NAME

dhclient-script - DHCP client network configuration script

DESCRIPTION

The DHCP client network configuration script is invoked from time to time by **dhclient(8)**. This script is used by the dhcp client to set each interface's initial configuration prior to requesting an address, to test the address once it has been offered, and to set the interface's final configuration once a lease has been acquired. If no lease is acquired, the script is used to test predefined leases, if any, and also called once if no valid lease can be identified.

This script is not meant to be customized by the end user. If local customizations are needed, they should be possible using the enter and exit hooks provided (see HOOKS for details). These hooks will allow the user to override the default behaviour of the client in creating a **/etc/resolv.conf** file.

No standard client script exists for some operating systems, even though the actual client may work, so a pioneering user may well need to create a new script or modify an existing one. In general, customizations specific to a particular computer should be done in the **/etc/dhcp/dhclient.conf** file. If you find that you can't make such a customization without customizing **/etc/dhcp/dhclient.conf** or using the enter and exit hooks, please submit a bug report.

HOOKS

When it starts, the client script first defines a shell function, **make_resolv_conf ,** which is later used to create the **/etc/resolv.conf** file. To override the default behaviour, redefine this function in the enter hook script.

After defining the make_resolv_conf function, the client script checks for the presence of an executable **/etc/dhcp/dhclient-enter-hooks** script, and if present, it invokes the script inline, using the Bourne shell ´.´ command. It also invokes all executable scripts in **/etc/dhcp/dhclient-enter-hooks.d/*** in the same way. The entire environment documented under OPERATION is available to this script, which may modify the environment if needed to change the behaviour of the script. If an error occurs during the execution of the script, it can set the exit_status variable to a nonzero value, and **/sbin/dhclient-script** will exit with that error code immediately after the client script exits.

After all processing has completed, **/sbin/dhclient-script** checks for the presence of an executable **/etc/dhcp/dhclient-exit-hooks** script, which if present is invoked using the '.' command. All executable scripts in **/etc/dhcp/dhclient-exit-hooks.d/*** are also invoked. The exit status of dhclient-script will be passed to dhclient-exit-hooks in the exit_status shell variable, and will always be zero if the script succeeded at the task for which it was invoked. The rest of the environment as described previously for dhclient-enter-hooks is also present. The **/etc/dhcp/dhclient-exit-hooks** and **/etc/dhcp/dhclient-exit-hooks.d/*** scripts can modify the value of exit_status to change the exit status of dhclient-script.

OPERATION

When dhclient needs to invoke the client configuration script, it defines a set of variables in the environment, and then invokes **/sbin/dhclient-script.** In all cases, $reason is set to the name of the reason why the script has been invoked. The following reasons are currently defined: MEDIUM, PREINIT, BOUND, RENEW, REBIND, REBOOT, EXPIRE, FAIL, STOP, RELEASE, NBI and TIMEOUT.

MEDIUM

The DHCP client is requesting that an interface's media type be set. The interface name is passed in $interface, and the media type is passed in $medium.

PREINIT

The DHCP client is requesting that an interface be configured as required in order to send packets prior to receiving an actual address. For clients which use the BSD socket library, this means configuring the interface with an IP address of 0.0.0.0 and a broadcast address of 255.255.255.255. For other clients, it may be possible to simply configure the interface up without actually giving it an IP address at all. The interface name is passed in $interface, and the media type in $medium.

If an IP alias has been declared in dhclient.conf, its address will be passed in $alias_ip_address, and that ip alias should be deleted from the interface, along with any routes to it.

BOUND

The DHCP client has done an initial binding to a new address. The new ip address is passed in $new_ip_address, and the interface name is passed in $interface. The media type is passed in $medium. Any options acquired from the server are passed using the option name described in **dhcp-options**, except that dashes (´-´) are replaced by underscores (´_´) in order to make valid shell variables, and the variable names start with new_. So for example, the new subnet mask would be passed in $new_subnet_mask. Options from a non-default universe will have the universe name prepended to the option name, for example $new_dhcp6_server_id. The options that the client explicitly requested via a PRL or ORO option are passed with the same option name as above but prepended with requested_ and with a value of 1, for example requested_subnet_mask=1. No such variable is defined for options not requested by the client or options that don't require a request option, such as the ip address (*_ip_address) or expiration time (*_expiry).

Before actually configuring the address, dhclient-script should somehow ARP for it and exit with a nonzero status if it receives a reply. In this case, the client will send a DHCPDECLINE message to the server and acquire a different address. This may also be done in the RENEW, REBIND, or REBOOT states, but is not required, and indeed may not be desirable.

When a binding has been completed, a lot of network parameters are likely to need to be set up. A new /etc/resolv.conf needs to be created, using the values of $new_domain_name and $new_domain_name_servers (which may list more than one server, separated by spaces). A default route should be set using $new_routers, and static routes may need to be set up using $new_static_routes.

If an IP alias has been declared, it must be set up here. The alias IP address will be written as $alias_ip_address, and other DHCP options that are set for the alias (e.g., subnet mask) will be passed in variables named as described previously except starting with $alias_ instead of $new_. Care should be taken that the alias IP address not be used if it is identical to the bound IP address ($new_ip_address), since the other alias parameters may be incorrect in this case.

RENEW

When a binding has been renewed, the script is called as in BOUND, except that in addition to all the variables starting with $new_, and $requested_ there is another set of variables starting with $old_. Persistent settings that may have changed need to be deleted - for example, if a local route to the bound address is being configured, the old local route should be deleted. If the default route has changed, the old default route should be deleted. If the static routes have changed, the old ones should be deleted. Otherwise, processing can be done as with BOUND.

REBIND

The DHCP client has rebound to a new DHCP server. This can be handled as with RENEW, except that if the IP address has changed, the ARP table should be cleared.

REBOOT

The DHCP client has successfully reacquired its old address after a reboot. This can be processed as with BOUND.

EXPIRE

The DHCP client has failed to renew its lease or acquire a new one, and the lease has expired. The IP address must be relinquished, and all related parameters should be deleted, as in RENEW and REBIND.

FAIL

The DHCP client has been unable to contact any DHCP servers, and any leases that have been tested have not proved to be valid. The parameters from the last lease tested should be deconfigured. This can be handled in the same way as EXPIRE.

STOP

The dhclient has been informed to shut down gracefully, the dhclient-script should unconfigure or shutdown the interface as appropriate.

RELEASE

The dhclient has been executed using the -r flag, indicating that the administrator wishes it to release its lease(s). dhclient-script should unconfigure or shutdown the interface.

NBI

No-Broadcast-Interfaces...dhclient was unable to find any interfaces upon which it believed it should commence DHCP. What dhclient-script should do in this situation is entirely up to the implementor.

TIMEOUT

The DHCP client has been unable to contact any DHCP servers. However, an old lease has been identified, and its parameters have been passed in as with BOUND. The client configuration script should test these parameters and, if it has reason to believe they are valid, should exit with a value of zero. If not, it should exit with a nonzero value.

The usual way to test a lease is to set up the network as with REBIND (since this may be called to test more than one lease) and then ping the first router defined in $routers. If a response is received, the lease must be valid for the network to which the interface is currently connected. It would be more complete to try to ping all of the routers listed in $new_routers, as well as those listed in $new_static_routes, but current scripts do not do this.

FILES

Each operating system should generally have its own script file, although the script files for similar operating systems may be similar or even identical. The script files included in Internet Systems Consortium DHCP distribution appear in the distribution tree under client/scripts, and bear the names of the operating systems on which they are intended to work.

BUGS

If more than one interface is being used, there's no obvious way to avoid clashes between server-supplied configuration parameters - for example, the stock dhclient-script rewrites /etc/resolv.conf. If more than one interface is being configured, /etc/resolv.conf will be repeatedly initialized to the values provided by one server, and then the other. Assuming the information provided by both servers is valid, this shouldn't cause any real problems, but it could be confusing.

SEE ALSO

dhclient(8), dhcpd(8), dhcrelay(8), dhclient.conf(5) and dhclient.leases(5).

AUTHOR

dhclient-script(8) To learn more about Internet Systems Consortium, see **https://www.isc.org.**