**Chapter 4: Advanced DataPower Networking**

Add a note here[Chapter 3](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=308" \l "308" \t "_parent), [“DataPower as a Network Device,”](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=308#308) presented the basic steps for configuring the DataPower appliance’s network stack. But what do all of the steps in [Chapter 3](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=308#308) actually mean? If you are new to networking or if you want more detail on how the internals of the device work, this chapter is for you! This is a detailed chapter with a lot of background information, because the complex topics covered cannot be explained without providing a detailed context. Stick with it, and the rewards will be great.

**Add a note here****First, Some Theory**

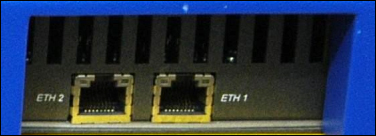
Add a note hereSince the dawn of time (which is even earlier than January 1, 1970 at 12:00 a.m., as the UNIX people would have us believe), computers have been networked together. In fact, much like the first UNICS system (the precursor of UNIX), the first “modern network” predates the UNIX zero time point. The creation in 1969 of the ARPANET (named after the U.S. Department of Defense’s Advanced Research Projects Agency) was the evolution of the first modern packet switched network, using various types of network protocols before standardizing on the ITU-T X.25 protocol over Network Control Protocol (NCP) at heady speeds of up to 50kbps.

Add a note hereIn 1974, NCP was replaced by a more robust protocol called Transmission Control Protocol (TCP), which in 1978 was split into two separate protocols; TCP was redefined as a “higher level” protocol to provide reliable end-to-end communication, while the parts that dealt with packet routing were rolled up into a new protocol called Internet Protocol (IP). It is because of this joint heritage, where both were originally part of a single protocol, that the suite of protocols together is known as TCP/IP—the protocols with which our modern Internet and all devices connected to it communicate.

Add a note hereThis chapter provides some detail about how TCP/IP works. It is framed in a context in which the DataPower-specific issues are presented and as a backdrop for the knowledge contained in the other chapters of the book. For a more advanced reference, we suggest readers use the TCP/IP bible, *TCP/IP Illustrated: Volume 1: The Protocols* by Richard Stevens (Boston, MA: Addison-Wesley: 1994).

**Add a note here****Terminology**

Add a note hereSOA appliances would be fundamentally useless if they could not communicate. DataPower appliances use Ethernet networking and the common networking formats and protocols of the Internet. Figure 4-1 shows two of the network ports of an XI50 device, standard CAT5 RJ45 Ethernet connections. But what does that actually mean?

  
Add a note hereFigure 4-1: RJ45 CAT5 Ethernet.

Add a note hereThe devices are complex in their use of networking, and to be able to properly appreciate and use them to their fullest potential, you must understand some networking technology. To explain the networking concepts, a common terminology and an explanation of the Internet are necessary.

Add a note hereLet’s start by defining a few basic concepts. A *packet* is simply a block of data; it is nothing more and nothing less. It makes no judgment on what that data is; there are no “better” or “worse” packets, and there is no inherent specific definition that determines what size a packet should be, where or how it should be processed, or what to do with it. Of course, just as with anything in IT, for packets to be of use, people need to agree on the standards for using them, which we discuss later in this chapter.

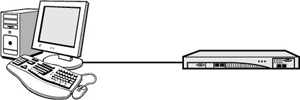
Add a note hereThese blocks of data are sent between separate *nodes* on a *network*. A network simply refers to a group of nodes that can communicate by sending packets to each other. A node is any device attached to the network that might want to share information; that is, to send and receive packets. A node can be everything from computers to routers to personal digital assistants to mobile telephones and even refrigerators. (It is true that a refrigerator can be a node on a network. Google it to learn more!) The nodes may all be similar or they might be different; there are no set rules, except that they are all nodes and are connected to the same network.

**Point-to-Point Connections**

Add a note hereAssume that there are two nodes that want to send packets of data to each other. The simplest form of network is a direct link between the two nodes; this is generally referred to as a *point-to-point* connection. This can be as simple as two computers with a null-modem cable joining them or a crossover Ethernet link. As long as the two nodes have agreed on the standards to use and the language they send, they will be able to communicate with each other. Sending data is simple; there are exactly two nodes, so when one node wants to send a packet of data, the only possible target to send it to is the other node (referred to as the “peer”). There doesn’t need to be a concept of “addressing”—each node understands only “me” and “the peer.”

Add a note hereA point-to-point connection is the technical equivalent of throwing a letter over your next-door neighbor’s fence. The letter goes directly to them; it could not have come from anyone but you, and when they want to reply, they just need to throw a letter back over the same fence. (Some people actually have neighbors like this and end up moving.)

Add a note hereThe serial link used to connect a workstation to the DataPower device for initial configuration and direct console access, shown in [Chapter 2](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=145#145), [“DataPower Quick Tour and Setup”](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=145#145) and depicted in Figure 4-2, is an example of a point-to-point connection. There is no other node on this connection, and to impersonate one of the nodes, an attacker would have to physically reconnect the link.

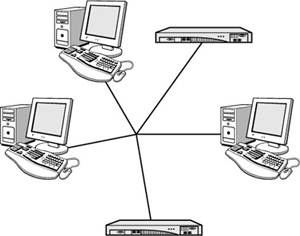
  
Add a note hereFigure 4-2: A serial link from a workstation to DataPower is a point-to-point connection.

**Broadcast Networks**

Add a note hereOf course the directly linked network is useful in some ways, but for networking, perhaps it might be of more value for more than two nodes to be involved. The simplest way to do this is to connect all the nodes to the same physical medium and let them communicate, in what is known as a *broadcast network*. An example of a shared physical medium in this context might be a hub to which all nodes are connected. When a node wants to send a packet of data to one of the other nodes, it puts the packet out onto the network. Because the medium is shared, all of the nodes can see all of the packets of data. This means that they now have to agree on a mechanism to be able to explain who the data is meant for; when node A wants to send data to node C, it needs to be able to say “this packet is destined for node C” so that node C knows to pick it up and use it.

Add a note hereAn analogy of a broadcast network in the real world would be a system of pigeonholes in a university. These are like a set of open mail boxes, often used in environments around the world where everyone is considered trusted. If a young student, Alice, needs to send a note to her professor, Bob, she goes to the campus post office and places the note in his pigeonhole. Anyone who goes into the post office can see that there is a note for Professor Bob. Indeed, if they were rude, they could take the note, read it, and put it back, and neither Professor Bob nor Alice would ever know that they did. But the note is clearly intended for Professor Bob because it is in his pigeonhole; this is a rudimentary form of addressing.

Add a note hereFigure 4-3 shows an example of a broadcast network; all the nodes are connected to the same physical network, and they all can communicate with each other.

  
Add a note hereFigure 4-3: A broadcast network with locally connected nodes.

Add a note hereIn this example, because all the nodes are linked together, if one of the nodes needs to contact one of the DataPower devices, it would simply send a packet onto the network using various protocols. (These protocols are described later in this chapter.) The key difference between a broadcast network and a point-to-point connection is that in the broadcast network, all other nodes sharing the same physical medium can see the data!

**Tip: Sensitive Connections**

Add a note hereJust as the most sensitive of routers do not accept administrative connections directly over their network interfaces, but via some form of secure console device, so can DataPower be configured to share its administrative connections only via its serial port. A point-to-point connection with a trusted device will always be more secure than a management network—the less machines involved in a network, the easier it is to secure them!

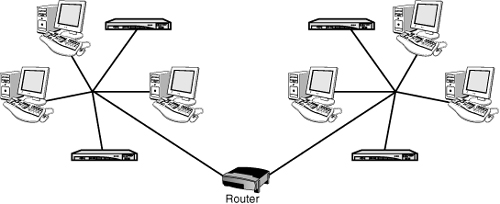
**Routed Networks**

Add a note hereWhat if you can’t physically link all the nodes together? When this is the case, you have to move on to a concept known as [*routing*](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=554#554). A given machine on a local broadcast network is given the role of taking packets that are not addressed to a local node and forwarding them to other nodes on other networks until the packets find the right node; this machine is known as a *router*. The packets are passed from one router to another until they reach their destinations.

Add a note hereThis is similar to what happens when you mail a letter using the British Royal Mail (and likely, the United States Postal Service). Let’s say Professor Bob has to consult about Alice’s question with his colleague, Dr. Charlie, who works at another university. That university also has a post office with a pigeonhole system, but it is too much hassle for Professor Bob to drive over to Dr. Charlie’s university just to put a note in his pigeonhole! So, he writes the address of Dr. Charlie’s university on the letter and makes it clear that the letter is destined for Dr. Charlie. He then puts it in the pigeonhole destined for outgoing mail.

Add a note hereAt some point, someone will take the letter out of the pigeonhole and deliver it to the local post office. At the post office, they will see that it is destined for Dr. Charlie’s university, and they will load the letter onto a truck or van, along with any other letters and packages that happen to be going in that general direction. In this way, the letter will be passed from one truck to another, until (if it is lucky enough not to be lost), it reaches the post office local to Dr. Charlie’s university. There it will be collected together along with any other mail for Dr. Charlie’s university and delivered to the campus post office, where eventually someone will sort through the incoming mail and place this one in Dr. Charlie’s pigeonhole. The letter will finally have been routed to its destination.

Add a note hereFigure 4-4 shows two discrete physically separate networks connected by a single router; of course, such a network may be made up of many hundreds of networks that are interconnected via routers. The term we use for a complex network such as this is a *packet switched network*; this simply refers to routing discrete blocks of data (packets) from one node to another over data links shared with other traffic.

[](javascript:PopImage('IMG_46','http://images.books24x7.com/bookimages/id_30903/04fig04_alt.jpg','545','222'))  
Add a note hereFigure 4-4: Two broadcast networks are connected via a router.

Add a note hereIn this routed network, if a node on one of the networks wants to connect to the DataPower appliance on the other side, it has to send the packets to the router, which then relays them on to the final destination. The appliance has an internal routing table, just like any network device, which defines how it will send packets on to routers for forwarding to other nodes; this is a complex subject. Routing is discussed in more detail later in this chapter.

Add a note hereIn summary, this is the terminology you should understand. Blocks of data called packets are sent between different nodes on a network, of which there are three main types: point-to-point, which is a direct connection between exactly two nodes; broadcast, which is when two or more nodes send packets to each other via a shared physical medium; and routed, where packets between two or more physically separate broadcast networks are sent between each other via intermediaries named routers.

**Add a note here****Abstractions**

Add a note hereIn order for computers to communicate with each other, it is vital that they agree on a form of protocol, that is, a common language that they can all speak. In the same way as two people from vastly different countries and cultures cannot easily communicate unless they share some form of language or reference point, two computers will be unable to share data unless they both know in advance what the bits and bytes will mean. Standards are required at many levels.

Add a note hereIn 1982 the International Standards Organization (ISO), along with many industry players, created something called the Open Systems Interconnection (OSI) initiative. One of the outputs of this initiative was a common model for how to define layers of protocols that clearly describe interoperability between network devices and software.

Add a note hereThe OSI model is comprised of seven layers. In this section, we discuss the first four of these, which deal with the network transports themselves; the others are application-level issues and are the subject of the rest of this book. The layers are shown in Table 4-1.

| Add a note hereTable 4-1: The OSI Layers  [[http://www.books24x7.com/images/b24-bluearrow.gif](http://www.books24x7.com/outputobject.asp?bookid=30903&chunkid=349370393&objectid=ch04table01&objecttype=spreadsheet)Open table as spreadsheet](http://www.books24x7.com/outputobject.asp?bookid=30903&chunkid=349370393&objectid=ch04table01&objecttype=spreadsheet) | | |
| --- | --- | --- |
| **Add a note hereLayer** | **Add a note hereName** | **Add a note hereFunction** |
| Add a note here7 | Add a note hereApplication | Add a note hereApplication service communication |
| Add a note here6 | Add a note herePresentation | Add a note hereData representation; encryption |
| Add a note here5 | Add a note hereSession | Add a note hereInter-host communication |
| Add a note here4 | Add a note hereTransport | Add a note hereEnd-to-end connections; reliability |
| Add a note here3 | Add a note hereNetwork | Add a note hereLogical addressing; routing |
| Add a note here2 | Add a note hereData Link | Add a note herePhysical addressing |
| Add a note here1 | Add a note herePhysical | Add a note herePhysical media; signaling; binary transmission |

Add a note hereThe first four layers, shown under the thick line in the table, represent the standards and terms agreed on for basic network communications; it is these that DataPower uses as a network device. These are discussed in the following sections.

**The Physical Layer**

Add a note hereIn the physical layer of the OSI model, there are many standards that define exactly how data is sent. These include pinouts, voltages, handshaking, and so on. They define, for instance, what exactly it means for the third pin on a serial connector to be +5v at a given moment. They define which wires in an Ethernet cable are used to send data, and what frequency pulses are sent at. They define when the ring signal should be sent over a 60-volt telephone wire and what frequency range wireless networks should use. Some examples of standards for the physical layer include RS-232 (serial connections), 1000BASE-TX (Gigabit Ethernet physical layer), 802.11b (Wireless Ethernet), and POTS (Plain Old Telephone System).

Add a note hereThe physical layer may seem out of scope for a discussion on XML-based SOA appliances—but you must remember that the appliance is still a network device, and the same kinds of issues that can affect any network device using the same technology still do apply. Indeed, the two points that follow *can and do* cause issues that are not easy to diagnose unless you understand how the network stack of the appliance is built!

Add a note hereFirst, because the appliance’s network interfaces are Gigabit Ethernet capable of using the 802.3ab protocol (1000BASE-TX) for high speeds but also able to utilize 802.3u (100BASE-TX) for slower speeds, it is essential that the correct communication parameters are configured. This may seem trivial—doesn’t this stuff happen automatically? The answer is that most of the time it does, and the appliance is, by default, configured to auto-negotiate transmission rates and duplex modes as required by 1000BASE-TX. But when communicating with other network devices, it is sometimes necessary to force the interface to use a lower level of negotiation (such as 802.3u) or to not negotiate at all but use a fixed communication. [Chapter 3](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=308#308) showed where this can be set.

Add a note hereThe second point has to do with the physical layer of the console interface. This uses the RS-232 standard for serial binary data communication with the device, and for successful communication, the parameters on both sides of the interface must be the same. In the past, some SOA appliances have been mistakenly pronounced dead and prepared for return shipment to IBM. In one case, the real culprit was a laptop reboot that had caused its serial port terminal emulation software to revert to a nonstandard setting. In another, a cable was incorrectly connected with voltages on the wrong pins making communication impossible!

**The Data Link Layer**

Add a note hereOn top of the physical layer, whatever it is, lies the data link layer. This layer includes the important concept of device addressing. Physical devices on a network must be addressable in some way so that packets can be addressed to a given device. This is commonly done by agreeing on a size and format for the packet. Certain sections of the packet are given special meaning; for instance, if a standard is adopted that says that bytes 9 through 15 are the destination address for the packet, all nodes can easily see which packets are destined for which nodes. Addressing in the data link layer refers to actual physical addresses of nodes on a network; because of this, it is not suitable for a routed network. It is suitable only for a broadcast network with a shared physical medium. Examples of standards for the data link layer are PPP (Point-to-Point Protocol) for serial or modem connections, 802.3 (Ethernet packet format), and 802.1Q (VLAN).

Add a note hereMany people do not realize that, in the same way as network layer addresses can be configured dynamically at runtime, so can the physical addresses of most devices. In [Chapter 3](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=308#308), we showed you how to modify the Ethernet (MAC) address of an interface instead of using the built in MAC address. There are often valid reasons to do this. One example might be when swapping out an appliance that has been physically destroyed or otherwise taken out of service; in this case, it may be prudent to configure the new replacement appliance with the MAC address of the old one so that switches and other network hardware do not consider the swap out to be a security violation, negating the requirement to flush ARP tables and so on at the switch.

**Tip: Hardware Addresses**

Add a note hereThe Hardware address for the Ethernet implementation of the data link layer is commonly referred to as a Media Access Control, or MAC, address. You will often see the terms “Hardware address,” “Ethernet address,” and “MAC address” used interchangeably. This is usually valid. Hardware address is more generic and can be used to refer to any technology’s hardware addresses, while MAC address and Ethernet address are only used for Ethernet networks. Whenever we say Hardware address in a DataPower context, we mean the MAC address because this is the networking technology used by DataPower.

**The Network Layer**

Add a note hereOn top of the data link layer lies the network layer. This is where devices can be given an address that is valid for more than just a local broadcast network with a shared physical medium. Of course, the address still has to uniquely identify the node in the network at large. Addressing in the network layer is performed in a similar manner to that of the data layer—all users agree that a certain set of bytes in a specific position in a packet refer to the addresses, and therefore it is obvious where a packet is meant to go. Examples of network layer standards include Internet Protocol (IP), Address Resolution Protocol (ARP), Internet Control Message Protocol (ICMP), and Routing Information Protocol (RIP).

Add a note hereDataPower SOA appliances are an advanced form of network device. As seen in [Chapter 3](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=308#308), there is a detailed level of configuration that can control how the appliance behaves, especially with respect to network layer protocols. This includes ways of responding to ICMP requests using ARP, and aliasing multiple IP addresses onto single physical interfaces.

**The Transport Layer**

Add a note hereFinally, on top of the network layer is the transport layer. The transport layer is responsible for providing an end-to-end connection between two nodes, regardless of where they lie on the network. It may use techniques such as flow and error control, and some more reliable transport protocols will retry packets that fail in order to keep track of connection state. Common standards for transport layer protocols are Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and Point-to-Point Tunneling Protocol (PPTP).

Add a note hereAs its primary transport layer protocol, DataPower primarily uses TCP. All administrative connections and the vast majority of client connections will be TCP. The main exceptions are that NFS can use UDP instead of TCP, and both Syslog and SNMP use UDP.

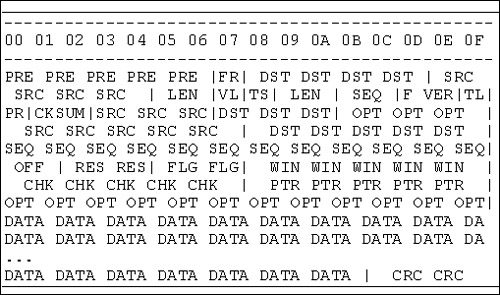
Add a note hereThe abstractions we have described show the four layers of the OSI model applicable to DataPower as a network device. A packet that is being sent over a network from one node to another, while being just a set of data, will need to be formatted specifically in a way that all the layers can understand. Each layer is encapsulated in the previous layer; for instance, a data layer packet has some information (such as physical addresses) and some data; that data will contain a network layer packet that has information (such as logical addresses) and some data, which contains a transport layer packet, and so on. In DataPower, these packets are usually TCP packets encapsulated within IP packets encapsulated within Ethernet packets; the [next section](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=503#503) explores the specific protocols used by the DataPower appliance.

**TCP/IP Primer**

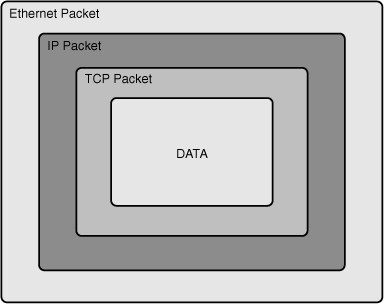
Add a note hereThe DataPower appliance uses a specific set of standards and protocols that are the common standards and protocols of the Internet: Ethernet interfaces, with an IP stack using TCP sockets. (As mentioned earlier, the device also supports UDP for some scenarios where it is relevant to the protocol being used.)

**Add a note here****Packet Structure**

Add a note hereFirst, let’s look at the specific structure of a TCP packet, as shown in Figure 4-5.

[](javascript:PopImage('IMG_47','http://images.books24x7.com/bookimages/id_30903/04fig05_alt.jpg','509','300'))  
Add a note hereFigure 4-5: The structure of a TCP packet.

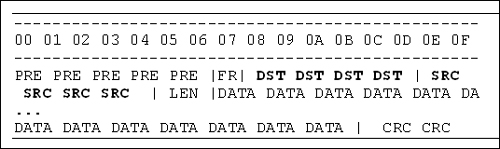
Add a note hereAs Figure 4-5 shows, the data that is being sent is encapsulated within a TCP packet, which is encapsulated as data within an IP packet, which is encapsulated as data within an Ethernet packet, which is sent out on the wire. The representation in Figure 4-6 helps make this clearer.

  
Add a note hereFigure 4-6: Packet encapsulation.

Add a note hereThe data is *inside* a TCP packet, which is *inside* an IP packet, which is *inside* an Ethernet packet. Let’s start by taking a look at each of the packet headers.

**Ethernet Headers**

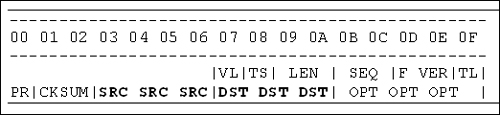
Add a note hereThe first sections, from PRE through to LEN, as shown in Figure 4-7, represent the Ethernet headers, which are the data link layer.

[](javascript:PopImage('IMG_49','http://images.books24x7.com/bookimages/id_30903/04fig07_alt.jpg','536','160'))  
Add a note hereFigure 4-7: The Ethernet headers.

Add a note hereThey include a destination address, DST, and a source address, SRC; these are the “hardware” addresses of the nodes on the network. When this packet is sent out onto a broadcastable Ethernet network (such as a hub), all the computers on that network can see which node the packet was sent from and which node it is meant for. In Ethernet terminology, the name for a hardware address is the Media Access Control Address, or MAC address; DST and SRC here represent the MAC addresses of the destination and source machines.

**IP Headers**

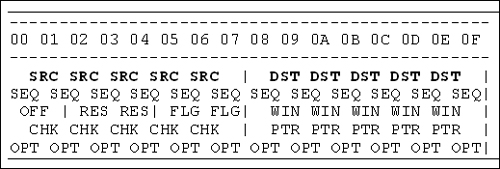
Add a note hereThe second part from VL through to OPT, depicted in Figure 4-8, represents the IP headers.

[](javascript:PopImage('IMG_50','http://images.books24x7.com/bookimages/id_30903/04fig08_alt.jpg','522','120'))  
Add a note hereFigure 4-8: The IP headers.

Add a note hereThese include a destination address, DST, and a source address, SRC. These are the logical or network addresses of the nodes on the network. When this packet is routed through an IP network, the nodes through which it is routed can see which node the packet was sent from and which node it is meant for. The terminology at the network level for IP simply refers to these as IP addresses.

**TCP Headers**

Add a note hereThe third part, as shown in Figure 4-9, from SRC through OPT, represents the TCP headers.

[](javascript:PopImage('IMG_51','http://images.books24x7.com/bookimages/id_30903/04fig09_alt.jpg','519','175'))  
Add a note hereFigure 4-9: The TCP headers.

Add a note hereThe most important fields in the TCP headers are the source port and the destination port, shown as SRC and DST in Figure 4-9. The concept of “port” is a TCP-specific construct for handling sessions between nodes; it provides a metaphor for slots where connections can be made. Each node will have a finite number of TCP ports that can be connected to; the IP address defines the node, and the port defines the connection on that node.

**Payload**

Add a note hereFinally, the sections marked DATA represent application data—also known as the “payload” of the packet. Anything that needs to be sent over the network is sent inside this part of the packet, with all the headers wrapped around it. It’s a lot of work and complexity just to send some data, isn’t it?

Add a note hereThe payload is what DataPower is all about. TCP is Layer 4 in the OSI model shown previously; the true *raison d’etre* of DataPower is to act as an OSI Layer 7 firewall. That doesn’t mean that the lower levels are any less important—indeed DataPower has advanced TCP/IP capabilities that are used to reinforce and strengthen the application layer functionality. It is vital to understand both the methods of configuration and the theory involved, so that the Layer 7 application level traffic can go where it is supposed to go.

**Add a note here****Address Resolution**

Add a note hereOn a broadcast Ethernet IP network, how do nodes determine how to address their packets? Let’s say we have two nodes on a network that want to communicate with each other. The first is a Web services client, and the second is a DataPower appliance. The DataPower appliance has IP address 192.168.10.25, and the Web services client has 192.168.10.1. The client wants to send a packet to the DataPower device, to initiate a connection. The packet contains the following information:

* Add a note hereClient Ethernet (MAC) address (source MAC address)
* Add a note hereClient IP address (source IP address)
* Add a note hereDataPower IP address (destination IP address)

Add a note hereHowever, it does not have the DataPower appliance’s Ethernet (MAC) address; it cannot actually create the correct packet to send out onto the network. This is where ARP comes in.

Add a note hereWhat the client needs to do is send out a special packet called an Address Resolution Protocol (ARP) packet. ARP is a network layer protocol like IP; it is sent out encapsulated in an Ethernet packet. However, it is sent to a special Ethernet destination address, called a broadcast address (for Ethernet, this is FF:FF:FF:FF:FF:FF). This means that every machine on the network should look at the packet to decide whether it should deal with it.

Add a note hereThe ARP packet contains a request that effectively says “I am the Web services client with IP address 192.168.10.1, and my Ethernet address is the source address of this packet. I am looking for the device whose IP address is 192.168.10.25. If this is you, please reply to this packet and tell me what your Ethernet address is so that I can communicate with you.” When the DataPower device sees the request packet, it picks it up (because it is a broadcast packet, addressed to everyone), and it sees that the Web services client wants to communicate with it. It then sends out an ARP reply packet that is addressed to the Web services client, on its correct Internet and hardware Ethernet address, saying “This is the device with IP address 192.168.10.25. If you want to talk to me, use the source Ethernet (MAC) address of this packet as your destination.”

**Add a note here****Subnetworks**

Add a note hereSo, how does a node determine whether another node that it wants to talk to is on the same physical medium? What tells it that it is on the same broadcastable network? To understand this, you need to know how IP addressing works.

Add a note hereAn IP address is known as a “dotted quad,” because of the most common representation—four numbers separated by periods. Each number ranges from 0–255, which is storable in one byte of data. The IP address uniquely identifies the device on a network. This means that there is a theoretical maximum of 255×255×255×255=4,228,250,625 unique nodes on an IP network such as the Internet as we currently know it. This seems like a lot of nodes! Clearly they are not all plugged in to the same local network; connecting from a Web service client in the UK to a DataPower appliance located physically in Brazil is obviously not a local network connection! But how does the Web service client know that the appliance is not on its local network so that it can send the packet to a router instead of issuing an ARP request for it?

Add a note hereThe answer is that we split the networks up by using *subnets*. A subnet is, like it sounds, a separate piece of the overall address space that is configured to tell a node which other nodes are on its own locally broadcastable network. It works as a bitmask—here is an example. Let’s take the IP address 192.168.58.31, with a subnet mask (commonly shortened to netmask) of 255.255.255.0. The way we calculate whether a node is on the same network is by performing a binary AND against the netmask. An example is shown in Listing 4-1.

Add a note hereListing 4-1: Subnet Calculation

Add a note hereIn binary

IP: 11000000 10101000 00111010 00011111

Netmask: 11111111 11111111 11111111 00000000

Subnet: 11000000 10101000 00111010 00000000

In decimal

IP: 192.168.58.31

Netmask: 255.255.255.0

Subnet: 192.168.58.0

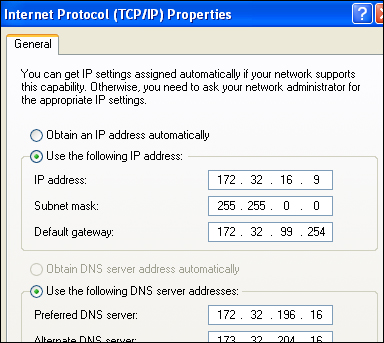
Add a note hereOur subnet in this instance is 192.168.58.0. This means that if the node wants to send a packet to any other node that is in the range 192.168.58.[0–255], it simply sends out an ARP request for the address and expects the node to be on the same physical network medium.

Add a note hereWhile subnets are usually used to group machines that are physically close together, this is not a requirement. Use of a subnet mask is in no way “enforced” on the wire, at any level; rather, subnets are more of a gentleman’s agreement. It is perfectly possible for two groups of computers configured with different subnet masks to share the same physical medium. What happens is each computer chooses to look only at packets for the subnet that it is configured to use. Again, this is in no way enforced; subnetting should not be seen as linked to security in any way.

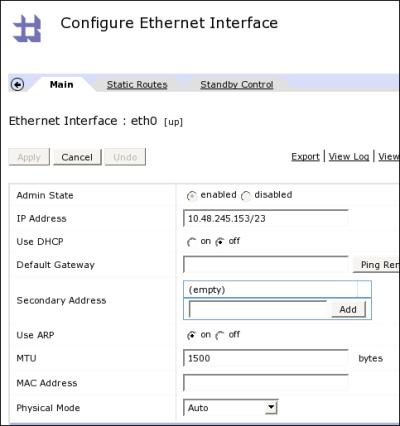
**CIDR Format**

Add a note hereAnother way of referring to the netmask is to use a shorthand called Classless Inter-Domain Routing (CIDR) notation. CIDR is an implementation of variable length subnet masking (VLSM) that exists to work around the limitations of classed networks. (These are explained in the next section.) The CIDR notation is simply a count of the number of 1s in the binary representation of the netmask. In the preceding example, there are twenty-four 1s before we get to zeros—we can write the subnet in CIDR as 192.168.58.0/24.

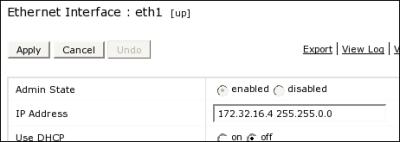
Add a note hereCIDR is often used by experienced network professionals because it is a compact representation. On the DataPower appliance, CIDR format is the primary method for specifying subnet masks and is used in many fields as the netmask representation. This sometimes confuses people, because they expect to see a separate field in which to specify a subnet mask, as you would in the Windows networking configuration panel (shown in Figure 4-10).

  
Add a note hereFigure 4-10: The Windows Networking dialog.

Add a note hereHowever, on the DataPower appliance, it is expected that subnet masks will be specified using CIDR format, and thus there is no subnet mask field on the form, as can be seen in Figure 4-11.

[](javascript:PopImage('IMG_53','http://images.books24x7.com/bookimages/id_30903/04fig11.jpg','483','515'))  
Add a note hereFigure 4-11: Subnet masks are specified in CIDR format.

Add a note hereIn fact, it is possible to specify the subnet mask using dotted quad notation by simply leaving a space between the address and the mask, as shown in Figure 4-12.

[](javascript:PopImage('IMG_54','http://images.books24x7.com/bookimages/id_30903/04fig12.jpg','462','165'))  
Add a note hereFigure 4-12: You can also use dotted quad to specify the subnet mask.

**Classed Networks**

Add a note hereSubnets are commonly described as being broken down into classes. The purpose of classes was, in the early days of the Internet, when things were more structured and there were many less nodes, to allow an allocation of a specific group of IP addresses to a company or an organization. This still holds true to some degree, in that the classes that were allocated cannot be taken away, although the concept of allocating a whole large class to a single organization is falling out of use.

Add a note hereIn fact, the concept of classed subnets significantly predates the use of subnets as we know them today on the Internet. It used to only be possible to purchase a whole class—and because these classes are not very granular, this lead to large-scale waste of IP addresses.

Add a note hereThe classes are split into A, B, and C. IP addressing uses a 32-bit binary word address space, which is split into four byte-long groups. Each of these classes is an easy- to-use-subnet in that they refer to a one-, two-, or three-byte netmasks. Table 4.2 shows the classes and their subnet masks, in both dotted quad and CIDR format. It also lists the *private* address space relevant to the class; these are addresses that have been pre-allocated for internal use, within private organizations and companies, and are not to be used out on the Internet. Where such networks require Internet access, they commonly use network address translation techniques to achieve this, where a router or firewall maps internal to external addresses dynamically to connect the two networks. Finally, Table 4-2 also contains some real examples of allocated classed networks. This information is publicly available on the Internet via the whois protocol, and most of the examples in the table were chosen at random.

| Add a note hereTable 4-2: Classed Networks  [[http://www.books24x7.com/images/b24-bluearrow.gif](http://www.books24x7.com/outputobject.asp?bookid=30903&chunkid=433389886&objectid=ch04table02&objecttype=spreadsheet)Open table as spreadsheet](http://www.books24x7.com/outputobject.asp?bookid=30903&chunkid=433389886&objectid=ch04table02&objecttype=spreadsheet) | | | | |
| --- | --- | --- | --- | --- |
| **Add a note hereClass** | **Add a note hereNetmask** | **Add a note hereCIDR** | **Add a note herePrivate** | **Add a note hereExamples** |
| Add a note hereClass A | Add a note here255.0.0.0 | Add a note here/8 | Add a note here10.0.0.0/8 | Add a note here9.0.0.0/8—owned by IBM. |
|  |  |  |  | Add a note here19.0.0.0/8—owned by the Ford Motor Company. |
| Add a note hereClass B | Add a note here255.255.0.0 | Add a note here/16 | Add a note here172.32.(16-31).0/16 | Add a note here156.26.0.0/16—owned by Wichita State University.  Add a note here207.36.0.0/16—owned by Affinity Internet Inc. |
| Add a note hereClass C | Add a note here255.255.255.0 | Add a note here/24 | Add a note here192.168.(0-255).0/24 | Add a note hereClass C subnets are usually allocated locally by the owner of the encompassing class B network. |

**Add a note here****Routing**

Add a note hereSo, how do the packets destined for nodes that are not connected to a single shared physical medium actually get to where they need to go? What will take a letter out of a pigeonhole in one university and deliver it to the other university where it can be placed in the appropriate pigeonhole? The answer is routing—the Internet equivalent of the postal service.

Add a note hereThe Internet is a big network that is actually made up of lots of small, interconnected networks. At home, you might have a small local area network of a couple of PCs; at work there maybe a network of hundreds or thousands of nodes. Each of these is a self-contained network in its own right, with a shared physical medium (probably Ethernet), and each is capable of sending traffic to the other machines on its own network. But when you want to access a machine that is not on the same physical network segment, you cannot simply put its hardware address into the destination field of your packets; it will never see your request.

Add a note hereWhat if we have a general purpose computer and a DataPower appliance that want to communicate, but they are actually on separate physical networks? Let’s say our computer is a PC, and it wants to make a Web service call to the DataPower appliance. They too are on separate subnets, because they are separately physically connected to completely different networks. This means that the PC initiating the connection knows that the target IP address it wants to contact is not on its own subnet, so it will not look for the node locally using ARP. Instead it will be configured to send the packet to the hardware address of a machine called a router.

Add a note hereThe router’s job is to forward on the packet to the target machine. If the router has another interface on the DataPower appliance’s subnet, this is as simple as issuing an ARP request for the DataPower appliance and sending the packet to the appropriate Ethernet address. However it may be that the router needs to forward the packet on to another router. Indeed, that other router may in turn need to forward the packet on to yet another router. And so on and so on until the packet finally reaches a router that is local to the DataPower appliance, which uses ARP to find the hardware address and passes on the packet.

**Add a note here****Routing Table**

Add a note hereSo, what does the node do when it realizes that the address it wants to send packets to is not in its subnet? This is what the routing table is for. Every machine on an IP network, be it a Windows PC, a Unix Server, or a DataPower appliance, has a routing table. The routing table tells the machine what to do with packets when they are on different subnets. Listing 4-2 contains an example of a routing table from a DataPower appliance.

Add a note hereListing 4-2: The Routing Table from a DataPower Appliance

Add a note herexi50# show route

Destination Interface Type Interface VLAN Gateway Metric

----------- -------------- --------- ---- ------- ------

0.0.0.0/0 Ethernet mgt0 192.168.58.1 0

192.168.58.0/24 Ethernet mgt0 0

10.0.0.0/8 Ethernet eth1 10.24.14.1 0

xi50#

Add a note hereThis routing table shows first that the machine has more than one physical interface in use; under Device Interface, you can see mgt0 and eth1 as active. This means that the machine is “multihomed”—which simply means that it has more than one interface. Each route is assigned a metric—in this case they are all set to 0, but if needed this could be used to differentiate between two routes that are valid for the same traffic. This routing table says the following:

* Add a note hereIf the destination address is on network 192.168.58.0/24, send the packet directly to the node via network interface mgt0. ARP will look for the destination node directly.
* Add a note hereIf the destination address is on network 10.0.0.0/8, send the packet to the router 10.24.14.1 via network interface eth1. ARP will look for the router 10.24.14.1 on this interface.
* Add a note hereIf the destination address does not match any other route, send the packet to the router 192.168.58.1 via network interface mgt0. ARP looks for the router and sends the packet with the target IP address of the destination node but with the Ethernet address of the router.

Add a note hereThe last of these routes is called the default route. The default route must obviously be set to a router that will know how to forward the packet, either directly or via one or more further routers, to the remote network where the remote node is connected.

**Tip: Default Routes**

Add a note hereThe DataPower appliance is perfectly capable of having more than one default route. If you choose to configure the network interfaces and routing table in this way, you need to be aware of the behavior and make a conscious decision to do it.

Add a note hereWhen outbound traffic has more than one network route available to it, the behavior is undefined; that is, the route chosen (and therefore interface used) will be random. In some instances, this may be appropriate, but it can also lead to some unexpected results!

Add a note hereConsider for instance if one of the “default” routes does not have a valid route to a specific server, but the other does? At random, connections will fail! Or what if there is more than one valid route to a server, but one of them goes over a trusted network and the other routes over an untrusted network? In this case, you’d better hope that you’re not sending sensitive data!

Add a note hereThe vast majority of configurations should have only a single default route configured on the main interface that is used to process traffic; more specific static routes should likely be used to explicitly define other routes to servers as needed. This is something best discussed with the administrator of your local network.

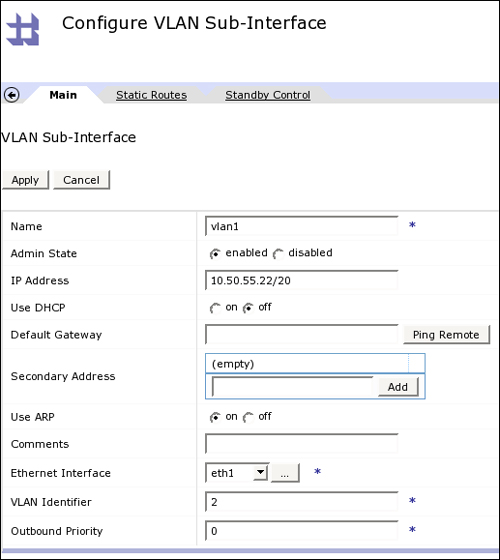
**Add a note here****Virtual LANs**

Add a note hereAs of the 3.6.1 firmware, DataPower has supported the ability to participate in 802.1Q VLAN trunking. What does that mean?

Add a note hereA Virtual Local Area Network, or VLAN, is a group of hosts that communicate with each other as if they were connected to the same physical network, regardless of whether they are actually physically connected. Everything that you could do on a normal physical network can be done over a VLAN. This includes sending broadcast packets, so ARP works on a VLAN just as it would on a local physical network.

Add a note hereThe use of Virtual LANs is a powerful advanced networking configuration. These are fundamentally OSI Layer 2 constructs; by tagging packets as belonging to a specific VLAN, you are effectively plugging in to this virtual transport layer, over which you will send IP traffic. For DataPower, the abstraction provided to encapsulate this is that we can configure a virtual network interface for the VLAN, and to all intents and purposes, we can configure and use that interface as if it were a real physical network interface.

Add a note hereThe configuration screen for a VLAN interface (or more properly sub-interface) on the DataPower appliance is shown in Figure 4-13.

[](javascript:PopImage('IMG_55','http://images.books24x7.com/bookimages/id_30903/04fig13_alt.jpg','517','579'))  
Add a note hereFigure 4-13: Configuring the VLAN interface.

Add a note hereNote that the IP address assigned to this interface is completely different to any of the “real” networks to which the device is connected from earlier configurations. The Virtual LAN uses eth1 as its physical medium, but packets are sent out tagged with a VLAN identifier. (In this case simply 2—note that the identifier 1 should not be used, as switches usually reserve this VLAN for unidentified packets.)

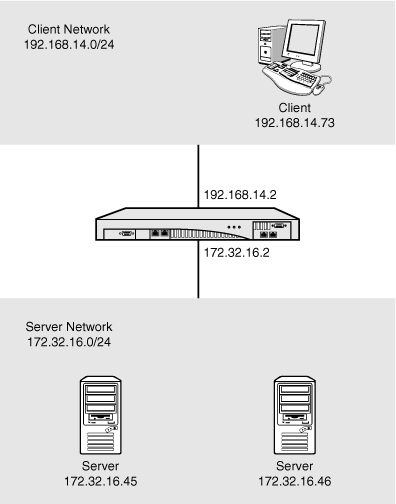
Add a note hereWhen something on the DataPower appliance tries to contact a server on the subnet served by the VLAN, the appliance sends out packets on eth1 but tagged with the VLAN identifier, and it expects the router or switch to which eth1 is connected will understand this identifier. This is key in using VLANs—if the switch does not understand the tags, the VLAN traffic cannot go anywhere.

**DataPower Networking Scenarios**

Add a note hereNow that you understand the basic terminology and concepts for networking and routing, at least at a high level, let’s go through a number of scenarios of typical appliance network deployments. [Chapter 5](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=635#635), [“Common DataPower Deployment Patterns,”](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=635#635) will detail deployment topologies and use cases; this section covers some specific example deployment scenarios explicitly from the point of view of the networking stack. It deliberately does not cover security (other than implied network security issues), performance, availability or any other issues; the idea is to present specific DataPower deployment examples and discuss the issues they raise.

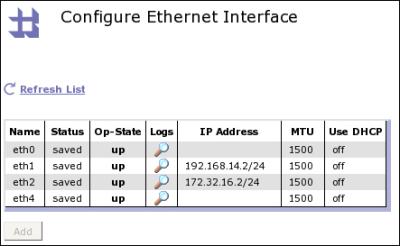
**Add a note here****Scenario: External and Internal**

Add a note hereThis configuration is one of the most common deployment scenarios. The network situation, as depicted in Figure 4-14, has two networks—the first of which is used for incoming client requests and the second to contact backend servers to service those requests. Let’s call the incoming network the client network, and the backend network the server network.

  
Add a note hereFigure 4-14: External and internal.

Add a note hereNote that this is very different and distinct to linking these networks using a typical packet processing and forwarding device or bridge. With DataPower connecting the networks, the only way that data can make it from one side to the other is if DataPower processes it and explicitly decides to proxy it through.

Add a note hereThe client network is configured to be 192.168.14.0/24 and the server network 172.32.16.0/24. The appliance is configured such that one interface is dedicated to the client network, which is 192.168.14.2, and one is dedicated to the server network on 172.32.16.2. It really doesn’t matter which interface is configured to which, as long as they are physically plugged into the correct networking infrastructure. For this discussion, let’s assume that the client network interface 192.168.14.2 is configured on eth1 and the server network interface 172.32.16.2 is configured on eth2, as shown in Figure 4-15.

[](javascript:PopImage('IMG_57','http://images.books24x7.com/bookimages/id_30903/04fig15.jpg','443','273'))  
Add a note hereFigure 4-15: External and internal interfaces.

Add a note hereThe client network accepts incoming requests from Web service clients—in this case, 192.168.14.73 is the Web service client. At a packet level, we call these *ingress* connections—simply meaning incoming. The server network is used to contact backend servers, such as 172.32.16.45 and 172.32.16.46 in the diagram. These are known as *egress* or outgoing connections.

Add a note hereThere are two important things to consider in this scenario. First, services (such as Web Service Proxies or Multi-Protocol Gateways), which listen for network connections on the device, should be configured to listen only on the client network. They should be defined such that, when the service is created, its interface setting lists only the host alias that resolves to 192.168.14.2. This means that the service is not accessible by other nodes on the 172.32.16.0/24 network—which is likely the desired configuration. If instead you choose to bind a service to the default of 0.0.0.0, which we always recommend against, it will be available on *both* network interfaces—including the internal interface where it should not be available!

**Tip: Host Aliases**

Add a note hereThroughout this book, you will notice us telling you to use host aliases. This section uses IP addresses to make it more obvious that routes and subnet masks are applicable to which network traffic; however, in a real configuration, host aliases should always be used to define IP addresses for interfaces, both to make the configuration more portable and to ensure consistency of this vital configuration information.

Add a note hereThe second consideration is a routing question—of the two interfaces, which should be the default route for the device? We showed in [Chapter 3](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=308#308) how to configure a default route on an interface. However, for most noncomplex configurations, there should be only one default route configured on a single interface on the device. This is the safest way to ensure that “unintended” routing behavior does not occur.

Add a note hereFor example, let’s configure the default route to be to the client interface, as shown in Listing 4-3.

Add a note hereListing 4-3: Default Route Configured on the Client Interface

Add a note herexi50# show route

Destination Interface Type Interface VLAN Gateway Metric

----------- -------------- --------- ---- ------- ------

0.0.0.0/0 Ethernet eth1 192.168.14.1 0

192.168.14.0/24 Ethernet eth1 0

172.32.16.0/24 Ethernet eth2 0

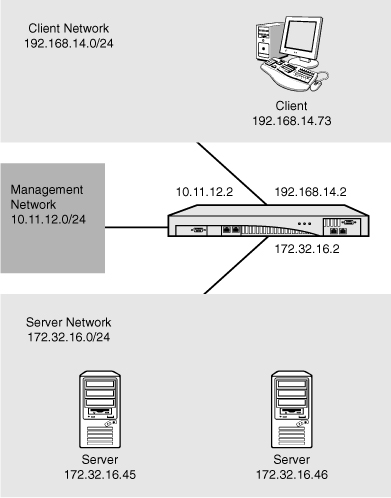
xi50#

Add a note hereWith this configuration, the only network connections that will be routed onto the server network are those that are explicitly to machines in the subnet 172.32.16.0/24. This is likely to be exactly the desired configuration!

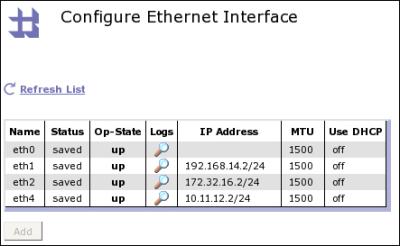
Add a note hereAs a matter of fact, some more complex configurations may actually have multiple default routes deliberately. For instance, you may want to have a secondary route available in the event of network congestion. In this instance, for a connection where there is no more specific route, DataPower sends packets randomly over one or other interface. This may or may not be the desired behavior! For static routes, the metric field could be used to specify a preference for sending traffic over a specific link (perhaps one which is less congested or cheaper to use). In the vast majority of cases, it is better to let these kinds of routing decisions be made by dedicated routers, which participate in distributed routing protocols and are best placed to understand how to route traffic appropriately.

**Add a note here****Scenario: Management Network**

Add a note hereThe previous scenario uses two interfaces: one for client and one for server connections. Let’s add to this configuration by including a commonly used pattern—a management network. The idea is that a separate network connection should be used for accessing the administrative interfaces (console, SSH, SOAP, and so on) so that this sensitive traffic is not sent over the interfaces that serve customer traffic, to clients or servers. This addition is depicted in Figure 4-16.

  
Add a note hereFigure 4-16: Management network.

Add a note hereThe management network has an IP range of 10.11.12.0/24, and the DataPower appliance’s interface on this network is 10.11.12.2. The updated DataPower configuration is shown in Figure 4-17, where 10.11.12.2 is configured on eth4 (also known as mgt0).

[](javascript:PopImage('IMG_59','http://images.books24x7.com/bookimages/id_30903/04fig17.jpg','443','273'))  
Add a note hereFigure 4-17: Three interfaces, including a management interface.

Add a note hereThis configuration has a number of further considerations that must be discussed before implementation.

Add a note hereFirst, in the same way client services should be made bind-specific and listen only to the client interface, it is important to ensure that the administrative interfaces bind only to the management network. This can be done by specifying the specific IP address (by choosing the host alias tied to that specific IP address) when defining the service—web-mgmt, SSH, and so on as shown in [Chapter 2](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=145#145). The listener configuration can be verified by using the TCP Port Status, as described in [Chapter 3](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=308#308) and depicted in Figure 4-18.

[](javascript:PopImage('IMG_60','http://images.books24x7.com/bookimages/id_30903/04fig18_alt.jpg','836','127'))  
Add a note hereFigure 4-18: Bind the web-mgmt to the management interface.

Add a note hereThe issue of routing is the next important issue. In general, there should be no routes configured over the management network, other than any of those that are explicitly needed to enable the specific traffic to flow. If the machines that are clients are on the same broadcastable subnet (for instance, if a machine was 10.11.12.64 on the management network), ARP is used to contact the server and no routing is required. If, on the other hand, the machines that are used to administer the device are remote but still accessed over the management interface, a specific static route should be added to enable packets to return to that machine.

Add a note hereFor instance, traffic coming back to administrative clients on 10.11.72.0/24 would require a static route telling the device how to pass on data for that network, and it would do this by providing a gateway address for that remote network. Listing 4-4 shows the routing table that would provide this configuration; notice that the route to the 10.11.72.0/24 subnet is explicitly being told which router to use, and that router is on the 10.11.12.0/24 subnet accessible via the mgt0 (eth4) interface.

Add a note hereListing 4-4: Routing with the Management Network

Add a note herexi50# show route

Destination Interface Type Interface VLAN Gateway Metric

----------- -------------- --------- ---- ------- ------

0.0.0.0/0 Ethernet eth1 192.168.14.1 0

192.168.14.0/24 Ethernet eth1 0

172.32.16.0/24 Ethernet eth2 0

10.11.12.0/24 Ethernet mgt0 0

10.11.72.0/24 Ethernet mgt0 10.11.12.1 0

xi50#

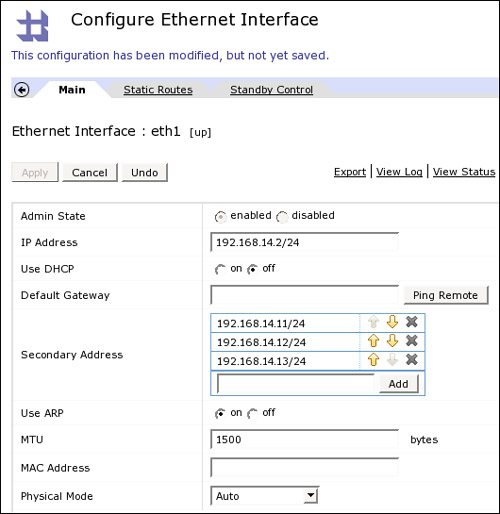
Add a note hereIt is especially important to ensure that no default route is set up for the management network. There is no route with a destination of 0.0.0.0 for mgt0 shown in Listing 4-4, and there should almost never be a need for this. The management network is designed for administering the device; application traffic should in all likelihood never be going onto this network.

Add a note hereAnother trap to be aware of is if there is a valid route from the device to another server via its management interface. If the routing table on DataPower is misconfigured, and the other server accepts the traffic over its interface to the management network, customer traffic could go over management network completely unintentionally!

Add a note hereFinally, you should consider whether using a management network is the right thing for you. This is a powerful concept, and is highly recommended, but it is *critical* to configure it correctly. Many people do not consider that when they create a management network, they are creating the perfect platform from which to attack their infrastructure. Often servers have misconfigured management type connections over dedicated management networks—perhaps using default usernames and passwords, and without really thinking about security, because they are considered to be on a “safe” network. However, this “safe” network provides a wonderful back door into systems—a great way for a bad guy to make out-of-band connections to other servers! Configure all nodes on a management network with care, and stay safe.

**Add a note here****Scenario: IP Aliases**

Add a note hereThe DataPower appliance has the capability, as shown in [Chapter 3](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=308#308), to add secondary IP addresses or IP aliases to a network interface. Many aliases can be configured on a single interface, and they are all valid IP addresses for that interface. An example is shown in Figure 4-19.

[](javascript:PopImage('IMG_61','http://images.books24x7.com/bookimages/id_30903/04fig19_alt.jpg','530','545'))  
Add a note hereFigure 4-19: IP aliases.

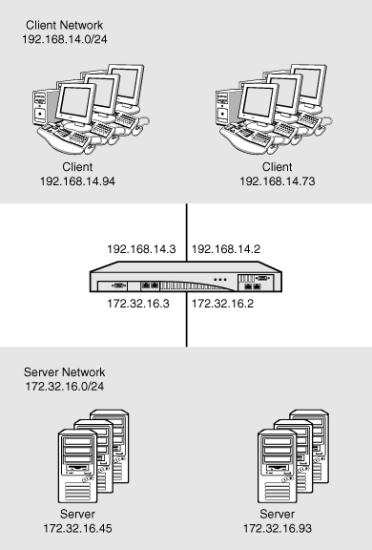
Add a note hereIP aliases have various uses, but one of the most important reasons they are used is so that different frontside handlers can use different SSL certificates. As [Chapter 18](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=3288#3288), [“DataPower and SSL,”](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=3288#3288) will demonstrate, only one SSL server certificate can be presented over a given IP address and port combination. This is because the SSL handshake has to complete before any application traffic can proceed, so the application level traffic *cannot* choose which SSL certificate is used.

Add a note hereBecause of this limitation, which is inherent to the protocol, a configuration such as that presented in Figure 4-19 comes into its own. It can allow four separate HTTPS frontside handlers all listening on the same port but on different IP addresses, on a single network interface.

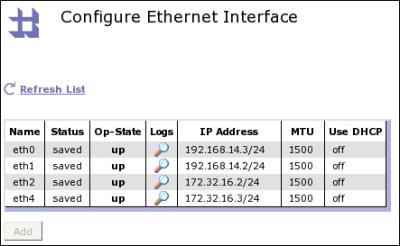
**Add a note here****Scenario: Multiple Interfaces on the Same Network**

Add a note hereDataPower appliances are high-performance devices. Even though the devices’ Ethernet ports are Gigabit Ethernet, the fastest standard in common use today, it is possible that the network speed may become the bottleneck. This might especially be an issue when connecting to older networking hardware—for instance, if the device is plugged into a 100Mb Ethernet port (of which there are still many in use), its available network bandwidth per interface drops by an order of magnitude.

Add a note hereIn this situation, it may be desirable to have more than one network interface plugged in to the same network. Consider Figure 4-20.

[](javascript:PopImage('IMG_62','http://images.books24x7.com/bookimages/id_30903/04fig20.jpg','390','576'))  
Add a note hereFigure 4-20: Two interfaces per network.

Add a note hereIn this example, there are two network interfaces plugged in to the client network and two network interfaces plugged in to the server network. Clients can connect to either 192.168.14.2 or 192.168.14.3; of course, this needs to somehow be either load balanced or built in to the client logic. Server connections may be initiated from either 172.32.16.2 or 172.32.16.3. The DataPower configuration for this is shown in Figure 4-21.

[](javascript:PopImage('IMG_63','http://images.books24x7.com/bookimages/id_30903/04fig21.jpg','443','273'))  
Add a note hereFigure 4-21: Two network interfaces per network.

Add a note hereIn this scenario, we explicitly want to configure the [routing](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=554#554) table such that there are two equal priority routes on each side. This way, the device will randomly decide which interface to use for both ingress and egress connections. An example of this [routing](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=554#554) configuration is shown in Listing 4-5.

Add a note hereListing 4-5: Routing Table for Two Network Interfaces per Network

Add a note herexi50# show route

Destination Interface Type Interface VLAN Gateway Metric

----------- -------------- --------- ---- ------- ------

192.168.14.0/24 Ethernet eth0 0

192.168.14.0/24 Ethernet eth1 0

172.32.16.0/24 Ethernet eth2 0

172.32.16.0/24 Ethernet mgt0 0

xi50#

Add a note hereIn this scenario, there happens to be no default route configured on any of the network interfaces of the appliance. This means that the only networks that are addressable are 172.32.16.0/24 and 192.168.14.0/24; should there be any packets for any reason destined to any other network, they would have no route to use and would fail. Packets that are destined to one of our two valid network targets will be sent out at random from either of the two interfaces configured for the network.

Add a note hereThis kind of configuration may be useful in any situation where the gigabit network interface is the limiting factor. These are likely to be few and far between! Possibly when performing fast requests for a high volume of client traffic—on the scale of say an international sporting championship final or a very popular online auction house—the network card may become the bottleneck, at which point this scenario may be useful.

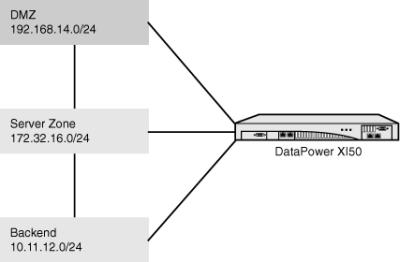
Add a note hereNotice that in this scenario there is no network interface configured for management. This is possibly a valid scenario—but it does not necessarily mean that we have to have the administrative GUI or SSH bound to one of our application traffic interfaces. Especially for high-volume, business-critical, and security-sensitive devices, it is perfectly possible to run the DataPower appliance without running any of the network-based administrative interfaces! After the configuration is uploaded, there may be no need or requirement for network-based administration, and any needed work can always take place via the serial console. Some customers even set up a completely private network where network addressable serial terminal emulators enable highly secure network access to the serial console for remote administration.

**Add a note here****Scenario: Different Network Zones**

Add a note hereThe last example scenario to be presented here is one that some DataPower users have concerns about. It is presented not as a definitive answer or statement saying that this is the best configuration to use; rather we demonstrate it to show how this may be a valid and safe configuration—using the same DataPower appliance in different network zones.

Add a note hereA typical secure middleware deployment with traffic facing an untrusted network such as the Internet might be deployed in three separate network zones. The first of these would likely be a demilitarized zone, or DMZ; the first stop for incoming connections, and a barren, hostile place for an attacker. Behind this may be a server zone, where middleware application servers running business logic would be connected; this zone is also well protected, and can only be accessed via the DMZ. Finally, there may be a backend network area, where Enterprise Systems or databases would run; this is the most protected zone, where actual business data is stored, and only applications running in the server zone can access the backend.

Add a note hereThis scenario suggests that it may be safely possible to use a single DataPower appliance to perform services across all these network zones, as in the layout depicted in Figure 4-22.

[](javascript:PopImage('IMG_64','http://images.books24x7.com/bookimages/id_30903/04fig22.jpg','427','280'))  
Add a note hereFigure 4-22: One device, three separate network zones!

Add a note hereDon’t stop reading just yet! It is true that traditional logic dictates that one should not share servers between different zones. That is certainly true for general purpose servers. But is it also true for appliances? For example, do you actually have separate physical firewalls between your DMZ and your server zone, and between your server zone and your backend network? Why not? Because the firewall is an appliance that is trusted to separate and secure these networks. Why might that same logic not apply to other appliances, especially security hardened appliances?

Add a note hereAn oft-asked question from those demilitarized zone DataPower customers who want to deploy the appliance in this manner—especially where the two networks are a DMZ and a server zone as above—is, “Can IBM guarantee that no traffic will be leaked onto the wrong network?” Well, the answer is, “It depends.” What is hopefully clear in this chapter is that the flow of packets from and to the device is decided by the routing table. It is, therefore, important to have the routing table well defined and configured such that there is no unexpected behavior.

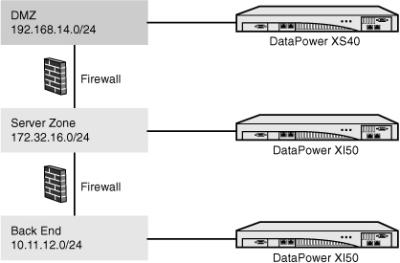
Add a note hereBefore we get into the routing configuration, let’s first ask a question: Why would you want to configure the device in this way? Primarily, those who want to do this are interested in saving money. They see a need for the DataPower appliance in each of their zones. For instance, in the DMZ, they want to perform XML Threat Protection and schema validation; in the server zone they are performing routing and mediation as part of an Enterprise Service Bus function; and at the backend, they are carrying out high-speed transformations of data and connectivity to legacy systems. These are all valid use cases for the appliance—indeed it excels at all three. But perhaps the customer doesn’t want to (or can’t afford, within the scope of a single project) to purchase three separate devices.

Add a note hereFundamentally, when choosing to save money, there is always a question of saving versus risk. The more you save, the higher the risk grows; this trade-off is well known in IT. The trick is to manage the risk, from both a technical and business perspective, and make informed decisions to try and ensure that you are choosing the best way forward for your company. There will be some level of risk involved in choosing to deploy like this. Let’s try to quantify this risk, and then think about how we might mitigate it.

Add a note hereWhat is the actual risk? This partially depends on how you choose to configure your services. In particular, are you choosing to truly act as if you had three separate devices, with no direct communication from one zone to another through the device, or are you taking shortcuts and in effect bypassing the firewalls? Both of these might be valid deployment decisions, but the implications of each one need to be understood.

**The Target Environment**

Add a note hereFigure 4-23 shows the same layout as before, but using three separate DataPower appliances—an XS40 and two XI50s. We have also explicitly shown the firewalls between the network zones, to make it clear that there is no direct traffic between the zones that does not go through the firewall. Note how each device is attached to a specific network; it can communicate only with other nodes in that network. If it does want to make a connection to a machine in the server zone, it cannot go directly—it has to go through the firewall that separates the DMZ from the server zone.

[](javascript:PopImage('IMG_65','http://images.books24x7.com/bookimages/id_30903/04fig23.jpg','418','275'))  
Add a note hereFigure 4-23: Three separate appliances.

Add a note hereThis is the safest possible configuration; separate physical devices, and no possible risk of cross contamination between the zones. Everything you do that is less than this is a compromise, balancing the cost of separate devices against the increased complexity of replicating the configuration on a single device.

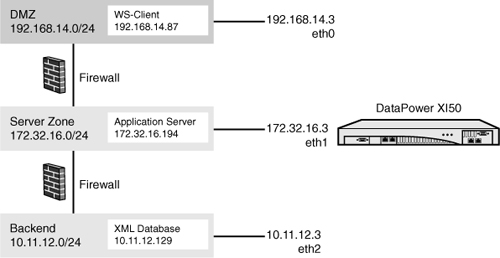
**Using a Single Device**

Add a note hereIt is possible to replicate this configuration with a single device, using the concept of domains, which will be detailed in [Chapter 12](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=2003#2003), [“Device Administration.”](http://www.books24x7.com/assetviewer.aspx?bkid=30903&destid=2003#2003) The device should be split up into three separate domains, one for the DMZ, one for the server zone, and one for the backend. Each domain is administered individually, by separate users, who have no ability to administer the other domains.

Add a note hereMost importantly, an interface needs to be assigned to each network zone, and an agreement has to be made by all administrators that within a given zone, services will *only* be bound to the specific interface for that zone. More than that, the developers for each domain need to agree that they will *not* under any circumstances attempt to make a network connection directly to servers in the other zones, which they theoretically could access.

Add a note hereFinally, what about the valid connections to servers in the other zones? How do you make sure that these go through the firewall? Well, again the answer is complex. The only way to do this is to make sure that the routing table contains a static entry for the specific host, which directs traffic for that host to the router on the correct side of the firewall. Because the device uses source-based routing, this static route entry will not affect connections *from* that host to the device, but it will ensure that any connections from the appliance to the host will use the route through the firewall.

Add a note hereThis is a nontrivial area, so let’s examine our configuration in more depth, putting specific servers and IP addresses onto the diagram, to understand what the routing table will need to contain. Consider the configuration in Figure 4-24.

[](javascript:PopImage('IMG_66','http://images.books24x7.com/bookimages/id_30903/04fig24_alt.jpg','596','308'))  
Add a note hereFigure 4-24: A single appliance doing the same job.

Add a note hereThe flow of traffic might go from the WS-Client on 192.168.14.87 to a service listening on eth0 of the DataPower device, on 192.168.14.3. This service is a WS-Proxy fronting the application server on 172.32.16.194. In order that outbound traffic from 192.168.14.3 goes to the Application Server via the firewall, we would need to add a static route to 172.32.16.194/32 (that is, to exactly that one host). The application server might then call a Multi-Protocol Gateway listening on eth1, on 172.32.16.3—the response packets for this connection will be sent out back over the same interface rather than using the static route because DataPower’s default behavior is to perform source-based routing.

Add a note hereIn the same way, the Multi-Protocol Gateway might need to make a call to the XML Database on the backend layer—and so a static route would need to be set to ensure that outbound calls to 10.11.12.129/32 are sent out via eth1 not eth2. The XML database may need to make use of an XML firewall performing complex transformation, running on eth2 10.11.12.3—and response packets back will be sent via eth2, rather than using the static route, because they came in on the eth2 interface.

Add a note hereThe routing table to implement such a scenario is shown in Listing 4-6.

Add a note hereListing 4-6: Routing Table for Two Network Interfaces per Network

Add a note herexi50# show route

Destination Interface Type Interface VLAN Gateway Metric

----------- -------------- --------- ---- ------- ------

0.0.0.0/0 Ethernet eth0 192.168.14.1 0

192.168.14.0/24 Ethernet eth0 0

172.32.16.0/24 Ethernet eth1 0

10.11.12.0/24 Ethernet eth2 0

172.32.16.194/32 Ethernet eth0 0

10.11.12.129/32 Ethernet eth1 0

xi50#

Add a note hereThe most important routes are highlighted. Our most specific route from the appliance to the Application Server is explicitly directed to eth0, ensuring that this connection will go through the firewall and not straight out through eth1. Likewise, our most specific route to the XML Database is explicitly set to go via eth1 rather than eth2, again making sure that the traffic will be sent via the firewall.

## Summary

Add a note hereThis chapter has gone into more detail on how the DataPower network stack works and has provided some example deployment scenarios. The terminology included should stand you in good stead for the rest of the book, but the theory has barely touched the surface. We strongly recommend that you explore further, with a dedicated book on networking.