

Section 19 - The iplookup router

19. The iplookup router

The iplookup router was written to fulfil a specific requirement in Cambridge University (which in fact no longer exists). For this reason, it is not included in the binary of Exim by default. If you want to include it, you must set `ROUTER_IPLOOKUP=yes`

in your Local/Makefile configuration file.

The iplookup router routes an address by sending it over a TCP or UDP connection to one or more specific hosts. The host can then return the same or a different address – in effect rewriting the recipient address in the message's envelope. The new address is then passed on to subsequent routers. If this process fails, the address can be passed on to other routers, or delivery can be deferred. Since iplookup is just a rewriting router, a transport must not be specified for it.

`hostsUse: iplookupType: stringDefault: unset`

This option must be supplied. Its value is a colon-separated list of host names. The hosts are looked up using `gethostbyname()` (or `getipnodebyname()` when available) and are tried in order until one responds to the query. If none respond, what happens is controlled by `optional`.

`optionalUse: iplookupType: booleanDefault: false`

If `optional` is true, if no response is obtained from any host, the address is passed to the next router, overriding `no_more`. If `optional` is false, delivery to the address is deferred.

`portUse: iplookupType: integerDefault: 0`

This option must be supplied. It specifies the port number for the TCP or UDP call.

`protocolUse: iplookupType: stringDefault: udp`

This option can be set to `“udp”` or `“tcp”` to specify which of the two protocols is to be used.

`queryUse: iplookupType: string†Default: $local_part@$domain $local_part@$domain`

This defines the content of the query that is sent to the remote hosts. The repetition serves as a way of checking that a response is to the correct query in the default case (see `response_pattern` below).

`rerouteUse: iplookupType: string†Default: unset`

If this option is not set, the rerouted address is precisely the byte string returned by the remote host, up to the first white space, if any. If set, the string is expanded to form the rerouted address. It can include parts matched in the response by `response_pattern` by means of numeric variables such as `$1`, `$2`, etc. The variable `$0` refers to the entire input string, whether or not a pattern is in use. In all cases, the rerouted address must end up in the form `local_part@domain`.

`response_patternUse: iplookupType: stringDefault: unset`

This option can be set to a regular expression that is applied to the string returned from the remote host. If the pattern does not match the response, the router declines. If `response_pattern` is not set, no checking of the response is done, unless the query was defaulted, in which case there is a check that the text returned after the first white space is the original address. This checks that the answer that has been received is in response to the correct question. For example, if the response is just a new domain, the following could be used: `response_pattern = ^([^@]+)$`
`reroute = $local_part@$1`

`timeoutUse: iplookupType: timeDefault: 5s`

This specifies the amount of time to wait for a response from the remote machine. The same timeout is used for the `connect()` function for a TCP call. It does not apply to UDP.