```
aux_verify_sender:

  # Only do sender callouts if the sender is not known to be incompetent
  # according to any of the preliminary ACL conditions. We assume that
  # the caller has already required verify = sender.

  accept
    condition      = \
      ${lookup {${lc:$sender_address_domain}} partial-cdb {DB/nocallout.cdb} \
               {yes} {${lookup {${lc:$sender_address}} cdb {DB/nocallout.cdb} \
                               {yes} {no} }} }

  accept
    dnslists       = dsn.rfc-ignorant.org/$sender_address_domain

  require
    verify         = sender/callout=CALLTIME,defer_ok

  accept
```

```
aux_check_spam:

  # Check for ratware HELO signatures. We don't use the full strictness of
  # verify=helo; if it fails we only check for a few choice stupidities.

  deny
    message        = Please use your name when saying HELO (not $sender_helo_name)
  ! verify         = helo
    condition      = ${if or{{ eq{$ACL_HELO}{bad} } \
                             { isip{$sender_helo_name} } \
                             { eq{$sender_helo_name}{$local_part} } \
                             { match{$sender_helo_name}{\N[.][.]|.{55}\N} } \
                             { match_domain{$sender_helo_name}{+our_domains} }} }
    set ACL_HELO   = bad

  # Look up in a few choice blacklists.

  deny
    message        = ${sender_host_address} is listed at ${dnslist_domain}; \
                     See ${dnslist_text}
    dnslists       = sbl-xbl.spamhaus.org

  deny
    message        = ${sender_host_address} is listed at ${dnslist_domain}; \
                     See http://mail-abuse.com/cgi-bin/lookup?${sender_host_address}
    dnslists       = rbl-plus.mail-abuse.ja.net

  deny
    message        = ${sender_address_domain} is listed at ${dnslist_domain}; \
                     ${dnslist_text}
    dnslists       = nomail.rhsbl.sorbs.net/$sender_address_domain

  # It has passed the tests.

  accept
```