**Appunti riorganizzati 26/11**

**1. Dynamic Host Configuration Protocol (DHCP)**

**Cos’è il DHCP?**

È un protocollo usato per assegnare automaticamente indirizzi IP ai dispositivi connessi a una rete, evitando la configurazione manuale.

**Come funziona?**

1. **Scopo del DHCP**:
   * **Indirizzo IP**: Identifica un dispositivo nella rete.
   * **Subnet mask**: Determina quali indirizzi sono sulla stessa rete.
   * **Gateway predefinito**: Indirizzo del router per accedere a internet.
   * **Server DNS**: Traduce nomi di dominio (es. google.com) in indirizzi IP.
2. **Livello di comunicazione**:
   * Lavora al livello applicativo del modello TCP/IP, usando il protocollo UDP.

**Fasi principali del DHCP**

1. **DISCOVER**: Il dispositivo invia un messaggio di broadcast per trovare un server DHCP.
2. **OFFER**: Il server DHCP risponde offrendo un indirizzo IP e altre informazioni.
3. **REQUEST**: Il dispositivo accetta l’offerta e invia una conferma.
4. **ACK (Acknowledgment)**: Il server conferma l’assegnazione dell’indirizzo IP.

**Altri messaggi DHCP**

* **DECLINE**: Rifiuto dell’indirizzo IP offerto (es. se già in uso).
* **RELEASE**: Rilascio dell’indirizzo IP quando il dispositivo si disconnette.
* **INFORM**: Richiesta di informazioni aggiuntive senza un nuovo indirizzo IP.

**Caratteristiche principali**

* **Plug-and-play**: Non richiede configurazione manuale.
* **Assegnazione temporanea**: Gli indirizzi IP sono assegnati per un periodo limitato (lease time).

**Esempio pratico**

1. Un laptop si connette al Wi-Fi di un hotel.
2. Invia un messaggio **DISCOVER** per cercare un server DHCP.
3. Il server risponde con un **OFFER**.
4. Il laptop invia un **REQUEST** accettando l’offerta.
5. Il server conferma con un **ACK**.

**2. Internet Control Message Protocol (ICMPv4)**

**Cos’è l’ICMP?**

Un protocollo utilizzato per inviare messaggi diagnostici ed errori nelle reti IPv4. Non risolve gli errori, ma li segnala e aiuta a diagnosticarli.

**Perché è necessario?**

Il protocollo IP non prevede meccanismi per:

* Segnalare errori.
* Verificare lo stato della rete.
* Effettuare richieste di informazioni.

**Caratteristiche principali**

1. Lavora al livello di rete, incapsulato nei datagrammi IP.
2. Ogni messaggio include:
   * **Tipo**: Categoria del messaggio (es. errore o richiesta).
   * **Codice**: Dettagli sull’errore.
   * **Checksum**: Verifica l’integrità del messaggio.
   * **Dati aggiuntivi** (es. parte del datagramma originale).

**Tipi di messaggi ICMP**

1. **Messaggi di errore**:
   * **Destination Unreachable**: La destinazione non è raggiungibile.
   * **Time Exceeded**: Il campo TTL è scaduto.
   * **Redirect**: Esiste un percorso migliore.
   * **Parameter Problem**: Problemi nei campi del datagramma.
2. **Messaggi di richiesta e risposta**:
   * **Echo Request/Echo Reply**: Usati dal comando **ping**.
   * **Timestamp Request/Reply**: Per sincronizzare gli orologi tra dispositivi.

**Applicazioni pratiche**

1. **Ping**: Verifica la raggiungibilità di un host.
2. **Traceroute**: Determina il percorso seguito dai pacchetti per raggiungere una destinazione.

**3. Routing Unicast**

**Cos’è?**

Il routing unicast permette di inviare pacchetti IP da una sorgente a una singola destinazione, utilizzando un percorso definito hop-by-hop.

**Concetti di base**

1. **Tabelle di instradamento**:
   * Contengono informazioni su:
     + Indirizzo di rete di destinazione.
     + Prossimo hop (router successivo).
     + Costo del percorso.
2. **Criterio del costo minimo**:
   * I percorsi vengono scelti in base a metriche come distanza, ritardo o larghezza di banda.

**Algoritmi di routing**

1. **Distance Vector (Bellman-Ford)**:
   * Ogni router scambia informazioni con i vicini.
   * Problemi: Conteggio all’infinito.
2. **Link State (Dijkstra)**:
   * Ogni router costruisce una mappa completa della rete.
3. **Path-Vector**:
   * Usato in BGP (es. politiche di instradamento).

**Problemi comuni**

1. **Conteggio all’infinito**:
   * Soluzioni: **Split Horizon**, **Poison Reverse**.
2. **Instabilità**:
   * I protocolli Link State reagiscono più velocemente ai cambiamenti.

**4. Algoritmo Bellman-Ford (Distance Vector)**

**Come funziona?**

1. Ogni router ha una tabella con:
   * Destinazioni.
   * Costi.
   * Prossimo hop.
2. I router scambiano periodicamente informazioni con i vicini.
3. Ogni router aggiorna la tabella se trova un percorso più economico.

**Esempio pratico**

1. Rete: **A → B (1)**, **B → C (2)**, **A → C (∞)**.
2. **Scambio**:
   * B informa A che può raggiungere C con costo 2.
   * A aggiorna: A → C via B (costo 3).

**5. Algoritmo di Dijkstra**

**Come funziona?**

1. Assegna al nodo di partenza un costo 00 e agli altri ∞∞.
2. Seleziona il nodo con il costo minimo.
3. Aggiorna i costi dei vicini.
4. Ripeti finché tutti i nodi sono visitati.

**Esempio pratico**

1. Grafo: **A → B (1)**, **A → C (4)**, **B → C (2)**, **B → D (6)**, **C → D (3)**.
2. Calcoli:
   * Nodo A: B (1), C (4).
   * Nodo B: C (3), D (7).
   * Nodo C: D (6).

**6. Routing Statico Centralizzato**

**Cos’è?**

Un metodo in cui le rotte sono configurate manualmente da un amministratore centrale.

**Vantaggi**

* Controllo totale.
* Efficienza in reti piccole.

**Svantaggi**

* Scalabilità limitata.
* Mancanza di flessibilità.

**Quando usarlo?**

* Reti piccole e stabili.
* Configurazioni specifiche come VPN.

**Conclusione**

Questi protocolli e algoritmi formano le basi della gestione e dell’ottimizzazione delle reti moderne. La scelta dipende dalle esigenze specifiche della rete e dalle sue dimensioni.