```
/* GHOST-Test by Qualys (http://seclists.org/oss-sec/2015/q1/274) */
#include <netdb.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>

#define CANARY "in_the_coal_mine"

struct {
  char buffer[1024];
  char canary[sizeof(CANARY)];
} temp = { "buffer", CANARY };

int main(void) {
  struct hostent resbuf;
  struct hostent *result;
  int herrno;
  int retval;

  /*** strlen (name) = size_needed - sizeof (*host_addr) - sizeof (*h_addr_ptrs) - 1;
***/
  size_t len = sizeof(temp.buffer) - 16*sizeof(unsigned char) - 2*sizeof(char *) - 1;
  char name[sizeof(temp.buffer)];
  memset(name, '0', len);
  name[len] = '\0';

  retval = gethostbyname_r(name, &resbuf, temp.buffer, sizeof(temp.buffer), &result,
&herrno);

  if (strcmp(temp.canary, CANARY) != 0) {
    puts("vulnerable");
    exit(EXIT_SUCCESS);
  }
  if (retval == ERANGE) {
    puts("not vulnerable");
    exit(EXIT_SUCCESS);
  }
  puts("should not happen");
  exit(EXIT_FAILURE);
}
```