

Руководство по установке KometaWS

Содержание

| | |
|---|----|
| Сокращения и термины | 2 |
| Описание | 3 |
| Минимальные системные требования..... | 5 |
| Базовая архитектура продукта | 6 |
| Первичная установка..... | 8 |
| Подготовка к установке..... | 11 |
| Установка зависимостей и библиотек | 13 |
| Создание пользователей..... | 14 |
| Настройка правил межсетевого экрана..... | 14 |
| Установка и настройка сервиса замера пропускной способности вычислительной сети | 15 |
| Подготовка рабочих директорий и системы хранения..... | 16 |
| Получение и распаковка дистрибутива..... | 19 |
| Установка и настройка DB..... | 19 |
| Установка и настройка java..... | 22 |
| Установка и настройка фреймворка dcm4chee..... | 23 |
| Установка и настройка web приложения..... | 26 |
| Конфигурация планировщика..... | 29 |
| Конфигурация WEB сервера..... | 29 |
| Подготовка подсистемы виртуализации | 35 |
| Подготовка процедур автоматического запуска..... | 43 |
| Процедура активации лицензии | 50 |
| Получение сведений о текущей версии продукта | 51 |
| Обновление..... | 51 |

Сокращения и термины

DICOM (*Digital Imaging and Communications in Medicine*) - медицинский отраслевой стандарт создания, обмена, хранения и визуализации цифровых медицинских изображений.

DICOM узел - участник взаимодействия в DICOM сети, сущность способная к взаимодействию через ip сети, обладающая уникальной комбинацией IP адрес TCP порт AETitle, готовая к взаимодействию по протоколу DICOM с другими узлами.

PACS (*Picture Archiving and Communication System*) - система обмена и хранения DICOM файлов.

СУБД - Система управления базами данных(Database Management System) — совокупность программных и средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных

MWL- (*Modality Worklist*) DICOM сервис предназначенный для передачи списка заданий из внешнего источника на диагностическое оборудования для дальнейшего проведения исследования.

URL (Uniform Resource Locator - унифицированный указатель ресурса - это унифицированная форма обозначения индивидуального адреса ресурса в вычислительной сети.

Пространство хранения DICOM данных - комплекс средств предназначенных для надёжного хранения DICOM данных предоставляющий PACS серверу совместимый интерфейс для хранения файлов. Представляет собой специально зарезервированный для хранения медицинских данных объём на дисковой системе устройства, SAN, NAS с гарантированной пропускной способностью и объёмом в соответствии с регламентом учреждения.

WEB-сервер - программное обеспечение выполняющее роль сервера, принимает HTTP-запросы от клиентов и выдаёт HTTP-ответы, как правило, вместе с HTML-страницей и соответствующим запросу контентом

ОС (операционная система) - Службное программное обеспечение организующее работу прикладных программ, предоставляет программный интерфейс прикладному программному обеспечению, обеспечивает распределение ресурсов. Может работать в виртуальной среде.

Сервис(демон) - Программное обеспечение запускаемое операционной системой способное работать в фоновом режиме.

Виртуальная машина - программная (программно-аппаратная) система эмуляции программного и аппаратного окружения выполняющаяся на host платформе и предназначенная для запуска гостевых платформ.

Описание

KometaWS — это программно-аппаратный комплекс для просмотра и анализа медицинских изображений всех модальностей, полученных с диагностического оборудования в формате DICOM со встроенным инструментарием экспертного анализа и сервисом списка направлений на исследование.

KometaWS построена с учётом принципов открытой интеграции с медицинскими / радиологическими информационными системами и идеально подходит для экспертного анализа исследований медицинскими специалистами, систематизации и учета диагностических картин пациентов и их личных данных (DICOM, HL7).

Особенности системы KometaWS:

- Поддержка **KometaWS** всех модальностей (КТ, МРТ, УЗИ, ангиография, рентген, оцифровка, маммография, флюорограф, ЭКГ, эндоскопия, ПЭТ, ОФЭКТ и пр.)
- Совместимость с медицинским диагностическим оборудованием в соответствии с международным стандартам (DICOM 3.0, HL7)
- Неограниченное количество одновременно работающих пользователей
- Интеграция **KometaWS** в сетевую инфраструктуру учреждения в том числе с сторонними PACS-серверами
- Возможность подключения стороннего сервера ИИ для машинной обработки исследований в качестве дополнительного инструмента специалисту для анализа данных
- Сохранение в базе данных дополнительных изображений, текстовой информации, как в виде отдельных файлов, так и в виде дополнительных серий к существующим исследованиям
- Удобство администрирования и управления базой данных пациентов через веб-интерфейс
- Интеграция с внешними сервисами аутентификации
- Динамически масштабируемая система хранения исследований с возможностью «горячего» добавления объёма
- Бесплатная годовая техническая поддержка с обновлением ПО
- Очное обучение по взаимодействию с системой медицинских и прикладных специалистов

Особенности KometaWS:

- Приложение с графическим веб-интерфейсом для взаимодействия с данными пациентов и управления ими
- Безопасный доступ с разграничением пользовательских прав
- Отображение и компоновка рабочего списка в форме таблицы с настраиваемыми фильтрами данных для отображения и систематизации
- Поиск исследований по различным параметрам: ФИО, дата рождения, дата исследования, номер исследования, регистрационный номер, описание серии

- Поиск и отображение исследований за текущие сутки, за предыдущие сутки, за неделю
- Встроенный модуль врачебных заключений и ведения пациентов с возможностью работы с таблицами, формулами, шаблонами, электронной подписью врача, экспортом во внешние текстовые редакторы, выводом на печать, встроенная проверка правописания
- Экспорт и импорт исследований пациентов на прочие сетевые узлы по протоколу DICOM
- Импорт исследований внешних источников с возможностью указания своего AETitle
- Выбор нескольких исследований пациента для динамического анализа в режиме сравнения
- Отображение статуса исследования
- Предварительный просмотр и загрузка
- Удобная постраничная навигация
- Генерация уникальной WEB ссылки на исследование с указанием ограничения времени доступа
- Отображение предыдущих исследований пациента и их загрузка при открытии последнего
- Настройка протоколов открытия исследований при работе с несколькими мониторами
- Удаление и редактирование исследований пациентов и их данных (анонимизация)
- Добавление файлов в оригинальном формате либо в DICOM (дайкомизация)
- Гибкая настройка рабочего пространства интерфейса, включая выбор цветовой палитры
- Инструменты для измерения значений длины, величины вокселя, углов (в т.ч. угла Кобба), высоты и угла треугольника, отношения отрезков, объёма
- Анализ произвольной области интереса с измерением значений интенсивности, площади, дисперсии
- Управление списком результатов измерений в отдельной панели с навигацией и экспортом
- Инверсия и встроенный набор фильтров для изображений
- Инструмент точечной межсерийной синхронизации
- Мультипланарная реконструкция с возможностью регулирования толщины срезов
- Объемная реконструкция с набором протоколов визуализации: кости, кожа, кровь, воздух
- Режимы проекций максимальной (MIP), средней (Avg), минимальной интенсивности (MinIP)
- Ограничение области визуализации объемной реконструкции - инструмент куб
- Удаление произвольного участка из объемной реконструкции – инструмент скальпель
- Манипулирование объемной реконструкцией без потери качества
- Ограничение области визуализации реконструкции ломаными секущими
- Сегментирование выборочных структур с вычислением их характеристик
- Автоматическое удаление стола пациента
- Вывод исследования на печать, в том числе на DICOM-принтер
- Сравнение исследований в режиме МПР с синхронизацией

Дополнительные модули анализа KometaWS :

- ИНСТРУМЕНТЫ ДЛЯ АНАЛИЗА ИССЛЕДОВАНИЙ ПЭТ/КТ:
- ИНСТРУМЕНТЫ ДЛЯ АНАЛИЗА СОСУДОВ (КРИВОЛИНЕЙНАЯ МПР):
- ИНСТРУМЕНТЫ ДЛЯ АНАЛИЗА ЛЕГОЧНЫХ УЗЕЛКОВ:
- ИНСТРУМЕНТЫ ДЛЯ АНАЛИЗА ДАННЫХ КТ-ПЕРФУЗИИ ГОЛОВНОГО МОЗГА:
- ИНСТРУМЕНТЫ ДЛЯ ЭНДОАНАЛИЗА (ВИРТУАЛЬНАЯ ЭНДОСКОПИЯ):

Минимальные системные требования.

| | |
|--------------------|---------------------------------------|
| OS | Linux >=5.15, Windows >=8.1 |
| CPU | Core >= 4, L3 >= 6MB , freq >= 3.6GHz |
| CPU virtualization | VT-x |
| RAM | 8 GB * |
| HDD | SSD =>120GB |
| Display | 2x 1920:1080 |
| Browser | chrome(x64)>=92 , Firefox >= 115 |

Порты и протоколы

| | |
|------------|-------|
| ICMP | echo |
| TCP dicom | 11112 |
| TCP dicom | 11105 |
| TCP dicom | 104 |
| TCP dicom | 105 |
| TCP WEB | 443 |
| TCP WEB | 80 |
| TCP ssh | 22 |
| TCP iperf3 | 5202 |

Целевое назначение

Рабочая станция KometaWS для работы с цифровыми медицинскими изображениями предназначена для организации процесса просмотра и анализа медицинских изображений

Базовая архитектура продукта

Продукт состоит из нескольких объединённых в комплекс подсистем.

1 Подсистема визуального интерфейса. Подсистема предназначена для взаимодействия пользователей с потребительской и административной функциональностью продукта. Используется для :

- Просмотра и анализа исследований

- Манипуляции исследованиями
- Манипуляции пациентами
- Редактирования метаданных исследований
- Процедур экспорта импорт , отправки на печать.
- Администрирования.

2 База данных. Подсистема предназначена для хранения потребительской и служебной информации о исследованиях, о состоянии внутренних компонентов продукта, информации о пользователях. Является связующим звеном между подсистемой просмотра, подсистемой хранения информации и PACS фреймворком.

3 Система хранения данных. Подсистема предназначена для хранения полученных от DICOM узлов изображений.

4 PACS фреймворк. Подсистема предназначена для хранения обеспечения DICOM взаимодействия с другими узлами, медицинским оборудованием. Обеспечивает приём , передачу и хранение исследований на системе хранения данных,

Схема типового процесса работы с медицинскими изображениями.

Первичная установка

Первичная установка приложения разделена на логические этапы. Завершение каждого этапа сопровождается проверкой работоспособности затронутых компонентов приложения, такой подход позволяет относительно быстро найти проблему и не приступать к конфигурации следующего компонента пока не будут исправно работать все необходимые для него зависимости предыдущего этапа установки.

Установка на различные поддерживаемые дистрибутивы выполняется аналогичным способом но с использованием рекомендуемого в конкретном дистрибутиве пакетного менеджера .

Этапы установки приложения и ожидаемый результат.

1 Подготовка к установке

Действия: Конфигурация площадки на которой будет развёрнуто приложение, получение всей необходимой информации от администрации площадки, подготовка сетевой связности.

Результат : Корректно сконфигурированная площадка , сконфигурированная ОС, сконфигурированные работоспособные каналы связи. Сервер готов к установке приложения

2 Установка зависимостей и библиотек для работы PACS.

Действия: Получение необходимых зависимостей .

Результат : Все необходимые зависимости установлены.

3 Создание пользователей.

Действия: Создание пользователей.

Результат : Пользователи сервисов и служебные пользователи успешно созданы.

4 Настройка правил межсетевого экрана.

Действия: Конфигурация правил межсетевого экрана.

Результат : Межсетевой экран сконфигурирован в соответствии с минимальными требованиями продукта и требованиями информационной безопасности площадки размещения.

5 Настройка сервиса замера пропускной способности вычислительной сети.(необязательный шаг)

Действия: Конфигурация утилиты замера пропускной способности сети .

Результат : Ответчик утилиты замера пропускной способности работоспособен и запускаться автоматически.

6 Подготовка рабочих директорий и системы хранения.

Действия: Создаются рабочие каталоги приложений. Устанавливаются владельцы каталогов .

Результат : Все требуемые каталоги созданы. Владельцы каталогов имеют возможность манипуляции с каталогами в соответствии с привилегиями.

7 Получение и распаковка дистрибутива.

Действия: Получение файла дистрибутива и патчей. Распаковка. Установка привилегий.

Результат : Содержимое дистрибутива находится в штатных директориях с заданными привилегиями.

8 Установка и настройка DB

Действия: Конфигурация СУБД. Настройка доступа. Размещение шаблона схемы DB. Размещение шаблонной DB.

Результат : Работоспособный сервер СУБД корректно отвечает на запросы от заданного пользователя. База данных содержит схему и необходимые таблицы.

9 Установка и настройка java

Действия: Установка и конфигурация java в соответствии с требованиями площадки и имеющимся оборудованием.

Результат : Работоспособная подсистема java.

10 Установка и настройка фреймворка dcm4chee.

Действия: Установка и предварительная конфигурация фреймворка.

Результат : Фреймворк dcm4chee работает без ошибок, отвечает на эхо запросы и готов к взаимодействию с другими dicom узлами.

11 Установка и настройка web приложения.

Действия: Настройка web приложения и его компонентов.

Результат : Web приложение настроено, в журналах событий php ошибок не наблюдается.

12 Конфигурация планировщика.

Действия: Подготовка заданий планировщика .

Результат : Задания планировщика корректно сформированы, задания выполняются.

13 Конфигурация WEB сервера.

Действия: Конфигурация WEB сервера, настройка выдачи динамического и статического контента .

Результат : Сконфигурированный веб сервер корректно выдаёт необходимый контент. Работоспособна WEB страница продукта, пользователь может пройти процедуру аутентификации и воспользоваться основной функциональностью.

14 Подготовка подсистемы виртуализации.

Действия: Настройка библиотеки виртуализации. Получение распространяемого виртуального компонента PACS.

Результат : Библиотека виртуализации настроена и корректно функционирует. Виртуальный компонент PACS успешно стартует и доступен для основного приложения.

15 Подготовка процедур автоматического запуска.

Действия: Подсистемы продукта настраиваются для запуска в установленной последовательности, настраиваются реакции на события перезагрузки и отказа.

Результат : Подсистемы продукта корректно запускаются в установленной последовательности, корректно реагируют на события.

16 Процедура активации лицензии.

Действия: Запрос идентификатора установки, получение лицензионного ключа, установка лицензионного ключа .

Результат : Активация лицензии, общая работоспособность установленного продукта.

17 Установка и конфигурация сервиса MWL (необязательный шаг)

Действия: Конфигурация сервиса MWL в соответствии с требованиями регламента предприятия .

Результат : Сервис MWL успешно получает информацию о списках заданий на исследования от указанного источника, Успешно выдаёт список заданий на аппаратуру (предварительно корректно настроенную) или тестовому узлу.

1 Подготовка к установке

Получение сведений о настройках вычислительной сети

Продукт может использоваться непосредственно как группа приложений установленная на сервер так и группа приложений в среде виртуализации. В данном руководстве не рассматривается настройка среды виртуализации и каких либо методов оптимизации производительности в виртуальной среде . Установка продукта непосредственно на сервер или с использованием среды виртуализации не имеет принципиальных отличий за исключением особенностей настройки конкретной виртуальной среды.

В этом руководстве не рассматривается настройка сетевых соединений и коммутационного оборудования. Предполагается что специалист производящий установку обладает достаточной компетенцией, осведомлён о принципах работы вычислительной сети и информационной безопасности, имеет базовые представления о работе протокола DICOM и взаимодействии медицинского оборудования. Для получения дополнительных сведений о базовых принципах работы протокола DICOM и методах организации работы вычислительных сетей лечебного учреждения обратитесь к "Руководству администратора" раздел "сетевой аспект протокола DICOM" .

Если вы являетесь сотрудником лечебного учреждения и самостоятельно выполняете развёртывание системы - обязательно ознакомьтесь с документом "Руководство администратора" перед началом установки.

Используйте сервис DHCP для получения адресов системой PACS только в случае абсолютной уверенности в корректной и надёжной работе сервиса. Некорректная работа сервиса может привести как к недоступности PACS так и сбоям в работе медицинского оборудования.

Сведения о необходимых для работы системы протоколах и портах содержатся в разделе "Системные требования".

Примеры конфигурационных файлов и названия библиотек показаны для дистрибутива redhat 8.

В процессе установки может потребоваться доступ к сети интернет.

В процессе установки потребуются административные привилегии.

Предполагается что метод доступа для удалённой технической поддержки выбран заранее и оборудование и программное обеспечение сконфигурировано должным образом.

Перед началом процесса установки необходимо получить сведения о вычислительной сети учреждения.

- IP адрес
- Маска сети
- Основной шлюз (адреса шлюзов и таблица маршрутов в случае нескольких шлюзов)
- Сервер доменных имён
- DNS name для сервера
- ProxyServer (если используется)
- Атрибуты 802.1x (если используется)
- Настройки 802.3ad (если используется)

Проверка наличия административных привилегий.

Удостоверьтесь в наличии административных привилегий у пользователя, от имени которого приводится установка.

Настройка сетевых интерфейсов.

Выполните конфигурацию сетевых интерфейсов операционной системы в соответствии с полученными от администрации учреждения настройками.

- Сконфигурируйте IP адрес и сетевую маску.
- Сконфигурируйте адрес шлюза . (нескольких шлюзов и таблицу маршрутизации)
- Сконфигурируйте DNS клиент.
- Сконфигурируйте DNS имя узла в соответствии с полученными от администрации учреждения настройками.
- Сконфигурируйте 802.3ad если протокол используется
- Сконфигурируйте 802.1x если протокол используется

Выполните конфигурацию локального резолвера DNS, внесите в конфигурацию собственное доменное имя узла.

Выполните проверку доступности узла для указанного доменного имени.

```
ping pacs.services.local
```

Удостоверьтесь в доступности узла по указанному доменному имени узла.

Выполните проверку проверки доступности узла содержащего установочные файлы дистрибутива.

```
ping demo.kometa-pacs.info
```

Удостоверьтесь в доступности узла .

Выполните проверку проверки доступности любого готового к взаимодействию узла медицинского оборудования.

```
ping ct-002.diag.med.local
```

Удостоверьтесь в доступности узла.

Настройка SELINUX.

Выполните конфигурацию подсистемы SELINUX.

- Внесите изменения в конфигурационный файл /etc/selinux/config

```
sudo sed -i 's/^SELINUX=.*$/SELINUX=disabled/'
/etc/selinux/config
```
- Измените режим работы SELINUX

```
sudo setenforce 0
```

2 Установка зависимостей и библиотек необходимых для работы продукта.

Перечень необходимых программных компонентов для работы продукта :

- dcm4chee = 2.4
- PHP >= 7.4
- wine >= 7.5
- nginx >=1.18
- dcmtk >=3.6
- qemu-kvm >= 6.1
- postgresql >=12.1
- java >=1.6.0_45
- Драйвер видеоадаптера >=510.68.02 (устанавливается на компьютер)
- Драйвер принтера >=2.4.10 (устанавливается на компьютер)

Необходимые программные компоненты пред устанавливаются заранее.

Обновления версий драйверов не требуется .

Специальных требований к лицензиям драйверов не предъявляться .

Установка зависимостей

Выполните обновление репозитория.

```
sudo dnf clean all && dnf update -y
```

Выполните установку необходимых для работы библиотек

```
sudo dnf install -y iperf3 qemu-guest-agent sudo tmux ntfs-3g s3fs fbset libblkid-dev  
bridge-utils virt-manager qemu-kvm libvirt virt-install libguestfs-tools seabios binutils  
fuse lsof psmisc sysvinit-tools libnsl net-tools tcpdump ethtool curl wget nuttcp rsync  
pv lynx lshw sysstat ca-certificates python3 python3-pip nginx nginx-all-modules  
nginx-core nginx-filesystem nginx-mimetypes nginx-mod-http-geoip nginx-mod-http-  
image-filter nginx-mod-http-modsecurity nginx-mod-http-perl nginx-mod-http-push-  
stream nginx-mod-http-xslt-filter nginx-mod-mail nginx-mod-stream php php-cli php-  
common php-fpm php-json php-ldap php-mbstring php-mysqlnd php-opcache php-pdo  
php-pecl-zip php-process php-soap php-sodium php-xml dnf-automatic libtiff dcmtk  
CharLS php-mbstring php-pdo php-dom php-ldap php-mysql php-pecl-zip php-gd  
monit ImageMagick wine
```

3 Создание пользователей.

Создайте служебных и рабочих пользователей pacs support

```
sudo useradd -G adm,wheel -p upl_user_adm_pass -s /bin/bash pacs
```

```
sudo useradd -G adm,wheel -p upl_user_adm_pass -s /bin/bash support
```

4 Настройка правил межсетевого экрана.

В примере представлены базовые правила, настройка правил межсетевого экрана может потребовать согласование с политикой информационной безопасности вашего предприятия. Удостоверьтесь что правила межсетевого экрана удовлетворяют требованиям политики предприятия. За дополнительной информацией обратитесь к производителю программного обеспечения.

Создайте правило перенаправления с TCP порта 104 на TCP порт 11112

```
sudo iptables -t nat -A PREROUTING -p tcp --dport 104 -j REDIRECT --to-port 11112
```

```
iptables -t nat -I OUTPUT --src 0/0 --dst 127.0.0.1 -p tcp --dport 104 -j REDIRECT --  
to-ports 11112
```

Создайте группу правил для межсетевого экрана.

```
firewall-cmd --permanent --add-port=8080/tcp
firewall-cmd --permanent --add-port=104/tcp
firewall-cmd --permanent --add-port=11112/tcp
firewall-cmd --permanent --add-port=22/tcp
firewall-cmd --permanent --add-port=5201/tcp
firewall-cmd --permanent --add-port=5201/udp
```

Задействуйте созданные правила.

```
firewall-cmd --reload
systemctl enable firewalld
systemctl start firewalld
```

5 Установка и настройка сервиса замера пропускной способности вычислительной сети.

Создайте файл юнита systemd для запуска утилиты замера.

```
sudo touch /etc/systemd/system/iperfd.service
```

Создайте директорию для pid файла

```
sudo mkdir /run/iperf3d
```

Назначите владельца директории pid файла

```
sudo chown -R pacs:pacs /run/iperf3d
```

Выясните расположение исполняемого файла утилиты

```
which iperf3
```

запишите путь где расположена утилита.

Внесите конфигурацию в файл юнита systemd для запуска утилиты замера. (можно скопировать в терминал текст от начала команды до метки EoF или выполнить действия вручную)

```
cat > /etc/systemd/system/iperfd.service <<- EoF
```

```
[Unit]
```

```
Description=iPerf3_Service
```

```
After=network.target
```

```
[Service]
```

```
Type=forking
```

```
PIDFile=/run/iperf3d/iperf3d.pid
User=pacs
Group=pacs
ExecStart=$supl_iperf3Path -s -D -I /run/iperf3d/iperf3d.pid
ExecReload=/bin/kill -HUP $MAINPID
Restart=always
[Install]
WantedBy=multi-user.target
EoF
```

Запустите сервис замера пропускной способности.

```
systemctl daemon-reload
```

```
systemctl enable iperfd
```

```
systemctl start iperfd
```

Удостоверьтесь что сервис работает

```
systemctl is-active iperfd.service
```

6 Подготовка рабочих директорий и системы хранения.

Подготовка системы хранения данных для работы с исследованиями.

Выбор системы хранения.

Подсистема хранения данных **KometaWS** разработана с учётом возможности использования различных вариантов хранения данных в том числе и с использованием объектного хранилища S3. Метод подключения системы хранения может быть любым.

Выбор архитектуры системы хранения зависит от конкретных потребностей учреждения:

- Количество диагностического оборудования.
- Количество производимых исследований за период.
- Регламентированное учреждением содержимое исследования (отчёты полученных доз , дополнительные серии)
- Требуемое время хранения исследований
- Требуемая скорость доступа к исследованию.

Выбор стратегии хранения данных осуществляется на этапе проектирования системы. Наиболее частым вариантом для небольшого учреждения (не более 2 аппаратов СТ) является использование raid массива 5 или 6 уровня в зависимости от технической возможности.

В данном руководстве рассматривается общий случай для типового учреждения с использованием встроенного в сервер raid массива.

Подготовка хранилища.

Убедитесь что дополнительное программное обеспечение в операционной системе для вашего raid установлено и исправно работает.

Убедитесь в том что вы контролируете raid из операционной системы.

Выполните конфигурацию RAID контроллера для использования массива raid 6 с учётом:

- Предполагаемого максимального объёма для хранения в требуемый промежуток времени.
- Динамического использования пространства, если ваш raid контроллер позволяет выделять место в процессе работы без снижения производительности.
Выделяйте место для хранения конкретного промежутка времени например один год - для следующего года создайте другое пространство, это позволит легко перенести архивные данные на более медленное хранилище и высвободить место для дальнейшего использования.
- Выделения минимум одного диска для hotswap для создаваемого массива .

Внимание !

Обязательно наличия BBU (Battery Backup Unit) при использовании write-back caching.

Удостоверьтесь в корректности работы cache памяти вашего контроллера многократными тестами корректности чтения записи больших объёмов данных.

Проведите тестирование производительности дисковой подсистемы.

Подготовка рабочей директории.

Подготовка директории хранения исследований

- Убедитесь что файловая система содержащая пространство хранения DICOM доступно и корректно настроено.
- Создайте точку монтирования в файловой системе

```
mkdir -p /mnt/dicom/
```

- Выполните монтирование пространства хранения в созданную директорию.

```
sudo mount -o umask=0022,gid=1000,uid=1000 /dev/{storage} mnt/dicom/
```

- Назначьте владельца директории.

```
sudo chown -R pacs:pacs /mnt/dicom/
```

- Настройте автоматическое монтирование пространства хранения во время загрузки операционной системы в зависимости от регламентированного на предприятии метода монтирования.

- Выполните перезагрузку операционной системы или контейнера и убедитесь в успехе операций:

Файловая система содержащая пространство хранения DICOM.

```
sudo lsblk
```

Пользователь pacs является владельцем директории и способен производить создавать файлы и директории.

```
su pacs
```

```
touch /mnt/dicom/testdir/testfile
```

```
exit
```

Подготовка рабочей директории PACS.

Не используйте для рабочей директории PACS пространство для хранения исследований.

- Создайте рабочую директорию PACS

```
sudo mkdir -p /usr/share/pacs
```

- Назначьте владельца директории.

```
sudo chown -R pacs:pacs /usr/share/pacs
```

- Убедитесь что пользователь pacs является владельцем директории и способен производить создавать файлы и директории.

```
su pacs
```

```
touch /usr/share/pacs/testdir/testfile
```

```
exit
```

7 Получение и распаковка дистрибутива.

- Убедитесь в наличии доступа к сети интернет.

- Перейдите в рабочую директорию PACS.
`cd /usr/share/pacs`
- Загрузите дистрибутив .
`wget -q --show-progress -O ./pacs.tgz
http://demo.kometa-pacs.info/kometa/CloudWebServer/Resources/kometa_redos.tgz`
- Распакуйте дистрибутив в директорию.
`sudo -u pacs -i bash -c "tar -xzf ./pacs.tgz --directory ./"`
- Измените владельца директории и содержимого.
`sudo chown -R pacs:pacs /usr/share/pacs/*`
- Убедитесь что пользователь pacs является владельцем рабочей директории и содержимого.
`ls -hals /usr/share/pacs/`

8 Установка и настройка DB

Продукт может использовать в качестве СУБД MariaDB и PostgreSQL.

Если если регламент вашего предприятия подразумевает использование выделенного сервера или группы серверов для хранения баз данных - согласуйте установку базы данных для pacs с администратором баз данных вашего предприятия.

В качестве примера показана установка СУБД на локальный сервер совместно с PACS.

Установка и настройка MariaDB .

- Выполните установку MariaDB.
`sudo dnf install -y mariadb-server mariadb-common`
- Запустите сервер баз данных.
`sudo systemctl start mariadb.service`
- Убедитесь в том что сервер базы данных работает
`sudo systemctl status mariadb.service`
- Обеспечите автоматический запуск СУБД
`sudo systemctl enable mariadb.service`
- Выполните настройку MariaDB

Выполните группу команд (можно скопировать в терминал текст от начала команды до метки ЕоМ или выполнить действия вручную) :

```
sudo mysql_secure_installation <<-EoM
```

```
у  
пароль суперпользователя СУБД ($password)  
пароль суперпользователя СУБД ($password)  
у  
у  
у  
у  
EoM  
##
```

Внесите изменения в конфигурационный файл `/etc/my.cnf.d/server-utf.cnf` (можно скопировать в терминал текст от начала команды до метки EoF или выполнить действия вручную)

```
sudo cat > /etc/my.cnf.d/server-utf.cnf <<- EoF
```

```
[server]  
character-set-server=utf8  
collation-server=utf8_general_ci  
EoF
```

- Выполните перезапуск СУБД

```
sudo systemctl restart mariadb.service
```
- Убедитесь в работоспособности СУБД.

```
systemctl status mariadb.service
```
- Создайте схему данных и сконфигурируйте привилегии для базы данных PACS.

(\$password) - пароль суперпользователя СУБД
(\$dbuser пользователь БД) - рабочий пользователь БД

```
sudo mysql -uroot -p -e($password) " CREATE SCHEMA  
( название БД); GRANT ALL ON pacodb.* TO ($dbuser  
пользователь БД)@localhost IDENTIFIED BY '($dbuser  
пользователь БД)'; FLUSH PRIVILEGES;"
```

- Импортируйте базу данных PACS.

```
sudo mysql -uroot -p($password) pacsdb <
/usr/share/pacs/sql/create.mysql
```

- Убедитесь в работоспособности и корректной настройке СУБД.

```
sudo mysql -uroot -p($password) -e "select * from
information_schema.user_privileges;" | grep ($dbuser пользователь
БД)
```

Ожидаемый результат - вывод имени рабочего пользователя БД

Установка и настройка PostgreSQL .

- Выполните установку PostgreSQL.

```
sudo dnf install -y postgresql-server php-sqlsrv
```

- Инициализируйте PostgreSQL.

```
sudo postgresql-setup initdb
```

- Запустите сервер баз данных.

```
sudo systemctl start postgresql.service
```

- Убедитесь в том что сервер базы данных работает

```
sudo systemctl status postgresql.service
```

- Обеспечите автоматический запуск СУБД

```
sudo systemctl enable postgresql.service
```

- Убедитесь в работоспособности СУБД.

```
systemctl status postgresql.service
```

- Создайте схему данных и сконфигурируйте привилегии для базы данных PACS.

(\$password) - пароль суперпользователя СУБД

(\$dbuser пользователь БД) - рабочий пользователь БД

```
su postgres -c 'createdb -h 127.0.0.1 -U
postgres pacsdb'
```

- Импортируйте базу данных PACS.

```
psql -h 127.0.0.1 -U postgres pacsdb
<
```

```
/usr/share/pacs/dcm4chee.postgres/sql/create.psql
```

9 Установка и настройка java

Выполните установку java.

- Разрешите выполнение установщика

```
sudo chmod +x /usr/share/pacs/distr/jdk-6u45-linux-x64.bin
```
- Запустите установщик

```
sudo /usr/share/pacs/distr/jdk-6u45-linux-x64.bin
```
- Скопируйте распакованные файлы

```
sudo cp -r /usr/share/pacs/jdk1.6.0_45 /usr/share/pacs/java
```

```
sudo cp -r /usr/share/pacs/jdk1.6.0_45 /usr/lib/jvm/
```
- Измените владельца директории и содержимого.

```
chown chown -R pacs:pacs /usr/share/pacs/java/*
```

```
chown chown -R pacs:pacs /usr/lib/jvm/jdk1.6.0_45/*
```
- Измените задействованную по умолчанию версию java.

```
alternatives --set java /usr/lib/jvm/jdk1.6.0_45/bin/java
```

```
alternatives --set javac /usr/lib/jvm/jdk1.6.0_45/bin/javac
```

```
update-alternatives --install /usr/bin/java java /usr/lib/jvm/jdk1.6.0_45/bin/java 1
```

```
update-alternatives --install /usr/bin/javac javac /usr/lib/jvm/jdk1.6.0_45/bin/javac 1
```

```
update-alternatives --install /usr/bin/jar jar /usr/lib/jvm/jdk1.6.0_45/bin/jar 1
```
- Убедитесь что версия java соответствует установленной

```
java -version 2>&1 | grep -i "java version"
```

Выполните настройку java.

Внесите изменения в конфигурацию java. /usr/share/pacs/bin/run.conf

```
sed -i "s/Xms[0-9]*m/Xms1024m/; s/Xmx[0-9]*m/Xmx2048m/; s/XX:MaxPermSize=[0-9]*m/XX:MaxPermSize=128m/" /usr/share/pacs/bin/run.conf
```

```
sed -i "s/gcInterval=[0-9]*s/gcInterval=1600000/g;" /usr/share/pacs/bin/run.conf
```

```
sed -i '/sun.rmi.dgc.client.gcInterval/ s/"/ -XX:+CMSClassUnloadingEnabled  
-XX:+UseConcMarkSweepGC \"/ /usr/share/pacs/bin/run.conf
```

10 Установка и настройка фреймворка dcm4chee.

Установка dcm4chee.

- Создайте файл юнита systemd для запуска фреймворка dcm4chee.

```
sudo touch /etc/systemd/system/pacs_init.service
```
- Внесите конфигурацию в файл юнита systemd для запуска фреймворка dcm4chee. (можно скопировать в терминал текст от начала команды до метки EoF или выполнить действия вручную)

```
cat > /usr/lib/systemd/system/$1 <<- EoF  
  
[Unit]  
Description=PACS Server (Dcm4chee-2.18)  
DefaultDependencies=no  
After=syslog.target network.target network-online.target  
After=mariadb.service  
After=postgresql-13.service postgresql-13-slave.service  
After=postgresql-14.service postgresql-14-slave.service  
Wants=network-online.target  
  
[Service]  
Type=simple  
User=pacs  
Group=pacs  
ExecStart=/usr/share/pacs/bin/run.sh  
ExecStop=/usr/share/pacs/bin/stop.sh  
Restart=always  
RestartSec=10s  
TimeoutSec=30  
  
[Install]  
WantedBy=multi-user.target
```

EoF

- Обновите конфигурацию systemd
`sudo systemctl daemon-reload`
- Запустите сервис фреймворка dcm4chee
`sudo systemctl start pacs_init.service`
- Убедитесь в работоспособности сервиса dcm4chee
`sudo systemctl status pacs_init.service`
- Обеспечите автоматический запуск СУБД
`sudo systemctl enable mariadb.service`
- Убедитесь в отсутствии ошибок в работе сервиса dcm4chee
`tail -f /usr/share/pacs/server/default/log/server.log`
Ожидаемое поведение - наличие сообщений о запуске сервера ,
отсутствие сообщений о ошибках
- Выполните проверку доступности DICOM сервера .(Раздел "Инструментарий управления и тестирования")
- Выполните проверку корректности работы API для dcm4chee.(Раздел "Инструментарий управления и тестирования")
`/usr/share/pacs/bin/twiddle.sh -u admin -p admin get --
noprefix jboss.system:type=Server`
Ожидаемый результат - вывод версии фреймворка.

Настройка dcm4chee.

- Измените AE PACS сервера в соответствии с запланированной схемой.
`/usr/share/pacs/bin/twiddle.sh -u admin -p admin invoke
dcm4chee.archive:service=AE DCM4CHEE ($ назначенный
AE PACS)`
*За дополнительной информацией о понимании работы AE обратитесь к
"Руководству администратора"*
- Измените путь хранилища изображений в соответствии с запланированной схемой.
`/usr/share/pacs/bin/twiddle.sh -u admin -p admin set
dcm4chee.archive:service=FileSystemMgt,group=ONLINE_S
TORAGE DefaultStorageDirectory /mnt/dicom/`

```
/usr/share/pacs/bin/twiddle.sh -u admin -p admin invoke  
dcm4chee.archive:service=FileSystemMgt,group=ONLINE_S  
TORAGE addRWFileSystem mnt/dicom/
```

```
/usr/share/pacs/bin/twiddle.sh -u admin -p admin invoke  
dcm4chee.archive:service=FileSystemMgt,group=ONLINE_S  
TORAGE updateFileSystemStatus mnt/dicom/ RW+
```

- Если есть необходимость дублировать исследования на другой PACS внесите конфигурацию для DICOM узла получателя

За дополнительной информацией о понимании работы DICOM узлов обратитесь к "Руководству администратора"

```
/usr/share/pacs/bin/twiddle.sh -u admin -p admin invoke  
dcm4chee.archive:service=AE addAE EXT_PACS 172.22.14.253 11112 ""  
iusePaID iuseAsN autoconf autoconf - - add_via_rmi "" st_name inst_name  
dep_name false false
```

11 Установка и настройка web приложения.

Установка библиотек необходимых для работы web приложения .

- Получите и установите необходимые для работы библиотеки.

```
wget http://rpms.remirepo.net/enterprise/7/remi/x86_64/oniguruma5php-6.9.8-  
1.el7.remi.x86_64.rpm
```

```
sudo rpm -Uvh oniguruma5php-6.9.8-1.el7.remi.x86_64.rpm
```

```
wget https://download-ib01.fedoraproject.org/pub/epel/7/x86_64/Packages/l/  
libzstd-1.5.5-1.el7.x86_64.rpm
```

```
sudo rpm -Uvh libzstd-1.5.5-1.el7.x86_64.rpm
```

Установка менеджера зависимостей PHP

- Установите менеджер зависимостей PHP.

```
curl -sS https://getcomposer.org/installer | php
```

```
sudo mv composer.phar /usr/local/bin/composer
```

```
sudo chmod +x /usr/local/bin/composer
```

Установка ioncube.

- Получите ioncube и распакуйте в каталог /usr/lib64/php/modules

```
wget
https://downloads.ioncube.com/loader_downloads/ioncube_loaders_lin_x86-64.zip
```

```
unzip ioncube_loaders_lin_x86-64.zip
```

```
sudo mv ./ioncube/ioncube_loader_lin_7.4.so /usr/lib64/php/modules/
```

- Установите привилегии выполнения для ioncube.

```
sudo chmod +x /usr/lib64/php/modules/ioncube_loader_lin_7.4.so
```

- Создайте конфигурационный файл /etc/php.d/00-ioncube.ini

```
sudo echo 'zend_extension=/usr/lib64/php/modules/ioncube_loader_lin_7.4.so'
> /etc/php.d/00-ioncube.ini
```

- Перезапустите сервис php-fpm.

```
sudo systemctl restart php-fpm.service
```

- Убедитесь в отсутствии ошибок и предупреждений в работе PHP.

```
sudo systemctl status php-fpm.service
```

```
sudo journalctl -e -u php-fpm.service
```

Получение и распаковка дистрибутива.

- Получите дистрибутив продукта.

```
wget -O dest72.tgz http://craft.kometa-pacs.info/mdistr.php?file=dest72.tgz
```

```
wget -O vendor.tgz http://craft.kometa-pacs.info/mdistr.php?file=vendor.tg
```

- Создайте рабочие каталоги web приложения

```
sudo mkdir /var/www/html/kmtpacs
```

```
sudo mkdir /var/www/html/kmtpacs/storage/app/dicom_cache
```

```
sudo mkdir /var/www/html/kmtpacs/storage/app/dicom_cache/tmp
```

```
sudo mkdir /var/www/html/kmtpacs/storage/app/dicom_cache/third
```

```
sudo mkdir /var/www/html/kmtpacs/storage/app/dicom_cache/imagelist
```

- Распакуйте содержимое дистрибутива в созданные каталоги.

```
tar -xzf dest72.tgz -C /var/www/html/kmtpacs/ --strip-components 1
```

```
tar -xzf vendor.tgz -C /var/www/html/kmtpacs/
```

- Выполните корректировку привилегий.

```
find /var/www/html/kmtpacs/storage -type d -exec chmod g+w+s {} \;
```

```
find /var/www/html/kmtpacs/storage -type f -exec chmod g+w {} \;
```

```
sudo chown -Rc pacs:nginx
```

```
sudo chown -R pacs:nginx /var/www/html/kmtpacs/storage/app/dicom_cache
sudo chmod -R 775 /var/www/html/kmtpacs/storage/app/dicom_cache
```

Конфигурация PHP.

- Внесите изменения в конфигурационный файл `.etc/php-fpm.d/www.conf`

```
sed -i -e 's/user = apache/user = nginx/; s/group = apache/group = nginx/;
s/listen = 127.0.0.1:9000/listen = \var\run\php7.2-fpm.sock/;
s/\.php4 \.php5 \.php7/.php4 \.php5 \.php7 \.ashx \.asmx/;
s;/security.limit_extension/security.limit_extension/; s;/listen.owner =
nobody/listen.owner = nginx/; s;/listen.group = nobody/listen.group =
nginx/' /etc/php-fpm.d/www.conf
```
- Внесите изменения в конфигурационный файл `/etc/php.ini`

```
sed -i 's/upload_max_filesize = 2M/upload_max_filesize =
2048M/g' /etc/php.ini

sed -isystemctl restart php-fpm 's/max_file_uploads =
20/max_file_uploads = 20000/g' /etc/php.ini
```
- Выполните перезапуск сервисов

```
systemctl restart nginx
systemctl restart php-fpm
```
- Убедитесь в работоспособности PHP.

```
systemctl status nginx
systemctl status php-fpm
php -v
```

Конфигурация WEB приложения.

- Скопируйте шаблон конфигурационного файла `.env` в рабочую директорию WEB приложения.

```
cp /var/www/html/kmtpacs/.env.example /var/www/html/kmtpacs/.env
```
- Внесите корректировки в конфигурационный файл `.env`

```
cd /var/www/html/kmtpacs/ && sed -i -e
's/DB_DATABASE=homestead/DB_DATABASE=pacsdb/g;
s/DB_USERNAME=homestead/DB_USERNAME=pacs/g;
s/DB_PASSWORD=secret/DB_PASSWORD=pacs/g' .env

echo "ABSURL='http://100.69.0.100:9191/abs.php'" >>
/var/www/html/kmtpacs/.env
```

- Выполните процедуру генерации ключа приложения.
`cd /var/www/html/kmtpacs/ && php artisan key:generate`
- Перезапустите PHP.
`systemctl restart php-fpm`

12 Конфигурация планировщика.

- Внесите изменения в конфигурационный файл планировщика/etc/monit.d/pacs.conf
(можно скопировать в терминал текст от начала команды до метки EoF или
выполнить действия вручную)

```
cat <<"EOM" > /etc/monit.d/pacs.conf
```

```
check process pacs-data-commands-queue-1 with pidfile  
/var/www/html/kmtpacs/storage/app/monit/queue-pacs-data-commands-  
1.pid
```

```
start program = "/var/www/html/kmtpacs/monit/queue.sh start pacs-data-  
commands 1"
```

```
as uid kmt and gid nginx
```

```
stop program = "/var/www/html/kmtpacs/monit/queue.sh stop pacs-data-  
commands 1"
```

```
check process pacs-data-commands-queue-2 with pidfile  
/var/www/html/kmtpacs/storage/app/monit/queue-pacs-data-commands-  
2.pid
```

```
start program = "/var/www/html/kmtpacs/monit/queue.sh start pacs-data-  
commands 2"
```

```
as uid kmt and gid nginx
```

```
stop program = "/var/www/html/kmtpacs/monit/queue.sh stop pacs-data-  
commands 2"
```

```
EOM
```

- Перезапустите monit.
`systemctl restart monit.service`
`sudo monit start all`

13 Конфигурация WEB сервера.

- Внесите изменения в конфигурационный файл веб сервера для приложения .
/etc/nginx/conf.d/kometa.conf (можно скопировать в терминал текст от начала команды до метки EoF или выполнить действия вручную)

```
cat <<"EOM" > /etc/nginx/conf.d/kmtpacs.conf

server {

    listen                80;

    root    /var/www/html/kmtpacs/public;
    index index.php home.php index.html index.htm;
    client_max_body_size 2048m;
    charset utf-8;

    ### Section redirections
    #

    ### Replace argumets in URI
    #

    ### patientId > patId
    #
    if ($args ~* "(.*)"patientId)(.*)") {
        set $args "$1patId$3";
        return 301 $uri?$args;
    }

    ### AccNum > accessionNo
    #
    if ($args ~* "(.*)"AccNum)(.*)") {
        set $args "$1accessionNo$3";
        return 301 $uri?$args;
    }

    #

    #####
}
```

```
### Section proxy forward for check license
#
# location ~ ^/rview.php {
#     proxy_pass http://100.69.0.100:9191;
#     proxy_set_header Host      $host;
#     proxy_set_header X-Real-IP  $remote_addr;
#     proxy_set_header X-Forwarded-For
$proxy_add_x_forwarded_for;
# }
#
#####
### Section with access from all IP (no restrictions)
#
location /mans {
    autoindex_localtime on;
    charset utf-8;
    autoindex on;
}

location /soft {
    autoindex_localtime on;
    charset utf-8;
    autoindex on;
}
#
#####

### Section for real IP view(bykey+wgviewer) from crypted links,
generated by web-server (no restrictions)
#
location /bykey {
    try_files $uri $uri/ /index.php?$query_string;
```

```
fastcgi_pass unix:/run/php7.4-fpm.sock;
fastcgi_param SCRIPT_FILENAME
$document_root$fastcgi_script_name;
include fastcgi_params;
fastcgi_read_timeout 6000s;
}

location /wgviewer {
    try_files $uri $uri/ /index.php?$query_string;

    fastcgi_pass unix:/run/php7.4-fpm.sock;
    fastcgi_param SCRIPT_FILENAME
$document_root$fastcgi_script_name;
    include fastcgi_params;
    fastcgi_read_timeout 6000s;
}
#
#####

### Section for real IP view(api+smartphone) from app, accessed
through API (no restrictions)
#
location /api {
    try_files $uri $uri/ /index.php?$query_string;

    fastcgi_pass unix:/run/php7.4-fpm.sock;
    fastcgi_param SCRIPT_FILENAME
$document_root$fastcgi_script_name;
    include fastcgi_params;
    fastcgi_read_timeout 6000s;
}
#
#####
```

```
### Section with access at checked IP (with restrictions)
#
### Check access from networks
#
location @check_access {
    allow 0.0.0.0/0;
#   allow 10.0.0.0/8;
#   allow 172.16.0.0/12;
#   allow 192.168.0.0/16;
#   allow 127.0.0.1;
#   deny all;

    try_files $uri $uri/ /index.php?$query_string;

    fastcgi_pass unix:/run/php7.4-fpm.sock;
    fastcgi_param SCRIPT_FILENAME
$document_root$fastcgi_script_name;
    include fastcgi_params;
    proxy_read_timeout 6000s;
    fastcgi_read_timeout 6000s;
}
### Root section all access through '@check_access'
#
location / {
    try_files $uri $uri/ /index.php?$query_string @check_access;
}
location /admin/ {
    try_files $uri $uri/ /index.php?$query_string @check_access;
}
location ~ \.ashx$ {
    try_files $uri $uri/ /getwebpbdata?$query_string @check_access;
```

```
    }  
    location ~ /\.wsl$ {  
        try_files $uri $uri/ @check_access;  
    }  
    location ~ /\.php$ {  
        try_files $uri $uri/ /index.php?$query_string @check_access;  
  
        fastcgi_pass unix:/run/php7.4-fpm.sock;  
        fastcgi_param SCRIPT_FILENAME  
$document_root$fastcgi_script_name;  
        include fastcgi_params;  
        fastcgi_read_timeout 6000s;  
    }  
}  
EOM
```

- Перезапустите WEB сервер.
systemctl restart nginx

14 Подготовка подсистемы виртуализации.

Убедитесь что ваш CPU поддерживает технологию виртуализации.

- Убедитесь что функциональность в виртуализации активирована.
cat /proc/cpuinfo | egrep "(vmx|svm)"

Убедитесь в работоспособности сервиса обеспечения виртуализации.

- Убедитесь что сервис в виртуализации активирован и работает без ошибок.
sudo systemctl start libvirt

```
sudo systemctl enable libvirt
```

```
sudo systemctl status libvirt
```

Конфигурация виртуальной вычислительной сети гипервизора и гостевой машины.

- Внесите изменения в конфигурационный файл /usr/share/libvirt/networks/default.xml.(можно скопировать в терминал текст от начала команды до метки EoF или выполнить действия вручную)

```
sudo cat > /usr/share/libvirt/networks/default.xml <<- EOM
```

```
<network>
  <name>default</name>
  <uuid>34fec45c-09c2-456b-a79b-0090ef38e3e8</uuid>
  <forward mode='nat'>
    <nat>
      <port start='1024' end='65535'>
    </nat>
  </forward>
  <bridge name='virbr0' stp='on' delay='0'>
  <mac address='52:54:00:d6:bc:f9'>
  <ip address='100.69.0.1' netmask='255.255.255.0'>
    <dhcp>
      <range start='100.69.0.2' end='100.69.0.254'>
      <host mac='52:54:00:f1:04:09' name='license' ip='100.69.0.100'>
    </dhcp>
  </ip>
</network>
```

```
EOM
```

Запуск конфигурации виртуальной сети.

- Запустите виртуальную выделительную сеть.

```
sudo virsh net-destroy default
```

```
sudo virsh net-undefine default
```

```
sudo virsh net-define /usr/share/libvirt/networks/default.xml
```

```
sudo virsh net-autostart default
```

```
sudo virsh net-start default
```

- Убедитесь что виртуальная вычислительная сеть работает .

Конфигурация ip forward для виртуальной вычислительной сети .

- Создайте директорию для хранения конфигурации qemu.

```
sudo mkdir /etc/libvirt/hooks
```

- Внесите изменения в конфигурационный файл /usr/share/libvirt/networks/default.xml.(можно скопировать в терминал текст от начала команды до метки EoM или выполнить действия вручную)

```
sudo cat > /etc/libvirt/hooks/qemu <<- EoM
```

```
#!/bin/bash
```

```
if [ "${1}" = "ROcentos7.0" ]; then
```

```
    # Update the following variables to fit your setup
```

```
    GUEST_IP=100.69.0.100
```

```
    GUEST_PORT=9191
```

```
    HOST_PORT=9090
```

```
    if [ "${2}" = "stopped" ] || [ "${2}" = "reconnect" ]; then
```

```
        /sbin/iptables -D FORWARD -o virbr0 -p tcp -d $GUEST_IP --dport \
        $GUEST_PORT -j ACCEPT
```

```
        /sbin/iptables -t nat -D PREROUTING -p tcp --dport $HOST_PORT -
        j DNAT --to $GUEST_IP:$GUEST_PORT
```

```
    fi
```

```
    if [ "${2}" = "start" ] || [ "${2}" = "reconnect" ]; then
```

```
        /sbin/iptables -I FORWARD -o virbr0 -p tcp -d $GUEST_IP --dport \
        $GUEST_PORT -j ACCEPT
```

```
        /sbin/iptables -t nat -I PREROUTING -p tcp --dport $HOST_PORT -j
        DNAT --to $GUEST_IP:$GUEST_PORT
```

```
    fi
```

```
fi
```

```
EoM
```

- Разрешите выполнение файла /etc/libvirt/hooks/qemu .

```
sudo chmod +x /etc/libvirt/hooks/qemu
```

- Активируйте ip forward для systemd .


```
sudo touch /etc/sysctl.d/99-sysctl.conf
sudo echo "" >> /etc/sysctl.d/99-sysctl.conf
sudo sed -i
'/^net.ipv4.ip_forward={h;s/=.*=/1/};${x;/^$/s//net.ipv4.ip_forward=1;/H};x
}' /etc/sysctl.d/99-sysctl.conf
sudo sysctl -p /etc/sysctl.d/99-sysctl.conf
```
- Перезапустите libvirtd .


```
sudo systemctl restart libvirtd
```
- Убедитесь что libvirtd работает .


```
sudo systemctl status libvirtd
```
- Создайте каталог для образа виртуальной машины .


```
sudo mkdir -p /kvm
```
- Получите образ виртуальной машины .


```
wget -O ~/ROcentos7.0 http://craft.kometa-pacs.info/mdistr.php?
file=ROcentos7.0
sudo mv ~/ROcentos7.0 /kvm/
```
- Внесите изменения в конфигурационный файл /kvm/domxmlROperm.xml.(можно скопировать в терминал текст от начала команды до метки ЕоМ или выполнить действия вручную)


```
sudo cat > /kvm/domxmlROperm.xml <<- EOF
<domain type='kvm'>
  <name>ROcentos7.0</name>
  <uuid>85ef22c5-d115-4e71-9a21-5d78c3c67e90</uuid>
  <metadata>
    <libosinfo:libosinfo
xmlns:libosinfo="http://libosinfo.org/xmlns/libvirt/domain/1.0">
      <libosinfo:os id="http://redhat.com/rhel/7.0"/>
```

```
</libosinfo:libosinfo>
</metadata>
<memory unit='KiB'>2097152</memory>
<currentMemory unit='KiB'>2097152</currentMemory>
<vcpu placement='static'>2</vcpu>
<os>
  <type arch='x86_64' machine='pc'>hvm</type>
  <boot sudo systemctl restart libvirtddev='hd'/>
  <bootmenu enable='yes'/>
</os>
<features>
  <acpi/>
  <apic/>
</features>
<cpu mode='custom' match='exact' check='partial'>
  <model fallback='allow'>core2duo</model>
</cpu>
<clock offset='utc'>
  <timer name='rtc' tickpolicy='catchup'/>
  <timer name='pit' tickpolicy='delay'/>
  <timer name='hpet' present='no'/>
</clock>
<on_poweroff>destroy</on_poweroff>
<on_reboot>restart</on_reboot>
<on_crash>destroy</on_crash>
<pm>
<suspend-to-mem enabled='no'/>
  <suspend-to-disk enabled='no'/>
</pm>
<devices>
  <emulator>/usr/bin/qemu-system-x86_64</emulator>
  <disk type='file' device='disk'>
```

```
<driver name='qemu' type='qcow2'/>
<source file='/kvm/ROcentos7.0'/>
<target dev='vda' bus='virtio'/>
<address type='pci' domain='0x0000' bus='0x00' slot='0x07'
function='0x0'/>
</disk>
<disk type='file' device='cdrom'>
<driver name='qemu' type='raw'/>
<target dev='hda' bus='ide'/>
<readonly/>
<address type='drive' controller='0' bus='0' target='0' unit='0'/>
</disk>
<controller type='pci' index='0' model='pci-root'/>
<controller type='ide' index='0'>
<address type='pci' domain='0x0000' bus='0x00' slot='0x01'
function='0x1'/>
</controller>
<controller type='virtio-serial' index='0'>
<address type='pci' domain='0x0000' bus='0x00' slot='0x06'
function='0x0'/>
</controller>
<interface type='network'>
<mac address='52:54:00:f1:04:09'/>
<source network='default'/>
<model type='virtio'/>
<link state='up'/>
<address type='pci' domain='0x0000' bus='0x00' slot='0x03'
function='0x0'/>
</interface>
<serial type='pty'>
<target type='isa-serial' port='0'>
<model name='isa-serial'/>
</target>
```

```
</serial>
<console type='pty'>
  <target type='serial' port='0'/>
</console>
<channel type='unix'>
  <target type='virtio' name='org.qemu.guest_agent.0'/>
  <address type='virtio-serial' controller='0' bus='0' port='1'/>
</channel>
<channel type='spicevmc'>
  <target type='virtio' name='com.redhat.spice.0'/>
  <address type='virtio-serial' controller='0' bus='0' port='2'/>
</channel>
<input type='tablet' bus='usb'>
  <address type='usb' bus='0' port='1'/>
</input>
<input type='mouse' bus='ps2'/>
<input type='keyboard' bus='ps2'/>
<graphics type='spice' autoport='yes'>
  <listen type='address'/>
  <image compression='off'/>
</graphics>
<sound model='ich6'>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x04'
function='0x0'/>
</sound>
<video>
  <model type='qxl' ram='65536' vram='65536' vgamem='16384' heads='1'
primary='yes'/>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x02'
function='0x0'/>
</video>
<redirdev bus='usb' type='spicevmc'>
  <address type='usb' bus='0' port='2'/>
```

```

</redirdev>

<memballoon model='virtio'>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x08'
function='0x0' />
</memballoon>

<rng model='virtio'>
  <backend model='random'>/dev/urandom</backend>

  <address type='pci' domain='0x0000' bus='0x00' slot='0x09'
function='0x0' />
</rng>
</devices>
</domain>

```

EOF

- Импортируйте созданную конфигурацию в libvirt .


```
sudo virsh define /kvm/domxmlROperm.xml
```
- Активируйте импортированную в libvirt конфигурацию .


```
sudo virsh autostart ROcentos7.0
```
- Внесите корректировки в конфигурацию автозапуска виртуальных машин .


```
sudo sed -i "s/#ON_SHUTDOWN=suspend/ON_SHUTDOWN=shutdown;/
s/#ON_BOOT=start/ON_BOOT=start;/
s/#SHUTDOWN_TIMEOUT=300/SHUTDOWN_TIMEOUT=300 /g;"
/etc/sysconfig/libvirt-guests
```
- Внесите изменения в конфигурацию NetworkManager если он используется .


```
sudo touch /etc/NetworkManager/conf.d/mac.conf
sudo echo -ne "[connection]\nethernet.cloned-mac-address=preserve\n" >>
/etc/NetworkManager/conf.d/mac.conf
sudo echo -ne "[keyfile]\nunmanaged-devices=interface-name:virbr0;interface-
name:virbr0-nic;interface-name:vnet0\n" >>
/etc/NetworkManager/NetworkManager.conf
```

- Выполните перезагрузку libvirt и убедитесь что виртуальная машина автоматически запускается при старте .

```
sudo systemctl restart libvirtd
```

```
sudo virsh list
```

- Убедитесь что виртуальная машина доступна .

```
ping 100.69.0.100
```

15 Подготовка скриптов управления PACS.

Скрипт запуска PACS

- Внесите изменения в файл /usr/bin/pacs-start.sh . (можно скопировать в терминал текст от начала команды до метки EoF или выполнить действия вручную)

```
cat <<"EOM" > /usr/bin/pacs-start.sh
```

```
#!/bin/bash
```

```
echo;
```

```
echo "Проверяем на запуск с правами sudo";
```

```
echo;
```

```
##-- Проверяем на запуск с правами sudo
```

```
if env | grep ^SUDO_USER >/dev/null 2>&1
```

```
then
```

```
    echo "Запущено с sudo от имени "$SUDO_USER";
```

```
    echo;
```

```
else
```

```
    echo "Недостаточно прав для выполнения!";
```

```
    echo;
```

```
    echo "Выполните скрипт от имени админа например: $ sudo  
./kmtfull-start.sh";
```

```
    echo;
```

```
        exit 1;
```

```
fi
```

```
echo;
```

```
echo 'Start postgresql services';
```

```
systemctl start postgresql-13.service
sleep 1;
systemctl start postgresql-13-slave.service
sleep 1;
echo;
echo 'Start pacs_init.service';
systemctl start pacs_init.service;
sleep 3;
echo;
echo 'Start php-fpm.service';
systemctl start php-fpm.service;
sleep 1;
echo;
echo 'Start nginx.service';
systemctl start nginx.service;
sleep 1;
echo;
echo 'Start crond.service';
systemctl start crond.service;
echo;
echo 'Все сервисы запущены';
echo;
```

EOM

Скрипт перезапуска PACS

- Внесите изменения в файл /usr/bin/pacs-restart.sh (можно скопировать в терминал текст от начала команды до метки EoF или выполнить действия вручную)

```
cat <<"EOM" > /usr/bin/pacs-restart.sh
#!/bin/bash
echo;
echo "Проверяем на запуск с правами sudo";
echo;
```

```
#!/bin/bash
#-- Проверяем на запуск с правами sudo
if env | grep ^SUDO_USER >/dev/null 2>&1
then
    echo "Запущено с sudo от имени "$SUDO_USER";
    echo;
else
    echo "Недостаточно прав для выполнения!";
    echo;
    echo "Выполните скрипт от имени админа например: $ sudo
./kmtfull-restart.sh";
    echo;
    exit 1;
fi
echo;
echo 'Stop crond.service';
systemctl stop crond.service
sleep 1;
echo;
echo 'Stop nginx.service';
systemctl stop nginx.service;
fuser -k 80/tcp;
sleep 1;
echo;
echo 'Stop php-fpm.service';
systemctl stop php-fpm.service;
sleep 2;
echo;
echo 'Stop pacs_init.service';
systemctl stop pacs_init.service;
sleep 3;
echo;
echo 'Stop postgresql services';
```

```
systemctl stop postgresql-13-slave.service
sleep 1;
systemctl stop postgresql-13.service
sleep 1;)
cat <<"EOM" > /usr/bin/pacs-restart.sh
echo;
echo 'Все сервисы остановлены';
echo;
echo;
echo 'Start postgresql services';
systemctl start postgresql-13.service
sleep 1;
systemctl start postgresql-13-slave.service
sleep 1;
echo;
echo 'Start pacs_init.service';
systemctl start pacs_init.service;
sleep 3;
echo;
echo 'Start php-fpm.service';
systemctl start php-fpm.service;
sleep 1;
echo;
echo 'Start nginx.service';
systemctl start nginx.service;
sleep 1;
echo;
echo 'Start crond.service';
systemctl start crond.service;
echo;
echo 'Все сервисы запущены';
echo;
```

ЕОМ

Скрипт остановки PACS

- Внесите изменения в файл /usr/bin/pacs-restart.sh (можно скопировать в терминал текст от начала команды до метки EOF или выполнить действия вручную)

```
cat <<"ЕОМ" > /usr/bin/pacs-restart.sh

#!/bin/bash

echo;

echo "Проверяем на запуск с правами sudo";

echo;

#-- Проверяем на запуск с правами sudo
if env | grep ^SUDO_USER >/dev/null 2>&1
then
    echo "Запущено с sudo от имени "$SUDO_USER;
    echo;
else
    echo "Недостаточно прав для выполнения!";
    echo;
    echo "Выполните скрипт от имени админа например: $ sudo
./kmtfull-restart.sh";
    echo;
    exit 1;
fi

echo;

echo 'Stop crond.service';
systemctl stop crond.service

sleep 1;

echo;

echo 'Stop nginx.service';
systemctl stop nginx.service;

fuser -k 80/tcp;
```

```
sleep 1;
echo;
echo 'Stop php-fpm.service';
systemctl stop php-fpm.service;
sleep 2;
echo;
echo 'Stop pacs_init.service';
systemctl stop pacs_init.service;
sleep 3;
echo;
echo 'Stop postgresql services';
systemctl stop postgresql-13-slave.service
sleep 1;
systemctl stop postgresql-13.service
sleep 1;
echo;
echo 'Все сервисы остановлены';
echo;
echo;
echo 'Start postgresql services';
systemctl start postgresql-13.service
sleep 1;
systemctl start postgresql-13-slave.service
sleep 1;
sleep 1;
echo;
echo 'Start pacs_init.service';
systemctl start pacs_init.service;
sleep 3;
echo;
echo 'Start php-fpm.service';
systemctl start php-fpm.service;
```

```
sleep 1;
echo;
echo 'Start nginx.service';
systemctl start nginx.service;
sleep 1;
echo;
echo 'Start crond.service';
systemctl start crond.service;
echo;
echo 'Все сервисы запущены';
echo;
EOM
```

16 Процедура активации лицензии.

- Убедитесь в доступности ip интерфейса виртуальной машины развёрнутой на сервере с уставленным продуктом
- Подключитесь к виртуальной машине через SSH по адресу 100.69.0.100

```
ssh root@100.69.0.100
```
- Перейдите в каталог /var/www/html/kmtpacs/public

```
cd /var/www/html/kmtpacs/public
```
- Получите запрос идентификатора установки.

```
php genlic.php
```
- Скопируйте содержимое файла rrlc.txt .

```
cat rrlc.txt
```
- Передайте идентификатор разработчику программного обеспечения.
- Получите от разработчика программного обеспечения лицензию.
- Поместите содержимое лицензии в файл rrlicense.txt

```
echo"содержимое_лицензии" > rrlicense.txt
```

Получение сведений о текущей версии продукта.

Получение сведений о текущей версии продукта.

- Убедитесь в доступности ip интерфейса виртуальной машины развёрнутой на сервере с уставленным продуктом
- Подключитесь к виртуальной машине через SSH по адресу 100.69.0.100

```
ssh root@100.69.0.100
```
- Перейдите в каталог /var/www/html/kmtpacs/public

```
cd /var/www/html/kmtpacs/public
```
- Получите сведения о текущей версии.

```
php gen.php --ver
```

Обновление PACS

Запуск процедур обновления должен происходить от имени пользователя- владельца файлов каталога /var/www/html/ kmtpacs

Удостоверьтесь что вам известен пользователь владелец каталогов и файлов рабочей директории PACS

```
ls -l /var/www/html/ kmtpacs
```

Подготовка к процессу обновления.

Остановка выполнения заданий crond.

- Закомментируйте задания связанные с продуктом.

```
crontab -e
```

Закомментировать - поставить знак диез перед строкой задания и сохранить изменения.

Пример

```
## * * * * * cd /var/www/html/kmtpacs/ && php artisan schedule:run >> /dev/null  
2>&1
```

Сохраните изменения

Остановка заданий сервиса monit.

- Остановите задания monit.

```
sudo monit stop all
```

Дождитесь остановки заданий.

- Остановите сервис monit
`sudo systemctl stop monit`
- Убедитесь что сервис успешно остановлен.
`sudo systemctl status monit`

Получение пакета обновлений.

- Перейдите в рабочий каталог продукта `/var/www/html/kmtpacs/`
`cd /var/www/html/kmtpacs/`
- Сделайте резервную копию файла `client.php` .
`mkdir -p ~/backup/client/`
`mv client.php ~/backup/client/`
`cp .env ~/backup/client/`

Всегда фиксируйте значения специально настроенных переменных в документации вашего предприятия , следите за актуальностью зафиксированных значений.

- Получите пакет обновлений.
`wget -O client.php http://craft.kometa-pacs.info/mdistr.php?file=client.php`

Запуск процесса обновления

- Убедитесь что текущая директория соответствует рабочей директории продукта.
`/var/www/html/kmtpacs/`
`ls -l /var/www/html/kmtpacs/`
- Убедитесь что текущий пользователь является владельцем каталогов и файлов рабочей директории продукта
`whoami`
- Запустите процесс обновления
`php client.php`

Возможно использование дополнительных параметров:

Адрес прокси (пример `--проху=127.0.0.1:1080`)

Группа владельца PHP-файлов (по-умолчанию `nginx --file-group=www-data`)

Путь к рабочей директории (обязателен слеш/ в конце пути)

`--app-dir=/var/www/html/kmtpacs/`

```
php client.php --proxy=127.0.0.1:1080 --file-group=www-data
--app-dir=/var/www/html/kmtpacs/
```

- В процессе обновления будет предложено сделать резервную копию.
Do you confirm these changes ? (y - yes / n -no / b - backup and yes)
Следует выполнить резервное копирование и нажать "b"
- Дождитесь окончания процесса обновления.
- Оцените разницу между эталонными значениями файла .env.example и кастомизированными для вашего предприятия значениями .env .При необходимости внесите новые переменные.
- Запустите менеджер пакетов php
composer install
- Запустите утилиту artisan с ключом migrate
php artisan migrate
- Скопируйте файлы
scp public/rview.php root@100.69.0.100:/var/www/html/
scp public/checklic.php root@100.69.0.100:/var/www/html/

Завершение процесса обновления

- Запустите сервис monit
sudo systemctl start monit
sudo monit start all
- Активируйте задания связанные с продуктом в планировщике заданий crond.
crontab -e
Разкомментировать - удалить знак диез перед строкой задания и сохранить измерения.
Пример
* * * * * cd /var/www/html/kmtpacs/ && php artisan schedule:run >> /dev/null
2>&1
Сохраните изменения
- Убедитесь в работоспособности продукта .

Процедура деинсталляции.

Запуск процедур деинсталляции должен происходить от имени пользователя обладающего административными полномочиями .

- Выполните остановку работы демонов.

```
systemctl stop kometa_init.service
systemctl disable kometa_init.service
systemctl stop iperfd.service
systemctl disable iperfd.service
```

- Удалите каталоги

```
rm -rf /usr/lib/systemd/system/$upl_Dcm4SystemdUnitFileName
rm -rf /etc/systemd/system/iperfd.service
rm -rf /mnt/dicom/
rm -rf /usr/share/kometa/
rm -rf /run/iperf3d
```

- Удалите схему DB

```
$my_mysql -uroot -p$upl_RDBpass -e " DROP SCHEMA $upl_Dbname;"
systemctl stop mariadb.service && systemctl disable mariadb.service
```

Процедура деинсталляции продукта завершена.

Установка драйверов.

Установка драйверов видеоадаптера.

Перед установкой убедитесь что в вашем дистрибутиве подключен репозиторий содержащий обновления ядра.

Получите сведения о текущем драйвере видеоадаптера.

- `sudo -E lshw -numeric -C display`

При необходимости установите проприетарный драйвер видеоадаптера:

Получите последние обновления для вашего дистрибутива :

- `sudo dnf clean all && dnf update -y`
- `sudo apt update && apt update -y`

Выполните установку драйвера:

- `sudo -E dnf install nvidia-settings nvidia-kmod nvidia-persistenced nvidia-xconfig nvidia-modprobe xorg-x11-drv-nvidia xorg-x11-drv-nvidia-cuda-libs`

- `sudo -E apt install nvidia-settings nvidia-kmod nvidia-persistenced nvidia-xconfig nvidia-modprobe xorg-x11-driv-nvidia xorg-x11-driv-nvidia-cuda-libs`

Перезагрузите рабочую станцию .

Убедитесь в наличии корректно установленного драйвера

- `inxi -G`

Установка драйверов видеоадаптера завершена.

Установка драйверов принтера.

Пример установки драйвера принтера HP LaserJet Enterprise , установка других моделей принтеров выполняется аналогично.

Получите последние обновления для вашего дистрибутива :

- `sudo dnf clean all && dnf update -y`
- `sudo apt update && apt update -y`

Перейдите в сеанс пользователя root:

- `su -`

Установите компоненты и драйверы работы с принтером:

- `dnf install hplip-gui python3-distro`
- `apt install hplip-gui python3-distro`

Установка драйверов принтера завершена.

В случае сбоя в работе драйвера рекомендуется перезагрузить компьютер.

Список возможных ошибок в процессе установки обновления.

- `.version file doesnt exist or not readable`

Установите привилегии чтения и записи на файл `.version` в рабочей директории проекта `/var/www/html/kmtracs/` для рабочего пользователя `rac` , если файл отсутствует - обратитесь к разработчикам.

- `Current user is not a member of the group "group name"`

Следует добавить рабочего пользователя `рас` в группу владеющую `php` файлами каталога `/var/www/html/kmtпacs/`

- This update script has more recent version, download it to perform update

Используемая версия `client.php` уже устарела , получите последнюю версию установщика и перезапустите процесс обновления.

- PHP Fatal error: Allowed memory size of X bytes exhausted (tried to allocate Y bytes) in `/var/www/html/kmtпacs/client.php` on line Z

Необходимо увеличить лимит `memory_limit` в `/etc/php.ini`