WSA - HW - 3:

5. Describe how the APIPA (IPv4) Auto-Configuration works. Provides additional information what is the mechanism used to prevent duplicate IP Address.

From Wikipedia, the free encyclopedia

Въведение:
Link-local address

In a computer network, a link-local address is a network address that is intended and valid only for communications within a network segment (a single network link, or often: one broadcast domain) that a host is connected to.
Usually, link-local addresses are not guaranteed to be unique beyond a single network segment. In comparison, other classes of addresses like organisation-wide or world-wide (or global) addresses, are unique within an organisation, or even world-wide. Routers therefore do not forward packets with link-local addresses.
The term link-local address would normally be used in cases where a protocol also supports non-link-local (e.g. global) addresses (like IPv4). For protocols that support only link-local addresses (like Ethernet, USB), the plain term 'address' suffices.
Link-local addresses for IPv4 are defined in the address block 169.254.0.0/16, in CIDR notation.

На кратко какво е APIPA:

Automatic Private IP Addressing (APIPA)

When a globally routable or a private address becomes available after a link-local address has been assigned, the use of the new address should generally be preferred to the link-local address for new connections but communication via the link-local address is still possible.
Microsoft refers to this address autoconfiguration method as Automatic Private IP Addressing (APIPA). It is sometimes also casually referred to as auto-IP.

Повече информация относно Automatic Private IP Addressing (APIPA):

The IP address of a TCP/IP host is, in many ways, its identity. Every TCP/IP network requires that all hosts have unique addresses to facilitate communication. When a network is manually configured with a distinct IP address for each host, the hosts permanently know “who they are”. When hosts are made DHCP clients, they no longer have a permanent identity; they rely on a DHCP server to tell them “who they are”.
Client Recovery From Failure to Obtain an IP Address
The dependency of DHCP clients on servers is not a problem as long as DHCP is functioning normally and a host can get a lease, and in fact has many benefits that we have explored. Unfortunately, a number of circumstances can arise that result in one of the DHCP processes not resulting in a lease for the client. The client may not be able to obtain a lease, re-acquire one after reboot, or renew an existing lease. There are many possible reasons why this might happen:

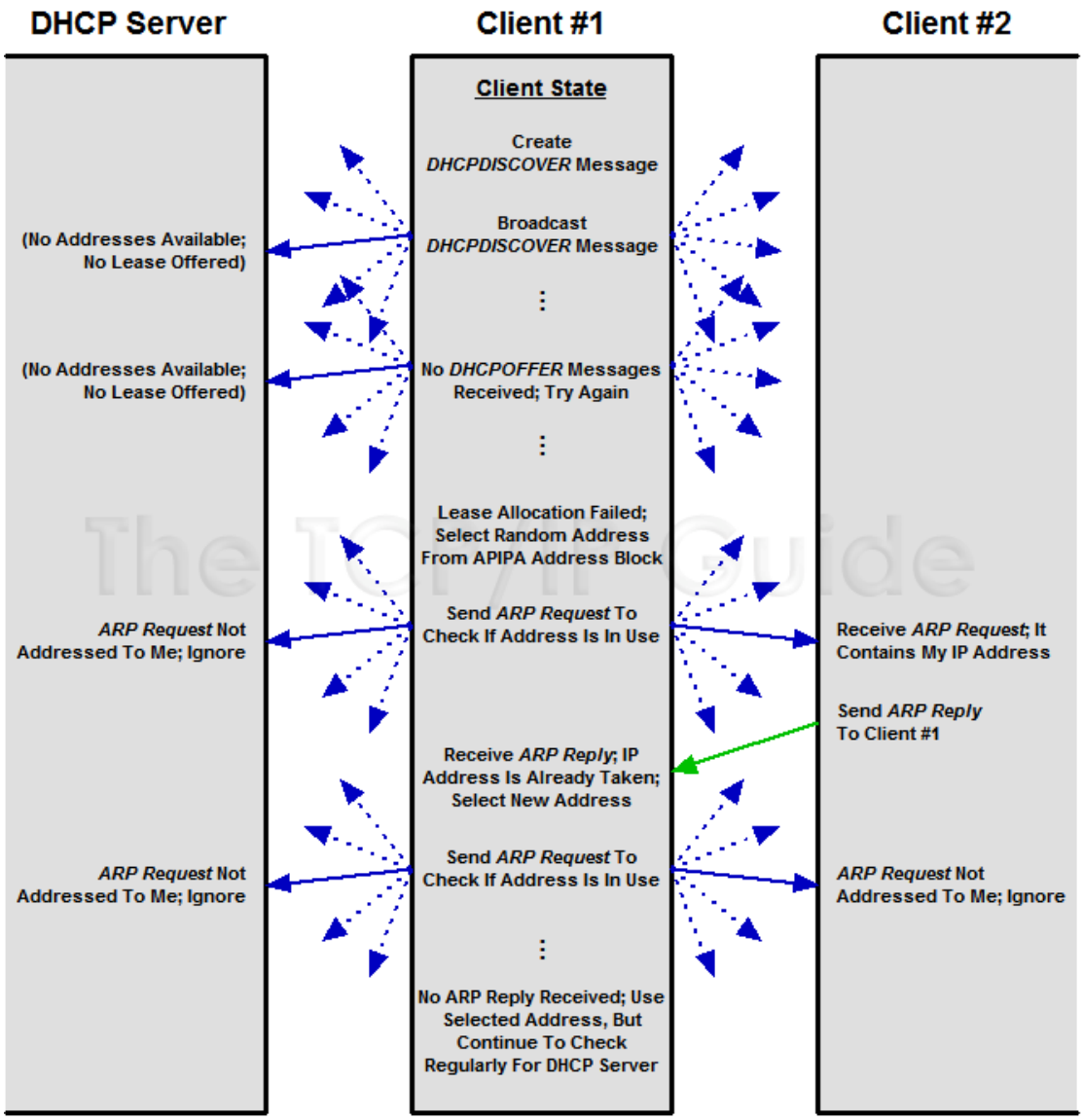
- The DHCP server may have experienced a failure, or may be taken down for maintenance;
- The relay agent on the client's local network may have failed;

- Another hardware malfunction or power failure may make communication impossible;
- The network may have run out of allocatable addresses.

Without a lease, the host has no IP address, and without an address, the host is effectively dead in the water. The base DHCP specification doesn't really specify any recourse for the host in the event that it cannot successfully obtain a lease. It is essentially left up to the implementor to decide what to do, and when DHCP was first created, many host implementations would simply display an error message and leave the host unusable until an administrator or user took action.
Clearly this is far from an ideal situation. It would be better if we could just have a DHCP client that is unable to reach a server automatically configure itself. In fact, the IETF reserved a special IP address block for this purpose. This block, 169.254.0.1 through 169.254.255.254 (or 169.254.0.0/16 in classless notation) is reserved for autoconfiguration, as mentioned in RFC 3330:
"Hosts obtain these addresses by auto-configuration, such as when a DHCP server may not be found."
Strangely, however, no TCP/IP standard was defined to specify how such autoconfiguration works. To fill the void, Microsoft created an implementation that it calls Automatic Private IP Addressing (APIPA). Due to Microsoft's market power, APIPA has been deployed on millions of machines, and has thus become a de facto standard in the industry. Many years later, the IETF did define a formal standard for this functionality, in RFC 3927, Dynamic Configuration of IPv4 Link-Local Addresses.

APIPA Operation

APIPA is really so simple that it's surprising it took so long for someone to come up with the idea. It takes over at the point where any DHCP lease process fails. Instead of just halting with an error message, APIPA randomly chooses an address within the aforementioned private addressing block. It then performs a test very similar to the one in step #13 in the DHCP allocation process: it uses ARP to generate a request on the local network to see if any other client responds using the address it has chosen. If there is a reply, APIPA tries another random address and repeats the test. When the APIPA software finds an address that is not in use, it is given to the client as a default address. The client will then use default values for other configuration parameters that it would normally receive from the DHCP server.



In this example, Client #1 is trying to get an IP address from its DHCP server, but the server is out of addresses, so it does not respond to the client's requests. The client is configured to use APIPA, so it randomly selects an address from the APIPA address block. It sends an ARP Request on the local network to see if any other device is using that address; in this case, Usually there will be no conflict, but here Client #2 is in fact using the address, so it responds. Client #1 chooses a different address and this time gets no reply. It begins using that address, while continuing to check regularly for a DHCP server to come online.

A client using an autoconfigured address will continue to try to contact a DHCP server periodically. By default, this check is performed every five minutes. If and when it finds one, it will obtain a lease and replace the autoconfigured address with the proper leased address. APIPA is ideally suited to small networks, where all devices are on a single physical link. Conceivably, with 20 APIPA-enabled DHCP clients on a network with a single DHCP server, you could take the server down for maintenance and still have all the clients work properly, using 169.254.x.x addresses.

APIPA Limitations

Bear in mind that APIPA is not a proper replacement for full DHCP. The 169.254.0.0/16 block is a private IP range and comes with all the limitations of private IP

addresses, including inability to use these addresses on the Internet. Also, APIPA cannot provide the other configuration parameters that a client may need to get from a DHCP server. Finally, APIPA will not work properly in conjunction with proxy ARP, because the proxy will respond for any of the private addresses, so they will all appear to be used.
Since it uses ARP to check for address conflicts, APIPA is not well-suited for large internetworks. To use it on an internetwork with multiple subnets, you would require software that allows each subnet to use a different portion of the full 169.254.0.0/16 blocks, to avoid conflicts. In practice, APIPA is a solution for small networks; large internetworks deal with the problem of not being able to contact a DHCP server by making sure that a client can always contact a DHCP server.

Извод:

An optional DHCP feature called Automatic Private IP Addressing (APIPA) was developed to allow clients to still be able to communicate in the event that they are unable to obtain an IP address from a DHCP server. When enabled, the client chooses a random address from a special reserved block of private IP addresses, and checks to make sure the address is not already in use by another device. It continues to check for a DHCP server periodically until it is able to find one.

Източници и допълнителна информация:

http://en.wikipedia.org/wiki/Private_network
http://en.wikipedia.org/wiki/Link-local_address
http://www.ietf.org/rfc/rfc3927.txt
http://www.tcpipguide.com/free/t_DHCPAutoconfigurationAutomaticPrivateIPAddressingA.htm
http://www.rfc-base.org/txt/rfc-3330.txt
http://tools.ietf.org/search/rfc3330
http://support.microsoft.com/kb/220874
http://technet.microsoft.com/en-us/library/dd163570.aspx