Московский Авиационный Институт

(Национальный исследовательский Университет)

Факультет: «Информационные технологии и прикладная математика»
Кафедра: 806 «Вычислительная математика и программирование»

Лабораторная работа № 6
по курсу «Операционные системы»

Студент:	Обыденкова Ю. Ю.
Группа:	М8О-308Б-18
Вариант:	41
Преподаватель:	Миронов Е.С.
Оценка:	
Дата:	

Москва, 2020

Постановка задачи

Реализовать распределенную систему по обработке запросов. В данной системе должно существовать 2 вида узлов: «управляющий» и «вычислительный». Необходимо объединить данные узлы в соответствии с той топологией, которая определена вариантом. Связь между узлами необходимо осуществить при помощи сервера сообщений zmq. Также в данной системе необходимо предусмотреть проверку доступности узлов в соответствии с вариантом.

Вариант задания: 41. Топология — бинарное дерево. Тип вычислительной команды — сумма n чисел. Тип проверки узлов на доступность — пинг всех узлов.

Общие сведения о программе

Программа состоит из двух файлов, которые компилируются в исполнительные файлы (которые представляют управляющий и вычислительные узлы), а так же из статической библиотеки, которая подключается к вышеуказанным файлам. Общение между процессами происходит с помощью библиотеки zmq.

Общий метод и алгоритм решения

- Управляющий узел принимает команды, обрабатывает их и пересылает дочерним узлам (или выводит сообщение об ошибке).
- Дочерние узлы проверяют, может ли быть команда выполнена в данном узле, если нет, то команда пересылается в один из дочерних узлов, из которого возвращается некоторое сообщение (об успехе или об ошибке), которое потом пересылается обратно по дереву.
- Для корректной проверки на доступность узлов, используется дерево, эмулирующее поведение узлов в данной топологии (например, при удалении узла, удаляются все его потомки).
- Если узел недоступен, то по истечении таймаута будет сгенерировано сообщение о недоступности узла и оно будет передано вверх по дереву, к управляющему узлу.
- При удалении узла, все его потомки рекурсивно уничтожаются.

Код программы

main_node.cpp:

```
#include <iostream>
#include "zmq.hpp"
#include <string>
#include <zconf.h>
#include <vector>
#include <signal.h>
#include <sstream>
#include <set>
#include <algorithm>
#include "server_functions.h"

class IdTree {
public:
  IdTree() = default;
  ~IdTree() {
    delete_node(head_);
  }
  bool contains(int id) {
    TreeNode* temp = head_;
    while(temp != nullptr) {
      if (temp->id_ == id) {
        break;
      }
      if (id > temp->id_) {
        temp = temp->right;
      }
      if (id < temp->id_) {
        temp = temp->left;
      }
    }
    return temp != nullptr;
  }
  void insert(int id) {
    if (head_ == nullptr) {
      head_ = new TreeNode(id);
      return;
    }
    TreeNode* temp = head_;
    while(temp != nullptr) {
      if (id == temp->id_) {
        break;
      }
      if (id < temp->id_) {
        if (temp->left == nullptr) {
          temp->left = new TreeNode(id);
          break;
        }
        temp = temp->left;
      }
      if (id > temp->id_) {
        if (temp->right == nullptr) {
          temp->right = new TreeNode(id);
          break;
        }
        temp = temp->right;
      }
    }
  }
  void erase(int id) {
    TreeNode* prev_id = nullptr;
    TreeNode* temp = head_;
    while (temp != nullptr) {
      if (id == temp->id_) {
        if (prev_id == nullptr) {
          head_ = nullptr;
        } else {
          if (prev_id->left == temp) {
            prev_id->left = nullptr;
          } else {
            prev_id->right = nullptr;
          }
        }
        delete_node(temp);
      } else if (id < temp->id_) {
        prev_id = temp;
        temp = temp->left;
      } else if (id > temp->id_) {
        prev_id = temp;
        temp = temp->right;
      }
    }
  }
  std::vector<int> get_nodes() const {
    std::vector<int> result;
    get_nodes(head_, result);
    return result;
  }
private:
  struct TreeNode {
    TreeNode(int id) : id_(id) {}
    int id_;
    TreeNode* left = nullptr;
    TreeNode* right = nullptr;
  };
  void get_nodes(TreeNode* node, std::vector<int>& v) const {
    if (node == nullptr) {
      return;
    }
    get_nodes(node->left,v);
    v.push_back(node->id_);
    get_nodes(node->right, v);
  }
  void delete_node(TreeNode* node) {
    if (node == nullptr) {
      return;
    }
    delete_node(node->right);
    delete_node(node->left);
    delete node;
  }
  TreeNode* head_ = nullptr;
};
int main() {
  std::string command;
  IdTree ids;
  size_t child_pid = 0;
  int child_id = 0;
  zmq::context_t context(1);
  zmq::socket_t main_socket(context, ZMQ_REQ);
  int linger = 0;
  main_socket.setsockopt(ZMQ_SNDTIMEO, 2000);
  //main_socket.setsockopt(ZMQ_RCVTIMEO, 2000);
  main_socket.setsockopt(ZMQ_LINGER, &linger, sizeof(linger));
  //main_socket.connect(get_connect_name(30000));
  int port = bind_socket(main_socket);
  while (true) {
    std::cin >> command;
    if (command == "create") {
      size_t node_id;
      std::string result;
      std::cin >> node_id;
      if (child_pid == 0) {
        child_pid = fork();
        if (child_pid == -1) {
          std::cout << "Unable to create first worker node\n";
          child_pid = 0;
          exit(1);
        } else if (child_pid == 0) {
          create_node(node_id, port);
        } else {
          child_id = node_id;
          send_message(main_socket,"pid");
          result = recieve_message(main_socket);
        }
      } else {
//            if (child_id == node_id) {
//               std::cout << "Error: Already exists";
//            }
        std::ostringstream msg_stream;
        msg_stream << "create " << node_id;
        send_message(main_socket, msg_stream.str());
        result = recieve_message(main_socket);
      }
      if (result.substr(0,2) == "Ok") {
        ids.insert(node_id);
      }
      std::cout << result << "\n";
    } else if (command == "remove") {
      if (child_pid == 0) {
        std::cout << "Error:Not found\n";
        continue;
      }
      size_t node_id;
      std::cin >> node_id;
      if (node_id == child_id) {
        kill(child_pid, SIGTERM);
        kill(child_pid, SIGKILL);
        child_id = 0;
        child_pid = 0;
        std::cout << "Ok\n";
        ids.erase(node_id);
        continue;
      }
      std::string message_string = "remove " + std::to_string(node_id);
      send_message(main_socket, message_string);
      std::string recieved_message = recieve_message(main_socket);
      if (recieved_message.substr(0, std::min<int>(recieved_message.size(), 2)) ==
"Ok") {
        ids.erase(node_id);
      }
      std::cout << recieved_message << "\n";
    } else if (command == "exec") {
      int id, n;
      std::cin >> id >> n;
      std::vector<int> numbers(n);
      for (int i = 0; i < n; ++i) {
        std::cin >> numbers[i];
      }
      std::string message_string = "exec " + std::to_string(id) + " " + std::to_string(n);
      for (int i = 0; i < n; ++i) {
        message_string += " " + std::to_string(numbers[i]);
      }
      send_message(main_socket, message_string);
      std::string recieved_message = recieve_message(main_socket);
      std::cout << recieved_message << "\n";
    } else if (command == "pingall") {
      send_message(main_socket,"pingall");
      std::string recieved = recieve_message(main_socket);
      std::istringstream is;
      if (recieved.substr(0,std::min<int>(recieved.size(), 5)) == "Error") {
        is = std::istringstream("");
      } else {
        is = std::istringstream(recieved);
      }

      std::set<int> recieved_ids;
      int rec_id;
      while (is >> rec_id) {
        recieved_ids.insert(rec_id);
      }
      std::vector from_tree = ids.get_nodes();
      auto part_it = std::partition(from_tree.begin(), from_tree.end(), [&recieved_ids] (int
a) {
        return recieved_ids.count(a) == 0;
      });
      if (part_it == from_tree.begin()) {
        std::cout << "Ok: -1\n";
      } else {
        std::cout << "Ok:";
        for (auto it = from_tree.begin(); it != part_it; ++it) {
          std::cout << " " << *it;
        }
        std::cout << "\n";
      }
    } else if (command == "exit") {
      break;
    }
  }
  return 0;
}
```

child_node.cpp:

```
#include <iostream>
#include "zmq.hpp"
#include <string>
#include <sstream>
#include <zconf.h>
#include <exception>
#include <signal.h>
#include "server_functions.h"
int main(int argc, char** argv) { //аргументы - айди и номер порта, к которому нужно подключиться
//    zmq::context_t context (1);
//    zmq::message_t msg(strlen(argv[1]));
//    zmq::message_t msg_2(strlen(argv[2]));
//    zmq::message_t rcv;
//    memcpy(msg.data(), argv[1],strlen(argv[1]));
//    memcpy(msg_2.data(), argv[2],strlen(argv[2]));
//    socket.send(msg);
//    socket.recv(&rcv);
//    socket.send(msg_2);
  int id = std::stoi(argv[1]);
  int parent_port = std::stoi(argv[2]);
  zmq::context_t context(3);
  zmq::socket_t parent_socket(context, ZMQ_REP);
//    zmq::socket_t socket (context, ZMQ_REQ);
//    socket.connect ("tcp://127.0.0.1:5555");
  parent_socket.connect(get_port_name(parent_port));
  int left_pid = 0;
  int right_pid = 0;
  int left_id = 0;
  int right_id = 0;

  zmq::socket_t left_socket(context, ZMQ_REQ);
  zmq::socket_t right_socket(context, ZMQ_REQ);
  int linger = 0;
  left_socket.setsockopt(ZMQ_SNDTIMEO, 2000);
  //left_socket.setsockopt(ZMQ_RCVTIMEO, 2000);
  left_socket.setsockopt(ZMQ_LINGER, &linger, sizeof(linger));
  right_socket.setsockopt(ZMQ_SNDTIMEO, 2000);
  //right_socket.setsockopt(ZMQ_RCVTIMEO, 2000);
  right_socket.setsockopt(ZMQ_LINGER, &linger, sizeof(linger));
  int left_port = bind_socket(left_socket);
  int right_port = bind_socket(right_socket);
  while (true) {
    std::string request_string;
    request_string = recieve_message(parent_socket);
//        std::ostringstream stream;
//        stream << "Worker: id:" << id << "\n"
//               << "pid:" << getpid() << "\n"
//       << "parent port:" << parent_port << "\n"
//       << "left port:" << left_port << "\n"
//       << "right port:" << right_port << "\n"
//       << "left child: id:" << left_id << " pid:" << left_pid << "\n"
//       << "right child: id:" << right_id << " pid:" << right_pid << "\n"
//       << "request:" << request_string << "\n\n";
//    send_message(socket, stream.str());
//    recieve_message(socket);
  std::istringstream command_stream(request_string);
  std::string command;
  command_stream >> command;
  if (command == "id") {
   std::string parent_string = "Ok:" + std::to_string(id);
   send_message(parent_socket, parent_string);
  } else if (command == "pid") {
   std::string parent_string = "Ok:" + std::to_string(getpid());
   send_message(parent_socket, parent_string);
  } else  if (command == "create") {
   int id_to_create;
   command_stream >> id_to_create;
   // управляюший узел сообщает id нового узла и порт, к которому его надо
подключить
   if (id_to_create == id) {
    // если id равен данному, значит узел уже существует, посылаем ответ с
ошибкой
    std::string message_string = "Error: Already exists";
    send_message(parent_socket, message_string);
   } else if (id_to_create < id) {
    if (left_pid == 0) {
     left_pid = fork();
     if (left_pid == -1) {
      send_message(parent_socket, "Error: Cannot fork");
      left_pid = 0;
     } else if (left_pid == 0) {
      create_node(id_to_create,left_port);
     } else {
      left_id = id_to_create;
      send_message(left_socket, "pid");
      send_message(parent_socket, recieve_message(left_socket));
     }
    } else {
     send_message(left_socket, request_string);
     send_message(parent_socket, recieve_message(left_socket));
    }
   } else {
    if (right_pid == 0) {
     right_pid = fork();
     if (right_pid == -1) {
      send_message(parent_socket, "Error: Cannot fork");
      right_pid = 0;
     } else if (right_pid == 0) {
      create_node(id_to_create,right_port);
     } else {
      right_id = id_to_create;
      send_message(right_socket, "pid");
      send_message(parent_socket, recieve_message(right_socket));
     }
```

```
        } else {
          send_message(right_socket, request_string);
          send_message(parent_socket, recieve_message(right_socket));
        }
      }
    } else if (command == "remove") {
      int id_to_delete;
      command_stream >> id_to_delete;
      if (id_to_delete < id) {
        if (left_id == 0) {
          send_message(parent_socket, "Error: Not found");
        } else if (left_id == id_to_delete) {
          send_message(left_socket, "kill_children");
          recieve_message(left_socket);
          kill(left_pid,SIGTERM);
          kill(left_pid,SIGKILL);
          left_id = 0;
          left_pid = 0;
          send_message(parent_socket, "Ok");
        } else {
          send_message(left_socket, request_string);
          send_message(parent_socket, recieve_message(left_socket));
        }
      } else {
        if (right_id == 0) {
          send_message(parent_socket, "Error: Not found");
        } else if (right_id == id_to_delete) {
          send_message(right_socket, "kill_children");
          recieve_message(right_socket);
          kill(right_pid,SIGTERM);
          kill(right_pid,SIGKILL);
          right_id = 0;
          right_pid = 0;
          send_message(parent_socket, "Ok");
        } else {
          send_message(right_socket, request_string);
          send_message(parent_socket, recieve_message(right_socket));
        }
      }
    } else if (command == "exec") {
      int exec_id;
      command_stream >> exec_id;
      if (exec_id == id) {
        int n;
        command_stream >> n;
        int sum = 0;
        for (int i = 0; i < n; ++i) {
          int cur_num;
          command_stream >> cur_num;
          sum += cur_num;
        }
        std::string recieve_message = "Ok:" + std::to_string(id) + ":" +
std::to_string(sum);
        send_message(parent_socket, recieve_message);

      } else if (exec_id < id) {
        if (left_pid == 0) {
          std::string recieve_message = "Error:" + std::to_string(exec_id) + ": Not
found";
          send_message(parent_socket, recieve_message);
        } else {
          send_message(left_socket, request_string);
          send_message(parent_socket, recieve_message(left_socket));
        }
      } else {
        if (right_pid == 0) {
          std::string recieve_message = "Error:" + std::to_string(exec_id) + ": Not
found";
          send_message(parent_socket, recieve_message);
        } else {
          send_message(right_socket, request_string);
          send_message(parent_socket, recieve_message(right_socket));
        }
      }
    } else if (command == "pingall") {
      std::ostringstream res;
      std::string left_res;
      std::string right_res;
      if (left_pid != 0) {
        send_message(left_socket, "pingall");
        left_res = recieve_message(left_socket);
      }
      if (right_pid != 0) {
        send_message(right_socket, "pingall");
        right_res = recieve_message(right_socket);
      }
      if (!left_res.empty() && left_res.substr(std::min<int>(left_res.size(),5)) != "Error")
{
        res << left_res;
      }
      if (!right_res.empty() && right_res.substr(std::min<int>(right_res.size(),5)) !=
"Error") {
        res << right_res;
      }
      send_message(parent_socket, res.str());
    } else if (command == "kill_children") {  // УБИТЬ ВСЕХ ДЕТЕЙ
      if (left_pid == 0 && right_pid == 0) {
        send_message(parent_socket, "Ok");
      } else {
        if (left_pid != 0) {
          send_message(left_socket, "kill_children");
          recieve_message(left_socket);
          kill(left_pid,SIGTERM);
          kill(left_pid,SIGKILL);
        }
        if (right_pid != 0) {
          send_message(right_socket, "kill_children");
          recieve_message(right_socket);
          kill(right_pid,SIGTERM);
          kill(right_pid,SIGKILL);
        }
        send_message(parent_socket, "Ok");
      }
    }
    if (parent_port == 0) {
      break;
    }
  }
}
```

server_functions.cpp:

```
#include "server_functions.h"
bool send_message(zmq::socket_t& socket, const std::string& message_string) {
  zmq::message_t message(message_string.size());
  memcpy(message.data(), message_string.c_str(), message_string.size());
  return socket.send(message);
}
std::string recieve_message(zmq::socket_t& socket) {
  zmq::message_t message;
  bool ok;
  try {
    ok = socket.recv(&message);
  } catch (...) {
    ok = false;
  }
  std::string recieved_message(static_cast<char*>(message.data()), message.size());
  if (recieved_message.empty() || !ok) {
    return "Error: Node is not available";
  }
  return recieved_message;
}
std::string get_port_name(int port) {
  return "tcp://127.0.0.1:" + std::to_string(port);
}
int bind_socket(zmq::socket_t& socket) {
  int port = 30000;
  while (true) {
    try {
      socket.bind(get_port_name(port));
      break;
    } catch(...) {
      port++;
    }
  }
  return port;
}
void create_node(int id, int port) {
  char* arg1 = strdup((std::to_string(id)).c_str());
  char* arg2 = strdup((std::to_string(port)).c_str());
  char* args[] = {"./child_node", arg1, arg2, NULL};
  execv("./child_node", args);
}
```

server_functions.h:

```
#pragma once
#include <string>
#include <zconf.h>
#include "zmq.hpp"
```

```
bool send_message(zmq::socket_t& socket, const std::string& message_string);
std::string recieve_message(zmq::socket_t& socket);
std::string get_port_name(int port);
int bind_socket(zmq::socket_t& socket);
void create_node(int id, int port);
```

Демонстрация работы программы

```
julia@julia-:~/CLionProjects/os_lab_06/src/cmake-build-debug$ ./terminal
create 1
Ok:10200
create 2
Ok:10205
create 3
Ok:10210
create 5
Ok:10215
exec 5 3 1 2 3
Ok:5:6
pingall
Ok: 1 2 3 5
create 5
Error: Already exists
exec 6 3 1 2 3
Error:6: Not found
remove 2
Ok
pingall
Ok: 1
exec 3 3 1 2 3
Error:3: Not found
exit
julia@julia-:~/CLionProjects/os_lab_06/src/cmake-build-debug$ strace ./terminal
execve("./terminal", ["./terminal"], 0x7ffec98bdc20 /* 53 vars */) = 0
brk(NULL)                               = 0x55bfa1b1d000
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/julia/CLionProjects/os_lab_05/cmake-build-debug/tls/x86_64/x86_64/libzmq.so.5", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/home/julia/CLionProjects/os_lab_05/cmake-build-debug/tls/x86_64/x86_64", 0x7ffcffc49e10) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/julia/CLionProjects/os_lab_05/cmake-build-debug/tls/x86_64/libzmq.so.5", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/home/julia/CLionProjects/os_lab_05/cmake-build-debug/tls/x86_64", 0x7ffcffc49e10) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/julia/CLionProjects/os_lab_05/cmake-build-debug/tls/x86_64/libzmq.so.5", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/home/julia/CLionProjects/os_lab_05/cmake-build-debug/tls/x86_64", 0x7ffcffc49e10) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/julia/CLionProjects/os_lab_05/cmake-build-debug/tls/libzmq.so.5", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/home/julia/CLionProjects/os_lab_05/cmake-build-debug/tls", 0x7ffcffc49e10) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/julia/CLionProjects/os_lab_05/cmake-build-debug/x86_64/x86_64/libzmq.so.5", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/home/julia/CLionProjects/os_lab_05/cmake-build-debug/x86_64/x86_64", 0x7ffcffc49e10) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/julia/CLionProjects/os_lab_05/cmake-build-debug/x86_64/libzmq.so.5", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/home/julia/CLionProjects/os_lab_05/cmake-build-debug/x86_64", 0x7ffcffc49e10) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/julia/CLionProjects/os_lab_05/cmake-build-debug/x86_64/libzmq.so.5", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
```

```
stat("/home/julia/CLionProjects/os_lab_05/cmake-build-debug/x86_64", 0x7ffcffc49e10) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/julia/CLionProjects/os_lab_05/cmake-build-debug/libzmq.so.5", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/home/julia/CLionProjects/os_lab_05/cmake-build-debug", 0x7ffcffc49e10) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3

fstat(3, {st_mode=S_IFREG|0644, st_size=152516, ...}) = 0

mmap(NULL, 152516, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f48cb898000

close(3)                                = 0

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/libzmq.so.5", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P?\1\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=630464, ...}) = 0

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f48cb896000

mmap(NULL, 2725560, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f48cb3fd000

mprotect(0x7f48cb490000, 2097152, PROT_NONE) = 0

mmap(0x7f48cb690000, 28672, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x93000) = 0x7f48cb690000

close(3)                                = 0

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\220\304\10\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=1594864, ...}) = 0
```

```
mmap(NULL, 3702848, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f48cb074000

mprotect(0x7f48cb1ed000, 2097152, PROT_NONE) = 0

mmap(0x7f48cb3ed000, 49152, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x179000) =
0x7f48cb3ed000

mmap(0x7f48cb3f9000, 12352, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f48cb3f9000

close(3)                    = 0

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libgcc_s.so.1",
O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\300*\0\0\0\0\0\0"..., 832)
= 832

fstat(3, {st_mode=S_IFREG|0644, st_size=96616, ...}) = 0

mmap(NULL, 2192432, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f48cae5c000

mprotect(0x7f48cae73000, 2093056, PROT_NONE) = 0

mmap(0x7f48cb072000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x16000) =
0x7f48cb072000

close(3)                    = 0

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6",
O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\260\34\2\0\0\0\0\0"...,
832) = 832

fstat(3, {st_mode=S_IFREG|0755, st_size=2030544, ...}) = 0

mmap(NULL, 4131552, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f48caa6b000

mprotect(0x7f48cac52000, 2097152, PROT_NONE) = 0

mmap(0x7f48cae52000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1e7000) =
0x7f48cae52000
```

```
mmap(0x7f48cae58000, 15072, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f48cae58000

close(3)                                = 0

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/libsodium.so.23",
O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\340\251\0\0\0\0\0\0"...,
832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=330440, ...}) = 0

mmap(NULL, 2425864, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f48ca81a000

mprotect(0x7f48ca86a000, 2093056, PROT_NONE) = 0

mmap(0x7f48caa69000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x4f000) =
0x7f48caa69000

close(3)                                = 0

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/libpgm-5.2.so.0",
O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0000;\0\0\0\0\0\0"..., 832) =
832

fstat(3, {st_mode=S_IFREG|0644, st_size=293784, ...}) = 0

mmap(NULL, 2406448, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f48ca5ce000

mprotect(0x7f48ca615000, 2093056, PROT_NONE) = 0

mmap(0x7f48ca814000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x46000) =
0x7f48ca814000

mmap(0x7f48ca816000, 14384, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f48ca816000

close(3)                                = 0

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
```

```
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/libnorm.so.1",
O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0000\374\1\0\0\0\0\0"...,
832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=522248, ...}) = 0

mmap(NULL, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f48cb894000

mmap(NULL, 3340624, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f48ca29e000

mprotect(0x7f48ca31b000, 2097152, PROT_NONE) = 0

mmap(0x7f48ca51b000, 12288, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x7d000) =
0x7f48ca51b000

mmap(0x7f48ca51e000, 719184, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f48ca51e000

close(3)                                = 0

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/librt.so.1",
O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\"\0\0\0\0\0\0"..., 832) =
832

fstat(3, {st_mode=S_IFREG|0644, st_size=31680, ...}) = 0

mmap(NULL, 2128864, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f48ca096000

mprotect(0x7f48ca09d000, 2093056, PROT_NONE) = 0

mmap(0x7f48ca29c000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x6000) =
0x7f48ca29c000

close(3)                                = 0

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libpthread.so.0",
O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0000b\0\0\0\0\0\0"..., 832) =
832
```

```
fstat(3, {st_mode=S_IFREG|0755, st_size=144976, ...}) = 0

mmap(NULL, 2221184, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f48c9e77000

mprotect(0x7f48c9e91000, 2093056, PROT_NONE) = 0

mmap(0x7f48ca090000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x19000) =
0x7f48ca090000

mmap(0x7f48ca092000, 13440, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f48ca092000

close(3)                                = 0

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libm.so.6",
O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\200\272\0\0\0\0\0\0"...,
832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=1700792, ...}) = 0

mmap(NULL, 3789144, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f48c9ad9000

mprotect(0x7f48c9c76000, 2093056, PROT_NONE) = 0

mmap(0x7f48c9e75000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x19c000) =
0x7f48c9e75000

close(3)                                = 0

mmap(NULL, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f48cb892000

mmap(NULL, 12288, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f48cb88f000

arch_prctl(ARCH_SET_FS, 0x7f48cb88fb80) = 0

mprotect(0x7f48cae52000, 16384, PROT_READ) = 0

mprotect(0x7f48c9e75000, 4096, PROT_READ) = 0

mprotect(0x7f48ca090000, 4096, PROT_READ) = 0

mprotect(0x7f48ca29c000, 4096, PROT_READ) = 0

mprotect(0x7f48cb072000, 4096, PROT_READ) = 0
```

```
mmap(NULL, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f48cb88d000
mprotect(0x7f48cb3ed000, 40960, PROT_READ) = 0
mprotect(0x7f48ca51b000, 8192, PROT_READ) = 0
mprotect(0x7f48ca814000, 4096, PROT_READ) = 0
mprotect(0x7f48caa69000, 4096, PROT_READ) = 0
mprotect(0x7f48cb690000, 24576, PROT_READ) = 0
mprotect(0x55bfa0f00000, 4096, PROT_READ) = 0
mprotect(0x7f48cb8be000, 4096, PROT_READ) = 0
munmap(0x7f48cb898000, 152516)          = 0
set_tid_address(0x7f48cb88fe50)         = 10254
set_robust_list(0x7f48cb88fe60, 24)     = 0
rt_sigaction(SIGRTMIN, {sa_handler=0x7f48c9e7ccb0, sa_mask=[],
sa_flags=SA_RESTORER|SA_SIGINFO, sa_restorer=0x7f48c9e89890}, NULL,
8) = 0
rt_sigaction(SIGRT_1, {sa_handler=0x7f48c9e7cd50, sa_mask=[],
sa_flags=SA_RESTORER|SA_RESTART|SA_SIGINFO,
sa_restorer=0x7f48c9e89890}, NULL, 8) = 0
rt_sigprocmask(SIG_UNBLOCK, [RTMIN RT_1], NULL, 8) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024,
rlim_max=RLIM64_INFINITY}) = 0
brk(NULL)                               = 0x55bfa1b1d000
brk(0x55bfa1b3e000)                     = 0x55bfa1b3e000
futex(0x7f48cb3fa09c, FUTEX_WAKE_PRIVATE, 2147483647) = 0
futex(0x7f48cb3fa0a8, FUTEX_WAKE_PRIVATE, 2147483647) = 0
openat(AT_FDCWD, "/sys/devices/system/cpu/online",
O_RDONLY|O_CLOEXEC) = 3
read(3, "0-3\n", 8192)                  = 4
close(3)                                = 0
openat(AT_FDCWD, "/sys/devices/system/cpu",
O_RDONLY|O_NONBLOCK|O_CLOEXEC|O_DIRECTORY) = 3
fstat(3, {st_mode=S_IFDIR|0755, st_size=0, ...}) = 0
```

```
getdents(3, /* 22 entries */, 32768)    = 656
getdents(3, /* 0 entries */, 32768)     = 0
close(3)                                = 0
getpid()                                = 10254
sched_getaffinity(10254, 128, [0, 1, 2, 3]) = 8
openat(AT_FDCWD, "/etc/nsswitch.conf", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=556, ...}) = 0
read(3, "# /etc/nsswitch.conf\n#\n# Example"..., 4096) = 556
read(3, "", 4096)                       = 0
close(3)                                = 0
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=152516, ...}) = 0
mmap(NULL, 152516, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f48cb898000
close(3)                                = 0
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/tls/x86_64/x86_64/libnss_db.so.2",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu/tls/x86_64/x86_64", 0x7ffcffc476a0) = -1 ENOENT
(No such file or directory)
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/tls/x86_64/libnss_db.so.2",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu/tls/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such
file or directory)
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/tls/x86_64/libnss_db.so.2",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu/tls/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such
file or directory)
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/tls/libnss_db.so.2",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu/tls", 0x7ffcffc476a0) = -1 ENOENT (No such file or
directory)
```

```
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/x86_64/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/lib/x86_64-linux-gnu/x86_64/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/lib/x86_64-linux-gnu/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/lib/x86_64-linux-gnu/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/lib/x86_64-linux-gnu", {st_mode=S_IFDIR|0755, st_size=12288, ...}) = 0

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/tls/x86_64/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/usr/lib/x86_64-linux-gnu/tls/x86_64/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/tls/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/usr/lib/x86_64-linux-gnu/tls/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/tls/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/usr/lib/x86_64-linux-gnu/tls/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/tls/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/usr/lib/x86_64-linux-gnu/tls", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/x86_64/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
```

```
stat("/usr/lib/x86_64-linux-gnu/x86_64/x86_64", 0x7ffcffc476a0) = -1 ENOENT
(No such file or directory)

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/x86_64/libnss_db.so.2",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/usr/lib/x86_64-linux-gnu/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such
file or directory)

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/x86_64/libnss_db.so.2",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/usr/lib/x86_64-linux-gnu/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such
file or directory)

openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/libnss_db.so.2",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/usr/lib/x86_64-linux-gnu", {st_mode=S_IFDIR|0755, st_size=122880, ...}) =
0

openat(AT_FDCWD, "/lib/tls/x86_64/x86_64/libnss_db.so.2",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/lib/tls/x86_64/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "/lib/tls/x86_64/libnss_db.so.2",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/lib/tls/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/tls/x86_64/libnss_db.so.2",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/lib/tls/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/tls/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1
ENOENT (No such file or directory)

stat("/lib/tls", 0x7ffcffc476a0)        = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64/x86_64/libnss_db.so.2",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

stat("/lib/x86_64/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "/lib/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC)
= -1 ENOENT (No such file or directory)

stat("/lib/x86_64", 0x7ffcffc476a0)     = -1 ENOENT (No such file or directory)
```

```
openat(AT_FDCWD, "/lib/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64", 0x7ffcffc476a0)     = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib", {st_mode=S_IFDIR|0755, st_size=4096, ...}) = 0
openat(AT_FDCWD, "/usr/lib/tls/x86_64/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/tls/x86_64/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/tls/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/tls/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/tls/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/tls/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/tls/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/tls", 0x7ffcffc476a0)    = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/x86_64/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/x86_64/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/x86_64/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/x86_64", 0x7ffcffc476a0) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/libnss_db.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib", {st_mode=S_IFDIR|0755, st_size=12288, ...}) = 0
```

```
munmap(0x7f48cb898000, 152516)          = 0

openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3

fstat(3, {st_mode=S_IFREG|0644, st_size=152516, ...}) = 0

mmap(NULL, 152516, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f48cb898000

close(3)                                = 0

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libnss_files.so.2", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P#\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=47568, ...}) = 0

mmap(NULL, 2168632, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f48c98c7000

mprotect(0x7f48c98d2000, 2093056, PROT_NONE) = 0

mmap(0x7f48c9ad1000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xa000) = 0x7f48c9ad1000

mmap(0x7f48c9ad3000, 22328, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f48c9ad3000

close(3)                                = 0

mprotect(0x7f48c9ad1000, 4096, PROT_READ) = 0

munmap(0x7f48cb898000, 152516)          = 0

openat(AT_FDCWD, "/etc/protocols", O_RDONLY|O_CLOEXEC) = 3

fstat(3, {st_mode=S_IFREG|0644, st_size=2932, ...}) = 0

read(3, "# Internet (IP) protocols\n#\n# Up"..., 4096) = 2932

read(3, "", 4096)                       = 0

close(3)                                = 0

eventfd2(0, EFD_CLOEXEC)                = 3

fcntl(3, F_GETFL)                       = 0x2 (flags O_RDWR)

fcntl(3, F_SETFL, O_RDWR|O_NONBLOCK)    = 0

fcntl(3, F_GETFL)                       = 0x802 (flags O_RDWR|O_NONBLOCK)
fcntl(3, F_SETFL, O_RDWR|O_NONBLOCK)    = 0

getrandom("\x51\xd8\x0f\x14\xf8\xc3\xe0\xea\xe3\x16\x8a\x50\xa2\xee\x9f\x8a",
16, 0) = 16

getrandom("\x90\x23\x51\x70\xb7\x05\x5d\x81\x97\x5d\x81\x47\x9a\x67\x09\x12
", 16, 0) = 16

eventfd2(0, EFD_CLOEXEC)                = 4

fcntl(4, F_GETFL)                       = 0x2 (flags O_RDWR)

fcntl(4, F_SETFL, O_RDWR|O_NONBLOCK)    = 0

fcntl(4, F_GETFL)                       = 0x802 (flags O_RDWR|O_NONBLOCK)

fcntl(4, F_SETFL, O_RDWR|O_NONBLOCK)    = 0

epoll_create1(EPOLL_CLOEXEC)            = 5

epoll_ctl(5, EPOLL_CTL_ADD, 4, {0, {u32=2712860704,
u64=94281539975200}}) = 0

epoll_ctl(5, EPOLL_CTL_MOD, 4, {EPOLLIN, {u32=2712860704,
u64=94281539975200}}) = 0

mmap(NULL, 8392704, PROT_NONE,
MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) = 0x7f48c90c6000

mprotect(0x7f48c90c7000, 8388608, PROT_READ|PROT_WRITE) = 0

clone(child_stack=0x7f48c98c5b70,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_T
HREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CL
ONE_CHILD_CLEARTID, parent_tidptr=0x7f48c98c69d0, tls=0x7f48c98c6700,
child_tidptr=0x7f48c98c69d0) = 10255

openat(AT_FDCWD, "/proc/self/task/10255/comm", O_RDWR) = 6

write(6, "ZMQbg/0", 7)                  = 7

close(6)                                = 0

eventfd2(0, EFD_CLOEXEC)                = 6

fcntl(6, F_GETFL)                       = 0x2 (flags O_RDWR)

fcntl(6, F_SETFL, O_RDWR|O_NONBLOCK)    = 0

fcntl(6, F_GETFL)                       = 0x802 (flags O_RDWR|O_NONBLOCK)

fcntl(6, F_SETFL, O_RDWR|O_NONBLOCK)    = 0

epoll_create1(EPOLL_CLOEXEC)            = 7
```

```
epoll_ctl(7, EPOLL_CTL_ADD, 6, {0, {u32=2712876144,
u64=94281539990640}}) = 0

epoll_ctl(7, EPOLL_CTL_MOD, 6, {EPOLLIN, {u32=2712876144,
u64=94281539990640}}) = 0

mmap(NULL, 8392704, PROT_NONE,
MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) = 0x7f48c88c5000

mprotect(0x7f48c88c6000, 8388608, PROT_READ|PROT_WRITE) = 0

clone(child_stack=0x7f48c90c4b70,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_T
HREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CL
ONE_CHILD_CLEARTID, parent_tidptr=0x7f48c90c59d0, tls=0x7f48c90c5700,
child_tidptr=0x7f48c90c59d0) = 10256

openat(AT_FDCWD, "/proc/self/task/10256/comm", O_RDWR) = 8

write(8, "ZMQbg/1", 7)            = 7

close(8)                          = 0

eventfd2(0, EFD_CLOEXEC)          = 8

fcntl(8, F_GETFL)                 = 0x2 (flags O_RDWR)

fcntl(8, F_SETFL, O_RDWR|O_NONBLOCK)    = 0

fcntl(8, F_GETFL)                 = 0x802 (flags O_RDWR|O_NONBLOCK)

fcntl(8, F_SETFL, O_RDWR|O_NONBLOCK)    = 0

poll([{fd=8, events=POLLIN}], 1, 0)     = 0 (Timeout)

socket(AF_NETLINK, SOCK_RAW|SOCK_CLOEXEC, NETLINK_ROUTE) =
9

bind(9, {sa_family=AF_NETLINK, nl_pid=0, nl_groups=00000000}, 12) = 0

getsockname(9, {sa_family=AF_NETLINK, nl_pid=10254,
nl_groups=00000000}, [12]) = 0

sendto(9, {{len=20, type=RTM_GETLINK,
flags=NLM_F_REQUEST|NLM_F_DUMP, seq=1577382168, pid=0},
{ifi_family=AF_UNSPEC, ...}}, 20, 0, {sa_family=AF_NETLINK, nl_pid=0,
nl_groups=00000000}, 12) = 20

recvmsg(9, {msg_name={sa_family=AF_NETLINK, nl_pid=0,
nl_groups=00000000}, msg_namelen=12, msg_iov=[{iov_base=[{{len=1316,
type=RTM_NEWLINK, flags=NLM_F_MULTI, seq=1577382168, pid=10254},
{ifi_family=AF_UNSPEC, ifi_type=ARPHRD_LOOPBACK,
ifi_index=if_nametoindex("lo"),
ifi_flags=IFF_UP|IFF_LOOPBACK|IFF_RUNNING|0x10000, ifi_change=0},
[{{nla_len=7, nla_type=IFLA_IFNAME}, "lo"}, {{nla_len=8,
nla_type=IFLA_TXQLEN}, 1000}, {{nla_len=5, nla_type=IFLA_OPERSTATE},
0}, {{nla_len=5, nla_type=IFLA_LINKMODE}, 0}, {{nla_len=8,
nla_type=IFLA_MTU}, 65536}, {{nla_len=8, nla_type=0x32 /* IFLA_??? */},
"\x00\x00\x00\x00"}, {{nla_len=8, nla_type=0x33 /* IFLA_??? */},
"\x00\x00\x00\x00"}, {{nla_len=8, nla_type=IFLA_GROUP}, 0}, {{nla_len=8,
nla_type=IFLA_PROMISCUITY}, 0}, {{nla_len=8,
nla_type=IFLA_NUM_TX_QUEUES}, 1}, {{nla_len=8,
nla_type=IFLA_GSO_MAX_SEGS}, 65535}, {{nla_len=8,
nla_type=IFLA_GSO_MAX_SIZE}, 65536}, {{nla_len=8,
nla_type=IFLA_NUM_RX_QUEUES}, 1}, {{nla_len=5,
nla_type=IFLA_CARRIER}, 1}, {{nla_len=12, nla_type=IFLA_QDISC},
"noqueue"}, {{nla_len=8, nla_type=IFLA_CARRIER_CHANGES}, 0},
{{nla_len=5, nla_type=IFLA_PROTO_DOWN}, 0}, {{nla_len=8, nla_type=0x2f
/* IFLA_??? */}, "\x00\x00\x00\x00"}, {{nla_len=8, nla_type=0x30 /* IFLA_???
*/}, "\x00\x00\x00\x00"}, {{nla_len=36, nla_type=IFLA_MAP}, {mem_start=0,
mem_end=0, base_addr=0, irq=0, dma=0, port=0}}, {{nla_len=10,
nla_type=IFLA_ADDRESS}, "\x00\x00\x00\x00\x00\x00"}, {{nla_len=10,
nla_type=IFLA_BROADCAST}, "\x00\x00\x00\x00\x00\x00"}, {{nla_len=196,
nla_type=IFLA_STATS64}, {rx_packets=3702, tx_packets=3702,
rx_bytes=277184, tx_bytes=277184, rx_errors=0, tx_errors=0, rx_dropped=0,
tx_dropped=0, multicast=0, collisions=0, rx_length_errors=0, rx_over_errors=0,
rx_crc_errors=0, rx_frame_errors=0, rx_fifo_errors=0, rx_missed_errors=0,
tx_aborted_errors=0, tx_carrier_errors=0, tx_fifo_errors=0, tx_heartbeat_errors=0,
tx_window_errors=0, rx_compressed=0, tx_compressed=0, rx_nohandler=0}},
{{nla_len=100, nla_type=IFLA_STATS}, {rx_packets=3702, tx_packets=3702,
rx_bytes=277184, tx_bytes=277184, rx_errors=0, tx_errors=0, rx_dropped=0,
tx_dropped=0, multicast=0, collisions=0, rx_length_errors=0, rx_over_errors=0,
rx_crc_errors=0, rx_frame_errors=0, rx_fifo_errors=0, rx_missed_errors=0,
tx_aborted_errors=0, tx_carrier_errors=0, tx_fifo_errors=0, tx_heartbeat_errors=0,
tx_window_errors=0, rx_compressed=0, tx_compressed=0, rx_nohandler=0}},
{{nla_len=12, nla_type=IFLA_XDP}, {{nla_len=5,
nla_type=IFLA_XDP_ATTACHED}, 0}}, {{nla_len=760,
nla_type=IFLA_AF_SPEC},
"\x88\x00\x02\x00\x84\x00\x01\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x
00\x00\x01\x00\x00\x00\x01\x00\x00\x00\x01\x00\x00\x00"...}]}, {{len=1324,
type=RTM_NEWLINK, flags=NLM_F_MULTI, seq=1577382168, pid=10254},
{ifi_family=AF_UNSPEC, ifi_type=ARPHRD_ETHER,
ifi_index=if_nametoindex("enp1s0"),
ifi_flags=IFF_UP|IFF_BROADCAST|IFF_MULTICAST, ifi_change=0},
[{{nla_len=11, nla_type=IFLA_IFNAME}, "enp1s0"}, {{nla_len=8,
nla_type=IFLA_TXQLEN}, 1000}, {{nla_len=5, nla_type=IFLA_OPERSTATE},
2}, {{nla_len=5, nla_type=IFLA_LINKMODE}, 0}, {{nla_len=8,
nla_type=IFLA_MTU}, 1500}, {{nla_len=8, nla_type=0x32 /* IFLA_??? */},
"\x3c\x00\x00\x00"}, {{nla_len=8, nla_type=0x33 /* IFLA_??? */},
"\xf0\x23\x00\x00"}, {{nla_len=8, nla_type=IFLA_GROUP}, 0}, {{nla_len=8,
nla_type=IFLA_PROMISCUITY}, 0}, {{nla_len=8,
nla_type=IFLA_NUM_TX_QUEUES}, 1}, {{nla_len=8,
nla_type=IFLA_GSO_MAX_SEGS}, 65535}, {{nla_len=8,
nla_type=IFLA_GSO_MAX_SIZE}, 65536}, {{nla_len=8,
nla_type=IFLA_NUM_RX_QUEUES}, 1}, {{nla_len=5,
nla_type=IFLA_CARRIER}, 0}, {{nla_len=13, nla_type=IFLA_QDISC},
"fq_codel"}, {{nla_len=8, nla_type=IFLA_CARRIER_CHANGES}, 1},
{{nla_len=5, nla_type=IFLA_PROTO_DOWN}, 0}, {{nla_len=8, nla_type=0x2f
/* IFLA_??? */}, "\x00\x00\x00\x00"}, {{nla_len=8, nla_type=0x30 /* IFLA_???
*/}, "\x01\x00\x00\x00"}, {{nla_len=36, nla_type=IFLA_MAP}, {mem_start=0,
mem_end=0, base_addr=0, irq=0, dma=0, port=0}}, {{nla_len=10,
nla_type=IFLA_ADDRESS}, "\x54\xe1\xad\x32\x70\xf3"}, {{nla_len=10,
nla_type=IFLA_BROADCAST}, "\xff\xff\xff\xff\xff\xff"}, {{nla_len=196,
nla_type=IFLA_STATS64}, {rx_packets=0, tx_packets=0, rx_bytes=0,
tx_bytes=0, rx_errors=0, tx_errors=0, rx_dropped=0, tx_dropped=0, multicast=0,
collisions=0, rx_length_errors=0, rx_over_errors=0, rx_crc_errors=0,
rx_frame_errors=0, rx_fifo_errors=0, rx_missed_errors=0, tx_aborted_errors=0,
tx_carrier_errors=0, tx_fifo_errors=0, tx_heartbeat_errors=0, tx_window_errors=0,
rx_compressed=0, tx_compressed=0, rx_nohandler=0}}, {{nla_len=100,
nla_type=IFLA_STATS}, {rx_packets=0, tx_packets=0, rx_bytes=0, tx_bytes=0,
rx_errors=0, tx_errors=0, rx_dropped=0, tx_dropped=0, multicast=0, collisions=0,
rx_length_errors=0, rx_over_errors=0, rx_crc_errors=0, rx_frame_errors=0,
rx_fifo_errors=0, rx_missed_errors=0, tx_aborted_errors=0, tx_carrier_errors=0,
tx_fifo_errors=0, tx_heartbeat_errors=0, tx_window_errors=0, rx_compressed=0,
tx_compressed=0, rx_nohandler=0}}, {{nla_len=12, nla_type=IFLA_XDP},
{{nla_len=5, nla_type=IFLA_XDP_ATTACHED}, 0}}, {{nla_len=760,
nla_type=IFLA_AF_SPEC},
"\x88\x00\x02\x00\x84\x00\x01\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x
00\x00\x01\x00\x00\x00\x01\x00\x00\x00\x01\x00\x00\x00"...}]}],
iov_len=4096}], msg_iovlen=1, msg_controllen=0, msg_flags=0}, 0) = 2640
```

```
recvmsg(9, {msg_name={sa_family=AF_NETLINK, nl_pid=0,
nl_groups=00000000}, msg_namelen=12, msg_iov=[{iov_base={{len=1316,
type=RTM_NEWLINK, flags=NLM_F_MULTI, seq=1577382168, pid=10254},
{ifi_family=AF_UNSPEC, ifi_type=ARPHRD_ETHER,
ifi_index=if_nametoindex("wlp2s0"),
ifi_flags=IFF_UP|IFF_BROADCAST|IFF_RUNNING|IFF_MULTICAST|0x1000
0, ifi_change=0}, [{{nla_len=11, nla_type=IFLA_IFNAME}, "wlp2s0"},
{{nla_len=8, nla_type=IFLA_TXQLEN}, 1000}, {{nla_len=5,
nla_type=IFLA_OPERSTATE}, 6}, {{nla_len=5, nla_type=IFLA_LINKMODE},
1}, {{nla_len=8, nla_type=IFLA_MTU}, 1500}, {{nla_len=8, nla_type=0x32 /*
IFLA_??? */}, "\x00\x01\x00\x00"}, {{nla_len=8, nla_type=0x33 /* IFLA_???
*/}, "\x00\x09\x00\x00"}, {{nla_len=8, nla_type=IFLA_GROUP}, 0},
{{nla_len=8, nla_type=IFLA_PROMISCUITY}, 0}, {{nla_len=8,
nla_type=IFLA_NUM_TX_QUEUES}, 4}, {{nla_len=8,
nla_type=IFLA_GSO_MAX_SEGS}, 65535}, {{nla_len=8,
nla_type=IFLA_GSO_MAX_SIZE}, 65536}, {{nla_len=8,
nla_type=IFLA_NUM_RX_QUEUES}, 1}, {{nla_len=5,
nla_type=IFLA_CARRIER}, 1}, {{nla_len=7, nla_type=IFLA_QDISC}, "mq"},
{{nla_len=8, nla_type=IFLA_CARRIER_CHANGES}, 4}, {{nla_len=5,
nla_type=IFLA_PROTO_DOWN}, 0}, {{nla_len=8, nla_type=0x2f /* IFLA_???
*/}, "\x02\x00\x00\x00"}, {{nla_len=8, nla_type=0x30 /* IFLA_??? */},
"\x02\x00\x00\x00"}, {{nla_len=36, nla_type=IFLA_MAP}, {mem_start=0,
mem_end=0, base_addr=0, irq=0, dma=0, port=0}}, {{nla_len=10,
nla_type=IFLA_ADDRESS}, "\x34\xf6\x4b\x2d\x96\xfa"}, {{nla_len=10,
nla_type=IFLA_BROADCAST}, "\xff\xff\xff\xff\xff\xff"}, {{nla_len=196,
nla_type=IFLA_STATS64}, {rx_packets=401331, tx_packets=202239,
rx_bytes=523984672, tx_bytes=25490344, rx_errors=0, tx_errors=0,
rx_dropped=0, tx_dropped=0, multicast=0, collisions=0, rx_length_errors=0,
rx_over_errors=0, rx_crc_errors=0, rx_frame_errors=0, rx_fifo_errors=0,
rx_missed_errors=0, tx_aborted_errors=0, tx_carrier_errors=0, tx_fifo_errors=0,
tx_heartbeat_errors=0, tx_window_errors=0, rx_compressed=0, tx_compressed=0,
rx_nohandler=0}}, {{nla_len=100, nla_type=IFLA_STATS},
{rx_packets=401331, tx_packets=202239, rx_bytes=523984672,
tx_bytes=25490344, rx_errors=0, tx_errors=0, rx_dropped=0, tx_dropped=0,
multicast=0, collisions=0, rx_length_errors=0, rx_over_errors=0, rx_crc_errors=0,
rx_frame_errors=0, rx_fifo_errors=0, rx_missed_errors=0, tx_aborted_errors=0,
tx_carrier_errors=0, tx_fifo_errors=0, tx_heartbeat_errors=0, tx_window_errors=0,
rx_compressed=0, tx_compressed=0, rx_nohandler=0}}, {{nla_len=12,
nla_type=IFLA_XDP}, {{nla_len=5, nla_type=IFLA_XDP_ATTACHED}, 0}},
{{nla_len=760, nla_type=IFLA_AF_SPEC},
"\x88\x00\x02\x00\x84\x00\x01\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x
00\x00\x01\x00\x00\x00\x01\x00\x00\x00\x01\x00\x00\x00"...}]},
iov_len=4096}], msg_iovlen=1, msg_controllen=0, msg_flags=0}, 0) = 1316
```

```
recvmsg(9, {msg_name={sa_family=AF_NETLINK, nl_pid=0,
nl_groups=00000000}, msg_namelen=12, msg_iov=[{iov_base={{len=20,
type=NLMSG_DONE, flags=NLM_F_MULTI, seq=1577382168, pid=10254},
0}, iov_len=4096}], msg_iovlen=1, msg_controllen=0, msg_flags=0}, 0) = 20
```

```
sendto(9, {{len=20, type=RTM_GETADDR,
flags=NLM_F_REQUEST|NLM_F_DUMP, seq=1577382169, pid=0},
{ifa_family=AF_UNSPEC, ...}}, 20, 0, {sa_family=AF_NETLINK, nl_pid=0,
nl_groups=00000000}, 12) = 20
```

```
recvmsg(9, {msg_name={sa_family=AF_NETLINK, nl_pid=0,
nl_groups=00000000}, msg_namelen=12, msg_iov=[{iov_base=[{{len=76,
type=RTM_NEWADDR, flags=NLM_F_MULTI, seq=1577382169, pid=10254},
{ifa_family=AF_INET, ifa_prefixlen=8, ifa_flags=IFA_F_PERMANENT,
ifa_scope=RT_SCOPE_HOST, ifa_index=if_nametoindex("lo")}, [{{nla_len=8,
nla_type=IFA_ADDRESS}, 127.0.0.1}, {{nla_len=8, nla_type=IFA_LOCAL},
127.0.0.1}, {{nla_len=7, nla_type=IFA_LABEL}, "lo"}, {{nla_len=8,
nla_type=IFA_FLAGS}, IFA_F_PERMANENT}, {{nla_len=20,
nla_type=IFA_CACHEINFO}, {ifa_prefered=4294967295,
ifa_valid=4294967295, cstamp=638, tstamp=638}}]}, {{len=88,
type=RTM_NEWADDR, flags=NLM_F_MULTI, seq=1577382169, pid=10254},
{ifa_family=AF_INET, ifa_prefixlen=24, ifa_flags=0,
ifa_scope=RT_SCOPE_UNIVERSE, ifa_index=if_nametoindex("wlp2s0")},
[{{nla_len=8, nla_type=IFA_ADDRESS}, 192.168.0.103}, {{nla_len=8,
nla_type=IFA_LOCAL}, 192.168.0.103}, {{nla_len=8,
nla_type=IFA_BROADCAST}, 192.168.0.255}, {{nla_len=11,
nla_type=IFA_LABEL}, "wlp2s0"}, {{nla_len=8, nla_type=IFA_FLAGS},
IFA_F_NOPREFIXROUTE}, {{nla_len=20, nla_type=IFA_CACHEINFO},
{ifa_prefered=5042, ifa_valid=5042, cstamp=745272, tstamp=745312}}]}],
iov_len=4096}], msg_iovlen=1, msg_controllen=0, msg_flags=0}, 0) = 164
```

```
recvmsg(9, {msg_name={sa_family=AF_NETLINK, nl_pid=0,
nl_groups=00000000}, msg_namelen=12, msg_iov=[{iov_base=[{{len=72,
type=RTM_NEWADDR, flags=NLM_F_MULTI, seq=1577382169, pid=10254},
{ifa_family=AF_INET6, ifa_prefixlen=128, ifa_flags=IFA_F_PERMANENT,
ifa_scope=RT_SCOPE_HOST, ifa_index=if_nametoindex("lo")}, [{{nla_len=20,
nla_type=IFA_ADDRESS}, ::1}, {{nla_len=20, nla_type=IFA_CACHEINFO},
{ifa_prefered=4294967295, ifa_valid=4294967295, cstamp=638, tstamp=638}},
{{nla_len=8, nla_type=IFA_FLAGS}, IFA_F_PERMANENT}]}, {{len=72,
type=RTM_NEWADDR, flags=NLM_F_MULTI, seq=1577382169, pid=10254},
{ifa_family=AF_INET6, ifa_prefixlen=64, ifa_flags=IFA_F_PERMANENT,
ifa_scope=RT_SCOPE_LINK, ifa_index=if_nametoindex("wlp2s0")},
[{{nla_len=20, nla_type=IFA_ADDRESS}, fe80::8a79:36f5:d59e:af00},
{{nla_len=20, nla_type=IFA_CACHEINFO}, {ifa_prefered=4294967295,
ifa_valid=4294967295, cstamp=745266, tstamp=745434}}, {{nla_len=8,
nla_type=IFA_FLAGS}, IFA_F_PERMANENT|IFA_F_NOPREFIXROUTE}]}],
iov_len=4096}], msg_iovlen=1, msg_controllen=0, msg_flags=0}, 0) = 144
```

```
recvmsg(9, {msg_name={sa_family=AF_NETLINK, nl_pid=0,
nl_groups=00000000}, msg_namelen=12, msg_iov=[{iov_base={{len=20,
type=NLMSG_DONE, flags=NLM_F_MULTI, seq=1577382169, pid=10254},
0}, iov_len=4096}], msg_iovlen=1, msg_controllen=0, msg_flags=0}, 0) = 20

close(9)                              = 0

socket(AF_INET, SOCK_STREAM|SOCK_CLOEXEC, IPPROTO_TCP) = 9

setsockopt(9, SOL_SOCKET, SO_REUSEADDR, [1], 4) = 0

bind(9, {sa_family=AF_INET, sin_port=htons(30000),
sin_addr=inet_addr("127.0.0.1")}, 16) = 0

listen(9, 100)                        = 0

getsockname(9, {sa_family=AF_INET, sin_port=htons(30000),
sin_addr=inet_addr("127.0.0.1")}, [128->16]) = 0

write(6, "\1\0\0\0\0\0\0\0", 8)         = 8

write(8, "\1\0\0\0\0\0\0\0", 8)         = 8

fstat(0, {st_mode=S_IFCHR|0620, st_rdev=makedev(136, 0), ...}) = 0

read(0, create 1

"create 1\n", 1024)             = 9

clone(child_stack=NULL,
flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLD,
child_tidptr=0x7f48cb88fe50) = 10257

poll([{fd=8, events=POLLIN}], 1, 0)     = 1 ([{fd=8, revents=POLLIN}])

read(8, "\1\0\0\0\0\0\0\0", 8)          = 8

poll([{fd=8, events=POLLIN}], 1, 0)     = 0 (Timeout)

write(6, "\1\0\0\0\0\0\0\0", 8)         = 8

poll([{fd=8, events=POLLIN}], 1, -1)    = 1 ([{fd=8, revents=POLLIN}])

read(8, "\1\0\0\0\0\0\0\0", 8)          = 8

poll([{fd=8, events=POLLIN}], 1, 0)     = 0 (Timeout)

write(6, "\1\0\0\0\0\0\0\0", 8)         = 8

fstat(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(136, 0), ...}) = 0

write(1, "Ok:10200\n", 9Ok:10200

)               = 9

read(0, create 2

"create 2\n", 1024)             = 9
```

```
poll([{fd=8, events=POLLIN}], 1, 0)     = 1 ([{fd=8, revents=POLLIN}])
read(8, "\1\0\0\0\0\0\0\0", 8)          = 8
poll([{fd=8, events=POLLIN}], 1, 0)     = 0 (Timeout)
write(6, "\1\0\0\0\0\0\0\0", 8)         = 8
poll([{fd=8, events=POLLIN}], 1, -1)    = 1 ([{fd=8, revents=POLLIN}])
read(8, "\1\0\0\0\0\0\0\0", 8)          = 8
poll([{fd=8, events=POLLIN}], 1, 0)     = 0 (Timeout)
write(1, "Ok:10262\n", 9Ok:10262
)               = 9
read(0, exec 2 3 1 2 3
"exec 2 3 1 2 3\n", 1024)       = 15
poll([{fd=8, events=POLLIN}], 1, 0)     = 0 (Timeout)
write(6, "\1\0\0\0\0\0\0\0", 8)         = 8
poll([{fd=8, events=POLLIN}], 1, -1)    = 1 ([{fd=8, revents=POLLIN}])
read(8, "\1\0\0\0\0\0\0\0", 8)          = 8
poll([{fd=8, events=POLLIN}], 1, 0)     = 0 (Timeout)
write(6, "\1\0\0\0\0\0\0\0", 8)         = 8
write(1, "Error:2: Not found\n", 19Error:2: Not found
)    = 19
read(0, exit
"exit\n", 1024)                 = 5
write(4, "\1\0\0\0\0\0\0\0", 8)         = 8
write(8, "\1\0\0\0\0\0\0\0", 8)         = 8
poll([{fd=3, events=POLLIN}], 1, -1)    = 1 ([{fd=3, revents=POLLIN}])
read(3, "\1\0\0\0\0\0\0\0", 8)          = 8
write(6, "\1\0\0\0\0\0\0\0", 8)         = 8
close(7)                                = 0
close(6)                                = 0
close(5)                                = 0
close(4)                    = 0
close(3)                    = 0
lseek(0, -1, SEEK_CUR)               = -1 ESPIPE (Illegal seek)
exit_group(0)                  = ?
+++ exited with 0 +++
```

Вывод

В результате данной лабораторной работы я научилась работать с технологией очереди сообщений, создавать программы, создающие и связывающие процессы в определенные топологии. Так же я приобрела полезные навыки в отладке многопроцессорных приложений.