МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГУМАНИТАРНО-ПЕДАГОГИЧЕСКАЯ АКАДЕМИЯ (ФИЛИАЛ) ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО АВТОНОМНОГО ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ «КРЫМСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ   
ИМЕНИ В.И. ВЕРНАДСКОГО» В Г. ЯЛТЕ

УТВЕРЖДАЮ

Заведующий кафедрой информатики и информационных технологий

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

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2018 г.

**Пояснительная записка**

к выпускной квалификационной работе магистра

|  |  |
| --- | --- |
| на тему |  |
|  |  |
|  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Выполнил: студент | |  | | курса, группы | 61/3 зИН |
| направления подготовки | | |  | | |
| профиль |  | | | | |
|  |  | | | | |
|  | | | | | |
| (фамилия, имя,отчество студента) | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Руководитель |  |  |  | |
|  | (фамилия,инициалы,степень,звание,должность) |  | (подпись) |

|  |  |  |  |
| --- | --- | --- | --- |
| Нормоконтролёр |  |  |  |
|  | (подпись) |  | (инициалы, фамилия) |

|  |  |  |  |
| --- | --- | --- | --- |
| Дата допуска к защите | «\_\_\_» |  | 20\_\_\_г. |

|  |  |  |  |
| --- | --- | --- | --- |
| Зав. кафедрой |  |  |  |
|  | (подпись) |  | (инициалы, фамилия) |

2018 г.

**ЗАДАНИЕ**

для выпускной квалификационной работы по направлению подготовки 09.04.03 «Прикладная информатика»

Студент **Шкабатур Денис Георгиевич**

Тема выпускной квалификационной работы: **Информационная система анализа и мониторинга локальной вычислительной сети лагеря «Артек»,**

утвержденная приказом по университету № \_\_\_\_от “\_\_\_” \_\_\_\_\_\_\_\_\_ 2018г.

2. Срок сдачи студентом законченной выпускной квалификационной работы

«05» декабря 2018 г.

**Выходные данные к выпускной квалификационной работе**

1. План расположения лагеря «Артек»
2. Перечень установленного оборудования для локальной вычислительной сети в лагере «Артек»
3. Перечень программного обеспечения, установленного для функционирования локальной вычислительной сети лагеря «Артек»

**Содержание** **выпускной квалификационной работы**:

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc530403686)

[1. ИССЛЕДОВАНИЕ ТЕХНОЛОГИЙ МОНИТОРИНГА ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ 4](#_Toc530403687)

[1.1. Технологии мониторинга компьютерных сетей 4](#_Toc530403688)

[1.2. Протоколы мониторинга компьютерных сетей 10](#_Toc530403689)

[1.2.1. Протокол межсетевых управляющих сообщений ICMP 10](#_Toc530403690)

[1.2.2. Протокол сетевого управления SNMP 12](#_Toc530403691)

[1.2.3. Дистанционный мониторинг сети (RMON) 16](#_Toc530403692)

[1.3 Анализ существующих программных средств, реализующих мониторинг компьютерных систем 18](#_Toc530403693)

[1.3.1 Zabbix 18](#_Toc530403694)

[1.3.2 Nagios 21](#_Toc530403695)

[1.3.3 Spiceworks Network Monitor 23](#_Toc530403696)

[1.3.4 Eltex.EMS 28](#_Toc530403697)

[Вывод раздела 1 29](#_Toc530403698)

[2. ИССЛЕДОВАНИЕ МЕТОДОВ И СРЕДСТВ ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ МОНИТОРИНГА КОМПЬЮТЕРНЫХ СЕТЕЙ 30](#_Toc530403699)

[2.1. Описание средств реализации программного продукта 30](#_Toc530403700)

[2.1.1. Обоснование выбора платформы разработки 30](#_Toc530403701)

[2.1.2. Обоснование выбора языка программирования 33](#_Toc530403702)

# ВВЕДЕНИЕ

Актуальность\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Проблема, исследуемая в выпускной квалификационной работе.*Бесперебойное функционирование локальной вычислительной сети невозможно без программного обеспечения, проводящего мониторинг и диагностику работоспособности сети. Имеющееся программное обеспечение либо является слишком многофункциональным, сложным и трудоемким в обслуживании, либо дорогим и требующим дополнительного сопровождения разработчика, либо не решает некоторых необходимых задач анализа. Разработка информационной системы мониторинга и анализа ЛВС даст возможность отслеживать проблемы предоставления ресурсов локальной вычислительной сети и предоставлять качественные интернет-услуги пользователям лагеря «Артек».

**Цель выпускной квалификационной работы** – анализ технологий мониторинга локальной вычислительной сети, исследование возможных ее неполадок и способов их устранения, а также разработка информационной системы мониторинга локальной вычислительной сети лагеря «Артек»

**Задачи выпускной квалификационной работы**

1. Анализ технологий мониторинга ЛВС.
2. Исследование средств проектирования информационной системы.
3. Разработка информационной системы мониторинга локальной вычислительной сети лагеря «Артек».

**Объект и предмет исследования выпускной квалификационной работы**

**Объект исследования выпускной квалификационной работы –** технологии мониторинга локальной вычислительной сети

**Предмет исследования выпускной квалификационной работы –** методы и модели проектирования информационной системы мониторинга локальной вычислительной сети лагеря «Артек».

**Теоретическая значимость выпускной квалификационной работы** состоит в создании методики разработки мониторинга и анализа локальной вычислительной сети лагеря «Артек».

**Практическая значимость выпускной квалификационной работы** состоит в практическом применении предложенной методики в лагере «Артек»

# ИССЛЕДОВАНИЕ ТЕХНОЛОГИЙ МОНИТОРИНГА ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ

## Технологии мониторинга компьютерных сетей

В последние годы информационные технологии претерпевают значительные и постоянные изменения. По некоторым оценкам, за последние пять лет объем сетевого трафика локальных сетей вырос в десять раз. Таким образом, локальные сети должны обеспечивать все большую пропускную способность и необходимый уровень качества обслуживания. Однако какие бы ресурсы ни имела сеть, они все-таки конечны, поэтому для сети необходима возможность управления трафиком.

А для того чтобы управление было максимально эффективным, требуется возможность контроля над пакетами, передающимися между устройствами вашей сети. Также у администратора существует великое множество обязательных для исполнения ежедневных операций. Это, например, проверка правильности функционирования электронной почты, просмотр регистрационных файлов на предмет выявления ранних признаков неисправностей, контроль за подключением локальных сетей и за наличием системных ресурсов. И здесь на помощь могут прийти средства, применяемые для мониторинга и анализа вычислительных сетей.

Чтобы не запутаться в многообразии методик, средств и продуктов, созданных для мониторинга, начнем с краткого описания нескольких крупных классов этих продуктов.

Системы управления сетью (Network Management Systems). Это централизованные программные системы, которые собирают данные о состоянии узлов и коммуникационных устройств сети, а также о трафике, циркулирующем в сети. Эти системы не только осуществляют мониторинг и анализ сети, но и выполняют в автоматическом или полуавтоматическом режиме действия по управлению сетью — включение и отключение портов устройств, изменение параметров мостов адресных таблиц мостов, коммутаторов и маршрутизаторов и т.п.

Средства управления системой (System Management). Средства управления системой часто выполняют функции, аналогичные функциям систем управления, но по отношению к другим объектам. В первом случае объектом управления является программное и аппаратное обеспечение компьютеров сети, а во втором — коммуникационное оборудование. Вместе с тем некоторые функции этих двух видов систем управления могут дублироваться, например средства управления системой могут выполнять простейший анализ сетевого трафика.

Встроенные системы диагностики и управления (Embedded systems). Эти системы выполняются в виде программно-аппаратных модулей, устанавливаемых в коммуникационное оборудование, а также в виде программных модулей, встроенных в операционные системы. Они выполняют функции диагностики и управления только одним устройством, и в этом их основное отличие от централизованных систем управления.

Анализаторы протоколов (Protocol analyzers). Представляют собой программные или аппаратно-программные системы, которые ограничиваются, в отличие от систем управления, лишь функциями мониторинга и анализа трафика в сетях. Хороший анализатор протоколов может захватывать и декодировать пакеты большого количества применяемых в сетях протоколов — обычно несколько десятков. Анализаторы протоколов позволяют установить некоторые логические условия для захвата отдельных пакетов и выполняют полное декодирование захваченных пакетов, то есть показывают в удобной для специалиста форме вложенность пакетов протоколов разных уровней друг в друга с расшифровкой содержания отдельных полей каждого пакета.

Экспертные системы. Системы этого вида аккумулируют человеческие знания о выявлении причин аномальной работы сетей и возможных способах приведения сети в работоспособное состояние. Экспертные системы часто реализуются в виде отдельных подсистем различных средств мониторинга и анализа сетей: систем управления сетями, анализаторов протоколов, сетевых анализаторов. Простейшим вариантом экспертной системы является контекстно-зависимая help-система. Более сложные экспертные системы представляют собой так называемые базы знаний, обладающие элементами искусственного интеллекта.

Многофункциональные устройства анализа и диагностики. В последние годы в связи с повсеместным распространением локальных сетей возникла необходимость разработки недорогих портативных приборов, совмещающих функции нескольких устройств: анализаторов протоколов, кабельных сканеров и даже некоторых возможностей ПО сетевого управления.

Кроме визуального наблюдения за состоянием сетевых устройств, программное обеспечение дает возможность организации проверки хостов и служб (в том числе и локальных ресурсов или интернет-серверов) с помощью разнообразных сетевых протоколов настройки и использования удобного способа оповещения системных администраторов о положительных или отрицательных результатах тестирования. Способы могут быть разными: появление сообщения на экране компьютера ИТ специалиста, специальный звук, отправка электронного письма или смс на телефон. Приложение для мониторинга компьютеров в локальной сети предприятия в ряде случаев может перезапустить какую-либо удаленную службу либо выполнить заранее написанный для него скрипт (тогда некоторые сбои будут устраняться в автоматическом режиме).

Если в программе реализована такая функция, то все подключенные к сети устройства будут наглядно отображаться на ее схеме. Уже по одному виду их иконок специалисту станет понятно, какие из них работают нормально, а какие функционируют неверно. Такая возможность облегчает диагностику групповых сбоев. Полученные результаты тестирования заносятся в единую базу данных; по мере накопления статистической информации можно будет строить графики, чтобы изучать изменение отклика устройств и отслеживать другие проверяемые параметры.

Современные программы мониторинга компьютеров в локальной сети позволяют создать своего рода пульт управления сетевой инфраструктурой предприятия, при помощи которого отвечающий за сеть сотрудник сможет как наблюдать за ее важными элементами и проверять параметры оборудования, так и вести эффективное управление удаленными хостами. С помощью контекстного меню хостов можно просматривать различные данные об удаленных узлах в сети: проверять информацию по SNMP с коммутаторов, получить доступ к реестрам удаленных компьютеров, просматривать запущенные процессы и журналы событий, производить перезапуск служб и выполнять другие действия. Некоторые программы ведут не только мониторинг ресурсов компьютера, но и помогают проводить учет устройств и программных приложений на сетевых ПК. Благодаря им системный администратор имеет возможность получать практически любые сведения об аппаратном и программном обеспечении на компьютерах в корпоративной сети. Сбор данных проходит удаленно, это позволяет не мешать работать сотрудникам предприятия, и экономит рабочее время системных администраторов.

Программы наблюдения за ПК в локальной сети позволяют вести точный учет аппаратного обеспечения. ИТ специалисты смогут оперативно узнать о пропаже и неисправности какого-либо компонента либо о его замене. При обнаружении изменений они заносятся в журнал, и о них оповещается администратор сети. Если нужно отслеживать определенные параметры на рабочих станциях пользователей с конкретной частотой и получать оповещения при их изменении, возможно будет настроить сбор данных по расписанию. В этом случае мониторинг ресурсов компьютера будет вестись в автоматическом режиме.

Многие программы не только формируют отчеты по компонентам компьютеров, но и следят за их работоспособностью – контролируется функционирование жестких дисков и их температура. Когда какой-либо диск перегревается или приложение выдает прогноз о том, что он может выйти из строя, сисадмин увидит отчет, содержащий критические замечания по работе ПК с предупреждением.

При необходимости софт мониторинга ПО в локальных сетях предприятия позволяет отслеживать и изменения в установленных программах. В случае, когда права пользователей строго не ограничиваются, кто-либо из сотрудников предприятия может установить на своем компьютере нежелательное или нелицензионное приложение. При мониторинге, когда происходит инсталляция или удаление программного обеспечения, любое изменение фиксируется и заносится в журнал. Это означает, что системный администратор будет всегда знать, какие именно программы и куда были инсталлированы или откуда удалены.

Часто при мониторинге ресурсов компьютера происходит отслеживание серийных номеров и лицензий программ, подсчет количества установок ПО, контроль правильного использования серийных номеров. Все эти меры реально помогают избежать проблем при проверке корпоративного программного обеспечения на предмет лицензионной чистоты. С целью повышения уровня безопасности и отказоустойчивости компьютеров ряд специальных программ мониторинга имеют функции просмотра проведенных обновлений ПО и системы и составления отчета по работе антивирусного софта и актуальности его баз.

Данные, собираемые программой с сетевых компьютеров и отображаемые на экране ПК системного администратора, могут быть занесены в отчет. Далее их можно распечатать или экспортировать в определенную базу. Кроме автоматически собираемой информации, многие приложения для мониторинга ресурсов компьютера разрешают вводить вручную серийные номера оборудования, номера офисов их пользователей и их контактные данные.

Таким образом на сегодняшний день существует большое количество различных технологий мониторинга компьютерных систем использующих различные сетевые протоколы.

* 1. **Протоколы мониторинга компьютерных сетей**
     1. **Протокол межсетевых управляющих сообщений ICMP**

Протокол межсетевых управляющих сообщений ICMP преимущественно используется для передачи сообщений об ошибках передачи данных. Он используется сетевыми устройствами, включая маршрутизаторы, для отправки сообщений об ошибках и оперативной информации, указывающей, например, что запрошенная услуга недоступна или что хост или маршрутизатор не могут быть достигнуты. ICMP отличается от транспортных протоколов, таких как TCP и UDP, тем, что он обычно не используется для обмена данными между системами, и не используется обычным сетевым приложениям конечного пользователя (за исключением некоторых диагностических инструментов, таких как ping и traceroute).

Сообщения ICMP обычно используются для целей диагностики или контроля и генерируются в ответ на ошибки в IP-операциях (как указано в RFC 1122). ICMP-ошибки направляются на исходный IP-адрес исходного пакета.

Каждое устройство (например, промежуточный маршрутизатор), перенаправляющее IP-датаграмму, сначала уменьшает поле времени для жизни (TTL) в заголовке IP на единицу. Если итоговый TTL равен 0, пакет отбрасывается и время ICMP, повышенное в транзитном сообщении, отправляется на адрес источника дейтаграммы.

Многие широко используемые сетевые утилиты основаны на сообщениях ICMP. Команда traceroute может быть реализована путем передачи IP-дейтаграмм с помощью специально заданных полей заголовка IP TTL и поиска времени ICMP, превышенного в пути, и недостижимых сообщений, сгенерированных в ответ. Связанная утилита ping реализована с использованием ICMP-эхо-запроса и сообщений эхо-ответа.

ICMP использует базовую поддержку IP, как если бы это был протокол более высокого уровня, однако ICMP на самом деле является неотъемлемой частью IP. Хотя ICMP-сообщения содержатся в стандартных IP-пакетах, сообщения ICMP обычно обрабатываются как особый случай, отличающийся от обычной обработки IP-адресов. Во многих случаях необходимо проверить содержимое ICMP-сообщения и доставить соответствующее сообщение об ошибке в приложение, ответственное за передачу IP-пакета, запрашивающего отправку ICMP-сообщения.

ICMP - это протокол сетевого уровня. Номер порта TCP или UDP не связан с ICMP-пакетами, так как эти числа связаны с транспортным уровнем выше.

Таким образом ICMP представляет собой наиболее простой сетевой протокол, применяющийся для базового мониторинга сети и выявления проблем.

* + 1. **Протокол сетевого управления SNMP**

Этот протокол интернет-стандарта для сбора и организации информации об управляемых устройствах в IP-сетях и для изменения этой информации для изменения поведения устройства. Устройства, которые обычно поддерживают SNMP, включают в себя кабельные модемы, маршрутизаторы, коммутаторы, серверы, рабочие станции, принтеры и так далее.

SNMP широко используется в управлении сетью и для сетевого мониторинга. SNMP предоставляет данные управления в виде переменных в управляемых системах, организованных в информационной базе управления (MIB), которые описывают состояние системы и ее конфигурацию. Затем эти переменные могут быть дистанционно запрошены.

Разработаны и развернуты три значимые версии SNMP. SNMPv1 является исходной версией протокола. Более свежие версии, SNMPv2c и SNMPv3, улучшают производительность, гибкость и безопасность.

SNMP является компонентом пакета интернет-протокола, как определено целевой группой Internet Engineering Task Force (IETF). Он состоит из набора стандартов для управления сетью, включая протокол уровня приложения, схему базы данных и набор объектов данных.

При типичном использовании SNMP один или несколько компьютеров, называемых менеджерами, выполняют задачу мониторинга управления группы хостов или устройств в компьютерной сети. Каждая управляемая система выполняет программный компонент, называемый агентом, который передает информацию через SNMP менеджеру.

Сеть, управляемая SNMP, состоит из трех ключевых компонентов:

* Управляемые устройства
* Агент - программное обеспечение, которое работает на управляемых устройствах
* Станция управления сетью (NMS) - программное обеспечение, которое работает у менеджера

Управляемое устройство является сетевым узлом, который реализует SNMP-интерфейс, который позволяет однонаправленный (только для чтения) или двунаправленный (чтение и запись) доступ к информации, специфичной для узла. Управляемые устройства обмениваются узловыми данными с NMS. Иногда называемые сетевые элементы, управляемые устройства могут быть устройствами любого типа, включая маршрутизаторы, серверы доступа, коммутаторы, кабельные модемы, мосты, концентраторы, IP-телефоны, IP-камеры, компьютерные хосты и принтеры.

Агент - это программный модуль сетевого управления, который находится на управляемом устройстве. Агент имеет локальное знание управляющей информации и переводит эту информацию в форму, определенную SNMP.

На станции управления сетью выполняются приложения, которые контролируют и управляют управляемыми устройствами. NMS обеспечивают основную часть ресурсов обработки и памяти, необходимых для управления сетью. Одна или несколько NMS могут существовать в любой управляемой сети.

Агент SNMP предоставляет данные управления управляемыми системами как переменные. Протокол также разрешает активные задачи управления, такие как изменения конфигурации, посредством удаленной модификации этих переменных. Переменные, доступные через SNMP, организованы в иерархии. Сам SNMP не определяет, какие переменные должна предлагать управляемая система. Скорее, SNMP использует расширяемый дизайн, который позволяет приложениям определять свои собственные иерархии. Эти иерархии описываются как информационная база управления (MIB). MIB описывают структуру данных управления подсистемой устройства; они используют иерархическое пространство имен, содержащее идентификаторы объектов (OID).

SNMP работает на прикладном уровне пакета интернет-протокола. Все сообщения SNMP передаются через протокол пользовательских дейтаграмм (UDP). Агент SNMP принимает запросы на порт 16 UDP. Менеджер может отправлять запросы от любого доступного порта источника к порту 161 в агенте. Ответ агента отправляется обратно в исходный порт диспетчера. Менеджер получает уведомления (Ловушки и InformRequests) на порту 162. Агент может генерировать уведомления из любого доступного порта. При использовании с защитой транспортного уровня или защитой транспортного уровня датаграммы запросы принимаются на порт 10161, а уведомления отправляются на порт 10162. [3]

SNMPv1 определяет пять основных блоков данных протокола (PDU). В SNMPv2 добавлены два других блока PDU, GetBulkRequest и InformRequest, а в SNMPv3 добавлен PDU отчета. Все блоки протоколов SNMP построены следующим образом:

IP-заголовок UDP-версия заголовка сообщества PDU-тип request-id error-status error-index привязки переменных

Семь типов PDU SNMP, идентифицированные полем типа PDU, следующие:

* GetRequest- запрос менеджера-агента для получения значения переменной или списка переменных. Желаемые переменные указываются в привязках переменных (поле значения не используется). Получение указанных переменных значений должно выполняться агентом. Возвращается ответ с текущими значениями.
* SetRequest - запрос менеджера-агента для изменения значения переменной или списка переменных. Переменные привязки указываются в теле запроса. Изменения во всех указанных переменных должны выполняться агентом. Возвращается ответ с (текущими) новыми значениями для переменных.
* GetNextRequest - запрос менеджера-агента для обнаружения доступных переменных и их значений. Возвращает ответ с привязкой переменной для лексикографически следующей переменной в MIB. Весь MIB агента можно пройти путем итеративного применения GetNextRequest, начиная с OID 0. Строки таблицы можно прочитать, указав OID столбца в привязках переменных запроса.
* GetBulkRequest - запрос менеджера-агента для нескольких итераций GetNextRequest. Оптимизированная версия GetNextRequest. Возвращает ответ с несколькими привязками переменных, идущими от привязки переменных или привязок в запросе. Для контроля поведения ответчика используются конкретные не повторительные PDU и max-repetitions. GetBulkRequest был представлен в SNMPv2.
* Response - возвращает привязки переменных и подтверждение от агента к менеджеру для GetRequest, SetRequest, GetNextRequest, GetBulkRequest и InformRequest. Отчеты об ошибках предоставляются по положению об ошибках и по индексам ошибок. Хотя он использовался как ответ на получение и набор, этот PDU назывался GetResponse в SNMPv1.
* Trap - асинхронное уведомление от агента к менеджеру. В то время как в другой SNMP-связи менеджер активно запрашивает информацию у агента, это PDU, которые отправляются от агента менеджеру без явного запроса.
* InformRequest - подтвержденное асинхронное уведомление. Этот PDU был введен в SNMPv2 и первоначально был определен как менеджер для обмена менеджером.
  + 1. **Дистанционный мониторинг сети (RMON)**

RMON был разработана IETF для поддержки мониторинга и анализа протоколов локальных сетей. Первоначальная версия (иногда называемая RMON1) была посвящена информации уровня OSI 1 и уровня 2 в сетях Ethernet и Token Ring. Он был расширен до RMON2, который добавляет поддержку мониторинга сетевого и прикладного уровня и SMON, добавляет поддержку коммутируемых сетей. Это стандартная отраслевая спецификация, которая обеспечивает большую часть функциональных возможностей, предлагаемых проприетарными анализаторами сети. Агент RMON встроен во многие высококлассные коммутаторы и маршрутизаторы.

Удаленный мониторинг (RMON) - стандартная контрольная спецификация, которая позволяет различным сетевым мониторам и консольным системам обмениваться данными мониторинга сети. RMON предоставляет сетевым администраторам больше свободы в выборе консолей мониторинга сети с функциями, отвечающими их конкретным сетевым потребностям. Реализация RMON обычно работает в модели клиент / сервер. Устройства мониторинга (обычно называемые «зонды» в этом контексте) содержат программные агенты RMON, которые собирают информацию и анализируют пакеты. Эти зонды действуют как серверы, а приложения Network Management, которые общаются с ними, действуют как клиенты. Хотя конфигурация и сбор данных агента использует SNMP, RMON предназначен для работы иначе, чем другие системы на базе SNMP:

У зондов больше ответственности за сбор и обработку данных, что снижает трафик SNMP и нагрузку на обработку клиентов.

Информация передается только в приложение управления, если требуется, вместо непрерывного опроса и мониторинга. Короче говоря, RMON предназначен для мониторинга на основе потока, тогда как SNMP часто используется для управления на основе устройств. RMON похож на другие технологии мониторинга потока, такие как NetFlow и SFlow, поскольку собранные данные в основном касаются шаблонов трафика, а не состояния отдельных устройств. Одним из недостатков этой системы является то, что удаленные устройства несут большую часть бремени управления и требуют больше ресурсов для этого.

RMON состоит из десяти групп:

* Статистика: статистика LAN в реальном времени, например. использование, столкновение, ошибки CRC
* История: история выбранной статистики
* Тревога: определения сообщений SNMP SNMP, которые должны быть отправлены, когда статистика превышает определенные пороговые значения
* Хосты: конкретная локальная статистика LAN, например. байтов, отправленных / полученных, отправленных / полученных кадров
* Ведущие вершины: запись N наиболее активных подключений за определенный период времени
* Матрица: принятая схема трафика между системами
* Фильтр: определяет интересующие образцы пакетов данных, например MAC-адрес или порт TCP
* Захват: сбор и пересылка пакетов, соответствующих фильтру
* Событие: отправьте предупреждения (SNMP-ловушки) для группы «Тревога»
* Token Ring: расширения, характерные для Token Ring

## Анализ существующих программных средств, реализующих мониторинг компьютерных систем

* + 1. **Zabbix**

Zabbix - это программный инструмент для мониторинга открытых источников для различных ИТ-компонентов, включая сети, серверы, виртуальные машины (VM) и облачные сервисы. Zabbix обеспечивает показатели мониторинга, такие как использование сети, загрузка процессора и потребление дискового пространства. Программное обеспечение контролирует работу в Linux, Hewlett Packard Unix (HP-UX), Mac OS X, Solaris и других операционных системах (ОС); однако мониторинг Windows возможен только через агентов.

Zabbix может быть развернут для контроля на основе агентов и без агента . Агенты установлены на ИТ-компонентах для проверки производительности и сбора данных. Затем агент возвращается к централизованному серверу управления Zabbix. Эта информация включена в отчеты или представлена визуально в графическом пользовательском интерфейсе Zabbix ( GUI ) . Если есть какие-либо проблемы в отношении того, что отслеживается, Zabbix отправит уведомление или оповещение пользователю. Безрисковый мониторинг осуществляет такой же тип мониторинга, используя существующие ресурсы в системе или устройстве для эмуляции агента.

Веб-интерфейс Zabbix позволяет пользователям просматривать свою ИТ-среду с помощью настраиваемых панелей мониторинга на основе виджетов , графиков, сетевых карт, слайд-шоу и отчетов. Например, пользователь может настроить отчет для отображения показателей, связанных как с соглашениями об уровне обслуживания ( SLA ), так и с ключевыми показателями эффективности ( KPI ) при загрузке ЦП.

Zabbix работает в трех режимах обнаружения:

* + Обнаружение сети периодически сканирует ИТ-среду и записывает тип устройства, IP-адрес, статус, время простоя и время простоя.
  + Обнаружение низкого уровня автоматически создает элементы, триггеры и графики на основе обнаруженного устройства. Низкоуровневое обнаружение может создавать показатели из идентификаторов объектов Simple Network Management Protocol ( SNMP ) , служб Windows, запросов на структурированный запрос запросов (SQL), связанных с базами данных ( ODBC ), сетевых интерфейсов и т.д.
  + Автоматическое обнаружение автоматически начинает мониторинг любого обнаруженного устройства с помощью агента Zabbix.

Zabbix может отправлять уведомления по электронной почте на основе предопределенных событий в ИТ-среде пользователя. Еще один способ для пользователей Zabbix оставаться в курсе своей ИТ-среды - это мобильные приложения от таких поставщиков, как M7 Monitoring или их собственное создание.

Zabbix предлагает несколько вариантов мониторинга вне агентов.

Простая проверка может проверить доступность и отзывчивость стандартной службы, такой как уведомления или HTTP.

Расширения управления Java ( JMX ), веб-мониторинг и другие методы также являются альтернативой использованию агентов. В Zabbix JMX можно использовать для мониторинга приложений на Java. Веб-мониторинг используется для проверки доступности веб-сайтов и поддерживает HTTP и HTTPS. Zabbix собирает данные, касающиеся средней скорости загрузки сценария, ошибок и сообщений об ошибках, времени отклика и т.д.

Интерфейс программирования Zabbix представляет собой веб-интерфейс для создания новых приложений, автоматизации задач и интеграции с сторонним программным обеспечением, таким как go-zabbix, Zabbix :: Tiny или Zabbix отправитель. JavaScript Object Notation ( JSON - формат) используется для базировать API в качестве интерфейсного веб - интерфейс.

API Zabbix состоит из множества методов , которые сгруппированы в отдельные API-интерфейсы, каждый из которых выполняет определенную службу. Например, метод создания нового хоста - host.create; метод входа в систему в качестве администратора - user.login.

Используя API, пользователи могут создавать приложения для работы и отображения информации Zabbix.

Также Zabbix имеет встроенную поддержку шаблонов. Шаблоны - это настраиваемые надстройки, расширяющие функциональность Zabbix. Некоторые шаблоны сделаны Zabbix и поставляются в комплекте с готовым к использованию программным обеспечением, а другие - пользователями Zabbix. Шаблоны позволяют пользователям Zabbix отслеживать сетевые устройства от таких поставщиков, как Cisco, Dell, HP и Juniper. Другие шаблоны могут использоваться для мониторинга серверов IBM, HP и Super Micro. Шаблоны для служб на основе приложений включают Microsoft Exchange и Exchange Server, Zenoss , PowerDNS, Authoritative Server Stats и другие. Шаблоны могут быть созданы для мониторинга ОС и гипервизоров.

### Nagios

Nagios был разработан для Linux, но теперь способен отслеживать операционные системы на базе UNIX. Он состоит из трех компонентов: демона, веб-интерфейса и плагинов. Nagios имеет четыре основных типа объектов, которые перечислены ниже.

* Команды: они используются для взаимодействия с плагинами и для управления обработчиками событий, уведомлениями и проверками.
* Контактные и контактные группы: они определяют лиц, с которыми необходимо связаться в случае события.
* Группы хостов и хостов: они используются для указания служб и хостов в определенной сети.
* Уведомления. Они определяют, какой контент должен быть отправлен контактам и контактным группам, когда обнаружены.

Определения проверки хоста и службы определяют, какие плагины вызывать для получения статуса хоста или службы. Определения в плагинах проверяются на этом конкретном хосте / службе. Если видно, что значения выходят за порог, Nagios уведомляет контакты / контактные группы. Nagios возвращает любой из следующих четырех кодов статуса для любого события:

1. OK
2. Предупреждение
3. Критический
4. Неизвестно

Nagios можно настроить для проведения пассивных или активных проверок. В активных проверках хост, который так часто запускает устройства или службы Nagios для получения информации о статусе, тогда как пассивные проверки инициируются и выполняются внешними приложениями / процессами. Затем результаты отправляются на хост мониторинга (где работает Nagios) для обработки. Основное различие между активными и пассивными проверками заключается в том, что активные проверки инициируются и выполняются Nagios, в то время как пассивные проверки выполняются внешними приложениями. Пассивные проверки могут использоваться для мониторинга асинхронных служб и мониторинга служб, расположенных за брандмауэром. Активные проверки могут использоваться для проверки по требованию и регулярных проверок интервалов. Результаты могут контролироваться через веб-интерфейс, основанный на CGI.

Главные особенности системы:

* Планирование обновления инфраструктуры до того, как устаревшие системы вызывают сбои
* Отвечать на вопросы при первом признаке проблемы
* Автоматическое исправление проблем при обнаружении
* Согласовать ответы технической команды
* Обеспечьте соблюдение SLA вашей организации
* Обеспечить минимальные последствия для ИТ-инфраструктур на нижней строке вашей организации
* Мониторинг всей инфраструктуры и бизнес-процессов

### Spiceworks Network Monitor

Spiceworks Network Monitor - бесплатный инструмент, предназначенный для мониторинга и статистики в реальном времени для серверов и сетевых устройств, поддерживающих SNMP. Хотя он бесплатный, он не является программным продуктом с открытым исходным кодом, так же в правом верхнем углу присутствует реклама. Сетевой монитор Spiceworks можно использовать вместе с инструментами технической поддержки Spiceworks и инструментами управления ресурсами, но я рассматриваю его как отдельный программный продукт.

Сетевой монитор работает с любой версией Windows с Windows Server 2008 R2, и мы установили его на сервере Windows 2012 R2, на котором не было никаких других служб. Установка и настройка не очень привлекательны: просто загрузите программное обеспечение с сайта Spiceworks и запустите программу установки. После завершения работы ярлык на рабочем столе приведет к веб-интерфейсу, где можно завершить процесс начальной настройки.

Перед тем, как войти в систему мониторинга, вам понадобится учетная запись Spiceworks - нужно будет перейти на spiceworks.com, чтобы установить ее. Для этого потребуется имя, адрес электронной почты и пароль. После входа в систему нам был представлен экран панели управления по умолчанию. Ниже горизонтального меню и предупреждающих полосок в верхней части экрана есть поля для сетевых и сетевых наблюдателей. Ниже расположены пробелы для добавления 3 устройств для более тщательного мониторинга, в которых будет отображаться более подробная информация о каждом из этих устройств.

При первом входе в систему пользовательская панель автоматически открывается через левую треть страницы, раздражающе скрывая элементы панели инструментов под ней. Основная справочная информация, которую он отображает, не очень полезна, но после ее закрытия она остается закрытой для будущих логинов, если она не будет повторно открыта.

Помимо основной информационной панели, для устройств есть выделенные страницы, где можно добавлять и просматривать детали машин, которые вы хотите отслеживать, и параметры, в которых можно настраивать оповещения и добавлять пользователей. Существует также опция меню для справки, но вместо того, чтобы сразу предоставить вам соответствующую документацию, она приведет вас на сайт сообщества Spiceworks, где нужно настроить имя форума, прежде чем сможете получить доступ к любой полезной информации. В процессе создания учетной записи настраивается профиль для своей компании, а также один для себя, и хотя можно пропустить большую часть этого, это раздражает, если вам просто нужна немедленная помощь с программным обеспечением.

Еще более раздражающе, как только вы зарегистрировали профиль, вы попадаете на главную страницу сообщества Spiceworks вместо справочных страниц сетевого монитора. К счастью, после регистрации в будущем попытки получить доступ к справке через веб-интерфейс Network Monitor вы попадете прямо на страницу поддержки сообщества, где вы можете искать ранее заданные вопросы или перейти на форумы поддержки.

Программное обеспечение отслеживает 26 различных параметров для каждого устройства, разделенных на 5 категорий: хост, процессор, память, диск и сеть. Они могут отслеживать конкретные условия, такие как постоянная высокая загрузка процессора, спайки в использовании памяти, низкое дисковое пространство и узкие места в сети. Однако, в отличие от более полнофункциональных решений мониторинга (как коммерческих, так и открытых), у него нет возможности отслеживать конкретные процессы или каким-либо образом создавать пользовательские предупреждения для определенных условий ошибки.

На странице настроек есть только две вкладки, перечисленные в левой части экрана, «Мониторы по умолчанию» и «Учетные записи пользователей». Первый позволяет настроить пороговые значения по умолчанию для контролируемых параметров и выбрать, какие будут генерировать оповещения по электронной почте. Вторая вкладка позволяет добавлять или удалять пользователей для системы мониторинга. К сожалению, пользователи не могут быть добавлены напрямую. Вместо этого вы вводите свое имя и адрес электронной почты, и система отправляет им приглашение по электронной почте со ссылкой. Если они войдут в систему, для них автоматически будет создана учетная запись spiceworks.com.

Страница устройств начинается с одного только перечисленного устройства: машины, на которой работает программа мониторинга. Нажатие кнопки добавления устройства открывает новую панель с вкладками для добавления компьютеров под управлением Windows или Linux, а также сетевых устройств, таких как маршрутизаторы и брандмауэры. Чтобы добавить устройство, все, что вам нужно, это его IP-адрес или имя хоста и логин для этой системы с достаточными привилегиями. При добавлении сервера Windows мы использовали учетную запись с правами локального администратора на этом сервере. Для серверов Linux мы использовали стандартные учетные записи пользователей, которым был предоставлен полный доступ к sudo.

После добавления устройства пороговые значения и параметры электронной почты для каждого предупреждения могут быть оставлены по умолчанию или настроены для этого устройства. Мы моделировали различные системные ошибки на наших тестовых серверах. Большинство из них сообщалось точно, но когда мы использовали программный инструмент для поддержания центрального процессора нашего сервера Windows на 100%, Spiceworks Network Monitor показал правильность загрузки процессора на своем графике, но его список процессов показал только 50% загрузки процессора для а не почти 100% нагрузки, отображаемой диспетчером задач на самом сервере.

Оповещения отображаются в веб-интерфейсе сетевого монитора и отправляются по электронной почте, если вы включили оповещения по электронной почте для данного параметра. Мы обнаружили, что письмо первому пользователю прибыло незамедлительно, но письмо второму пользователю, которого мы добавили, всегда было на 15 минут позже первого. Нелья указать, какие оповещения будут отправляться пользователям, либо: всем пользователям отправляются все предупреждения. Это нормально, если вы контролируете только несколько серверов, но в более крупной компании, где разные сотрудники могут нести ответственность за разные группы машин, могут возникнуть проблемы с ограниченными возможностями настройки для оповещений. Пока предупреждение не будет очищено, электронные письма с напоминанием отправляются каждые 30 минут. Окончательное письмо отправляется после устранения проблемы.

Стоит отметить, что сообщения электронной почты отправляются через системы spiceworks.com, вместо того, чтобы использовать собственный внутренний почтовый сервер, поэтому, если интернет-соединение не работает, вы не получите предупреждающие сообщения. Сетевой монитор аналогично полагается на Spiceworks для аутентификации пользователей, поэтому вы не можете установить средство мониторинга на сервере в защищенном разделе сети с ограниченным доступом в Интернет.

Хотя мы обнаружили, что Spiceworks Network Monitor был быстр и прост в установке и настройке, отсутствие детализации конфигурации и невозможность определения пользовательских предупреждений были разочаровывающими. В сочетании с его зависимостью от внешних систем spiceworks.com для аутентификации пользователей и электронной почты и ограниченными возможностями мониторинга эти факторы делают его непригодным для настроек мониторинга корпоративного уровня или даже для среднего бизнеса. Это странно, так как Network Monitor может, по данным Spiceworks, отслеживать до 1000 устройств без ущерба для замедления.

Тем не менее, пользователи начального уровня с ограниченными требованиями к мониторингу найдут Spiceworks Network Monitor для быстрой настройки и удобства работы. Пользователи с более жесткими требованиями к мониторингу и малым предприятиям, которые хотят иметь возможность настроить свою настройку мониторинга на свои точные требования, должны выбрать более настраиваемый инструмент сетевого мониторинга.

### Eltex.EMS

Eltex.EMS - это централизованная система управления сетевым оборудованием производства Eltex Enterprise Ltd.

Система Eltex.EMS представляет собой архитектуру на основе клиент-сервер. Один сервер доступа представлен веб-интерфейсом, который позволяет осуществлять независимое одновременное управление различными сетевыми элементами.

Подсистема управления автоматизацией (Northbound Interface) позволяет взаимодействовать с EMS с превосходным поставщиком OSS / BSS. В частности, он позволяет стыковать с биллинговой системой оператора с использованием открытых стандартизованных протоколов, что позволяет автоматизировать рутинные операции, такие как массовое отключение абонентских портов в случае неоплаченных услуг и последующее повторное подключение в случае оплаты, а также изменение конфигурации устройств.

Возможность установки пакетов для используемого типа оборудования позволяет оптимально загружать серверные ресурсы оператора, отслеживать данные о состоянии сети в реальном времени и эффективное использование человеческих ресурсов. Таким образом, вы можете получить наиболее эффективный выход из системы.

Система Eltex.EMS может быть представлена в виде стандартных Linux-дистрибутивов двух основных популярных форматов rpm и deb, а также в виде готового ISO-образа, который может быть быстро установлен как на реальном хосте, так и на супервизоре виртуальных машин. Это позволяет быстро развернуть систему мониторинга в кратчайшие сроки.

## Вывод раздела 1

Рассмотренные программные средства мониторинга сетей условно можно поделить на 3 части:

* Бесплатные с закрытым исходным кодом
* Условно бесплатные с ограниченным периодом использования и возможностями
* Платные, цена которых для нас неприемлема

Некоторые из них (Eltex.EMS, UniFi controller) работают только с оборудованием своих производителей. Другие перегружены функционалом и сложны в настройке. В связи с этим было принято решение о создании своей системы мониторинга.

1. **ИССЛЕДОВАНИЕ МЕТОДОВ И СРЕДСТВ ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ МОНИТОРИНГА КОМПЬЮТЕРНЫХ СЕТЕЙ**
   1. **Описание средств реализации программного продукта**
      1. **Обоснование выбора платформы разработки**

В качестве платформы разработки была выбрана виртуальная машина Java(JVM).

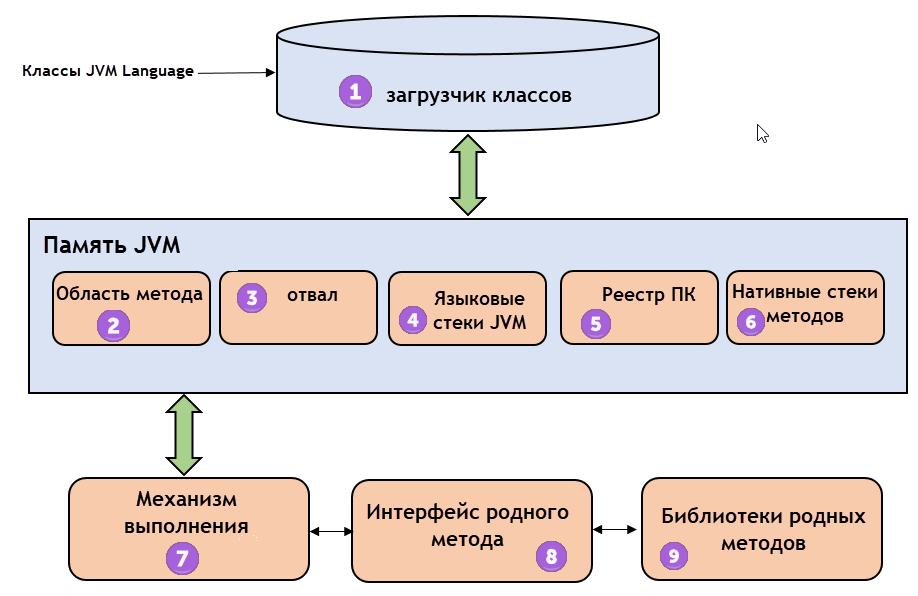
JVM – это механизм, обеспечивающий среду выполнения для управления кодом Java или приложениями. Он преобразует байт-код Java в язык машин. JVM является частью JRE (среда запуска Java). Это означает, что виртуальная машина Java

В других языках программирования компилятор создает машинный код для конкретной системы. Однако компилятор Java создает код для виртуальной машины, известной как виртуальная машина Java.

* Во-первых, код Java выполняется в байт-код. Этот байт-код интерпретируется на разных машинах
* Между хост-системой и Java-источником Bytecode является языком-посредником.
* JVM отвечает за выделение памяти.



Архитектура JVM содержит загрузчик классов, область памяти, механизм выполениня и т.д.



* Загрузчик классов - подсистема, используемая для загрузки файлов классов. Он выполняет три основные функции: Загрузка, связывание и инициализация.
* Область методов JVM хранит структуры классов, такие как метаданные, постоянный пул времени выполнения и код методов.
* Куча. Все объекты, связанные с ними переменные экземпляра и массивы хранятся в куче. Эта память является общей и разделяется между несколькими потоками.
* Стеки языков JVM хранит локальные переменные, и это частичные результаты. Каждый поток имеет свой собственный стек JVM, создаваемый одновременно с созданием потока. Новый кадр создается всякий раз, когда вызывается метод, и он удаляется, когда процесс вызова метода завершен.
* Реристры ПК хранит адрес инструкции виртуальной машины Java, которая в настоящее время выполняется. В Java каждый поток имеет отдельный регистр ПК.
* Нативные стеки методов содержат инструкцию нативного кода, зависящую от исходной библиотеки. Он написан на другом языке вместо Java.
* Механизм исполнения - тип программного обеспечения, используемого для тестирования оборудования, программного обеспечения или полных систем. Механизм выполнения теста никогда не несет никакой информации о тестируемом продукте.
* Интерфейс на основе метода — это программная среда. Он позволяет Java-коду, запущенному в JVM, вызывать библиотеки и собственные приложения.
* Нативные библиотеки методов - это сборник Native Libraries (C, C ++), который необходим движку Execution Engine.

Благодаря такой архитектуре jvm позволяет запускать программы, написанные на любом языке программирования имеющим возможность компиляции в byte код. Так же это позволяет запускать один и тот же код на разных аппаратных платформах без перекомпиляции.

### Обоснование выбора языка программирования

В качестве языка программирования для разработки системы мониторинга был выбран язык Clojure.

Clojure – это функциональный, LISP-подобный язык программирования общего назначения, с мощной системой макросов. Главными особенностями языка являются:

* Разработка в REPL-цикле
* Функции являются объектами первого класса
* Параллельное программирование с поддержкой агентной системы, транзакционной памяти и динамических переменных
* Компиляция в Java байт код
* Тесная интеграция с JVM платформой
* Поддержка ленивых последовательностей
* Неизменяемые типы данных

## Особенности технической реализации системы мониторинга компьютерных систем

## Особенности построения современных компьютерных сетей на примере ФГБОУ "МДЦ "Артек"

## Выводы по второму разделу

# РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ МОНИТОРИНГА ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ ЛАГЕРЯ «АРТЕК»

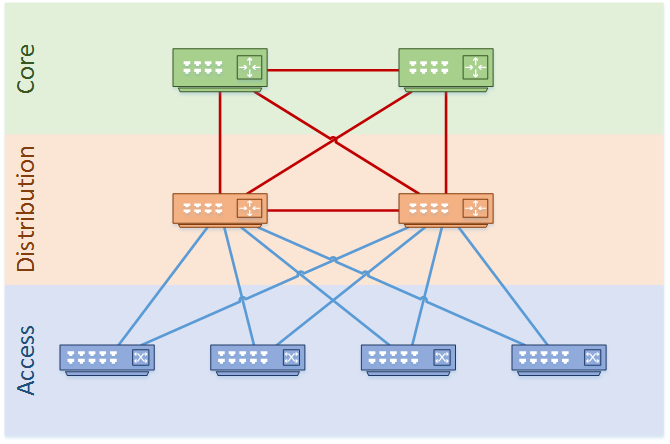
## Описание архитектуры построения компьютерной сети ФГБОУ "МДЦ "Артек"

Сеть Артека нуждается в хорошей структуре. Устройства перемещаются по сети. Пользователи проводят телефонные звонки и видеоконференции в режиме реального времени. И, как правило, гостевой WiFi защищен. Иерархическая модель сети позволяет удобно управлять всеми возложенными на нее задачами.

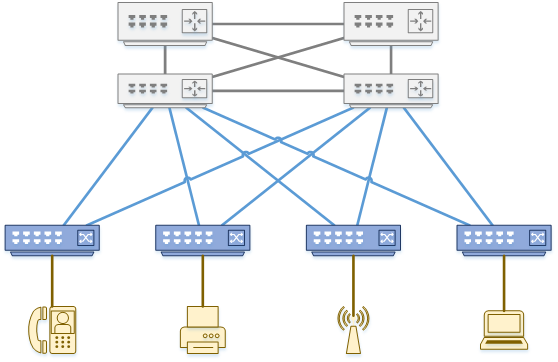
Однако эта модель не нова. Он уже много лет используется и зарекомендовал себя. Эта модель дает нам справочную топологию, которая разделяет сеть на слои. Каждый слой имеет свои собственные цели и функции. Это помогает сделать сеть масштабируемой, стабильной и детерминированной. Следуя этой модели, администраторы могут поддерживать сеть как самым естественным образом.

Как показано ниже, иерархическая сетевая модель использует три уровня. Это уровни Core, Distribution и Access.

Требования и функции каждого уровня различны. Чтобы решить эту проблему, подход к дизайну каждого уровня должен быть индивидуальным.



Уровень доступа предназначен для удовлетворения потребностей подключения к конечным устройствам. Уровень доступа - это край сети, в которой подключены хост-устройства. Это включает в себя рабочие станции и принтеры. Устройства, которые расширяют сеть, например телефоны и точки доступа, также присоединяются сюда. Это слой, на котором администратор проводит большую часть своего времени. Это очень функциональный слой, так как он должен поддерживать множество разных конечных точек.

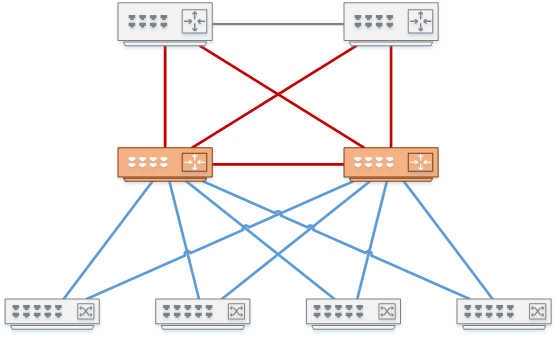


Этот уровень обычно включает в себя такие сервисы как

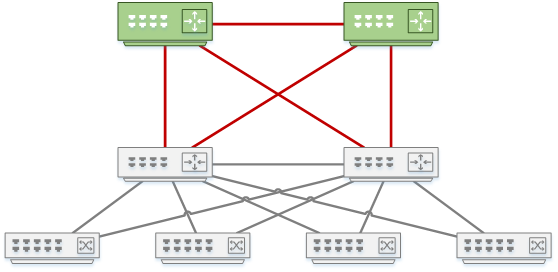
* Открытие и настройка - CDP и LLDP
* Безопасность и сетевая идентификация - 802.1x, безопасность портов, отслеживание DHCP, DAI, Dource Guard, Idendity Based Network Services и Web-auth
* Признание приложения - маркировка QoS, контроль, очередность, NBAR
* Управление сетью - протоколы маршрутизации, связующее дерево, DTP, LACP, UDLD, FlexLink
* Физическая инфраструктура - PoE

Уровень распределения - многоцелевой слой. В частности, ему необходимо агрегировать трафик уровня доступа и перенаправить его в остальную сеть. В сети, вероятно, много коммутаторов уровня доступа. Каждый из этих переключателей имеет восходящие линии связи с коммутаторами уровня распространения. Многие конечные устройства через коммутаторы уровня доступа агрегируются на уровне распространения.

Использование коммутаторов распределения для совокупного трафика логически создает «блоки распределения». Рассмотрим университетский городок с четырьмя зданиями. Каждое здание имеет два распределительных коммутатора и восемь переключателей уровня доступа. Каждое из этих зданий является распределительным блоком. Трафик может маршрутизировать между блоками распределения через основной уровень.



Уровень распределения обеспечивает точку демаркации между блоком и остальной частью сети. Это делает его хорошим местом для применения сетевых политик. Это также хорошая граница безопасности между уровнем доступа и остальной частью сети.



Уровень ядра – основной слой сети. Он соединяет все блоки распределения. Основной слой имеет одну цель и не требует много функций. Нет никаких политик безопасности, никакого QoS и никаких конечных точек. Ключевыми принципами проектирования основного слоя является то, что он должен быть быстрым, он должен быть всегда доступен, и он должен быть надежным. Крайне важно, что нет единственной точки отказа. Если произошел сбой, восстановление должно быть как можно быстрее. Ядро работает с чистыми каналами уровня 3 (маршрутизируемыми портами) и высоко настраиваемыми протоколами маршрутизации.