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: heartbeat 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

```
#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 <functional>

bool inputAvailable();
std::time_t t_now();

enum com : char {
    None = 0,
    Create = 1,
    Ping = 2,
    ExecAdd = 3,
    ExecFnd = 4,
    ExecErr = 5,
    HeartBeat = 6  // Новый тип сообщения для heartbeat
};

class message {
public:
    message() {}
    message(com command, int id, int num)
        : command(command), id(id), num(num), sent_time(t_now()) {}
    message(com command, int id, int num, char s[])
        : command(command), id(id), num(num), sent_time(t_now()) {
        strncpy(st, s, 30);
    }

    bool operator==(const message &other) const {
        return command == other.command && id == other.id && num == other.num;
    }

    com command;
    int id;
    int num;
    std::time_t sent_time;
    char st[30];
};

class Node {
public:
    int id;
```

```
    pid_t pid;
    void* context;
    void* socket;
    std::string address;

    Node* left = nullptr;  // Левый потомок (меньшие id)
    Node* right = nullptr; // Правый потомок (большие id)

    int beat_counter = 0;
};

Node createNode(int id, bool is_child);
Node createProcess(int id);
Node* insertChild(Node* root, Node* newChild);
void send_mes(Node &node, message m);
message get_mes(Node &node);
void traverseChildren(Node* root, const std::function<void(Node&)>& f);
Node* searchChild(Node* root, int id);

// Функции для heartbeat
void handle_heartbeat_command(Node* children_root);
void check_beats();
void update_beat(int node_id);
```

lib.cpp

```
#include "lib.h"
#include <fcntl.h>   // Для fcntl, O_NONBLOCK
#include <errno.h>
#include <signal.h>
#include <chrono>

// Функция проверки ввода
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());
}
```

```
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);

   node.address = "tcp://127.0.0.1:" + std::to_string(5555 + id);

   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)
   {
      // Дочерний процесс
      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);
}

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)
      return message(None, -1, -1);
```

```
    message m;
    std::memcpy(&m, zmq_msg_data(&request), sizeof(message));
    return m;
}

void traverseChildren(Node* root, const std::function<void(Node&)>& f) {
    if (!root)
        return;
    traverseChildren(root->left, f);
    f(*root);
    traverseChildren(root->right, f);
}

Node* insertChild(Node* root, Node* newChild) {
    if (root == nullptr)
        return newChild;
    if (newChild->id < root->id)
        root->left = insertChild(root->left, newChild);
    else if (newChild->id > root->id)
        root->right = insertChild(root->right, newChild);
    return root;
}

// Внутренние статические переменные для heartbeat
static std::chrono::milliseconds heartbeat_interval(0);
static std::map<int, std::chrono::steady_clock::time_point> beat_tracker;

// Вспомогательная функция, возвращающая текущее время (steady_clock)
std::chrono::steady_clock::time_point now() {
    return std::chrono::steady_clock::now();
}

// Обновление времени последнего heartbeat для узла
void update_beat(int node_id) {
    beat_tracker[node_id] = now();
}

// Функция обработки команды heartbeat:
// Читает новый интервал с консоли, сохраняет его и рассылает сообщение  детям.
void handle_heartbeat_command(Node* children_root) {
    int time;
    std::cin >> time;
    heartbeat_interval = std::chrono::milliseconds(time);
    // Сброс времени для всех узлов
    for (auto& kv : beat_tracker) {
        kv.second = now();
    }
    // Рассылка нового интервала детям
    message msg(HeartBeat, -1, time);
    traverseChildren(children_root, [&](Node& child) {
        send_mes(child, msg);
    });
}

// Функция проверки полученных heartbeat от узлов.
```

```
// Сообщения о недоступности выводятся не чаще, чем раз в 5 секунд.
void check_beats() {
    static auto last_print_time = now();
    auto current = now();
    auto diff = std::chrono::duration_cast<std::chrono::milliseconds>(current - last_print_time).count();
    if (diff < 5000) {
        return;
    }
    last_print_time = current;
    for (auto& kv : beat_tracker) {
        auto elapsed = std::chrono::duration_cast<std::chrono::milliseconds>(current - kv.second).count();
        if (elapsed > 4 * heartbeat_interval.count()) {
            std::cout << "Node: " << kv.first << " is unavailable now" << std::endl;
            kv.second = current; // Сброс, чтобы избежать повторного вывода до следующего периода
        }
    }
}

Node* searchChild(Node* root, int id) {
    if (root == nullptr)
        return nullptr;
    if (root->id == id)
        return root;
    Node* found = searchChild(root->left, id);
    if (found)
        return found;
    return searchChild(root->right, id);
}
```

control.cpp

```
#include "lib.h"
#include <cstdio>      // для fgets, sscanf
#include <fcntl.h>     // для fcntl, O_NONBLOCK
#include <unistd.h>
#include <errno.h>
#include <string>

Node* children_root = nullptr;

int main()
{
    std::unordered_set<int> all_id;
    // Управляющий узел имеет id -1
    all_id.insert(-1);
    std::list<message> saved_mes;
    std::string command;

    while (true)
    {
        // Обрабатываем входящие сообщения от прямых детей
```

```
traverseChildren(children_root, [&](Node& child) {
    message m = get_mes(child);
    switch (m.command)
    {
    case Create:
        all_id.insert(m.num);
        std::cout << "Ok: " << m.num << std::endl;
        for (auto it = saved_mes.begin(); it != saved_mes.end(); ++it)
        {
            if (it->command == Create && it->num == m.num)
            {
                saved_mes.erase(it);
                break;
            }
        }
        break;
    case Ping:
        std::cout << "Ok: " << m.id << " is available" << std::endl;
        for (auto it = saved_mes.begin(); it != saved_mes.end(); ++it)
        {
            if (it->command == Ping && it->id == m.id)
            {
                saved_mes.erase(it);
                break;
            }
        }
        break;
    case ExecErr:
        std::cout << "Ok: " << m.id << " '" << m.st << "' not found" << std::endl;
        for (auto it = saved_mes.begin(); it != saved_mes.end(); ++it)
        {
            if (it->command == ExecFnd && it->id == m.id)
            {
                saved_mes.erase(it);
                break;
            }
        }
        break;
    case ExecAdd:
        std::cout << "Ok: " << m.id << std::endl;
        for (auto it = saved_mes.begin(); it != saved_mes.end(); ++it)
        {
            if (it->command == ExecAdd && it->id == m.id)
            {
                saved_mes.erase(it);
                break;
            }
        }
        break;
    case ExecFnd:
        std::cout << "Ok: " << m.id << " '" << m.st << "' " << m.num << std::endl;
        for (auto it = saved_mes.begin(); it != saved_mes.end(); ++it)
        {
            if (it->command == ExecFnd && it->id == m.id)
            {
```

```
                saved_mes.erase(it);
                break;
            }
        }
        break;
    case HeartBeat:
        update_beat(m.id);
        break;
    default:
        break;
    }
});

// Проверяем недошедшие сообщения
for (auto it = saved_mes.begin(); it != saved_mes.end(); ++it)
{
    if (std::difftime(t_now(), it->sent_time) > 5)
    {
        switch (it->command)
        {
        case Ping:
            std::cout << "Error:" << it->id << " is unavailable" << std::endl;
            break;
        case Create:
            std::cout << "Error: Parent " << it->id << " is unavailable" << std::endl;
            break;
        case ExecAdd:
        case ExecFnd:
            std::cout << "Error: Node " << it->id << " is unavailable" << std::endl;
            break;
        default:
            break;
        }
        saved_mes.erase(it);
        break;
    }
}

// Проверяем heartbeat – сообщения о недоступности выводятся не чаще, чем раз в 5 секунд
check_beats();

// Обрабатываем команды ввода
if (!inputAvailable())
    continue;
std::cin >> command;
if (command == "create")
{
    // Читаем оставшуюся часть строки
    char input_line[100];
    fgets(input_line, sizeof(input_line), stdin);
    int child_id, parent_id = -1;
    int count = sscanf(input_line, "%d %d", &child_id, &parent_id);
    if (count < 1)
    {
        std::cout << "Error: Missing child id" << std::endl;
```

```
            continue;
        }
        if (count == 1 || (count == 2 && parent_id == -1))
        {
            if (all_id.count(child_id))
            {
                std::cout << "Error: Node with id " << child_id << " already exists" << std::endl;
            }
            else
            {
                Node child = createProcess(child_id);
                Node* childPtr = new Node(child);
                children_root = insertChild(children_root, childPtr);
                all_id.insert(child_id);
                std::cout << "Ok: " << child.pid << std::endl;
            }
        }
        else
        {
            message m(Create, parent_id, child_id);
            saved_mes.push_back(m);
            Node* parentNode = searchChild(children_root, parent_id);
            if (parentNode)
                send_mes(*parentNode, m);
            else
                std::cout << "Error: Parent with id " << parent_id << " not found" << std::endl;
        }
    }
    else if (command == "exec")
    {
        char input_line[100];
        fgets(input_line, sizeof(input_line), stdin);
        int id, val;
        char key[30];
        if (sscanf(input_line, "%d %30s %d", &id, key, &val) == 3)
        {
            if (!all_id.count(id))
            {
                std::cout << "Error: Node with id " << id << " doesn't exist" << std::endl;
                continue;
            }
            message m = {ExecAdd, id, val, key};
            saved_mes.push_back(m);
            Node* target = searchChild(children_root, id);
            if (target)
                send_mes(*target, m);
            else
                std::cout << "Error: Node with id " << id << " is not directly accessible" << std::endl;
        }
        else if (sscanf(input_line, "%d %30s", &id, key) == 2)
        {
            if (!all_id.count(id))
            {
                std::cout << "Error: Node with id " << id << " doesn't exist" << std::endl;
                continue;
```

```
            }
            message m = {ExecFnd, id, -1, key};
            saved_mes.push_back(m);
            Node* target = searchChild(children_root, id);
            if (target)
                send_mes(*target, m);
            else
                std::cout << "Error: Node with id " << id << " is not directly accessible" << std::endl;
        }
    }
    else if (command == "ping")
    {
        int id;
        std::cin >> id;
        if (!all_id.count(id))
        {
            std::cout << "Error: Node with id " << id << " doesn't exist" << std::endl;
        }
        else
        {
            message m(Ping, id, 0);
            saved_mes.push_back(m);
            Node* target = searchChild(children_root, id);
            if (target)
                send_mes(*target, m);
            else
                std::cout << "Error: Node with id " << id << " is not directly accessible" << std::endl;
        }
    }
    else if (command == "heartbeat")
    {
        // Устанавливаем новый интервал heartbeat и рассылаем его всем прямым детям
        handle_heartbeat_command(children_root);
    }
    else
        std::cout << "Error: Command doesn't exist!" << std::endl;
    usleep(1000000);
  }
  return 0;
}
```

computing.cpp

```
#include "lib.h"
#include <iostream>
#include <map>
#include <chrono>
#include <thread>
#include <functional>

using namespace std::chrono;
```

```
Node* children_root = nullptr;

int main(int argc, char *argv[])
{
    // Создаем узел текущего процесса
    Node I = createNode(atoi(argv[1]), true);
    std::map<std::string, int> dict;

    // Переменные для heartbeat
    std::chrono::milliseconds local_heartbeat(0);
    auto last_beat = steady_clock::now();

    while (true)
    {
        // Периодическая отправка heartbeat-сообщения родительскому узлу
        if (local_heartbeat > std::chrono::milliseconds::zero() &&
            duration_cast<milliseconds>(steady_clock::now() - last_beat).count() > local_heartbeat.count())
        {
            message hb(HeartBeat, I.id, -1);
            send_mes(I, hb);
            last_beat = steady_clock::now();
        }

        // Обрабатываем сообщения от дочерних узлов
        traverseChildren(children_root, [&](Node& child) {
            message m = get_mes(child);
            if (m.command != None)
                send_mes(I, m);
        });

        // Проверяем сообщение от родителя
        message m = get_mes(I);
        switch (m.command)
        {
        case Create:
            if (m.id == I.id)
            {
                Node child = createProcess(m.num);
                Node* childPtr = new Node(child);
                children_root = insertChild(children_root, childPtr);
                send_mes(I, {Create, child.id, child.pid});
            }
            else
                traverseChildren(children_root, [&](Node& child) {
                    send_mes(child, m);
                });
            break;
        case Ping:
            if (m.id == I.id)
                send_mes(I, m);
            else
                traverseChildren(children_root, [&](Node& child) {
                    send_mes(child, m);
                });
```

```
            break;
        case ExecAdd:
            if (m.id == I.id)
            {
                dict[std::string(m.st)] = m.num;
                send_mes(I, m);
            }
            else
                traverseChildren(children_root, [&](Node& child) {
                    send_mes(child, m);
                });
            break;
        case ExecFnd:
            if (m.id == I.id)
            {
                if (dict.find(std::string(m.st)) != dict.end())
                    send_mes(I, {ExecFnd, I.id, dict[std::string(m.st)], m.st});
                else
                    send_mes(I, {ExecErr, I.id, -1, m.st});
            }
            else
                traverseChildren(children_root, [&](Node& child) {
                    send_mes(child, m);
                });
            break;
        case HeartBeat:
            // Обновляем локальный интервал heartbeat при получении команды от управляющего узла
            local_heartbeat = std::chrono::milliseconds(m.num);
            break;
        default:
            break;
        }
        usleep(1000000);
    }
    return 0;
}
```

Протокол работы программы

Тестирование:

```
exec 10 X
Error: Node with id 10 doesn't exist
create 8
Ok: 102478
exec 8 X
Ok: 8 'X' not found
exec 8 X 16
Ok: 8
exec 8 X
Ok: 8 'X' 16
create 11
```

Ok: 102552
create 6
Ok: 102569
create 13
Ok: 102603
create 9
Ok: 102645
exec 11 X 11
Ok: 11
exec 6 X 6
Ok: 6
exec 9 X 18
ping 11
Ok: 11 is available
ping 21
Error: Node with id 21 doesn't exist

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\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\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\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\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\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\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\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\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\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\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\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\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\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\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\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"..., 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\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\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\1\177", 10, 0, NULL, 0) = 10
82157 sendto(10, "\377\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\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\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\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\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\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\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\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\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\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\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\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\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 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\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\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\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\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\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\3\177\3", 12, 0, NULL, NULL) = 11
82215 sendto(11, "\377\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"..., 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\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"..., 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"..., 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"..., 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"..., 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\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\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\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