**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова»**

**(БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  | ДОПУСКАЕТСЯ К ЗАЩИТЕ: | | | | | | | | | | | | | | |
| Факультет | И | |  | Заведующий кафедрой | | | | | |  | | | | И9 | | | | | | |
|  | индекс факультета | |  |  | | | | |  | | | | | | индекс кафедры | | |
| Выпускающая кафедра | И9 | |  | | Матвеев С.А. | | |  | | | |  | | | | | | | |
|  | индекс кафедры | |  | | Фамилия ИО | |  | | | | подпись | | | | | | | | | | |
| Группа | И9М33 | |  | «\_\_\_\_\_» | |  | | | | | | | | | | 2018 г. | | |
|  | индекс группы | |  |  | |  | | | | | | |  | | | |

**отчет**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **о прохождении** | | | | научно-производственной | | | | | | | | | | | | | | **практики** | | | | |
| наименование практики | | | | | | | | | | | | | | | | | | | | |
| Крылова Кирилла Андреевича | | | | | | | | | | | | | | | | | | | | |
| Фамилия, имя, отчество обучающегося | | | | | | | | | | | | | | | | | | | | |
| **Обучающегося по**  **направлению** | | | 09.04.04 | | | | |  | | Программная инженерия | | | | | | | | | | | |
|  | | | код | | | | | | | | |  | полное наименование направления | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
| **Руководитель практики:** | | | | | | | | Снижко Е.А., к.п.н., доцент, доцент каф. И9 | | | | | | | | | | | | |
|  | | | Фамилия ИО, ученая степень, ученое звание, должность | | | | | | | | | | | | | | | | |
| **Срок прохождения практики:** | | | | | с | 02.07.2018 | | | | | | | г. |  | по | 22.07.2018 | | | г. |
| **Должность обучающегося на практике:** | | | | | | | | | | магистрант | | | | | | | | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Руководитель практики:** | | | |  | |  | |
|  | |  | | Снижко Е.А. | |  | |
| Подпись | |  | | Фамилия ИО | |  | |
| «\_\_\_» | \_\_\_\_\_\_\_\_\_\_\_ |  | 2018 г. | |  | |

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc524790707)

[1 СВЕДЕНИЯ О ПРЕДПРИЯТИИ 4](#_Toc524790708)

[1.1 Административно-финансовая деятельность предприятия 4](#_Toc524790709)

[1.2 Эксплуатация вычислительной техники 4](#_Toc524790710)

[1.3 Программное и аппаратное обеспчение технологических процессов 5](#_Toc524790711)

[1.4 Организация патентных исследований 5](#_Toc524790712)

[1.5 Методы анализа технического уровня изучаемого аппаратного программного обеспечения 5](#_Toc524790713)

[1.6 Методы тестирования, верификации и валидации программных продуктов 6](#_Toc524790714)

[2 ЗАДАЧИ ПРАКТИКИ 7](#_Toc524790715)

[3 Обоснование средств разработки 8](#_Toc524790716)

[4 РЕЗУЛЬТАТЫ РАБОТЫ 11](#_Toc524790717)

[4.1 Серверная часть системы 11](#_Toc524790718)

[4.1.1 Предоставление удалённого доступа к файлам 11](#_Toc524790719)

[4.1.2 Загрузка и хранение пакетов с необходимыми зависимостями для различных архитектур 13](#_Toc524790720)

[4.1.3 Создание и хранение файлов конфигурации клиента 15](#_Toc524790721)

[4.2 Клиентская часть системы 17](#_Toc524790722)

[4.2.1 Первичная настройка 17](#_Toc524790723)

[4.2.2 Использование файла конфигурации клиента для установки требуемого установленного программного обеспечения 17](#_Toc524790724)

[ЗАКЛЮЧЕНИЕ 24](#_Toc524790725)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 25](#_Toc524790726)

# **ВВЕДЕНИЕ**

В качестве места прохождения практики был выбран Балтийский Государственный Технический Университет «ВОЕНМЕХ» им. Д.Ф. Устинова, кафедра И9 БГТУ «ВОЕНМЕХ».

БГТУ «ВОЕНМЕХ» является ярким представителем инженерной школы России, сумевшим сохранить и приумножить достижения отечественного и мирового инженерно-технического образования.

За свою 75-летнюю историю вуз подготовил для оборонной промышленности, народно-хозяйственного комплекса страны более 60 000 первоклассных специалистов, многие из которых стоят сегодня руководят предприятиями, фирмами, конструкторскими бюро, а также возглавляют научные коллективы.

Целью практики является создание системы автоматизированной установки программного обеспечения на компьютеры в аудиториях кафедры И9.

1. **СВЕДЕНИЯ О ПРЕДПРИЯТИИ**

## 1.1 Административно-финансовая деятельность предприятия

Административно-финансовая деятельность предприятия предоставляется в следующих документах, которые можно найти на официальном сайте БГТУ «ВОЕНМЕХ» [1], а именно:

1. Устав БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова [2];
2. Правила внутреннего распорядка [3];
3. Положение об оплате труда работников Университета [4];
4. Положение о надбавках и доплатах стимулирующего характера к должностным окладам [5];
5. Положение о премировании (установлении поощрительных выплат) [6];
6. Справка о результатах исполнения программы развития БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова на 2010-2020 г.г. по итогам 2014 года [7];
7. Положение о закупках товаров, выполнения работ, оказания услуг для нужд федерального государственного бюджетного образовательного учреждения высшего профессионального образования «Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова» [8];
8. Лицензия на право ведения образовательной деятельности [9];
9. Свидетельство о государственной аккредитации [10].

## 1.2 Эксплуатация вычислительной техники

Нормы эксплуатации вычислительной техники формируются на основе СанПиН 2.2.2/2.3.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» [11]. Для возможности эксплуатации вычислительной техники сотрудникам ВУЗа необходимо пройти инструктаж и занести в протокол факт ознакомления.

## 1.3 Программное и аппаратное обеспчение технологических процессов

Ввиду того, что БГТУ «ВОЕНМЕХ» имеет большое множество различных кафедр для реализации программ обучения, детально систематизировать программное и аппаратное обеспечение не представляется возможным. Среди аппаратного обеспечения можно выделить:

1. ЭВМ/ПЭВМ,
2. вычислительные кластеры,
3. демонстрационные стенды,
4. измерительная аппаратура,
5. периферийные устройства.

Среди программного обеспечения можно выделить: ПО общего назначения, Прикладное ПО специального (профессионального) назначения.

## 1.4 Организация патентных исследований

Для обучающихся в БГТУ «ВОЕНМЕХ» функционирует Центр научного и технического творчества студентов, который способствует развитию студентов в научной сфере, доводятся идеи обучающихся до состояния патентов. Также на базе университета существует патентная база для упрощенной регистрации патентов, реализованных в БГТУ «ВОЕНМЕХ».

## 1.5 Методы анализа технического уровня изучаемого аппаратного программного обеспечения

Технический уровень информационных систем и их компонентов является удовлетворительным, они удовлетворяют техническим условием и стандартам. Однако нужно отметить общий низкий уровень информационной коммуникации внутри ВУЗа. Несмотря на наличие ЕКС, электронный документооборот либо не внедрен, либо внедрен частично. Также стоит отметить, что сайт БГТУ «ВОЕНМЕХ» не полностью отвечаем всем требованиям, которые выставляются к сайтам ВУЗов, но работа над новой версией сайта в данный момент ведется отделом УИТ.

## 1.6 Методы тестирования, верификации и валидации программных продуктов

Среди методов тестирования, верификации и валидации программных продуктов применяется: экспертиза; мониторинг (профилирование); модульное тестирование; интеграционное тестирование; системное тестирование; тестирование на отказ; нагрузочное тестирование; тестирование в предельных режимах.

1. **ЗАДАЧИ ПРАКТИКИ**

На компьютерах, установленных в аудиториях кафедры И9, необходимо массово устанавливать или обновлять требуемое для учебного процесса программное обеспечение. На данный момент уходит очень много времени, так как этот процесс выполняется вручную для каждого рабочего места.

Задачей практики было обеспечение автоматизированной установки программного обеспечения на компьютеры в аудиториях кафедры И9. Автоматизированная система должна отвечать следующим требованиям:

1. поддержка дистрибутивов для различных архитектур операционных систем Ubuntu, Kubuntu;
2. поддержка конфигураций для множества аудиторий с различающимися наборами программ;
3. минимальное использование интернет-трафика – однажды загруженный дистрибутив должен впоследствии использоваться, не требуя повторной загрузки из интернета.
4. **Обоснование средств разработки**

Пригодная для работы пользователя система состоит из множества программ и утилит. В Linux каждый компонент системы представлен в виде пакета. Все операции, связанные с изменением состава системы: установка, удаление, проверка, обновление компонентов, — производятся над пакетами. В целом, пакет — это средство сделать так, чтобы пользователь-администратор, изменяя или обновляя программное наполнение системы, работал не с файлами (имена которых ему подчас неизвестны), а с определёнными функциональностями самой системы.

Нет жёсткого требования, чтобы один пакет содержал только одну программу. В пакет естественно объединять такие ресурсы, с которыми удобно работать как с одним целым. Это может быть отдельная программа или набор утилит (например, coreutils, основные утилиты, унаследованные Linux от UNIX) или модуль с дополнительными возможностями программы, или общие для нескольких программ ресурсы. В процессе развития и/или устаревания программного обеспечения выделение некоторых задач в отдельный пакет может приобретать или терять смысл, поэтому способ объединения ресурсов в пакеты — это не что-то раз и навсегда выбранное: пакеты могут разделяться и сливаться.

Чтобы сделать процедуру установки, удаления и обновления компонента системы атомарной, были разработаны менеджеры пакетов. Менеджер пакетов — это программа, которая вычисляет весь комплекс операций над отдельными пакетами, который нужно произвести для установки/удаления нового компонента (пакета), и сама запускает установщик пакетов сколько нужно раз с нужными параметрами. Кроме того, менеджер пакетов хранит информацию не только о пакетах, уже установленных в системе, но и обо всех, которые доступны для установки с какого-либо носителя или по Сети.

Наиболее широко используемые менеджер пакетов называется APT (Advanced Package Tool). Первоначально он был разработан в рамках дистрибутива Debian и работал только с установщиком пакетов dpkg, впоследствии для других дистрибутивов была разработана версия, работающая с rpm. Так как он уже предустановлен в операционных системах Ubuntu и Kubuntu, а также обладает всеми необходимыми для решения поставленной задачи возможностями, было принято решение использовать его.

В Linux нет отдельного объекта под именем «система», однако существует множество различных компонентов взаимодействующих друг с другом. Главный из системных компонентов — пользователь. Для предоставления пользователю возможности обращаться к ресурсам системы, существует специальная программа, с которой пользователь ведёт диалог, а она уже переводит запросы пользователя в последовательность системных и библиотечных вызовов. Такая программа называется интерпретатором командной строки или командная оболочка («shell»). «Оболочкой» она названа потому, что всё управление системой идёт «изнутри» неё: пользователь общается с нею на удобном ему языке (с помощью текстовой командной строки), а она общается с другими частями системы на удобном им языке (вызывая запрограммированные функции).

Это происходит следующим образом: пользователь вводит с терминала строку, shell считывает её, иногда — преобразует по определённым правилам, получившуюся строку разбивает на команду и параметры, а затем выполняет команду, передавая ей эти параметры. Команда, в свою очередь, анализирует параметры, выделяет среди них ключи, и делает что попросили, попутно выводя на терминал данные для пользователя, после чего завершается. Так продолжается до тех пор, пока пользователь не скомандует оболочке завершиться самой.

Конечно, командных интерпретаторов в Linux несколько. Самый простой из них, появившийся в ранних версиях UNIX, назывался sh, или «Bourne Shell» — по имени автора, Стивена Борна (Stephen Bourne). Со временем его — везде, где только можно — заменили на более мощный, bash, «Bourne Again Shell».

bash превосходит sh во всём, особенно в возможностях редактирования командной строки. Также bash может выполнять команды, которые описаны в текстовом файле, что позволяет создавать скрипты, которые автоматически исполняются, исключая участие пользователя, что подходит для автоматизации процессов. bash предустановлен на операционных системах Ubuntu и Kubuntu, что подходит установленные требования [12].

Для тестирования получившейся системы в окружении, приближённом к тому, в котором система будет исполняться в конечном итоге, было принято решение воспользоваться возможностями виртуализации. Виртуальная машина – это программа, которая эмулирует реальный (физический) компьютер со всеми его компонентами (жёсткий диск, привод, BIOS, сетевые адаптеры и т.д.). На такой виртуальный компьютер можно установить, например, операционную систему, драйверы, программы и т.д. Таким образом, можно запустить на своем реальном компьютере еще несколько виртуальных компьютеров, с такой же или другой операционной системой. Виртуальную машину используют для различных целей и задач:

1. установка второй/другой операционной системы;
2. тестирование программного обеспечения;
3. безопасный запуск подозрительных программ;
4. эмуляция компьютерной сети;
5. запуск приложений, которые нельзя запустить из установленной операционной системы.

В качестве программы, обеспечивающей виртуализацию, было принято решение использовать «Oracle VM VirtualBox», так как это бесплатная программа с открытым исходным кодом, в полной мере подходящая под условия поставленной задачи.

1. **РЕЗУЛЬТАТЫ РАБОТЫ**

В результате проделанной работы был создан набор скриптов, предназначенных для исполнения целевых компьютерах. Получившаяся система организована по принципу клиент-сервер. Часть системы, на стороне сервера ответственна за выполнение следующих задач:

1. предоставление удалённого доступа к файлам;
2. загрузка и хранение пакетов с необходимыми зависимостями для различных архитектур;
3. создание и хранение файлов конфигурации клиента.

Часть системы, расположенная на стороне клиента выполняет следующие задачи:

1. загрузка и хранение файла конфигурации клиента;
2. периодическая проверка актуальности файла конфигурации клиента, путём сравнения с последней версией на сервере;
3. использование файла конфигурации клиента для установки требуемого установленного программного обеспечения.

Описание принципов работы системы и процесса установки окружения будет приведено в том же порядке, в котором эти пункты представлены выше.

## Серверная часть системы

В этом разделе будет описана серверная часть системы, которая будет установлена на одном из компьютеров, имеющим доступ к сети интернет и к локальной сети, в которой расположены клиенты этой системы.

## Предоставление удалённого доступа к файлам

Необходимо определить способ, посредством которого можно предоставить клиентам возможность скачивать файлы, которые будут централизовано храниться на сервере. В результате поиска, было найдено два возможных решения:

1. использовать файловый сервер;
2. использовать scp (Secure Copy).

Так как в основе scp лежат те же принципы, что и в ssh, для её использования потребуется пароль, чтобы получить доступ к удалённым ресурсам. Также, для того, чтобы поддержать возможность использования apt-get, клиенту потребуется загружать файлы локально, а только затем применять apt-get.

С другой стороны, файловый сервер предоставляет возможность открытого чтения и загрузки удалённых ресурсов, что хорошо подходит для решения поставленной задачи. К тому же, на компьютерах клиентов не будет необходимости производить сложные манипуляции по настройке доступа к ресурсам.

Существует два сервера, которые подходят под поставленные задачи, а также имеют широкое распространение: apache и nginx. Оба сервера предоставляют схожий набор функций, однако установка и настройка apache занимает меньше времени. Это и является фактором, согласно которому для организации удалённого доступа к файлам будет использован сервер apache [13].

Для установки apache необходимо воспользоваться apt-get, а затем запустить сервис сервера apache:

sudo apt-get install apache2

sudo service apache2 start

Также, если данный файловый сервер будет использоваться только в пределах локальной сети, необходимо установить ограничение доступа с прочих IP адресов, так как сервер подключен к сети интернет. Для этого в файле “etc/apache2/apache2.conf” необходимо изменить раздел “/var/www” следующим образом:

<Directory /var/www/>

Require host localhost

Require ip 127.0.0.1

Require ip 192.168

Require ip 10

</Directory>

Затем, для применения изменений необходимо перезапустить сервис сервера:

sudo service apache2 restart

В директории “/var/www/” впоследствии будут храниться все необходимые для функционирования системы пакеты.

## Загрузка и хранение пакетов с необходимыми зависимостями для различных архитектур

Для того, чтобы пользователь успешно установил программное обеспечение необходимо предоставить не только пакет, содержащий необходимую программу, но и все зависимости, которые требует этот пакет. Для скачивания пакета с зависимостями был написан bash-скрипт “mydebs-download.bash”, который скачивает необходимый пакет указанной архитектуры, а затем, используя возможности утилиты apt-cache, получает список зависимостей для данного пакета и скачивает их[14].

#!/bin/bash

package=$1

arch=$2

if [[ -z $package ]]; then

echo "Error: no package given!"

exit 0

fi

if [[ ! -z $arch ]]; then

arch=":$arch"

fi

package="$package$arch"

echo "Downloading package $package with dependencies"

sudo apt-get download $package

packages\_list=$(apt-cache depends --recurse --no-recommends --no-suggests --no-conflicts --no-breaks --no-replaces --no-enhances --no-pre-depends $package | grep "^\w" | sort -u)

for i in $packages\_list; do

sudo apt-get download $i 2>>errors.txt

done

Скачав все требуемые пакеты, а также их зависимости, необходимо предоставить клиенту доступ к скачиванию пакетов. С целью повторного использования установленных на клиенте программ, загрузка пакетов клиентом с сервера будет осуществляться наиболее распространённым путём – с помощью утилиты apt-get. Однако для того, чтобы поддержать возможность скачивания наших пакетов с помощью apt-get будет необходимо создать на сервере локальный репозиторий.

Следует упомянуть, что для создания репозитория необходимо создать персональный ключ шифрования, которым будет подписан репозиторий. Процесс подписи производится с использованием ассиметричного алгоритма шифрования, что значит, что в результате будет создана пара ключей – закрытый и открытый. Используя закрытый ключ, администратор будет создавать репозиторий, а открытый ключ будет использоваться клиентом, для подтверждения того, что репозиторий был создан администратором, с помощью парного закрытого ключа [15].

На стороне сервера для создания ключа необходимо воспользоваться утилитой “gpg”:

gpg --gen-key

В результате выполнения всех предъявляемых требований, будет получен открытый ключ, который будет идентифицировать администратора. Впоследствии данный ключ будет необходимо сохранить на всех клиентах.

Для автоматизации создания репозитория был написан bash-скрипт “mydebs-repo-create.bash”, который создаёт требуемые для организации полноценного репозитория файлы:

#! /bin/bash

DST=${1}

if [[ -z $DST ]]; then

echo "Usage: mydebs-repo-create [dest folder]"

exit 0

fi

cd $DST

dpkg-scanpackages . /dev/null > Packages

gzip -9c Packages > Packages.gz

# Generate the Release file

cat conf/distributions > Release

# The Date: field has the same format as the Debian package changelog entries,

# that is, RFC 2822 with time zone +0000

echo -e "Date: `LANG=C date -Ru`" >> Release

# Release must contain MD5 sums of all repository files (in a simple repo just the Packages and Packages.gz files)

echo -e 'MD5Sum:' >> Release

printf ' '$(md5sum Packages.gz | cut --delimiter=' ' --fields=1)' %16d Packages.gz' $(wc --bytes Packages.gz | cut --delimiter=' ' --fields=1) >> Release

printf '\n '$(md5sum Packages | cut --delimiter=' ' --fields=1)' %16d Packages' $(wc --bytes Packages | cut --delimiter=' ' --fields=1) >> Release

# Release must contain SHA256 sums of all repository files (in a simple repo just the Packages and Packages.gz files)

echo -e '\nSHA256:' >> Release

printf ' '$(sha256sum Packages.gz | cut --delimiter=' ' --fields=1)' %16d Packages.gz' $(wc --bytes Packages.gz | cut --delimiter=' ' --fields=1) >> Release

printf '\n '$(sha256sum Packages | cut --delimiter=' ' --fields=1)' %16d Packages' $(wc --bytes Packages | cut --delimiter=' ' --fields=1) >> Release

gpg --armor --detach-sign --output Release.gpg Release

При его выполнении будет предъявлено требование ввести закрытый ключ администратора для подписи созданного репозитория.

## Создание и хранение файлов конфигурации клиента

На данном этапе пользователю доступны вне необходимые пакеты, однако для их установки требуется вручную на каждом клиенте для каждого требуемого пакета выполнить команду вида:

sudo apt-get install <имя пакета>

Для автоматизации процесса установки были созданы специальные файлы конфигурации клиента, с внутренним представлением вида:

configuration ID:room101

version:1.1

packages:python;apache2

Где поле “configuration ID” является идентификатором аудитории, либо некоторого подмножества компьютеров, на которых должна быть установлена одинаковый набор пакетов. Поле “version” содержит версию конфигурационного файла - если клиент, при обращении на сервер, находит новую версию конфигурационного файла, то он откатывает текущую конфигурацию, скачивает новую и применяет её. Поле “packages” содержит список пакетов, которые должны быть установлены при применении конфигурации.

Для создания файлов конфигурации был написан bash-скрипт “mydebs-ver-create.bash”:

#!/bin/bash

configId=$1

version=$2

packages=$3

if [[ -z $configId || -z $version || -z $packages ]]; then

echo "Usage: mydebs-version-file [configId] [version] [packages]"

exit 0

fi

filename=mydebs-ver-file-$configId

echo -e "configuration ID:$configId" > $filename

echo -e "version:$version" >> $filename

echo -e "packages:$packages" >> $filename

echo "Version file $filename created."

После выполнения скрипта создастся файл с названием вида “mydebs-ver-file-room101”, который должен находиться на файловом сервере, так как клиенты, использующие конфигурацию, например “room101”, будут периодически искать этот файл на сервере, и проверять актуальность его версии.

## Клиентская часть системы

В этом разделе будет описана клиентская часть системы, которая будет установлена на компьютерах в аудиториях, имеющим доступ к локальной сети через которую доступна серверная часть системы, описанная в предыдущем разделе.

## Первичная настройка

Для того чтобы клиент смог загружать пакеты посредством утилиты apt-get, необходимо произвести несколько действий. Во-первых, следует сохранить открытый ключ администратора, полученный при выполнении пункта 4.1.2. Для этого можно выполнить команду:

sudo apt-key add <имя файла с открытым ключом>

После выполнения этой команды клиент сможет скачивать любые пакеты, подписанные ключом администратора.

Во-вторых, необходимо изменить файл “/etc/apt/sources.list”, добавив в его начало строку:

deb http://<адрес сервера в локальной сети>/<путь до файла на сервере, относительно раздела www> /

После этого необходимо вызвать команду:

sudo apt-get update

Теперь клиент готов скачивать пакеты с файлового сервера.

## Использование файла конфигурации клиента для установки требуемого установленного программного обеспечения

Для применения файлов конфигурации клиента был написан bash-скрипт “ mydebs-ver-apply.bash”:

#!/bin/bash

function readVersionFile()

{

echo "$(cut --d=':' --f=2 $1)"

}

function printParams()

{

local filename=$1

shift

local params=(${@})

local configId=${params[0]}

local version=${params[1]}

local packages=$(echo ${params[2]} | tr ';' '\n')

echo "====== Contents of file $filename"

echo "configuration ID:$configId"

echo "version:$version"

echo "packages:"

for i in ${packages[@]}

do

echo " $i"

done

echo "======"

}

function revertVersion()

{

local packages=${@}

for i in ${packages[@]}

do

echo "Removing $i"

apt-get remove $i

done

echo "Removing unused dependencies"

apt-get autoremove

}

function applyVersion()

{

local packages=${@}

for i in ${packages[@]}

do

echo "Downloading package $i"

apt-get install $i

done

}

filename=$1

if [[ -z $filename ]]; then

echo "Usage: mydebs-apply-version [version file]"

exit 0

fi

params=($(readVersionFile $filename))

configId=${params[0]}

version=${params[1]}

packages=$(echo ${params[2]} | tr ';' '\n')

printParams $filename "${params[@]}"

# Resolve environment variable

envVarName=MYDEBS\_VERSION\_PATH

curVerPath=$(env | grep $envVarName | cut --d='=' --f=2)

if [[ -z $curVerPath ]]

then

curVerPath="/etc/mydebs-current-version"

echo "Can't find envivorment variable for version file. Adding envivorment variable $envVarName=$curVerPath"

envFilename="/etc/environment"

if [[ -z $(cat $envFilename | grep "$envVarName") ]]

then

echo "Adding environment variable to $envFile"

echo "$envVarName=$curVerPath" >> $envFilename

else

echo "WARNING: file $envFilename already contains variable $envVarName! If you see this error after restart, something gone wrong - please tell your administrator."

fi

fi

echo "Current version path: $curVerPath"

# Is version file exists?

if [[ ! -f "$curVerPath" ]]

then

while true; do

read -p "There is no current version file. You want to set file $filename as your current version file? (y/n) " yn

case $yn in

[Yy]\* ) echo "File $filename set as current version file."

echo "Applying new version: $filename"

applyVersion $packages

cp $filename $curVerPath

echo "Done."

exit 0;;

[Nn]\* ) echo "Stopping version apply procedure because you have no current version file."

exit 0;;

\* ) echo "Please answer yes or no.";;

esac

done

# Version file exists

else

echo "Version file already exists:"

curParams=($(readVersionFile "$curVerPath"))

printParams $curVerPath ${curParams[@]}

curVersion=${curParams[1]}

if [[ "$version" == "$curVersion" ]]

then

echo "Current version is actual, aborting."

exit 0

else

versions=($(echo "$version $curVersion" | sort -V -r | tr ';' '\n'))

highestVersion=${versions[0]}

if [[ "$version" == "$highestVersion" ]]

then

while true; do

read -p "Version of $filename is higher: $version (current $curVersion). Update needed - proceed? (y/n) " yn

case $yn in

[Yy]\* ) break;;

[Nn]\* ) echo "Aborting."

exit 0;;

\* ) echo "Please answer yes or no.";;

esac

done

fi

while true; do

read -p "You want to revert $curVerPath and apply $filename as your current version file? (y/n) " yn

case $yn in

[Yy]\* ) echo "Reverting version: $curVerPath:"

revertVersion $packages

echo "Done."

echo "Applying new version: $filename"

curPackages=$(echo ${Params[2]} | tr ';' '\n')

applyVersion $newPackages

cp $filename $curVerPath

break;;

[Nn]\* ) echo "Aborting."

exit 0;;

\* ) echo "Please answer yes or no.";;

esac

done

fi

fi

Использование скрипта пользователем, в случае применения новой конфигурации происходит следующим образом:

sudo ./mydebs-apply-version mydebs-version-file-room101

====== Contents of file mydebs-version-file-room101

configuration ID:room101

version:1.0

packages:

netcf

apache2

======

Current version path: /etc/mydebs-current-version

There is no current version file. You want to set file mydebs-version-file-room101 as your current version file? (y/n) y

File mydebs-version-file-room101 set as current version file.

Applying new version: mydebs-version-file-room101

Downloading package netcf

Reading package lists... Done

Building dependency tree

Reading state information... Done

The following additional packages will be installed:

augeas-lenses libaugeas0 libnetcf1

Suggested packages:

augeas-doc augeas-tools

The following NEW packages will be installed:

augeas-lenses libaugeas0 libnetcf1 netcf

0 upgraded, 4 newly installed, 0 to remove and 1 not upgraded.

Need to get 535 kB of archives.

After this operation, 2 526 kB of additional disk space will be used.

Do you want to continue? [Y/n] y

Get:1 http://10.10.10.3/debs augeas-lenses 1.10.1-2 [300 kB]

Get:2 http://10.10.10.3/debs libaugeas0 1.10.1-2 [172 kB]

Get:3 http://10.10.10.3/debs libnetcf1 1:0.2.8-1ubuntu2 [49,3 kB]

Get:4 http://10.10.10.3/debs netcf 1:0.2.8-1ubuntu2 [13,8 kB]

Fetched 535 kB in 0s (34,9 MB/s)

Selecting previously unselected package augeas-lenses.

(Reading database ... 180800 files and directories currently installed.)

Preparing to unpack .../augeas-lenses\_1.10.1-2\_all.deb ...

Unpacking augeas-lenses (1.10.1-2) ...

Selecting previously unselected package libaugeas0:i386.

Preparing to unpack .../libaugeas0\_1.10.1-2\_i386.deb ...

Unpacking libaugeas0:i386 (1.10.1-2) ...

Selecting previously unselected package libnetcf1:i386.

Preparing to unpack .../libnetcf1\_1%3a0.2.8-1ubuntu2\_i386.deb ...

Unpacking libnetcf1:i386 (1:0.2.8-1ubuntu2) ...

Selecting previously unselected package netcf.

Preparing to unpack .../netcf\_1%3a0.2.8-1ubuntu2\_i386.deb ...

Unpacking netcf (1:0.2.8-1ubuntu2) ...

Processing triggers for ureadahead (0.100.0-20) ...

Processing triggers for libc-bin (2.27-3ubuntu1) ...

Processing triggers for systemd (237-3ubuntu10.3) ...

Setting up augeas-lenses (1.10.1-2) ...

Processing triggers for man-db (2.8.3-2) ...

Setting up libaugeas0:i386 (1.10.1-2) ...

Setting up libnetcf1:i386 (1:0.2.8-1ubuntu2) ...

Setting up netcf (1:0.2.8-1ubuntu2) ...

Processing triggers for libc-bin (2.27-3ubuntu1) ...

Downloading package apache2

Reading package lists... Done

Building dependency tree

Reading state information... Done

apache2 is already the newest version (2.4.29-1ubuntu4.2).

0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.

Done.

В случае обновления конфигурации пользовательский диалог выглядит следующим образом:

sudo ./mydebs-apply-version mydebs-version-file-room101

====== Contents of file mydebs-version-file-room101

configuration ID:room101

version:1.1

packages:

netcf

apache2

======

Current version path: /etc/mydebs-current-version

Version file already exists:

====== Contents of file /etc/mydebs-current-version

configuration ID:room101

version:1.0

packages:

netcf

apache2

======

Version of mydebs-version-file-room101 is higher: 1.1 (current 1.0). Update needed - proceed? (y/n) y

You want to revert /etc/mydebs-current-version and apply mydebs-version-file-room101 as your current version file? (y/n) y

Reverting version: /etc/mydebs-current-version:

Removing netcf

Reading package lists... Done

Building dependency tree

Reading state information... Done

The following packages were automatically installed and are no longer required:

augeas-lenses libaugeas0 libnetcf1

Use 'sudo apt autoremove' to remove them.

The following packages will be REMOVED:

netcf

0 upgraded, 0 newly installed, 1 to remove and 1 not upgraded.

After this operation, 49,2 kB disk space will be freed.

Do you want to continue? [Y/n] y

(Reading database ... 180750 files and directories currently installed.)

Removing netcf (1:0.2.8-1ubuntu2) ...

Processing triggers for man-db (2.8.3-2) ...

Removing apache2

Reading package lists... Done

Building dependency tree

Reading state information... Done

Package 'apache2' is not installed, so not removed

The following packages were automatically installed and are no longer required:

augeas-lenses libaugeas0 libnetcf1

Use 'sudo apt autoremove' to remove them.

0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.

Removing unused dependencies

Reading package lists... Done

Building dependency tree

Reading state information... Done

The following packages will be REMOVED:

augeas-lenses libaugeas0 libnetcf1

0 upgraded, 0 newly installed, 3 to remove and 1 not upgraded.

After this operation, 2 477 kB disk space will be freed.

Do you want to continue? [Y/n] y

(Reading database ... 180741 files and directories currently installed.)

Removing libnetcf1:i386 (1:0.2.8-1ubuntu2) ...

Removing libaugeas0:i386 (1.10.1-2) ...

Removing augeas-lenses (1.10.1-2) ...

Processing triggers for libc-bin (2.27-3ubuntu1) ...

Done.

Applying new version: mydebs-version-file-room101

<продолжение вывода как в примере выше>

# **ЗАКЛЮЧЕНИЕ**

В ходе научно-производственной практики были получены знания об организации управления установкой программного обеспечения на операционных системах Ubuntu и Kubuntu, также были получены навыки написания скриптов на языке Bash.

В результате практики была создана система автоматизированной установки программного обеспечения на компьютеры в аудиториях кафедры И9. Данная система поддерживает дистрибутивы для различных архитектур для операционных систем Ubuntu и Kubuntu. За счёт использования файлов конфигурации клиента осуществляется поддержка конфигураций для множества аудиторий с различающимися наборами программ. Также было выполнено требование на минимальное использование интернет-трафика, так как всё необходимое программное обеспечение загружается из интернета на сервер администратором только один раз, а затем скачивается клиентами уже из локальной сети.

Цель и задачи научно-производственной практики можно считать выполненными.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Официальный сайт БГТУ «ВОЕНМЕХ» [Электронный ресурс]. URL: http://www.voenmeh.ru (дата обращения 20.07.2018).
2. Устав БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова [Электронный ресурс]. URL: http://voenmeh.ru/files/0/29.10.2015.pdf (дата обращения 20.07.2018).
3. Правила внутреннего распорядка [Электронный ресурс]. URL: http://voenmeh.ru/files/0/pravila\_vnutr\_rasporyadka\_2011.pdf (дата обращения 20.07.2018).
4. Положение об оплате труда работников Университета [Электронный ресурс]. URL: http://voenmeh.ru/files/0/Pologenie\_o\_oplate\_truda\_2013.pdf (дата обращения 20.07.2018).
5. Положение о надбавках и доплатах стимулирующего характера к должностным окладам [Электронный ресурс]. URL: http://voenmeh.ru/files/0/Pologenie\_o\_nadbavkah\_2013.pdf (дата обращения 20.07.2018).
6. Положение о премировании (установлении поощрительных выплат) [Электронный ресурс]. URL: http://voenmeh.ru/files/0/Pologenie\_o\_premiyah\_2013.pdf (дата обращения 21.07.2018).
7. Справка о результатах исполнения программы развития БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова на 2010-2020 г.г. по итогам 2014 года [Электронный ресурс]. URL: http://voenmeh.ru/files/0/Spravka\_programma\_razvitiya2015\_otchet.pdf (дата обращения 21.07.2018).
8. Положение о закупках товаров, выполнения работ, оказания услуг для нужд федерального государственного бюджетного образовательного учреждения высшего профессионального образования «Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова» [Электронный ресурс]. URL: http://voenmeh.ru/ files/0/Pologenie\_o\_zakupkah2014.pdf (дата обращения 21.07.2018).
9. Лицензия на право ведения образовательной деятельности [Электронный ресурс]. URL: http://voenmeh.ru/files/0/License\_2238\_2p.pdf (дата обращения 22.07.2018).
10. Свидетельство о государственной аккредитации [Электронный ресурс]. URL: http://voenmeh.ru/files/0/accred-0854\_12.12.2013.pdf (дата обращения 22.07.2018).
11. СанПиН 2.2.2/2.3.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» [Электронный ресурс]. URL: https://rg.ru/2003/06/21/134.html: (дата обращения 22.07.2018).
12. Newham C., Rosenblatt B. Learning the bash shell: Unix shell programming. – " O'Reilly Media, Inc.", 2005.
13. Apt offline repository [Электронный ресурс]. URL: https://help.ubuntu.com/community/AptGet/Offline/Repository (дата обращения 22.07.2018).
14. Download packages dependencies locally ubuntu [Электронный ресурс]. URL: https://www.ostechnix.com/download-packages-dependencies-locally-ubuntu/ (дата обращения 22.07.2018).
15. Creating a Trusted Local Repository from which Software Updates can be installed [Электронный ресурс]. URL: https://ubuntuforums.org/showthread.php?t=1090731 (дата обращения 22.07.2018).