Министерство науки и высшего образования РФ
Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра вычислительной техники

УТВЕРЖДАЮ
Заведующий кафедрой
_____О.В. Непомнящий
«___»_____2024 г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Разработка системы контроля доступа и защищенных протоколов обмена к лабораторному оборудованию

09.04.01 Информатика и вычислительная техника

09.04.01.11 «Вычислительные системы и сети»

Руководитель	_____	_____	доцент, канд. техн. наук	С.Н. Титовский
	подпись	дата	должность, ученая степень	
Выпускник	_____	_____		А.В. Данилович
	подпись	дата		
Рецензент	_____	_____	доцент, канд. техн. наук	А.А. Мазуров
	подпись	дата	должность, ученая степень	
Консультант	_____	_____	доцент, канд. техн. наук	К.В. Коршун
	подпись	дата	должность, ученая степень	
Нормоконтролер	_____	_____	доцент, канд. техн. наук	С.Н. Титовский
	подпись	дата	должность, ученая степень	

Красноярск 2024

РЕФЕРАТ

Выпускная квалификационная работа по теме «Разработка системы контроля доступа и защищенных протоколов обмена к лабораторному оборудованию» содержит 48 страниц текстового документа, 26 рисунков, 1 таблицу, 37 использованных источников.

Ключевые слова: Лабораторное оборудование, система контроля доступа, протокол, контейнер, фаервол.

Цель работы – обеспечение защищенности системы дистанционного доступа к лабораторному оборудованию посредством настройки программного обеспечения от несанкционированного доступа злоумышленниками.

В первой главе рассмотрены известные программные и аппаратные решения по удаленному доступу к лабораторному оборудованию. Рассмотрено понятие сетевого протокола. Также рассмотрены существующие способы атак и методы защиты от них.

Во второй главе проведена настройка операционной системы для обеспечения защиты данных и создан SSL-сертификат, как способ защиты данных.

В третьей главе проведено тестирование операционной системы на уязвимости до и после настройки системы для защиты.

В результате работы над магистерской диссертации разработана система контроля доступа к лабораторному оборудованию, создан и внедрен в систему SSL-сертификат для защищенной передачи данных между клиентом и сервером.

СОДЕРЖАНИЕ

АННОТАЦИЯ

Проводятся исследования методов, способов и протоколов защищенного дистанционного доступа к оборудованию. Разработка программных моделей и протоколов для дистанционного доступа к лабораторному оборудованию ИКИТ СФУ. Разработка математического, алгоритмического и программного обеспечения, поддержки протоколов удаленного доступа к лабораторному оборудованию. Тестирование разработанных программных моделей и протокола обмена для удаленного доступа к лабораторному оборудованию.

Отличительной особенностью разработки является предложенный метод обмена данными, базирующийся на предложенных оригинальных протоколах и алгоритмах кодирования и передачи данных, которая позволяет обеспечить бесперебойный многопользовательский защищенный доступ к лабораторному оборудованию.

ВВЕДЕНИЕ

Актуальность проблемы. В настоящее время защищенные протоколы обмена данными используются практически во всех ресурсах сети Интернет. Это позволяет пользователям без опаски использовать Интернет-ресурсы для получения различной информации. Если не использовать защищенные протоколы обмена данными для защиты своего Интернет-ресурса, то и посетитель, и сервер подвергается опасности перехвата пакетов данных, что позволит хакеру сначала прослушивать передаваемые пакеты данных, так и в последствии перехватывать эти пакеты и подменять их для взлома сервера. Что позволит хакеру скомпрометировать персональные данные пользователя и использовать их в личных целях. В таком случае безопасная передача данных между пользователем и сервером является самым важным аспектом для публикации различных ресурсов в сети Интернет.

Для защиты данных требуется установка различного защитного сетевого программного обеспечения на компьютер и защищенного протокола обмена данными на *web*-сервер системы удаленного доступа к лабораторному оборудованию.

На данный момент времени данная система не защищена от различных атак и уязвима для публикации в сеть Интернет.

Таким образом, актуальностью работы является обеспечение защищенного обмена данными между сервером и клиентом с помощью программного обеспечения и использования защищенного протокола обмена данными для обеспечения безопасной передачи данных между клиентом и сервером с последующей публикацией данного ресурса в сеть Интернет.

Цель диссертационной работы: обеспечение защищенности системы дистанционного доступа к лабораторному оборудованию посредством настройки программного обеспечения от несанкционированного доступа злоумышленниками.

Для достижения поставленной цели сформулированы следующие задачи исследования:

– Исследовать методы защищенности вычислительных машин

– Исследовать существующие способы защиты вычислительных машин;

– Обеспечить безопасность данных для сервера от различных атак

– Обеспечить безопасную передачу пакетов данных между сервером и клиентом

Предполагаемая научная новизна исследования заключается в предложенном методе защиты серверов для системы удаленного доступа к лабораторному оборудованию, основывающийся на комплексном решении существующих методов защиты, который позволит усилить безопасность передачи данных в сети интернет между клиентом и сервером.

1 Анализ предметной области

1.1. Удаленный доступ, основные термины и определения

Удаленный доступ – это технология, которая позволяет дистанционно взаимодействовать с компьютером, сетями и приложениями при использовании глобальных каналов, локальных или глобальных сетей. Для настройки удаленного доступа необходимо, чтобы оба компьютера – клиент, от которого идет подключение и хост, к которому идет подключение [1].

В случае нашего проекта наиболее интересно удаленное подключение к оборудованию для обучения студентов (научному, лабораторному). Такой доступ возможен при организации удаленного рабочего стола.

Удаленный рабочий стол (Remote Desktop) — это режим управления, при котором один компьютер получает права администратора по отношению к другому, удаленному. Связь между компьютерами организуется в реальном времени через Интернет или по локальной сети [2]. При этом режимы доступа задаются в зависимости от конкретных задач и могут конфигурироваться, например:

- подключение к рабочей сессии обеспечивает полный контроль и взаимодействие с удаленным компьютером. При этом разрешается запуск на нем приложений и любые манипуляции с файлами.

- в режиме ограниченного удаленного доступа разрешено вести наблюдения за процессами, а какие-либо изменения в системе недопустимы.

Обеспечение удаленного доступа и удаленного администрирования поддерживается практически всеми операционными системами. Кроме того, имеется множество приложений, обеспечивающих удаленный доступ, которые расширяют встроенные функции операционных систем [3, 4].

1.2. Программы и мобильные приложения для создания виртуальных рабочих мест

Среди приложений удаленного доступа выделяются несколько типов в зависимости от операционной системы, с помощью которых они функционируют. Например, утилиты для MS Windows, Android и др. При этом приложения могут функционировать не только через Bluetooth или Wi-Fi, но и через мобильный интернет.

Компания Microsoft внедрила в собственную систему Windows возможность удаленного подключения к компьютеру посредством RDP протокола, который является защищенным и довольно безопасен. Однако, данный способ подключения не позволяет его использовать с любой другой операционной системы, кроме линейки операционный систем Windows, что можно считать критичным недостатком в условиях необходимо подключения к лабораторному оборудованию [5].

1.2.1. AnyDesk

Одной из самых популярных программ для удаленного доступа является AnyDesk.

Основным преимуществом данного программного обеспечения является его кроссплатформенность. ПО может использоваться практически на любой операционной системе: Windows, macOS, Android, Linux, iOS, FreeBSD, Raspberry Pi,. Также данное ПО обеспечивает безопасное и конфиденциальное подключение между подключаемыми друг к другу устройствами, что крайне необходимо в наше время. С помощью данного ПО можно оформлять передачу файлов, управлять компьютером после удаленного подключения к системе. Программа позволяет управлять правами доступа подключаемого пользователя к машине, что позволяет защитить машину от различных опасных для работы системы действий.

С помощью AnyDesk возможно организовать удаленный доступ к устройству, что позволяет воспользоваться файлами или документами из любой точки мира с помощью сети Интернет. AnyDesk имеет историю подключений, к которой производились подключения от определенной машины. Это непосредственно позволяет проверить статус машины: возможно ли к ней подключиться в данный момент времени или нет. Так же, разработчики AnyDesk гарантируют, что при плохой скорости подключения к сети Интернет, доступ к машине не будет прекращен.

Кроме того, AnyDesk может произвести перезагрузку устройства, к которому имеется активное подключение, также имеется возможность создавать различные отчеты о сеансах и их анализировать. Данные инструменты позволяют администратору устранить проблему в системе удаленно, без непосредственного посещения компьютера.

Таким образом, AnyDesk считается сильным инструментом для удаленного подключения к рабочему столу. Для использования данного ПО не требуется установка программы или административного доступа к компьютеру. Для использования программы необходимо только предварительно загрузить EXE-файл на компьютере (5 МБ) программы и запустить его, что отображено на рисунке 1 [6].

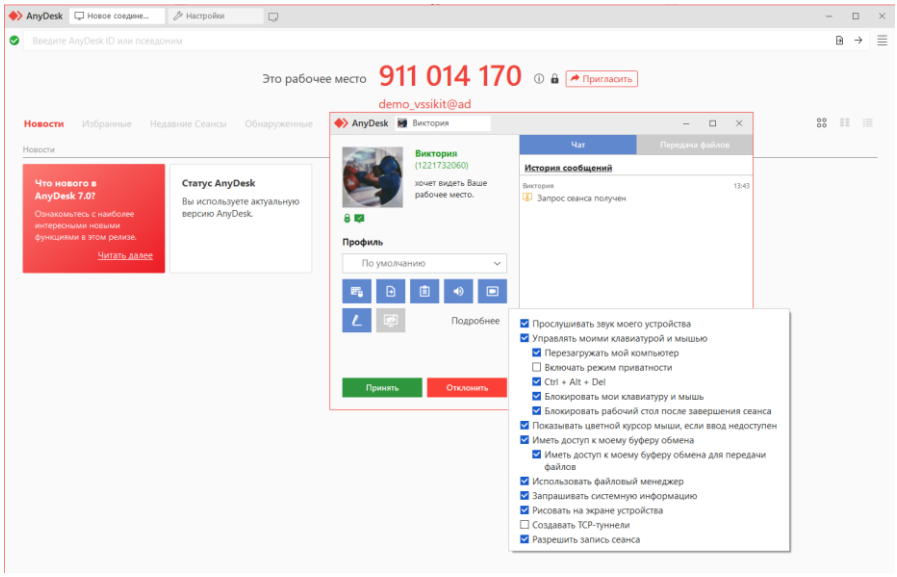

Рисунок 1 – Окно подключения к рабочему столу в AnyDesk

Также, программа имеет возможность записывать рабочий стол, а это важная функция в образовательном процессе студентов, так как с помощью данного функционала возможна фиксация процесса выполнения лабораторных работ или проведения различных экспериментов студентами с оборудование.

AnyDesk имеет одну из самых безопасных систем удаленного доступа – стандартизированный протокол TLS 1.2, который предотвращает несанкционированный доступ ко всем подключениям.

AnyDesk имеет следующие функциональные возможности:

– Удаленная печать;
– TCP-туннелирование;
– История сеансов;
– Автоматическое обнаружение компьютеров в локальной сети;
– Чат между сторонами подключения;
– Кроссплатформенный удаленный доступ;
– Возможность подключения без подтверждения подключения на удаленном компьютере;
– Двухфакторная аутентификация;
– Передача файлов и диспетчер файлов;
– Хост-сервер индивидуален;
– Протокол сеанса.

Одним из главных преимуществ является поддержка AnyDesk для Android, но данное ПО является условно-платным для большего количества одновременного подключения требуется платная версия.

1.2.2. DWService

Компания DWSNET s.r.l. разработала браузерное программное обеспечение для удаленного доступа к компьютеру с открытым исходным кодом. Также имеется программа, которая кроссплатформенна и может быть

установлена на такие ОС: Windows, Linux, macOS, Raspberry, Wandboard, Pine64. Все данные шифруются с помощью SSL-сертификата в соответствии современных стандартов безопасности. Используется двухфакторная авторизация, генерация одноразового пароля (TOTP) для учетных записей [7].

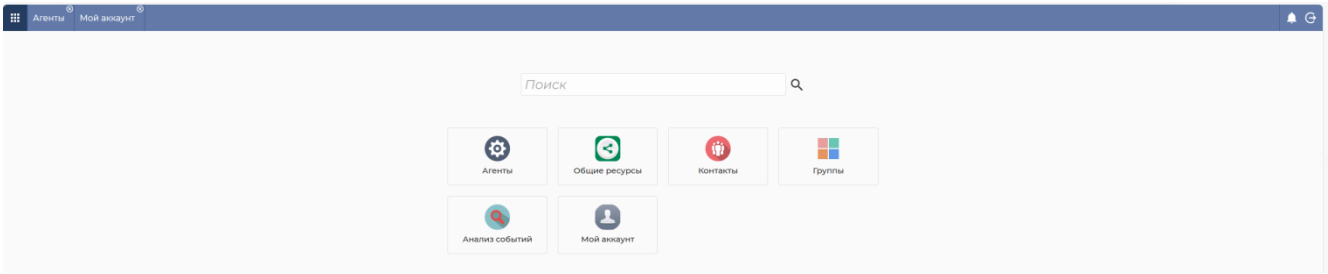

Рисунок 2– Рабочее окно DWService

1.2.3. Chrome Remote Desktop

Другим способом удаленного подключения к компьютеру можно считать программное обеспечение – браузерное расширение компании Google. Это расширение полностью работает на всех системах, где может быть установлен браузер Google Chrome. Также расширение можно использовать на Android и iOS, только необходима установка браузера на смартфон.

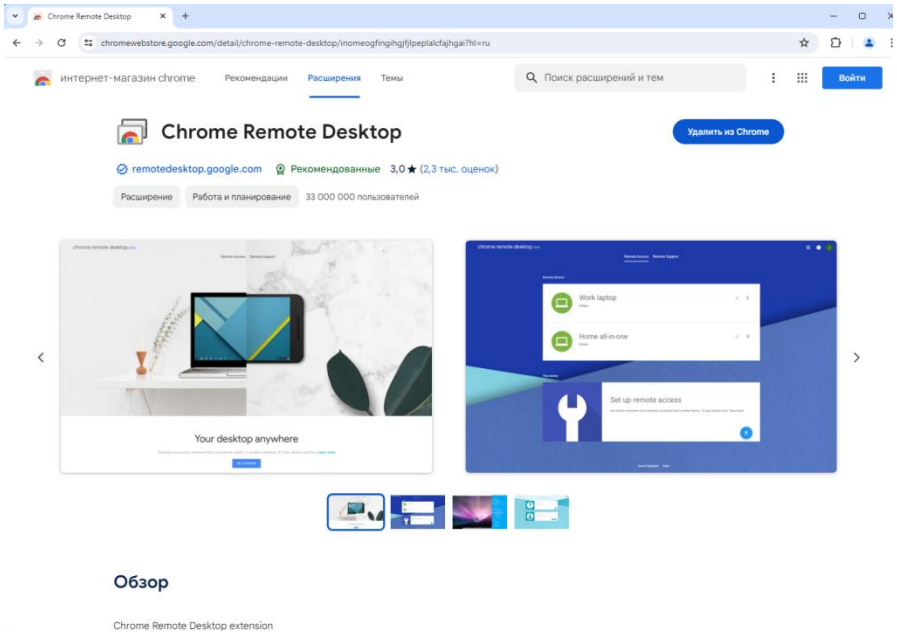

Рисунок 3 – Окно установки расширения в браузере

Для непосредственного использования отображаемого расширения требуется наличие аккаунта Google. Достоинством данного расширения является высокий уровень безопасности подключения, который гарантирует Google. Но недостатком является довольно сложный интерфейс для пользователя и обязательная установка приложения «Удаленный рабочий стол Chrome» для принимающей подключения устройства.

1.2.4. Remote Utilities

Также выделяется программа Remote Utilities, которая возможна как в личном использовании, так и в коммерческом. ПО позволяет организовать удаленный доступ одновременно до десяти ПК (Рисунок 4).

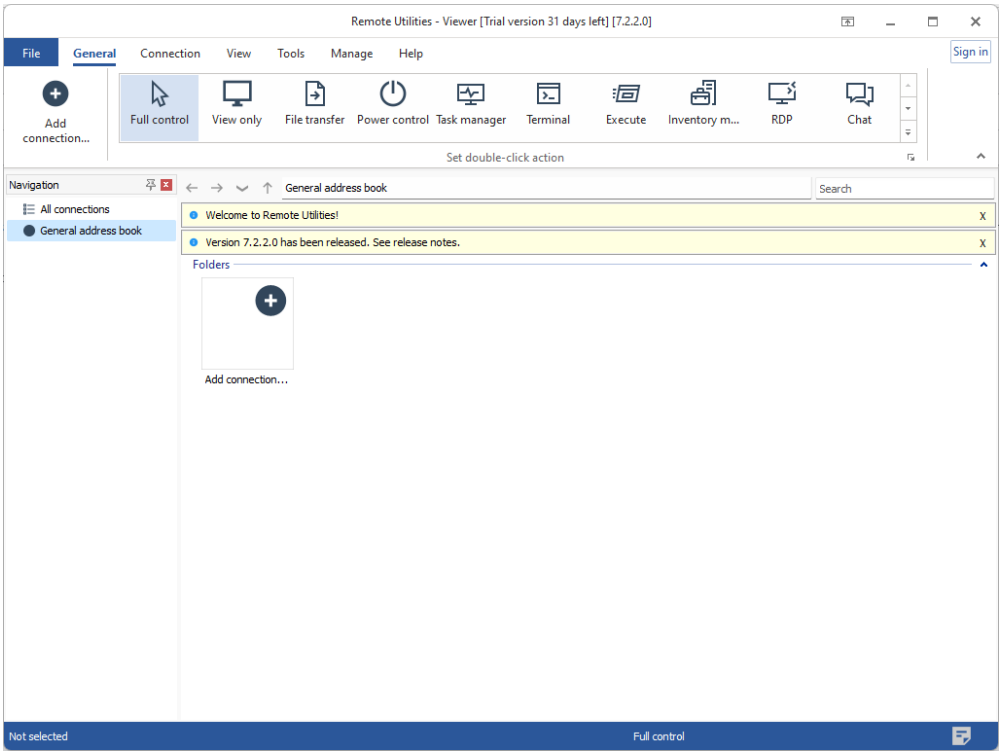

Рисунок 4 – Рабочее окно Remote Utilities

Данное ПО поддерживает следующие функции:

– Поддержка двух и более мониторов
– Передача файлов с помощью Drag & Drop
– Открытие и получение доступа к реестру и др.

1.2.5. Unified Remote

Среди мобильных приложений выделяется приложение Unified Remote, с помощью которого можно полностью управлять компьютером, к которому было произведено подключение. ПО поддерживает Windows, Linux и Mac. Приложение условно-платное, т.к. бесплатная версия приложения ограничена и для полного функционала программы необходимо провести единовременную оплату.

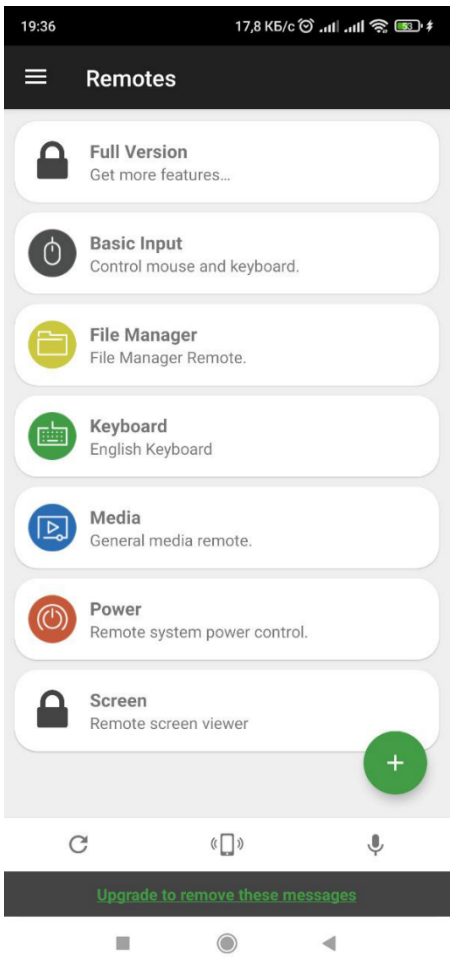

Рисунок 5 – Рабочее окно Unified Remote

1.3. Обзор известных решений по управлению лабораторными стендами

1.3.1. Система удаленного доступа НИЯУ ВШЭ

Национальный исследовательский университет «Высшая школа экономики» г. Москва, использует в собственном образовательном процессе департамента компьютерной инженерии учебную лабораторию (УЛ) систем автоматизированного проектирования, которая предлагает удаленный доступ к оборудованию УЛ САПР [8].

В лаборатории развернуты рабочие станции с подключенными к ним отладочными платами ПЛИС DE1-SoC, DE10-standart и DE10-lite (Рисунок 6).

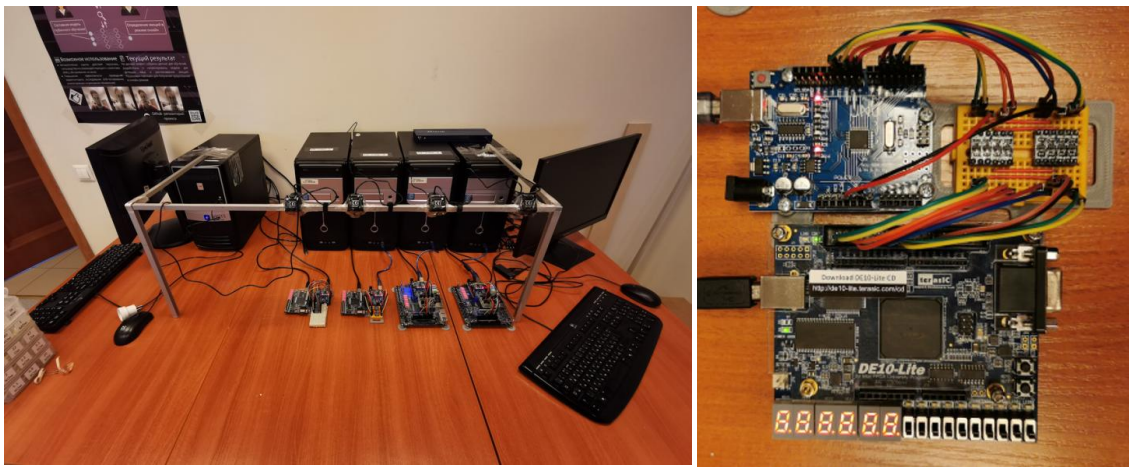

Рисунок 6 – Оборудование УЛ САПР НИУ ВШЭ

На компьютерах установлено все необходимое программное обеспечение Quartus, Modelsim и т.д. Студенты могут самостоятельно подключиться к машинам через ПО AnyDesk. Через неё могут загрузить или выполнить сборку собственного проекта, после чего могут выполнить программирование стенда.

Для того, чтобы наблюдать за результатом работы проекта на плате стенды снабжены веб-камерами и на ПК установлена программа AMCap. С ее помощью можно наблюдать за платой и выводом информации на семисегментные индикаторы и лампочки индикации. Возможна настройка, зум и коррекция изображения в зависимости от условий освещенности.

Для того, чтобы осуществить управление платами дистанционно, разработано специализированное ПО Butt Emulator. С помощью этой программы студенты через COM-порт подключаются к Arduino, которая связана своими выводами с GPIO на ПЛИС.

Описание режимов работы, способов подключения и прошивки ПЛИС отображено в [9].

Недостатком системы является обязательный запуск прикладного ПК на удаленном компьютере с обязательным подтверждением для авторизации и работа с ограниченным типом отладочных плат.

1.3.2. Виртуальные лаборатории ТПУ

Томский политехнический университет предоставляет свои собственные онлайн лаборатории, которые позволяют использовать широкий сервис к виртуальным лаборатория от аналитической химии до электроснабжения. Все лабораторные установки виртуальные и представлены в 3D – формате (рисунок 7), но именно это и является основным недостатком.

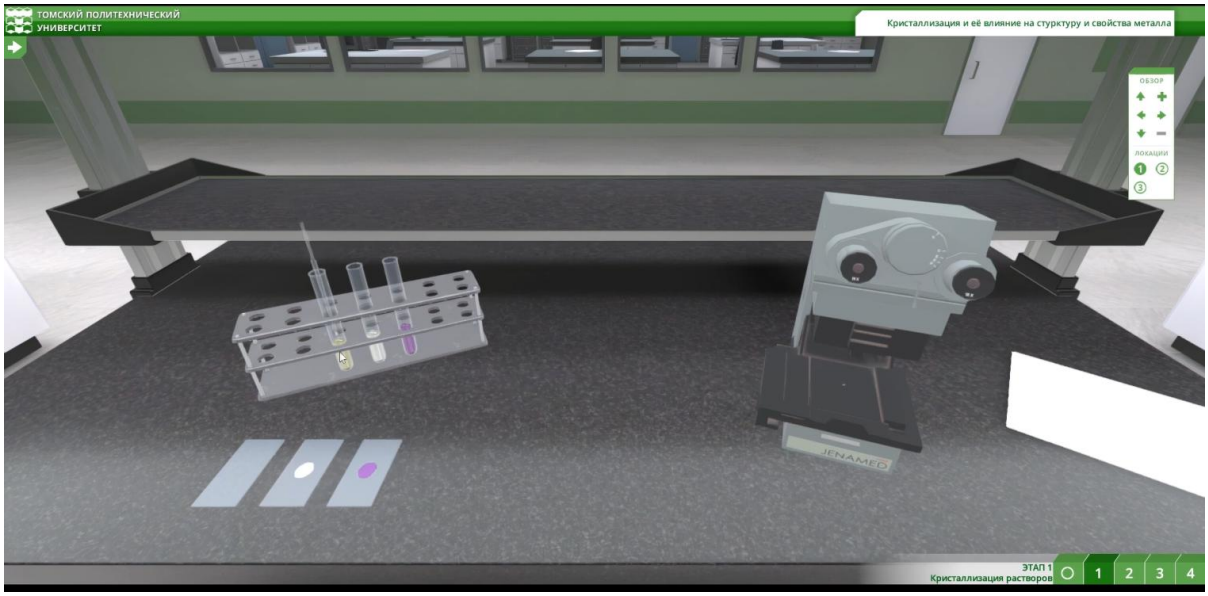

Рисунок 7 – Пример виртуального стенда ТПУ

1.3.3. Роботизированные линии удаленного доступа С-Петербурского политехнического университета им. П. Великого.

Большой интерес вызывает решение Санкт-Петербурского университета им. Петра Великого, представленной Северо-Западным межвузовским региональным учебно-научный центром "СПбПУ - ФЕСТО". В состав объединения входят более 20-и ведущих Вузов России и Европы. Центр предоставляет студентам удаленный доступ к робототехническим комплексам, роботам и оборудованию. Проводятся региональные и

международные соревнования по программированию роботов. Проводятся зимние и летние школы, реализуется ряд программ дистанционного обучения для российских и иностранных студентов в области интеллектуального управления и робототехники (Рисунок 8) [10].

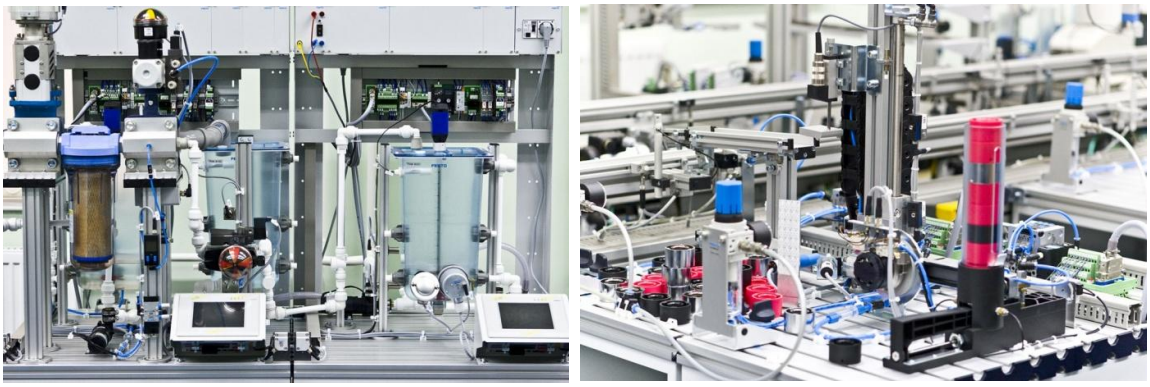

Рисунок 8 – Роботизированные линии удаленного доступа СЗ МУНЦ СПбПУ-ФЕСТО

В иностранной литературе можно встретить гораздо больше описаний различных способов практических решений и методик по применению лабораторий удаленного доступа в обучении и создании программ и сервисов виртуальных и удаленных лабораторий [11,12,13]. Но в большинстве разработанных проектов требуется непосредственное удаленное подключение к оборудованию или виртуальному стенду.

В случае системы удаленного доступа к лабораторному оборудованию, которая была ранее разработана студентами ИКИТ СФУ непосредственное подключение к машине не требуется и требуется только доступ к сети Интернет с VPN сети СФУ.

Но для последующего выхода системы в открытую сеть интернет требуется обеспечить безопасность передаваемых данных между пользователем и сервером.

Для этого необходимо проанализировать, что требуется для обеспечения безопасности и изучить виды и методы сетевых атак и способов защиты от них.

1.4. Сетевой протокол и программное обеспечение, основные термины и определения

Сетевой протокол – это набор правил и соглашений, используемых для связи устройств на определенном сетевом уровне. Протоколы обеспечивают и определяют формат обмена информации между участниками компьютерных сетей. В работе сетей используется различное количество протоколов [14].

Программное обеспечение (ПО) – это набор различных программ, которые необходимы для работы с компьютером. ПО в данном случае позволит контролировать стабильность работы системы и уведомлять о каких-либо изменениях.

В данном случае нам интересны сетевые протоколы прикладного уровня и программное обеспечение, которое позволит защитить сервер и клиента от несанкционированного доступа:

HTTP – протокол, позволяющий получать различные ресурсы посредством обмена данными в сети Интернет. HTTP является протоколом клиент-серверного взаимодействия, что позволяет инициировать запросы к серверу самим получателем. Полученный пакет данных может состоять из различных документов, изображений, видеофайлов, скриптов, текста и других файлов [15]. Для защиты данного протокола требуется протокол SSL/TLS.

SSL и TLS – это протоколы, который используется для защиты конфиденциальности данных, передаваемых через сеть Интернет. Передача данных обеспечивается через данные протоколы осуществляется при помощи аутентификации и шифрования данных, что обеспечивает безопасную передачу данных между клиентом и сервером [16, 17].

Transmission Control Protocol / Internet Protocol (TCP/IP) – Промышленный стандарт стека протоколов, разработанный для глобальных сетей. Основным элементом данного стека протоколов является Internet

Protocol (IP) – протокол межсетевого взаимодействия, с помощью которого реализуется процесс передачи пакетов данных в сети. Надежность доставки передаваемых пакетов обеспечивает протокол управления передачей данных с организацией виртуальный соединений на транспортном уровне (TCP). Транспортный уровень, формирует данные сегментами (пакетами), которые передаются сетевому уровню – IP-протоколу, который имеет свою задачу – маршрутизация при доставке пакетов от отправителя к получателю [18].

На основании темы и задания ВКР нас интересует программное обеспечение и защищенные протоколы обмена данными позволяющие обеспечить сетевую защиту к лабораторному оборудованию.

1.5. Виды и методы сетевых атак

Перед осуществлением настройки сервера необходимо рассмотреть какие на данный момент времени существуют виды и методы атак на сервер и систему. Виды сетевых атак рассмотрены в таблице 1:

Таблица 1 Виды сетевых атак

Наименование атаки	**Описание атаки**
1	**2**
1 Denial of Service (DoS)	Атака с целью вызвать перегрузку подсистемы, через которую работает атакуемый сервис, что в последствии может вызвать выход из строя сам сервер. Но с помощью использования Firewall можно довольно просто отразить данную атаку, т.к. она сильно заметна по содержимому лог-файла системы, которое обнаружит система мониторинга и предупредит администратора о происходящей атаке в реальном времени.
2 Distributed Denial of Service (DDoS)	Улучшенная форма DoS-атаки на веб-систему с целью вывода из строя или затруднения доступа для обычных пользователей к ресурсу. Это улучшенная версия DoS-атаки, т.к. проводится не из одной точки устройства, а с распределенной сети, которая содержит в себе большое количество различных устройств. С помощью данной сети происходит одновременная отправка большого количества запросов на сервер, что позволяет перегрузить сервер до последующего его выхода из строя.
3 SQL-инъекции	Атака, которая содержит вредоносный код SQL для получения информации, которая не предназначена для отображения. Данные могут содержать в себе персональные данные, конфиденциальную информацию пользователей, логины и пароли [19].

Продолжение таблицы 1

1	2
4 Man-In-The-Middle (MITM)	Атака, для перехвата разговора или передачи данных между двумя клиентами или клиентом и сервером притворяясь легальным участником процесса.
5 Снифферы	Программное или аппаратное средство, которое отслеживает интернет-трафик и перехватывает все данные, поступающие на компьютер и отправляемые с него в режиме реального времени [20].
6 Фишинг	Вид атаки, когда злоумышленник вынуждает совершить действие, которое позволит ему получить доступ к устройству, учетным и персональным данным [21].
7 Атака на цепочку поставок	Атака, с помощью которой происходит внедрение вирусов или другого вредоносного ПО посредством стороннего человеках [22].
8 Bruteforce	Попытка подбора пароля перебором всех комбинаций.

После анализа было определено, что основной проблемой для корректной работы сервера и системы могут стать: DoS-атаки, DDoS-атаки, SQL-инъекции, снифферы пакетов, MITM-атаки и Bruteforce:

1.6. Методы и подходы защиты данных

Межсетевой экран (Firewall) – система защиты компьютерной сети, которая позволяет ограничить прохождение входящего, исходящего и внутрисетевого трафика. Он контролирует разрешенную и запрещенную веб-активность в частной сети, что позволит предотвратить несанкционированные действиями пользователями как внутри частной сети, так и за её пределами [23].

Системы мониторинга сервера – это процесс сбора и анализа данных о текущей производительности системы хранения данных. Для этого используются различные утилиты, функциональность которые отличаются друг от друга. Например, некоторые программы позволят контролировать количество свободной памяти и нагрузку на CPU сервера, как другие подходят для расширенного мониторинга систем хранения данных.

Данные системы работают в реальном времени и позволяют обнаружить проблему до того, как пользователи ощутят эти проблемы на

себе. Например, если будет произведена DoS (Denial of Service) атака, то нагрузка на RAM память будет колоссальной, что позволит вывести из строя работу сервера и повлечет за собой большое количество проблем. Для этого требуется иметь достаточное количество ресурсов, чтобы система работала стабильно. Так, для стабильной работы системы и усиления работы системы требуется оптимизация используемых ресурсов и усиление мер безопасности, с чем хорошо справляется контейнеризация используемого продукта.

Контейнеризация – метод, с помощью которого программный код упаковывается в один исполняемый файл вместе с библиотеками и зависимостями, чтобы обеспечить корректный запуск программы. При возможных последующих переносах готового проекта можно избежать большинства различных проблем, которые связаны с настройками сервера, т.к. сформированный контейнер не зависит от настроек основной операционной системы и может работать на любой платформе или в облаке [24].

Указанные выше протоколы и программное обеспечение имеют различные варианты настройки, что обеспечит безопасную работу сервера и клиента, который обращается к серверу. Но также необходимо учитывать возможные сетевые атаки на сервер, от которых также необходима защита.

1.7. Выбор программного и аппаратного обеспечения

Docker – платформа контейнеризации, предназначенная для разработки, доставки и организации работы приложений. С помощью данной платформы можно упаковать и запустить приложение в изолированном окружении – контейнере. Docker позволяет управлять жизненным циклом контейнеров:

— Инкапсулировать готовые приложения в контейнеры Docker;

— Распространять и доставлять собранные контейнеры остальным участникам проекта для дальнейшей разработки и последующего тестирования продукта;

— Разворачивать приложения в практически любом окружении, например, какой-либо датацентр или облако [25].

Следующее программное обеспечение – SELinux. Может использоваться во всех Linux системах. Является архитектурой безопасности, которая позволяет администраторам лучше контролировать пользователей, кто может получить доступ к системе.

SELinux определяет средства управления доступ для приложений, процессов и файлов в системе. Он использует политики безопасности, которые представляют собой набор политик безопасности, которые указывают SELinux, к чему можно получить доступ, а к чему нет [26].

Iptables – утилита командной строки, является стандартным интерфейсом управления работой межсетевого экрана (брандмауэра) для ядер Linux. С помощью данной утилиты управлять фильтрацией и перенаправлением пакетов через права администратора [27].

Для создания SSL-сертификата потребуется бесплатный центр сертификации Let’s Encrypt и Certbot для создания и активации сертификата.

Let’s Encrypt – это бесплатный, автоматизированный и открытый центр сертификации, созданный некоммерческой организацией Internet Security Research Group (ISRG). Более 420 миллионов сайтов получили сертификат от центра сертификации Let’s Encrypt. Этим можно сказать, что Let’s Encrypt является одним из самых популярных центров сертификации в мире. Для автоматизации создания SSL-сертификата владельцы данного центра сертификации разработали утилиту с открытым исходным кодом для автоматизации выдачи сертификата на вручную управляемых веб-сайтах для обеспечения работы HTTPS протокола – Certbot. Все это позволит автоматизировать выдачу сертификата, продление и при необходимости отзывы сертификата. Срок действия сертификата 90 календарных дней. Но,

если настроить Certbot на автоматическое продление сертификата, то тогда не будет необходимости каждый раз, перед завершением срока действия сертификата его продлевать вручную. Но если у нас имеется веб-сервер отличный от имеющегося списка веб-серверов разработчиков, то потребуется оформлять сертификат вручную. Это позволит создать бесплатный ssl-сертификат для Node.js, что является в нашем случае обязательным фактором разработки [28, 29].

1.8. Выводы по главе 1

Рассмотрены различные способы атак на сервер. Определено, что в нашем случае критически важно иметь защиту от DoS, DDoS атак, SQL-инъекций, MITM-атак, снифферов пакетов и предотвращения побега из контейнера docker. Были рассмотрены способы защиты от хакерских атак на сервер. Для обеспечения стабильной работы сервера достаточно установить SSL/TLS-сертификат на web-сервер Node.js с помощью встроенного функционала программной платформы и провести настройку Firewall для стабильной работы сервера с последующей контейнеризацией и ежедневным резервным копированием. Также необходимо обеспечить защищенность операционной системы. Для этого требуется подключение программного обеспечения SELinux, чтобы при нестандартном поведении системы производилась рассылка уведомлений администратору системы и производился разрыв сессии с пользователем при таких атаках. Данный вывод позволяет перейти к следующему этапу работы.

2 Настройка и обеспечение защиты данных системы удаленного доступа к лабораторному оборудованию

2.1. Аппаратное обеспечение лабораторного оборудования

Согласно заданию на ВКР, требуется обеспечить безопасное подключение пользователей к лабораторному оборудованию. В ИКИТ СФУ развернута лаборатория удаленного доступа, состав которой приведен на рисунке 9.

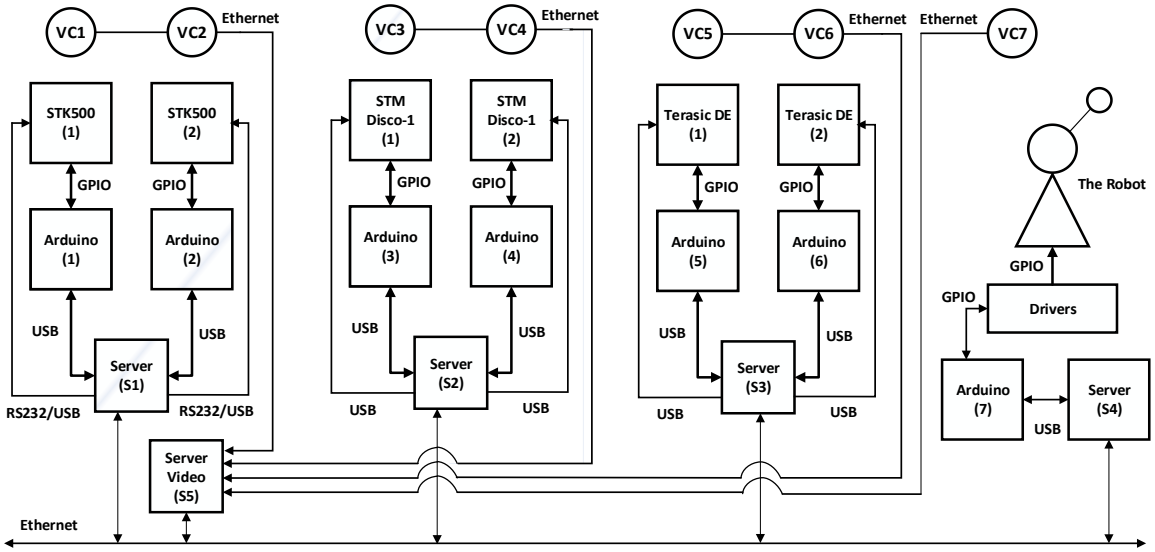

Рисунок 9 – Состав лаборатории удаленного доступа ИКИТ СФУ

В состав лаборатории включают в себя стенды STK500, STM32 и DE1-SoC.

Данные стенды присоединены к серверам и управляются с помощью одноплатных систем Arduino и/или RAsberyPI. Для обработки электрических сигналов в сигналы управления платой используются усилители (драйвера) двигателей. При подключении лабораторных стендов электрические параметры сигналов не требуют преобразования и поэтому они подключатся непосредственно при помощи кабельной системы. Однако для выполнения некоторых лабораторных работ студентами к собранным стендам

подключены дополнительные датчики, интерфейсы и исполнительные устройства. Например, ЦАП, ЖКИ, кнопки и переключатели, они ходят в состав стендов. Для них, при помощи управляющих одноплатных компьютеров, происходит имитация входных сигналов для плат (Рисунок 10).

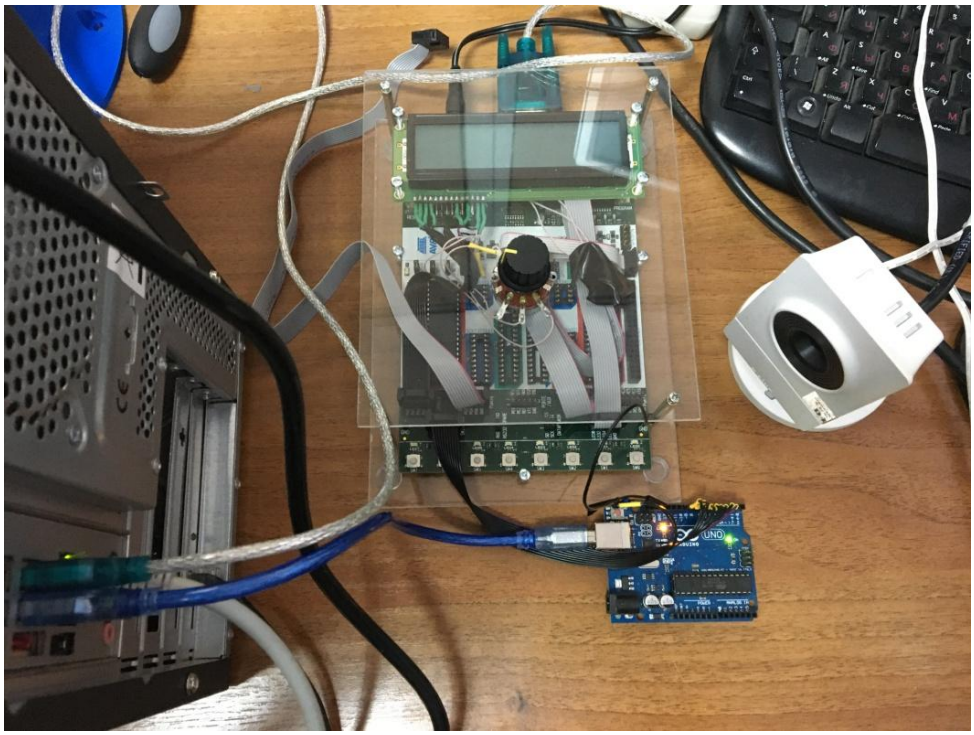

Рисунок 10 – Пример подключения платы STK500 к серверу

На рисунке 11 представлена общая архитектура системы.

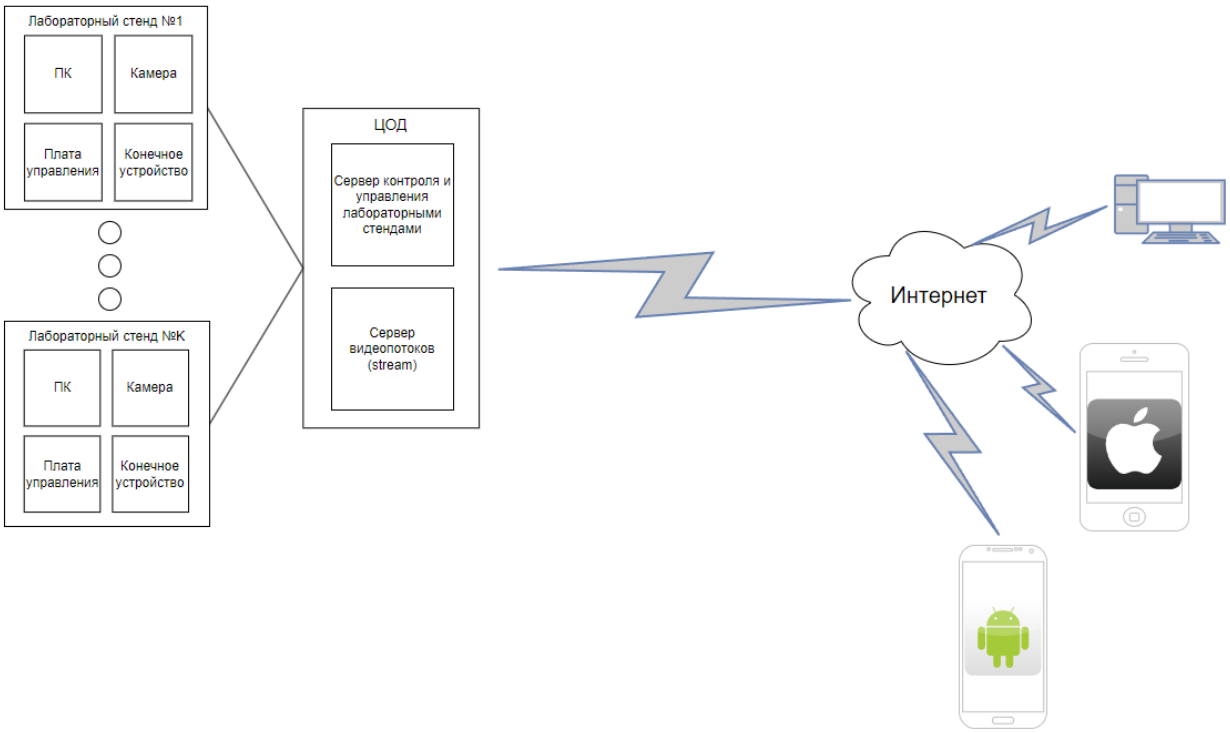

Рисунок 11 – Общая архитектура системы удаленного доступа

,

На рисунке 12 представлена система взаимодействия клиента с сервером.

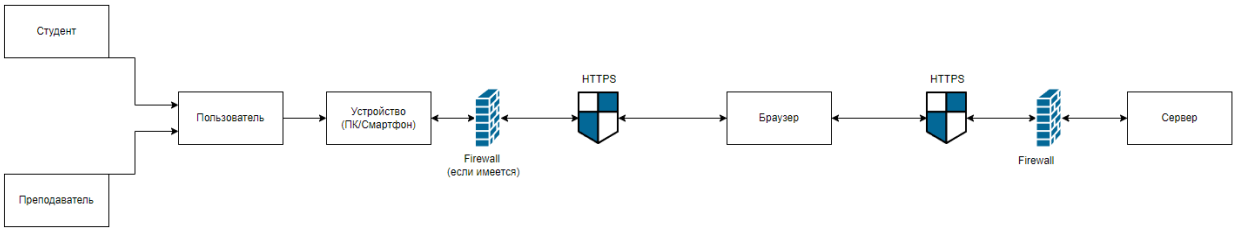

Рисунок 12 – Схема взаимодействия пользователя с сервером

Разрабатываемая система обладает сложной архитектурой, которая сочетает в себе большое количество различных аппаратных и программных компонентов.

Основными аппаратными элементами системы являются:

- конечное устройство пользователя;
- центр обработки данных;
- лабораторный стенд.

Для доступа к стенду пользователю необходимо иметь устройство с доступом в Интернет. Таким устройством может быть любой персональный компьютер или смартфон на базе операционной системы Android или IOS и др..

Центр обработки данных имеет два веб-сервера:

- сервер контроля и управления лабораторным стендом;
- сервер видеопотоков(stream).

Сервер контроля и управления стендом отвечает за взаимодействие с базой данных и передачу данных пользователю, а также за предоставление пользователю доступа к лабораторному стенду.

Сервер видеопотока отвечает за обеспечение прямой трансляции с собранного лабораторного стенда. Он выступает в роли прокси-сервера и занимается перекодированием исходных видеопотоков, полученных по протоколу RTSP, в необходимый формат, который зависит от конечного устройства пользователя. После транскодирования видеопотоки могут быть переданы клиентскому приложению по протоколу HLS, который может быть прочитан браузерами на любом устройстве.

2.2. Организация сетевого взаимодействия

На момент начала работы доступ организован в внутренней сети СФУ ИКИТ и реализуется следующим образом:

1. Администратор подключается к серверу при помощи текстового (терминального) интерфейса, используя встроенные в ОС средства (ssh, PuTTY).
2. На сервер загружается исходный код для Arduino в текстовом виде.

3. Код компилируется и отправляется в Arduino. Изначально это производилось вручную, далее при помощи разработанных скриптов все операции выполняются автоматически.

4. После отправки программы в Arduino она начинает выполняться.

При этом изображение работающего оборудования захватывается камерой, выполняется перекодировка видео для требуемых показателей качества.

Видео публикуется в формате HLS, зрители могут просматривать его при помощи браузера или любого видеоплеера. Администратор сам может быть зрителем (Рисунок 13).

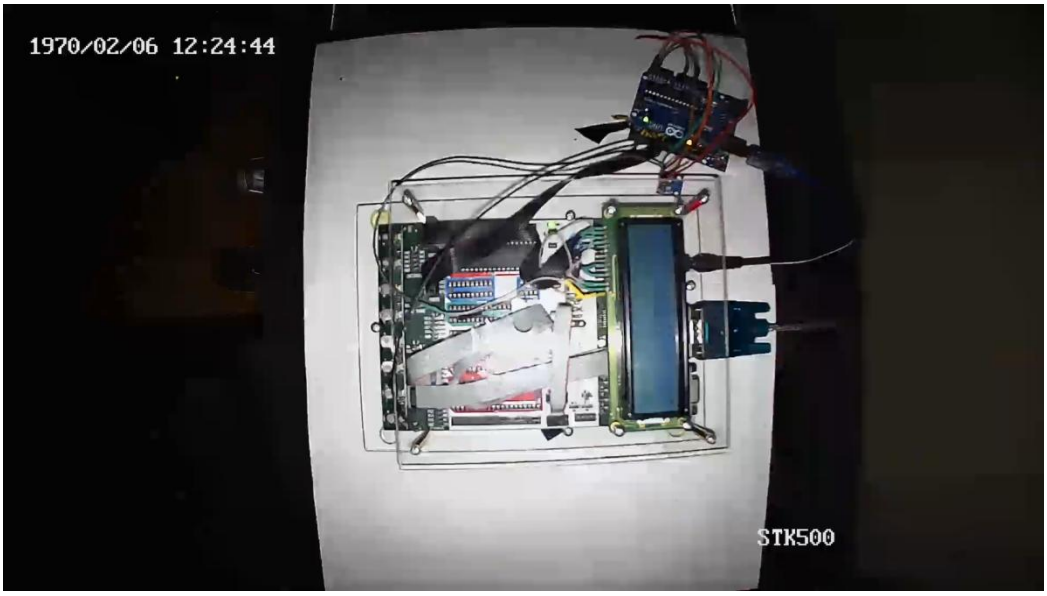

Рисунок 13 – Пример видео транслируемого через сервер

2.3. Установка необходимого программного обеспечения и настройка операционной системы

Для обеспечения защиты системы удаленного доступа к лабораторному оборудованию требуется дополнительная установка и настройка программного обеспечения, такого как: Docker, IPTables, Certbot.

Также перед началом работы был проведен анализ используемых WEB-приложений. Ими оказались Angular, Strapi и NestJS.

Angular – это frontend веб-фреймворк, позволяющий разработчикам создавать быстрые и надежные приложения.

Поддерживаемый специальной командой Google, Angular предоставляет широкий набор инструментов, API и библиотек для упрощения и оптимизации рабочего процесса разработки. Angular - это надежная платформа, на которой можно создавать быстрые и надежные приложения, масштабируемые как с ростом численности вашей команды, так и с ростом вашей кодовой базы [30].

Strapi – это headless-CMS с открытым исходным кодом имеющая в себе панель администратора и API, которые являются расширяемыми – и каждая часть которой имеет настройку под конкретный случай использования. Также Strapi имеет встроенную систему пользователей, позволяющую детально управлять тем, к чему имеют доступ администраторы и конечные пользователи. [31]

NestJS – платформа для создания эффективных масштабируемых программ Node.js на стороне сервера.

Платформа использует JavaScript, создан и полностью поддерживает TypeScript, объединяет элементы объектно-ориентированного программирования, функционального программирования и функционального реактивного программирования [32].

2.3.1. Установка и создание контейнеров Docker

Для установки Docker требуются установить пакеты docker-ce, docker-ce-cli, containerd.io, docker-buildx-plugin и docker-compose-plugin, ca-certificates, apt-transport-https, software-properties-common. Все действия необходимо с правами sudo. Также необходимо ввести данные команды:

sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc

sudo chmod a+r /etc/apt/keyrings/docker.asc

После установки пакетов можно проверить статус работы Docker (Рисунок 14) [33]:

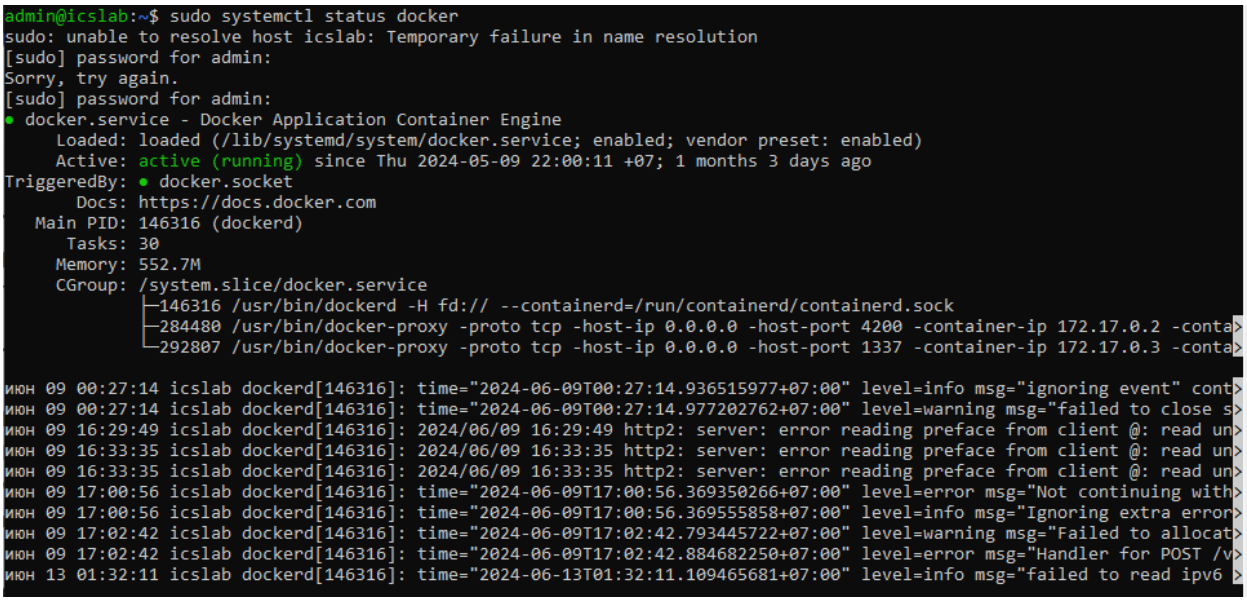

Рисунок 14 – Проверка статуса работы Docker

Далее необходимо создать Dockerfile и заполнить его в директории проекта для создания контейнера с нужными приложениями, а также скопировать содержимое проекта для последующего запуска приложения в контейнере. Содержимое Dockerfile для frontend приложения описано на рисунке 15:

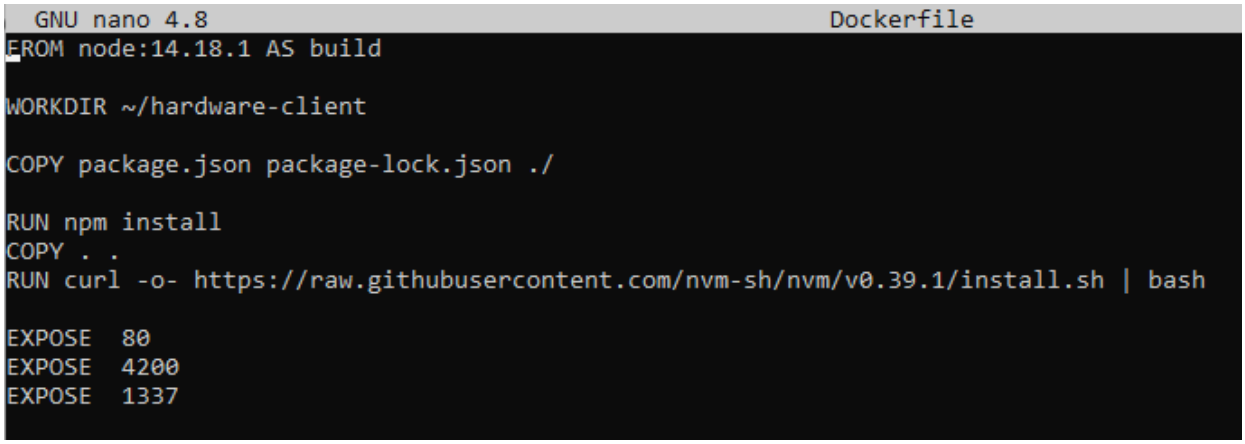

Рисунок 15 – Конфигурация Dockerfile для frontend приложения

Для административной панели и платформы NestJS требуются аналогичные образы, только без установки Certbot. (Рисунок 16):

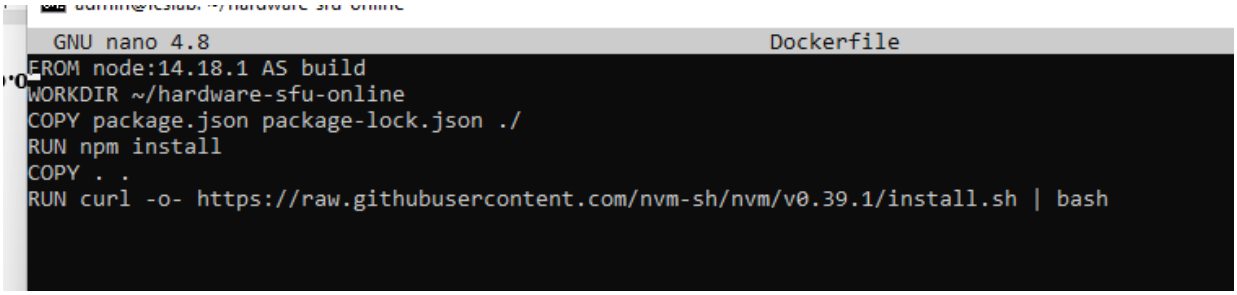

Рисунок 16 – Конфигурация Dockerfile для административной панели и NestJS

Создание отдельной сети docker не требуется, т.к. docker самостоятельно создает сеть между созданными контейнерами (Рисунок 17):

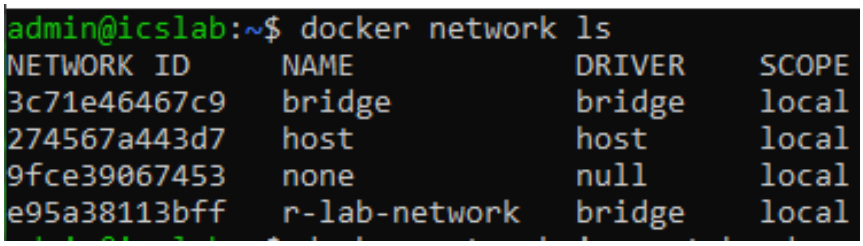

Рисунок 17 – Список существующих сетей Docker

Для сети между контейнерами используются стандартные настройки (Рисунок 18):

Рисунок 18 – Конфигурация docker network bridge

Далее запускаем контейнеры в стандартном режиме без каких-либо привилегий. Это требуется для исключения возможности побега из контейнера. Иначе при возможном взломе сервера злоумышленник может сбежать в host-систему.

После входим в контейнеры, обновляем версию Node.js и устанавливаем пакет nvm предварительно зайдя в контейнер с помощью команды curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.37.2/install.sh

| bash и запускаем приложение с помощью команды npm run start, аналогично для другого контейнера, только необходимо запустить проект Strapi командой npm run build.

2.3.2. Настройка межсетевого экрана

Далее требуется настроить межсетевой экран. Настройка будет проводиться с помощью утилиты iptables.

2.3.2.1. Настройка таблицы filter

Для корректной работы системы, перед переводом таблицы для входящих пакетов в статус DROP требуется предварительно открыть доступ по ssh к машине и веб-приложению необходимо открыть порты для взаимодействия между сервисами. Это порты: 22, 4200, 1337 и 3001.

Для того, чтобы разрешить подключение к машине по SSH, требуется ввести команду sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT. Далее открываем доступ к портам веб – приложения с помощью команд:

- sudo iptables -A INPUT -p tcp --dport 1337 -j ACCEPT
- sudo iptables -A INPUT -p tcp --dport 3001 -j ACCEPT
- sudo iptables -A INPUT -p tcp --dport 4200 -j ACCEPT

После открываем для сервера возможность обрабатываться HTTP пакеты:

- sudo iptables -A INPUT -p tcp 80 -j ACCEPT.

Далее требуется разрешить обрабатывать транзитные пакеты для docker:

- sudo iptables -A INPUT -i docker0 -j ACCEPT
- sudo iptables -A FORWARD -i docker0 -j ACCEPT
- sudo iptables -A FORWARD -i docker0 ! -o docker0 -j ACCEPT

После перевода политики в DROP будет обнаружено замедление работы прав sudo. Эта проблема решается с помощью разрешения входящих, исходящих и транзитных пакетов на порты DNS для TCP и UDP пакетов с последующим разрешением уже устоявшихся соединений:

- sudo iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
- sudo iptables -A INPUT -p tcp --dport 53 -j ACCEPT
- sudo iptables -A INPUT -p udp --dport 53 -j ACCEPT
- sudo iptables -A OUTPUT -p tcp --dport 53 -j ACCEPT
- sudo iptables -A OUTPUT -p udp --dport 53 -j ACCEPT

Конечная настройка таблицы filter отображена на рисунке 19:

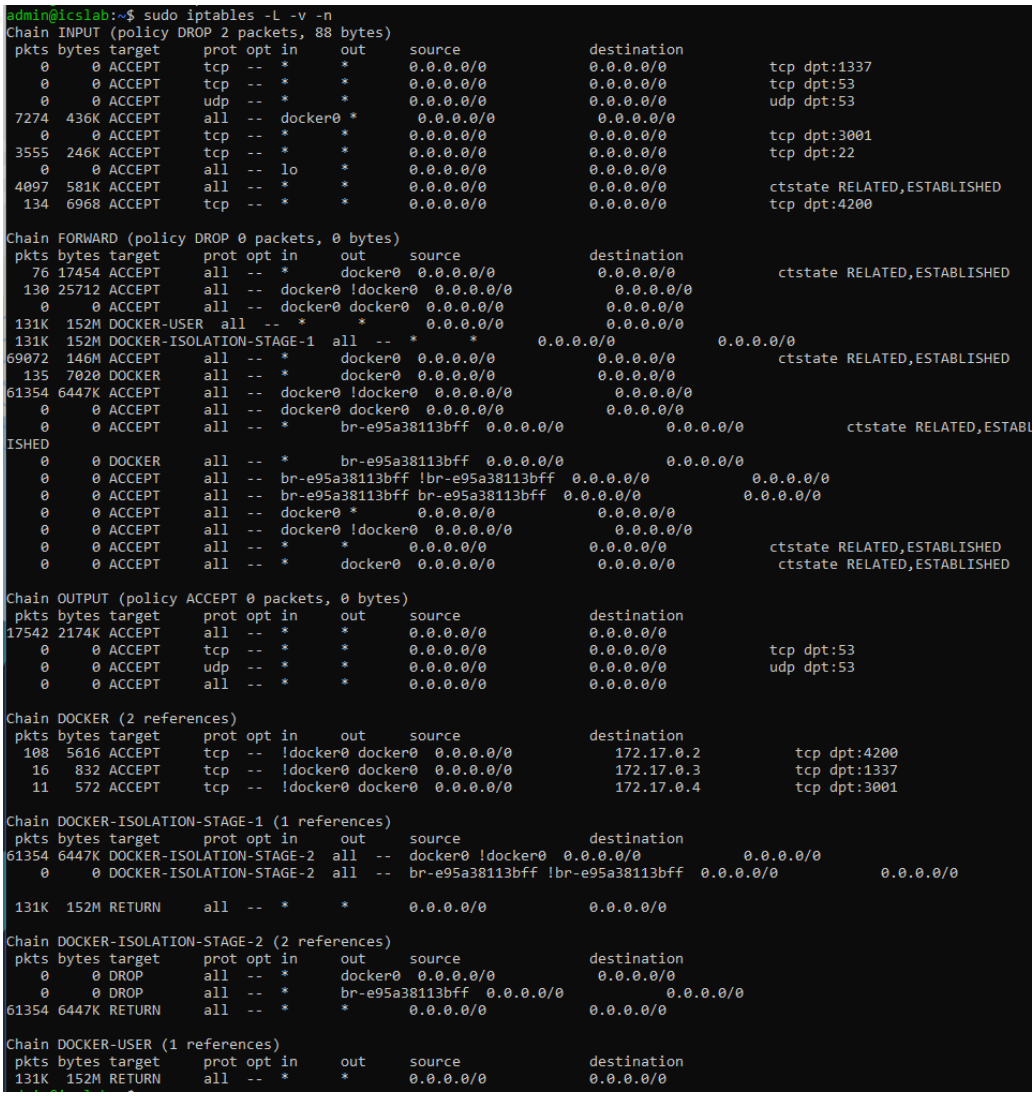

Рисунок 19 – Содержимое таблицы filter

Данные настройки позволят обезопасить систему от несанкционированного доступа и обеспечить бесперебойную работу системы.

2.3.2.2. Настройка таблицы nat

Для запуска веб-приложения также требуется настройка таблицы Nat.

С помощью данной таблицы будут преобразованы IP-адреса транзитных пакетов. Для переадресации входящих пакетов на 80 и 443порт проводим переадресацию на порт веб-приложения – 4200. Это можно сделать с помощью команд:

sudo iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 4200

sudo iptables -t nat -A PREROUTING -p tcp --dport 443 -j REDIRECT --to-port 4200

Далее необходимо провести маскировку IP-адреса компьютера. Внешний сетевой доступ оформляется с помощью использования команды – Маскарада: sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE. Это позволит опубликоваться нашему сайту на ресурс r-lab.ikit.sfu-kras.ru.

Конечная настройка таблицы nat отображена на рисунке 20

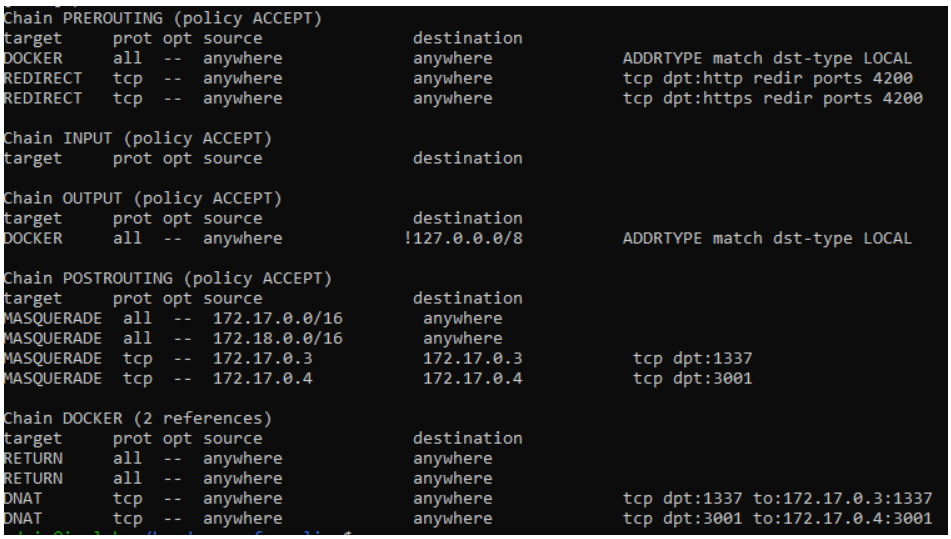

Рисунок 20 – Содержимое таблицы nat

2.3.3. Установка Certbot в контейнер и создание SSL-сертификата

Для установки Certbot на операционную систему, необходимо ввести следующие команды:

- sudo apt install software-properties-common
- sudo add-apt-repository ppa:certbot/certbot
- sudo apt update
- sudo apt install certbot

После установки можно будет произвести создание SSL-сертификата. Создать SSL-сертификат можно с помощью команды:

sudo certbot certonly --webroot -w /home/admin/ssl -d r-lab.ikit.sfu-kras.ru -d www.r-lab.ikit.sfu-kras.ru

С помощью данной команды будут получены SSL-сертификаты для системы удаленного доступа к лабораторному оборудованию. Далее требуется внести изменения в среду запуска проекта, а именно включить использование ssl с помощью флага --ssl true и взять выборку сертификатов из директории с помощью команд --ssl-cert и --ssl-key (рисунок 21):

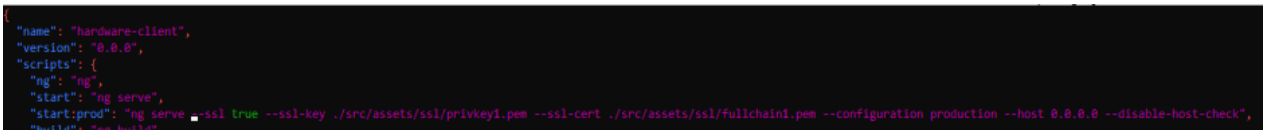

Рисунок 21 – Подключение SSL-сертификата в проект

После данной процедуры требуется провести повторную сборку контейнера с frontend приложением используя команду docker build.

Созданный SSL-сертификат необходим для защиты данных пользователей, и он обязателен в нашем случае. Данные действия требуются для исключения возможного перехвата учетных данных злоумышленниками, например человеком по середине (MITM-атака) или сниффер-атакой.

2.4. Выводы по главе 2

1. Сформированы контейнеры для взаимодействия готового веб-приложения между собой.

2. Произведена настройка Firewall на уровне операционной системы для корректного взаимодействия веб-приложения с поступающим трафиком

3. Произведена переадресация входящего трафика с использованием masquarade для возможности использовать доменное имя в сети интернет, исключив необходимость использования прямого адреса ресурса.

3 Тестирование системы удаленного доступа

Тестирование системы будет проводиться с помощью таких утилит как: nikto, chkrootkit, rkhunter, lynis. Данные утилиты позволяют найти различные уязвимости в системе и укажут на них для дальнейшей возможности их исправить.

Nikto – бесплатный сканер для поиска уязвимостей в веб-серверах. Утилита относится к классу blackbox сканер, т.е. сканеров, использующих стратегию сканирования методом черного ящика. Это значит, что заранее неизвестно о внутреннем устройстве программы или сайта (отсутствие доступа к исходному коду) и упор сделан на функциональность [34].

Rkhunter – это инструмент мониторинга безопасности для систем, совместимых с POSIX. Он сканирует на наличие руткитов и других возможных уязвимостей. Он делает это путем поиска каталогов по умолчанию (руткитов), неправильно настроенных разрешений, скрытых файлов, модулей ядра, содержащих подозрительные строки, и сравнения хэшей важных файлов с известными исправными [https://wiki.archlinux.org/title/Rkhunter].

Chkrootkit – это сканер поиска признака наличия «руткитов» в системе. Сканер может найти свыше 70 различных руткитов. Список руткитов, которые может обнаружить chkrootkit указан на официальном сайте разработчиков [35, 36].

Для проведения аудиты системы подойдет утилита Lynis.

Lynis – это инструмент для анализа безопасности системы для UNIX и macOS систем. Он выполняет обширное сканирование состояния системы для поддержки укрепления системы и проверки соответствия требованиям. Программа является открытым программным обеспечением с лицензией GPL.

Поскольку Lynis является гибким, он используется для нескольких различных целей, такие как:

- Аудит безопасности
- Тестирование на соответствие требования, такие как: PCI, HIPAA, Sox)
- Тестирование на проникновение
- Обнаружение уязвимостей
- Повышение надежности системы

Варианты использования Lynis:

- Разработчики – тестирование образа Docker или улучшение надежности развернутого веб-приложения
- Системные администраторы – ежедневное использование утилиты для обнаружения новых уязвимостей
- Тесты на проникновение – выявление слабых мест в системе безопасности, которые могут привести к компрометации данных или системы.

С помощью Lynis можно настроить проверку безопасности. Например, если проверка системы является слишком строгой для собственных целей, то можно её отключить и настроить как требуется пользователю [37].

3.1. Тестирование системы перед настройкой

Перед началом работы было необходимо провести тестирование системы на уязвимости. Проверка осуществлялась на адресе 10.3.3.20:4200.

Проверка с помощью утилиты Nikto не нашла критических проблем в работе системы (рисунок 22):

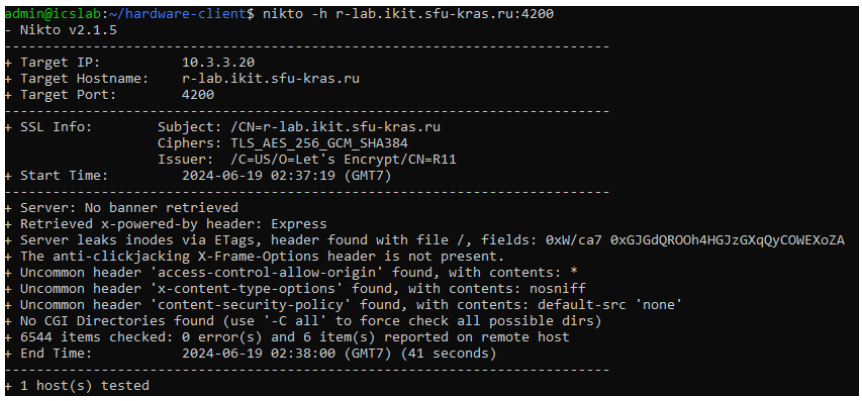

Рисунок 22 – Результаты проверки веб-приложения утилитой Nikto

Далее были проведены тестирования с помощью утилит chkrootkit и rkhunter. Во время проверки chkrootkit наличия руткитов в системе не было обнаружено, а rkhunter обнаружил 2 возможных руткина (рисунки 23, 24)

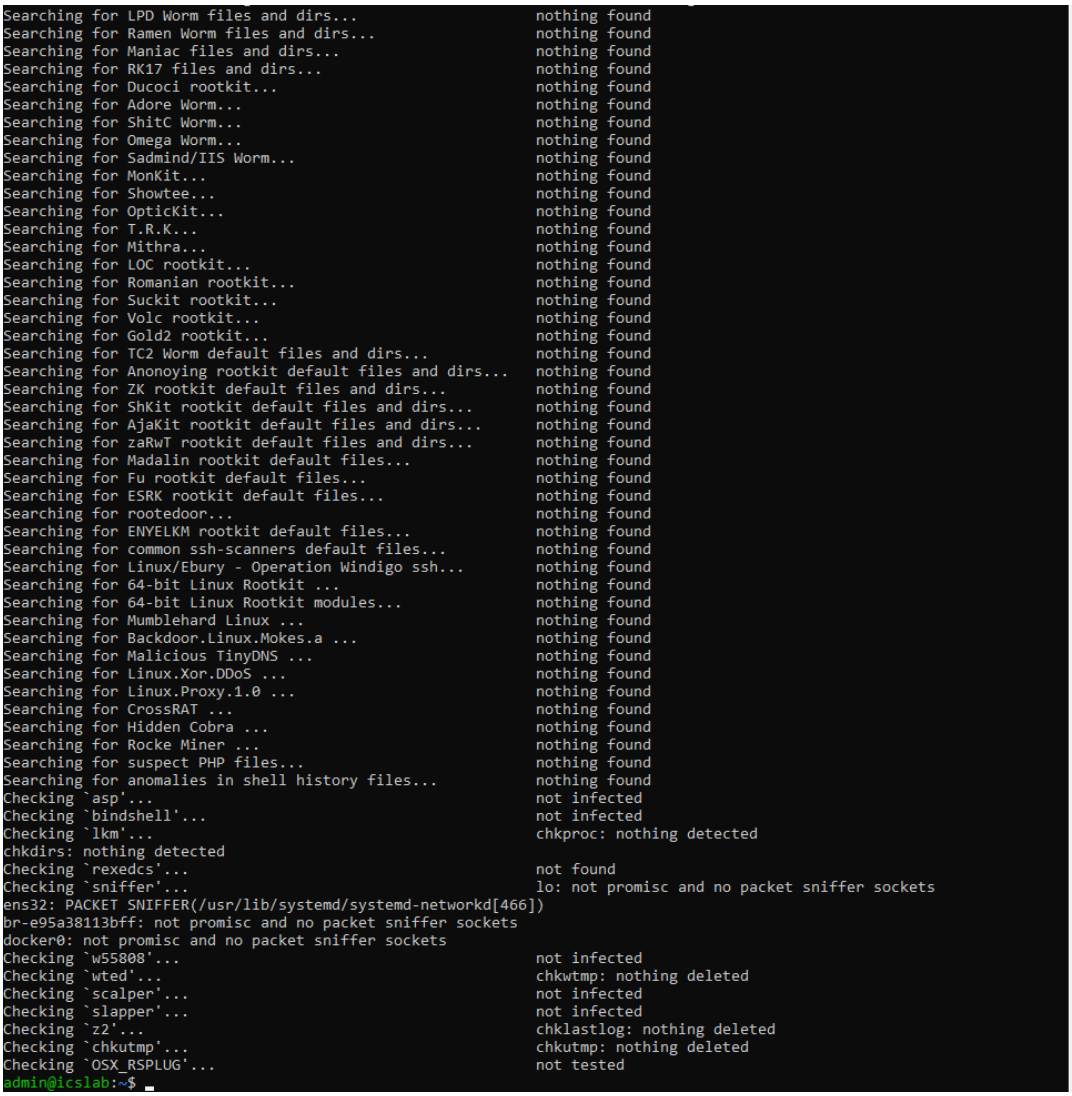

Рисунок 23 – Результаты проверки chkrootkit

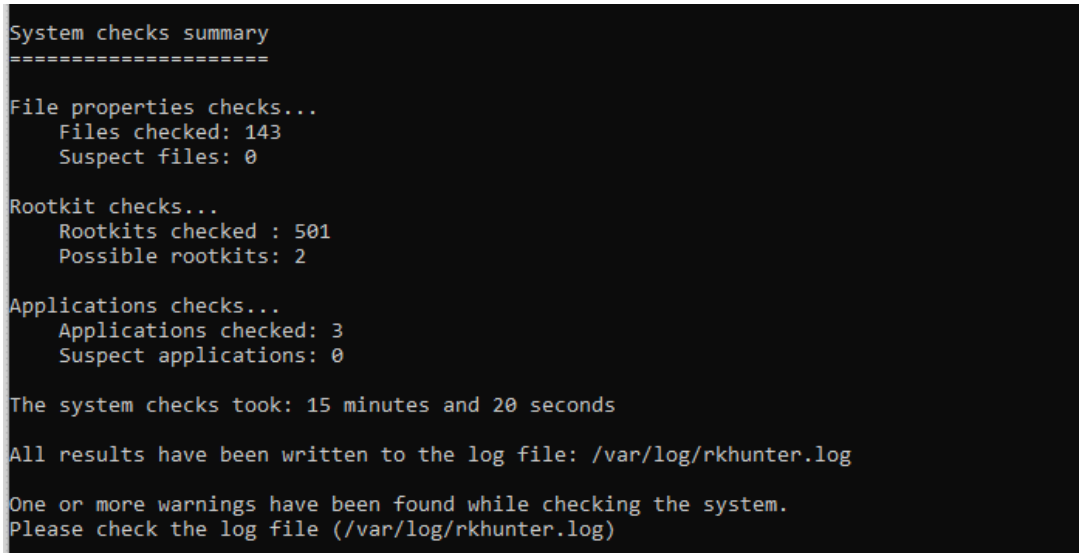

Рисунок 24 – Результаты проверки rkhunter

После требовалось провести проверку совместно с аудитом системы, в чем успешно справилась утилита Lynis (рисунок 25):

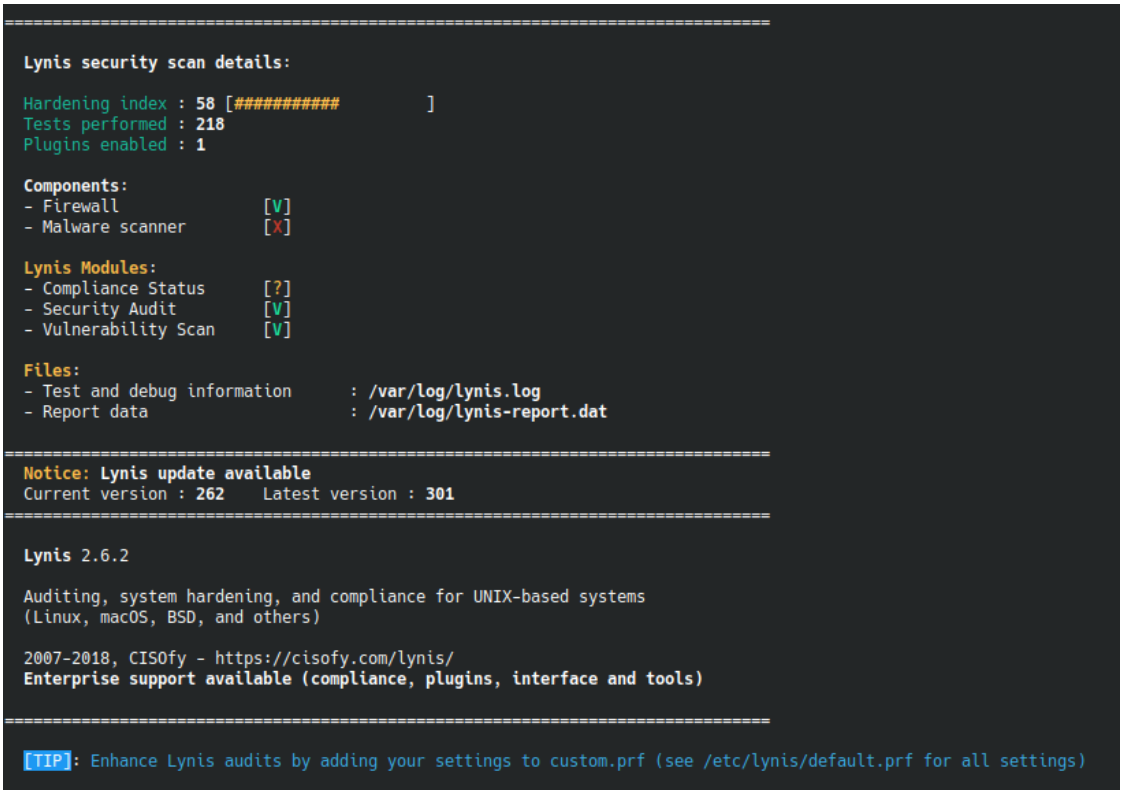

Рисунок 25 – Результаты проверки Lynis

Утилита указала на недостатки, что не установлен Антивирусный сканер для быстрого реагирования на возможное распространение вирусов при хакерских атаках.

3.2. Тестирование системы после настройки

Проведя контейнеризацию веб-приложений и настроив маршрутизацию пакетов, были проведены все тестирования повторно.

Первые три тестирования, которые использовались ранее, отобразились без каких-либо изменений, но утилита Lynis отобразила улучшение результатов, что появился Malware scanner (Рисунок 26).

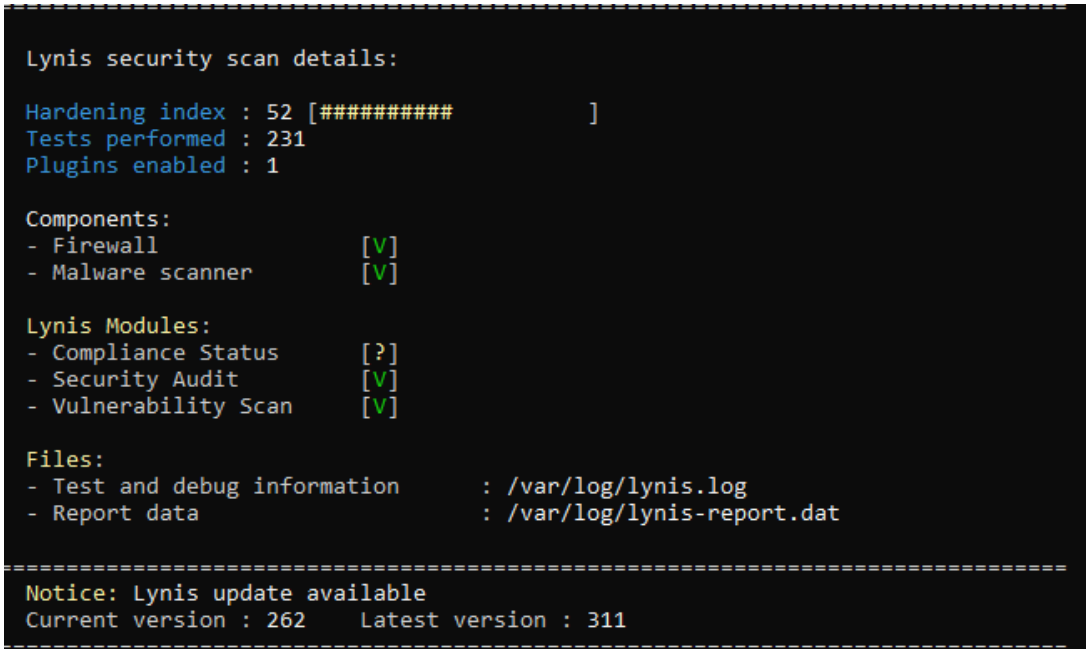

Рисунок 26 – Повторная проверка Lynis

3.3 Выводы по главе 3

Было проведено два этапа тестирования: тестирование до настройки безопасности и тестирование после настройки безопасности. Итоговые тесты показали положительные результаты. Разработанные правила управления трафиком и системой работают корректно. Имеется возможность пользоваться ресурсом беспрепятственно.

ЗАКЛЮЧЕНИЕ

В процессе реализации осуществления защищенности сервера были решены поставленные задания на ВКР. На начальном этапе были рассмотрены известные решения по удаленному доступу к лабораторному оборудованию, что позволило корректно оформить подключение программной и аппаратной части системы. Также были рассмотрены существующие способы атак. Это позволило определить какие атаки опасны для системы и предопределить способы защиты от определенных атак. Так же на основании анализа атак были выбраны утилиты Docker и iptables как основной способ защиты данных сервера. Это позволило перейти к разработке SSL-сертификата для сервера, чтобы защитить данные пользователей от перехвата учетных данных злоумышленниками. Полученные результаты второго этапа позволили перейти к тестированию системы и веб-приложения. Выполнялось тестирование на наличие уязвимостей и руткитов в системе. Указанные тестирования показали, что после произведения настроек защиты данных улучшили безопасность системы и сократили возможность внедриться к машине, что позволяет лабораторному оборудованию работать в стабильном режиме без опаски перехвата данных.

Таким образом, все поставленные задачи ВКР решены, что позволяет сделать вывод о достижении цели работы.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. 7 популярных программ для удалённого доступа к компьютеру | Медиа Нетологии: сайт. – URL: https://netology.ru/blog/01-2024-remote-access-software/ (дата обращения: 05.04.2024)
2. Блог компании СТЕК: сайт. – URL: https://stekspb.ru/blog/remote-desktop/ (дата обращения: 05.04.2024)
3. СОФТЛИСТ. ТОП-4 программ удаленного доступа к компьютеру в 2021 году: сайт. – URL: https://softlist.com.ua/articles/top-5-programm-udalennogo-dost/ (дата обращения: 05.04.2024)
4. Т. Лэммл, Ш. Одом, Р. Педжен. CCNP. Удаленный доступ. Учебное руководство. Лори, – 2018. – С. 412.
5. Протокол удаленного рабочего стола – Win32 apps | Microsoft Learn: сайт. – URL: https://learn.microsoft.com/ru-ru/windows/win32/termserv/remote-desktop-protocol (дата обращения: 07.04.2024).
6. Приложение для быстрого удаленного доступа — AnyDesk: сайт. – URL: https://anydesk.com/ru (дата обращения: 07.04.2024)
7. ТОП-15 бесплатных программ для удаленного доступа 2024 | Чем заменить TeamViewer?: сайт. – URL: https://amssoft.ru/amsblog/programmy-dlya-udalennogo-dostupa.php (дата обращения: 07.04.2024)
8. НИУ ВШЭ. Удаленный доступ к оборудованию УЛ САПР: сайт. – URL: https://miem.hse.ru/edu/ce/cadsystem/remote_access (дата обращения: 07.04.2024)
9. Лаборатория Электронных Средств Обучения (ЛЭСО) СибГУТИ: сайт. – URL: http://www.labfor.ru/articles/education/philosophy (дата обращения: 07.04.2024)
10. Oriel A. Herrera, Gustavo R. Alves, David Fuller, Roberto Aldunate. Remote Lab Experiments: Opening Possibilities for Distance Learning in Engineering Fields. IFIP World Computer Congress, TC 3 IFIP WCC TC3 2006:

Education for the 21st Century — Impact of ICT and Digital Resources p.p. 321-325.

11. The Hong Kong Polytechnic University. Department of Applied Physics. Remote Lab: сайт. – URL: https://remotelab.ap.polyu.edu.hk/ (дата обращения: 07.04.2024)

12. Javier García-Zubía. Remote Laboratories. Empowering STEM Education with Technology. Remote Laboratories, pp. i-xxiii (2021) – 268 с.

13. Северо-Западный межвузовский региональный учебно-научный центр "СПбПУ - ФЕСТО": сайт. _ URL: https://www.spbstu.ru/structure/educational-scientific-center-spbpu-festo/ (дата обращения: 07.04.2024)

14. Сетевые протоколы – Timeweb Cloud: сайт. – URL: https://timeweb.cloud/blog/setevye-protokoly (дата обращения: 15.04.2024)

15. Обзор протокола HTTP - HTTP | MDN: сайт. – URL: https://developer.mozilla.org/ru/docs/Web/HTTP/Overview (дата обращения: 15.04.2024)

16. Протокол TLS - Win32 apps | Microsoft Learn: сайт. – URL: https://learn.microsoft.com/ru-ru/windows/win32/secauthn/transport-layer-security-protocol (дата обращения: 15.04.2024)

17. TLS и SSL: Необходимый минимум знаний: сайт. – URL: https://mnorin.com/tls-ssl-neobhodimy-j-minimum-znanij.html (дата обращения: 15.04.2024)

18. 21 TCP_IP: сайт. – URL: https://studfile.net/preview/6071181/ (дата обращения: 15.04.2024)

19. Types of SQL Injection (SQLi) – GeeksforGeeks: сайт. – URL: https://www.geeksforgeeks.org/types-of-sql-injection-sqli/ (дата обращения: 15.04.2024)

20. Что такое сниффер? Способы защиты - Блог Kraden: сайт. URL: https://blog.kraden.com/ru/packet-sniffer (дата обращения: 15.04.2024)

21. Фишинговые атаки и письма - что это и как защититься: сайт. URL: https://www.kaspersky.ru/resource-center/preemptive-safety/phishing-prevention-tips (дата обращения: 15.04.2024)

22. Что такое атака на цепочку поставок?: сайт. URL: https://www.keepersecurity.com/ru_RU/threats/supply-chain-attack.html (дата обращения: 15.04.2024)

23. Что такое межсетевой экран, брандмауэр, файрвол? Какие типы сетевых экранов существуют и как работают: сайт. URL: https://www.kaspersky.ru/resource-center/definitions/firewall (дата обращения 15.04.2024)

24. Контейнеризация: основы и преимущества | Yandex Cloud – Документация: сайт. URL: https://yandex.cloud/ru/docs/glossary/containerization (дата обращения: 15.04.2024)

25. Что такое Docker и зачем он нужен, Docker Engine простыми словами, основы для начинающих, как пользоваться, как работает, введение в архитектуру, основные компоненты и понятия: сайт. – URL: https://cloud.croc.ru/blog/about-technologies/docker-vvedenie/ (дата обращения: 20.04.2024)

26. What is SELinux?: сайт. – URL: https://www.redhat.com/en/topics/linux/what-is-selinux (дата обращения: 20.04.2024)

27. Iptables | Русскоязычная документация по Ubuntu: сайт. – URL: https://help.ubuntu.ru/wiki/iptables (дата обращения 20.04.2024)

28. How It Works - Let's Encrypt: сайт. – URL: https://letsencrypt.org/how-it-works/ (дата обращения: 25.04.2024)

29. Certbot Instructions | Certbot: сайт. – URL: https://certbot.eff.org/instructions?ws=other&os=ubuntufocal (дата обращения: 25.04.2024)

30. What is Angular? • Angular: сайт. – URL: https://angular.dev/overview (дата обращения: 25.04.2024)

31. Welcome to the Strapi Developer Docs! | Strapi Documentation: сайт. – URL: https://docs.strapi.io/dev-docs/intro (дата обращения: 25.04.2024)

32. Documentation | NestJS - A progressive Node.js framework: сайт. – URL: https://docs.nestjs.com (дата обращения: 25.04.2024)

33. Install Docker Engine on Ubuntu | Docker Docs: сайт. – URL: https://docs.docker.com/engine/install/ubuntu/ (дата обращения: 28.04.2024)

34. Обзор сканера Nikto для поиска уязвимостей в веб-серверах: сайт. – URL: https://habr.com/ru/companies/first/articles/731696/ (дата обращения: 28.04.2024)

35. chkrootkit -- locally checks for signs of a rootkit: сайт. – URL: https://chkrootkit.org (дата обращения: 04.05.2024)

36. chkrootkit | Kali Linux Tools: сайт. – URL: https://www.kali.org/tools/chkrootkit/ (дата обращения: 04.05.2024)

37. Lynis - Security auditing tool for Linux, macOS, and Unix-based systems – CISOfy: сайт. – URL: https://cisofy.com/lynis/ (дата обращения: 04.05.2024)

Министерство науки и высшего образования РФ
Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра вычислительной техники

УТВЕРЖДАЮ
Заведующий кафедрой
_____ О. В. Непомнящий
подпись
«20» 06 2024 г

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Разработка системы контроля доступа и защищенных протоколов обмена к лабораторному оборудованию

09.04.01 – «Информатика и вычислительная техника»
09.04.01.11 «Вычислительные системы и сети»

Руководитель:	_____ подпись	19.06.24 дата	доцент, канд. техн. наук	С.Н. Титовский
Выпускник:	_____ подпись	19.06.24 дата		Данилович А.В.
Рецензент:	_____ подпись	20.06.24 дата	доцент, канд. техн. наук	Мазуров А. А.
Консультант	_____ подпись	20.06.24 дата	доцент, канд. техн. наук	К.В. Коршун
Нормоконтролер:	_____ подпись	19.06.24 дата	доцент, канд. техн. наук	С.Н. Титовский

Красноярск 2024