Министерство образования и науки Российской Федерации

Югорский государственный университет

Институт цифровой экономики

Отчет о лабораторной работе №2

По дисциплине «Разработка системных приложений»

Выполнил: студент группы 1162б

Стародубов Юрий

Проверил: Шицелов А.В.

Ханты-Мансийск

2019

Цель работы: изучить как программа может заменить свой код на код из другого исполняемого файла.

Задание 1, 3

Напишите программу, создающую дочерний процесс для исполнения команды. Напишите оболочку – программу, ожидающую ввод команды и выполняющая ее в дочернем процессе.

void lab3() {

int returnStatus;

cout << "Enter command\n";

char par[1000];

char asd[1000];

memset(par, 0, sizeof(par));

memset(asd, 0, sizeof(asd));

string str;

getline(cin, str);

stringstream ss(str);

string token;

string d;

//vector<string> str1;

string str1 = "";

char delim = ' ';

getline(ss, d, delim);

strcpy(par, d.c\_str());

int \_size = 0;

while (getline(ss, token, delim)) {

//str1.push\_back(token);

str1 += token + " ";

\_size++;

}

if (str1.size() != 0)

str1.resize(str1.size() - 1);

strcpy(asd, str1.c\_str());

char \*mas[\_size + 2];

char \*p = strtok(asd, " ");

mas[0] = par;

int ind = 1;

while (p != NULL)

{

mas[ind++] = p;

p = strtok(NULL, " ");

}

mas[\_size + 1] = NULL;

pid\_t PID = fork();

int ex = 0;

if (PID > 0) cout << PID << endl;

if (PID == 0) {

cout << "Parent: " << getppid() << " Me: " << getpid()<<endl;

ex = execvp(par, mas);

cout << "EXEC Error: " << ex << "No such command" << endl;

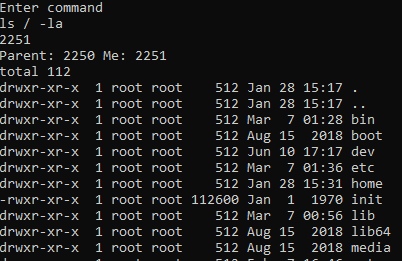
}

waitpid(PID, &returnStatus, 0);

cout << "Press ENTER or any text to exit" << endl;

exit(1);

}



Скриншот 1 – вывод команды ls/ -la

Задание 2. Напишите программу, которая создает 2 дочерних процесса и ждет их завершения и по-разному реагирует на их завершение. Выводите сообщения о ходе выполнения программы и ошибках. При завершении дочернего вызывайте exit

void lab2() {

int returnStatus;

pid\_t PID = fork();

if (PID == 0) {

printf("I am process %d, My parent is %d\n", getpid(), getppid());

exit(0);

}

else {

pid\_t PID2 = fork();

if (PID2 == 0) {

printf("I am process %d, My parent is %d\n", getpid(), getppid());

exit(0);

}

else {

waitpid(PID2, &returnStatus, 0);

cout << "end2" << endl;

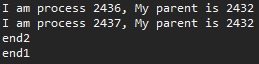
}

waitpid(PID, &returnStatus, 0);

cout << "end1" << endl;

}

}



Скриншот 2 – сообщение о ходе выполнения программы

Вывод: было изучено как программа может заменить свой код на код из другого исполняемого файла.