

- Hubs vs. Switches vs. Routers -

Layered Communication

Network communication models are generally organized into **layers**. The **OSI model** specifically consists of **seven layers**, with each layer representing a specific networking function. These functions are controlled by **protocols**, which govern end-to-end communication between devices.

As data is passed from the user application down the virtual layers of the OSI model, each of the lower layers adds a **header** (and sometimes a **trailer**) containing protocol information specific to that layer. These headers are called **Protocol Data Units (PDUs)**, and the process of adding these headers is referred to as **encapsulation**.

The PDU of each lower layer is identified with a unique term:

#	<i>Layer</i>	<i>PDU Name</i>
7	Application	-
6	Presentation	-
5	Session	-
4	Transport	Segments
3	Network	Packets
2	Data-link	Frames
1	Physical	Bits

Commonly, network devices are identified by the OSI layer they *operate* at (or, more specifically, what *header* or *PDU* the device processes).

For example, **switches** are generally identified as Layer-2 devices, as switches process information stored in the **Data-Link** header of a frame (such as MAC addresses in Ethernet). Similarly, **routers** are identified as Layer-3 devices, as routers process *logical* addressing information in the **Network** header of a packet (such as IP addresses).

However, the strict definitions of the terms *switch* and *router* have blurred over time, which can result in confusion. For example, the term *switch* can now refer to devices that operate at layers higher than Layer-2. This will be explained in greater detail in this guide.

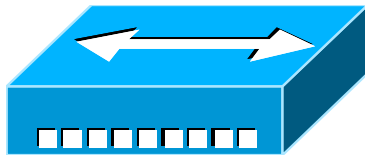
* * *

All original material copyright © 2011 by Aaron Balchunas (aaron@routeralley.com),
unless otherwise noted. All other material copyright © of their respective owners.

This material may be copied and used freely, but may not be altered or sold without the expressed written consent of the owner of the above copyright. Updated material may be found at <http://www.routeralley.com>.

Icons for Network Devices

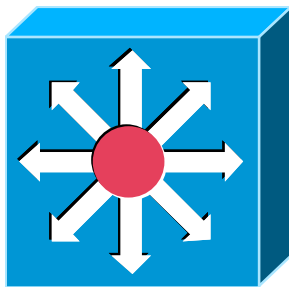
The following icons will be used to represent network devices for all guides on routeralley.com:



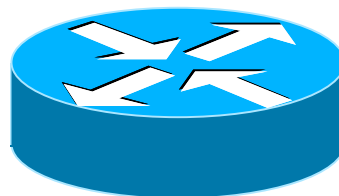
Hub



Switch



Multilayer Switch



Router

All original material copyright © 2011 by Aaron Balchunas (aaron@routeralley.com), unless otherwise noted. All other material copyright © of their respective owners. This material may be copied and used freely, but may not be altered or sold without the expressed written consent of the owner of the above copyright. Updated material may be found at <http://www.routeralley.com>.

Layer-1 Hubs

Hubs are Layer-1 devices that physically connect network devices together for communication. Hubs can also be referred to as **repeaters**.

Hubs provide *no intelligent forwarding* whatsoever. Hubs are incapable of processing either Layer-2 or Layer-3 information, and thus cannot make decisions based on hardware or logical addressing.

Thus, hubs will always forward *every* frame out *every* port, excluding the port originating the frame. Hubs do not differentiate between frame types, and thus will always forward unicasts, multicasts, and broadcasts out *every* port but the originating port.

Ethernet hubs operate at **half-duplex**, which allows a device to either transmit or receive data, but not simultaneously. Ethernet utilizes **Carrier Sense Multiple Access with Collision Detect (CSMA/CD)** to control media access. Host devices monitor the physical link, and will only transmit a frame if the link is idle.

However, if two devices transmit a frame simultaneously, a **collision** will occur. If a collision is detected, the hub will discard the frames and signal the host devices. Both devices will wait a random amount of time before resending their respective frames.

Remember, if *any* two devices connected to a hub send a frame simultaneously, a collision *will* occur. Thus, all ports on a hub belong to the same **collision domain**. A collision domain is simply defined as any physical segment where a collision can occur.

Multiple hubs that are uplinked together still all belong to *one* collision domain. Increasing the number of host devices in a single collision domain will increase the number of collisions, which can significantly degrade performance.

Hubs also belong to only one **broadcast domain** – a hub will forward both broadcasts and multicasts out *every* port but the originating port. A broadcast domain is a logical segmentation of a network, dictating how far a broadcast (or multicast) frame can propagate.

Only a Layer-3 device, such as a router, can separate broadcast domains.

* * *

All original material copyright © 2011 by Aaron Balchunas (aaron@routeralley.com), unless otherwise noted. All other material copyright © of their respective owners.

This material may be copied and used freely, but may not be altered or sold without the expressed written consent of the owner of the above copyright. Updated material may be found at <http://www.routeralley.com>.

Layer-2 Switching

Layer-2 devices build **hardware address tables**, which will contain the following at a minimum:

- Hardware addresses for host devices
- The port each hardware address is associated with

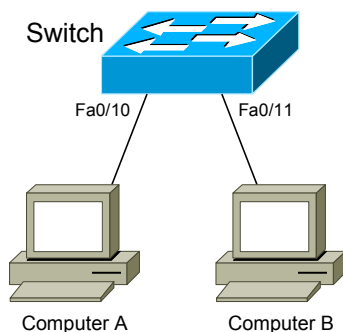
Using this information, Layer-2 devices will make intelligent forwarding decisions based on frame (Data-Link) headers. A frame can then be forwarded out *only* the appropriate destination port, instead of *all* ports.

Layer-2 forwarding was originally referred to as **bridging**. Bridging is a largely deprecated term (mostly for marketing purposes), and Layer-2 forwarding is now commonly referred to as **switching**.

There are some subtle technological differences between *bridging* and *switching*. Switches usually have a higher port-density, and can perform forwarding decisions at wire speed, due to specialized hardware circuits called **ASICs (Application-Specific Integrated Circuits)**. Otherwise, bridges and switches are nearly identical in function.

Ethernet switches build **MAC-address tables** through a dynamic learning process. A switch behaves much like a hub when first powered on. The switch will flood every frame, including unicasts, out *every* port but the originating port.

The switch will then build the MAC-address table by examining the **source MAC address** of each frame. Consider the following diagram:



When ComputerA sends a frame to ComputerB, the switch will add *ComputerA's* MAC address to its table, associating it with port fa0/10. However, the switch will not learn *ComputerB's* MAC address until ComputerB sends a frame to ComputerA, or to another device connected to the switch. Switches **always learn from the source MAC address**.

A switch is in a perpetual state of learning. However, as the MAC-address table becomes populated, the flooding of frames will decrease, allowing the switch to perform more efficient forwarding decisions.

* * *

All original material copyright © 2011 by Aaron Balchunas (aaron@routeralley.com), unless otherwise noted. All other material copyright © of their respective owners.

This material may be copied and used freely, but may not be altered or sold without the expressed written consent of the owner of the above copyright. Updated material may be found at <http://www.routeralley.com>.

Layer-2 Switching (continued)

While hubs were limited to half-duplex communication, switches can operate in **full duplex**. *Each individual port* on a switch belongs to its *own collision domain*. Thus, switches create **more collision domains**, which results in **fewer collisions**.

Like hubs though, switches belong to only *one broadcast domain*. A Layer-2 switch will forward both broadcasts and multicasts out *every port* but the originating port. Only Layer-3 devices separate broadcast domains.

Because of this, Layer-2 switches are poorly suited for large, scalable networks. The Layer-2 header provides no mechanism to differentiate one *network* from another, only one *host* from another.

This poses *significant* difficulties. If *only* hardware addressing existed, all devices would technically be on the *same* network. Modern internetworks like the Internet could not exist, as it would be impossible to separate *my* network from *your* network.

Imagine if the entire Internet existed purely as a Layer-2 switched environment. Switches, as a rule, will forward a broadcast out *every port*. Even with a conservative estimate of a billion devices on the Internet, the resulting broadcast storms would be devastating. The Internet would simply collapse.

Both hubs and switches are susceptible to **switching loops**, which result in destructive broadcast storms. Switches utilize the **Spanning Tree Protocol (STP)** to maintain a loop-free environment. STP is covered in great detail in another guide.

Remember, there are three things that switches do that hubs do not:

- **Hardware address learning**
- **Intelligent forwarding of frames**
- **Loop avoidance**

Hubs are almost entirely deprecated – there is no advantage to using a hub over a switch. At one time, switches were more expensive and introduced more latency (due to processing overhead) than hubs, but this is no longer the case.

* * *

All original material copyright © 2011 by Aaron Balchunas (aaron@routeralley.com), unless otherwise noted. All other material copyright © of their respective owners.

This material may be copied and used freely, but may not be altered or sold without the expressed written consent of the owner of the above copyright. Updated material may be found at <http://www.routeralley.com>.

Layer-2 Forwarding Methods

Switches support three **methods** of forwarding frames. Each method copies all or part of the frame into memory, providing different levels of latency and reliability. **Latency** is *delay* - less latency results in quicker forwarding.

The **Store-and-Forward** method copies the *entire* frame into memory, and performs a Cycle Redundancy Check (CRC) to completely ensure the integrity of the frame. However, this level of error-checking introduces the highest latency of any of the switching methods.

The **Cut-Through (Real Time)** method copies only enough of a frame's header to determine its destination address. This is generally the *first 6 bytes* following the preamble. This method allows frames to be transferred at *wire speed*, and has the least latency of any of the three methods. No error checking is attempted when using the cut-through method.

The **Fragment-Free (Modified Cut-Through)** method copies only the *first 64 bytes* of a frame for error-checking purposes. Most collisions or corruption occur in the first 64 bytes of a frame. Fragment-Free represents a compromise between reliability (store-and-forward) and speed (cut-through).

* * *

All original material copyright © 2011 by Aaron Balchunas (aaron@routeralley.com), unless otherwise noted. All other material copyright © of their respective owners.

This material may be copied and used freely, but may not be altered or sold without the expressed written consent of the owner of the above copyright. Updated material may be found at <http://www.routeralley.com>.

Layer-3 Routing

Layer-3 **routing** is the process of forwarding a packet from one *network* to another *network*, based on the Network-layer header. Routers build **routing tables** to perform forwarding decisions, which contain the following:

- The destination network and subnet mask
- The **next hop** router to get to the destination network
- Routing *metrics* and Administrative Distance

Note that Layer-3 forwarding is based on the destination *network*, and not the destination *host*. It is possible to have *host routes*, but this is less common.

The routing table is concerned with two types of Layer-3 protocols:

- **Routed protocols** - assigns logical addressing to devices, and routes packets between networks. Examples include IP and IPX.
- **Routing protocols** - dynamically builds the information in routing tables. Examples include RIP, EIGRP, and OSPF.

Each individual interface on a router belongs to its *own collision domain*. Thus, like switches, routers create **more collision domains**, which results in **fewer collisions**.

Unlike Layer-2 switches, Layer-3 routers also **separate broadcast domains**. As a rule, a router **will never forward broadcasts** from one network to another network (unless, of course, you explicitly configure it to). ☺

Routers will not forward multicasts either, unless configured to participate in a multicast tree. Multicast is covered in great detail in another guide.

Traditionally, a router was required to copy each individual packet to its buffers, and perform a route-table lookup. Each packet consumed CPU cycles as it was forwarded by the router, resulting in latency. Thus, routing was generally considered **slower** than switching.

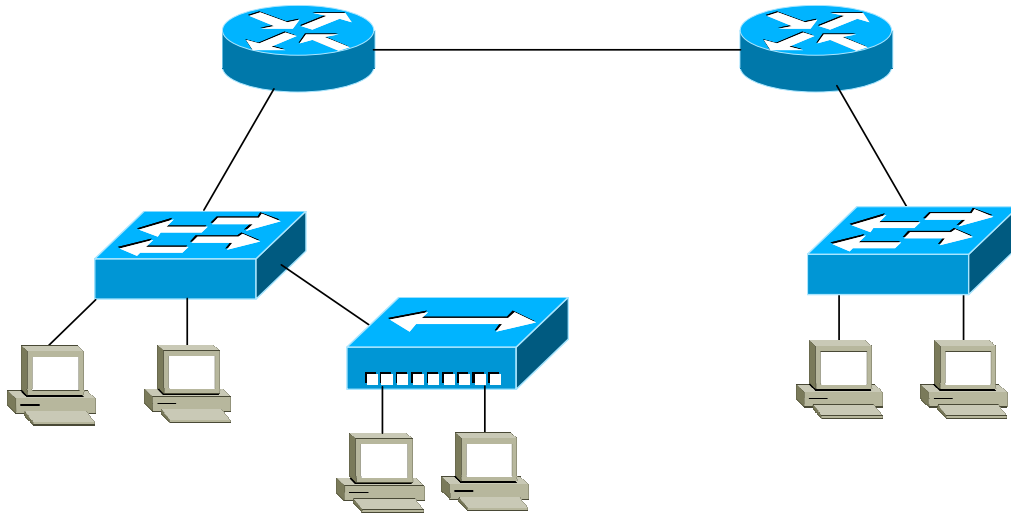
It is now possible for routers to *cache* network-layer flows in hardware, greatly reducing latency. This has blurred the line between *routing* and *switching*, from both a technological and marketing standpoint. Caching network flows is covered in greater detail shortly.

* * *

All original material copyright © 2011 by Aaron Balchunas (aaron@routeralley.com), unless otherwise noted. All other material copyright © of their respective owners.

This material may be copied and used freely, but may not be altered or sold without the expressed written consent of the owner of the above copyright. Updated material may be found at <http://www.routeralley.com>.

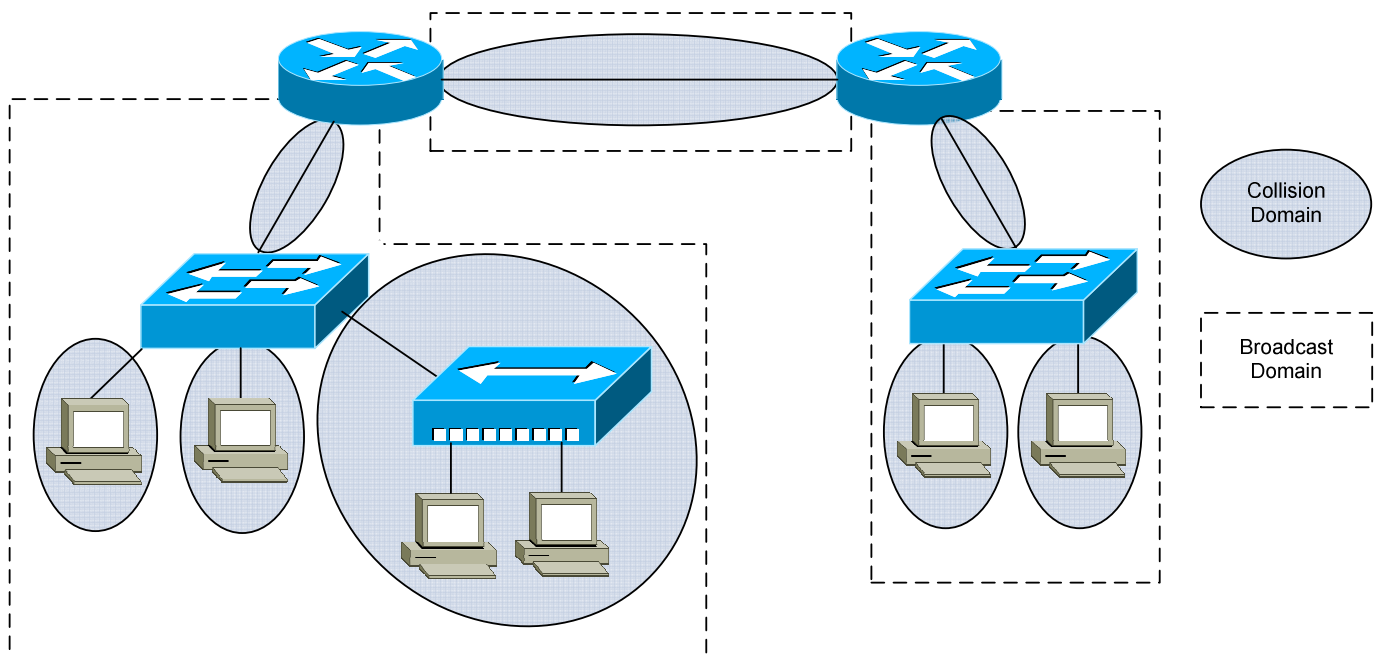
Collision vs. Broadcast Domain Example



Consider the above diagram. Remember that:

- Routers separate *broadcast* and *collision* domains.
- Switches separate *collision* domains.
- Hubs belong to only one *collision* domain.
- Switches and hubs both only belong to one *broadcast* domain.

In the above example, there are **THREE** broadcast domains, and **EIGHT** collision domains:



VLANs – A Layer-2 or Layer-3 Function?

By default, a switch will forward both broadcasts and multicasts out *every* port but the originating port.

However, a switch can be logically segmented into multiple broadcast domains, using **Virtual LANs** (or **VLANs**). VLANs are covered in extensive detail in another guide.

Each VLAN represents a unique broadcast domain:

- Traffic between devices within the *same* VLAN is switched (forwarded at Layer-2).
- Traffic between devices in *different* VLANs requires a Layer-3 device to communicate.

Broadcasts from one VLAN will not be forwarded to another VLAN. This separation provided by VLANs is **not a Layer-3 function**. VLAN tags are inserted into the **Layer-2 header**.

Thus, a switch that supports VLANs is not necessarily a Layer-3 switch. However, a purely Layer-2 switch cannot route between VLANs.

Remember, though VLANs provide separation for *Layer-3* broadcast domains, and are often associated with IP subnets, they are still a *Layer-2* function.

* * *

All original material copyright © 2011 by Aaron Balchunas (aaron@routeralley.com), unless otherwise noted. All other material copyright © of their respective owners.

This material may be copied and used freely, but may not be altered or sold without the expressed written consent of the owner of the above copyright. Updated material may be found at <http://www.routeralley.com>.

Layer-3 Switching

In addition to performing Layer-2 switching functions, a **Layer-3 switch** must also meet the following criteria:

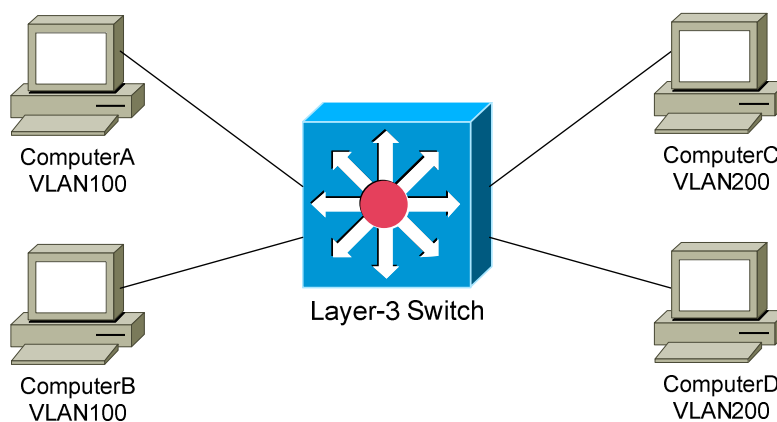
- The switch must be capable of making Layer-3 forwarding decisions (traditionally referred to as routing).
- The switch must cache network traffic flows, so that Layer-3 forwarding can occur in hardware.

Many older modular switches support Layer-3 route processors – this alone does not qualify as Layer-3 switching. Layer-2 and Layer-3 processors can act independently within a single switch chassis, with each packet requiring a route-table lookup on the route processor.

Layer-3 switches leverage ASICs to perform Layer-3 forwarding in hardware. For the first packet of a particular traffic flow, the Layer-3 switch will perform a standard route-table lookup. This flow is then *cached* in hardware – which preserves required routing information, such as the destination network and the MAC address of the corresponding next-hop.

Subsequent packets of that flow will bypass the route-table lookup, and will be forwarded based on the cached information, reducing latency. This concept is known as *route once, switch many*.

Layer-3 switches are predominantly used to route between VLANs:



Traffic between devices within the same VLAN, such as ComputerA and ComputerB, is *switched* at Layer-2 as normal. The first packet between devices in different VLANs, such as ComputerA and ComputerD, is *routed*. The switch will then cache that IP traffic flow, and subsequent packets in that flow will be *switched* in hardware.

* * *

All original material copyright © 2011 by Aaron Balchunas (aaron@routeralley.com), unless otherwise noted. All other material copyright © of their respective owners.

This material may be copied and used freely, but may not be altered or sold without the expressed written consent of the owner of the above copyright. Updated material may be found at <http://www.routeralley.com>.

Layer-3 Switching vs. Routing – End the Confusion!

The evolution of network technologies has led to considerable confusion over the terms *switch* and *router*. Remember the following:

- The traditional definition of a *switch* is a device that performs Layer-2 forwarding decisions.
- The traditional definition of a *router* is a device that performs Layer-3 forwarding decisions.

Remember also that, switching functions were typically performed in *hardware*, and routing functions were typically performed in *software*. This resulted in a widespread perception that switching was *fast*, and routing was *slow* (and *expensive*).

Once Layer-3 forwarding became available in hardware, marketing gurus muddied the waters by distancing themselves from the term *router*. Though Layer-3 forwarding in hardware is still *routing* in every technical sense, such devices were rebranded as Layer-3 switches.

Ignore the marketing noise. **A Layer-3 switch is still a router.**

Compounding matters further, most devices still currently referred to as *routers* can perform Layer-3 forwarding in hardware as well. Thus, both Layer-3 switches *and* Layer-3 routers perform nearly identical functions at the same performance.

There are some differences in *implementation* between Layer-3 switches and routers, including (but not limited to):

- Layer-3 switches are optimized for Ethernet, and are predominantly used for inter-VLAN routing. Layer-3 switches can also provide Layer-2 functionality for intra-VLAN traffic.
- Switches generally have higher port densities than routers, and are considerably cheaper per port than routers (for Ethernet, at least).
- Routers support a large number of WAN technologies, while Layer-3 switches generally do not.
- Routers generally support more advanced feature sets.

Layer-3 switches are often deployed as the backbone of LAN or campus networks. Routers are predominantly used on network perimeters, connecting to WAN environments.

(Fantastic Reference: <http://blog.ioshints.info/2011/02/how-did-we-ever-get-into-this-switching.html>)

* * *

All original material copyright © 2011 by Aaron Balchunas (aaron@routeralley.com), unless otherwise noted. All other material copyright © of their respective owners.

This material may be copied and used freely, but may not be altered or sold without the expressed written consent of the owner of the above copyright. Updated material may be found at <http://www.routeralley.com>.

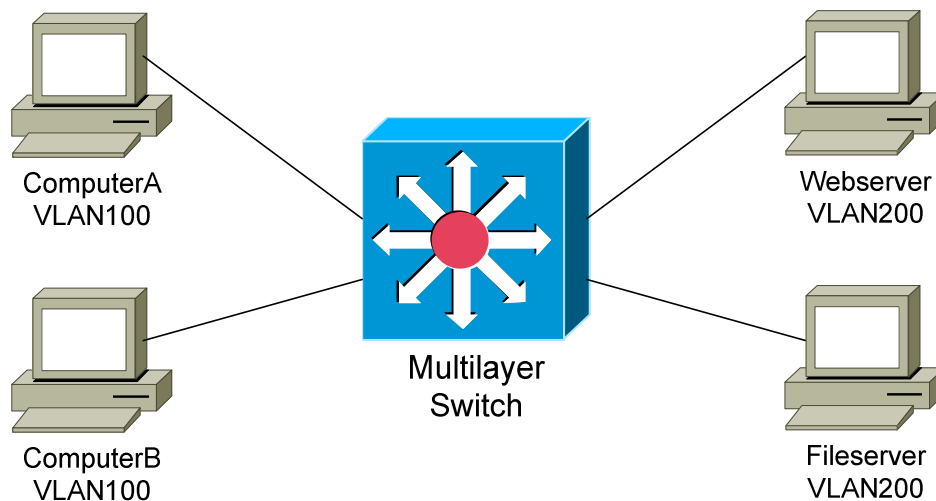
Multilayer Switching

Multilayer switching is a generic term, referring to any switch that forwards traffic at layers higher than Layer-2. Thus, a Layer-3 switch is considered a multilayer switch, as it forwards frames at Layer-2 and packets at Layer-3.

A **Layer-4 switch** provides the same functionality as a Layer-3 switch, but will additionally examine and cache **Transport-layer application flow** information, such as the TCP or UDP port.

By caching application flows, **QoS (Quality of Service)** functions can be applied to preferred applications.

Consider the following example:



Network and application traffic flows from ComputerA to the Webserver and Fileserver will be cached. If the traffic to the Webserver is preferred, then a higher QoS priority can be assigned to that application flow.

Some advanced multilayer switches can provide load balancing, content management, and other application-level services. These switches are sometimes referred to as **Layer-7 switches**.

* * *

All original material copyright © 2011 by Aaron Balchunas (aaron@routeralley.com), unless otherwise noted. All other material copyright © of their respective owners.

This material may be copied and used freely, but may not be altered or sold without the expressed written consent of the owner of the above copyright. Updated material may be found at <http://www.routeralley.com>.