**6. Describe the main differences between IPv4 and IPv6. Provide information for the important improvements in IPv6.**

**What is Internet Protocol - IP?**

IP (short for [Internet Protocol](http://www.webopedia.com/TERM/I/IP.html)) specifies the technical format of [packets](http://www.webopedia.com/TERM/P/packet.html) and the addressing scheme for computers to communicate over a [network](http://www.webopedia.com/TERM/N/network.html). Most networks combine IP with a higher-level protocol called [Transmission Control Protocol (TCP)](http://www.webopedia.com/TERM/T/TCP.html), which establishes a virtual connection between a destination and a source.   
  
IP by itself can be compared to something like the postal system. It allows you to address a package and drop it in the system, but there's no direct link between you and the recipient. [TCP/IP](http://www.webopedia.com/TERM/T/TCP_IP.html), on the other hand, establishes a connection between two [hosts](http://www.webopedia.com/TERM/H/host.html) so that they can send messages back and forth for a period of time.   
  
There are currently two version of Internet Protocol (IP): **IPv4** and a new version called **IPv6*.*** IPv6 is an evolutionary [upgrade](http://www.webopedia.com/TERM/U/upgrade.html) to the Internet Protocol. IPv6 will coexist with the older IPv4 for some time.

**What is IPv4 - Internet Protocol Version 4?**

IPv4 (Internet Protocol Version 4) is the fourth revision of the Internet Protocol (IP) used to to identify [devices](http://www.webopedia.com/TERM/D/device.html) on a [network](http://www.webopedia.com/TERM/N/network.html) through an addressing system. The Internet Protocol is designed for use in interconnected systems of packet-switched computer communication networks.  
IPv4 is the most widely deployed Internet protocol used to connect devices to the Internet. IPv4 uses a [32-bit](http://www.webopedia.com/TERM/3/32_bit.html) address scheme allowing for a total of 2^32 addresses (just over 4 billion addresses).  With the growth of the [Internet](http://www.webopedia.com/TERM/I/Internet.html) it is expected that the number of unused IPv4 addresses will eventually run out because every device -- including computers, smartphones and game consoles -- that connects to the Internet requires an address.

A new Internet addressing system Internet Protocol version 6 (IPv6) is being deployed to fulfill the need for more Internet addresses.

**What is IPv6 - Internet Protocol Version 6?**

IPv6 (Internet Protocol Version6) is also called IPng (Internet Protocol next generation) and it is the newest version of the Internet Protocol (IP) reviewed in the [IETF](http://www.webopedia.com/TERM/I/IETF.html) standards committees to replace the current version of IPv4 (Internet Protocol Version 4).   
  
IPv6 is the successor to Internet Protocol Version 4 (IPv4). It was designed as an evolutionary upgrade to the Internet Protocol and will, in fact, coexist with the older IPv4 for some time. IPv6 is designed to allow the Internet to grow steadily, both in terms of the number of hosts connected and the total amount of data traffic transmitted.  
  
IPv6 is often referred to as the ["next generation" Internet standard](http://www.enterprisenetworkingplanet.com/reports/article.php/1347761/IPv6:+What+You+Need+to+Know.htm) and has been under development now since the mid-1990s. IPv6 was born out of concern that the demand for IP addresses would exceed the available supply.

While increasing the pool of addresses is one of the most often-talked about benefit of IPv6, there are other important technological changes in IPv6 that will improve the IP protocol:

* No more NAT (Network Address Translation)
* Auto-configuration
* No more private address collisions
* Better multicast routing
* Simpler header format
* Simplified, more efficient routing
* True quality of service (QoS), also called "flow labeling"
* Built-in authentication and privacy support
* Flexible options and extensions
* Easier administration (say good-bye to DHCP)

**The Difference Between, IPv6 and IPv4 IP Addresses**

An [IP address](http://www.webopedia.com/TERM/I/IP_address.html) is [binary](http://www.webopedia.com/TERM/B/binary.html) numbers but can be stored as text for human readers.  For example, a 32-bit numeric address (IPv4) is written in decimal as four numbers separated by periods. Each number can be zero to 255. For example, **1.160.10.240** could be an IP address.

IPv6 addresses are 128-bit IP address written in hexadecimal and separated by colons. An example IPv6 address could be written like this: **3ffe:1900:4545:3:200:f8ff:fe21:67cf**.

**6. Find and document as much as possible for the IPv6 stateless and stateful Auto-Configuration.**

An important feature of IPv6 is that it allows plug and play option to the network devices by allowing them to configure themselves independently. It is possible to plug a node into an IPv6 network without requiring any human intervention. This feature was critical to allow network connectivity to an increasing number of mobile devices.   
  
The proliferation of network enabled mobile devices has introduced the requirements of a mobile device to arbitrarily change locations on an IPv6 network while still maintaining its existing connections. To offer this functionality, a mobile device is assigned a home address where it remains always reachable. When the mobile device is at home, it connects to the home link and makes use of its home address. When the mobile device is away from home, a home agent (router) acts as a conduit and relays messages between the mobile device and other devices on the network to maintain the connection. IPv6 offers two types of auto-configuration: Stateful auto configuration and stateless auto configuration.   
  
Stateful auto-configuration: This configuration requires some human intervention as it makes use of the Dynamic Host Configuration Protocol for IPv6 (DHCPv6) for installation and administration of nodes over a network. The DHCPv6 server maintains a list of nodes and the information about their state to know the availability of each IP address from the range specified by the network administrator.

**ORACLE**

# IPv6 Stateless Address Autoconfiguration

A host performs several steps to autoconfigure its interfaces in IPv6. The autoconfiguration process creates a link-local address. The autoconfiguration process verifies its uniqueness on a link. The process also determines which information should be autoconfigured, addresses, other information, or both. The process determines if the addresses should be obtained through the stateless mechanism, the stateful mechanism, or both mechanisms. This section describes the process for generating a link-local address. This section also describes the process for generating site-local and global addresses by stateless address autoconfiguration. Finally, this section describes the procedure for duplicate address detection.

## Stateless Autoconfiguration Requirements

IPv6 defines mechanisms for both stateful address and stateless address autoconfiguration. Stateless autoconfiguration requires no manual configuration of hosts, minimal (if any) configuration of routers, and no additional servers. The stateless mechanism enables a host to generate its own addresses. The stateless mechanism uses local information as well as non-local information that is advertised by routers to generate the addresses. Routers advertise prefixes that identify the subnet or subnets that are associated with a link. Hosts generate an **interface identifier** that uniquely identifies an interface on a subnet. An address is formed by combining the prefix and the interface identifier. In the absence of routers, a host can generate only link-local addresses. However, link-local addresses are only sufficient for allowing communication among nodes that are attached to the same link.

## Stateful Autoconfiguration Model

In the stateful autoconfiguration model, hosts obtain interface addresses or configuration information and parameters from a server. Servers maintain a database that checks which addresses have been assigned to which hosts. The stateful autoconfiguration protocol allows hosts to obtain addresses and other configuration information from a server. Stateless and stateful autoconfiguration complement each other. For example, a host can use stateless autoconfiguration to configure its own addresses, but use stateful autoconfiguration to obtain other information.

## When to Use Stateless and Stateful Approaches

The stateless approach is used when a site is not concerned with the exact addresses that hosts use. However, the addresses must be unique. The addresses must also be properly routable. The stateful approach is used when a site requires more precise control over exact address assignments. Stateful and stateless address autoconfiguration can be used simultaneously. The site administrator specifies which type of autoconfiguration to use through the setting of appropriate fields in router advertisement messages.

IPv6 addresses are leased to an interface for a fixed, possibly infinite, length of time. Each address has an associated lifetime that indicates how long the address is bound to an interface. When a lifetime expires, the binding, and address, become invalid and the address can be reassigned to another interface elsewhere. To handle the expiration of address bindings gracefully, an address experiences two distinct phases while the address is assigned to an interface. Initially, an address is preferred, meaning that its use in arbitrary communication is unrestricted. Later, an address becomes **deprecated** in anticipation that its current interface binding becomes invalid. When the address is in a deprecated state, the use of the address is discouraged, but not strictly forbidden. New communication, for example, the opening of a new TCP connection, should use a preferred address when possible. A deprecated address should be used only by applications that have been using the address. Applications that cannot switch to another address without a service disruption can use a deprecated address.

## Duplicate Address Detection Algorithm

To ensure that all configured addresses are likely to be unique on a particular link, nodes run a **duplicate address detection** algorithm on addresses. The nodes must run the algorithm before assigning the addresses to an interface. The duplicate address detection algorithm is performed on all addresses.

The autoconfiguration process that is specified in this document applies only to hosts and not routers. Because host autoconfiguration uses information that is advertised by routers, routers need to be configured by some other means. However, routers probably generate link-local addresses by using the mechanism that is described in this document. In addition, routers are expected to pass successfully the duplicate address detection procedure on all addresses prior to assigning the address to an interface.

## Autoconfiguration Process

This section provides an overview of the typical steps that are performed by an interface during autoconfiguration. Autoconfiguration is performed only on multicast-capable links. Autoconfiguration begins when a multicast-capable interface is enabled, for example, during system startup. Nodes, both hosts and routers, begin the autoconfiguration process by generating a link-local address for the interface. A link-local address is formed by appending the interface's identifier to the well-known link-local prefix.

A node must attempt to verify that a tentative link-local address is not already in use by another node on the link. After verification, the link-local address can be assigned to an interface. Specifically, the node sends a neighbor solicitation message that contains the tentative address as the target. If another node is already using that address, the node returns a neighbor advertisement saying that the node is using that address. If another node is also attempting to use the same address, the node also sends a neighbor solicitation for the target. The number of neighbor solicitation transmissions or retransmissions, and the delay between consecutive solicitations, are link specific. These parameters can be set by system management.

If a node determines that its tentative link-local address is not unique, autoconfiguration stops and manual configuration of the interface is required. To simplify recovery in this instance, an administrator can supply an alternate interface identifier that overrides the default identifier. Then, the autoconfiguration mechanism can be applied by using the new, presumably unique, interface identifier. Alternatively, link-local and other addresses need to be configured manually.

After a node determines that its tentative link-local address is unique, the node assigns the address to the interface. At this point, the node has IP-level connectivity with neighboring nodes. The remaining autoconfiguration steps are performed only by hosts.

**Obtaining Router Advertisement**

The next phase of autoconfiguration involves obtaining a router advertisement or determining that no routers are present. If routers are present, the routers send router advertisements that specify what type of autoconfiguration a host should perform. If no routers are present, stateful autoconfiguration is invoked.

Routers send router advertisements periodically. However, the delay between successive advertisements is generally longer than a host that performs autoconfiguration can wait. To obtain an advertisement quickly, a host sends one or more router solicitations to the all-routers multicast group. Router advertisements contain two flags that indicate what type of stateful autoconfiguration (if any) should be performed. A **managed address configuration** flag indicates whether hosts should use stateful autoconfiguration to obtain addresses. An **other stateful configuration** flag indicates whether hosts should use stateful autoconfiguration to obtain additional information, excluding addresses.

**Prefix Information**

Router advertisements also contain zero or more prefix information options that contain information that stateless address autoconfiguration uses to generate site-local and global addresses. The stateless address and stateful address autoconfiguration fields in router advertisements are processed independently. A host can use both stateful address and stateless address autoconfiguration simultaneously. One option field that contains prefix information, the **autonomous address-configuration** flag, indicates whether the option even applies to stateless autoconfiguration. If the option field does apply, additional option fields contain a subnet prefix with lifetime values. These values indicate how long addresses that are created from the prefix remain preferred and valid.

Because routers generate router advertisements periodically, hosts continually receive new advertisements. Hosts process the information that is contained in each advertisement as described previously. Hosts add to the information. Hosts also refresh the information that is received in previous advertisements.

**Address Uniqueness**

For safety, all addresses must be tested for uniqueness prior to their assignment to an interface. The situation is different for addresses that are created through stateless autoconfiguration. The uniqueness of an address is determined primarily by the portion of the address that is formed from an interface identifier. Thus, if a node has already verified the uniqueness of a link-local address, additional addresses need not be tested individually. The addresses must be created from the same interface identifier. In contrast, all addresses that are obtained manually should be tested individually for uniqueness. The same is true for addresses that are obtained by stateful address autoconfiguration. Some sites believe that the overhead of performing duplicate address detection outweighs its benefits. For these sites, the use of duplicate address detection can be disabled by setting a per-interface configuration flag.

To accelerate the autoconfiguration process, a host can generate its link-local address, and verify its uniqueness, while the host waits for a router advertisement. A router might delay a response to a router solicitation for a few seconds. Consequently, the total time necessary to complete autoconfiguration can be significantly longer if the two steps are done serially.

**A website with more information:**

**<http://tools.ietf.org/search/rfc2462>**