|  |  |
| --- | --- |
| [Герб МГТУ.png](https://wiki.bmstu.ru/zav/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:%D0%93%D0%B5%D1%80%D0%B1_%D0%9C%D0%93%D0%A2%D0%A3.png) | **Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)» (МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Информационная безопасность (ИУ8)

**Безопасность Систем Баз Данных**

Отчет

по Лабораторной работе №3

**“Получение потоков запускаемого приложения (shell)”**

**Выполнил:**

Евула А. С.,

студент группы ИУ8-63

**Проверил:**

Зенькович С. А.,

старший преподаватель

кафедры ИУ8

Москва, 2021

# Оглавление

[Цель работы 3](#_Toc75095801)

[Основная часть 3](#_Toc75095802)

[1. Теоретическая часть 3](#_Toc75095803)

[2. Практическая часть 3](#_Toc75095804)

[Запускаемое приложение 3](#_Toc75095805)

[Запускающее приложение 4](#_Toc75095806)

[Выводы 4](#_Toc75095807)

[Приложение А 5](#_Toc75095808)

[Приложение Б 5](#_Toc75095809)

# Цель работы

Необходимо реализовать POSIX-совместимый shell-скрипт для получения потоков запускаемого приложения.

# Основная часть

## Теоретическая часть

POSIX (Portable Operating System Interface) - "Портативный Интерфейс Операционной Системы". Набор стандартов, описывающих интерфейсы между ОС и прикладной программой, библиотеку языка C и набор приложений и их интерфейсов. Необходим для обеспечения совместимости UNIX-ОС и переносимости прикладных программ.

POSIX нужен для облегчения переноса кодов на иные платформы; унификации интерфейсов; определения минимума интерфейсов программ; развития стандартов в направлениях коммуникационные сети; обработки данных, защиты информации.

## Практическая часть

### Запускаемое приложение

Создадим скрипт hello с помощью команды nano hello.sh

При наличии аргументов скрипт выведет “hello, <аргумент>, иначе – “hello” и ошибку “name not provided”.

Также выдадим права на него - chmod 755 hello.sh

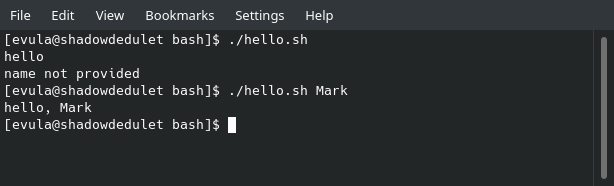


Рисунок - использование скрипта hello.sh с разными аргументами

Исходный код в Приложение А.

### Запускающее приложение

Создадим скрипт open с помощью команды nano open.sh, который будет запускать приложения. Поток вывода приложения переходит в stdout.log, ошибок переходит в stderr.log, код завершения – в return.log.

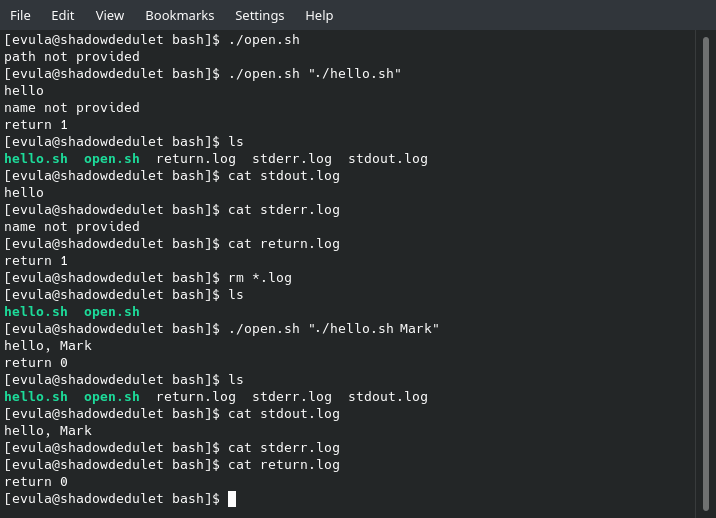


Рисунок - использование скрипта open.sh с различными аргументами

Исходный код в

Приложение Б.

# Выводы

Были получены знания о стандартах POSIX, получении потоков приложения, написании и использовании shell скриптов.

# Приложение А

#!/bin/bash  
# greeting script  
  
  
if [ $# -gt 0 ]; then  
 echo "hello, $1";  
 exit 0;  
fi  
  
echo "hello";  
echo "name not provided" 1>&2;  
exit 1;

# Приложение Б

#!/bin/bash  
# open script  
  
  
if [ $# -gt 0 ]; then  
 app\_path=$1;  
 $app\_path 1>stdout.log 2>stderr.log;  
 echo return $?>return.log;  
 cat stdout.log stderr.log return.log;  
 exit 0;  
fi  
echo "path not provided">&2;  
exit 1;