МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«САМАРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА С.П. КОРОЛЕВА»

Институт информатики, математики и электроники

Факультет информатики

Кафедра геоинформатики и информационной безопасности

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

**(специалист)**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

**РАЗРАБОТКА ЛАБОРАТОРНОГО СТЕНДА ДЛЯ МОДЕЛИРОВАНИЯ АТАК НА КАНАЛ ПЕРЕДАЧИ ДАННЫХ ПО ПРОТОКОЛУ MODBUS**

Выпускник Резванов Дмитрий Андреевич

(фамилия, имя, отчество)

Руководитель (Жмуров Д.Б.)

(фамилия,И.О.)

Рецензент ( ) (фамилия, И.О.)

Нормоконтролёр (Клевцова Е.В.)

(фамилия, И.О.)

****

**МИНОБРНАУКИ РОССИИ**

**федеральное государственное автономное**

**образовательное учреждение высшего образования**

**«самарский национальный исследовательский университет**

**имени академика с. п. королЕва»**

Кафедра геоинформатики и информационной безопасности

УТВЕРЖДАЮ

Заведующий кафедрой

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /В.В. Сергеев/

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_ г.

**Задание на выпускную квалификационную работу (ВКР)**

Студенту Резванову Д.А. 6511 С 317 группы

1. Тема ВКР: **«Разработка лабораторного стенда для моделирования атак на канал передачи данных по протоколу Modbus»** утверждена приказом по университету от «?» марта 2017 г. № 291-ст.

2. Исходные данные к ВКР: ?

3. Перечень вопросов, подлежащих разработке в ВКР:

3.1. Изучение вопросов информационной безопасности в АСУ ТП.

3.2. Изучение протокола Modbus, поиск уязвимостей в протоколе, разработка методов атаки.

3.3. Разработка аппаратного и программного обеспечения лабораторного стенда.

3.4. Проверка реализуемости атак на лабораторном стенде.

3.5. Разработка рекомендаций по повышению устойчивости систем АСУ ТП к атакам на каналы передачи данных.

4. Консультанты по разделам ВКР:

раздел ВКР:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

разрабатываемые вопросы: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

*должность, степень подпись И.О.Фамилия*

5. Дата выдачи задания: «?» марта 2017 г.

6. Срок представления на кафедру законченной ВКР: «?» июня 2017 г.

Руководитель ВКР

доцент кафедры ГИиИБ, к.т.н. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Д.Б. Жмуров/

*должность, степень подпись И.О.Фамилия*

Задание принял к исполнению \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Д.А. Резванов/

*подпись студента И.О.Фамилия студента*

**РЕФЕРАТ**

**Выпускная квалификационная работа специалиста:** ? c., ? рисунка, ? таблицы, ? источников.

Презентация: ? слайдов MicrosoftPowerPoint.

ИНФОРМАЦИЯ, АСУ ТП, MODBUS, STM32, МИКРОКОНТРОЛЛЕРЫ, RS-485, ПРОТОКОЛ ПЕРЕДАЧИ ДАННЫХ, UART, SPI.

Объектом разработки является лабораторный стенд для моделирования атак на канал передачи данных по протоколу Modbus на базе микроконтроллера семейства STM32.

Цель работы – анализ возможных направлений атак на протокол Modbus, разработка аппаратного и программного обеспечения лабораторного стенда, проверка реализуемости атак на лабораторном стенде.

Результатом данной работы является лабораторный стенд, который состоит из трёх устройств: ведущего, ведомого и злоумышленника, на котором можно демонстрировать атаки на канал передачи данных по протоколу Modbus.

СОДЕРЖАНИЕ

[ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ 7](#_Toc484426227)

[ВВЕДЕНИЕ 8](#_Toc484426228)

[1 ТЕХНОЛОГИИ ОБЕСПЕЧЕНИЯ ЗАЩИЩЕННОГО ДОКУМЕНТООБОРОТА 11](#_Toc484426229)

[1.1Понятие и структура защищенного документооборота 11](#_Toc484426230)

[1.2 Выбор программного средства 14](#_Toc484426231)

[2 ПРОГРАММНЫЙ КОМПЛЕКС VIPNET 16](#_Toc484426232)

[2.1Общие сведения 16](#_Toc484426233)

[2.2ViPNet Administrator 18](#_Toc484426235)

[2.2.1ViPNet Центр управления сетью (ЦУС) 19](#_Toc484426236)

[2.2.2ViPNet Удостоверяющий и ключевой центр (УКЦ) 20](#_Toc484426237)

[2.3 ПО ViPNetClient 22](#_Toc484426238)

[2.4ViPNetCoordinator 23](#_Toc484426239)

[2.4.1Назначение ПО ViPNetCoordinator 23](#_Toc484426240)

[2.4.2 Сервер-маршрутизатор 24](#_Toc484426241)

[2.4.3Маршрутизатор VPN-пакетов 25](#_Toc484426242)

[2.4.4Сервер IP-адресов 26](#_Toc484426243)

[2.4.5 Межсетевой экран 26](#_Toc484426244)

[3 ПРОЕКТИРОВАНИЕ ЗАЩИЩЕННОЙ СЕТИ 28](#_Toc484426245)

[3.1Описание вариантов развертывания 28](#_Toc484426246)

[3.2Проектирование топологии сети 31](#_Toc484426247)

[4КОНФИГУРИРОВАНИЕ ЗАЩИЩЕННОЙ СЕТИ 34](#_Toc484426248)

[4.1Первичная установка ПО на абонентские пункты 34](#_Toc484426249)

[5ОРГАНИЗАЦИЯ ЗАЩИЩЕННОГО КАНАЛА СВЯЗИ 38](#_Toc484426250)

[5.1 Формирование двух защищенных сетей 38](#_Toc484426251)

[5.1.1 Тестирования 42](#_Toc484426252)

[5.1.2 Удаление СУ из межсетевого взаимодействия 44](#_Toc484426253)

[5.2Создание иерархической структуры удостоверяющих центров 46](#_Toc484426279)

[5.2.1Основные понятия кросс-сертификации 46](#_Toc484426280)

[5.2.2Издание кросс –сертификата 47](#_Toc484426281)

[5.2.3Отзыв сертификата пользователя 49](#_Toc484426282)

[5.3Использование электронной подписи 50](#_Toc484426283)

[ЗАКЛЮЧЕНИЕ 52](#_Toc484426284)

[ПРИЛОЖЕНИЕ А Ключевая информация пользователей 54](#_Toc484426285)

# **ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ**

**АСУ ТП** – автоматизированная система управления технологическим процессом.

**ИС** – интегральная схема.

**МК** – микроконтроллер.

**ПК** – персональный компьютер.

**ОС –** операционная система.

**UART** – универсальный асинхронный приёмопередатчик.

**SPI** – последовательный периферийный интерфейс.

**IIC** – последовательная ассиметричная шина для связи между ИС внутри электронных приборов.

# **ВВЕДЕНИЕ**

## ПРОБЛЕМА БЕЗОПАСНОСТИ КАНАЛОВ ПЕРЕДАЧИ ДАННЫХ В АСУ ТП

* 1. **Понятие АСУ ТП**

Автоматизированная система управления технологическим процессом (АСУ ТП) представляет собой комплекс технических и программных средств, которые предназначены для управления технологическим оборудованием на промышленных предприятиях. Структура системы приведена на рисунке 1.1 [1].

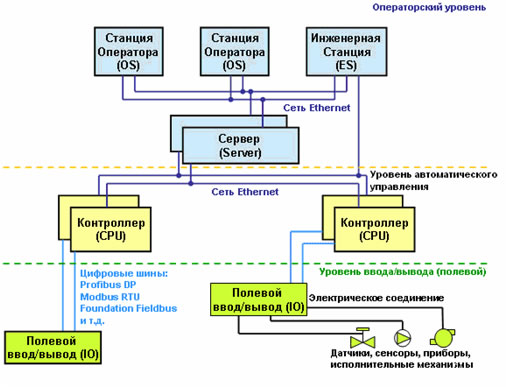


Рисунок 1.1 – структура АСУ ТП.

Как видно на рисунке, система делится на три уровня:

1. Нижний уровень (уровень ввода/вывода)
2. Средний уровень (уровень автоматического управления)
3. Верхний уровень (операторский уровень)

К нижнему уровню относится всё, чем управляют или с чего считывают показания. Например, электродвигатели, нагреватели, датчики температуры, давления и т.д. На рисунке 1.1 изображён пример исполнительного устройства.

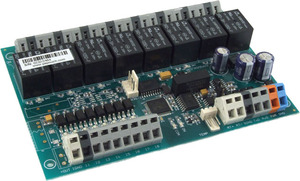


Рисунок 1.1 – релейный модуль с 8 релейными выходами и 8 логическими входами.

Средний уровень – это промышленные логические контроллеры (ПЛК). Это центр всей системы, именно он в зависимости от заложенной в него программы опрашивает датчики, управляет механизмами, а также собирает в процессе работы данные и отправляет их на верхний уровень. На рисунке 1.2 изображён пример ПЛК.



Рисунок 1.2 – программируемый логический контроллер Siemens Simatic S7-300 с тремя модулями ввода.

Верхний уровень – это то, что позволяет общаться контроллеру и человеку. Обычно это ПК с установленным пакетом программ для сбора, обработки и отображения информации. Такой программный пакет сокращённо называется SCADA. На рисунке 1.3 изображён интерфейс SCADA-системы КРУГ-2000.

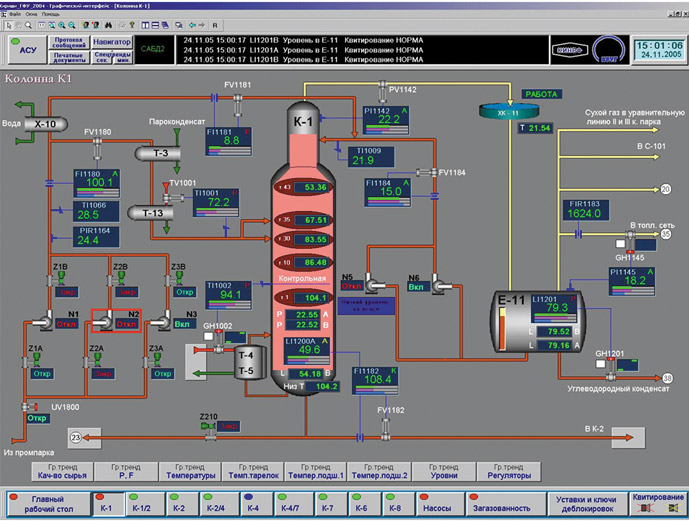


Рисунок 1.3 – интерфейс SCADA-системы КРУГ-2000. Установка газофракционирования

Для обмена между различными уровнями АСУ ТП используются каналы передачи данных. Между средним и верхним уровнем используется преимущественно сеть Ethernet, между средним и верхним – множество различных шин, таких как Modbus, Profibus, OPC и т.д.

* 1. **Актуальность проблемы безопасности АСУ ТП**

Так как в АСУ ТП широко используется сеть Ethernet, многие компоненты системы присутствуют в глобальной сети. [2] Чаще всего это SCADA-системы, на них приходится 70% компонентов АСУ ТП, которые доступны в глобальной сети. Ещё 27% - это программируемые логические устройства. Оставшиеся 3% - это остальные различные сетевые устройства (например, устройства нижнего уровня, которые используют сеть Ethernet). Соотношение компонентов АСУ ТП, которые присутствуют в глобальной сети, представлено на рисунке 1.4.

Рисунок 1.4 – соотношение компонентов АСУ ТП в глобальной сети

Как минимум 42% доступных через Интернет компонентов АСУ ТП содержат уязвимости. Ещё 41% находится в зоне риска. Таким образом, доля систем, безопасность которых достоверно подтверждена, составляет всего лишь 17%. Если смотреть статистику по странам, то наибольшая доля уязвимых систем, которые видны из сети Интернет, приходится на Швейцарию, там уязвимы все подобные системы. В России, например, уязвима ровно половина (50%) таких систем АСУ ТП.

Резкий рост интереса к проблемам безопасности АСУ ТП произошёл в 2010 году, когда был обнаружен компьютерный червь Stuxnet. Червь использовал четыре уязвимости системы Microsoft Windows, через которые распространялся первоначально через сменные носители, а затем по локальной сети, через Интернет и сеть Profibus. Незамеченным он оставался благодаря украденным настоящим цифровым подписям компаний Realtek и JMicron. Червь отличался высокой избирательностью, целенаправленно искал и заражал компьютеры с установленной SCADA-системой WinCC и системой программирования ПЛК Step 7. Вирус заражал файлы проекта, перехватывая на себя связь между WinCC и ПЛК при помощи подмены динамической библиотеки связи ПК и ПЛК на свою. Таким образом, червь может замаскировать своё присутствие и установить руткит непосредственно в ПЛК. Руткит атаковал только ПЛК Siemens Simatic S7-300, к которым были подключены частотные приводы только от двух поставщиков: Vacon и Fararo Paya. Кроме того, он атакует только те приводы, которые вращаются с частотой от 807 Гц до 1210 Гц, в этом диапазоне как раз вращаются газовые центрифуги для разделения изотопов. Когда все критерии выполнены, руткит изменяет скорость вращения до 1410 Гц, затем до 2 Гц, а затем до 1064 Гц, при этом системе сбора данных сообщается нормальное значение. Такие резкие колебания скорости вывели порядка 1000 центрифуг из строя. Из-за специфических требований к системе, существует предположение, что Stuxnet является специализированной разработкой спецслужб Израиля и США, направленной против ядерных проектов Ирана. [3]  
 Всё это привело к тому, что с 2010 по 2012 год было обнаружено в 20 раз больше уязвимостей в системах АСУ ТП, чем за период с 2005 по 2010 год.

Но при этом существует ещё один путь проникновения в систему, которому не уделяется достаточное внимание – через каналы связи нижнего уровня с верхним. Такой путь гораздо более трудоёмкий, так как в большинстве случаев требуется доступ непосредственно в охраняемый периметр предприятия, но от этого не менее опасный, ведь в случае с Stuxnet первичное заражение произошло через USB-носитель, что автоматически означает доступ к периметру на самой ранней стадии распространения. Атаки на данные каналы менее распространённые, но всё же встречаются. Например, в январе 2016 году на GitHub в открытый доступ был опубликован исходный код фреймворка для тестирования на проникновение под названием smod. Данный фреймворк задумывался для оценки безопасности протокола Modbus. В итоге это привело к тому, что за 2016 год количество атак на АСУ ТП, использующие протокол Modbus, выросло на 110% по сравнению с 2015 годом. [4]

Из протоколов передачи данных между нижним и верхним уровнями самыми распространёнными являются Modbus, Profibus и OPC. На рисунке 1.5 представлена распространённость данных протоколов. У Modbus (RTU+TCP) и Profibus примерно равные доли по 33%. Доля OPC составляет 25%, оставшиеся 9% приходятся на различные другие протоколы.

Рисунок 1.5 – распространённость промышленных протоколов передачи данных

Из двух лидеров рынка, Modbus и Profibus, первый более простой для понимания и реализации. В данной работе в качестве цели для анализа и моделирования атак будет рассматриваться последовательная (RTU) версия протокола Modbus, так как TCP-версия протокола использует в качестве среды передачи Ethernet и атаки на неё осуществляются теми же путями, что и на каналы связи между верхним и средним уровнями.

1. **ПРОТОКОЛ MODBUS**
   1. **История и общие сведения о протоколе**

Modbus – открытый коммуникационный протокол, базирующийся на архитектуре master-slave (ведущий-ведомый). Применяется в промышленности для связи между электронными устройствами через последовательные линии связи RS-232, RS-485, RS-422 и сети TCP/IP.

Протокол был разработан компанией Modicon (в настоящее время – Schneider Electric) для использования в своих программируемых логических контроллерах. Первая версия спецификации была опубликована в 1979 году. Стандарт протокола описывает лишь структуру обмена сообщениями, без привязки к конкретному физическому уровню. Но в большинстве случаев используются последовательные линии и сеть Ethernet.

Благодаря своей гибкости, но в то же время и простоте реализации на слабых вычислительных платформах (таких как микроконтроллеры), протокол Modbus получил крайне широкое распространение в сфере промышленной автоматики.

Спецификация протокола устанавливает три основных режима передачи данных: Modbus ASCII (American Standard Code for Information), Modbus RTU (Remote Terminal Unit) и Modbus TCP (Transmission Control Protocol). Первые два используются для линий последовательной связи, а последний – для сети TCP/IP. Основное отличие – в режиме Modbus TCP в сообщении отсутствует контрольная сумма, так как контроль целостности сообщения обеспечивается контрольной суммой в заголовке сегмента TCP.

В структуре сети допускается наличие только одного ведущего устройства и до 247 ведомых устройств. Каждое ведомое устройство имеет свой уникальный (в пределах конкретной сети) адрес от 1 до 247. Ведущее устройство не имеет адреса вообще. Адреса 248 – 255 зарезервированы, адрес 0 зарезервирован для широковещательной передачи сообщений. Сообщение с этим адресом должны обрабатывать все устройства, но отвечать на него не нужно. По спецификации протокола Modbus, инициировать передачу сообщения может только ведущее устройство, ведомое устройство делать этого не имеет права. Поэтому если ведомое устройство зафиксировало на своей стороне какое-либо событие (например, сработал концевой выключатель сигнализации), то ведущее устройство об этом не узнает до тех пор, пока само не отправит сообщение с запросом статуса этого события.

Сообщения, исходящие от ведущего устройства, называются запросами, а ответные сообщения от ведомого – ответами. Обмен ведётся в полудуплексном режиме, это значит, что в любой момент времени передачу ведёт только одно устройство.

* 1. **Физический уровень RS-485**

В качестве физического уровня для передачи сообщений в лабораторном стенде используется RS-485, как наиболее распространённый в промышленности.

RS-485 – стандарт физического уровня для асинхронного интерфейса (полное название ANSI TIA/EIA-485-A:1998). В качестве физической среды для передачи и приёма данных используется одна витая пара, иногда с экранирующей оплёткой или общим проводом. Соответственно, используемая сетевая топология – «общая шина», структура которой показана на рисунке 2.1.

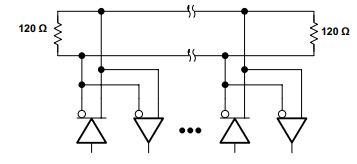


Рисунок 2.1 – структура топологии «общая шина».

По стандарту, максимальная длина линии может составлять 1200 метров. При такой большой длине возникают эффекты длинных линий, так как длинную витую пару можно представить как распределённую ёмкость и индуктивность. Так как витая пара на всей своей протяжённости имеет одинаковую конструкцию, следовательно, и одинаковые емкостные и индуктивные свойства на единицу длины, то данное свойство характеризуется таким параметром как волновое сопротивление. Поэтому, чтобы ослабить отражение сигнала от концов витой пары, в начале и в конце линии между проводниками пары устанавливается так называемый терминатор – согласующий резистор, сопротивление которого равно волновому сопротивлению витой пары. В большинстве случаев используются резисторы номиналом 120 Ом. На рисунке 2.2 представлены осциллограммы сигнала на стороне приёмника, в обоих случаях использовался приёмопередатчик ADM3485, витая пара с волновым сопротивлением 120 Ом и резистор-терминатор номиналом 120 Ом со стороны передатчика. Сигнал слева снят с резистором-терминатором номиналом 54 Ом со стороны приёмника, сигнал справа – с резистором номиналом 120 Ом.

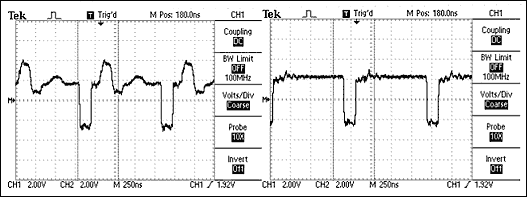


Рисунок 2.2 – осциллограммы сигнала на стороне приёмника.

Как видно на осциллограммах, в случае установки резистора-терминатора неверного номинала возникает рассогласование, которое становится причиной отражений сигнала. В результате корректный приём становится невозможным.

Режим передачи – дифференциальный. Это означает, что за логический 0 или логическую 1 принимается не напряжение относительно общего провода, а напряжение между проводниками пары. Сделано это в первую очередь для защиты от внешних помех. Так как проводники пары скручены между собой, то внешняя помеха будет наведена на оба проводника с одинаковым знаком, амплитудой и фазой. Такая помеха называется синфазной. В итоге, разница напряжений между линиями пары останется неизменной и помеха не окажет никакого влияния на передаваемые данные. Линии пары обозначаются как A - неинвертирующая линия, B инвертирующая линия и С – необязательный общий провод. За логическую 1 принимается положительное напряжение (A-B) > +200мВ, за логический 0 – отрицательное напряжение (A-B) < -200мВ. Состояние, когда напряжение меньше 200мВ, стандартом не определено. При этом стандарт ограничивает допустимый диапазон напряжений линий A и B относительно общего провода диапазоном от -7В до +12В.

Из-за того, что любой реальный, а не теоретический, приёмник и передатчик имеют конечный импеданс, максимальное количество устройств в одной линии ограничено. Для облегчения проектирования сети RS-485 была введена гипотетическая единица нагрузки, так называемая «единица нагрузки». По стандарту, максимальная нагрузочная способность линии – 32 единицы нагрузки. Количество единиц нагрузки указывается в технической документации приёмопередатчика. Например, приёмопередатчик ADM3485 нагружает линию как 1 единичная нагрузка, а MAX487 – как ¼ единичной нагрузки. Для данного конкретного примера, если вся линия будет состоять из однотипных приёмопередатчиков, то всего в линию можно будет включить 32 устройства с ADM3485 и 128 устройств с MAX487. Для обхода ограничения на максимальное количество устройств в линии может применяться специальное устройство – повторитель интерфейса RS-485. Повторитель считывает сигнал с одной линии и передаёт его в другую и наоборот. Помимо этого, некоторые повторители могут обладать такой полезной функцией как гальваническая развязка – сигнал передаётся из одной линии в другую без непосредственного электрического контакта. В итоге, цепи оказываются электрически разделёнными, что ограничивает передачу энергии из одной цепи в другую, но возможна передача сигнала.

* 1. **Протокол Modbus RTU**

Режим RTU (Remote Terminal Unit) является самым наиболее часто применяемым режимом Modbus для последовательных сетей. Главное преимущество RTU перед ASCII – компактность. В режиме ASCII каждый байт данных кодируется двумя байтами. Например, для передачи байта 0x45 в режиме ASCII необходимо передать два байта, ASCII-код символов ‘4’ и ‘5’. Однако, есть и преимущество – для устройств, которые не умеют отсчитывать малые временные интервалы, ASCII-режим становится единственно возможным, так как конец пакета определяется не по интервалу тишины на линии, а по последовательности CR+LF. В промышленности датчики и исполнительные устройства реализуются, в основном, на микроконтроллерах. Так как большинство микроконтроллеров имеют в своём составе минимум один аппаратный таймер, данное преимущество ASCII-режима не имеет особого значения, на первый план выходит потребление оперативной памяти. В этом плане Modbus RTU выигрывает у Modbus ASCII примерно в два раза.

Общая структура сообщения Modbus RTU приведена на рисунке 2.3.



Рисунок 2.3 – структура сообщения Modbus RTU.

Адрес – адрес ведомого устройства, которому предназначен запрос.

Функция – поле длиной 1 байт, которое говорит ведомому, какое именно действие от него требуется.

Данные – информация, необходимая ведомому устройству для выполнения заданной ведущим устройством функции или данные, передаваемые ведомым в ответ. Формат данного поля зависит от конкретной функции.

Контрольная сумма – поле длиной 2 байта, контрольная сумма формата CRC16-Modbus.

Максимальная длина сообщения ограничена 256 байтами.

Спецификация протокола определяет четыре типа данных:

1. Флаги – 1 бит, чтение и запись
2. Дискретные входы – 1 бит, только чтение
3. Регистры хранения – 16 бит, чтение и запись
4. Регистры ввода – 16 бит, только чтение

Флаги – 1-битная структура данных, доступная для чтения и записи. Основное применение - управление устройствами, которые могут находится только в двух состояниях, такие как электромагнитное реле или светодиод.

Дискретные входы – 1-битная структура данных, доступная только для чтения. Основное применение – внешние датчики, которые могут находится в двух состояниях, такие как концевые выключатели.

Регистры хранения – 16-битная структура данных, доступная для чтения и записи. Основное применение – различные настройки ведомого устройства, такие как адрес, скорость обмена и т.д.

Регистры ввода – 16-битная структура данных, доступная только для чтения. Основное применение – хранение показаний различных датчиков, например, датчик температуры, влажности, давления и т.д.

У каждого элемента есть адрес, адресация начинается с 0. Адресация внутри своей группы уникальна. Например, регистр хранения с адресом 1 и регистр ввода с тем же адресом – это разные регистры. Соответственно, и доступ к ним осуществляется с помощью разных кодов функций.

Рассмотрим коды функций. Они делятся на 5 групп, первые три группы называются стандартными:

1. Чтение данных
2. Запись одного значения
3. Запись нескольких значений
4. Пользовательские функции
5. Зарезервированные функции

Для пользовательских кодов функций выделены два диапазона: 0x41 – 0x48, 0x64 – 0x6E. При этом не гарантируется, что какое-то другое устройство не будет использовать тот же код для выполнения другой функции.

В категорию зарезервированных попадают коды, которые не являются стандартными, но уже используются устройствами, которые производят другие различные компании. Это коды 0x09, 0x0A, 0x0D, 0x0E, 0x29, 0x2A, 0x5A, 0x5B, 0x7D, 0x7E, 0x7F.

Чтение данных осуществляется с помощью следующих кодов функций:

0x01 – чтение значений из нескольких регистров флагов

0x02 – чтение значений из нескольких дискретных входов

0x03 – чтение значений из нескольких регистров хранения

0x04 – чтение значений из нескольких регистров ввода

Запрос состоит из адреса первого элемента и количества, которое необходимо прочитать. Так как длина пакета данных ограничена, максимальное количество входов, которое можно прочитать за один раз, ограничено 2000. Максимальное количество регистров, доступное для чтения за один раз – 125. Структура запроса приведена в таблице 2.1.

Таблица 2.1 – структура запроса на чтение данных

|  |  |  |
| --- | --- | --- |
| Код функции | 1 байт | 0x01, 0x02, 0x03, 0x04 |
| Начальный адрес | 2 байта | 0x0000 – 0xFFFF |
| Количество входов для чтения | 2 байта | Для 0x01, 0x02 – от 1 до 2000 (0x07D0). Для 0x03,0x04 – от 1 до 125 (0x007D) |

В ответе передаются запрошенные данные. Значения регистров хранения и дискретных входов передаются по два байта на один регистр, порядок байтов MSB (Most Signification Byte) – то есть, старший байт идёт первым. Флаги и дискретные входы передаются в упакованном виде, один флаг или вход занимает 1 бит. Единица – включённое состояние, ноль – выключенное. Младший бит первого байта содержит значение первого запрошенного флага или входа, второй бит – второго и т.д. Первый бит второго байта содержит значение девятого флага или входа. Если запрошенное количество не кратно 8 – лишние биты заполняются нулями.

Структура ответа показана в таблице 2.2

Таблица 2.2 – структура ответа на запрос чтения данных.

|  |  |  |
| --- | --- | --- |
| Код функции | 1 байт | 0x01, 0x02, 0x03, 0x04 |
| Количество байт | 1 байт | Для 0x01, 0x02:  N = количество выходов/8. Если остаток > 0, то N = N+1  Для 0x03, 0x04:  N = 2\*n, где n – количество регистров |
| Данные | N байт |  |

Запись одного значения осуществляется с помощью следующих кодов функций:

0x05 – запись значения одного флага

0x06 – запись значения в один регистр хранения

Запрос состоит из адреса элемента и устанавливаемого значения. Для флага значение 0xFF00 означает включенное состояние, 0x0000 – выключенное. В случае записи других значений – ведомое устройство сообщит об ошибке. В случае успешного выполнения команды – ведомое устройство вернёт копию запроса. Структура запроса представлена в таблице 2.3.

Таблица 2.3 – структура запроса записи одного значения

|  |  |  |
| --- | --- | --- |
| Код функции | 1 байт | 0x05, 0x06 |
| Адрес флага или регистра | 2 байта | 0x0000 – 0xFFFF |
| Записываемые данные | 2 байта |  |

Запись нескольких значений осуществляется с помощью следующих кодов функций:

0x0F – запись значений нескольких флагов

0x10 – запись значений в несколько регистров хранения

Запрос состоит из адреса первого элемента, количества записываемых элементов, количества передаваемых байт и самих данных. Упаковка данных полностью аналогична упаковке данных в командах чтения. Ответ состоит из начального адреса и количества записанных элементов. Структура запроса представлена в таблице 2.4, структура ответа в таблице 21.5.

Таблица 2.4 – структура запроса записи нескольких значений

|  |  |  |
| --- | --- | --- |
| Код функции | 1 байт | 0x0F, 0x10 |
| Начальный адрес | 2 байта | 0x0000 – 0xFFFF |
| Количество записываемых элементов | 2 байта | Для 0x0F – от 1 до 2000 (0x07D0). Для 0x10 – от 1 до 123 (0x007B) |
| Количество байт | 1 байт | Для 0x0F:  N = количество выходов/8. Если остаток > 0, то N = N+1  Для 0x10:  N = 2\*n, где n – количество регистров |
| Данные | N байт |  |

Таблица 2.5 – структура ответа на запрос записи нескольких значений

|  |  |  |
| --- | --- | --- |
| Код функции | 1 байт | 0x0F, 0x10 |
| Начальный адрес | 2 байта | 0x0000 – 0xFFFF |
| Количество записываемых элементов | 2 байта | Для 0x0F – от 1 до 2000 (0x07D0). Для 0x10 – от 1 до 123 (0x007B) |

В результате обмена могут возникать некоторые ошибки. Ошибки могут быть связаны с искажениями при передаче данных или в логике сообщений (когда запрос принят верно, но не может быть выполнен).

Ошибки первого типа обнаруживаются при вычислении контрольной суммы. Если вычисленная сумма не совпадает с указанной в сообщении – сообщение игнорируется. Для предотвращения зависания системы в целом, ведущее устройство должно отсчитывать определённый таймаут. По истечении таймаута, устройство может сделать вывод, что ведомое устройство не присутствует на линии, либо произошли искажения при передаче.

При ошибках второго типа ведомое устройство посылает ведущему специальное сообщение об ошибке. Признаком того, что текущее сообщение сообщает об ошибке, является установленный в единицу старший бит кода функции. Следующий за кодом функции байт сообщает о типе ошибки. Стандартные коды ошибок перечислены в таблице 2.6.

Таблица 2.6 – список стандартных кодов ошибок

|  |  |  |
| --- | --- | --- |
| Код ошибки | Расшифровка | Примечание |
| 0x01 | Принятый код функции не поддерживается устройством |  |
| 0x02 | Адрес данных, указанный в запросе, недоступен |  |
| 0x03 | Недопустимое значение в поле данных | Данная ошибка будет вызываться, если, например, для функции записи значения одного флага (0x05) указать значение, отличное от 0xFF00 или 0x0000 |
| 0x04 | Невосстанавливаемая ошибка |  |
| 0x05 | Ведомое устройство приняло запрос, но на выполнение требуется время | Данная ошибка генерируется для того, чтобы ведомое устройство не генерировало ошибку таймаута ответа |
| 0x06 | Ведомое устройство занято | При получении данной ошибки ведущее устройство должно повторить свой запрос позднее |

* 1. **Анализ возможных уязвимостей протокола Modbus**

На рисунке 1.4 показана временная диаграмма одного сеанса обмена (запрос ведущего + ответ ведомого) в идеальном случае, когда ведомому не требуется время на обработку). t3.5 – интервал тишины, после которого пакет считается принятым.



Рисунок 1.4 – временная диаграмма обмена в идеальном случае

Но указанная выше ситуация в реальной системе не будет наблюдаться никогда по причине того, что ведомому необходимо время на проверку контрольной суммы, разбор принятого пакета, опрос датчиков\входов, формирование ответа и т.д. Поэтому реальная картина будет соответствовать той, что показана на рисунке 1.5. tобработки – время, необходимое ведомому на обработку запроса. Пример: ведомое устройство имеет в своей основе 32-разрядный микроконтроллер STM32 с ядром Cortex-M3, работающим на частоте 72МГц, обмен ведётся на скорости 9600. Обработка запроса на запись 17 регистров флагов занимает ? мкс, что соответствует ? символов на данной скорости обмена. В случае использования 8 и 16-разрядных архитектур это время может быть больше в несколько раз.

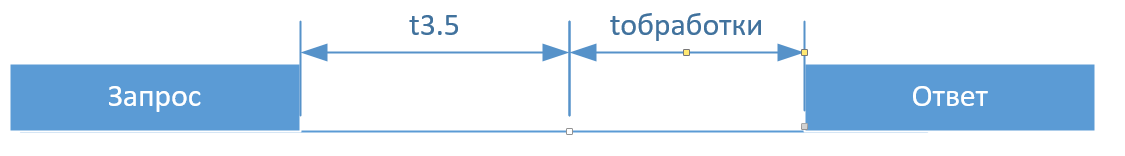


Рисунок 1.5 – временная диаграмма обмена в реальной системе

Далее необходимо проанализировать, как можно вмешаться в работу системы. Так как физическая среда передачи – витая пара, способ только один – физически подсоединиться к линии. Если линия имеет большую протяжённость (по стандарту до 1200 метров, но с повторителями может быть и гораздо больше), то поиск вторгшегося устройства становится очень непростой задачей. Далее необходимо понять, как именно вмешиваться в обмен.

Первый, самый очевидный способ, который не требует сложной разработки – постоянно отправлять в линию случайные данные. В данном случае, периодически будет возникать ситуация, когда ведущее или ведомое устройство выставляет на линии один логический уровень, а злоумышленник – другой. В результате во время передачи этого бита разница напряжений между линиями A и B будет примерно в районе 0 В, что соответствует неопределённому состоянию. В итоге, обмен непременно будет нарушен. Также, этот способ атаки можно рассматривать как атаку на переполнение буфера. Поступающие данные хранятся в буфере, который располагается в оперативной памяти ведущего или ведомого. Если в ПО устройства отсутствует проверка на предмет того, не превысило ли принятое количество байт максимальную длину пакета, то после 256 байт произойдёт переполнение буфера, так как устройство злоумышленника отправляет данные непрерывно, без интервалов тишины, предусмотренных по стандарту. Защитой от этого может служить прекращение приёма после 256 принятых байт, либо использование 8-битной переменной для хранения количества принятых байт. Во втором случае произойдёт переполнение переменной и отсчёт количества принятых байт начнётся с нуля. Временная диаграмма обмена для этого случая показана на рисунке 1.7.

Второй способ основан на использовании интервала t3.5+tобработки, называется «ложный ведомый». Действительно, если нам примерно известно среднее время обработки запроса настоящим ведомым, то можно взять микроконтроллер с производительностью сильно превосходящей производительность ведомого и отправить ложный ответ в то время, пока настоящий ведомый будет обрабатывать запрос. Ответ же настоящего ведомого будет отброшен ведущим, так как он уже принял ответ от ложного ведомого. Временная диаграмма обмена для этого случая показана на рисунке 1.6. t1 – время обработки запроса ложного ведомого.



Рисунок 1.6 – временная диаграмма для атаки «ложный ведомый»

Данный тип атаки является самым опасным с точки зрения технологического процесса. Допустим, ведомое устройство – поплавок с концевым выключателем, который контролирует наполнение какой-либо ёмкости. Ведущее устройство постоянно опрашивает состояние выключателя и в зависимости от него выдаёт команды на насос. Если в данном случае осуществить атаку «ложный ведомый», то можно заставить ведущего думать, что концевой выключатель не сработал. Тогда ведущее устройство будет продолжать выдавать команды на насос, что при отсутствии защиты от перелива будет означать разлив жидкости.

Третий способ основан на том, что запросы в реальной линии идут не непрерывно, а периодически. Между сеансами обмена между ведущим и ведомым случаются длинные временные интервалы, в течении которых можно успеть отправить запрос ведомому и получить от него ответ. Соответственно, способ атаки называется «ложный ведущий». Временная диаграмма данной атаки приведена на рисунке 1.7.

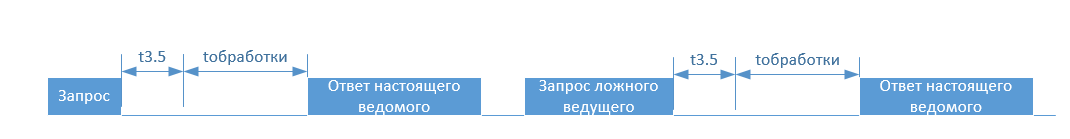


Рисунок 1.6 – временная диаграмма для атаки «ложный ведущий»

Опасность данной атаки состоит в том, что если точно известен тип ведомого устройства и оно позволяет изменять настройки обмена (например, скорость) через регистры хранения, то ложный ведущий может перезаписать эти самые настройки. В результате, настоящий ведущий уже не сможет установить связь с ведомым при следующем сеансе обмена.

Для моделирования данных атак был разработан лабораторный стенд.

# **3 РАЗРАБОТКА ЛАБОРАТОРНОГО СТЕНДА**

## Описание вариантов развертывания

Программный комплекс ViPNetCustom позволяет, по общедоступным каналам связи различного типа, создать среду безопасного обмена информацией. Создаются логические контуры сети,защищенных криптографическими средствами высокой надежности. Контуры сетей бывают двух типов:

* сеть ViPNet для организации электронного документооборота (частный вариант);
* сеть ViPNet со всеми функциями VPN.

Первый вид контура сети ViPNet позволяет организовать систему электронного документооборота со следующим основным функционалом:

* конфиденциальный обмен информацией;
* юридическая значимость документов;
* гибкая настройка автоматической обработки исходящих и входящих документов (файлов);
* возможность плавного расширения сети до уровня VPN.

Типовая схема сети ViPNet (электронный документооборот) представлена на рисунке 6.

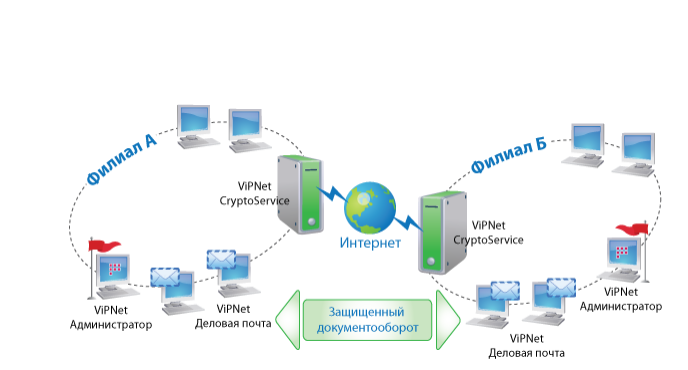


Рисунок 6 - Схема сети ViPNet-электронный документооборот

Второй тип контура сети ViPNet легко преобразуется из первого путем добавления соответствующих лицензий и установкой нужного ПО (рисунок 7). Этот вид контура позволяет развертывать частные виртуальные сети любых конфигураций, обеспечивающих прозрачное взаимодействие компьютеров сети ViPNet независимо от способа, места и типа IP-адреса при их подключении к сети. При этом весь трафик, циркулирующий по виртуальному контуру этой сети, защищен криптографическими методами [6].

Основные преимущества и функционал сети ViPNet (контур частная виртуальная сеть):

* плавная интеграция в структуру существующей сети;
* гибкая настройка фильтрации закрытого (защищенного средствами ViPNet) и открытого трафика;
* широкие средства внутренней и внешней коммуникации (почта, чат, обмен файлами);
* организация юридически значимого электронного документооборота;
* система слежения за состоянием сети;
* дополнительный функционал по регистрации пользователей, публикации сертификатов в общедоступных хранилищах.

Построение сетей ViPNet на базе частных виртуальных сетей — процесс индивидуальный и дифференцированный. В большей части он зависит от текущей топологии сети и от тех коммуникационных задач, которые необходимо решить.Исходя из этого универсальную схему ViPNet сети изобразить сложно.

Важно отметить, что место администратора может быть разделено на ЦУС и УКЦ. Типовым является когда ЦУС и УКЦ устанавливаются совместно на компьютер, выделенный под рабочее место администратора сети ViPNet.

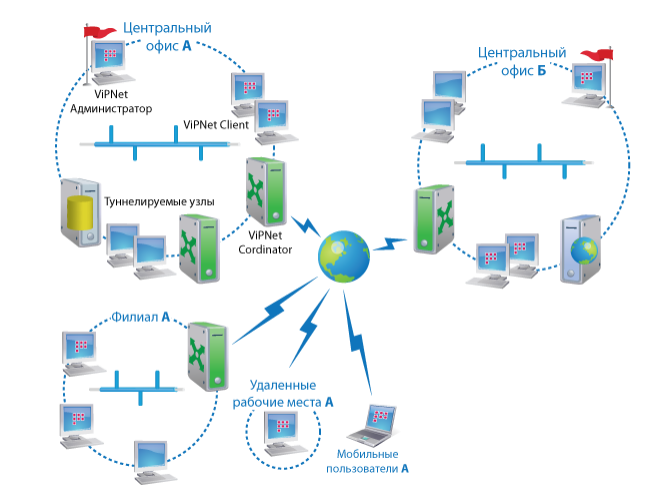


Рисунок 7- Схема сети на базе программного комплекса ViPNetCustom

Использование технологии ViPnet позволяет объединить офис компании «А», филиалы, удаленных пользователей и мобильных пользователей в одну защищенную сеть. Можно организовать защищенное взаимодействие сети компании «А» с сетью компании «Б», если в сетях используются технологииViPNet.

Развертывание контура сети ViPNet для организации электронного документооборота обеспечивает защиту только следующих видов информации: почтовые сообщения и файлы (вложения к сообщениям).Вся остальная информация, передающаяся по сети, останется незащищенной (Рисунок 8).

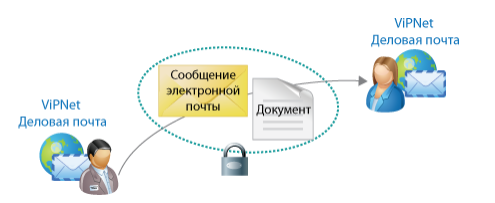


Рисунок 8 - Защита документов и сообщений в сети ViPNet.

При организации данного вида ViPNet сети нужно определить круг пользователей, которые участвуют в электронном документообороте, и разместить в сети сервер, который выполняет роль координатора.

## 3.2 Проектирование топологии сети

Реализация взаимодействия между двумя сетями с помощью ViPNetобеспечивает защищенный документооборот. Имеется сервер в лабораторном классе университета. Для создания комплекса по изучению технологии ViPNet,установлены виртуальные машины в приложении VirtualBox. В сетях включены все основные компоненты программного продукта ViPNet.Используется линейка продуктов CUSTOM:

* ViPNetAdministrator.
* ViPNetRegistrationPoint.
* ViPNetCoordinator (Windows).
* ViPNetCoordinator (Linux).
* ViPNetFailover (Linux).
* ViPNetClient.

На рисунке 9 представлена схема развертываемой учебной сети. На ней видно, что у нас есть две сети. В каждой сети присутствует администратор. Задачей администраторов сети ViPNetявляется управление сетью,работа удостоверяющим центром. В каждой сети есть два координатора (СМ Офис, СМ Филиал), между ними есть межсерверный канал. Предварительно на машины устанавливается операционные системы, в скобках на рисунке 9 указаны имена компьютеров в виртуальной среде. Установка незащищенных компьютеров выполнялось для того, чтобы было понимание того, что не вся сеть физическая может быть защищена и практические задачи этого не требуют.ViPNetклиент устанавливается на машины где необходимо передача информации по защищенному каналу.

Задача стоит в том, что создать сети по отдельности и установить межсетевое взаимодействия для использования защищенного документооборота, выдачи сертификатов .

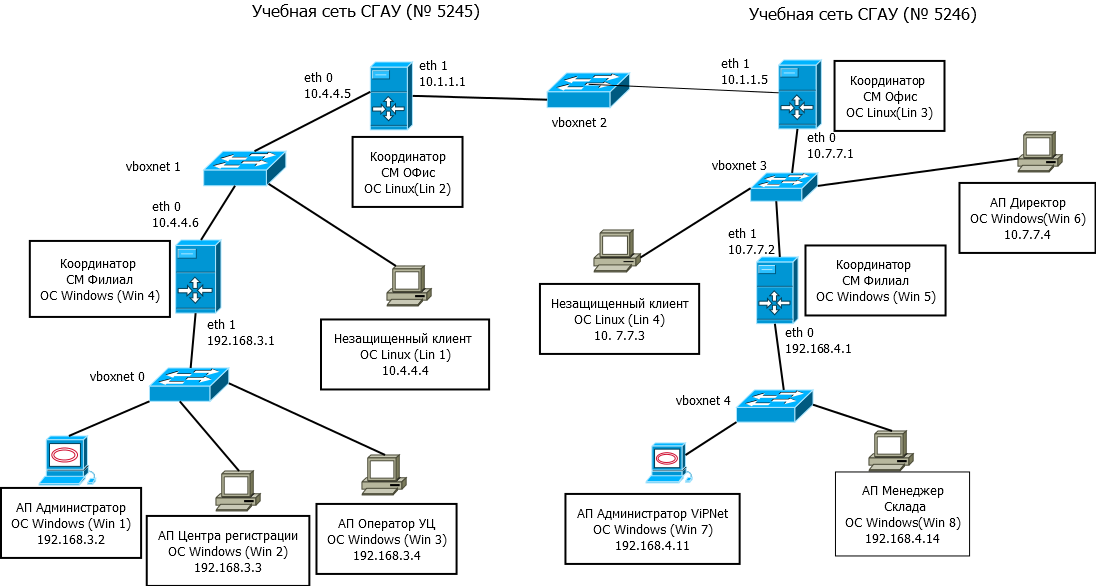
****

Рисунок 9-Схема развертываемой сети

# **4КОНФИГУРИРОВАНИЕ ЗАЩИЩЕННОЙ СЕТИ**

**4.1Первичная установка ПО на абонентские пункты**

Для реализации этой схемы мы первоначально устанавливаем виртуальные машины с интерфейсами и настройками как показано на рисунке10.

У нас в сети будут две отдельные независимые друг от друга сети. Установки программного ПО аналогично для каждой сети. Первоначально на компьютере администратора устанавливается ViPNetAdministrator и ViPNetClient. После установки заходим в ЦУС не меняя путей в окне настроек. Настройки указаны на рисунке 10 .В ЦУСе формируется структура нашей сети.

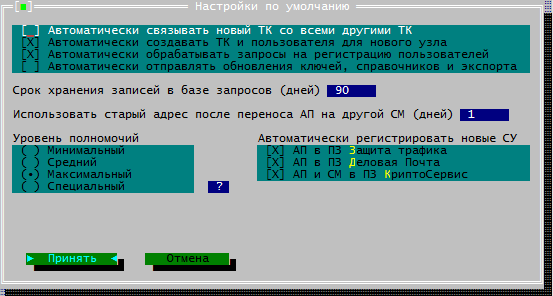


Рисунок 10 -Настройки по умолчанию в ЦУСе

Формирование адресной структуры сети ViPNet состоит из следующих этапов:

1. Регистрация Серверов-маршрутизаторов.
2. Установка межсерверного канала связи.
3. Регистрация Абонентских пунктов (АП) на серверах-маршрутизаторах.
4. Процедура проведения маршрутизации.

В Администрации сетевого уровня настраиваем нужную конфигурацию сети (рисунок 11).

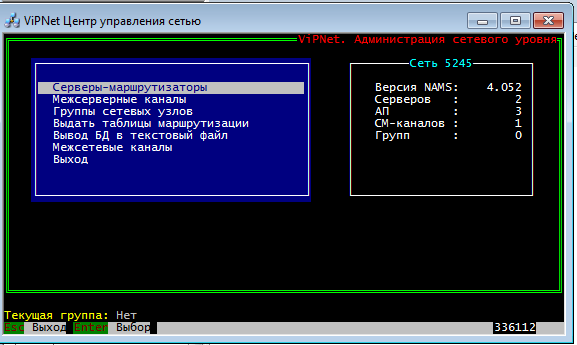


Рисунок 11-Меню Администрации сетевого уровня

На рисунке 12 показано связь Сервера маршрутизатора «СМ ФИЛИАЛ» абонентскими пунктами.

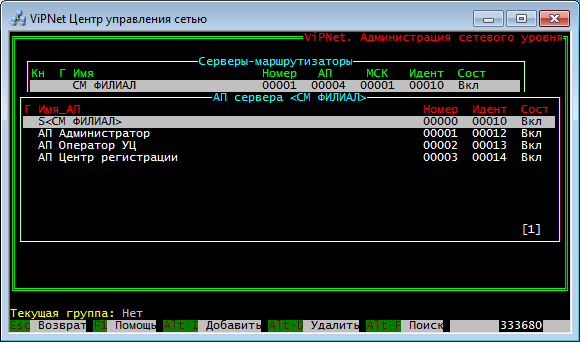


Рисунок 12- Окно Администрации сетевого уровня

У «СМ ОФИС» нету АП, он только соединен с незащищенным компьютером и связан с «СМ Филиал» межсерверным каналом. После регистрации серверов-маршрутизаторов, установки межсерверных каналов и регистрации АП, проводим маршрутизацию.

Регистрация типов коллективов (ТК) и связь между ними является очень важным моментом в формировании структуры защищенной сети, так как:

* Если между ТК имеется связь, то между пользователями, которые зарегистрированы в этих коллективах, имеется защищенный канал связи.
* Если нет связи между типами коллективов , то нет и защищенного канала связи.

В соответствии с условием создания защищенной сети необходимо полна связь всех пользователей сети. После этого необходимо зарегистрировать пользователей. Все имена представлены на рисунке 9.Важнейшим шагом является регистрация АП в прикладных задачах от этого зависит какими функциями мы наделим пользователей. Если все сформированоправильно,то при проверки на аномальные ситуации программа выдаст отрицательный результат.

Заключительным этапом работы в ЦУСе является формирование справочников. Программаформирует:

* Справочники АП.
* Таблицы маршрутизации.
* Справочники для УКЦ.

При первом запуске УКЦ открывается окно Мастера первичной установки.Мастер инициализации создает начальную ключевую информацию для работы программы УКЦ:

* персональный ключ защиты администратора;
* пароль администратора;
* ключ защиты КЦ;
* ключ подписи и сертификат администратора;
* мастер-ключи;

После прохождения этапа первичной инициализации и осуществление первичных настроек необходимо сформировать дистрибутивы ключей (dst-файл (смотреть в приложение А) ) для пользователей защищенной сети При создании дистрибутива ключей автоматически формируется его сертификат электронной подписи (ЭП)[1].Дистрибутивы следует перенести в папку.Администратор УКЦ доверенным путем передает пользователю три компонента:

1. Дистрибутивы ключей.
2. Резервный набор персональных ключей.
3. Пароль для дистрибутива.

Развертывание защищенного узла сети будет происходить при помощи программы ViPNetClient Монитор. Проходим первичную инициализацию. Мастер запросит у нас \*.dst (дистрибутива(смотреть приложение А)) пользователя, выбираем нужный и вводим пароль.

Для проверки правильности в программе ViPNetClientМонитор должны появиться защищенные узлы (рисунок 13)

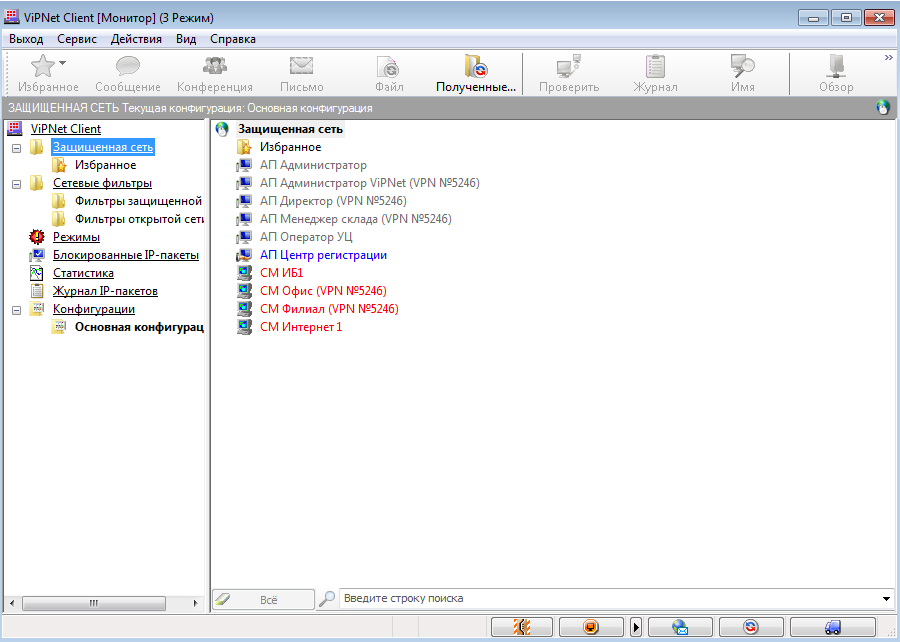


Рисунок 13-Защищенная сеть в ViPNetClient

Так для каждого узла сети устанавливаем нужное ПО.Проделываем те же действия и для другой сети , в соответствии со схемой (рисунок 9).

# **Типовые учебные задачи лабораторного комплекса**

## 5.1 Формирование двух защищенных сетей

Задачей данного раздела является формирование двух защищенных сетей, для безопасного обмена документами.Для построения межсетевого взаимодействия будет использоваться ИСММК, потому что ключ создается для двух защищенных сетей. Для создания защищенного документооборота необходимо проделать действия:

Шаг 1. Создание архива и формирование начального экспорта в сети №5245.

Перед началом создаем архив конфигураций сети. Формируем начальный экспорт. На экспорт передаем сетевые узлы и соответствующие типы коллективов сети №5245.Это нужно ,чтобы организовать связи этих коллективов сети №5245 с коллективами сети № 5246 и сформировать ключи шифрования.

Шаг 2.Задание шлюза в сети №5245.

Необходимо определить в обеих сетях СМ ,который будет являться шлюзом. В нашей сети это СМ Офис.

Шаг 3. ЦУС сети № 5245- копирование начального экспорта.

При копировании начального экспорта, задается каталог назначения (по умолчанию это папка ..\NEW\EXPORT).На этом этапе подготовка начального экспорта в ЦУСе сети №5245 завершена.

Шаг 4. Генерация ИСММК для сети № 5246.

Создаем ИСММК сети №5245 для второй сети и подготавливаем его на экспорт. Для этого в УКЦ создаем мастер ключ. Тип ключа задаем индивидуальный, а также внешнюю сеть №5246.

Шаг 5 ЭКСПОРТ ИСММК для сети № 5246.

Подготавливаем созданный индивидуальный мастер- ключ на экспорт для другой сети. При экспортировании нужно задать пароль защиты ИСММК. После этого вводим его в действие .

Шаг 6. Передача начального экспорта для чети № 5246 Администратору.

Копируем все составляющие начального экспорта в одну папку. Это справочники, ключ межсетевого взаимодействия.Теперь всю папку нужно передать администратору сети № 5246, по надежному каналу передачи данных. На этом этапе работа в сети №5245 завершена.

Шаг 7. Размещение начального экспорта в сети № 5246.

После принятие Администратором сети №5246 начального экспорта он должен их разместить в папках импорта ЦУСа и УКЦ сети.

Шаг 8.Создание архива и межсетевого взаимодействия в сети №5246.

Для приема начального экспорта при запуске ЦУС появится сообщение о принятых файлов для УКЦ.Далее следуя подсказкам нажимаем клавиши. Требуется создать архив конфигураций. Теперь требуется в окне Администрация сетевого уровня выбрать межсетевой канал и включить его. Теперь следует провести маршрутизацию.

Шаг 9.Установка связей ТК сетей.

Теперь приустановки связей ТК у нас появились коллективы и первой сети (рисунок 14), связываем каждый с каждым.

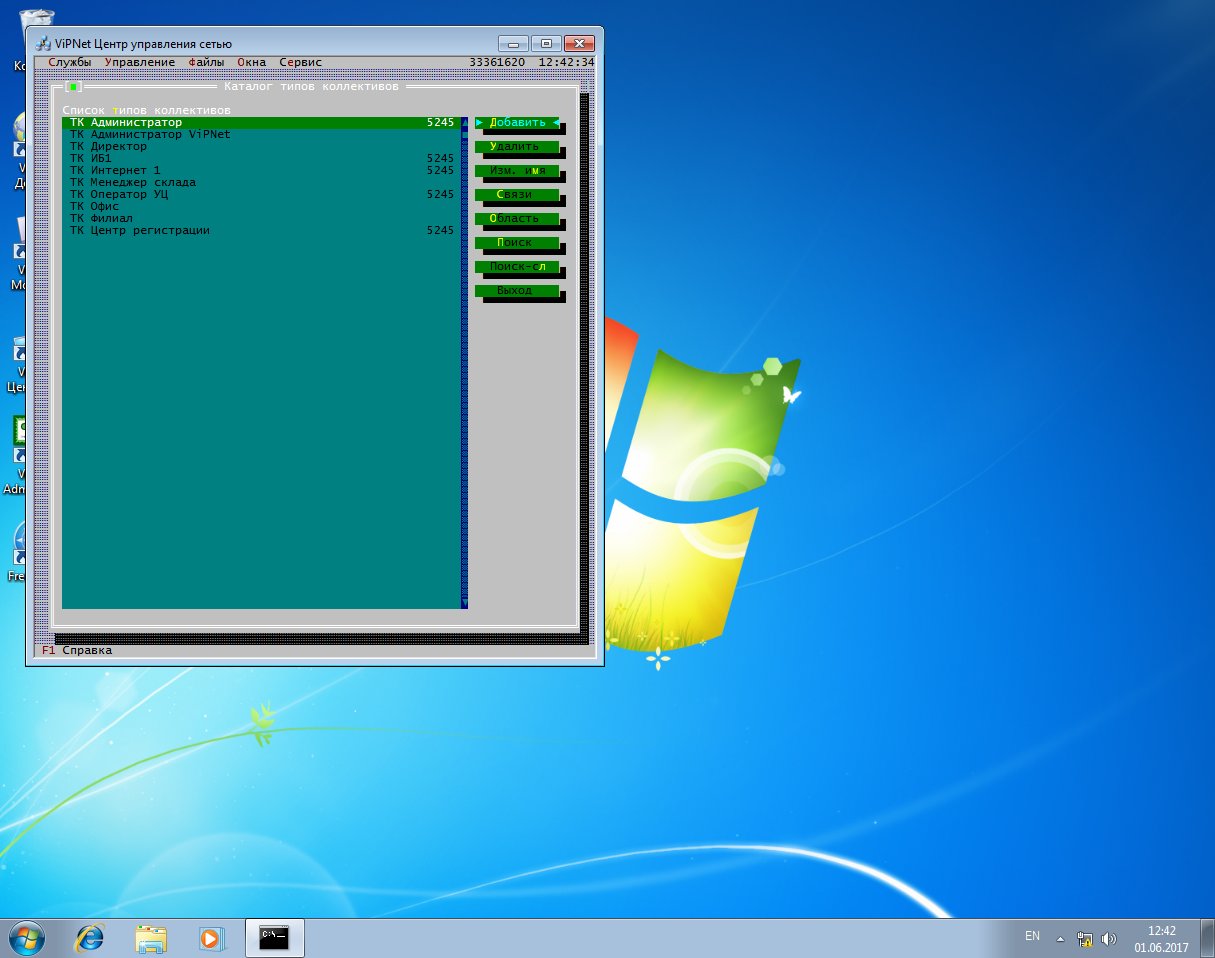


Рисунок 14 – Списки ТК после экспорта узлов сети 5245

Шаг 10.Формирование ответного экспорта в сети № 5246 для сети №5246.

Формируем ответный экспорт, выбираем состав ответного экспорта. Добавляем все ТК и все узлы на экспорт.

Шаг 11. Задание СМ-шлюза и копирование ответного экспорта.

Задаем СМ-шлюз, в данном случае у нас СМ ОФИС. Копируем ответный экспорт в папку по умолчанию. Теперь нужно провести импорт информации начального экспорта в сети № 5246.

Шаг 12. Проверка на наличия аномальных ситуаций и формирование справочников в сети №5246.

Для начала убедимся, что вся информация о структуре была введена правильно. Для этого проверим конфигурацию сети. Если аномальных ситуаций нет, то сделано все правильно. После этого формируем справочники.

Шаг 13. Импорт сертификатов ЭП администраторов сети № 5245 в УКЦ сети № 5246.

Переходим в УКЦ сети №5246. Необходимо обработать принятые справочники сертификатов ЭП администраторов сети №5245. Для этого заходим в папку (доверенные сети ViPNet>Входящие). Выделяем сертификат и импортируем.

Шаг 14. Импорт ИСММК из сети №5245.

Импортируем созданный ИСММК для формирования межсетевого взаимодействия. Выбираем раздел КЦ>Доверенные сети>Мастер ключи>входящие. В контекстном меню нажимаем импорт. Нужно вести пароль который вводил администратор сети №5245. В разделе текущем вводим его в действие. Если прошло все правильно вылезет окно на рисунке15.

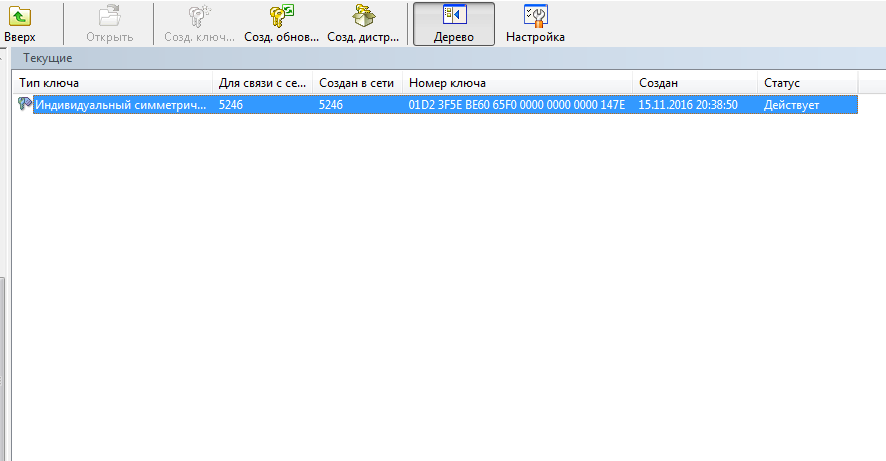


Рисунок 15-Информация об ИСММК

Шаг 15. Формирование новых КУ в УКЦ сети № 5246 и перенос их в ЦУС.

Только после того как ИСММК импортирован, создаем новую ключевую информацию для узлов данной сети. Теперь рассылаем обновления.

Шаг 16. Процедура импорта ответного экспорта в сети №5245.

Аналогично принимаем импорт из сети №5246, как из сети № 5245.

Шаг17.Подключение межсетевого канала в ЦУС сети № 5245.

Запускаем ЦУС и принимаем новые файлы .Настраиваем межсетевой канал. СМ шлюз, в нашем случае СМ Офис. После изменений в структуре необходимо провести маршрутизацию.

Шаг 17. Проверка наличия связей между ТК сетей № 5245 и № 5246.

Необходимо проверить все ли сделано правильно. Выбираем пункт регистрация типов коллективов. Здесь отражен список ТК (рисунок 16).

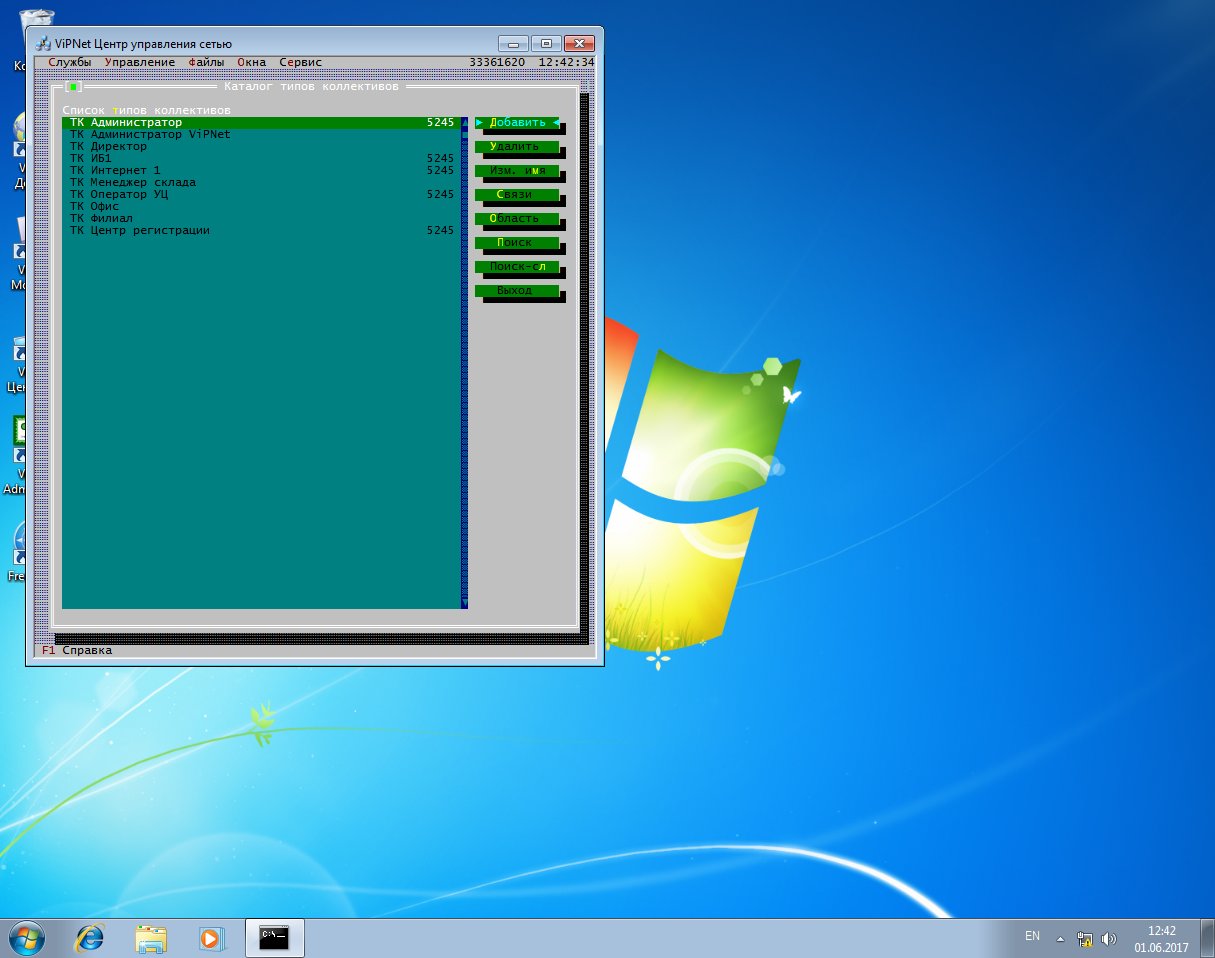


Рисунок 16 – Списки ТК после экспорта узлов сеть 5246

Шаг 18. Формирование справочников и проверка на аномальные ситуации.

Проверяем на аномальные ситуации, если все в порядке сформировываем справочники для УКЦ.

Шаг 19.Импорт сертификатов ЭП второй сети в УКЦ первой.

Нам осталось совершить импорт сертификата из доверенной сети, ввести его в действие.

Шаг 20. Формирование новых ключевых узлов в УКЦ сети №5245.

Необходимо создать КУ и перенести их в ЦУС .Выходим из УКЦ.

Шаг 21.Рассылка обновлений КУ и адресных справочников.

После этого нужно проверить правильность взаимодействия.

### 5.1.1 Тестирование

Для проверки работы сети заходим в ViPNetClient Монитор теперь после настройки межсетевого взаимодействия в защищенных сетях у нас появится пользователи из другой сети. Для того чтобы убедиться связь есть запускаем «Деловую почту» и отправляем письмо на АП (Рисунок 15).

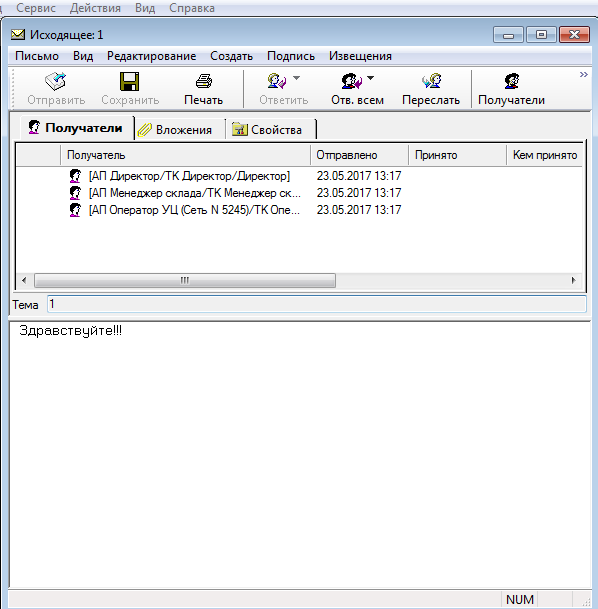


Рисунок 16- Отправка сообщения в ПО «Деловая почта»

Для того чтобы убедиться что все работает открывает деловую почту кому

мы отправили .

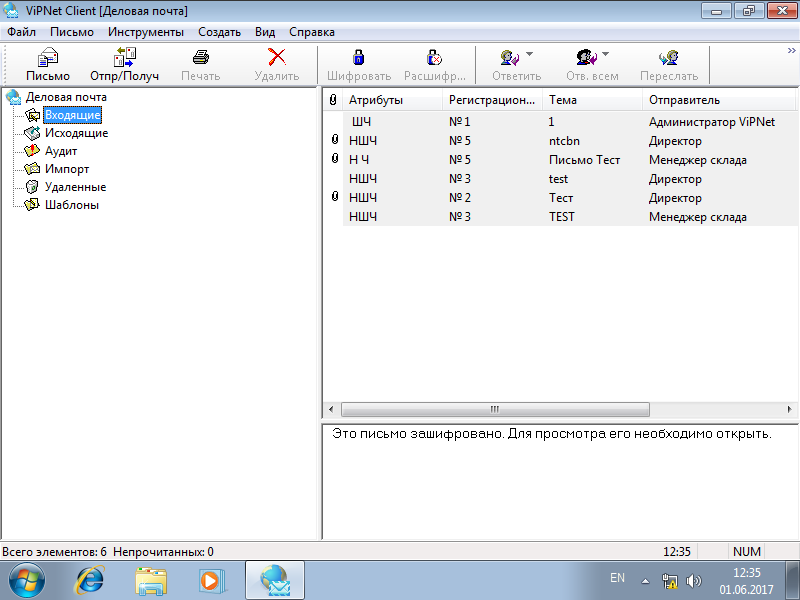


Рисунок 17- Деловая почта Оператор УЦ(входящие сообщения)

На рисунке 17 показано , что Оператор УЦ принял несколько сообщений. Исходя из этого делаем вывод, что сеть работает правильно.

### 5.1.2 Удаление СУ из межсетевого взаимодействия

Имеем межсетевое взаимодействие сетей №5245 и №5246.В сети №5246 имеется узел АП Менеджер Склада (Менеджер Склада),на котором зарегистрирован один ТК Менеджер Склада и работает один пользователь, который участвует в межсетевом взаимодействии сети №5246 cсетью. №5245.

Предположим, что сотрудника уволили и нам нужно удалить узел из межсетевого взаимодействия.

Необходимые действия для этого:

Шаг 1.Удаление АП Менеджер Склада из сети №5246. Формирование начального экспорта сети №5246 в сеть №5245.

Для удаления нужно зайти в ЦУС сети №5246 и в администрации сетевого уровня удалить его (рисунок 18).

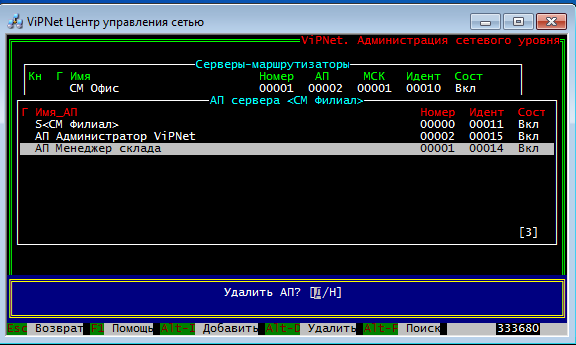


Рисунок 18 – Удаление узла АП менеджер склада

После этого формируем начальный экспорт для отправки в сеть №5245.

Шаг 2. Обрабатываем импорт в ЦУСе сети № 5245 (рисунок 19).

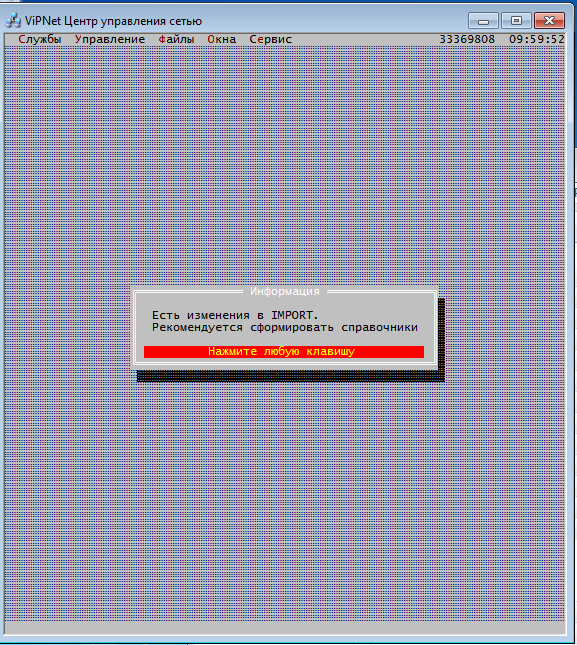


Рисунок 19- Окно диалога обработки импорта

Шаг 3.Формируем справочники. Создадутся новые справочники АП, таблицы маршрутизации, экспорт(в сеть № 5246), информация для УКЦ.

Шаг 4.Сформированные файлы отправляем на экспорт в сеть № 5246.После приема экспорта на узлах сети № 5245 в программе ViPNetClient Монитор в окне защищенная сеть отсутствует АП Менеджер Склада (рисунок 20).

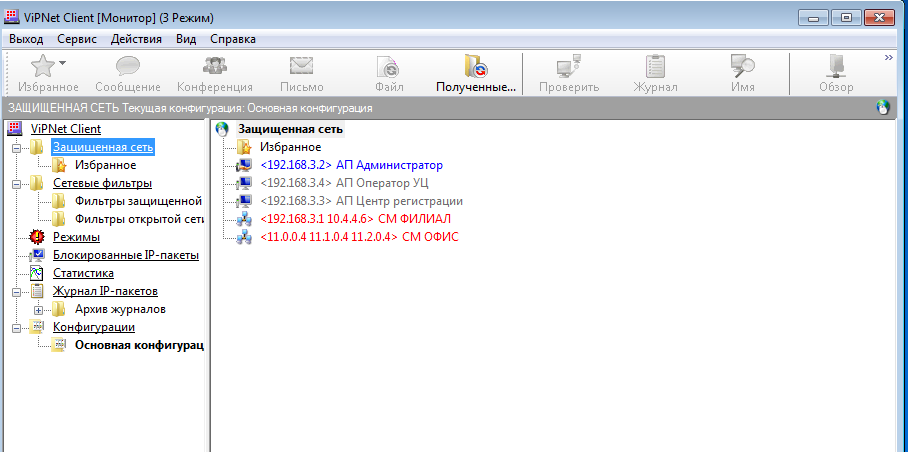


Рисунок 20- Окно защищенная сеть (сети №5245)

Шаг 5 Создание ключевых узлов УКЦ и отправка их в ЦУС (для двух сетей, рисунок 21).

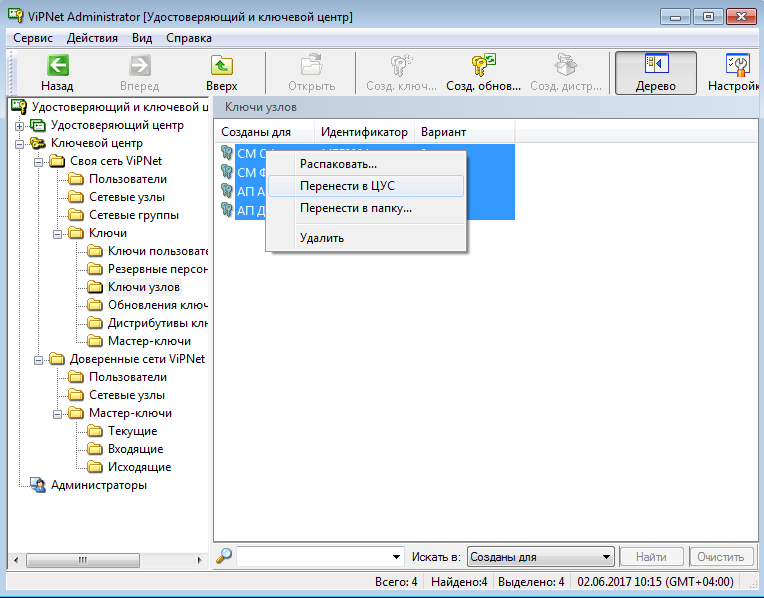


Рисунок 21-Перенос ключей узлов из УКЦ в ЦУС

После этого в ЦУС сетей появляется сообщение (рисунок 22).Следует отправить все ключи.

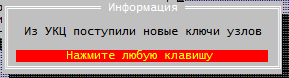


Рисунок 22- Обновление ключей в ЦУС

Для проверки заходим в ViPNetClient монитор сети № 5246, окно Незащищенная сеть (рисунок 23 ).

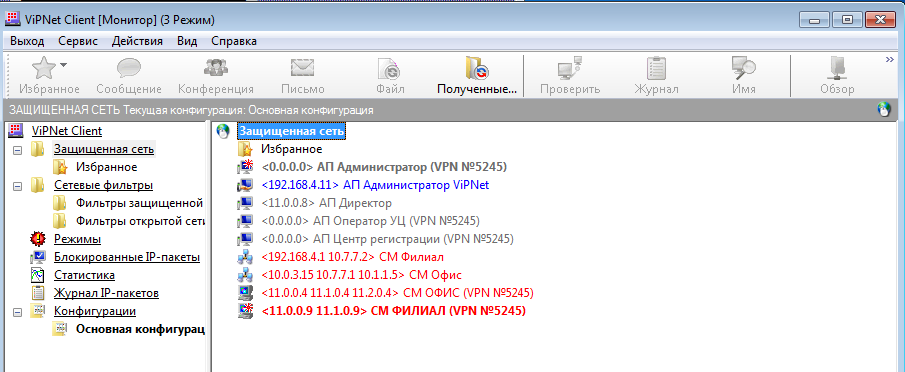


Рисунок 23- Окно незащищенная сеть (№ 5246)

В итоге получили,что АП Менеджер Склада больше не находится в межсетевом взаимодействии ViPnetсетей.

## Создание иерархической структуры удостоверяющих центров

### Основные понятия кросс-сертификации

Кросс-сертификация – это механизм установки доверительных отношений между удостоверяющими центрами. Для администраторов удостоверяющих центров могут быть сформированы следующие типы сертификатов:

* Корневые сертификаты – это сертификаты, в которых издатель сертификата является одновременно и владельцем сертификата. В таком сертификате совпадают поля Издатель и Владелец, определяющие администратора одного удостоверяющего центра.
* Кросс – сертификаты – это сертификаты, в которых издатель сертификата не является одновременно владельцем сертификата. В таком сертификате поля Издатель и Владелец не совпадают, и определяют администраторов различных удостоверяющих центров.

При помощи формирования кросс-сертификатов устанавливаются доверительные отношения между удостоверяющими центрами.

### Издание кросс –сертификата

Задача состоит в том, что необходимо создать иерархическую структуру удостоверяющих центров между сетями № 5245 и № 5246.

Выделяются два типа УЦ:

* Головной УЦ – удостоверяющий центр сети № 5245.
* Подчиненный УЦ – удостоверяющий центр сети № 5246, участвующий в межсетевом взаимодействии.

Шаг 1.Создание запроса к вышестоящему УЦ.

Администратор сети 5246 из меню УКЦ создает запрос к вышестоящему УЦ (УЦ сети 5245) на получения сертификата для Уполномоченного лица своего УКЦ. Заполняем сведения о владельце сертификата (рисунок 24).

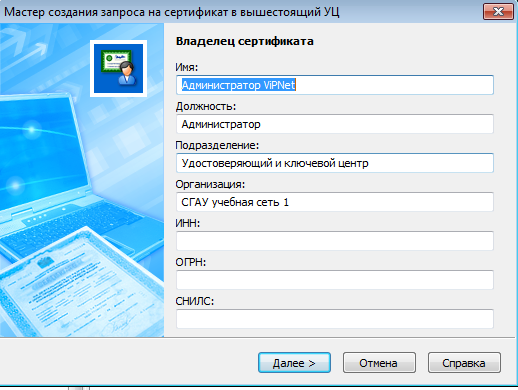


Рисунок 24- Создание запроса на сертификат в вышестоящий УЦ

После заполнения всех пунктов мастера создания запроса на сертификат в вышестоящий откроется окно завершающее работу мастера (рисунок 25).

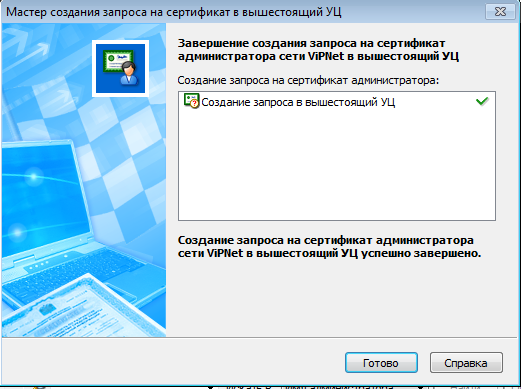


Рисунок 25- Завершение создания запроса на сертификат

В результате работы мастера появится файл с запросом \*.p10. Сформированный файл с запросом передаем в УЦ сети №5245.Администратор сети №5245 по полученному запросу издает кросс-сертификат для уполномоченного лица подчиненного УКЦ.

Шаг 2. Издание кросс-сертификата и его экспорт.

Для издания необходимо в УКЦ выбрать кросс-сертификат и издать его. Потом производим экспорт кросс-сертификата для уполномоченного лица подчиненного УЦ.

Экспортированный файл отправляем вподчиненный УЦ по защищенному каналу связи. Администратор сети ViPNet (Сеть №5246) после получения должен вводит его в действие и проводит обновление ключевых наборов на сетевых узлах.

Шаг 4 Вод сертификата в действие.

Администратор ViPNet (сеть №5245) после получения сертификата должен ввести его в действииПосле проделанных операций нужно посмотреть путь сертификата и убедиться, что он действителен (рисунок 26).

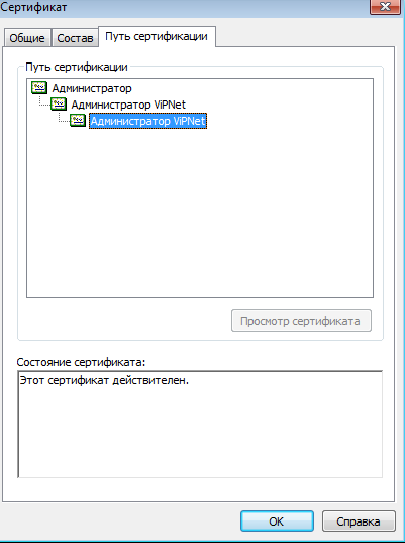


Рисунок 26 –Проверка кросс-сертификата

Исходя из пути сертификата,выдача его прошла успешна. Примерами созданной иерархической структуры УЦ являются структуры сетей таких организаций, как Пенсионный Фонд РФ, Фонд специального страхования.

### Отзыв сертификата пользователя

Пусть необходимо отозвать сертификат пользователя Администратор ViPnet.Для этого выбираем сертификат и меню отозвать. Производим обновление ключей узлов ,переносим их в ЦУС, рассылаем обновления из ЦУС (рисунок 27)

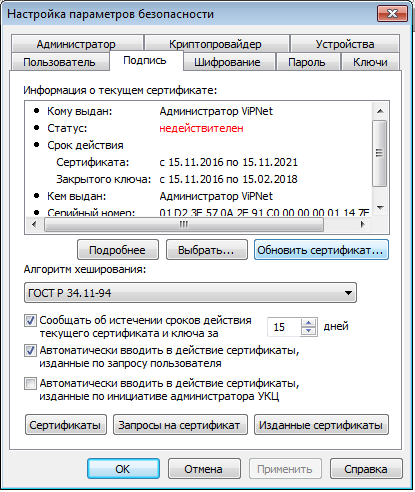


Рисунок 27 – Просмотр информации о сертификате

Если потребуется выпустить новый сертификат, то в УКЦ Администратором создается запрос, ключи узлов и переносится вся информация в ЦУС. Из ЦУСа после обновление справочников и ключей узлов ,их нужно перенести в УКЦ.

* 1. **Использование электронной подписи**

Дляобеспечение целостности документов в ViPNetсуществует электронная подпись. Любой пользователь защищенной сети , у которого есть сертификат может с помощью деловой почты подписывать и отправлять файлы.

Нам нужно отправить картинку (цус1) другому пользователю сети. Подпишем изображение ЭП(рисунок 28).

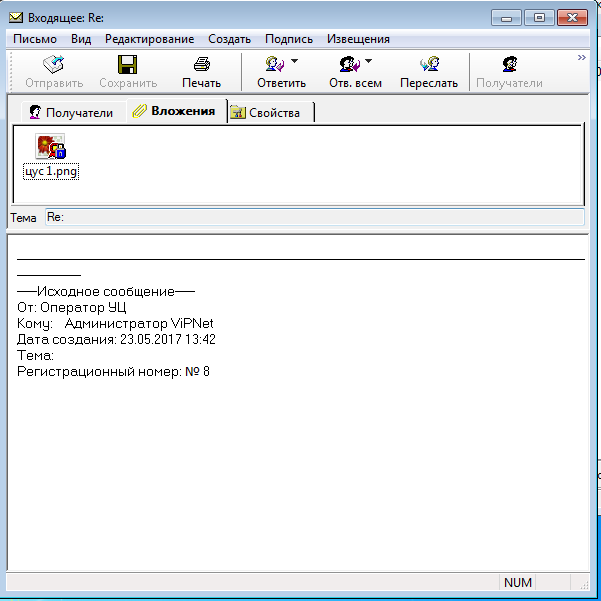


Рисунок 28- Подпись документа (Оператор УЦ)

Отправляем данный файл пользователю Администратор сети ViPNet. Для того чтобы убедиться что документ не был изменен. Открываем проверку электронной подписи (рисунок 23).

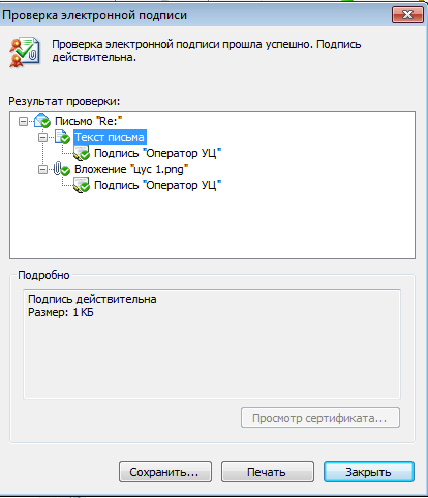


Рисунок 28- Проверка электронной подписи

Как видим из рисунка 28 документ не был никем изменен, потому что подпись действительна.

# **ЗАКЛЮЧЕНИЕ**

В ходе данной работы было изучено существующие программные решения в области безопасности сетей, посредством технологии VPN.C помощью программного продукта ViPNet, компании ОАО «ИнфоТеКС».Была создана виртуальная сеть для последующего изучения студентами этой технологии. Данная сеть является учебной и никакие конфиденциальные данные на ней не нужно хранить потому, что доступ к сети будут иметь студенты университета. Для легкости модификаций в программе VirtualBoxбыли созданы снимки состояний машин. Программный продукт ViPNetCustomпоказал себя очень удобным в использовании, понятный интерфейс программ.

В итоге были получены следующие результаты:

1. Проведенысравнительный обзор существующих программных решений VPN.
2. Создана две сети ViPNetи установлено между ними связь посредством индивидуального симметричного межсетевого мастера-ключа.
3. Проведено тестирование ПО ViPNet на проверку корректности работы созданной сети. Проделан ряд заданий на использование защищенной сети и программного обеспечения ViPNetCustom.

**СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ**

1. Cтруктура распределённой АСУ ТП: [Электронный ресурс]. URL: <http://www.teh-lib.ru/atpip/struktura-raspredeljonnoj-asu-tp/Vse-stranitsy.html> (дата обращения: 05.05.2018)
2. Безопасность промышленных систем: [Электронный ресурс]. URL: https://www.ptsecurity.com/upload/corporate/ru-ru/analytics/SCADA-analytics-rus.pdf (дата обращения: 05.05.2018)
3. Stuxnet и промышленная безопасность: [Электронный ресурс]. URL: http://www.phocus-scada.com/rus/pub/Stuxnet&IndustrialSecurity.html (дата обращения: 05.05.2018)
4. Число атак на АСУ ТП в 2016 году выросло на 110%: [Электронный ресурс]. URL: https://www.securitylab.ru/news/484930.php (дата обращения: 05.05.2018)
5. ОАО ИнфоТекс. Руководство Администратора ViPnetCoordinatorMonitor 3.1: [Электронный ресурс]. 2011 URL: http://ic-dv.ru/files/Producti/Vipnet/Documents/co\_vipnet\_coordinator\_ru.pdf
6. ОАО ИнфоТекс. Развертывание сети ViPNet Руководство администратора: [Электронный ресурс]. 2011 URL:http://ic-dv.ru/files/Producti/Vipnet/Documents/co\_vipnet\_deployment\_ru.pdf
7. ОАО ИнфоТекс. ViPNetAdministrator Центр Управления Сетью 2011 :[Электронныйресурс].2011URL:http://icdv.ru/files/Producti/Vipnet/Documents/adm\_vipnet\_ncc\_ru.pdf
8. Зачем нужен VPN? Принцип соединения и работы в сети: [Электронный ресурс]// информационно технический дневник. URL:<https://hobbyits.com/cifrovye-texnologii/zachem-nuzhen-vpn.html> (дата обращения :12.04.2017)

# **ПРИЛОЖЕНИЕ А КЛЮЧЕВАЯ ИНФОРМАЦИЯ ПОЛЬЗОВАТЕЛЕЙ**

Файлы \*.dst

Сеть №5246

Имя пользователя: СМ Офис

Идентификатор: 147E0001

Тип пароля: Случайный

Пароль: ,kt;tydsr,tl

Парольная фраза: бледная женщина выкидывает бедро

Имя пользователя: CМ Филиал

Идентификатор: 147E0002

Тип пароля: Случайный

Пароль: ptvdlj;fkgfk

Парольная фраза: земляной вдовец жалеет палку

Имя пользователя: Менеджер склада

Идентификатор: 147E0003

Тип пароля: Случайный

Пароль: ecnbyjgjjvfh

Парольная фраза: уставная иностранка поощрила маркизу

Имя пользователя: Администратор ViPNet

Идентификатор: 147E0004

Тип пароля: Случайный

Пароль: gtygjkrfn;fd

Парольная фраза: пенный полярник катает жаворонка

Имя пользователя: Директор

Идентификатор: 147E0005

Тип пароля: Случайный

Пароль: cjcuhtabkgkj

Парольная фраза: составной грешник фильтрует пловчиху

Имя пользователя:Администратор УКЦ

Тип пароля: Случайный

Пароль: ,hfgjighbutj

Парольная фраза: братский пошляк приручил географа

Имя пользователя:Администратор всей сети

Тип пароля: Случайный

Пароль: dt;fnfjxbahfy

Парольная фраза: вежливый атаман очищает франта.

Сеть № 5245

Имя пользователя: СМ ФИЛИАЛ

Идентификатор: 147D0001

Тип пароля: Случайный

Пароль: fh[ghjwtlbtp

Парольная фраза: архаичный прозаик цедит иезуита

Имя пользователя: СМ ОФИС

Идентификатор: 147D0002

Тип пароля: Случайный

Пароль: jlt,jrbpsvfr

Парольная фраза: одержимый боксер изымает макаку

Имя пользователя: Администратор

Идентификатор: 147D0003

Тип пароля: Случайный

Пароль: cnslfvuhehtf

Парольная фраза: стыдливая дама грузит реалиста

Имя пользователя: Оператор УЦ

Идентификатор: 147D0004

Тип пароля: Случайный

Пароль: cdbvjlgj[cnj

Парольная фраза: свирепая модистка похитила сторожа

Имя пользователя: Центр регистрации

Идентификатор: 147D0005

Тип пароля: Случайный

Пароль: uybcxtcxbcj,

Парольная фраза: гнилой счетовод считает собачонку

Имя пользователя: Администратор УКЦ

Тип пароля: Случайный

Пароль: kb[hbcgjvlth

Парольная фраза: лихой рисовод поменял дервиша