, HTTP/3: evoluzione del protocollo

7.7 Scambio File e Peer-to-Peer

- **FTP** (File Transfer Protocol):
    - Protocollo per trasferimento file
    - Porte: TCP 21 (controllo), TCP 20 (dati) o porte dinamiche
    - Modalità: attiva e passiva
    - Comandi: USER, PASS, LIST, CWD, STOR, RETR
    - Non sicuro: credenziali in chiaro
- **FTPS** (FTP Secure):
    - FTP su SSL/TLS
    - Porte: varie, spesso TCP 990
    - Sicurezza significativamente migliore
- **Protocolli P2P**:
    - **Gnutella**:
        - Rete P2P completamente decentralizzata
        - Query flooding per ricerca
        - Scalabilità limitata
    - **BitTorrent**:
        - Protocollo per condivisione file
        - File divisi in pezzi (chunks)
        - Tracker o DHT per coordinamento

        - Algoritmi: rarest first, tit-for-tat
        - Swarm: seeders (completi) e leechers (parziali)

7.8 Concetto di File Torrent

- **File .torrent**:
    - Metafile con informazioni per download
    - Contiene:
        - Announce: URL tracker
        - Info hash: identificatore univoco
        - Piece length: dimensione dei pezzi
        - Pieces: hash SHA-1 di ogni pezzo
        - Nome, dimensione, struttura file
- **Processo BitTorrent**:
    1. Client scarica file .torrent
    2. Contatta tracker o DHT
    3. Riceve lista di peer
    4. Connessione ai peer e scambio pezzi
    5. Diventa seeder dopo download completo
- **Magnet link**: alternativa al file .torrent, contiene info hash e tracker

8. SICUREZZA DELLE RETI

8.1 Sicurezza Software

- **Vulnerabilità software**:
    - Buffer overflow
    - Injection (SQL, XSS, CSRF)
    - Errori logici, race condition
    - Configurazioni insicure
    - Dipendenze vulnerabili
- **Pratiche sicure**:
    - Secure coding
    - Code review
    - Testing (SAST, DAST, IAST)
    - Patch management
    - Principle of least privilege

8.2 Tipi di File Dannosi

- **Virus**:

    - Si replica inserendo codice in altri file
    - Richiede azione utente per diffondersi
    - Tipi: file, boot, macro, polimorfici
- **Worm**:
    - Si diffonde autonomamente via rete
    - Non richiede intervento umano
    - Consumo risorse, backdoor
- **Trojan**:
    - Appare legittimo ma contiene malware
    - Non si auto-replica
    - Tipi: backdoor, downloader, banking, RAT
- **Ransomware**:
    - Cifra dati e chiede riscatto
    - Propagazione via phishing, vulnerabilità
    - Impatto severo su organizzazioni
- **Spyware**:
    - Raccoglie informazioni senza consenso
    - Keylogger, screen capture, data exfiltration
    - Privacy breach

8.3 Misure di Prevenzione

- **Hardware**:
    - Firewall hardware
    - IDS/IPS fisici
    - Dispositivi di autenticazione (token, smartcard)
    - Airgap per sistemi critici
- **Software**:
    - Antivirus/antimalware
    - Firewall software
    - Patch management
    - Whitelisting applicazioni
- **Sociali**:
    - Formazione utenti
    - Security awareness
    - Policy e procedure
    - Social engineering testing

9. AUDITING E COMPLIANCE

9.1 Tipi di Audit

- **Audit interno**:
  - Condotto da personale dell'organizzazione
  - Scopo: miglioramento continuo
  - Generalmente meno formale
  - Preparazione per audit esterni
- **Audit esterno**:
  - Condotto da terze parti indipendenti
  - Maggiore credibilità e imparzialità
  - Può essere richiesto per conformità normativa
  - Risulta in report formale con findings
- **Certificazione**:
  - Verifica conformità a standard specifici
  - Rilascio di certificato ufficiale
  - Esempi: ISO 27001, PCI DSS, SOC 2
  - Periodicità: iniziale e mantenimento

9.2 Penetration Testing e Vulnerability Assessment

- **Vulnerability Assessment**:
  - Identificazione sistematica vulnerabilità
  - Approccio ampio ma meno profondo
  - Tool automatizzati + analisi manuale
  - Output: lista vulnerabilità con severità e rimedio
- **Penetration Testing**:
  - Simulazione attacchi reali
  - Sfrutta vulnerabilità per dimostrare impatto
  - Tipi: black box, white box, grey box
  - Fasi: reconnaissance, scanning, exploitation, post-exploitation, reporting

9.3 Gestione delle Non Conformità

- **Identificazione**:
  - Audit, controlli, incident
  - Classificazione per gravità
  - Documentazione dettagliata
- **Analisi cause**:
  - Root cause analysis
  - Tecniche: 5 Why, fishbone, fault tree
  - Identificazione cause sistemiche

- **Azioni correttive**:
  - Piano di remediation
  - Responsabilità assegnate
  - Timeline definite
  - Verifica efficacia
- **Prevenzione**:
  - Misure per evitare ricorrenza
  - Miglioramento processi
  - Formazione
  - Aggiornamento controlli

9.4 Security Operation Center (SOC)

- **Struttura**:
  - Team dedicato alla cybersecurity
  - Monitoring 24/7
  - Livelli: L1 (triage), L2 (analisi), L3 (risposta avanzata)
  - Integrazione con CERT/CSIRT
- **Funzionamento**:
  - Monitoraggio continuo
  - Detection eventi sospetti
  - Analisi e correlazione
  - Risposta agli incidenti
  - Intelligence e threat hunting
- **Tecnologie**:
  - SIEM (Security Information and Event Management)
  - EDR (Endpoint Detection and Response)
  - NDR (Network Detection and Response)
  - SOAR (Security Orchestration, Automation and Response)
  - TIP (Threat Intelligence Platform)

10. EVOLUZIONE DEI SISTEMI DI AUTENTICAZIONE

10.1 Fattori di Autenticazione

- **Conoscenza** (something you know):
  - Password, PIN, pattern
  - Domande di sicurezza
  - Frasi segrete
  - Vantaggi: facili da implementare
  - Svantaggi: vulnerabili a phishing, social engineering

- **Possesso** (something you have):
  - Token fisici, smartcard
  - Mobile device (OTP via app o SMS)
  - Chiavi di sicurezza (FIDO2, YubiKey)
  - Vantaggi: difficili da duplicare
  - Svantaggi: possono essere persi o rubati
- **Inerenza** (something you are):
  - Biometria: impronte, volto, iride, retina
  - Comportamentale: digitazione, firma, voce
  - Vantaggi: unici per ogni persona
  - Svantaggi: non modificabili se compromessi, falsi positivi/negativi

10.2 Autenticazione Multi-Fattore

- **MFA** (Multi-Factor Authentication):
  - Combinazione di più fattori diversi
  - Significativamente più sicura
  - Implementazioni:
    - 2FA (due fattori)
    - 3FA (tre fattori)
    - Adattiva (risk-based)
- **Metodologie**:
  - OTP (One-Time Password)
  - App authenticator (TOTP/HOTP)
  - Push notification
  - Biometria + possesso
  - SMS (considerato meno sicuro)
- **Standard**:
  - FIDO2/WebAuthn
  - OATH (Initiative for Open AuTHentication)
  - OAuth 2.0 (per autorizzazione)
  - OpenID Connect (per identità)

10.3 Sistemi Biometrici

- **Tipi**:
  - **Fisici**:
    - Impronte digitali
    - Riconoscimento facciale
    - Scansione iride/retina
    - Geometria mano

        - DNA
    - **Comportamentali**:
        - Dinamica di digitazione
        - Riconoscimento voce
        - Firma grafometrica
        - Analisi andatura
        - Pattern comportamentali
- **Funzionamento**:
    1. Acquisizione
    2. Pre-elaborazione
    3. Estrazione caratteristiche
    4. Confronto con template
    5. Decisione (match/no match)
- **Metriche**:
    - FAR (False Acceptance Rate)
    - FRR (False Rejection Rate)
    - EER (Equal Error Rate)
    - Threshold di decisione

10.4 Single Sign-On e Identity Federation

- **SSO** (Single Sign-On):
    - Autenticazione unica per più servizi
    - L'utente si autentica una volta sola
    - Sessione condivisa tra applicazioni
    - Tipi:
        - Enterprise SSO
        - Web SSO
        - Federated SSO
- **Identity Federation**:
    - Gestione identità distribuita tra organizzazioni
    - Trust relationship tra identity provider
    - L'utente si autentica presso un IdP e accede a più SP
    - Protocolli:
        - SAML
        - OAuth 2.0 / OpenID Connect
        - WS-Federation
- **Vantaggi e rischi**:
    - Pro: usabilità, gestione centralizzata
    - Contro: single point of failure, maggiore superficie d'attacco

11. FIRMA DIGITALE E PKI

11.1 Infrastruttura a Chiave Pubblica

- **PKI** (Public Key Infrastructure):
  - Insieme di hardware, software, politiche e procedure
  - Gestisce creazione, distribuzione, revoca certificati
  - Basata su crittografia asimmetrica
  - Componenti:
    - CA (Certificate Authority)
    - RA (Registration Authority)
    - Repository certificati
    - Sistema di gestione
- **Gerarchia**:
  - Root CA (auto-firmata)
  - Intermediate CA
  - Issuing CA
  - End entity

11.2 Certificati Digitali e CA

- **Certificato digitale**:
  - Documento elettronico che associa chiave pubblica a identità
  - Standard X.509
  - Contiene:
    - Dati titolare
    - Chiave pubblica
    - Periodo validità
    - Dati CA emittente
    - Firma della CA
    - Policy e utilizzi
- **CA** (Certificate Authority):
  - Emette e firma certificati
  - Verifica identità richiedenti
  - Pubblica CRL (Certificate Revocation List)
  - Fornisce OCSP (Online Certificate Status Protocol)
  - Commerciali: DigiCert, Sectigo, GlobalSign
  - Free: Let's Encrypt

11.3 Normativa eIDAS e Standard Italiani

- **eIDAS** (electronic IDentification, Authentication and trust Services):
  - Regolamento UE n. 910/2014
  - Quadro normativo per identità elettronica e servizi fiduciari
  - Riconoscimento transfrontaliero
  - Livelli di garanzia: basso, significativo, elevato
- **Standard italiani**:
  - CAD (Codice dell'Amministrazione Digitale)
  - AgID (Agenzia per l'Italia Digitale)
  - SPID (Sistema Pubblico di Identità Digitale)
  - CIE (Carta d'Identità Elettronica)
  - CNS (Carta Nazionale dei Servizi)

11.4 Applicazioni Pratiche della Firma Digitale

- **Documenti legali**:
  - Contratti
  - Atti notarili
  - Documenti fiscali
  - Fascicolo sanitario
- **e-Government**:
  - Servizi PA online
  - Procedure amministrative
  - Procurement pubblico
- **Business**:
  - Fatturazione elettronica
  - Ordini e contratti
  - Firme multiple e workflow approval
  - Conservazione a norma
- **Tecnologie**:
  - PAdES (PDF)
  - XAdES (XML)
  - CAdES (CMS/PKCS#7)
  - JAdES (JSON)

12. RESPONSIBLE DISCLOSURE E SECURITY RESEARCH

12.1 Principi della Responsible Disclosure

- **Definizione**:
  - Processo etico di segnalazione vulnerabilità

    - Comunicazione privata all'organizzazione interessata
    - Tempo ragionevole per fix prima di disclosure pubblica
    - Bilanciamento tra sicurezza e trasparenza
- **Fasi**:
    1. Scoperta vulnerabilità
    2. Documentazione dettagliata
    3. Contatto responsabile sicurezza
    4. Collaborazione per verifica e fix
    5. Disclosure coordinata
- **Timeframe**:
    - Tipicamente 30-90 giorni
    - Variabile per severità e complessità
    - Possibilità di estensione per vulnerabilità complesse
    - Negoziabile tra researcher e organizzazione

12.2 Bug Bounty Programs

- **Definizione**:
    - Programmi che premiano ricercatori per la scoperta di vulnerabilità
    - Incentivi monetari o riconoscimenti
    - Regole d'ingaggio chiare
    - Piattaforme: HackerOne, Bugcrowd, Intigriti
- **Vantaggi**:
    - Crowdsourcing della security
    - Riduzione costi rispetto a penetration testing tradizionale
    - Diversità di approcci e competenze
    - Miglioramento continuo
- **Componenti**:
    - Scope (in/out of scope)
    - Regole di engagement
    - Scala di ricompense
    - Processo di triage e validazione
    - Gestione disclosure

12.3 Framework Legali per Security Testing

- **Legislazione informatica**:
    - Variabile per giurisdizione
    - Computer Fraud and Abuse Act (USA)
    - Direttiva NIS (UE)
    - Computer Misuse Act (UK)

    - Legge 48/2008 (Italia)
- **Autorizzazioni**:
    - Permesso scritto esplicito
    - Limiti chiari (scope)
    - Non-disclosure agreement
    - Rules of engagement
    - Safe harbor agreements
- **Rischi legali**:
    - Accesso non autorizzato
    - Eccessivo danno o interruzione servizio
    - Data breach
    - Export control per strumenti di sicurezza
    - Responsabilità civile

12.4 Etica Hacker e Responsabilità Professionale

- **Etica hacker**:
    - Principi di comportamento responsabile
    - Non arrecare danno
    - Rispetto privacy e proprietà intellettuale
    - Condivisione conoscenza per miglioramento collettivo
    - Trasparenza e onestà
- **Responsabilità professionale**:
    - Competenza tecnica adeguata
    - Aggiornamento continuo
    - Due diligence
    - Proporzionalità negli interventi
    - Documentazione completa
- **Codici di condotta**:
    - (ISC)² Code of Ethics
    - EC-Council Code of Ethics
    - SANS Institute guidelines
    - OWASP principles

13. CONCETTI UTILI PER L'ESAME DI STATO

13.1 Software Libero e Licenze

- **Software libero vs open source**:
    - Free software: libertà di eseguire, studiare, modificare, ridistribuire
    - Open source: accessibilità codice, collaborazione

    - Differenze filosofiche ma sovrapposizioni pratiche
- **Licenze**:
    - **Copyleft**:
        - GPL (GNU General Public License): obbliga derivati a rimanere open
        - LGPL: permette linking da software proprietario
        - AGPL: copyleft anche per servizi di rete
    - **Permissive**:
        - MIT: minime restrizioni, possibile uso commerciale
        - BSD: simile a MIT, varianti con diverse clausole
        - Apache 2.0: tutela brevetti, trademark
- **License compatibility**:
    - Interazione tra codice con licenze diverse
    - Matrice di compatibilità
    - Obblighi di attribuzione e licenza

13.2 Virtualizzazione e Ambienti Distribuiti

- **Virtualizzazione**:
    - **Hypervisor**:
        - Tipo 1 (bare metal): VMware ESXi, Hyper-V, KVM
        - Tipo 2 (hosted): VirtualBox, VMware Workstation
    - **Tipi di virtualizzazione**:
        - Server (macchine virtuali complete)
        - Desktop (VDI - Virtual Desktop Infrastructure)
        - Applicativa (singole app virtualizzate)
        - Network (SDN - Software Defined Networking)
        - Storage (SAN, NAS virtualizzati)
- **Container**:
    - Isolamento a livello OS senza hypervisor
    - Leggeri, portabili, efficienti
    - Docker, LXC, containerd
    - Immagini e registry
- **Orchestrazione**:
    - Kubernetes
    - Docker Swarm
    - Apache Mesos
    - Automazione deployment, scaling, management
- **Ambienti distribuiti**:
    - Cluster
    - Grid computing

    - Cloud computing (IaaS, PaaS, SaaS)
    - Edge computing
    - Fog computing

13.3 Frontend, Backend e Full-stack

- **Frontend**:
    - Interfaccia utente
    - Presentazione dati
    - Tecnologie:
        - HTML, CSS, JavaScript
        - Framework: React, Angular, Vue
        - Mobile: Swift, Kotlin, Flutter
    - Responsività e UX/UI
- **Backend**:
    - Logica server-side
    - Gestione dati
    - Tecnologie:
        - Linguaggi: Python, Java, PHP, Node.js, C#
        - Framework: Django, Spring, Laravel, Express
        - Database: MySQL, PostgreSQL, MongoDB, Redis
    - API, sicurezza, scalabilità
- **Full-stack**:
    - Competenze su entrambi i fronti
    - Visione d'insieme dell'applicazione
    - DevOps: CI/CD, containerizzazione
    - Architetture: monolitica, microservizi, serverless
- **Comunicazione**:
    - REST API
    - GraphQL
    - WebSocket
    - gRPC
    - Messaging (AMQP, Kafka)

GLOSSARIO TERMINI CHIAVE

- **AES** (Advanced Encryption Standard): algoritmo di crittografia simmetrica
- **API** (Application Programming Interface): interfaccia per interazione tra software
- **ARP** (Address Resolution Protocol): protocollo per mappare IP in MAC

- **BGP** (Border Gateway Protocol): protocollo di routing tra AS
- **CA** (Certificate Authority): ente che emette certificati digitali
- **CIDR** (Classless Inter-Domain Routing): metodo flessibile di assegnazione IP
- **CSRF** (Cross-Site Request Forgery): attacco che sfrutta l'identità di un utente autenticato
- **DDoS** (Distributed Denial of Service): attacco di negazione del servizio distribuito
- **DHCP** (Dynamic Host Configuration Protocol): assegnazione automatica indirizzi IP
- **DNS** (Domain Name System): sistema di risoluzione nomi di dominio
- **FTTH** (Fiber To The Home): fibra ottica fino all'abitazione
- **FTP** (File Transfer Protocol): protocollo per trasferimento file
- **HDLC** (High-level Data Link Control): protocollo di livello data link
- **HTTPS** (HTTP Secure): HTTP su connessione crittografata
- **ICMP** (Internet Control Message Protocol): protocollo di controllo per IP
- **IMAP** (Internet Message Access Protocol): protocollo per accesso email su server
- **IPsec** (IP Security): suite protocolli per sicurezza IP
- **IPv4/IPv6**: versioni del protocollo IP
- **ISO** (International Organization for Standardization): ente di standardizzazione
- **LAN** (Local Area Network): rete locale
- **LLC** (Logical Link Control): sottolivello superiore data link
- **MAC** (Media Access Control): sottolivello inferiore data link
- **MFA** (Multi-Factor Authentication): autenticazione a più fattori
- **NAT** (Network Address Translation): traduzione indirizzi di rete
- **OSPF** (Open Shortest Path First): protocollo di routing link state
- **P2P** (Peer-to-Peer): architettura decentralizzata
- **PKI** (Public Key Infrastructure): infrastruttura a chiave pubblica
- **POP3** (Post Office Protocol v3): protocollo per scaricamento email
- **QoS** (Quality of Service): qualità del servizio
- **REST** (Representational State Transfer): architettura per sistemi distribuiti
- **RIP** (Routing Information Protocol): protocollo di routing distance vector
- **RSA**: algoritmo di crittografia asimmetrica
- **SMTP** (Simple Mail Transfer Protocol): protocollo per invio email
- **SOC** (Security Operation Center): centro operativo sicurezza
- **SSH** (Secure Shell): protocollo per connessione sicura
- **SSL/TLS** (Secure Sockets Layer/Transport Layer Security): protocolli per comunicazione sicura
- **TCP** (Transmission Control Protocol): protocollo di trasporto affidabile
- **UDP** (User Datagram Protocol): protocollo di trasporto non affidabile
- **VLSM** (Variable Length Subnet Mask): subnet mask di lunghezza variabile
- **VPN** (Virtual Private Network): rete privata virtuale
- **WPA/WPA2/WPA3**: standard di sicurezza Wi-Fi

- **XSS** (Cross-Site Scripting): attacco che inietta script dannosi in pagine web