

Федеральное агентство связи  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Сибирский государственный университет телекоммуникаций и  
информатики»  
(СибГУТИ)

Кафедра вычислительных систем

## Лабораторная работа №1

«Система виртуализации VitrualBox.

Запуск и конфигурирование узлов под управлением GNU/Linux»

Выполнил:

студент гр. МГ-211

\_\_\_\_\_ /Бурдуковский И.А./

подпись

Проверил:

Профессор

кафедры ВС

\_\_\_\_\_ /Мамойленко С.Н./

ОЦЕНКА, подпись

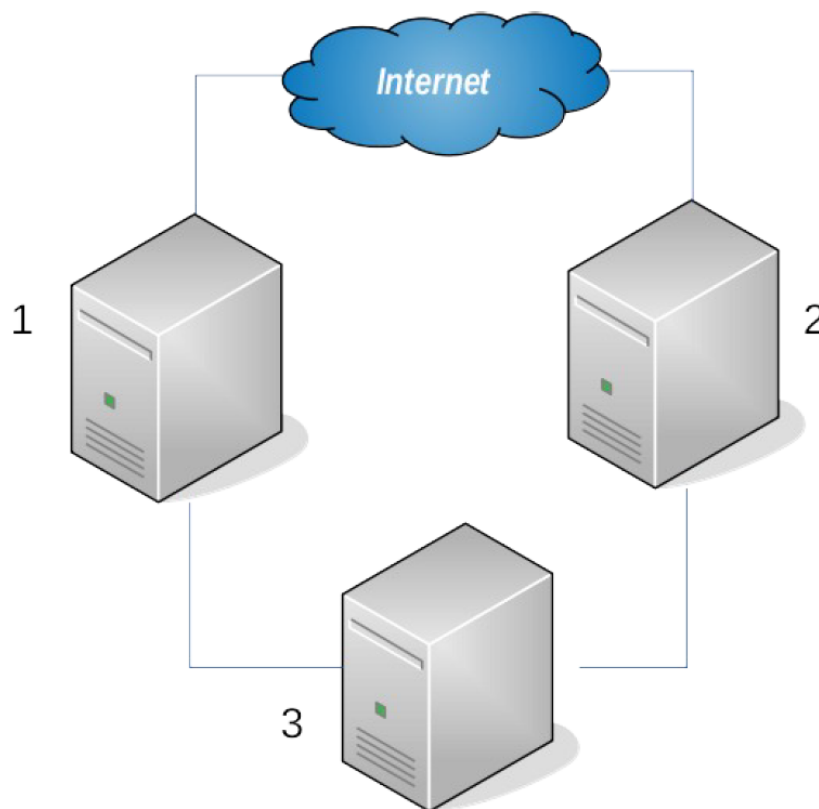
Новосибирск, 2023 г.

## Оглавление

Задание.....	3
Результаты работы .....	4

## ЗАДАНИЕ

1. Установите и сконфигурируйте среду виртуализации VirtualBox (версия не ниже 6.1).
2. Создайте базовый образ виртуальной машины, функционирующей под управлением ОС Debian (версия не ниже 11). Образ обязательно должен быть оснащен средствами интеграции с VirtualBox (VirtualBox Additions).
3. Разработайте скрипт автоматической конфигурации виртуального узла (для системы запуска SystemD) позволяющий при загрузке системы: устанавливать имя машины (статическое и в человеко-понятной форме), оформлять приветствие при регистрации в системе, содержащее IP адреса виртуальных машин, назначаемых ей системой виртуализации. Скрипт должен запускаться при загрузке системы.
4. Используя базовый образ сконфигурируйте инфраструктуру из трех узлов, как показано на Рисунке 1.



*Рисунок 1 - Сетевая конфигурация практического занятия 1*

## РЕЗУЛЬТАТЫ РАБОТЫ

Результатом первых двух заданий стала установленная на VirtualBox ОС Debian, оснащенная средствами интеграции с VirtualBox.

В результате выполнения третьего задания были разработаны скрипты, которые устанавливают при запуске ОС имя машины и приветствие при регистрации в системе. Скрипт приведен ниже.

Рисунок 1 – Скрипт автоматической конфигурации имени ОС

```
#!/usr/bin/env bash

set -x

NEW_HOSTNAME=$(VBoxControl --nologo guestproperty get /msn/HostName)
EXIT_CODE="$?"
if [[ "${EXIT_CODE}" != "0" ]]; then echo "# ERROR when get VBoxControl external hostname"; exit 1;
fi

NEW_HOSTNAME_PRETTY=$(VBoxControl --nologo guestproperty get /msn/HostNamePretty)
EXIT_CODE="$?"
if [[ "${EXIT_CODE}" != "0" ]]; then echo "# ERROR when get VBoxControl external hostname"; exit 1;
fi

NEW_HOSTNAME=${NEW_HOSTNAME##Value: }
NEW_HOSTNAME_PRETTY=${NEW_HOSTNAME_PRETTY##Value: }
CURRENT_HOSTNAME=$(hostnamectl --static)
CURRENT_HOSTNAME_PRETTY=$(hostnamectl --pretty)

if [[ ! -z "${NEW_HOSTNAME}" && "${CURRENT_HOSTNAME}" != "${NEW_HOSTNAME}" ]]; then
    hostnamectl --static set-hostname "${NEW_HOSTNAME}"
fi

if [[ ! -z "${NEW_HOSTNAME_PRETTY}" && "${CURRENT_HOSTNAME_PRETTY}" != "${NEW_HOSTNAME_PRETTY}" ]];
then
    hostnamectl --pretty set-hostname "${NEW_HOSTNAME_PRETTY}"
fi
```

Рисунок 2 – Скрипт установки MOTD

```
#!/usr/bin/env bash

MSG=""
#####
Hello! My hostname is \"$(hostname)\"

subnet 1-3: 10.0.1.0/24
1: 10.0.1.2
3: 10.0.1.1
subnet 2-3: 10.0.2.0/24
2: 10.0.2.2
3: 10.0.2.1
#####
"

echo "$MSG" > /etc/motd
```

Скрипт автоматической конфигурации ОС получает имя машины и ее IP от системы виртуализации с помощью средств интеграции VirtualBox Additions. Имя для машины устанавливалось вручную с помощью создание переменных /msn/HostName в VirtualBox. Эти переменные в последствии используются в скрипте.

Скрипт автоматической конфигурации ОС запускается при старте системы с помощью системы запуска SystemD. Сервисы SystemD для запуска скриптов приведены в рисунках 3 и 4.

Рисунок 3 – Сервис для запуска скрипта setup-node-name

```
/usr/lib/systemd/system/setup-node-name.service
[Unit]
Description=Service for hostname change from lab1
Documentation=
Requires=networking.service
Wants=network.target
After=network.target
Before=shutdown.target network-online.target
Conflicts=shutdown.target

[Service]
Type=oneshot
ExecStart=/bin/setup-node-name
RemainAfterExit=true
StandartOutput=journal

[Install]
WantedBy=multi-user.target
Alias=setup-node-name.service
```

Рисунок 4– Сервис для запуска скрипта setup-motd

```
/usr/lib/systemd/system/setup-motd.service
[Unit]
Description=Service for motd change from lab1
Documentation=
Requires=networking.service
Wants=network.target setup-node-name.service
After=network.target setup-node-name.service
Before=shutdown.target network-online.target
Conflicts=shutdown.target

[Service]
Type=oneshot
ExecStart=/root/lab1/setup-motd.sh
RemainAfterExit=true
StandartOutput=journal

[Install]
WantedBy=multi-user.target
Alias=setup-motd.service
```

Результат работы скрипта приведен на рисунке 5.

Рисунок 5. Результат работы скриптов

```
#####  
Hello! My hostname is "cluster-skylake01"  
  
subnet 1-3: 10.0.1.0/24  
  1: 10.0.1.2  
  3: 10.0.1.1  
subnet 2-3: 10.0.2.0/24  
  2: 10.0.2.2  
  3: 10.0.2.1  
#####  
root@cluster-skylake01:~/lab1# _
```

Для конфигурации инфраструктуры из четвертого задания были созданы три виртуальных машины на основе базового образа, настроенного ранее. Каждой виртуальной машине были добавлены по 2 сетевых адаптера. Для первой и второй машины типами подключения этих адаптеров являются «сеть NAT» и «внутренняя сеть», а для третьей машины оба адаптера имеют тип «внутренняя сеть».

На третьей виртуальной машине был настроен gateway, перенаправляющий все запросы с неизвестным IP-адресом первой виртуальной машине. На первой виртуальной машине с помощью iptables был настроен NAT, оборачивающий запросы от третьей виртуальной машины. Таблица правил NAT приведена на рисунке 6. Данные действия были необходимы для предоставления доступа третьей машине в интернет.

Рисунок 6. Конфигурация NAT на первом хосте

```
# Generated by iptables-save v1.8.7 on Fri Dec 23 13:08:13 2022  
*filter  
:INPUT ACCEPT [0:0]  
:FORWARD ACCEPT [0:0]  
:OUTPUT ACCEPT [0:0]  
-A FORWARD -j ACCEPT  
COMMIT  
# Completed on Fri Dec 23 13:08:13 2022  
# Generated by iptables-save v1.8.7 on Fri Dec 23 13:08:13 2022  
*nat  
:PREROUTING ACCEPT [0:0]  
:INPUT ACCEPT [0:0]  
:OUTPUT ACCEPT [0:0]  
:POSTROUTING ACCEPT [0:0]  
-A POSTROUTING -s 10.0.1.0/24 -j MASQUERADE  
COMMIT  
# Completed on Fri Dec 23 13:08:13 2022
```

Также третья машина была настроена в качестве маршрутизатора между сетями первой и третьей машин и второй и третьей машин. Для этого в файл /etc/sysctl.conf была добавлена строка net.ipv4.ip\_forward=1.