UNIT III

THE NETWORK LAYER

Routing algorithms: Distance vector, link state, and path vector

Distance vector, link state, and path vector are all dynamic routing algorithms used in the network layer to find the best path for data packets. Distance vector routers share their distance vectors with neighbors to calculate routes based on hop count, while link state routers flood a complete network map to all routers, allowing each to independently compute the shortest path using an algorithm like Dijkstra's. Path vector routing, designed for large, complex networks, maintains the entire path to a destination to avoid issues like routing loops.

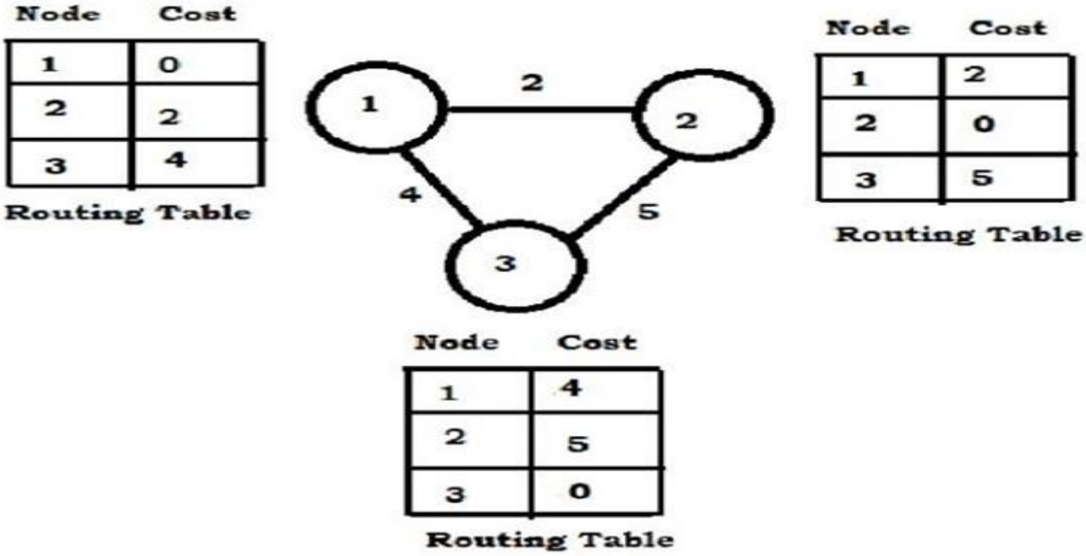
Distance vector

Distance-vector routing is a type of dynamic routing protocol where each router maintains a routing table with the "distance" (cost) to reach other nodes and the next hop to use. Routers periodically exchange their routing tables with their immediate neighbors to learn about the network topology and update their own tables with the best-known routes. This is also known as the Bellman-Ford algorithm.
How it works

- **Initialization**: When a router starts, it sets the distance to itself as 0 and the distance to all other routers as infinity.
- **Sharing information**: Routers share their routing tables with their direct neighbors. This exchange happens either periodically or when a change is detected in the network.
- **Calculating distance**: When a router receives an update from a neighbor, it uses the neighbor's information to recalculate its own routing table. The new distance to a destination is calculated by taking the minimum of the costs through all its neighbors.
- **Convergence**: Routers continue to exchange information until the tables stabilize, and all routers have learned the best paths to all destinations in the network.

Key characteristics

- **Metric**: Uses distance, often measured in hops (the number of routers a packet passes through), to determine the best path.
- **Information**: Each router only sends its distance and next-hop information to its neighbors, not the entire network map.
- **Advantages**: Simple to implement and has a lower processing overhead compared to other protocols.
- **Disadvantages**: Can lead to **counting-to-infinity problems** and has **slower convergence** compared to link-state routing protocols, making it less suitable for large or highly dynamic networks.



Link state

Link-state routing is a dynamic routing algorithm where each router builds a complete map of the network's topology, not just its immediate neighbors. Routers achieve this by flooding their local link information to all other routers, and each router then independently uses an algorithm like Dijkstra's algorithm to calculate the shortest path to every other destination on the network. Common examples of link-state protocols include OSPF and IS-IS.

How it works

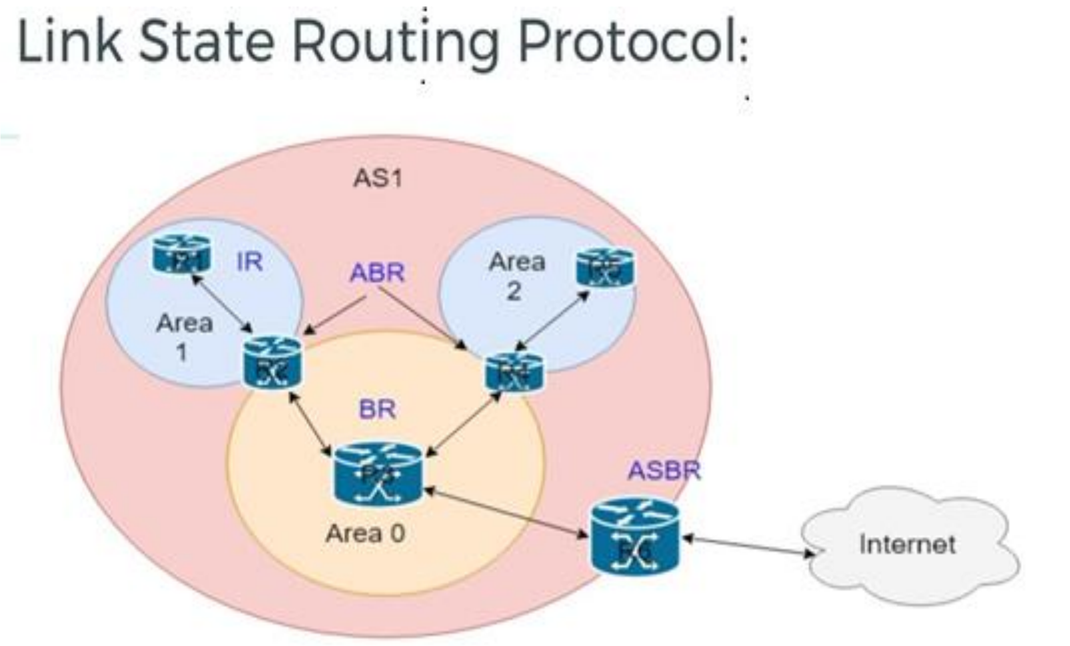
- **Neighbor discovery:** Each router initially learns about its direct neighbors by sending out "hello" packets.
- **Link-state packet (LSP) creation:** After discovering neighbors, each router creates an LSP containing information about its directly connected links and their cost or status.
- **Flooding:** The LSP is then flooded throughout the entire network, ensuring every router receives a copy. Routers use a sequence number in the LSP to prevent loops and ensure they have the most recent information.
- **Topology map creation:** Each router stores all the LSPs it receives to build a complete, synchronized map of the entire network's topology.
- **Shortest-path calculation:** Independently, each router runs Dijkstra's algorithm on its topology map to compute the shortest path to every other router.
- **Routing table generation:** The results of the Dijkstra's algorithm are used to build the router's final routing table, which determines the best path for outgoing data packets.

Advantages

- **Global network view:** Provides routers with a comprehensive understanding of the entire network.
- **Efficient routing:** Enables efficient routing by calculating the shortest path to all destinations.
- **Faster convergence:** Convergence, the process of all routers agreeing on the network topology, is typically faster than in other protocols.

Disadvantages

- **Higher CPU and memory usage:** Requires more processing power and memory to run Dijkstra's algorithm on a large network.
- **Increased traffic:** Flooding LSPs can create a significant amount of network traffic, especially during network changes.



Path vector

Path vector routing is a routing algorithm where "speaker nodes" in each autonomous system (AS) advertise the full path of ASs to reach a destination, instead of just the distance. This method is primarily used for inter-AS routing, and its main benefit is preventing routing loops by making the entire path visible to routers, which can then use policy to choose the best route rather than just the shortest one. The Border Gateway Protocol (BGP) is a well-known example of a path vector protocol.

How it works

- **Path advertisement**: A "speaker node" in an autonomous system advertises the full sequence of ASs a packet must pass through to reach a destination.
- **Loop prevention**: By knowing the entire path, a router can easily detect if a path has already visited its own AS, and it can then discard the looped route to avoid a routing loop.
- **Policy-based routing**: Unlike distance vector routing which often defaults to the shortest path, path vector routing allows routers to make decisions based on policies imposed by the source AS, such as cost, security, or other attributes, not just the number of hops.
- **Inter-domain routing**: It is ideal for routing between different autonomous systems (e.g., between different Internet Service Providers) because it provides the necessary information for each AS to manage its traffic based on its own policies.
- **Routing decisions**: A router receives both the path and other attributes (like distance metrics) and uses the BGP path selection process to determine the best route.

Key differences from distance vector routing

- **Information shared**: Path vector shares the complete path of ASs, whereas distance vector only shares the distance (or cost) and next hop.
- **Routing decision basis**: Path vector decisions are based on path attributes and policy, while distance vector primarily aims for the shortest or least-cost path.