Министерство Науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский Томский политехнический Университет»



Инженерная школа информационных технологий и робототехники

Отделение информационных технологий

Направления подготовки: 09.04.01 «Информатика и вычислительная техника»

**Лабораторная работа №5**

**Проектирование и развертывание приложений с использованием платформы VK Cloud Solutions**по дисциплине:

**Облачные технологии**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Исполнитель:** |  | | | | |
| студент группы | 8ВМ22 |  | Ямкин Н.Н. |  |  |
|  |  |  |  |  |  |
| **Руководитель:** | . | | | | |
| к.т.н доцент ОИТ, ИШИТР |  |  | Ботыгин И.А. |  |  |
|  |  |  |  |  |  |

Томск – 2023

**Теоретическая часть**

Облачные технологии радикально изменили не только повседневную жизнь каждого из нас, но и кардинально трансформировали технологический бизнес-ландшафт. Если компания хочет стать лидером на рынке услуг, то ей не обойтись без современных облачных сервисов.

Использование облачных решений сегодня – ключевое преимущество, которое поможет отстроиться от конкурентов и опережать меняющиеся потребности бизнеса.

Бессерверные вычисления позволяют разработчикам приобретать бэкенд-сервисы с оплатой по мере использования, что означает, что разработчикам нужно платить только за те услуги, которые они используют. Это похоже на переход с тарифного плана мобильного оператора с фиксированным ежемесячным лимитом на тариф, где плата взимается только за каждый фактически использованный байт данных.

Термин «бессерверный» несколько вводит в заблуждение, поскольку все ещё существуют серверы, предоставляющие эти внутренние сервисы. Но все проблемы, связанные с серверным пространством и инфраструктурой, решаются поставщиком.

Понятие «веб-сервер» может относиться как к аппаратной начинке, так и к программному обеспечению. Или даже к обеим частям, работающим совместно.

С точки зрения "железа", «веб-сервер» — это компьютер, который хранит файлы сайта (HTML-документы, CSS-стили, JavaScript-файлы, картинки и другие) и доставляет их на устройство конечного пользователя (веб-браузер и т.д.). Он подключён к сети Интернет и может быть доступен через доменное имя, подобное mozilla.org.

С точки зрения ПО, веб-сервер включает в себя несколько компонентов, которые контролируют доступ веб-пользователей к размещённым на сервере файлам, как минимум — это HTTP-сервер. HTTP-сервер — это часть ПО, которая понимает URL-адреса (веб-адреса) и HTTP (протокол, который ваш браузер использует для просмотра веб-страниц).

Статический веб-сервер, или стек, состоит из компьютера ("железо") с сервером HTTP (ПО). Мы называем это «статикой», потому что сервер посылает размещённые файлы в браузер «как есть».

Динамический веб-сервер состоит из статического веб-сервера и дополнительного программного обеспечения, чаще всего сервера приложения и базы данных. Мы называем его «динамическим», потому что сервер приложений изменяет исходные файлы перед отправкой в ваш браузер по HTTP.

Виртуальная машина (часто сокращается до ВМ) не отличается от физических компьютеров, например ноутбука, смартфона или сервера. Она имеет ЦП, память, диски для хранения файлов и при необходимости может подключаться к Интернету. Хотя компоненты вашего компьютера (аппаратная часть) материальны, виртуальные машины часто рассматриваются как виртуальные компьютеры или программно-определяемые компьютеры в физических серверах, существуя только в виде кода.

Windows Server 2012 (кодовое имя «Windows Server 8») — версия серверной операционной системы от Microsoft. Принадлежит семейству ОС Microsoft Windows. Была выпущена 4 сентября 2012 года на смену Windows Server 2008 R2 как серверная версия Windows 8.

**Практическая часть**

В этой лабораторной работе будет создана виртуальная машина на ОС Ubuntu 22.04 на платформе VK Cloud Solution.

Заходим на главную страницу и выбираем Облачные вычисления => Виртуальные машины.

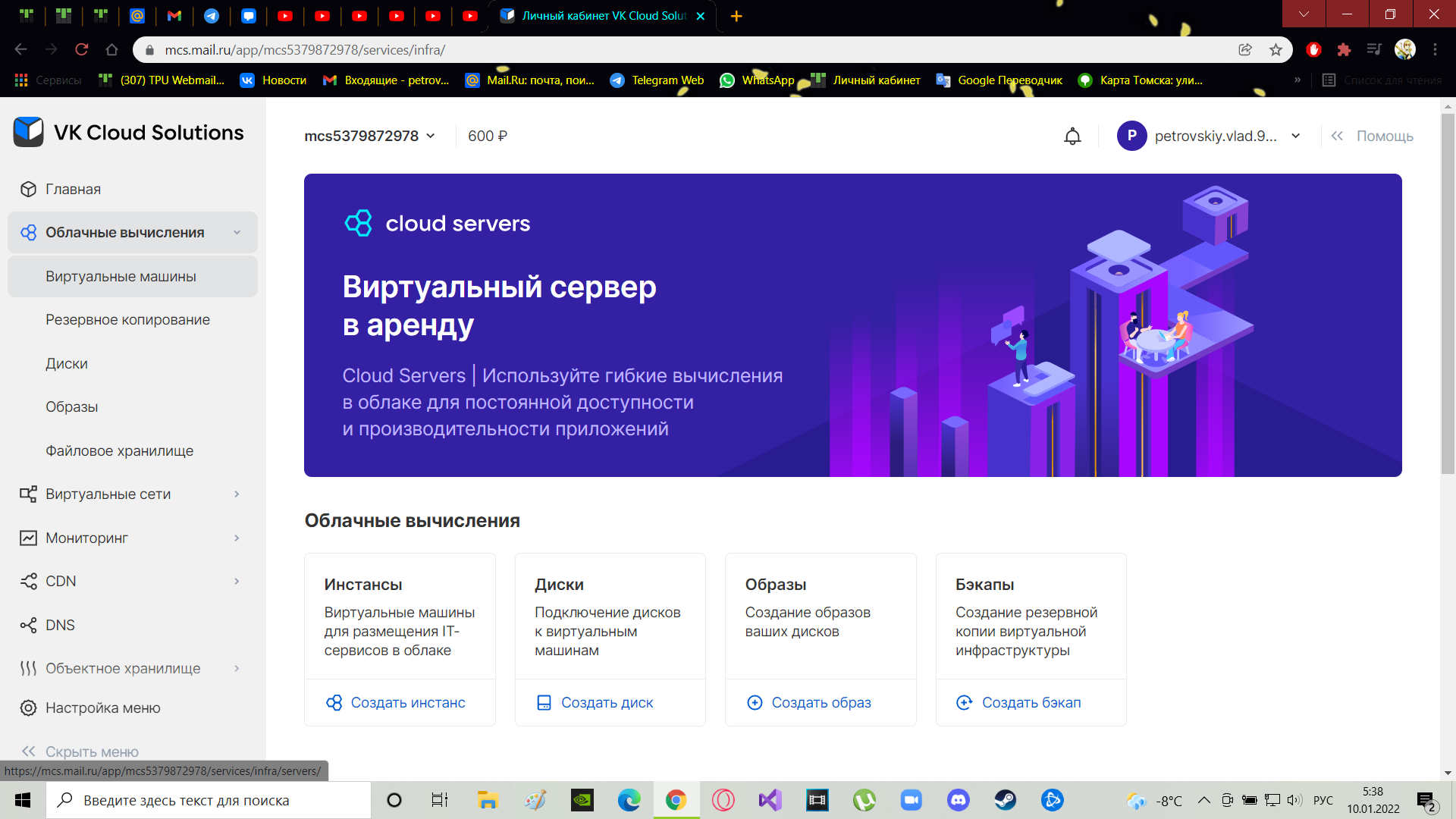


Рисунок 1 – Главный экран VK Cloud Solutions

Далее нажимаем кнопку Создать инстанс.

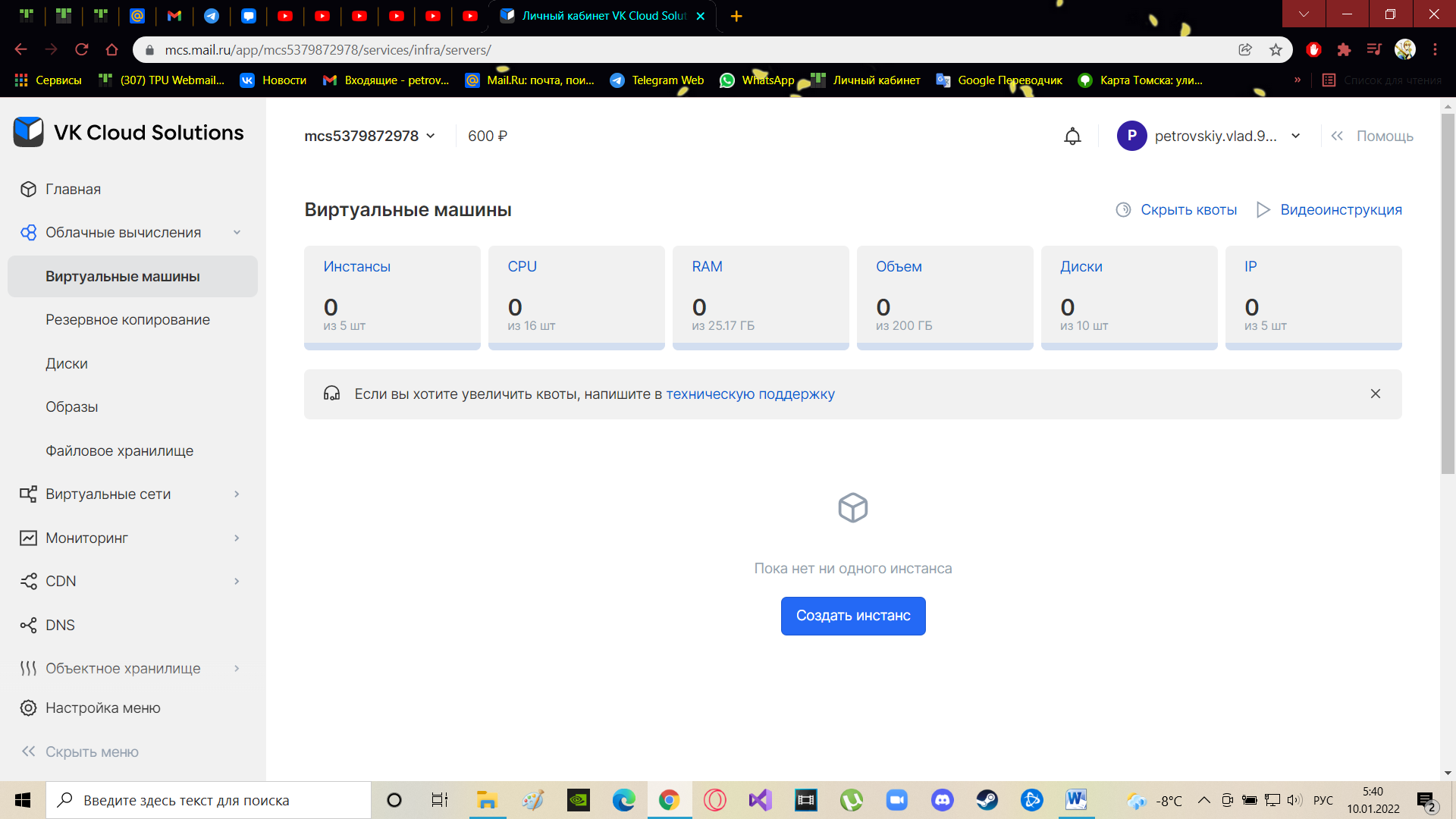


Рисунок 2 – Создание первой виртуальной машины

Здесь нужно ввести Имя и Тип виртуальной машины. Можно выбрать уже готовые сборки ВМ или сделать ее самостоятельно. Выбираем готовую Basic 1-2. Выбираем Зону доступности, Количество машин, размер диска (выбрал 10 ГБ) и тип диска.

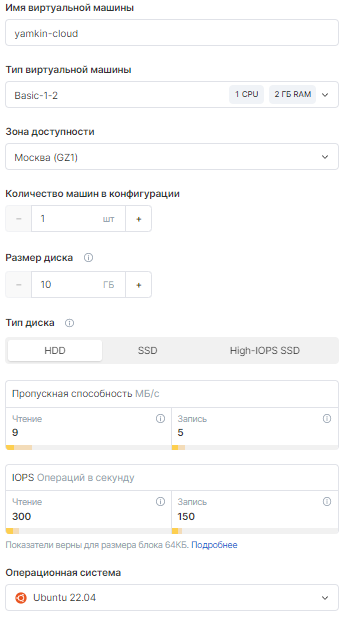


Рисунок 3 – Конфигурация ВМ

Далее выбираем операционную систему Ubuntu 22.04, и нажимаем Следующий шаг.

Здесь проводится настройка сети.

Выбираем внешнюю сеть (ext-net), генерируем SSH-ключи, оставляем настройки фаервола по умолчанию и нажимаем Следующий шаг.

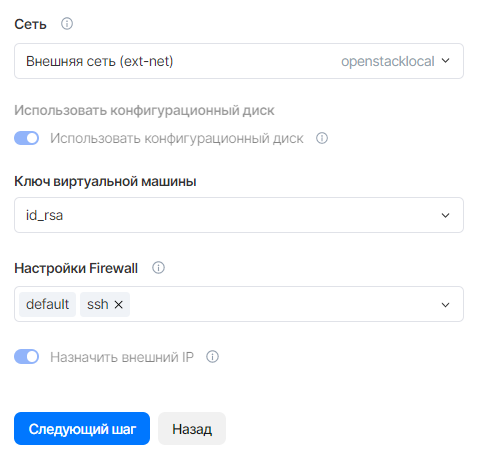


Рисунок 4 – Настройка сети

Отключаем резервное копирование системы, за ненадобностью.

Нажимаем Создать инстанс.

В появившемся окне видим, что виртуальная машина готова.

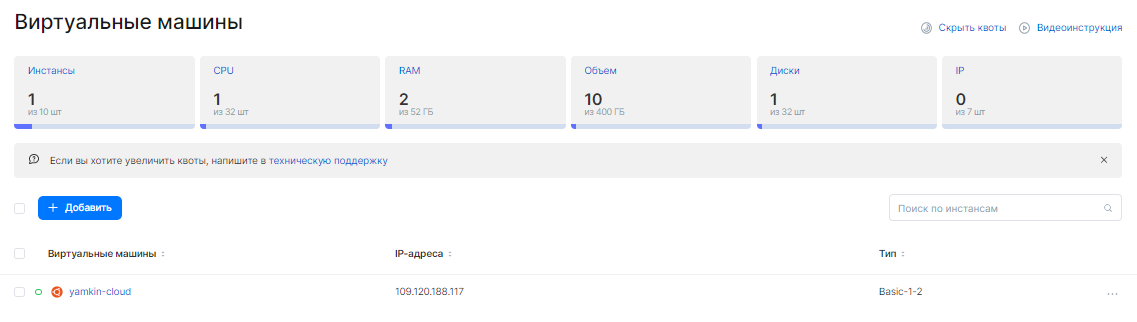


Рисунок 5 – ВМ готова

Далее подключимся к созданной машине через командную строку и произведем базовую конфигурацию Ubuntu 22.04.

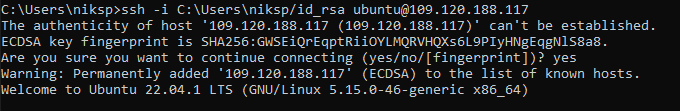


Рисунок 6 – Подключение к ВМ по SSH протоколу

Работать через командную строку неудобно, поэтому подключимся к этой же машине через VNС (Virtual Network Computing).

VNC — это широко распространенный метод удаленного доступа к рабочему столу компьютера по сети. Данные о нажатии клавиш и движении мыши, выполняемых пользователем на собственном компьютере передаются по сети на удаленный компьютер и воспринимаются им действия с его собственными клавиатурой и мышью. Информация с экрана удаленного компьютера выводится на экране компьютера пользователя. Работа по VNC через интернет с удаленным компьютером, находящимся в противоположной точке мира, для пользователя выглядит так, как будто этот компьютер находится непосредственно перед ним. Особенно VNC удобен при работе с графическим интерфейсом — с рабочим столом и программами для рабочего стола операционных систем Windows, Linux и других.

Перед началом выполнения каких-либо действий нужно обновить компоненты системы.

|  |
| --- |
| sudo apt update -y && sudo apt upgrade -y |

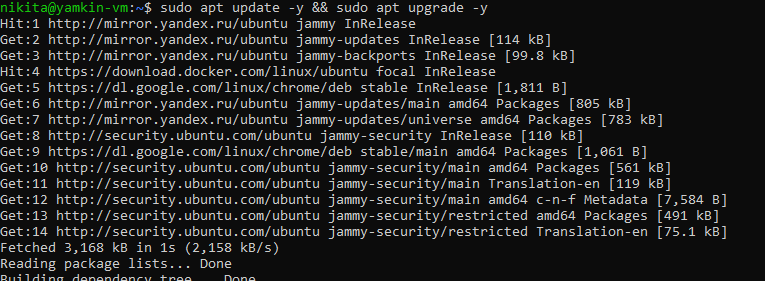


Рисунок 7 – Обновление компонентов системы

Управление системой через подключение VNC осуществляется посредством графической среды. За основу будет взято окружение XFCE, стабильно работающее в связке с TightVNCServer. Чтобы установить эту оболочку со всеми дополнениями, нужно ввести следующую команду в терминал:

|  |
| --- |
| sudo apt install xfce4 xfce4-goodies |

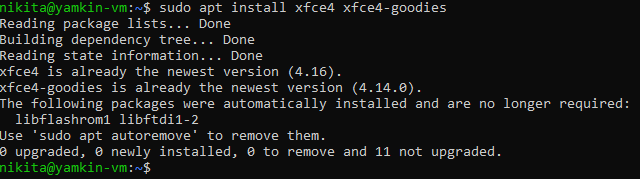


Рисунок 8 – Установка окружения XFCE

Инсталлировать пакет TightVNCServer вместе с зависимостями можно командой:

|  |
| --- |
| sudo apt install tightvncserver -y |

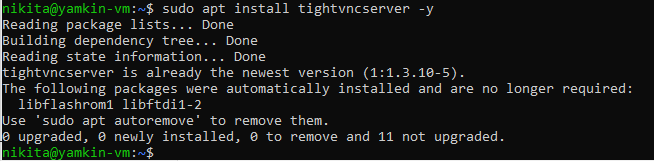


Рисунок 9 – Установка tightvncserver

**Настройка VNC-сервера:**

Запустить VNC-сервер можно следующей командой:

|  |
| --- |
| vncserver |

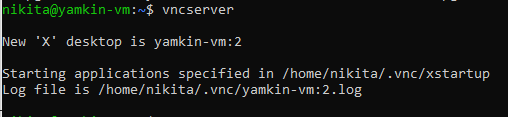


Рисунок 10 – Запуск VNC сервера

Далее программа попросит придумать пароль (от 6 до 8 символов), который в дальнейшем будет использоваться для подключения к сессии VNC.

**Подготовка VNC для работы с окружением XFCE**

Чтобы при подключении к сессии запускалась графическая среда, необходимо выполнить настройку VNC для работы с XFCE.

Для начала нужно остановить работающий дисплей. Чуть раньше говорилось, что ему присвоен порядковый номер «1», поэтому команда составляется следующим образом:

|  |
| --- |
| vncserver -kill :1 |



Рисунок 11 – Остановка сервера

Теперь нужно удалить старый конфигурационный файл, отвечающий за запуск графической оболочки и создать вместо него новый:

|  |
| --- |
| rm -f ~/.vnc/xstartup  nano ~/.vnc/xstartup |

Откроется пустой документ, в который необходимо добавить следующий код:

|  |
| --- |
| #!/bin/sh  unset SESSION\_MANAGER  unset DBUS\_SESSION\_BUS\_ADDRESS  exec startxfce4 |

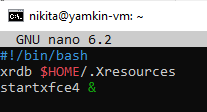


Рисунок 12 – Запуск графической оболочки

**Настройка автозапуска**

После перезагрузки удалённого компьютера, запуск VNC-сервера не будет выполняться самостоятельно. Это можно исправить, добавив службу автозапуска.

Для начала нужно создать скрипт, при выполнении которого будет запускаться TightVNCServer с определённым набором параметров:

|  |
| --- |
| sudo nano /usr/local/bin/myvnc |

Теперь в открывшийся пустой файл нужно занести следующий код:

|  |
| --- |
| #!/bin/bash  PATH="$PATH:/usr/bin/"  DISPLAY="1"  DEPTH="16"  GEOMETRY="1024x768"  OPTIONS="-depth ${DEPTH} -geometry ${GEOMETRY} :${DISPLAY}"  **case** "$1" in  start)  /usr/bin/vncserver **${OPTIONS}**  ;;  stop)  /usr/bin/vncserver -kill :**${DISPLAY}**  ;;  restart)  **$0** stop  **$0** start  ;;  esac  **exit** 0 |

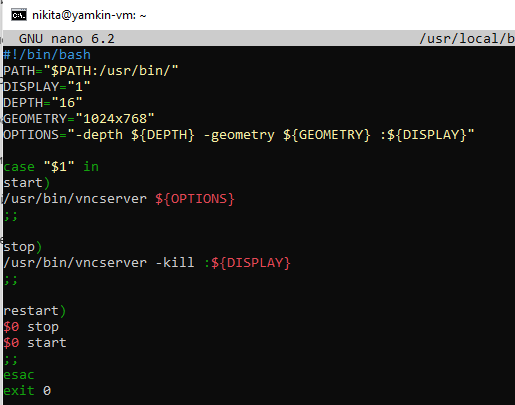


Рисунок 13 – Настройка автозапуска

Остаётся сделать файл исполняемым, чтобы скрипт стал доступен для запуска:

|  |
| --- |
| sudo chmod +x /usr/local/bin/myvnc |

Скрипт готов, но теперь нужно сделать так, чтобы запуск производился каждый раз при загрузке системы. Для этого создаётся юнит:

|  |
| --- |
| sudo nano /lib/systemd/system/myvnc.service |

В создаваемый файл вносится следующий код:

|  |
| --- |
| [Unit]  Description=MyVnc  [Service]  Type=forking  ExecStart=/usr/local/bin/myvnc start  ExecStop=/usr/local/bin/myvnc stop  ExecReload=/usr/local/bin/myvnc restart  User=username  [Install]  WantedBy=multi-user.target |

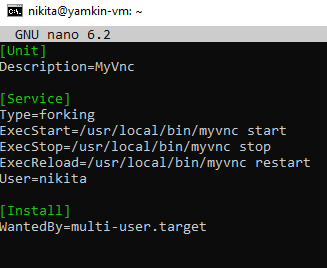


Рисунок 14 – Настройка автозапуска

В переменную «User» вводится логин пользователя Linux, от имени которого будет запускаться VNC-сервер.

Теперь нужно сохранить файл и добавить созданную службу в автозагрузку:

|  |
| --- |
| sudo systemctl enable myvnc.service |

Для применения настроек остаётся перезапустить демон systemd и VNC-сервер следующими командами:

|  |
| --- |
| sudo systemctl daemon-reload  myvnc restart |

В результате, после настройки VNC-сервера (виртуальная машина) и VNC-клиента (локальный компьютер), имеем возможность администрировать машину через графический интерфейс.

**Подключение к ВМ**

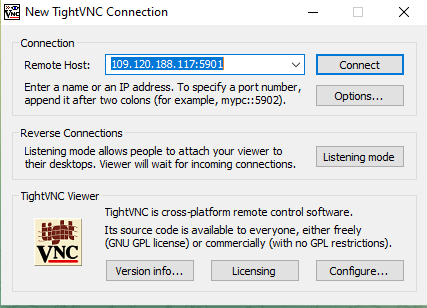


Рисунок 15 – Подключение к ВМ через VNC

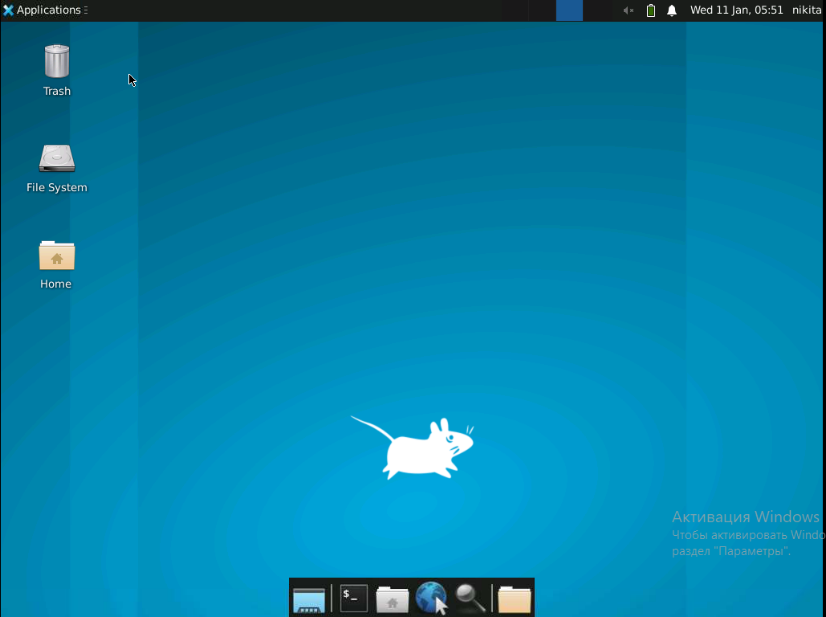


Рисунок 16 – Рабочий стол ВМ

Проверим подключение к интернету, хотя это и не надо было, так как наша виртуальная машина находится не у нас на рабочем компьютере, а в Облаке VK.

Скачаем, установим и запустим браузер Google Chrome.

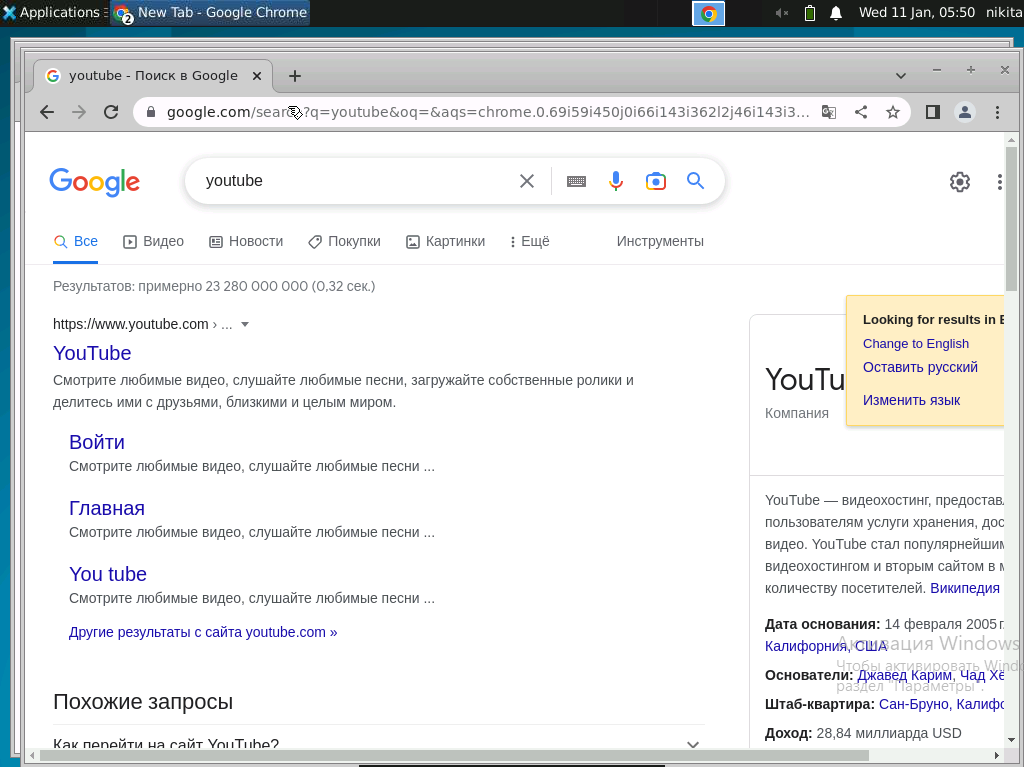


Рисунок 17 – Интернет

**Заключение**

В ходе лабораторной работы была проведена ознакомительная работа с инструментами и услугами, предлагаемыми IT-платформой VK Cloud Solutions.

В качестве практической реализации был создана виртуальная машина Ubuntu 22.04 на VK Cloud Solutions.

**Список использованных источников**

1. Видеогайд по созданию и настройке виртуальной машины // youtube.com. URL: <https://www.youtube.com/watch?v=eV7imTBAc-4/> (дата обращения: 11.01.2023).