



Microsoft 70-226

**Designing Highly Available Web  
Solutions with Microsoft Windows  
2000 Server Technologies**

Study Guide  
DEMO Version

Copyright (c) 2003 Chinatag LLC. All rights reserved.

## **Important Note Please Read Carefully**

For demonstration purpose only, this free version Chinatag study guide contains **10** full length questions selected from our full version products which have more than **200** questions each.

This Study guide has been carefully written and compiled by Chinatag certification experts. It is designed to help you learn the concepts behind the questions rather than be a strict memorization tool. Repeated readings will increase your comprehension.

For promotion purposes, all PDF files are **not** encrypted. Feel free to distribute copies among your friends and let them know Chinatag website. Our IT certification products start at only **\$7.99**.

## **Study Tips**

This product will provide you questions and answers along with detailed explanations carefully compiled and written by our experts. Try to understand the concepts behind the questions instead of cramming the questions. Go through the entire document at least twice so that you make sure that you are not missing anything.

## **Latest Version**

We are constantly reviewing our products. New material is added and old material is revised. Free updates are available for 180 days after the purchase. You should check the products page on the <http://www.chinatag.com> website for an update 3-4 days before the scheduled exam date.

Please tell us what you think of our products. We appreciate both positive and critical comments as your feedback helps us improve future versions. Feedback on specific questions should be send to [feedback@chinatag.com](mailto:feedback@chinatag.com).

Thanks for purchasing our products and look forward to supplying you with all your Certification training needs.

Good studying!

Technical and Support Team  
Chinatag LLC.

## TABLE OF CONTENTS

### Introduction

#### 1. Introduction

- 1.1 The Web Computing Model
- 1.2 System Availability
  - 1.2.1 Downtime
- 1.3 Strategies that Ensure a Highly Available Web Site Design
  - 1.3.1 Creating and Documenting Operational Processes
  - 1.3.2 Catering For Sufficient Web Site Capacity
  - 1.3.3 Decreasing the Likelihood of Failures
- 1.4 Strategies that Ensure a Highly Available Network Topology Design
  - 1.4.1 Redundant Components
  - 1.4.2 Redundant Paths
  - 1.4.3. Redundant Services
- 1.5 Design Elements for the TCP/IP Network
  - 1.5.1 Designing a Subnet Addressing Structure
  - 1.5.2 Designing DHCP Fault Tolerant Server(s)
    - 1.5.2.1 DHCP Scopes, Superscopes, and Relay Agents
    - 1.5.2.2 Configuring DHCP Servers for Fault Tolerance
    - 1.5.2.3 DHCP Servers and Clustering
  - 1.5.3 Designing Name Resolution (DNS)
    - 1.5.3.1 DNS Structure
    - 1.5.3.2 Using Active Directory with DNS
    - 1.5.3.3 Setting up Namespace

#### 2. Designing Fault Tolerant Server Configurations and Data Storage for High Availability

- 2.1 Highly Available Hardware and Software Configurations
- 2.2. Designing Disk Fault Tolerance
  - 2.2.1 Software Based RAID
    - 2.2.1.1 RAID 1 / Mirroring
    - 2.2.1.2 RAID 5 / Disk Striping with Disk Parity
  - 2.2.2 Hardware Based RAID
  - 2.2.3 Storage Area Networks (SANs)



### **3. Windows 2000 Cluster Service and Network Load Balancing**

- 3.1 Windows 2000 Cluster Service
  - 3.1.1 Cluster Service Implementation Models and Drive Technologies
  - 3.1.2 Objects Managed by Cluster Service
    - 3.1.2.1 Networks and Network Interfaces
    - 3.1.2.2 Cluster Nodes
    - 3.1.2.3 Resource Groups and Resources
  - 3.1.3 Planning Considerations for a Cluster Implementation
- 3.2 Windows 2000 Network Load Balancing (NLB)
  - 3.2.1 The Convergence Process
  - 3.2.2 The NLB Architecture
    - 3.2.2.1 NLB Port Rules
    - 3.2.2.3 Planning for NLB Installation

### **4. Microsoft Application Center 2000**

- 4.1 An Introduction to Application Center
- 4.2 Integrated NLB and Integrated CLB Load Balancing Options
- 4.3 Application Center Synchronization Capabilities
- 4.4 Architecture
- 4.5 Planning Considerations for Application Center Clusters
- 4.6 Designing and Implementing CLB Clusters

### **5. Active Directory**

- 5.1 Active Directory Logical Structure
- 5.2 Active Directory Physical Structure
- 5.3 Replication with Active Directory
- 5.4 Planning the Physical Structure of Active Directory Services

### **6. The Capacity Planning Process**

- 6.1 An Overview of Capacity Planning
  - 6.1.1 Traffic
    - 6.1.1.1 Client and Server Network Capacity



- 6.1.2 Performance
- 6.1.3 Availability and Scalability

- 6.2 Determining Costs per User
  - 6.2.1 Studying the Standard User
  - 6.2.2 Computing CPU Costs
  - 6.2.3 Computing Memory Costs
  - 6.2.4 Computing Disk Costs and Network Costs

- 6.3 Planning Network Capacity Requirements

## **7. Web Application Integration**

- 7.1 Locating the Components that Support Distributed Applications
- 7.2 Defining the Manner to Deploy and Synchronize Applications
- 7.3 Web Data Access Technologies
- 7.4 The Process of Integrating a Database in a Web Application
  - 7.4.1 Creating Permissions
  - 7.4.2 Methods for Optimizing Database Connections
  - 7.4.3 Managing Data
  - 7.4.4 Scaling out the Database for High Availability and Performance
- 7.5 Planning for a Microsoft Exchange Web Integration Implementation

## **8. Designing Security, Monitoring and Disaster Recovery Strategies**

- 8.1 IIS Client Authentication Methods
  - 8.1.1 Anonymous Access
  - 8.1.2 Basic Authentication
  - 8.1.3 Integrated Windows Authentication
  - 8.1.4 Digest Authentication
  - 8.1.5 Client Certificate Mapping
- 8.2 Authorizing Users to Access Web Resources
  - 8.2.1 IIS Permissions
  - 8.2.2 NTFS Permissions
- 8.3 Data Encryption Technologies
  - 8.3.1 SSL
  - 8.3.2 IPsec
  - 8.3.3 EFS
- 8.4 Implementing Firewall Security



- 8.4.1 Data Filtering
- 8.4.2. Proxy Servers
- 8.4.3 Network Address Translation (NAT)
- 8.4.4 Perimeter Networks

## 8.5 System Monitoring

- 8.5.1 Tools Available for Performance Monitoring
  - 8.5.1.1 The Task Manager
  - 8.5.1.2 The Performance Tool
  - 8.5.1.3 The Windows Management Instrumentation (WMI)
  - 8.5.1.4 The Event Viewer
- 8.5.2 Monitoring Performance on Components of the System
  - 8.5.2.1 Memory
  - 8.5.2.2 Processor Performance
  - 8.5.2.3 Network Input/Output
  - 8.5.2.4 Web Applications
  - 8.5.2.5 Security Overhead

## 8.6 Disaster Recovery

## 8.7 System Auditing

# 70-226 MCSE: Designing Highly Available Web Solutions with Microsoft Windows 2000 Server Technologies

**Exam Code: 70-226**

## **Certifications:**

**Microsoft Certified Systems Engineer (MCSE)**

**Design Elective**

## **Prerequisites:**

Exam 70-215

Exam 70-216

It is recommended that you have some level of expertise on network technologies; and implementing, configuring and administering operating systems, especially Internet Information Services (IIS) and Windows 2000 Server.

## **About This Study Guide**

This Study Guide is based on the exam questions for the 70-226 MCSE: Designing Highly Available Web Solutions with Microsoft Windows 2000 Server Technologies exam. It presents all the information necessary to pass the 70-226 Designing Highly Available Web Solutions with Microsoft Windows 2000 Server Technologies exam, and is detailed on the particular proficiencies examined in the exam. The Designing Highly Available Web Solutions with Microsoft Windows 2000 Server Technologies exam is designed to test your knowledge on designing cluster and server architectures; application and service infrastructures; and security strategies for Web solutions. The information provided in this Study Guide is specific to the 70- 226 examination, and does not symbolize a complete reference work on the any other areas covered by the MCSE series of exams.

The following topics are covered in this Study Guide: The Web Computing Model, System Availability, Strategies that Ensure a Highly Available Web Site Design, Strategies that Ensure a Highly Available Network Topology Design, Design Elements for the TCP/IP Network, Highly Available Hardware and Software Configurations, Designing Disk Fault Tolerance, Windows 2000 Cluster Service, Windows 2000 Network Load Balancing (NLB), Microsoft Application Center 2000, Integrated NLB and Integrated CLB Load Balancing Options, Designing and Implementing CLB Clusters, Active Directory Logical Structure, Replication with Active Directory, Planning the Physical Structure of Active Directory Services, An Overview of Capacity Planning, Determining Costs per User, Planning Network Capacity Requirements, Web Application Integration, The Process of Integrating a Database in a Web Application, IIS Client Authentication Methods, Authorizing Users to Access Web Resources, Data Encryption Technologies, Implementing Firewall Security, System Monitoring, Disaster Recovery, System Auditing



**Intended Audience**

This Study Guide is targeted specifically for those individuals that need to be able to design, implement, configure and administer a Web solution with Windows 2000 and IIS, who desire to complete the Designing Highly Available Web Solutions with Microsoft Windows 2000 Server Technologies exam. The information in this Study Guide is specific to that exam. It is not a complete reference work on the other exam topics relating to the MCSE qualification. Although our Study Guides are aimed at new comers to the world of IT, the concepts dealt with in this Study Guide are complex.

**How To Use This Study Guide**

To benefit from this Study Guide we recommend that you:

- Study each chapter carefully until you fully understand the information. This will require regular and disciplined work. Where possible, attempt to implement the information in a lab setup.
- Be sure that you have studied and understand the entire Study Guide before you take the exam.

Good luck!

## 1. Introduction

**Microsoft Windows 2000 Server** is best suited for organizations that are small to medium in size. These organizations typically have different branch locations. Microsoft Windows 2000 Server provides the infrastructure; file, print and the Web services to the organization and its branches.

**Microsoft Windows 2000 Advanced Server** on the other hand is best suited for enterprises that are medium to large in size, and for Internet Service Providers (ISPs). These large corporate enterprises and ISPs often have extensive data warehouses, e-commerce, online transaction processing (OLTP) and Web hosting services that needs the integrated clustering infrastructure that Windows 2000 Advanced Server provides. Windows 2000 Advanced Server ensures the **high availability** and **scalability** of these organization's **mission critical** applications and services by providing functionality such as **Cluster Service**, **Network Load balancing (NLB)**; and up to **8 GB** main memory support on Intel **Page Address Extension (PAE)** systems and eight-way **Symmetric multiprocessing (SMP)**.

Microsoft Windows 2000 Datacenter Server is the solution for large enterprises that require better availability and reliability than what Windows 2000 Advanced Server provides. Windows 2000 Datacenter Server provides the same services and operations as Windows 2000 Advanced Server.

### 1.1 The Web Computing Model

The Web computing model has progressed into loosely connected tiers of varied applications and data located on a combination of flexible hardware platforms. The hardware platforms are not restricted to two computing tiers. This led to the **Web computing model providing availability, reliability, scalability and manageability**. Availability is an assessment of the fault tolerance of a computer and its associated programs. It evaluates or measures the operations related to a service and determines if the particular service is operating as it should be. Reliability can be defined as an assessment of the time duration amid system failures, while scalability assesses the network's expansion ability.

The architecture that strives to meet business computing requirements should therefore provide availability, reliability, scalability, be manageable, and ensure sufficient security for the enterprise's data as well as infrastructure. Security domains need to be implemented to ensure constant security. A simple model of security for a site typically entails the establishment of one or multiple firewall systems that scrutinize network traffic. A firewall is made up of routers or secure servers that monitor and block network traffic according to predefined rules. The **perimeter network** is the area of the network that **resides amid the firewalls**. Needless to say, there are many security mechanisms that can be implemented.

Web site management turns out to be more important when systems scale and change quite rapidly. Management systems consist of management servers, management consoles and management agents. The management servers monitor the servers being managed, while the management consoles are utilized to access the servers being managed. Management servers are also regarded as monitoring servers. The management agents are located on a network device. These agents carry out management operations on these devices. For high availability, management systems normally reside on a different network.

The architectural components associated with an n-tier business Web site can be summarized as:



- **Clients** basically transmit their requests to a server on which the application they using resides. Clients typically perceive only the:
  - Uniform Resource Locator (URL) and
  - any related hyperlinks and forms
- **Front-end Systems** are servers that supply File Transfer Protocol (FTP), Hypertext Transfer Protocol HTTP, and Hypertext Transfer Protocol Secure HTTPS services, or the requested Web pages to the client. Front-end systems normally do not hold client information over a session. That is why they are regarded as being **stateless**. The servers operating as **front-end systems** are called **clusters or Web farms**. Web farms, can generally connect to common file shares, or database systems residing on the back-end systems.
- **Back-end System** includes those servers that store the data utilized by the front-end systems. The data is stored in flat files, applications or database servers.

## 1.2 System Availability

Availability is typically computed with the Mean Time to Failure (MTTF) and Mean Time to Recovery (MTTR) metrics:

- **Mean Time to Failure (MTTF):** This metric can be described as the mean time to when a device fails. When the MTTF is identifiable, one is able to foresee when a device is going to fail. Most hardware mechanisms or components contain an **exponential failure distribution**. Basically, the longer the time duration that a hardware device functions, it can be expected to fail more constantly. The MTTF metric only works well when devices have an exponential failure rate. Because hardware and software elements have unlike failure qualities, it can be quite intricate to determine when software failures are going to occur. This is why the MTTF metric is relevant to certain types of software failures. There are three phases to devices' life cycle with every phase being categorized by a particular behavior. The phases are listed below:
  - **Burn-in:** At this phase, failures occur quite often. Burn-in failures typically decrease quite fast as well.
  - **Normal Aging:** Devices rarely fail in this phase. A device's attributes can be monitored once in the normal aging phase to pinpoint behavior associated with defined rates of failure. The failure rate of devices can also be monitored and followed so that they can be swapped before the failure mode phase.
  - **Failure Mode:** Failures tend to increase more rapidly as the devices lifespan increases.

Clustering can be implemented to decrease MTTF.

- **Mean Time to Recovery (MTTR):** This metric can be described as the mean time to recover from downtime or a failure. The downtime percentage is determined with the following calculation:

$$MTTR \div MTTF$$

Because availability is typically computed with the MTTF and MTTR metrics, it can be defined as a scale extending from 100 percent – 0 percent. The availability percentage is therefore determined with the following calculation:



$$\text{MTTF} \div (\text{MTTR} + \text{MTTF})$$

### 1.2.1 Downtime

Organizations generally strive for 99.9 percent or three 9s unscheduled annual downtime for Web sites managed by them. This works out to roughly 8.75 hours of unscheduled annual downtime. The other categories of 9's are listed below:

- **Five 9s:** 99.999 percent = 5.3 minutes of unscheduled annual downtime
- **Four 9s:** 99.99 percent = 53 minutes of unscheduled annual downtime
- **Two 9s:** 99 percent = 3.7 days of unscheduled annual downtime

The following issues initiate downtime:

- **Hardware Failures:** Hardware failures can be described as failures that occur in mechanisms like disks or storage media. Hardware failures tend to offset other failures. It is recommended to utilize platforms that can monitor internal temperatures, as well as trigger alarms accordingly. With random access memories, error correcting codes (ECCs) can be utilized to identify and correct single errors and to identify two-bit errors.
- **Software Failures:** Determining the reason of a system outage can be quite intricate. Virus protection defects can cause system outages. Often, incorrect system configuration can also lead to system failures.
- **Network Failures:** Any changes to the network design or topology of a layer of the protocol stack can have an impact on the entire network. It is therefore better to assess each layer when making any network changes. Remember that the performance of the network has an effect on the performance of the system.
- **Operational Failures:** Utilizing stringent operational processes will greatly reduce operational failures. Backup strategies and processes should be defined and implemented.
- **Environmental failures** are failures that result in data loss or service loss that are brought about by power outages caused by disasters like hurricanes and snowstorms.

When monitoring the availability of a site, include the following:

- **System Availability:** The normal and abnormal shutdowns of the Operating System (OS) should be monitored
- **Network Availability:** Network Internet Control Message Protocol (ICMP) echo pings can be utilized to monitor network availability.
- **Bandwidth usage** should be monitored during peak and idle times.
- **Performance metrics** such as CPU utilization, memory; disk stage and disk input/output (I/O) should be utilized to make certain that availability is high.
- HTTP requests should also be monitored to ensure **HTTP availability**.



### 1.3 Strategies that Ensure a Highly Available Web Site Design

The strategies for designing a highly available Web site can be broken down into the following phases:

- **Creating, and documenting operational processes**
- **Catering for sufficient site capacity to deal with processing loads**
- **Decreasing the likelihood of a failure occurring.**

#### 1.3.1 Creating and Documenting Operational Processes

Creating, as well as documenting operational processes can assist in securing the availability of the Web site. Operational processes should entail the following:

- **Availability** management
- **Capacity** management
- **Change** management
- **Problem** management
- **Security** management and
- **Service level** management

Thorough, regular monitoring processes are vital when implementing and configuring systems for high availability. It is important to limit both physical and logical access to servers, and to examine the system event logs being generated by devices. Any devices that are included in the Hardware Compatibility List (HCL) must utilize event logs. It is good practice to implement automated processes to deal with alarm warnings. Administrators should also be knowledgeable on the advantages and risks associated with system upgrades. Redundant Array of Independent Disks (RAID) systems can be implemented to improve disk system performance and reliability.

#### 1.3.2 Catering For Sufficient Web Site Capacity

Planning for site capacity should include factors such as scaling the Web farm to cater for additional site traffic. It is important to ensure the performance of the Web site during extended peak time periods.

#### 1.3.3 Decreasing the Likelihood of Failures

The following strategies can assist in decreasing **server failures**:

- It is good practice to implement redundant load balancing servers.

The following strategies can assist in decreasing **hardware failures**:

- Hardware based RAID and dual disk controllers can be utilized for fault tolerance.
- When deploying Fibre Channel SAN, be sure to utilize redundant Fibre Channel host bus adapters, fabric switches or hubs. To ensure availability, reduce any single points of failure in the SAN.



The following strategies can assist in decreasing **climate control failures**:

- When it comes to the temperature of the hardware, do not stray from the hardware vendor's recommended temperature specifications. Too much humidity can result in electrical short circuits. Too much aridity can also be harmful to devices.

The following strategies can assist in decreasing **electrical failures**:

- Utilize an uninterruptible power supply (UPS) that has a power rating which matches the network mechanisms.
- Power generators can be utilized as a backup to the UPS.

The following strategies can assist in decreasing **network failures**:

- Utilize more than one Internet Service Provider
- Utilize several Local Area Networks (LANs), routers, firewalls, switches and Network Interface Cards (NICs).

The following strategies can assist in decreasing **applications failures**:

- Architecture should be robust and should comprise of load balancing and redundant servers.
- Code should be examined to safeguard against security loopholes, code crashes, and infinite loops.

The following strategies can assist in decreasing **data failures**:

- Perform constant backups that can be utilized to restore data when necessary.
- Log shipping to a warm backup server can maintain a disaster recovery site. This involves replaying transaction logs against a recognized suitable database.
- Failover clustering technologies such as Cluster Service can be utilized for back-end database servers. Clustering is suited for dynamic data. To safeguard against SQL Server failures, utilize clustering.
- The Active Directory service makes use of data replication to supply redundancy for static data. Active Directory can also supply authentication services. Active Directory stores should be backed up while it is online. A minimum of two domain controllers should be implemented at every physical site when utilizing Active Directory. This implementation strategy can also assist in reducing downtime associated with restoring from backups.

The following strategies can assist in decreasing **security failures**:

- Utilize several firewalls and implement Intrusion Detection Systems (IDSs).

#### **1.4 Strategies that Ensure a Highly Available Network Topology Design**

Redundant components, paths and services can be included in the network topology to steer clear of single points of failure, and to ensure a highly available topology.

### 1.4.1 Redundant Components

Redundant components in this chapter pertain to the following network components outside a computer:

- **Routers:** Routers are devices functioning at the network layer that link networks utilizing diverse architectures and protocols. Routers are capable of transmitting protocol information among different networks, and can route packets over many networks. Routers also calculate the best path for forwarding data. Routers do not fail all that often but it is extremely important to have a redundant routing capability at every single point of failure associated with a router. This in turn ensures a highly available network topology and reduces the possibility of a Web site being unavailable.
- **Hubs** are devices that supply a common connection to the network devices. They connect communication lines or wiring at a central location and broadcast traffic to ports. Active hubs are capable of retransmitting data. For a hub to do this, it needs electrical power. Passive hubs are hubs which merely set up wiring. Redundancy should be implemented even though hubs are extremely reliable.
- **Switches** are network devices that administer or manage routing functions. When utilized in clustering, switches are utilized to connect the network interface of the cluster node to the incoming network connection, such as a router. A switch is also utilized to create a path between two ports on the switch rather than broadcasting to all ports. Because multilayer switches supply the main network switching of numerous Web sites, they have to support among others, redundant power supplies, fast fault recovery, Quality of Service (QoS), a large quantity of user connections; and should also be capable of providing Layer 2 and Layer 3 switching. Utilizing redundant switches would ensure redundant paths of network connectivity even though switches are extremely reliable.

### 1.4.2 Redundant Paths

Redundant paths can be established in various points in the network topology:

- **LANs:** A LAN can be defined as a network which connects devices, and computers in a location such as a building, while also enabling its users to share network resources and storage devices. A network device that is connected to the LAN typically communicates with other network devices connected to the same LAN. Because LANs in a multi-tiered network are normally segmented into subnets, redundant switches should reside in every LAN subnet to ensure redundant LAN connectivity.
- **Multiple Sites:** Organizations can also utilize a multi-site architecture which comprises of implementing the identical service at two different locations or over two sites. A multi-site architecture is made up of a core or main Web site with one or multiple satellite sites that can consist of a part of or the whole architecture of organization's core Web site. A **geographical load balancer** can be utilized to **regulate client connection requests** when a Web site consists of distributed sites. The geographic load balancer obtains DNS requests and in turn transmits the unique IP address of the Web site's data center. The geographical load balancer also ensures that no client connection requests are transmitted to a site that is considered failed.
- **ISPs** are organizations that supply the enterprise with Internet and Web access. Web sites normally make use of an additional Internet connection to minimize the possibility of a route failure.



### 1.4.3. Redundant Services

Redundant servers can be implemented to ensure high availability for mission critical applications and services utilized for e-commerce or financial transactions, among other functions. The various techniques of implementing redundant services to ensure a high available Web site are discussed below:

- **Backup Servers:** Backup servers are frequently utilized to ensure high availability.
  - A **hot standby system** can be implemented that has automatic failover capabilities so that it can immediately replace a failed system.
  - A **spare system** can also be implemented. The spare system replaces the primary system when it has a failure.
- **Clustering and Load Balancing:** There are many clustering and load balancing methods available such as Windows 2000 Cluster Service and Network Load balancing (NLB), as well as round robin DNS and load balancing switches. These techniques supply access to resources (applications and network services) on a cluster of servers while ensuring that the client requests are distributed between the cluster's servers.
  - **Windows 2000 Clustering:** A cluster can be defined as the grouping of two or multiple physical servers that is portrayed as, and operates as one network server. These servers provide redundancy to the enterprise network by resuming operations of a failed server within the cluster. **Servers in the cluster are typically referred to as nodes**, while services and applications are referred to as resources. Clustering **increases server availability** for key business applications and network services. The cluster can also be **configured to provide load balancing features** so that the client requests can be distributed between the cluster nodes. A network utilizing Cluster Service provides **improved scalability** because servers can be expanded while client access is still ensured. NLB is a clustering technology that provides **availability** and **scalability**. With NLB, **client requests are load balanced** according to the configured load balancing parameters. The **Wlbs.sys driver of NLB** is configured for each server in the cluster, and functions between the network adapter and the TCP/IP protocol. The driver manages and allocates client requests to a server in the cluster. With NLB there is no single point of failure purely because it is regarded as a **distributed application**. NLB is backward compatible with Windows Load Balancing Service (WLBS).
  - **Load Balancing Switches** can be implemented to manage and distribute TCP requests over a group of servers. The user sees the group of servers as one virtual server. End users are directed to the most available server. Server load balancers too can identify servers that have a failure. When this occurs, users are routed to another server. Server load balancing therefore enhances site availability and end user response times. Cisco's LocalDirector is an example of load balancing switches.
  - **DNS Round Robin** is hardly ever implemented alone in Web farms that host mission critical network resources because it cannot identify when any of the Web farm's servers has an extensive load or a failure. The technique is utilized with DNS servers to provide basic load balancing. When there are many of the identical type resource records (RR) for a requested or queried DNS domain name, DNS round robin rotates the RR data's order in its query reply until all the applicable data for a domain name has been moved and rotated to the top of query reply.

## 1.5 Design Elements for the TCP/IP Network

### 1.5.1 Designing a Subnet Addressing Structure

A host or network component that utilizes TCP/IP has to have a **unique IP address**. The IP address is a network layer address that **identifies the TCP/IP host**.

When hosts reside on the identical physical network, the identical **network ID** is allocated to these hosts. The network ID is utilized to **enable communication between the hosts**. The network ID should be unique to the IP internetwork or Internet, whichever is applicable. The network ID bits cannot be specified as 1s or 0s. 1s are reserved for IP broadcast addresses while 0s are utilized to indicate a non routed local host. In addition to the previously mentioned rules, a network ID that commences with 127 is reserved for loop-back (internal) operations.

A **host ID** is utilized to distinguish a host using TCP/IP and has to be unique as well. A host ID with bits specified as 1 is reserved for broadcast addresses, while a host ID with bits specified as 0 is a reserved to indicate the network ID. An IP address's 32 bits are split between the host ID and network ID. The network ID and host ID are combined to form the unique IP address of a TCP/IP host

Hosts that reside on the identical physical network receive the identical broadcast traffic when they are bounded by IP routers. An **IP network can be further divided into subnets** to establish smaller sized broadcast domains. Every subnet has to be assigned with a **unique subnetted network ID**. The unique subnetted network ID is developed from the host ID of the primary class based network ID. A node examines the subnet mask when determining the manner in which to obtain the network ID, whether it is class based or subnetted. The subnet mask identifies the actual network, the IP address host portion, and any related subnetwork. Subnet masks are 32 bit numbers with bits set to 1 or 0.

The procedure that should be utilized when implementing subnetting is outlined next. The first stage in implementing subnetting is to **decide on the host bits that are going to be utilized**. It is good practice to utilize more bits than what is currently needed. This eliminates the likelihood of reassigning IP addresses at a later stage. The host bits actually calculate or specify how much hosts and subnets are practical for each subnet. Growth related to the number of hosts are restricted when a high figure of host bits are utilized. This is true, even though a higher usage of host bits caters for subnet growth. It is also true that subnet growth is restricted when an insufficient quantity of hosts are utilized. It is therefore important to foresee the number of subnets and hosts that are going to be utilized at a later stage. The **second stage is specifying the new subnetted network IDs**. Whether the decimal or binary method is utilized for specifying the subnetted network ID, the results should be converted to the dotted decimal notation. The last stage involves **determining or specifying the IP addresses for every new subnetted network ID**. The binary or decimal method can be utilized for this task as well. It is more practical to specify the IP address range than to specify every IP address in isolation.

Systems that have the same security requirement are typically grouped into the same segment when a business site makes use of more than one domain segment. A domain segment is separated and protected by a firewall. The main domain segments are listed below:

- **Public Network**
- **Perimeter Network:** Content servers and front end servers reside in the perimeter network. A perimeter network can be further segmented:



- A segment should be utilized to implement a management network.
  - The various forms of Internet traffic such as HTTP and FTP should be routed to separate Web clusters.
  - Non routable network addresses should be assigned to the internal networks of the Web site.
  - Internet traffic should be separated from the internal network or back end traffic.
  - Ensure that IP forwarding is not enabled for the front end servers.
- **Secure Network:** Data is stored and maintained in the secure network. Content is also created in this segment.

### **1.5.2 Designing a Dynamic Host Configuration Protocol (DHCP) Fault Tolerant Server(s)**

DHCP makes it possible for IP addresses to be **dynamically** assigned. DHCP too provides the TCP/IP configuration information and any other data needed for particular servers. DHCP obtains the IP addresses it assigns to a DHCP client from its DHCP database. The DHCP client typically receives the IP address and IP configurations from the DHCP servers for a specified lease period. This period is referred to as the **DHCP lease**.

The DHCP lease procedure initiates when the following occurs:

- A DHCP client uses TCP/IP for the first time
- A DHCP client no longer has its prior DHCP lease and needs to request a new DHCP lease.
- A client specifically requested an IP address that the DHCP server cannot assign to the client.

The DHCP lease procedure has the following stages:

- **DHCPDISCOVER:** The DHCP lease procedure is set off when a client transmits a DHCPDISCOVER message. The DHCPDISCOVER message is a request for the DHCP server's location and for additional IP addressing information, and includes the hardware address and computer name of the client. The DHCP server uses the hardware address and computer name details to identify the client that broadcasted the DHCPDISCOVER message.
- **DHCPOFFER:** The DHCP server(s) who obtained the broadcasted request message, that do possess IP configuration information, now broadcast the DHCPOFFER message. The message is broadcasted because the client at this point does not possess an IP address. The client chooses the IP address from the earliest DHCPOFFER message it receives.

The following details are included in the DHCPOFFER message broadcasted by the DHCP server(s)

- The offered **IP address** and **subnet mask**: The DHCP server reserves this information in the meantime for the client. This prevents the identical IP address from being offered to another client.
  - The duration or **period of the lease**.
  - The **IP address of the DHCP server** (server ID)
  - The **hardware address of the client**.
- **DHCPREQUEST:** The DHCPREQUEST message is broadcasted by the client to all DHCP servers. This message specifies that the client has chosen an IP address from a DHCPOFFER message it

obtained. The broadcasted **DHCPREQUEST** message holds the IP address of the DHCP server (server ID), from whom the client accepted the IP addressing information. The remainder of the DHCP servers no longer continues to reserve their IP addresses offered to the client. These IP addresses are now offered to other clients.

- **DHCPACK:** The DHCPACK message is broadcasted from the DHCP server to the client. The message contains the lease details for the IP address as well as any other additional IP configuration information. TCP/IP communication is initiated once the client obtains the DHCPACK message. The client now becomes a DHCP client.
- **DHCPNACK:** The DHCPNACK message is broadcasted from the DHCP server to a client when a DHCPREQUEST is not successful. This happens when:
  - The client wants to lease a prior IP address that no longer exists OR
  - The IP address is no longer valid given that the client computer is now located in another subnet.

A client that receives a DHCPNACK makes use of Automatic Private IP Addressing (APIPA) to automatically configure its IP address until it can obtain IP addressing information from the DHCP server(s).

At the point when 50 percent of the DHCP lease period has passed, the DHCP client transmits a DHCPREQUEST message to the DHCP server that it received its lease from, in order to renew it. The DHCP lease is renewed when the original DHCP server is accessible. The lease renewal is transmitted in the form of a DHCPACK message. This message holds the new lease period and any other relevant configuration information that the DHCP client requires to update its current configuration. When the original DHCP server is unavailable at the time that the client transmitted the DHCPREQUEST message, the client broadcasts a DHCPREQUEST message when 87.50 percent of the DHCP lease period has passed to any available DHCP server. A DHCP server can either reply to the client with a DHCPACK message that renews the lease, or a DHCPNACK message that compels the DHCP client to request a lease for a different IP address.

### 1.5.2.1 DHCP Scopes, Superscopes, and Relay Agents

With DHCP, a scope has to be configured for every subnet. A scope contains a range of IP addresses and TCP/IP configuration information that is offered by the DHCP server to clients. A scope must be enabled on the DHCP server so that it can provide IP addresses to DHCP clients. Scopes have to be allocated with a name, a subnet mask, lease period parameters and the IP address range for the DHCP lease offers. Virtual server addresses and any other statically assigned device addresses should be excluded from the DHCP scope. Scopes make it possible for network computers and devices to receive the **same reserved IP addresses**. These addresses are then not distributed to any other DHCP client. This feature can be utilized for WINS, DNS and print servers, and for the IP addresses of cluster nodes (servers). Lease durations are defined at the scope level for clients that are dynamically assigned with an IP address.

Multiple scopes can also be combined to form one **Superscope**. Superscopes provide a few benefits such as permitting IP addresses to be renumbered; and offering IP address leases to DHCP clients on a physical network segment that has many logical subnets. Any additional hosts added to the network do not have an impact on existing scopes. Scopes that are added to a superscope are referred to as **member scopes**.

A **BOOTP/DHCP relay agent** is a router or a host that is set up to recognize and detect DHCP broadcast messages. The BOOTP/DHCP relay agent transmits the client broadcast message to the DHCP server(s). It also relays DHCP response messages to clients. BOOTP/DHCP relay agents therefore make it possible for broadcasts to be communicated between the DHCP server(s) and a client. When BOOTP/DHCP relay agents are utilized, there is no longer a need to configure a DHCP server on each network segment.

### 1.5.2.2 Configuring DHCP Servers for Fault Tolerance

An active/online DHCP server and backup/standby DHCP server can handle a huge number of client requests. The active DHCP server provides the DHCP functionality and features, while the backup/standby DHCP server, merely monitors the state of the active DHCP server. Each DHCP server stores the same DHCP database. Windows by default performs a backup of the DHCP database at 60 minute time intervals. The backup file is placed in %systemroot%\System32\Dhcp\Backup\Jet\New folder. This default setting can however be modified. In the event of a failure on the active DHCP server, the backup/standby DHCP server resumes to supply the DHCP services to clients.

The following points should be kept in mind when planning the implementation of a DHCP server(s) and when determining the number of DHCP servers needed for the network:

- Ensure that the memory and CPU capacity are sufficient to support the operations of the DHCP server(s), and the services the DHCP server(s) should provide.
- Determine whether a DHCP server is required in each subnet, based on router location. It is important to ensure that there is sufficient bandwidth available for the DHCP server.
- The transmission speed between segments is another relevant factor.
- Decide on the computers/devices that should have statically assigned IP addresses.
- Determine the IP addresses that should not be part of the IP addresses utilized by DHCP when it assigns IP addresses to clients.
- The 80/20 rule should be utilized to divide the IP address pool among the DHCP servers to ensure that the IP addressing information assigned to clients are unique and are not duplicated.

### 1.5.2.3 DHCP Servers and Clustering

Cluster Service in Windows 2000 and Windows 2000 Datacenter can be utilized for the DHCP servers. Cluster Service ensures high availability and scalability for the DHCP servers because it provides **redundancy** for network failures. With Cluster Service, another server automatically resumes the services of a failed server in the cluster. This increases server availability for mission critical business applications and network services. Cluster Service also reduces downtime associated with **scheduled maintenance** downtime. When a server in the cluster is scheduled for an upgrade, its services / applications can be manually shifted to another node in the cluster. Cluster Service therefore provides high availability by not disrupting client access when upgrading servers, applications and resources. Rolling upgrades can also be performed with Cluster Service. When Cluster Service is utilized for clustering DHCP servers, it creates a virtual DHCP server. A virtual server makes it possible for clients to access the DHCP services via the identical IP address and NetBIOS name. With virtual servers, clients are not dependent on the actual DHCP server that is currently managing the services. A cluster can have multiple virtual servers that each has their

own individual IP address and NetBIOS name. Utilizing Cluster Service removes the need of dividing scopes.

### 1.5.3 Designing Name Resolution (DNS)

#### 1.5.3.1 DNS Structure

DNS, the name resolution service utilized in Windows 2000 environments, is a reliable and scalable database. DNS is also a hierarchical, distributed database. DNS 2000 can be integrated with Active Directory. Windows 2000 offer a DNS Server implementation that can operate with other standard based DNS Server implementations. DNS is utilized as well for locating domain controllers for logon. The **DNS** database store domain names in a hierarchical tree structure, referred to as the **domain space**. A domain name that has a format consisting of unique labels that is divided by dots is called the **Fully Qualified Domain Name (FQDN)**. The FQDN distinguishes the position or location of the host in the DNS hierarchical tree.

The DNS database contains various resource records (RRs) that basically identify a certain resource held in the DNS database. A few of the common RRs and their data options that are part of the Internet (IN) RRs' class are listed below:

- **A (Host):** Owner Name and Host IP Address
- **CNAME (Canonical Name):** Owner Name and Host DNS Name
- **MX (Mail Exchanger):** Owner Name, Mail Exchange Server DNS Name and Preference Number
- **NS (Name Server):** Owner Name and Name Server DNS Name
- **SOA (State of Authority):** Expire Time, Minimum Time to Live (TTL), Owner Name, Primary Name Server DNS Name, Refresh Interval, Retry Interval and Serial Number

The DNS database can be divided into multiple partitions. These partitions are called zones. It is possible to **configure a DNS server to have one, multiple, or no zones**. A zone can be described as a segment of the DNS database that has RRs that go with the DNS namespace's adjacent segment. Each zone has a root domain and holds information on the domain names that end with the root domain name of the zone. Zone files are retained and maintained on the DNS servers. A zone file's initial record is the State of Authority RR which pinpoints the zone's primary DNS name server. The primary DNS name server is regarded as the optimum source of information for data contained in the particular zone.

When multiple zones are configured, the identical portion of the namespace can be symbolized by these zones. With multiple zones, there is a primary zone file in which all zone updates are performed. Any modifications made to the primary zone file are replicated to the secondary zone file. The secondary zone file is only a copy of the primary zone file and is therefore read-only. Because multiple zones can reside on a name server, the server can hold a master copy associated with one zone file, as well as the secondary zone file of another zone. A zone file can be replicated to many name servers. Replication occurs when the master server's zone file that is the source of the zone information is replicated or copied to the secondary server. This procedure just described is known as **zone transfer**.

With **Internet DNS**, the domain names comply with the **International standard 3166**. The current DNS domain names are listed below:



- **com** indicates commercial organizations
- **edu** indicates educational organizations
- **gov** indicates government organizations that are not military
- **mil** indicates government organizations that are military
- **net** is utilized for networks
- **num** indicates phone numbers
- **org** indicates non-profit institutions
- **arpa** is utilized for reverse DNS
- **xx** indicates the region code

### 1.5.3.2 Using Active Directory with DNS

When DNS is installed on a domain controller, Active Directory can be enabled to supply **storage and replication**. A primary zone can also be stored in Active Directory. This makes it possible for zone information to be replicated with Active Directory replication to all the domain controllers. In addition to this, Active Directory enables each domain controller for a specific domain to **update** the zone; and to transmit or receive updates on the information that Active Directory holds. Information can therefore be replicated to the remainder of the domain controllers. This is known as **multimaster replication**. Using Active Directory leads to increased fault tolerance, effective replication when the zones are large and simplified management. There is no single point of failure for zone updates when Active Directory replication is enabled.

### 1.5.3.3 Setting up Namespace

An organization on the Internet has to have public namespace that is accessible to Internet users. Private namespace should be utilized for internal clients, that is, users within the organization. DNS servers should be set up to permit internal clients to resolve names within the public namespace and private namespace.

Factors relevant to namespace are listed below:

- A private root can be utilized when every client has:
  - A **name exclusion list** which holds internal DNS suffixes
  - A **Proxy AutoConfiguration (PAC) file** which holds internal or external DNS suffixes and names.
- The DNS server that hosts the internal domain has to forward queries to the Internet when an organization has no proxy capabilities or a Local Address Table (LAT) is being utilized.
- It is good practice to utilize one domain name for the internal namespace and another separate name for the external namespace. Although it is not recommended, the identical domain name can be utilized for the internal and external namespace. This can however lead to additional administrative overhead and configuration issues.

