The Network Layer: Control Plane

SOFTENG 364: Computer Networks

Compiled: May 24, 2018

5.1. Introduction

1. Identify or describe the sublayers of the transport layer. Which is "on top"?

Plane	Protocols	Entities
management	SNMP	CLI etc.; interface between operator and control plane
data/forwarding		switching ports and -fabric, FIB
control	OSPF, BG	updates FIB

The control plane is "on top"; the management plane is "down the side".

2. What is the role of the network control plane?

How forwarding/flow tables are *computed, maintained, and installed*, which considers:

- "optimal" paths
- loop avoidance
- flow classification
- failure recovery
- traffic engineering

3. What distinctions might someone (careful) make between **forwarding** and **routing**? Can you think of nice analogies?

According to KR-336:

Forwarding *router-local action of transferring a packet from an input link interface to the appropriate output link interface[s]*; short time-scales (nanoseconds); typically implemented in hardware; associated with L3 (cf. switching)

Routing *network-wide process that determines . . . end-to-end paths*; longer time-scales (seconds); software implementation; also associated with L3

Some people are less careful about this distinction. It seems to be OK to use forwarding and **switching** as synonyms.
Military analogy: strategy vs tactics/execution

4. Why is the network layer split into two planes? Why not regard the planes as separate layers?

Separation In light of the emergence of SDN, the separation clearly makes sense.

Layering The control plane's role is that which we've associated with the Network Layer: Communication between hosts. The data plane's relationship is obviously intimately related, but would it be reasonable to associate it with the Link Layer instead?

5. What comes between the input/**ingress** port and output/**egress** port as a packet travels through a router?

Switching fabric

6. What is the name of the **protocol data unit** (PDU) associated with the Network Layer? With the transport layer?

Layer	PDU
Application	file/data
Transport	TCP **segment**, UDP **datagram**
Network	IP **packet**
Link	Ethernet **frame**
Physical	bit/**symbol**

5.2. Routing Algorithms

7. What are the constituent parts of a **graph**? Recall alternative terms for each constituent.

A set of **nodes** and a set of **links**. Each link is an (unordered) pair of nodes. A weight may be associated with each link.

8. What is a **path** in a graph? What is the **length** of a path? What is the **cost** of path? What is a **least-cost** path? In what sense is a **shortest path** a least-cost path? Are least-cost paths **unique**?

Path A subset of edges $P = \{e_j\}_{j=1}^{K}$, all in the graph, with $\text{head}(e_j) \equiv \text{tail}(e_{j+1})$ and $0 \leq K$

length The number of edges, K

cost A sum of the form $c = \sum_{e \in P} c(e) = \sum_{j=1}^{N} c(\text{tail}(e_j), \text{head}(e_j))$ for some **cost function** c

least-cost path Minimizes the cost over all paths between $\text{tail}(e_1)$ and $\text{head}(e_K)$

relationship The length uses $c(\cdot) \equiv 1$

uniqueness cost function c may be chosen; for a fixed c, multiple paths may have identical cost/length

9. Describe how routing protocols may be characterised/classified.

Classification according to communication pattern:

Centralized algorithm e.g. link-state

Decentralized algorithm e.g. distance-vector

Warning Distinguish between centralized routing algorithm (link-state algorithm) with (logically) centralized routing controller (software defined networking)

Classification according to awareness of **topology**:

Static routing Routing tables are specified manually

Dynamic/adaptive routing Routing tables maintained automatically via **routing protocol** e.g. RIP

Classification according to awareness of **congestion** on network links:

Load-sensitive link costs reflect current level of congestion e.g. early ARPAnet

Load-insensitive link costs do not explicitly reflect current level of congestion e.g. RIP, OSPF

10. Contrast **centralized** and **decentralized** routing algorithms.

11. What does **link-state** refer to in LS routing algorithm? What does **distance vector** refer to in DV routing algorithm?

LS Costs on all links in the network

DV Distances from a specified node in a graph to all other nodes

Link costs are also called **administrative weights**.

12. Must link costs be symmetric (i.e. `c(u, v) == c(v, u)`)?

No: Effectively, we'd use two **directed** edges for each link

13. Under what condition(s) is a complete set of least-cost paths guaranteed to exist?

When the network has no negative-cost loops. In an undirected graph, any link with negative cost is a negative-cost loop.

14. What are **routing loops**?

Simplest case: Two nodes forward packets via each other.

15. Do routing loops actually give rise to infinite cycling? Why are they dangerous nonetheless?

TTL field prevent infinite cycling; packet doesn't reach destination in any case

5.2.1. LS Routing

16. Recall examples of real LS protocols.

OSPF	Open Shortest Path First	Discussed in this section
IS-IS	Intermediate System to Intermediate System	Close cousin of OSPF
OLSR	Optimized Link State Routing Protocol	For mobile ad hoc networks

17. What is the purpose of a **link-state broadcast**? How might it work?

Each node needs knowledge of all links and link costs:

(a) A **reachability protocol** is used to determine node's neighbours.

(b) Periodically sent **link-state advertisement** with (1) node's ID, (2) neighbour's IDs, (3) sequence number.

When receiving a link-state advertisement whose sequence number is newer than that already stored for originating node:

update saved copy;

forward to all neighbours

18. What is the correct pronunciation of **Dijkstra**?

Rhymes with "dyke"

19. What inputs does Dijkstra's algorithm require? What outputs does it produce?

Inputs graph (`N`, `E`), link costs `c`, source node `u`

Outputs value `D(v)` of least-cost path from source to each node `v`; predecessor `p(v)` of each destination `v` on least-cost path

20. What other data structure is required?

Set `N'` of destination nodes whose least-cost path value is known.

21. Describe the initialization phase of Dijkstra's algorithm.

```
for v in N: D(v) = c(u, v) if (u, v) in E else inf
```

22. Describe each step of the "main loop". What is the bottleneck? How are ties resolved?

find `w in N - N'` with minimal `D(w)`; this is the bottleneck
Ties resolved by arbitrary selection

finalize `N'.add(w)`

update `for v in G.neighbors(w) - N': D(v) = min(D(v), D(w) + c(w, v))`

23. How many steps does the algorithm require to solve the single-source least-cost path problem? How many comparisons are required at each step? Justify the claim of quadratic complexity.

Analysis for naive implementation of KS pseudocode:
Define $n := |N| - 1$:

find
- Require n steps after initialization
- Step 1 searches n candidates
- Step 2 searches $n-1$ candidates
- Step 3 searches $n-2$ candidates etc.

$$n + (n-1) + (n-2) + \cdots 1 \equiv \frac{n^2 - n}{2} \equiv \frac{n(n-1)}{2} \quad \text{i.e. } \mathcal{O}(n^2) \text{ complexity}$$

update Each node has $\leq n$ neighbours: $\leq n^2$ individual updates is still $\mathcal{O}(n^2)$ complexity

24. Explain how the complexity can be reduced?

Use a **priority queue** (implemented via binary/pairing/Fibonacci **heap**) to eliminate a complete search through the set of candidate nodes – reducing it from linear to logarithmic complexity;
Resulting complexity of combined **find** is $\mathcal{O}(n \log(n))$

25. ** Dijkstra's algorithm solves the **single-source** least-cost paths problem. On sufficiently dense graphs, the Floyd-Warshall algorithm is a more efficient means of solving the **all-pairs** least-cost path problem. Why don't we discuss that algorithm instead?

In the context of the traditional (per-router) control plane, a router only cares about paths from itself to other nodes.
The all-pairs problem would, presumably, be of interest in the context of SDN.

26. Which parts of the LS algorithm involve inter-node communication?

The link-state broadcast; not least-cost path algorithm itself.

27. How would we compute the **forwarding table** from the outputs of Dijkstra's algorithm?

> *it is only necessary to **walk the tree**, remembering the identity of the node at the head of each branch, and filling in the routing table entry for each node one comes across with that identity* – wikipedia.org

We'll code this up in the lab/assignment.

28. Can a link-state scheme produce **routing loops**? Explain.

Yes, unless all nodes have exactly the same link costs; see failure modes for details.

5.2.2. DV Routing

29. Recall examples of real DV protocols.

BGP	Border Gateway Protocol	To be discussed in class
RIP	Routing Information Protocol	One of the oldest DP protocols
ARPAnet		First network to implement TCP/IP
IDRP	InterDomain Routing Protocol	OSI/ISO relative of BGP
IPX	Internetwork Packet Exchange	Legacy Novell protocol

30. In the terms **distance-vector** (DV) protocol/algorithm, what does "distance-vector" refer to?

(Estimate of) path cost (cf. link cost)

31. How does the DV algorithm we have discussed relate to "the" Bellman-Ford algorithm?

It is a **distributed/asynchronous** variant.

32. What inputs are required by each router in a DV protocol?

Each node x requires:

Costs on incident links i.e. $\{(v, c(x, v)) : (x, v) \in E\}$

DVs of neighbours (only)

33. Recall and describe the **dynamic programming** equation? What does it characterize?

It relates the values $D_\bullet(y)$ of minimum-cost paths with common destination y:

$$D_x(y) = \min_v \; c(x, v) + D_v(y)$$

It characterizes minimal path lengths in a graph.

34. How is the equation **relaxed** in the DV algorithm? Describe the minimization that takes place at each step.

Use its right-hand side to update LHS $D_x(z)$, using current estimate of $D_y(z)$;

```
for y in N: D[x][y] = min([c(x, v) + D[v][y] for v in G.neighbors(x)])
```

35. In what sense is this DV procedure **iterative**, **asynchronous**, **distributed**?

Asynchronous Independent execution on distinct routers; no central "clock"/coordinator

Distributed Executed on (geographically) distinct nodes with separate address spaces

Iterative Number of steps to convergence is not known *a priori*

36. What would happen if a link cost changed?

(a) Attached nodes update their own DVs using old DVs of neighbours and new link cost

(b) Advertise updated DVs to neighbours

37. Can the DV iteration produce **routing loops**? Explain.

Like LS schemes, loops may arise unless data is consistent. In addition, loops may arise until the nodes reach convergence (recall that the iteration is asynchronous).

38. Describe the **count-to-infinity** problem, and a means of countering it. Is the remedy guaranteed to be effective?

Best demonstrated via one step of example in the slides:

Issue If z advertises existence of a path P to x (i.e. finite $D_z(x) < \infty$), y can't tell whether $y \in P$ (where affected link is incident on y and z).

Objective asymmetry Lower costs are used/preferred over higher costs even if they are out of date! $\therefore$ "Good news travels fast, bad news travels slowly"

Remedy? Poisoned reverse uses "infinite" $D_y(z)$ – not guaranteed to work.

5.2.1-2. Routing protocols

39. Contrast the efficacy of LS and DV algorithms in terms of message/communication complexity, speed of convergence, and robustness.

Message complexity LS: New link cost sent to all nodes. DV: New link cost sent to adjacent nodes only; subsequent communication with neighbours only if DV changes.

Speed of convergence LS is $\mathcal{O}(|N|^2)$ or $\mathcal{O}(|N|\log|N|)$. DV may suffer from count-to-infinity problem; admits routing loops while converging

Robustness in face of security breach or malfunction: Inconsistent costs/topology admit loops in either case. LS nodes are otherwise insulated. DV errors may propagate to entire network.

Ultimately, protocols of both types are supported by modern commercial routers.

40. How might the relative significance of these factors relate to network characteristics?

- In a small network, speed gains are small in absolute terms
- Processing power of nodes relative to link bandwidth/delay
- Network security (to be discussed); hardware reliability

41. How can oscillations arise in **congestion-sensitive** routing? Are these peculiar to LS or DV schemes?

Yes: If costs are based on current link load, reflecting delay.
They can arise in any congestion-sensitive scheme.

42. Why might this oscillation be regarded undesirable?

Large fluctuations in link load suggest that capacity is under-utilized.

43. How might oscillation be addressed in practice?

- (It is not helpful to suggest that link costs should not depend on load!)
- Ensure that routers do not run LS algorithm at the same time.
- Randomize link advertisement times to prevent synchronization.

44. What is a **tree**? What is a **shortest-path tree**? Why is it always a tree?

A tree is a graph without loops.
Shortest paths don't go around loops (if costs are positive)

45. What is the significance of the shortest path tree?

It carries all traffic i.e. no traffic is routed on links on the complement of the shortest path tree – a waste of an expensive resource.

46. What do we mean by traffic engineering?

The discipline of setting link costs to e.g. minimize maximum link utilization. See discussion on [KS-421].

5.3. Intra-AS Routing: OSPF

47. What is an **autonomous system** (**AS**)? What is the relationship between **ISP** and AS?

> *The use of the term Autonomous System here stresses the fact that, even when multiple IGPs and metrics are used, the administration of an AS appears to other ASes to have a single coherent interior routing plan and presents a consistent picture of what networks are reachable through it.*
> *To rephrase succinctly:*
> *An AS is a connected group of one or more IP prefixes run by one or more network operators which [present] a SINGLE and CLEARLY DEFINED routing policy [to the Internet].* – from RFC 1930, Section 3

AS A network whose routers are under common administrative control and (hence) running a common routing protocol.

ISP comprise one or more (up to "tens") interconnected ASs

48. How are autonomous systems addressed?

By **autonomous system numbers** (**ASN**s); like IP addresses, assigned uniquely by ICANN for **BGP** routing

> *The **Internet Assigned Numbers Authority** (IANA) is a department of [the Internet Corporation for Assigned Names and Numbers] **ICANN**, a non-profit private American corporation that oversees global IP address allocation, autonomous system number allocation, root zone management in the Domain Name System (DNS), media types, and other Internet Protocol-related symbols and Internet numbers.* — wikipedia.org

49. What explains/motivates/justifies the partitioning of the Internet into **autonomous systems**?

Scale Routing complexity (communication, table size) limited by AS size

Administrative autonomy of organizations with different requirements/interests

50. What does OSPF stand for? What does this tell us?

Open Shortest Path First; it is an open standard

51. What type of routing protocol does it employ?

LS protocol (Dijkstra) for intra-AS routing between subnets

52. How are OSPF messages transmitted? What does this imply?

- OSPF messages are carried directly by IP (i.e. L3 cf. L4 TCP/UDP); hence, OSPF is responsible for its own reliable message transfer and link-state broadcast (note that packets destined to IP broadcast address are never routed through routers i.e. they're restricted to a single subnet)
- OSPFv2 is IPv4-only; OSPFv3 support IPv6
- **Protocol number** 89 in IPv4 **Protocol** field or IPv6 **Next Header** field

53. How are OSPF link costs (**administration weights**) assigned?

The standard does not prescribed a particular assignment strategy. KS-421 provides nice discussion of *the goal of minimizing the maximum link utilization* (an instance of the **minimax path problem**), closely related to maximizing minimum link bandwidth (an instance of the **widest path problem**).

54. When do OSPF routers broadcast link-state information?

- whenever there is a change in link cost
- whenever there is a change in nodal up/down status
- periodically, in any case – for robustness

55. Which "advanced features" does OSPF provide?

Security authenticated messages (MD5, see Chapter 8)

Multiple same-cost paths for different TOS (e.g. best effort, real-time)

Multi-casting **MOSPF** adds additional link-state advertisement

Hierarchy For large networks

56. What is an OSPF **area**? How do areas limit the complexity of routing?

A logical partition of the ASN.
Link-state advertisements are restricted to single areas, limiting communication and the size of link-state databases

57. How are areas identified (addressed)?

> *Areas are uniquely identified with 32-bit numbers. The area identifiers are commonly written in the dot-decimal notation, familiar from IPv4 addressing. However, they are not IP addresses and may duplicate, without conflict, any IPv4 address. . . . When dotted formatting is omitted, most implementations expand area 1 to the area identifier `0.0.0.1`, but some have been known to expand it as `1.0.0.0`.* – en.wikipedia.org

58. Describe each category of router in OSPF. What is the difference between an Area Border Router (ABR) and a Backbone Router (BR)?

Internal Router (IR) All interfaces in same area

Area Border Router (ABR) Interfaces on Area 0 and ≥ 1 (non-backbone) areas; one LS database for each area

Backbone Router (BR) has an interface on Area 0; may be ABR or not

Autonomous System Boundary Router (ASBR) Connects to other ASs via exterior routing protocol (i.e. BGP)

5.4. Inter-AS Routing: BGP

59. *BGP is arguably the most important of all the Internet protocols (the only other contender would be the IP protocol . . .)* [KR-423]: Why?

It is the common glue linking all of the ISPs/ASNs that comprise the Internet.

60. What are the responsibilities of an BGP?

Reachability Allow AS to **advertise** own existence and obtain subnet **reachability** data from neighbours

Routing Select inter-AS routes based on reachability and **policy**

It operates in a way that respects the policies of independent operators.

61. Who/what sends BGP messages? (ASs or routers?) How are they messages sent?

- Sent between routers
- Semi-permanent links between gateway routers
- **TCP** connections (L4) using port 179

62. Describe the two categories of BGP routers?

Gateway router At least one interface connected to router in another AS

Internal router All interfaces connect to nodes in own AS

63. If BGP is an **inter**-AS protocol, why mention internal routers?

Internal routers need to send packets outside of AS

64. Contrast **eBGP connections** and **iBGP connections**. Do these involve different protocols?

*When two BGP-enabled devices are in the same autonomous system (AS), the BGP session is called an**internalBGP session**, or **IBGP session**. BGP uses the same message types on IBGP and **external BGP (EBGP) sessions**, but the rules for when to send each message and how to interpret each message differ slightly. For this reason, some people refer to IBGP and EBGP as two separate protocols.* — documentation from Juniper Networks

65. Relate and contrast **forwarding tables** and **routing tables**.

> *Routing tables are generally not used directly for packet forwarding in modern router architectures; instead, they are used to generate the information for a smaller forwarding table. A forwarding table contains only the routes which are chosen by the routing algorithm as preferred routes for packet forwarding. It is often in a compressed or pre-compiled format that is optimized for hardware storage and lookup.* — wikipedia.org

66. What kind of "addresses" appear in BGP forwarding tables?

- CIDRized prefixes, each representing an IP subnet (or collection of subnets)
- i.e. each "row" in routing table contains `(prefix, router interface)`
- *BGP routing tables often contain over half a million routes (that is, **prefixes** and corresponding **attributes**)* — [KR-430]

67. How many IPv4 addresses would the destination `A.B.C.D/24` include? What if we exclude reserved addresses?

- $2^{32-24} = 2^8 = 64$ possibilities
- `A.B.0.0` used to refer to network itself ("network `A.B`")
- `A.B.255.255` used as network broadcast address ("all hosts on network `A.B`")
- $\therefore\ 64 - 2 = 62$ addresses available for host addresses
- See Section 7 of IETF RFC 922

68. What is a BGP **route**?

$$\begin{aligned}(\text{BGP } \textbf{route}) &= \texttt{[(BGP attributes); prefix]}\\ &= \texttt{[(NEXT\_HOP, AS\_PATH, etc.); prefix]}\end{aligned}$$

69. Explain the `AS_PATH` **attribute**. Why store the entire path? (Isn't it enough to know the next hop?)

- A list of ASs through which the advertisement has passed.
- May be used to prevent **routing loops**: ASk rejects any advertisement whose `AS_PATH` already contains ASk

70. Explain the `NEXT_HOP` attribute.

- IP address of router interface that begins associated `AS_PATH` i.e. path back to advertised prefix
- Interface is on gateway router in neighbouring AS
- ... but direct connection is available from current AS

71. What is the `NEXT_HOP` if the destination subnet is contained in the current AS?

`0.0.0.0` i.e. “this network” in IP networking – see networklessons.com

72. Compare BGP’s **path vector** routing protocol with the **link-state** and **distance vector** protocols.

Context We discussed LS and DV in the context of intra-AS routing

History BGP superseded Exterior Gateway Protocol (EGP) for inter-AS routing:

> *EGP is a simple **reachability protocol**, and, unlike modern distance-vector and path-vector protocols, it is limited to **tree-like topologies**. . . . During the early days of the Internet, EGP version 3 (EGP3) was used to interconnect autonomous systems. Currently, BGP version 4 is the accepted standard for Internet routing and has essentially replaced the more limited EGP3.* — wikipedia.org

Complete elimination of **routing loops**

Complete control of inter-AS **routing policy**

73. What is **hot potato routing**? (Is it an alternative to OSPF shortest-path routing?)

Issue “Which local gateway to choose for external destination?”

Hot potato Prefer gateway closest to source

The (OSPF) least-cost route is taken to whatever gateway is chosen – whether or not that is the hot potato choice

74. Explain the BGP **route selection algorithm**.

Apply following criteria in sequence while ties persist:

(a) Highest `LOCAL_PREF` – set by administrator
(b) Shortest `AS_PATH` (cf. DV with hop count cost metric)
(c) Hot-potato routing to closest `NEXT_HOP`
(d) BGP identifiers

75. In the BGP route selection algorithm, why does Step 2 come before Step 3?

- The resulting process is “less selfish” i.e. considers end-to-end delay
- (2) relates to inter-AS efficiency, (3) to intra-AS efficiency

76. Explain the BGP **attributes**

(a) `NEXT_HOP`,
(b) `AS_PATH`, and
(c) `LOCAL_PREF`.

NEXT_HOP

AS_PATH

LOCAL_PREF

77. What is a BGP **identifier**?
78. What is the **IP-anycast** service?
79. Why is BGP not used to implement IP-anycast for **CDN** services?

Changes in the BGP routing table can result in different packets of the same connection arriving at different instances of the replicated server (which replicate content, but not session data)

80. How and why is BGP used to implement the DNS system?

- Multiple/distinct BGP advertisements are made for the same IP address
- BGP routers resolve one address to the "nearest" of these servers
- 100s of DNS root servers cf. 13 IP addresses

81. What is a **multi-homed** ISP? Why is BGP-style routing policy particularly relevant in this case?

Connects to > 1 backbone network;
It could be used as a transit network for traffic between backbone networks

82. Describe the **routing policy** typically adopted by commercial ISPs (in the absence of negotiated agreements).

Traffic flowing across an backbone network (or multi-homed ISP) must have a source and/or destination in a customer subnet i.e. there should be some revenue associated with the traffic

83. How does BGP's **path vector** routing compare with DV- and LS routing?

PV asynchronous, distributed; based on paths; hop-count cost in BGP; eliminates loops

DV asynchronous, distributed, iterative, count-to-infinity; admits loops

LS centralized (i.e. all routers must be notified of every single link cost change); robust convergence

84. Why are different protocols used for inter- and intra-AS routing?

See the nice discussion on [KR-433]:

	inter (BGP)	intra
Policy	allows individual operators to control transit traffic	Operators have full control to route for optimal utilization
Scale	Key concern: Can't ask an AS to split!	Easy to split areas
Performance	Policy has priority; hop-count is only inter-AS cost measure	Specified link costs

85. What means does BGP provide operators to control routing?

Route advertisements Whether to accept/send advertisements; controls reachability

`LOCAL_PREF` **attribute** Controls route selection

86. Describe the steps involved in establishing **Internet presence**.

(a) ISP provides subnet prefix
(b) Assign one address to your DNS server
(c) Domain name registrar associates your DNS server address with domain name
(d) ISP advertises your prefix via BGP
(e) Anyone with your domain name gets your DNS server's IP address via the Internet's DNS
(f) Your DNS server provides them with IP addresses of your other servers (web/mail etc.)

4.4. SDN: Data Plane ← From Chapter 4

87. Recall examples of **middleboxes** whose functions can be performed by SDN systems.

22 classes of middlebox are described in IETF RFC 3234, including NAT, firewalls (and DPI), load-balancing

88. In what way(s) does "generalized" forwarding generalize traditional **destination-based forwarding**?

Match Traditional forwarding matches on destination only, as opposed to multiple fields at different layers.

Action Traditional forwarding has more limited support: Only modifies TTL and checksum, and either drops or forwards to an output port.

89. Contrast the fields/columns of a traditional **forwarding table** with OpenFlow's **flow table**.

Forwarding IP routing prefix + outgoing interface

Flow Multiple header field values + zero-or-more actions

90. How are forwarding/flow tables populated?

Remotely, by a central controller. This is the role of the network layer's **control plane**, which we'll discuss in Chapter 5.

91. Is "generalized forwarding" standard/official terminology?

No, although it is accurate/suitable. Other terms relating to generalizations of traditional routing include **policy-based routing** (PBR), **filter-based forwarding** (FBF), **flow-based forwarding**, and **source routing**.

92. What are forwarding devices typically referred to at L2? At L3? In SDN?

L2 **switches**; L3 **routers**; SDN **packet switches**

93. Review Exercise 3.

94. Packet matching fields from the OpenFlow 1.0 Specifications are shown below: What do `src`, `dest`, `dl`, `nw`, and `tp` refer to? Contrast `in` and `src`.

```
// Fields to match against flows
struct ofp_match
{
uint32_t wildcards;           // Wildcard fields
uint16_t in_port;             // Input switch port
uint8_t dl_src[OFP_ETH_ALEN]; // Ethernet source address
uint8_t dl_dst[OFP_ETH_ALEN]; // Ethernet destination address
uint16_t dl_vlan;             // Input VLAN id
uint8_t dl_vlan_pcp;          // Input VLAN priority
uint8_t pad1[1];              // Align to 64-bits
uint16_t dl_type;             // Ethernet frame type.
uint8_t nw_tos;               // IP ToS (actually DSCP field, 6 bits)
uint8_t nw_proto;             // IP protocol or lower 8 bits of ARP opcode
uint8_t pad2[2];              // Align to 64-bits
uint32_t nw_src;              // IP source address
uint32_t nw_dst;              // IP destination address
uint16_t tp_src;              // TCP/UDP source port
uint16_t tp_dst;              // TCP/UDP destination port
};
OFP_ASSERT(sizeof(struct ofp_match) == 40); // bytes
```

95. ** Recall the fields that can be matched in OpenFlow 1.0 at each of the Internet protocol layers.

L4 (to L5) TCP/UDP ports (source and destination)

L3 (to L4) IP protocol field

L3 IP addresses (source and destination)

L3 IP type of service field

L2 MAC/physical addresses (source and destination)

96. Describe typical **counters** in OpenFlow tables.
97. Associate the various types of **actions** in the OpenFlow protocol with traditional network applications.

Forward to output port (or flooding) – destination-based forwarding at L2 or L3; load-balancing

Forward to controller – deep packet inspection

Drop i.e. "no action" – fire-wall

Modify field(s) (any except the IP protocol field) – NAT

98. Which fields are not available for matching in OpenFlow 1.0? Why not?

TTL, datagram length, header checksum (which depends on TTL); presumably the designers thought these weren't useful enough to warrant the additional complexity

Textbook Exercises

Please see P19–P22 on KR-397.

5.5. SDN: Control Plane

99. What is a **middlebox**? Identify or describe examples?

Middlebox *computer networking device that transforms, inspects, filters, or otherwise manipulates traffic for purposes other than packet forwarding* — wikipedia.org

Examples firewalls, NAT translators, load balancers, intrusion detection systems (deep packet inspection), WAN optimizers

100. What distinguishes SDN from "traditional" routing?

Complete unbundling of network functionality [KS-436]:

Generalized forwarding based on fields in L2–L4

Separation (1) of control- and data planes: Forwarding devices do not orchestrate routing

Separation (2) of **controller** (**network OS**) and network control applications

Network control applications programmable by operators – not just vendors

NB: Routing is just one of multiple network control applications.

101. Has unbundling has been beneficial for consumers in another market? Discuss.

In computing – unbundling of {hardware, operating system, applications} seems to have reduced costs and accelerated innovation in

102. ** In what sense might we view (SDN) packet switches as simpler than traditional routers? As more complex?

Simpler No routing capabilities

More complex matching and actions

103. Identify or describe the layers of the SDN **control plane**:

Northbound API for network control applications – typically ReSTful

Network-wide state management layer

Communication layer with **southbound** interface

104. What kind of "state" is kept in the state management layer?

Node info (hosts, switches), link info, flow tables, statistics

105. Where does OpenFlow fit into the SDN control plane?

Right at the bottom – it's a southbound interface

106. Describe OpenFlow's controller-to-switch message types.

features query

configuration parameters query/set

flow table query/set flow table entries

counters query

send packet out of switch port

107. Describe OpenFlow's switch-to-controller message types.

modifed flow table notification (response or after time-out)

port status notify of change

send packet for processing (action or "no-match")

5.6. ICMP

108. What is the role of ICMP? How does it relate to TCP?

- Specialized communication between hosts/routers
- ICMP messages are carried by TCP;
- it has an IP upper-layer protocol number of 1

109. Does ICMP only report errors? Discuss.

No! In addition to errors (incl. "bad header"), ICMP messages cover:
echo request/reply, **source quench** (congestion control), **router** advertisement/discovery, **TTL** expired

110. How are these messages specified?

Two integer values – so-called `type` (0–12) and `code` (0, 1, 2, ...)
e.g. all errors have `type` 3

111. How does `ping` work?

Echo request `type` 0, `code` 0
Echo response `type` 8, `code` 0

112. How does `traceroute` work?

(a) Sent UDP/ICMP message to destination
(b) with unused port number
(c) and TTL of 1;
(d) Repeat with incremented TTL with each "TTL expired" response (11/0)
(e) Stop with receipt of "destination port unreachable" (3/3)

113. The TTL-based implementation of `traceroute` seems inefficient: Wouldn't a dedicated "traceroute message type save on echo requests that differ only in TTL?

Yes: Presumably the designers noted that `traceroute` not performance-critical and preferred a simpler protocol.
Furthermore, the current scheme does provide for more accurate RTT estimates

5.7. Network Management and SNMP

114. Identify the key components of network management and given an example or description of each.

Managing server an application (with human supervisor)

Managed device hardware (e.g. host, router, middlebox, IoT device) or software

Managed objects on/within managed device: hardware (e.g. NIC) and software (e.g. routing agent)

Managment Information Base (MIB): features, configuration parameters, counters, state/errors

Managment agent running on each managed device

Management protocol for communication between managing server and managed devices e.g. **SNMP**

115. Describe the types of SNMP messages.

Get request value(s) from MIB object(s)

Set request value(s) to MIB object(s)

Response to request, with MIB object identifiers and values

Trap message – asynchronous notification of errors/events

116. Which protocol carries SNMP messages? What are the implications?

- **UDP** $\therefore$ no guarantee of reliable delivery
- SNMP doesn't specify if/how transmission errors should be handled

117. What is the latest version of SNMP? What features did it introduce?

Version 3; *additional **security** and administration capabilities*

Review

118. Which routing protocols have we discussed? How might they be classified?

Type	Examples
(Link-State) (LS)	OSPF, IS-IS
Distance-Vector] (DV)	RIP
Path-Vector	BGP
interior gateway	IGRP, IS-IS, OPFS, RIP
exterior gateway	BGP, EGP
classful	IGRP, EGP, RIPv1
classless	BGP, EIGRP, IS-IS, OSPF, RIPv2 (wikipedia.org)

Classless means "transmits **subnet mask** with address"