COMUNE DI NOVARA

RIF. CONCORSO n° C3_2020

CONCORSO PUBBLICO PER ESAMI PER LA COPERTURA A TEMPO PIENO E INDETERMINATO DI N. 1 POSTO DI ISTRUTTORE PROGRAMMATORE - CATEGORIA C.

QUESITI PROVA ORALE

BUSTA 1

1. Quali sono le funzioni dei dirigenti?
2. In un sito web cos'è il CMS (Content Management System)?
3. In un database cosa sono le Viste?

4.7 Data Link Layer Switching

Many organizations have multiple LANs and wish to connect them. LANs can be connected by devices called bridges, which operate in the data link layer. Bridges examine the data layer link addresses to do routing. Since they are not supposed to examine the payload field of the frames they route, they can transport IPv4 (used in the Internet now), IPv6 (will be used in the Internet in the future), AppleTalk, ATM, OSI, or any other kinds of packets. In contrast, routers examine the addresses in packets and route based on them. Although this seems like a clear division between bridges and routers, some modern developments, such as the advent of switched Ethernet, have muddied the waters, as we will see later. In the following sections we will look at bridges and switches, especially for connecting different 802 LANs. For a comprehensive treatment of bridges, switches, and related topics, see (Perlman, 2000).

Before getting into the technology of bridges, it is worthwhile taking a look at some common situations in which bridges are used. We will mention six reasons why a single organization may end up with multiple LANs. First, many university and corporate departments have their own LANs, primarily to connect their own personal computers, workstations, and servers. Since the goals of the various departments differ, different departments choose different LANs, without regard to what other departments are doing. Sooner or later, there is a need for interaction, so bridges are needed. In this example, multiple LANs came into existence due to the autonomy of their owners.

Second, the organization may be geographically spread over several buildings separated by considerable distances. It may be cheaper to have separate LANs in each building and connect them with bridges and laser links than to run a single cable over the entire site.

Third, it may be necessary to split what is logically a single LAN into separate LANs to accommodate the load. At many universities, for example, thousands of workstations are available for student and faculty computing. Files are normally kept on file server machines and are downloaded to users' machines upon request. The enormous scale of this system precludes putting all the workstations on a single LAN—the total bandwidth needed is far too high. Instead, multiple LANs connected by bridges are used, as shown in Fig. 4-39. Each LAN contains a cluster of workstations with its own file server so that most traffic is restricted to a single LAN and does not add load to the backbone.

BUSTA 2

1. Le determinazioni a contrarre.
2. Nel linguaggio PHP dove vengono configurate le variabili di sistema?
3. Cos'è un IDE (Integrated Development Environment)?

5.2.11 Node Lookup in Peer-to-Peer Networks

A relatively new phenomenon is peer-to-peer networks, in which a large number of people, usually with permanent wired connections to the Internet, are in contact to share resources. The first widespread application of peer-to-peer technology was for mass crime: 50 million Napster users were exchanging copyrighted songs without the copyright owners' permission until Napster was shut down by the courts amid great controversy.

Nevertheless, peer-to-peer technology has many interesting and legal uses. It also has something similar to a routing problem, although it is not quite the same as the ones we have studied so far. Nevertheless, it is worth a quick look.

What makes peer-to-peer systems interesting is that they are totally distributed. All nodes are symmetric and there is no central control or hierarchy. In a typical peer-to-peer system the users each have some information that may be of interest to other users. This information may be free software, (public domain) music, photographs, and so on. If there are large numbers of users, they will not know each other and will not know where to find what they are looking for. One solution is a big central database, but this may not be feasible for some reason (e.g., nobody is willing to host and maintain it). Thus, the problem comes down to how a user finds a node that contains what he is looking for in the absence of a centralized database or even a centralized index.

Let us assume that each user has one or more data items such as songs, photographs, programs, files, and so on that other users might want to read. Each item has an ASCII string naming it. A potential user knows just the ASCII string and wants to find out if one or more people have copies and, if so, what their IP addresses are.

As an example, consider a distributed genealogical database. Each genealogist has some on-line records for his or her ancestors and relatives, possibly with photos, audio, or even video clips of the person. Multiple people may have the same great grandfather, so an ancestor may have records at multiple nodes. The name of the record is the person's name in some canonical form. At some point, a genealogist discovers his great grandfather's will in an archive, in which the great grandfather bequeaths his gold pocket watch to his nephew. The genealogist now knows the nephew's name and wants to find out if any other genealogist has a record for him. How, without a central database, do we find out who, if anyone, has records?

BUSTA 3

1. Cosa si intende per “debiti fuori bilancio”?
2. Cos’è e cosa serve FTP?
3. Cos’è un Open Data? fare un esempio.

NAT—Network Address Translation

IP addresses are scarce. An ISP might have a /16 (formerly class B) address, giving it 65,534 host numbers. If it has more customers than that, it has a problem. For home customers with dial-up connections, one way around the problem is to dynamically assign an IP address to a computer when it calls up and logs in and take the IP address back when the session ends. In this way, a single /16 address can handle up to 65,534 active users, which is probably good enough for an ISP with several hundred thousand customers. When the session is terminated, the IP address is reassigned to another caller. While this strategy works well for an ISP with a moderate number of home users, it fails for ISPs that primarily serve business customers.

The problem is that business customers expect to be on-line continuously during business hours. Both small businesses, such as three-person travel agencies, and large corporations have multiple computers connected by a LAN. Some computers are employee Pcs; others may be Web servers. Generally, there is a router on the LAN that is connected to the ISP by a leased line to provide continuous connectivity. This arrangement means that each computer must have its own IP address all day long. In effect, the total number of computers owned by all its business customers combined cannot exceed the number of IP addresses the ISP has. For a /16 address, this limits the total number of computers to 65,534. For an ISP with tens of thousands of business customers, this limit will quickly be exceeded.

To make matters worse, more and more home users are subscribing to ADSL or Internet over cable. Two of the features of these services are (1) the user gets a permanent IP address and (2) there is no connect charge (just a monthly flat rate charge), so many ADSL and cable users just stay logged in permanently. This development just adds to the shortage of IP addresses. Assigning IP addresses on-the-fly as is done with dial-up users is of no use because the number of IP addresses in use at any one instant may be many times the number the ISP owns.

BUSTA 4

1. Quali sono gli organi fondamentali del comune?
2. Cosa sono le linee guida per l’accessibilità dei contenuti web?
3. In un database cosa sono i Trigger?

5.6.4 OSPF—The Interior Gateway Routing Protocol

We have now finished our study of Internet control protocols. It is time to move on the next topic: routing in the Internet. As we mentioned earlier, the Internet is made up of a large number of autonomous systems. Each AS is operated by a different organization and can use its own routing algorithm inside. For example, the internal networks of companies X, Y, and Z are usually seen as three ASes if all three are on the Internet. All three may use different routing algorithms internally. Nevertheless, having standards, even for internal routing, simplifies the implementation at the boundaries between ASes and allows reuse of code. In this section we will study routing within an AS.

In the next one, we will look at routing between ASes. A routing algorithm within an AS is called an interior gateway protocol; an algorithm for routing between ASes is called an exterior gateway protocol. The original Internet interior gateway protocol was a distance vector protocol (RIP) based on the Bellman-Ford algorithm inherited from the ARPANET. It worked well in small systems, but less well as ASes got larger. It also suffered from the count-to-infinity problem and generally slow convergence, so it was replaced in May 1979 by a link state protocol. In 1988, the Internet Engineering Task Force began work on a successor. That successor, called OSPF (Open Shortest Path First), became a standard in 1990. Most router vendors now support it, and it has become the main interior gateway protocol. Below we will give a sketch of how OSPF works. For the complete story, see RFC 2328.

Given the long experience with other routing protocols, the group designing the new protocol had a long list of requirements that had to be met. First, the algorithm had to be published in the open literature, hence the "O" in OSPF. A proprietary solution owned by one company would not do.

Second, the new protocol had to support a variety of distance metrics, including physical distance, delay, and so on.

Third, it had to be a dynamic algorithm, one that adapted to changes in the topology automatically and quickly.

Fourth, and new for OSPF, it had to support routing based on type of service. The new protocol had to be able to route real-time traffic one way and other traffic a different way. The IP protocol has a Type of Service field, but no existing routing protocol used it. This field was included in OSPF but still nobody used it, and it was eventually removed.

Fifth, and related to the above, the new protocol had to do load balancing, splitting the load over multiple lines. Most previous protocols sent all packets over the best route. The second-best route was not used at all. In many cases, splitting the load over multiple lines gives better performance.

Sixth, support for hierarchical systems was needed. By 1988, the Internet had grown so large that no router could be expected to know the entire topology. The new routing protocol had to be designed so that no router would have to.

Seventh, some modicum of security was required to prevent fun-loving students from spoofing routers by sending them false routing information. Finally, provision was needed for dealing with routers that were connected to the Internet via a tunnel. Previous protocols did not handle this well.

BUSTA 5

1. Quali sono le funzioni del Sindaco?
2. Cos’è e cosa server un gruppo di continuità?
3. In un Web Service cos’è il WSDL?

6.4.2 Remote Procedure Call

In a certain sense, sending a message to a remote host and getting a reply back is a lot like making a function call in a programming language. In both cases you start with one or more parameters and you get back a result. This observation has led people to try to arrange request-reply interactions on networks to be cast in the form of procedure calls. Such an arrangement makes network applications much easier to program and more familiar to deal with. For example, just imagine a procedure named get_IP_address (host_name) that works by sending a UDP packet to a DNS server and waiting for the reply, timing out and trying again if one is not forthcoming quickly enough. In this way, all the details of networking can be hidden from the programmer.

The key work in this area was done by Birrell and Nelson (1984). In a nutshell, what Birrell and Nelson suggested was allowing programs to call procedures located on remote hosts. When a process on machine 1 calls a procedure on machine 2, the calling process on 1 is suspended and execution of the called procedure takes place on 2. Information can be transported from the caller to the callee in the parameters and can come back in the procedure result. No message passing is visible to the programmer. This technique is known as RPC (Remote Procedure Call) and has become the basis for many networking applications. Traditionally, the calling procedure is known as the client and the called procedure is known as the server, and we will use those names here too.

The idea behind RPC is to make a remote procedure call look as much as possible like a local one. In the simplest form, to call a remote procedure, the client program must be bound with a small library procedure, called the client stub, that represents the server procedure in the client's address space. Similarly, the server is bound with a procedure called the server stub. These procedures hide the fact that the procedure call from the client to the server is not local.

The actual steps in making an RPC are shown in Fig. 6-24. Step 1 is the client calling the client stub. This call is a local procedure call, with the parameters pushed onto the stack in the normal way. Step 2 is the client stub packing the parameters into a message and making a system call to send the message. Packing the parameters is called marshaling. Step 3 is the kernel sending the message from the client machine to the server machine. Step 4 is the kernel passing the incoming packet to the server stub. Finally, step 5 is the server stub calling the server procedure with the unmarshaled parameters. The reply traces the same path in the other direction.