Московский Авиационный Институт

(Национальный Исследовательский Университет)

Институт №8 “Компьютерные науки и прикладная математика”

Кафедра №806 “Вычислительная математика и программирование”

**Лабораторная работа №5-7 по курсу**

**«Операционные системы»**

Группа: М8О-215Б-23

Студент: Голосов Г. С.

Преподаватель: Миронов Е.С.

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата: 06.03.25

Москва, 2025

**Постановка задачи**

**Вариант 32.**

Реализовать распределенную систему по асинхронной обработке запросов. В данной распределенной 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 технологии очередей сообщений. Также в данной системе необходимо предусмотреть проверку доступности узлов в соответствии с вариантом. Управляющий узел отвечает за ввод команд от пользователя и отправку этих команд на вычислительные узлы. Список основных поддерживаемых команд:

Создание нового вычислительного узла Формат команды: create id [parent]

id – целочисленный идентификатор нового вычислительного узла

parent – целочисленный идентификатор родительского узла.

Примечание: выполнение команд должно быть асинхронным. Т.е. пока выполняется команда на одном из вычислительных узлов, то можно отправить следующую команду на другой вычислительный узел.

Топология: все вычислительные узлы хранятся в бинарном дереве поиска. [parent] — является необязательным параметром.

Команда: локальный целочисленный словарь

Формат команды сохранения значения: exec id name value

id – целочисленный идентификатор вычислительного узла, на который отправляется команда

name – ключ, по которому будет сохранено значение (строка формата [A-Za-z0-9]+)

value – целочисленное значение

Формат команды загрузки значения: exec id name.

Проверка доступности: Формат команды: ping id

Команда проверяет доступность конкретного узла. Если узла нет, то необходимо выводить ошибку.

Формат команды: heartbit time   
Каждый узел начинает сообщать раз в time миллисекунд о том, что он работоспособен. Если от узла нет сигнала в течении 4\*time миллисекунд, то должна выводится пользователю строка: «Heartbit: node id is unavailable now», где id – идентификатор недоступного вычислительного узла

**Общий метод и алгоритм решения**

Для реализации системы очереди сообщений используем библиотеку ZeroMQ.

Использованные системные вызовы:

1. int select(int nfds, fd\_set \*readfds, fd\_set \*writefds, fd\_set \*exceptfds, struct timeval \*timeout); Ожидает готовности файловых дескрипторов.
2. pid\_t fork(void); Создает новый процесс.
3. int execl(const char \*path, const char \*arg, ...); Заменяет текущий процесс новым процессом.
4. pid\_t getpid(void); Возвращает идентификатор текущего процесса.
5. void zmq\_msg\_init\_size(zmq\_msg\_t \*msg, size\_t size); Инициализирует сообщение ZeroMQ с указанным размером.
6. int zmq\_msg\_send(zmq\_msg\_t \*msg, void \*socket, int flags); Отправляет сообщение через сокет ZeroMQ.
7. void zmq\_msg\_init(zmq\_msg\_t \*msg); Инициализирует сообщение ZeroMQ.
8. int zmq\_msg\_recv(zmq\_msg\_t \*msg, void \*socket, int flags); Получает сообщение через сокет ZeroMQ.
9. void \*zmq\_msg\_data(zmq\_msg\_t \*msg); Возвращает указатель на данные сообщения ZeroMQ.
10. void \*zmq\_ctx\_new(void); Создает новый контекст ZeroMQ.
11. void \*zmq\_socket(void \*context, int type); Создает новый сокет ZeroMQ.
12. int zmq\_connect(void \*socket, const char \*addr); Подключает сокет ZeroMQ к указанному адресу.
13. int zmq\_bind(void \*socket, const char \*addr); Привязывает сокет ZeroMQ к указанному адресу.

Управляющий узел в control.cpp принимает команды пользователя, создаёт вычислительные узлы (через fork/exec) и взаимодействует с ними посредством ZeroMQ. Он отслеживает, какие узлы активны, отправляет им команды и обрабатывает их ответы. Также он получает heartbeat-сообщения для мониторинга доступности узлов.

Вычислительные узлы в computing.cpp запускаются управляющим узлом через create. Каждый узел работает в своём процессе, имеет уникальный идентификатор (node\_id) и поддерживает локальный словарь для хранения пар «ключ-значение». Узел получает команды от управляющего узла через ZeroMQ, обрабатывает их (чтение или запись в словарь) и отправляет ответ обратно. Также узел периодически отправляет heartbeat-сообщения, информируя управляющий узел о своей работоспособности.  
Для обмена сообщениями через ZeroMQ используется модель ROUTER/DEALER. Управляющий узел использует ROUTER-сокет, а вычислительные узлы – DEALER-сокеты. Это позволяет адресовать сообщения конкретному узлу. Для heartbeat используется отдельная пара сокетов: вычислительные узлы отправляют сообщения через PUSH-сокет, а управляющий узел принимает их через PULL-сокет.

**Код программы**

lib.h

#ifndef LIB\_H

#define LIB\_H

#include <iostream>

#include <list>

#include <unordered\_set>

#include <chrono>

#include <ctime>

#include <string>

#include <cstring>

#include <unistd.h>

#include <sys/wait.h>

#include "zmq.h"

#include <sys/select.h>

#include <map>

#include <vector>

#include <sstream>

// Функция для проверки наличия ввода в STDIN без блокировки

bool inputAvailable();

// Функция для получения текущего времени в формате time\_t

std::time\_t t\_now();

// Функция для разбиения строки на токены по пробельным символам

std::vector<std::string> split(const std::string &s);

// Перечисление команд для обмена сообщениями между узлами

enum com : char {

    None = 0,

    Create = 1,

    Ping = 2,

    ExecAdd = 3,

    ExecFnd = 4,

    ExecErr = 5

};

class message {

public:

    message() {}

    // Конструктор без строки (только числовые данные)

    message(com command, int id, int num)

        : command(command), id(id), num(num), sent\_time(t\_now()) {}

    // Конструктор с дополнительной строкой (например, для передачи имени переменной)

    message(com command, int id, int num, char s[])

        : command(command), id(id), num(num), sent\_time(t\_now())

    {

        for (int i = 0; i < 30; ++i)

            st[i] = s[i];

    }

    bool operator==(const message &other) const {

        return command == other.command && id == other.id &&

               num == other.num && sent\_time == other.sent\_time;

    }

    com command;           // Тип команды

    int id;                // Идентификатор получателя (или иной контекстный id)

    int num;               // Числовой параметр (например, значение для записи)

    std::time\_t sent\_time; // Время отправки сообщения

    char st[30];           // Дополнительная строка (например, имя переменной)

};

class Node {

public:

    int id;            // Идентификатор узла

    pid\_t pid;         // Идентификатор процесса

    void \*context;     // Контекст ZeroMQ

    void \*socket;      // Сокет ZeroMQ

    std::string address; // Адрес (например, tcp://127.0.0.1:порт)

    bool operator==(const Node &other) const {

        return id == other.id && pid == other.pid;

    }

};

// Функции для создания и работы с узлами (минимальная реализация)

Node createNode(int id, bool is\_child);

Node createProcess(int id);

void send\_mes(Node &node, message m);

message get\_mes(Node &node);

#endif

lib.cpp

#include "lib.h"

#include <sstream>

#include <sys/time.h>

#include <cstdlib>

#include <cstring>

bool inputAvailable() {

    struct timeval tv;

    fd\_set fds;

    tv.tv\_sec = 0;

    tv.tv\_usec = 0;

    FD\_ZERO(&fds);

    FD\_SET(STDIN\_FILENO, &fds);

    select(STDIN\_FILENO + 1, &fds, NULL, NULL, &tv);

    return (FD\_ISSET(STDIN\_FILENO, &fds));

}

std::time\_t t\_now() {

    return std::chrono::system\_clock::to\_time\_t(std::chrono::system\_clock::now());

}

std::vector<std::string> split(const std::string &s) {

    std::istringstream iss(s);

    std::vector<std::string> tokens;

    std::string token;

    while (iss >> token)

        tokens.push\_back(token);

    return tokens;

}

Node createNode(int id, bool is\_child) {

    Node node;

    node.id = id;

    node.pid = getpid();

    node.context = zmq\_ctx\_new();

    node.socket = zmq\_socket(node.context, ZMQ\_DEALER);

    // Вычислительные узлы подключаются к ROUTER-сокету управляющего узла,

    // поэтому адрес используется фиксированный

    node.address = "tcp://localhost:5555";

    if (is\_child)

        zmq\_connect(node.socket, node.address.c\_str());

    else

        zmq\_bind(node.socket, node.address.c\_str());

    return node;

}

Node createProcess(int id) {

    pid\_t pid = fork();

    if (pid == 0) {

        // В дочернем процессе запускаем вычислительный узел

        execl("./computing", "computing", std::to\_string(id).c\_str(), NULL);

        std::cerr << "execl failed" << std::endl;

        exit(1);

    } else if (pid == -1) {

        std::cerr << "Fork failed" << std::endl;

        exit(1);

    }

    Node node = createNode(id, false);

    node.pid = pid;

    return node;

}

void send\_mes(Node &node, message m) {

    zmq\_msg\_t request\_message;

    zmq\_msg\_init\_size(&request\_message, sizeof(m));

    std::memcpy(zmq\_msg\_data(&request\_message), &m, sizeof(m));

    zmq\_msg\_send(&request\_message, node.socket, ZMQ\_DONTWAIT);

    zmq\_msg\_close(&request\_message);

}

message get\_mes(Node &node) {

    zmq\_msg\_t request;

    zmq\_msg\_init(&request);

    auto result = zmq\_msg\_recv(&request, node.socket, ZMQ\_DONTWAIT);

    if (result == -1) {

        zmq\_msg\_close(&request);

        return message(None, -1, -1);

    }

    message m;

    std::memcpy(&m, zmq\_msg\_data(&request), sizeof(message));

    zmq\_msg\_close(&request);

    return m;

}

control.cpp

#include <iostream>

#include <sstream>

#include <string>

#include <chrono>

#include <thread>

#include <mutex>

#include <queue>

#include <map>

#include <set>

#include <vector>

#include <cstdlib>

#include <csignal>

#include <cstring>

#include "zmq.h"

#include "lib.h"

using namespace std;

using namespace std::chrono;

struct NodeInfo {

    int id;

    int parent; // -1, если родитель не указан

    pid\_t pid;

    steady\_clock::time\_point lastHeartbeat;

};

mutex nodesMutex;

map<int, NodeInfo> nodes;

set<int> unavailableNodes;

int heartbeatInterval = 2000; // Интервал heartbeat (может задаваться командой heartbit)

mutex queueMutex;

queue<string> commandQueue;

// Поток для чтения команд с консоли

void inputThread() {

    string line;

    while (getline(cin, line)) {

        lock\_guard<mutex> lock(queueMutex);

        commandQueue.push(line);

    }

}

void spawnComputingNode(int nodeId, int parentId) {

    pid\_t pid = fork();

    if (pid < 0) {

        cerr << "Error: Could not fork process" << endl;

    } else if (pid == 0) {

        // В дочернем процессе запускаем вычислительный узел

        string nodeIdStr = to\_string(nodeId);

        string parentIdStr = (parentId != -1) ? to\_string(parentId) : "";

        if (parentId != -1)

            execl("./computing", "computing", nodeIdStr.c\_str(), parentIdStr.c\_str(), (char\*)NULL);

        else

            execl("./computing", "computing", nodeIdStr.c\_str(), (char\*)NULL);

        cerr << "Error: execl failed" << endl;

        exit(1);

    } else {

        // В родительском процессе сохраняем информацию об узле

        NodeInfo info;

        info.id = nodeId;

        info.parent = parentId;

        info.pid = pid;

        info.lastHeartbeat = steady\_clock::now();

        {

            lock\_guard<mutex> lock(nodesMutex);

            nodes[nodeId] = info;

        }

        cout << "Ok: " << pid << endl;

    }

}

bool isNodeAvailable(const NodeInfo &node) {

    auto now = steady\_clock::now();

    auto elapsed = duration\_cast<milliseconds>(now - node.lastHeartbeat).count();

    return elapsed <= 4 \* heartbeatInterval;

}

int main() {

    void\* context = zmq\_ctx\_new();

    // Создаем ROUTER-сокет для отправки команд вычислительным узлам

    void\* router = zmq\_socket(context, ZMQ\_ROUTER);

    zmq\_bind(router, "tcp://\*:5555");

    // Создаем PULL-сокет для приема heartbeat-сообщений от вычислительных узлов

    void\* heartbeatSocket = zmq\_socket(context, ZMQ\_PULL);

    zmq\_bind(heartbeatSocket, "tcp://\*:5557");

    // Запускаем поток для чтения команд с консоли

    thread t(inputThread);

    zmq\_pollitem\_t items[2];

    items[0].socket = router;

    items[0].fd = 0;

    items[0].events = ZMQ\_POLLIN;

    items[0].revents = 0;

    items[1].socket = heartbeatSocket;

    items[1].fd = 0;

    items[1].events = ZMQ\_POLLIN;

    items[1].revents = 0;

    while (true) {

        zmq\_poll(items, 2, 100);  // Таймаут 100 мс

        // Обработка heartbeat-сообщений

        if (items[1].revents & ZMQ\_POLLIN) {

            zmq\_msg\_t hbMsg;

            zmq\_msg\_init(&hbMsg);

            int r = zmq\_msg\_recv(&hbMsg, heartbeatSocket, 0);

            if (r != -1) {

                string hb((char\*)zmq\_msg\_data(&hbMsg), zmq\_msg\_size(&hbMsg));

                auto tokens = split(hb);

                if (tokens.size() == 2 && tokens[0] == "HEARTBEAT") {

                    int nodeId = stoi(tokens[1]);

                    lock\_guard<mutex> lock(nodesMutex);

                    if (nodes.find(nodeId) != nodes.end()) {

                        nodes[nodeId].lastHeartbeat = steady\_clock::now();

                        unavailableNodes.erase(nodeId);

                    }

                }

            }

            zmq\_msg\_close(&hbMsg);

        }

        // Обработка ответов от вычислительных узлов (через ROUTER-сокет)

        if (items[0].revents & ZMQ\_POLLIN) {

            // Ожидаем multipart-сообщение: [identity][empty][reply]

            zmq\_msg\_t identity, empty, replyMsg;

            zmq\_msg\_init(&identity);

            zmq\_msg\_recv(&identity, router, 0);

            zmq\_msg\_init(&empty);

            zmq\_msg\_recv(&empty, router, 0);

            zmq\_msg\_init(&replyMsg);

            zmq\_msg\_recv(&replyMsg, router, 0);

            string reply((char\*)zmq\_msg\_data(&replyMsg), zmq\_msg\_size(&replyMsg));

            cout << reply << endl;

            zmq\_msg\_close(&identity);

            zmq\_msg\_close(&empty);

            zmq\_msg\_close(&replyMsg);

        }

        // Проверяем очередь команд

        string cmd;

        {

            lock\_guard<mutex> lock(queueMutex);

            if (!commandQueue.empty()) {

                cmd = commandQueue.front();

                commandQueue.pop();

            }

        }

        if (!cmd.empty()) {

            auto tokens = split(cmd);

            if (tokens.empty())

                continue;

            if (tokens[0] == "create") {

                // Формат: create id [parent]

                if (tokens.size() < 2) {

                    cout << "Error: Invalid create command" << endl;

                    continue;

                }

                int newId = stoi(tokens[1]);

                int parentId = -1;

                if (tokens.size() >= 3) {

                    parentId = stoi(tokens[2]);

                    lock\_guard<mutex> lock(nodesMutex);

                    if (nodes.find(parentId) == nodes.end()) {

                        cout << "Error: Parent not found" << endl;

                        continue;

                    }

                    if (!isNodeAvailable(nodes[parentId])) {

                        cout << "Error: Parent is unavailable" << endl;

                        continue;

                    }

                }

                {

                    lock\_guard<mutex> lock(nodesMutex);

                    if (nodes.find(newId) != nodes.end()) {

                        cout << "Error: Already exists" << endl;

                        continue;

                    }

                }

                spawnComputingNode(newId, parentId);

            }

            else if (tokens[0] == "exec") {

                // Формат: exec id name [value]

                if (tokens.size() < 3) {

                    cout << "Error: Invalid exec command" << endl;

                    continue;

                }

                int targetId = stoi(tokens[1]);

                {

                    lock\_guard<mutex> lock(nodesMutex);

                    if (nodes.find(targetId) == nodes.end()) {

                        cout << "Error:" << targetId << ": Not found" << endl;

                        continue;

                    }

                    if (!isNodeAvailable(nodes[targetId])) {

                        cout << "Error:" << targetId << ": Node is unavailable" << endl;

                        continue;

                    }

                }

                string nodeCommand;

                for (size\_t i = 2; i < tokens.size(); ++i)

                    nodeCommand += tokens[i] + (i != tokens.size() - 1 ? " " : "");

                // Отправляем сообщение через ROUTER-сокет:

                // Формат: [identity][empty][command]

                string targetIdStr = to\_string(targetId);

                zmq\_msg\_t identityMsg;

                zmq\_msg\_init\_size(&identityMsg, targetIdStr.size());

                memcpy(zmq\_msg\_data(&identityMsg), targetIdStr.c\_str(), targetIdStr.size());

                zmq\_msg\_send(&identityMsg, router, ZMQ\_SNDMORE);

                zmq\_msg\_close(&identityMsg);

                zmq\_msg\_t emptyMsg;

                zmq\_msg\_init\_size(&emptyMsg, 0);

                zmq\_msg\_send(&emptyMsg, router, ZMQ\_SNDMORE);

                zmq\_msg\_close(&emptyMsg);

                zmq\_msg\_t commandMsg;

                zmq\_msg\_init\_size(&commandMsg, nodeCommand.size());

                memcpy(zmq\_msg\_data(&commandMsg), nodeCommand.c\_str(), nodeCommand.size());

                zmq\_msg\_send(&commandMsg, router, 0);

                zmq\_msg\_close(&commandMsg);

            }

            else if (tokens[0] == "heartbit") {

                // Формат: heartbit time

                if (tokens.size() < 2) {

                    cout << "Error: Invalid heartbit command" << endl;

                    continue;

                }

                heartbeatInterval = stoi(tokens[1]);

                cout << "Ok" << endl;

            }

            else if (tokens[0] == "ping") {

                // Формат: ping id

                if (tokens.size() < 2) {

                    cout << "Error: Invalid ping command" << endl;

                    continue;

                }

                int pingId = stoi(tokens[1]);

                bool available = false;

                {

                    lock\_guard<mutex> lock(nodesMutex);

                    if (nodes.find(pingId) != nodes.end())

                        available = isNodeAvailable(nodes[pingId]);

                }

                cout << "Ok: " << (available ? "1" : "0") << endl;

            }

            else {

                cout << "Error: Unknown command" << endl;

            }

        }

        // Периодическая проверка доступности узлов

        {

            lock\_guard<mutex> lock(nodesMutex);

            auto now = steady\_clock::now();

            for (auto &pair : nodes) {

                int nodeId = pair.first;

                auto &node = pair.second;

                auto elapsed = duration\_cast<milliseconds>(now - node.lastHeartbeat).count();

                if (elapsed > 4 \* heartbeatInterval) {

                    if (unavailableNodes.find(nodeId) == unavailableNodes.end()) {

                        cout << "Heartbit: node " << nodeId << " is unavailable now" << endl;

                        unavailableNodes.insert(nodeId);

                    }

                }

            }

        }

        this\_thread::sleep\_for(chrono::milliseconds(50));

    }

    t.join();

    zmq\_close(router);

    zmq\_close(heartbeatSocket);

    zmq\_ctx\_destroy(context);

    return 0;

}

computing.cpp

#include <iostream>

#include <string>

#include <chrono>

#include <thread>

#include <map>

#include <mutex>

#include <cstring>

#include "zmq.h"

#include "lib.h"

using namespace std;

std::map<std::string, int> localDict;

std::mutex dictMutex;

int main(int argc, char\* argv[]) {

    if (argc < 2) {

        std::cerr << "Usage: " << argv[0] << " <node\_id> [parent\_id]" << std::endl;

        return 1;

    }

    string node\_id = argv[1];

    // Создаем контекст и DEALER-сокет через C API

    void\* context = zmq\_ctx\_new();

    void\* dealer = zmq\_socket(context, ZMQ\_DEALER);

    // Устанавливаем identity (идентификатор узла) для возможности маршрутизации

    zmq\_setsockopt(dealer, ZMQ\_IDENTITY, node\_id.c\_str(), node\_id.size());

    zmq\_connect(dealer, "tcp://localhost:5555");

    // Создаем PUSH-сокет для отправки heartbeat-сообщений

    void\* heartbeatSocket = zmq\_socket(context, ZMQ\_PUSH);

    zmq\_connect(heartbeatSocket, "tcp://localhost:5557");

    int heartbeatInterval = 2000; // Интервал heartbeat в мс

    // Поток отправки heartbeat-сообщений

    std::thread heartbeatThread([&]() {

        while (true) {

            string hb = "HEARTBEAT " + node\_id;

            zmq\_msg\_t msg;

            zmq\_msg\_init\_size(&msg, hb.size());

            memcpy(zmq\_msg\_data(&msg), hb.c\_str(), hb.size());

            zmq\_msg\_send(&msg, heartbeatSocket, 0);

            zmq\_msg\_close(&msg);

            std::this\_thread::sleep\_for(std::chrono::milliseconds(heartbeatInterval));

        }

    });

    // Основной цикл обработки входящих команд

    while (true) {

        zmq\_msg\_t msg;

        zmq\_msg\_init(&msg);

        int res = zmq\_msg\_recv(&msg, dealer, 0);

        if (res == -1) {

            zmq\_msg\_close(&msg);

            continue;

        }

        // Если первый фрейм пустой (делимитер), пропускаем его и читаем следующий

        if (zmq\_msg\_size(&msg) == 0) {

            zmq\_msg\_close(&msg);

            zmq\_msg\_init(&msg);

            res = zmq\_msg\_recv(&msg, dealer, 0);

            if (res == -1) {

                zmq\_msg\_close(&msg);

                continue;

            }

        }

        string command((char\*)zmq\_msg\_data(&msg), zmq\_msg\_size(&msg));

        zmq\_msg\_close(&msg);

        auto tokens = split(command);

        string reply;

        if (tokens.size() == 1) {  // Запрос на чтение: exec id name

            string key = tokens[0];

            lock\_guard<mutex> lock(dictMutex);

            auto it = localDict.find(key);

            if (it != localDict.end())

                reply = "Ok:" + node\_id + ": " + to\_string(it->second);

            else

                reply = "Ok:" + node\_id + ": '" + key + "' not found";

        }

        else if (tokens.size() == 2) {  // Запись: exec id name value

            string key = tokens[0];

            int value = stoi(tokens[1]);

            {

                lock\_guard<mutex> lock(dictMutex);

                localDict[key] = value;

            }

            reply = "Ok:" + node\_id;

        } else {

            reply = "Error:" + node\_id + ": Invalid command format";

        }

        // Отправляем ответ управляющему узлу в виде мультифреймового сообщения:

        // первый фрейм – пустой делимитер, второй – данные ответа.

        zmq\_msg\_t emptyMsg;

        zmq\_msg\_init\_size(&emptyMsg, 0);

        zmq\_msg\_send(&emptyMsg, dealer, ZMQ\_SNDMORE);

        zmq\_msg\_close(&emptyMsg);

        zmq\_msg\_t replyMsg;

        zmq\_msg\_init\_size(&replyMsg, reply.size());

        memcpy(zmq\_msg\_data(&replyMsg), reply.c\_str(), reply.size());

        zmq\_msg\_send(&replyMsg, dealer, 0);

        zmq\_msg\_close(&replyMsg);

    }

    heartbeatThread.join(); // На самом деле выполнение сюда не доходит

    zmq\_close(dealer);

    zmq\_close(heartbeatSocket);

    zmq\_ctx\_destroy(context);

    return 0;

}

**Протокол работы программы**

**Тестирование:**

tobiklosj@LAPTOP-C3C2PI9E:~/labs\_OS/lab5\_7/build$ ./control

create 10

Ok: 40054

exec 10 X

Ok:10: 'X' not found

exec 10 X 5

Ok:10

exec 10 X

Ok:10: 5

exec 10 X 8

Ok:10

exec 10 X

Ok:10: 8

heartbit 2000

Ok

ping 10

Ok: 1

ping 2

Ok: 0

ping 10

Ok: 1

create 12 10

Ok: 40486

**Strace:**

82135 epoll\_create1(EPOLL\_CLOEXEC)      = 5

82135 epoll\_ctl(5, EPOLL\_CTL\_ADD, 4, {events=0, data={u32=717800640, u64=94421278835904}}) = 0

82135 epoll\_ctl(5, EPOLL\_CTL\_MOD, 4, {events=EPOLLIN, data={u32=717800640, u64=94421278835904}}) = 0

82135 epoll\_create1(EPOLL\_CLOEXEC)      = 7

82135 epoll\_ctl(7, EPOLL\_CTL\_ADD, 6, {events=0, data={u32=717804560, u64=94421278839824}}) = 0

82135 epoll\_ctl(7, EPOLL\_CTL\_MOD, 6, {events=EPOLLIN, data={u32=717804560, u64=94421278839824}}) = 0

82135 socket(AF\_INET, SOCK\_STREAM|SOCK\_CLOEXEC, IPPROTO\_TCP) = 9

82135 bind(9, {sa\_family=AF\_INET, sin\_port=htons(5555), sin\_addr=inet\_addr("0.0.0.0")}, 16) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_ADD, 9, {events=0, data={u32=2415922032, u64=140349062253424}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 9, {events=EPOLLIN, data={u32=2415922032, u64=140349062253424}}) = 0

82135 socket(AF\_INET, SOCK\_STREAM|SOCK\_CLOEXEC, IPPROTO\_TCP) = 11

82135 bind(11, {sa\_family=AF\_INET, sin\_port=htons(5557), sin\_addr=inet\_addr("0.0.0.0")}, 16) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_ADD, 11, {events=0, data={u32=2415922064, u64=140349062253456}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=2415922064, u64=140349062253456}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_ADD, 12, {events=0, data={u32=2415922160, u64=140349062253552}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 12, {events=EPOLLIN, data={u32=2415922160, u64=140349062253552}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 12, {events=EPOLLIN|EPOLLOUT, data={u32=2415922160, u64=140349062253552}}) = 0

82137 recvfrom(12, "\377\0\0\0\0\0\0\0\1\177", 12, 0, NULL, NULL) = 10

82137 recvfrom(12, 0x7fa590001322, 2, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 sendto(12, "\377\0\0\0\0\0\0\0\1\177\3", 11, 0, NULL, 0) = 11

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 12, {events=EPOLLIN, data={u32=2415922160, u64=140349062253552}}) = 0

82137 recvfrom(12, "\3\1", 2, 0, NULL, NULL) = 2

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 12, {events=EPOLLIN|EPOLLOUT, data={u32=2415922160, u64=140349062253552}}) = 0

82137 recvfrom(12, "NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 52, 0, NULL, NULL) = 52

82137 recvfrom(12, 0x7fa590003b68, 8192, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 sendto(12, "\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, 0) = 53

82137 sendto(12, "\4\32\5READY\vSocket-Type\0\0\0\4PULL", 28, 0, NULL, 0) = 28

82137 recvfrom(12, "\4\32\5READY\vSocket-Type\0\0\0\4PUSH\0\fHE"..., 8192, 0, NULL, NULL) = 2842

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 12, {events=EPOLLIN, data={u32=2415922160, u64=140349062253552}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_ADD, 13, {events=0, data={u32=2415988960, u64=140349062320352}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 13, {events=EPOLLIN, data={u32=2415988960, u64=140349062320352}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 13, {events=EPOLLIN|EPOLLOUT, data={u32=2415988960, u64=140349062320352}}) = 0

82137 recvfrom(13, "\377\0\0\0\0\0\0\0\3\177", 12, 0, NULL, NULL) = 10

82137 recvfrom(13, 0x7fa590011812, 2, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 sendto(13, "\377\0\0\0\0\0\0\0\1\177\3", 11, 0, NULL, 0) = 11

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 13, {events=EPOLLIN, data={u32=2415988960, u64=140349062320352}}) = 0

82137 recvfrom(13, "\3\1", 2, 0, NULL, NULL) = 2

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 13, {events=EPOLLIN|EPOLLOUT, data={u32=2415988960, u64=140349062320352}}) = 0

82137 recvfrom(13, "NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 52, 0, NULL, NULL) = 52

82137 recvfrom(13, 0x7fa590013fa8, 8192, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 sendto(13, "\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, 0) = 53

82137 sendto(13, "\4)\5READY\vSocket-Type\0\0\0\6ROUTER\10I"..., 43, 0, NULL, 0) = 43

82137 recvfrom(13, "\4+\5READY\vSocket-Type\0\0\0\6DEALER\10I"..., 8192, 0, NULL, NULL) = 45

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 13, {events=EPOLLIN, data={u32=2415988960, u64=140349062320352}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_ADD, 14, {events=0, data={u32=2416055760, u64=140349062387152}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 14, {events=EPOLLIN, data={u32=2416055760, u64=140349062387152}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 14, {events=EPOLLIN|EPOLLOUT, data={u32=2416055760, u64=140349062387152}}) = 0

82137 recvfrom(14, "\377\0\0\0\0\0\0\0\1\177", 12, 0, NULL, NULL) = 10

82137 recvfrom(14, 0x7fa590021d02, 2, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 sendto(14, "\377\0\0\0\0\0\0\0\1\177\3", 11, 0, NULL, 0) = 11

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 14, {events=EPOLLIN, data={u32=2416055760, u64=140349062387152}}) = 0

82137 recvfrom(14, "\3\1", 2, 0, NULL, NULL) = 2

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 14, {events=EPOLLIN|EPOLLOUT, data={u32=2416055760, u64=140349062387152}}) = 0

82137 recvfrom(14, "NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 52, 0, NULL, NULL) = 52

82137 recvfrom(14, 0x7fa590024498, 8192, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 sendto(14, "\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, 0) = 53

82137 sendto(14, "\4\32\5READY\vSocket-Type\0\0\0\4PULL", 28, 0, NULL, 0) = 28

82137 recvfrom(14, "\4\32\5READY\vSocket-Type\0\0\0\4PUSH\0\fHE"..., 8192, 0, NULL, NULL) = 2842

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 14, {events=EPOLLIN, data={u32=2416055760, u64=140349062387152}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_ADD, 15, {events=0, data={u32=2416121872, u64=140349062453264}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 15, {events=EPOLLIN, data={u32=2416121872, u64=140349062453264}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 15, {events=EPOLLIN|EPOLLOUT, data={u32=2416121872, u64=140349062453264}}) = 0

82137 recvfrom(15, "\377\0\0\0\0\0\0\0\3\177", 12, 0, NULL, NULL) = 10

82137 recvfrom(15, 0x7fa590031f42, 2, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 sendto(15, "\377\0\0\0\0\0\0\0\1\177\3", 11, 0, NULL, 0) = 11

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 15, {events=EPOLLIN, data={u32=2416121872, u64=140349062453264}}) = 0

82137 recvfrom(15, "\3\1", 2, 0, NULL, NULL) = 2

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 15, {events=EPOLLIN|EPOLLOUT, data={u32=2416121872, u64=140349062453264}}) = 0

82137 recvfrom(15, "NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 52, 0, NULL, NULL) = 52

82137 recvfrom(15, 0x7fa590034b68, 8192, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 sendto(15, "\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, 0) = 53

82137 sendto(15, "\4)\5READY\vSocket-Type\0\0\0\6ROUTER\10I"..., 43, 0, NULL, 0) = 43

82137 recvfrom(15, "\4+\5READY\vSocket-Type\0\0\0\6DEALER\10I"..., 8192, 0, NULL, NULL) = 45

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 15, {events=EPOLLIN, data={u32=2416121872, u64=140349062453264}}) = 0

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82155 epoll\_create1(EPOLL\_CLOEXEC)      = 5

82155 epoll\_ctl(5, EPOLL\_CTL\_ADD, 4, {events=0, data={u32=2545193568, u64=93869055447648}}) = 0

82155 epoll\_ctl(5, EPOLL\_CTL\_MOD, 4, {events=EPOLLIN, data={u32=2545193568, u64=93869055447648}}) = 0

82155 epoll\_create1(EPOLL\_CLOEXEC)      = 7

82155 epoll\_ctl(7, EPOLL\_CTL\_ADD, 6, {events=0, data={u32=2545214048, u64=93869055468128}}) = 0

82155 epoll\_ctl(7, EPOLL\_CTL\_MOD, 6, {events=EPOLLIN, data={u32=2545214048, u64=93869055468128}}) = 0

82157 socket(AF\_UNIX, SOCK\_STREAM|SOCK\_CLOEXEC|SOCK\_NONBLOCK, 0) = 10

82157 connect(10, {sa\_family=AF\_UNIX, sun\_path="/var/run/nscd/socket"}, 110) = -1 ENOENT (No such file or directory)

82157 socket(AF\_UNIX, SOCK\_STREAM|SOCK\_CLOEXEC|SOCK\_NONBLOCK, 0) = 10

82157 connect(10, {sa\_family=AF\_UNIX, sun\_path="/var/run/nscd/socket"}, 110) = -1 ENOENT (No such file or directory)

82157 socket(AF\_INET, SOCK\_STREAM|SOCK\_CLOEXEC, IPPROTO\_TCP) = 10

82157 connect(10, {sa\_family=AF\_INET, sin\_port=htons(5555), sin\_addr=inet\_addr("127.0.0.1")}, 16) = -1 EINPROGRESS (Operation now in progress)

82157 epoll\_ctl(7, EPOLL\_CTL\_ADD, 10, {events=0, data={u32=939534304, u64=140197262010336}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 socket(AF\_INET, SOCK\_STREAM|SOCK\_CLOEXEC, IPPROTO\_TCP) = 11

82157 connect(11, {sa\_family=AF\_INET, sin\_port=htons(5557), sin\_addr=inet\_addr("127.0.0.1")}, 16) = -1 EINPROGRESS (Operation now in progress)

82157 epoll\_ctl(7, EPOLL\_CTL\_ADD, 11, {events=0, data={u32=939530064, u64=140197262006096}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLOUT, data={u32=939530064, u64=140197262006096}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_ADD, 16, {events=0, data={u32=2416189632, u64=140349062521024}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 16, {events=EPOLLIN, data={u32=2416189632, u64=140349062521024}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 16, {events=EPOLLIN|EPOLLOUT, data={u32=2416189632, u64=140349062521024}} <unfinished ...>

82157 epoll\_ctl(7, EPOLL\_CTL\_DEL, 10, 0x7f82380027e4) = 0

82137 <... epoll\_ctl resumed>)          = 0

82137 recvfrom(16, 0x7fa5900427e8, 12, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82157 epoll\_ctl(7, EPOLL\_CTL\_DEL, 11, 0x7f8238001754) = 0

82137 sendto(16, "\377\0\0\0\0\0\0\0\1\177", 10, 0, NULL, 0) = 10

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 16, {events=EPOLLIN, data={u32=2416189632, u64=140349062521024}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_ADD, 10, {events=0, data={u32=939530064, u64=140197262006096}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN, data={u32=939530064, u64=140197262006096}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN|EPOLLOUT, data={u32=939530064, u64=140197262006096}}) = 0

82157 recvfrom(10, "\377\0\0\0\0\0\0\0\1\177", 12, 0, NULL, NULL) = 10

82137 epoll\_ctl(7, EPOLL\_CTL\_ADD, 17, {events=0, data={u32=2416193184, u64=140349062524576}} <unfinished ...>

82157 recvfrom(10,  <unfinished ...>

82137 <... epoll\_ctl resumed>)          = 0

82157 <... recvfrom resumed>0x7f8238002572, 2, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 17, {events=EPOLLIN, data={u32=2416193184, u64=140349062524576}} <unfinished ...>

82157 epoll\_ctl(7, EPOLL\_CTL\_ADD, 11, {events=0, data={u32=939534304, u64=140197262010336}} <unfinished ...>

82137 <... epoll\_ctl resumed>)          = 0

82157 <... epoll\_ctl resumed>)          = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 17, {events=EPOLLIN|EPOLLOUT, data={u32=2416193184, u64=140349062524576}} <unfinished ...>

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}} <unfinished ...>

82137 <... epoll\_ctl resumed>)          = 0

82157 <... epoll\_ctl resumed>)          = 0

82137 recvfrom(17,  <unfinished ...>

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}} <unfinished ...>

82137 <... recvfrom resumed>0x7fa5900435c8, 12, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82157 <... epoll\_ctl resumed>)          = 0

82157 recvfrom(11, 0x7f8238002f58, 12, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 sendto(17, "\377\0\0\0\0\0\0\0\1\177", 10, 0, NULL, 0) = 10

82157 sendto(10, "\377\0\0\0\0\0\0\0\3\177\3", 11, 0, NULL, 0 <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 17, {events=EPOLLIN, data={u32=2416193184, u64=140349062524576}} <unfinished ...>

82157 <... sendto resumed>)             = 11

82137 <... epoll\_ctl resumed>)          = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN, data={u32=939530064, u64=140197262006096}}) = 0

82157 sendto(11, "\377\0\0\0\0\0\0\0\1\177", 10, 0, NULL, 0 <unfinished ...>

82137 recvfrom(16,  <unfinished ...>

82157 <... sendto resumed>)             = 10

82137 <... recvfrom resumed>"\377\0\0\0\0\0\0\0\3\177\3", 12, 0, NULL, NULL) = 11

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}} <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 16, {events=EPOLLIN|EPOLLOUT, data={u32=2416189632, u64=140349062521024}} <unfinished ...>

82157 <... epoll\_ctl resumed>)          = 0

82137 <... epoll\_ctl resumed>)          = 0

82157 recvfrom(11,  <unfinished ...>

82137 recvfrom(16,  <unfinished ...>

82157 <... recvfrom resumed>"\377\0\0\0\0\0\0\0\1\177", 12, 0, NULL, NULL) = 10

82137 <... recvfrom resumed>0x7fa5900427f3, 53, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 recvfrom(11,  <unfinished ...>

82137 sendto(16, "\3\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 54, 0, NULL, 0 <unfinished ...>

82157 <... recvfrom resumed>0x7f8238002f62, 2, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 <... sendto resumed>)             = 54

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 16, {events=EPOLLIN, data={u32=2416189632, u64=140349062521024}}) = 0

82157 recvfrom(10,  <unfinished ...>

82137 recvfrom(17,  <unfinished ...>

82157 <... recvfrom resumed>"\3\1", 2, 0, NULL, NULL) = 2

82137 <... recvfrom resumed>"\377\0\0\0\0\0\0\0\1\177", 12, 0, NULL, NULL) = 10

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN|EPOLLOUT, data={u32=939530064, u64=140197262006096}} <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 17, {events=EPOLLIN|EPOLLOUT, data={u32=2416193184, u64=140349062524576}} <unfinished ...>

82157 <... epoll\_ctl resumed>)          = 0

82137 <... epoll\_ctl resumed>)          = 0

82157 recvfrom(10,  <unfinished ...>

82137 recvfrom(17,  <unfinished ...>

82157 <... recvfrom resumed>"NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 52, 0, NULL, NULL) = 52

82137 <... recvfrom resumed>0x7fa5900435d2, 2, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82157 recvfrom(10, 0x7f8238005048, 8192, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82157 sendto(11, "\3", 1, 0, NULL, 0 <unfinished ...>

82137 sendto(17, "\3", 1, 0, NULL, 0 <unfinished ...>

82157 <... sendto resumed>)             = 1

82137 <... sendto resumed>)             = 1

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}} <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 17, {events=EPOLLIN, data={u32=2416193184, u64=140349062524576}} <unfinished ...>

82157 <... epoll\_ctl resumed>)          = 0

82137 <... epoll\_ctl resumed>)          = 0

82157 sendto(10, "\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, 0 <unfinished ...>

82137 recvfrom(17,  <unfinished ...>

82157 <... sendto resumed>)             = 53

82137 <... recvfrom resumed>"\3", 2, 0, NULL, NULL) = 1

82157 recvfrom(11,  <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 17, {events=EPOLLIN|EPOLLOUT, data={u32=2416193184, u64=140349062524576}} <unfinished ...>

82157 <... recvfrom resumed>"\3", 2, 0, NULL, NULL) = 1

82137 <... epoll\_ctl resumed>)          = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}} <unfinished ...>

82137 recvfrom(17,  <unfinished ...>

82157 <... epoll\_ctl resumed>)          = 0

82137 <... recvfrom resumed>0x7fa5900435d3, 53, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82157 recvfrom(11, 0x7f8238002f63, 53, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 sendto(17, "\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, 0) = 53

82157 sendto(10, "\4+\5READY\vSocket-Type\0\0\0\6DEALER\10I"..., 45, 0, NULL, 0 <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 17, {events=EPOLLIN, data={u32=2416193184, u64=140349062524576}} <unfinished ...>

82157 <... sendto resumed>)             = 45

82137 <... epoll\_ctl resumed>)          = 0

82157 sendto(11, "\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, 0 <unfinished ...>

82137 recvfrom(16,  <unfinished ...>

82157 <... sendto resumed>)             = 53

82137 <... recvfrom resumed>"\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, NULL) = 53

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN, data={u32=939530064, u64=140197262006096}} <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 16, {events=EPOLLIN|EPOLLOUT, data={u32=2416189632, u64=140349062521024}} <unfinished ...>

82157 <... epoll\_ctl resumed>)          = 0

82137 <... epoll\_ctl resumed>)          = 0

82157 recvfrom(11,  <unfinished ...>

82137 recvfrom(16,  <unfinished ...>

82157 <... recvfrom resumed>"\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, NULL) = 53

82137 <... recvfrom resumed>"\4+\5READY\vSocket-Type\0\0\0\6DEALER\10I"..., 8192, 0, NULL, NULL) = 45

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 recvfrom(11,  <unfinished ...>

82137 recvfrom(17,  <unfinished ...>

82157 <... recvfrom resumed>0x7f823800bd48, 8192, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 <... recvfrom resumed>"\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, NULL) = 53

82157 sendto(11, "\4\32\5READY\vSocket-Type\0\0\0\4PUSH", 28, 0, NULL, 0) = 28

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 17, {events=EPOLLIN|EPOLLOUT, data={u32=2416193184, u64=140349062524576}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}} <unfinished ...>

82137 recvfrom(17,  <unfinished ...>

82157 <... epoll\_ctl resumed>)          = 0

82137 <... recvfrom resumed>"\4\32\5READY\vSocket-Type\0\0\0\4PUSH", 8192, 0, NULL, NULL) = 28

82137 sendto(16, "\4)\5READY\vSocket-Type\0\0\0\6ROUTER\10I"..., 43, 0, NULL, 0) = 43

82157 recvfrom(10, "\4)\5READY\vSocket-Type\0\0\0\6ROUTER\10I"..., 8192, 0, NULL, NULL) = 43

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN|EPOLLOUT, data={u32=939530064, u64=140197262006096}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN, data={u32=939530064, u64=140197262006096}}) = 0

82137 sendto(17, "\4\32\5READY\vSocket-Type\0\0\0\4PULL", 28, 0, NULL, 0) = 28

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 16, {events=EPOLLIN, data={u32=2416189632, u64=140349062521024}} <unfinished ...>

82157 recvfrom(11,  <unfinished ...>

82137 <... epoll\_ctl resumed>)          = 0

82157 <... recvfrom resumed>"\4\32\5READY\vSocket-Type\0\0\0\4PULL", 8192, 0, NULL, NULL) = 28

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 17, {events=EPOLLIN, data={u32=2416193184, u64=140349062524576}} <unfinished ...>

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0 <unfinished ...>

82137 <... epoll\_ctl resumed>)          = 0

82157 <... sendto resumed>)             = 14

82137 recvfrom(17,  <unfinished ...>

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}} <unfinished ...>

82137 <... recvfrom resumed>"\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 <... epoll\_ctl resumed>)          = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 13, {events=EPOLLIN|EPOLLOUT, data={u32=2415988960, u64=140349062320352}}) = 0

82137 sendto(13, "\1\0\0\1X", 5, 0, NULL, 0) = 5

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 13, {events=EPOLLIN, data={u32=2415988960, u64=140349062320352}}) = 0

82137 recvfrom(13, "\1\0\0\10Ok:12: 2", 8192, 0, NULL, NULL) = 12

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 13, {events=EPOLLIN|EPOLLOUT, data={u32=2415988960, u64=140349062320352}}) = 0

82137 sendto(13, "\1\0\0\4X 15", 8, 0, NULL, 0) = 8

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 13, {events=EPOLLIN, data={u32=2415988960, u64=140349062320352}}) = 0

82137 recvfrom(13, "\1\0\0\5Ok:12", 8192, 0, NULL, NULL) = 9

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82212 epoll\_create1(EPOLL\_CLOEXEC)      = 5

82212 epoll\_ctl(5, EPOLL\_CTL\_ADD, 4, {events=0, data={u32=707904096, u64=94249470247520}}) = 0

82212 epoll\_ctl(5, EPOLL\_CTL\_MOD, 4, {events=EPOLLIN, data={u32=707904096, u64=94249470247520}}) = 0

82212 epoll\_create1(EPOLL\_CLOEXEC)      = 7

82212 epoll\_ctl(7, EPOLL\_CTL\_ADD, 6, {events=0, data={u32=707924576, u64=94249470268000}}) = 0

82212 epoll\_ctl(7, EPOLL\_CTL\_MOD, 6, {events=EPOLLIN, data={u32=707924576, u64=94249470268000}}) = 0

82215 socket(AF\_UNIX, SOCK\_STREAM|SOCK\_CLOEXEC|SOCK\_NONBLOCK, 0) = 10

82215 connect(10, {sa\_family=AF\_UNIX, sun\_path="/var/run/nscd/socket"}, 110) = -1 ENOENT (No such file or directory)

82215 socket(AF\_UNIX, SOCK\_STREAM|SOCK\_CLOEXEC|SOCK\_NONBLOCK, 0) = 10

82215 connect(10, {sa\_family=AF\_UNIX, sun\_path="/var/run/nscd/socket"}, 110) = -1 ENOENT (No such file or directory)

82215 socket(AF\_INET, SOCK\_STREAM|SOCK\_CLOEXEC, IPPROTO\_TCP) = 10

82215 connect(10, {sa\_family=AF\_INET, sin\_port=htons(5555), sin\_addr=inet\_addr("127.0.0.1")}, 16) = -1 EINPROGRESS (Operation now in progress)

82215 epoll\_ctl(7, EPOLL\_CTL\_ADD, 10, {events=0, data={u32=1342187488, u64=140227729434592}}) = 0

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82215 socket(AF\_INET, SOCK\_STREAM|SOCK\_CLOEXEC, IPPROTO\_TCP) = 11

82215 connect(11, {sa\_family=AF\_INET, sin\_port=htons(5557), sin\_addr=inet\_addr("127.0.0.1")}, 16) = -1 EINPROGRESS (Operation now in progress)

82215 epoll\_ctl(7, EPOLL\_CTL\_ADD, 11, {events=0, data={u32=1342183248, u64=140227729430352}}) = 0

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLOUT, data={u32=1342183248, u64=140227729430352}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_ADD, 18, {events=0, data={u32=2416322128, u64=140349062653520}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 18, {events=EPOLLIN, data={u32=2416322128, u64=140349062653520}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 18, {events=EPOLLIN|EPOLLOUT, data={u32=2416322128, u64=140349062653520}} <unfinished ...>

82215 epoll\_ctl(7, EPOLL\_CTL\_DEL, 10, 0x7f89500027e4 <unfinished ...>

82137 <... epoll\_ctl resumed>)          = 0

82215 <... epoll\_ctl resumed>)          = 0

82137 recvfrom(18, 0x7fa590062d78, 12, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82215 epoll\_ctl(7, EPOLL\_CTL\_DEL, 11, 0x7f8950001754) = 0

82137 sendto(18, "\377\0\0\0\0\0\0\0\1\177", 10, 0, NULL, 0) = 10

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 18, {events=EPOLLIN, data={u32=2416322128, u64=140349062653520}}) = 0

82215 epoll\_ctl(7, EPOLL\_CTL\_ADD, 10, {events=0, data={u32=1342183248, u64=140227729430352}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_ADD, 19, {events=0, data={u32=2416325680, u64=140349062657072}} <unfinished ...>

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN, data={u32=1342183248, u64=140227729430352}} <unfinished ...>

82137 <... epoll\_ctl resumed>)          = 0

82215 <... epoll\_ctl resumed>)          = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 19, {events=EPOLLIN, data={u32=2416325680, u64=140349062657072}} <unfinished ...>

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN|EPOLLOUT, data={u32=1342183248, u64=140227729430352}} <unfinished ...>

82137 <... epoll\_ctl resumed>)          = 0

82215 <... epoll\_ctl resumed>)          = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 19, {events=EPOLLIN|EPOLLOUT, data={u32=2416325680, u64=140349062657072}} <unfinished ...>

82215 recvfrom(10,  <unfinished ...>

82137 <... epoll\_ctl resumed>)          = 0

82215 <... recvfrom resumed>"\377\0\0\0\0\0\0\0\1\177", 12, 0, NULL, NULL) = 10

82137 recvfrom(19,  <unfinished ...>

82215 recvfrom(10,  <unfinished ...>

82137 <... recvfrom resumed>0x7fa590063b58, 12, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82215 <... recvfrom resumed>0x7f8950002572, 2, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82215 epoll\_ctl(7, EPOLL\_CTL\_ADD, 11, {events=0, data={u32=1342187488, u64=140227729434592}}) = 0

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82137 sendto(19, "\377\0\0\0\0\0\0\0\1\177", 10, 0, NULL, 0 <unfinished ...>

82215 recvfrom(11,  <unfinished ...>

82137 <... sendto resumed>)             = 10

82215 <... recvfrom resumed>"\377\0\0\0\0\0\0\0\1\177", 12, 0, NULL, NULL) = 10

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 19, {events=EPOLLIN, data={u32=2416325680, u64=140349062657072}} <unfinished ...>

82215 recvfrom(11,  <unfinished ...>

82137 <... epoll\_ctl resumed>)          = 0

82215 <... recvfrom resumed>0x7f8950002f62, 2, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82215 sendto(10, "\377\0\0\0\0\0\0\0\3\177\3", 11, 0, NULL, 0) = 11

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN, data={u32=1342183248, u64=140227729430352}} <unfinished ...>

82137 recvfrom(18,  <unfinished ...>

82215 <... epoll\_ctl resumed>)          = 0

82137 <... recvfrom resumed>"\377\0\0\0\0\0\0\0\3\177\3", 12, 0, NULL, NULL) = 11

82215 sendto(11, "\377\0\0\0\0\0\0\0\1\177\3", 11, 0, NULL, 0 <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 18, {events=EPOLLIN|EPOLLOUT, data={u32=2416322128, u64=140349062653520}} <unfinished ...>

82215 <... sendto resumed>)             = 11

82137 <... epoll\_ctl resumed>)          = 0

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}} <unfinished ...>

82137 recvfrom(18,  <unfinished ...>

82215 <... epoll\_ctl resumed>)          = 0

82137 <... recvfrom resumed>0x7fa590062d83, 53, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 sendto(18, "\3\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 54, 0, NULL, 0) = 54

82215 recvfrom(10,  <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 18, {events=EPOLLIN, data={u32=2416322128, u64=140349062653520}} <unfinished ...>

82215 <... recvfrom resumed>"\3\1", 2, 0, NULL, NULL) = 2

82137 <... epoll\_ctl resumed>)          = 0

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN|EPOLLOUT, data={u32=1342183248, u64=140227729430352}} <unfinished ...>

82137 recvfrom(19,  <unfinished ...>

82215 <... epoll\_ctl resumed>)          = 0

82137 <... recvfrom resumed>"\377\0\0\0\0\0\0\0\1\177\3", 12, 0, NULL, NULL) = 11

82215 recvfrom(10,  <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 19, {events=EPOLLIN|EPOLLOUT, data={u32=2416325680, u64=140349062657072}} <unfinished ...>

82215 <... recvfrom resumed>"NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 52, 0, NULL, NULL) = 52

82137 <... epoll\_ctl resumed>)          = 0

82215 recvfrom(10, 0x7f8950005048, 8192, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82137 recvfrom(19, 0x7fa590063b63, 53, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82215 sendto(10, "\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, 0) = 53

82137 recvfrom(18, "\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, NULL) = 53

82215 sendto(10, "\4+\5READY\vSocket-Type\0\0\0\6DEALER\10I"..., 45, 0, NULL, 0) = 45

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN, data={u32=1342183248, u64=140227729430352}} <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 18, {events=EPOLLIN|EPOLLOUT, data={u32=2416322128, u64=140349062653520}} <unfinished ...>

82215 <... epoll\_ctl resumed>)          = 0

82137 <... epoll\_ctl resumed>)          = 0

82137 recvfrom(18, "\4+\5READY\vSocket-Type\0\0\0\6DEALER\10I"..., 8192, 0, NULL, NULL) = 45

82137 sendto(19, "\3\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 54, 0, NULL, 0) = 54

82215 recvfrom(11,  <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 19, {events=EPOLLIN, data={u32=2416325680, u64=140349062657072}} <unfinished ...>

82215 <... recvfrom resumed>"\3\1", 2, 0, NULL, NULL) = 2

82137 <... epoll\_ctl resumed>)          = 0

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82215 recvfrom(11, "NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 52, 0, NULL, NULL) = 52

82215 recvfrom(11, 0x7f895000bd48, 8192, 0, NULL, NULL) = -1 EAGAIN (Resource temporarily unavailable)

82215 sendto(11, "\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, 0) = 53

82215 sendto(11, "\4\32\5READY\vSocket-Type\0\0\0\4PUSH", 28, 0, NULL, 0 <unfinished ...>

82137 sendto(18, "\4)\5READY\vSocket-Type\0\0\0\6ROUTER\10I"..., 43, 0, NULL, 0 <unfinished ...>

82215 <... sendto resumed>)             = 28

82137 <... sendto resumed>)             = 43

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82215 recvfrom(10,  <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 18, {events=EPOLLIN, data={u32=2416322128, u64=140349062653520}} <unfinished ...>

82215 <... recvfrom resumed>"\4)\5READY\vSocket-Type\0\0\0\6ROUTER\10I"..., 8192, 0, NULL, NULL) = 43

82137 <... epoll\_ctl resumed>)          = 0

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN|EPOLLOUT, data={u32=1342183248, u64=140227729430352}} <unfinished ...>

82137 recvfrom(19,  <unfinished ...>

82215 <... epoll\_ctl resumed>)          = 0

82137 <... recvfrom resumed>"\1NULL\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..., 53, 0, NULL, NULL) = 53

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN, data={u32=1342183248, u64=140227729430352}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 19, {events=EPOLLIN|EPOLLOUT, data={u32=2416325680, u64=140349062657072}}) = 0

82137 recvfrom(19, "\4\32\5READY\vSocket-Type\0\0\0\4PUSH", 8192, 0, NULL, NULL) = 28

82137 sendto(19, "\4\32\5READY\vSocket-Type\0\0\0\4PULL", 28, 0, NULL, 0) = 28

82215 recvfrom(11, "\4\32\5READY\vSocket-Type\0\0\0\4PULL", 8192, 0, NULL, NULL) = 28

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}} <unfinished ...>

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 19, {events=EPOLLIN, data={u32=2416325680, u64=140349062657072}} <unfinished ...>

82215 <... epoll\_ctl resumed>)          = 0

82137 <... epoll\_ctl resumed>)          = 0

82215 sendto(11, "\0\fHEARTBEAT 64", 14, 0, NULL, 0) = 14

82137 recvfrom(19, "\0\fHEARTBEAT 64", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82215 sendto(11, "\0\fHEARTBEAT 64", 14, 0, NULL, 0) = 14

82137 recvfrom(19, "\0\fHEARTBEAT 64", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 18, {events=EPOLLIN|EPOLLOUT, data={u32=2416322128, u64=140349062653520}}) = 0

82137 sendto(18, "\1\0\0\1Y", 5, 0, NULL, 0) = 5

82215 recvfrom(10, "\1\0\0\1Y", 8192, 0, NULL, NULL) = 5

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 18, {events=EPOLLIN, data={u32=2416322128, u64=140349062653520}}) = 0

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN|EPOLLOUT, data={u32=1342183248, u64=140227729430352}}) = 0

82215 sendto(10, "\1\0\0\24Ok:64: 'Y' not found", 24, 0, NULL, 0) = 24

82137 recvfrom(18, "\1\0\0\24Ok:64: 'Y' not found", 8192, 0, NULL, NULL) = 24

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN, data={u32=1342183248, u64=140227729430352}}) = 0

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82215 sendto(11, "\0\fHEARTBEAT 64", 14, 0, NULL, 0) = 14

82137 recvfrom(19, "\0\fHEARTBEAT 64", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82215 sendto(11, "\0\fHEARTBEAT 64", 14, 0, NULL, 0) = 14

82137 recvfrom(19, "\0\fHEARTBEAT 64", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82215 sendto(11, "\0\fHEARTBEAT 64", 14, 0, NULL, 0) = 14

82137 recvfrom(19, "\0\fHEARTBEAT 64", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 18, {events=EPOLLIN|EPOLLOUT, data={u32=2416322128, u64=140349062653520}}) = 0

82137 sendto(18, "\1\0\0\3Y 2", 7, 0, NULL, 0) = 7

82215 recvfrom(10, "\1\0\0\3Y 2", 8192, 0, NULL, NULL) = 7

82137 epoll\_ctl(7, EPOLL\_CTL\_MOD, 18, {events=EPOLLIN, data={u32=2416322128, u64=140349062653520}}) = 0

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN|EPOLLOUT, data={u32=1342183248, u64=140227729430352}}) = 0

82215 sendto(10, "\1\0\0\5Ok:64", 9, 0, NULL, 0) = 9

82137 recvfrom(18, "\1\0\0\5Ok:64", 8192, 0, NULL, NULL) = 9

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 10, {events=EPOLLIN, data={u32=1342183248, u64=140227729430352}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82215 sendto(11, "\0\fHEARTBEAT 64", 14, 0, NULL, 0) = 14

82137 recvfrom(19, "\0\fHEARTBEAT 64", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82215 sendto(11, "\0\fHEARTBEAT 64", 14, 0, NULL, 0) = 14

82137 recvfrom(19, "\0\fHEARTBEAT 64", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82215 sendto(11, "\0\fHEARTBEAT 64", 14, 0, NULL, 0) = 14

82137 recvfrom(19, "\0\fHEARTBEAT 64", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82215 sendto(11, "\0\fHEARTBEAT 64", 14, 0, NULL, 0) = 14

82137 recvfrom(19, "\0\fHEARTBEAT 64", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82137 recvfrom(12, "\0\fHEARTBEAT 10", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=1342187488, u64=140227729434592}}) = 0

82215 sendto(11, "\0\fHEARTBEAT 64", 14, 0, NULL, 0) = 14

82137 recvfrom(19, "\0\fHEARTBEAT 64", 8192, 0, NULL, NULL) = 14

82215 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=1342187488, u64=140227729434592}}) = 0

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN|EPOLLOUT, data={u32=939534304, u64=140197262010336}}) = 0

82157 sendto(11, "\0\fHEARTBEAT 12", 14, 0, NULL, 0) = 14

82137 recvfrom(17, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82157 epoll\_ctl(7, EPOLL\_CTL\_MOD, 11, {events=EPOLLIN, data={u32=939534304, u64=140197262010336}}) = 0

82137 recvfrom(14, "\0\fHEARTBEAT 12", 8192, 0, NULL, NULL) = 14

82135 --- SIGINT {si\_signo=SIGINT, si\_code=SI\_KERNEL} ---

82212 --- SIGINT {si\_signo=SIGINT, si\_code=SI\_KERNEL} ---

82155 --- SIGINT {si\_signo=SIGINT, si\_code=SI\_KERNEL} ---

82216 +++ killed by SIGINT +++

82215 +++ killed by SIGINT +++

82138 +++ killed by SIGINT +++

82137 +++ killed by SIGINT +++

82158 +++ killed by SIGINT +++

82156 +++ killed by SIGINT +++

82214 +++ killed by SIGINT +++

82212 +++ killed by SIGINT +++

82157 +++ killed by SIGINT +++

82155 +++ killed by SIGINT +++

82136 +++ killed by SIGINT +++

82135 +++ killed by SIGINT +++

**Вывод**