0. Ripasso Iniziale: Reti e Protocolli Base

Classificazioni delle Reti di Calcolatori

Estensione Geografica

- **LAN** (Local Area Network): Rete locale, limitata geograficamente
- **MAN** (Metropolitan Area Network): Rete metropolitana
- **WAN** (Wide Area Network): Rete geografica estesa

Tipologia di Connessione

- **Point-to-Point**: Collegamento diretto tra due nodi
- **Broadcast**: Un mittente, più destinatari
- **Multicast**: Comunicazione verso un gruppo specifico

Modello OSI vs Architettura TCP/IP

Modello OSI (7 livelli)

1. **Fisico**: Trasmissione bit su mezzo fisico
2. **Data Link**: Controllo errori e flusso frame
3. **Rete**: Instradamento pacchetti (IP)
4. **Trasporto**: Affidabilità end-to-end (TCP/UDP)
5. **Sessione**: Gestione sessioni di comunicazione
6. **Presentazione**: Codifica, crittografia, compressione
7. **Applicazione**: Interfaccia con applicazioni utente

Architettura TCP/IP (4 livelli)

1. **Accesso alla Rete**: Corrisponde a Fisico + Data Link OSI
2. **Internet**: Instradamento (IP)
3. **Trasporto**: TCP/UDP
4. **Applicazione**: Servizi di rete (HTTP, SMTP, DNS, etc.)

Indirizzamento IP

Classful Addressing

- **Classe A**: 1-126 (subnet mask /8 - 255.0.0.0)
- **Classe B**: 128-191 (subnet mask /16 - 255.255.0.0)
- **Classe C**: 192-223 (subnet mask /24 - 255.255.255.0)

Classless (CIDR)

- Notazione CIDR: IP/prefisso (es. 192.168.1.0/24)
- Subnet mask variabile per ottimizzazione dello spazio di indirizzamento

Subnetting e Supernetting

- **Subnetting**: Divisione di una rete in sottoreti più piccole
- **Supernetting**: Aggregazione di più reti in una superrete
- **VLSM** (Variable Length Subnet Mask): Subnet di dimensioni diverse

Indirizzi Speciali

- **Indirizzo di rete**: Tutti i bit host = 0
- **Indirizzo di broadcast**: Tutti i bit host = 1
- **Indirizzi privati**: 10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16

Concetti Fondamentali

Routing

- **Routing statico**: Tabelle configurate manualmente
- **Routing dinamico**: Protocolli automatici (RIP, OSPF, BGP)

Switching

- **Commutazione di circuito**: Percorso dedicato
- **Commutazione di pacchetto**: Store-and-forward
- **Commutazione di cella**: Dimensione fissa (ATM)

TEORIA - Modulo 1: Livello di Trasporto

Funzioni del Livello 4 (Trasporto)

Responsabilità Principali

- **Segmentazione**: Suddivisione dati applicazione in segmenti
- **Multiplexing/Demultiplexing**: Gestione flussi multipli
- **Controllo flusso**: Regolazione velocità trasmissione
- **Controllo errori**: Rilevazione e correzione errori
- **Controllo congestione**: Prevenzione sovraccarico rete

Porte (Port Numbers)

Classificazione Porte

- **Well-known ports** (0-1023): Servizi standard
- **Registered ports** (1024-49151): Applicazioni registrate
- **Dynamic/Private ports** (49152-65535): Uso temporaneo

Porte Principali

- **21**: FTP, **22**: SSH, **23**: Telnet, **25**: SMTP
- **53**: DNS, **80**: HTTP, **110**: POP3, **143**: IMAP

- **443**: HTTPS, **993**: IMAPS, **995**: POP3S

Protocollo TCP

Caratteristiche

- **Connection-oriented**: Richiede stabilimento connessione
- **Reliable**: Garanzia consegna ordinata
- **Full-duplex**: Comunicazione bidirezionale simultanea
- **Byte-stream**: Flusso continuo di byte

Three-Way Handshake (Connessione)

1. **Client → Server**: SYN (seq=x)
2. **Server → Client**: SYN-ACK (seq=y, ack=x+1)
3. **Client → Server**: ACK (ack=y+1)

Disconnessione (Four-Way Handshake)

1. **Client → Server**: FIN
2. **Server → Client**: ACK
3. **Server → Client**: FIN
4. **Client → Server**: ACK

Struttura Header TCP

- **Source Port** (16 bit): Porta mittente
- **Destination Port** (16 bit): Porta destinatario
- **Sequence Number** (32 bit): Numerazione byte
- **Acknowledgment Number** (32 bit): Prossimo byte atteso
- **Flags** (9 bit): URG, ACK, PSH, RST, SYN, FIN
- **Window Size** (16 bit): Controllo flusso
- **Checksum** (16 bit): Controllo integrità

Protocollo UDP

Caratteristiche

- **Connectionless**: Nessun stabilimento connessione
- **Unreliable**: Nessuna garanzia consegna
- **Lightweight**: Header minimo (8 byte)
- **Broadcast/Multicast**: Supporto nativo

Struttura Header UDP

- **Source Port** (16 bit)
- **Destination Port** (16 bit)
- **Length** (16 bit): Lunghezza header + dati
- **Checksum** (16 bit): Controllo integrità

IGMP (Internet Group Management Protocol)

Funzione

- Gestione appartenenza a gruppi multicast
- Comunicazione tra host e router multicast
- Versioni: IGMPv1, IGMPv2, IGMPv3

Socket

Concetto

- **Endpoint** di comunicazione di rete
- Identificazione univoca: (IP, porta, protocollo)
- **Berkeley Socket API**: Standard de facto
- Operazioni: socket(), bind(), listen(), accept(), connect()

Servizi di Trasporto

Orientati alla Connessione

- **TCP**: Affidabile, controllo flusso/errori
- Applicazioni: HTTP, FTP, SMTP, SSH

Senza Connessione

- **UDP**: Veloce, overhead minimo
- Applicazioni: DNS, DHCP, streaming, gaming

Qualità del Servizio (QoS)

- **Throughput**: Banda garantita
- **Delay**: Latenza massima
- **Jitter**: Variazione delay
- **Loss**: Percentuale perdita pacchetti

TEORIA - Modulo 2: Livello di Applicazione

HTTP/HTTPS (HyperText Transfer Protocol)

Caratteristiche HTTP

- **Stateless**: Ogni richiesta è indipendente
- **Client-Server**: Architettura request-response
- **Porto standard**: 80 (HTTP), 443 (HTTPS)
- **Metodi**: GET, POST, PUT, DELETE, HEAD, OPTIONS

Struttura Messaggi

Request: `METODO URI HTTP/versione` + Header + Body
Response: `HTTP/versione status-code reason-phrase` + Header + Body

HTTPS

- HTTP + SSL/TLS per crittografia
- Autenticazione server tramite certificati
- Confidenzialità e integrità dati

SMTP (Simple Mail Transfer Protocol)

Funzione

- **Invio email** tra server di posta
- **Porto standard**: 25 (SMTP), 587 (submission)
- **Architettura**: Push protocol (client spinge al server)

Processo di Invio

1. Connessione TCP al server SMTP
2. Handshake e autenticazione
3. Trasferimento messaggio (MAIL FROM, RCPT TO, DATA)
4. Chiusura connessione

POP3/IMAP (Protocolli Ricezione Email)

POP3 (Post Office Protocol v3)

- **Porto standard**: 110 (POP3), 995 (POP3S)
- **Download and delete**: Email scaricate localmente
- **Modalità**: Online, offline, delete
- Limitazioni: Accesso da singolo dispositivo

IMAP (Internet Message Access Protocol)

- **Porto standard**: 143 (IMAP), 993 (IMAPS)
- **Server-side storage**: Email rimangono sul server
- **Sincronizzazione**: Multi-dispositivo
- **Funzionalità avanzate**: Cartelle, flag, ricerca server-side

DNS (Domain Name System)

Funzione

- **Risoluzione nomi**: Traduzione hostname → IP
- **Porto standard**: 53 (UDP/TCP)
- **Struttura gerarchica**: Root, TLD, domini autoritative

Tipi di Record

- **A**: IPv4 address
- **AAAA**: IPv6 address
- **CNAME**: Canonical name (alias)
- **MX**: Mail exchange
- **NS**: Name server
- **PTR**: Reverse lookup
- **SOA**: Start of authority

Risoluzione DNS

1. **Recursive query**: Client → Recursive resolver
2. **Iterative queries**: Resolver → Root → TLD → Authoritative
3. **Caching**: Ottimizzazione performance

FTP (File Transfer Protocol)

Caratteristiche

- **Porto standard**: 21 (controllo), 20 (dati)
- **Due connessioni**: Control channel + Data channel
- **Modalità**: Active, Passive
- **Autenticazione**: Username/password o anonymous

Modalità di Trasferimento

- **Active FTP**: Server apre connessione dati verso client
- **Passive FTP**: Client apre connessione dati verso server
- **FTPS**: FTP + SSL/TLS
- **SFTP**: SSH File Transfer Protocol

DHCP (Dynamic Host Configuration Protocol)

Funzione

- **Configurazione automatica** parametri rete
- **Porto standard**: 67 (server), 68 (client)
- **Protocollo**: UDP broadcast

Processo DORA

1. **Discover**: Client cerca server DHCP
2. **Offer**: Server propone configurazione
3. **Request**: Client richiede configurazione specifica
4. **Acknowledge**: Server conferma assegnazione

Parametri Assegnati

- **Indirizzo IP**: Lease temporaneo

- **Subnet mask**: Maschera di sottorete
- **Default gateway**: Router predefinito
- **DNS servers**: Server risoluzione nomi
- **Lease time**: Durata assegnazione

Vantaggi

- **Gestione centralizzata**: Configurazione uniforme
- **Riduzione errori**: Eliminazione configurazione manuale
- **Mobilità**: Configurazione automatica su reti diverse
- **Ottimizzazione IP**: Riutilizzo indirizzi disponibili

TEORIA - Modulo 3: Sicurezza delle Reti

Tecniche Crittografiche

Crittografia Simmetrica

- **Chiave unica**: Stessa chiave per cifratura/decifratura
- **Velocità**: Computazionalmente efficiente
- **Problema**: Distribuzione sicura della chiave
- **Algoritmi**: AES, DES, 3DES, Blowfish

Crittografia Asimmetrica (Chiave Pubblica)

- **Coppia di chiavi**: Pubblica (nota) + Privata (segreta)
- **Principio**: Cifratura con una chiave, decifratura con l'altra
- **Applicazioni**: Cifratura dati, firma digitale, scambio chiavi

Aritmetica Modulare

- **Operazione mod n**: Resto della divisione per n
- **Proprietà**: (a + b) mod n = ((a mod n) + (b mod n)) mod n
- **Esponenziazione modulare**: Calcolo efficiente di a^b mod n

Algoritmi Crittografici

Diffie-Hellman (Scambio Chiavi)

1. Parametri pubblici: p (primo), g (generatore)
2. Alice: sceglie a (privato), calcola A = g^a mod p (pubblico)
3. Bob: sceglie b (privato), calcola B = g^b mod p (pubblico)
4. Chiave condivisa: K = A^b mod p = B^a mod p = g^(ab) mod p

RSA (Rivest-Shamir-Adleman)

Generazione chiavi:

1. Scegli p, q primi grandi

2. $n = p \times q$, $\varphi(n) = (p-1)(q-1)$
3. Scegli e coprimo con $\varphi(n)$
4. Calcola d: $e \times d \equiv 1 \pmod{\varphi(n)}$
5. Chiave pubblica: (e, n), Chiave privata: (d, n)

Operazioni:

- Cifratura: c = m^e mod n
- Decifratura: m = c^d mod n

Certificati e Firma Digitale

Certificati Digitali X.509

- **Contenuto**: Chiave pubblica + identità + firma CA
- **CA (Certificate Authority)**: Autorità certificazione
- **Catena di fiducia**: Root CA → Intermediate CA → End entity
- **Verifica**: Validazione firma CA con chiave pubblica CA

Firma Digitale

1. **Creazione**: Hash documento + cifratura hash con chiave privata
2. **Verifica**: Decifratura firma + confronto con hash documento
3. **Proprietà**: Autenticità, integrità, non ripudio

SSL/TLS (Secure Socket Layer/Transport Layer Security)

Funzioni

- **Autenticazione**: Verifica identità server (e client)
- **Cifratura**: Protezione dati in transito
- **Integrità**: Rilevazione alterazioni

Handshake TLS

1. **Client Hello**: Versioni supportate, cipher suite
2. **Server Hello**: Scelta parametri, certificato server
3. **Key Exchange**: Scambio materiale crittografico
4. **Finished**: Conferma completamento handshake

Difesa Perimetrale

Firewall

Tipi:

- **Packet Filter**: Filtraggio pacchetti (IP, porte)
- **Stateful**: Controllo stato connessioni
- **Application Gateway**: Proxy applicativo

Configurazioni:

- **Host-based**: Software su singolo host
- **Network-based**: Dispositivo dedicato di rete

Architetture Firewall

Single-Homed Bastion

- Un'interfaccia di rete
- Host dedicato nella rete interna
- Proxy per servizi esterni

Dual-Homed Bastion

- Due interfacce di rete (interna/esterna)
- Gateway obbligato per traffico
- IP forwarding disabilitato

Screened Subnet (DMZ)

- **DMZ**: Zona demilitarizzata tra due firewall
- **Isolamento**: Server pubblici separati da rete interna
- **Controllo**: Doppio livello filtraggio

Proxy Server

- **Forward Proxy**: Client → Proxy → Internet
- **Reverse Proxy**: Internet → Proxy → Server
- **Funzioni**: Cache, filtraggio contenuti, anonimato
- **Tipi**: HTTP, SOCKS, transparent

VPN (Virtual Private Network)

Funzione

- **Tunnel cifrato**: Connessione sicura su rete pubblica
- **Estensione rete privata**: Accesso remoto sicuro

Tipi VPN

- **Site-to-Site**: Connessione tra reti locali
- **Remote Access**: Utente remoto verso rete aziendale
- **Client-to-Client**: Comunicazione diretta cifrata

Protocolli VPN

- **IPSec**: Cifratura livello IP
- **L2TP/IPSec**: Layer 2 Tunneling + IPSec
- **OpenVPN**: SSL/TLS-based
- **WireGuard**: Protocollo moderno ad alte prestazioni

Normativa

GDPR (General Data Protection Regulation)

- **Principi**: Liceità, correttezza, trasparenza
- **Diritti**: Accesso, rettifica, cancellazione, portabilità
- **Obblighi**: Privacy by design, valutazione impatto
- **Sanzioni**: Fino 4% fatturato o 20M€

Misure Tecniche e Organizzative

- **Pseudonimizzazione**: Separazione dati da identificazione
- **Cifratura**: Protezione confidenzialità
- **Backup**: Disponibilità e ripristino
- **Controllo accessi**: Autorizzazione granulare

TEORIA - Modulo 4: Sistemi Distribuiti e Web

Modello Client/Server

Caratteristiche

- **Architettura asimmetrica**: Client richiede, server fornisce
- **Centralizzazione servizi**: Logica business sul server
- **Scalabilità verticale**: Potenziamento server centrale
- **Single point of failure**: Dipendenza da server centrale

Vantaggi/Svantaggi

Vantaggi: Gestione centralizzata, sicurezza, controllo
Svantaggi: Collo di bottiglia server, disponibilità critica

Sistemi Distribuiti

Definizione

Sistema di componenti software indipendenti che cooperano per apparire come sistema unico agli utenti.

Caratteristiche Fondamentali

- **Concorrenza**: Esecuzione simultanea processi
- **Mancanza clock globale**: Sincronizzazione complessa
- **Failure indipendenti**: Guasti parziali possibili
- **Trasparenza**: Nascondere complessità distribuzione

Modelli Architetturali

Peer-to-Peer (P2P)

- **Simmetria**: Ogni nodo è client e server
- **Decentralizzazione**: Nessun controllo centrale
- **Scalabilità orizzontale**: Aggiunta peer aumenta capacità
- **Resilienza**: Tolleranza guasti elevata

Multi-tier Architecture

- **Presentation Tier**: Interfaccia utente
- **Business Logic Tier**: Elaborazione regole business
- **Data Tier**: Gestione persistenza dati
- **Separazione responsabilità**: Manutenibilità e scalabilità

Microservizi

- **Decomposizione**: Applicazione in servizi piccoli e indipendenti
- **Comunicazione**: API REST/gRPC
- **Deployment indipendente**: Cicli sviluppo separati
- **Tecnologie eterogenee**: Stack tecnologico per servizio

Architetture Sistemi Web

Evoluzione Architetturale

Static Web (Web 1.0)

- **Contenuto statico**: Pagine HTML predefinite
- **Server web**: Apache, Nginx
- **Protocollo**: HTTP per trasferimento file

Dynamic Web (Web 2.0)

- **Contenuto dinamico**: Generazione runtime
- **Server-side scripting**: PHP, ASP.NET, JSP
- **Database integration**: RDBMS per persistenza
- **Session management**: Stato utente

Single Page Applications (SPA)

- **Client-side rendering**: JavaScript framework
- **API backend**: Servizi REST/GraphQL
- **Asynchronous loading**: AJAX per aggiornamenti parziali

Componenti Architettura Web

Web Server

- **Funzioni**: Gestione richieste HTTP, routing, sicurezza
- **Esempi**: Apache HTTP Server, Nginx, IIS

- **Configurazione**: Virtual hosts, moduli, cache

Application Server

- **Funzioni**: Esecuzione logica business, gestione sessioni
- **Esempi**: Tomcat, JBoss, WebLogic
- **Integrazione**: Database, servizi esterni, message queue

Load Balancer

- **Distribuzione carico**: Round-robin, least connections, IP hash
- **High availability**: Failover automatico
- **SSL termination**: Decifratura centralizzata

Pattern Architetturali Web

MVC (Model-View-Controller)

- **Model**: Dati e logica business
- **View**: Presentazione interfaccia utente
- **Controller**: Gestione input e coordinamento

REST (Representational State Transfer)

- **Principi**: Stateless, cacheable, uniform interface
- **Metodi HTTP**: GET, POST, PUT, DELETE
- **Rappresentazioni**: JSON, XML, HTML

Amministrazione di Rete

Network Management

- **Monitoring**: Controllo stato dispositivi e servizi
- **Configuration**: Gestione configurazioni centralizzata
- **Performance**: Analisi metriche prestazioni
- **Security**: Controllo accessi e compliance

SNMP (Simple Network Management Protocol)

- **Architettura**: Manager, Agent, MIB
- **Operazioni**: GET, SET, TRAP
- **Versioni**: SNMPv1, v2c, v3 (sicurezza avanzata)

Strumenti Amministrazione

- **Network scanners**: Nmap, OpenVAS
- **Monitoring**: Nagios, Zabbix, PRTG
- **Traffic analysis**: Wireshark, tcpdump
- **Configuration management**: Ansible, Puppet, Chef

Troubleshooting

Metodologia Sistematica

1. **Identificazione problema**: Sintomi e impatto
2. **Raccolta informazioni**: Log, metriche, configurazioni
3. **Analisi**: Correlazione dati, ipotesi cause
4. **Risoluzione**: Implementazione fix
5. **Verifica**: Test funzionalità
6. **Documentazione**: Procedure e lesson learned

Modello OSI per Troubleshooting

Bottom-up approach:

1. **Fisico**: Cavi, connettori, alimentazione
2. **Data Link**: ARP table, switch MAC table
3. **Rete**: Routing table, ping, traceroute
4. **Trasporto**: Porte aperte, connessioni attive
5. **Applicazione**: Servizi, configurazione applicativa

Strumenti Diagnostici

- **ping**: Connettività base (ICMP echo)
- **traceroute**: Percorso pacchetti
- **nslookup/dig**: Risoluzione DNS
- **netstat**: Connessioni e porte
- **tcpdump/Wireshark**: Analisi traffico
- **telnet**: Test connectivity porte specifiche