

## 1. Назначение программы

ПУМ должна обеспечивать:

- Взаимодействие компонентов ККП, управление процессом передачи данных и управление сетевым распределением ключей ККП;
- Автоматизированную диагностику ККП, сбор статистики о её функционировании, состоянии ПКРК и ПДУ, в том числе с использованием мониторинга посредством протокола SNMP.
- ПУМ должна передавать исходные данные о всех событиях (выход значений параметров системы за пределы допустимого) с использованием брокеров сообщений в зонтичную систему промышленного партнёра. Формирование необходимых действий по событиям выполняет зонтичная система промышленного партнёра.
- ПУМ должна передавать исходные данные об использовании ресурсов с использованием брокера сообщений в систему биллинга промышленного партнёра. Формирование действий на основе данных ПУМ выполняет система биллинга промышленного партнёра.
- Выдачу команд в ПДУ на формирование и распределение квантово-защищенных ключей (КЗК) в автоматизированном режиме с помощью графической системы управления.
- Централизованный мониторинг заданных параметров ККП, с возможностью подачи сигналов в случае возникновения любых нештатных ситуаций в сфере ответственности ПУМ.
- Централизованное хранение программных компонент ПУМ (программы, конфигурационные файлы, описания) и

Инв. № подл.	
Подп. и дата	
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	

Изм.	Лист	№ докум.	Подп.	Дата		Лист
						4

мониторинговых данных, а также регулярное резервное копирование.

- Администрирование системы при размещении ПУМ в сети узлов системы.
- Реализацию API-интерфейсов к ПУМ в соответствии с п. 4.2.1.7 ЧТЗ ПУМ.

В целом ПУМ должна контролировать сервис безопасной передачи ключей кодирования и закодированных данных клиентов. Общая схема сервиса передачи ключей показана на рисунке 1.

Пример. Подключение сервиса по передаче ключа для Клиента

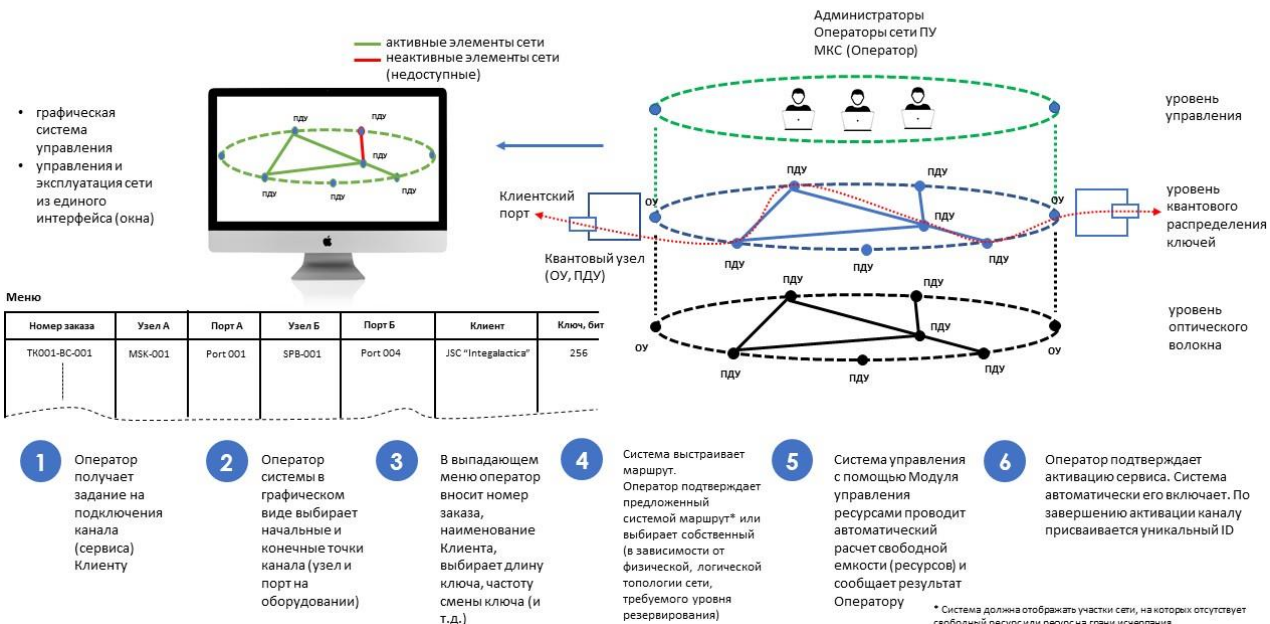


Рисунок 1 – Подключение сервиса передачи ключа кодирования Клиенту

Областью применения ПУМ является управление и мониторинг в рамках ККП.

Подп. и дата  
Инв. № дубл.  
Взам. инв. №  
Подп. и дата  
Инв. № подл.

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					5

## 2. Условия применения ПУМ

ПУМ должна устанавливаться как на выделенные серверы на основе архитектуры x86\_64, так и в виртуальной среде.

Средства вычислительной техники ПУМ должны поддерживать следующие режимы:

- Рабочий – плановое функционирование в соответствии с назначением;
- Тестовый (или сервисный) – проверочный режим, когда выполняется тестирование ПО и/или оборудования;
- Аварийный – режим, в котором некоторые части функционала могут быть недоступны, хотя в целом функционирование продолжается. Например, при выходе из строя одного из двух блоков питания функционирование может продолжаться.
- Загрузка ОС в локальный сервер управления – по сети.
- Управление и мониторинг в ПУМ – по каналу, защищенному СКЗИ.

Рабочие места администраторов модулей ПУМ должны содержать средства обеспечения защищенного канала управления и мониторинга для обмена информацией между АРМ администратора ПУМ и другими модулями ПУМ.

Каждый пользователь ПУМ должен иметь свой ключевой контейнер.

Аппаратные компоненты ПУМ должны функционировать в условиях для группы 4 ГОСТ 15150-69 исполнения УХЛ и группы 2 ГОСТ 21552-84, в частности:

- температуры окружающей среды: от +5°C до +40°C;
- относительной влажности до 80% при температуре +20°C;
- атмосферного давления от 84 до 107 кПа (от 630 до 800 мм рт. ст.) при отсутствии в воздухе хранилища агрессивных примесей.

Изн. № подл.	Подп. и дата
Взам. инв. №	Изн. № дубл.
Подп. и дата	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					6

### 3. Описание задачи ПУМ

#### 3.1. Подсистема управления

##### 3.1.1. Модуль управления ПДУ и КРК

Функционально модулем решаются следующие задачи:

- Взаимодействие с ПДУ и КРК;
- Управление ПДУ и КРК;
- Производится запись событий в электронный журнал.

##### 3.1.2. Модуль управления трафиком, топологией и технологическими параметрами ККП

Функционально модулем решаются следующие задачи:

- Обеспечивается доступ к сетевым и вычислительным устройствам системы для осуществления процесса управления ими;
- Обеспечивается безопасность передачи данных в соответствии с уровнем необходимой защищенности;
- Формируется топология сети и вычислительных устройств системы в представлении, соответствующем контексту использования, в достаточной мере обеспечивающем технологический процесс;
- Производится автоматический расчет емкости сети передачи данных с оценкой возможности построения канала передачи пользовательских данных в необходимом объеме;
- Производится оценка технической возможности подключения новых потребителей без ухудшения параметров и уровня обслуживания для подключённых ранее потребителей;
- Реализуется выдача диагностических сообщений;
- Производится запись событий в электронный журнал.

Изм.	Лист	№ докум.	Подп.	Дата	Изм.	Лист	№ докум.	Подп.	Дата	Изм.	Лист	№ докум.	Подп.	Дата

## 3.2. Подсистема мониторинга

### 3.2.1. Модуль мониторинга ПДУ и КРК

Модуль мониторинга ПДУ и КРК решает задачи осуществления автоматической диагностики, мониторинга и формирования предупреждений в реальном времени по следующим событиям:

- превышение порога QBER с подозрением на попытку несанкционированного доступа к квантовой сети;
- нарушение работоспособности оборудования ПКРК и ПДУ.

Для решения этих задач решено использовать заимствованную систему Zabbix. Все полученные данные мониторинга хранятся в базе данных Zabbix'a. Для отслеживания отклонения данных от приемлемых значений используются Zabbix триггеры. При срабатывании триггера создается событие и выполняются действия по уведомлению о данном событии.

В качестве языка разработки дополнений к программному обеспечению использовался python в среде Linux. Для передачи данных мониторинга ПДУ и КРК на Zabbix сервер использовался модуль Zabbix'a для python «py-zabbix». Для формирования списка аварий использовалась заимствованная система GLPI, которая получает сведения об авариях из Zabbix с помощью программного пакета для python «glpi-sdk-python».

### 3.2.2. Модуль мониторинга оборудования ККП

Модуль мониторинга оборудования ККП решает задачи осуществления автоматической диагностики и мониторинга следующей информации:

- температуры внутри аппаратных модулей узлов ПКРК;
- результаты контроля работоспособности вентиляторов и источников питания;
- параметров распределения вычислительных ресурсов процессорного блока;

Изн. № подл.	Подп. и дата
Взам. инв. №	Изн. № дубл.
Подп. и дата	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата
------	------	----------	-------	------

Лист

8

- результатов тестирования программных компонентов узлов ПКРК;
- Скорость генерации квантовых ключей между каждой парой узлов;
- QBER между каждой парой узлов;
- Частота обновления КЗК между парой узлов;
- иной информации.

Для решения этих задач решено использовать заимствованную систему Zabbix. Все полученные данные мониторинга хранятся в базе данных Zabbix’а. Для отслеживания отклонения данных от приемлемых значений используются Zabbix триггеры. При срабатывании триггера создается событие и выполняются действия по уведомлению о данном событии.

В качестве языка разработки дополнений к программному обеспечению использовался python в среде Linux. Для передачи данных мониторинга оборудования ККП на Zabbix сервер использовался модуль Zabbix’а для python «py-zabbix». Для формирования списка аварий использовалась заимствованная система GLPI, которая получает сведения об авариях из Zabbix с помощью программного пакета для python «glpi-sdk-python».

Модуль мониторинга оборудования ККП взаимодействует с внешней системой (ВС), которая предназначена для обеспечения единого непрерывного жизненного цикла управления технологическими сетями передачи данных.

Модуль мониторинга оборудования ККП обеспечивает передачу следующих данных во ВС:

- передачу событий на основе данных мониторинга в привязке к конфигурационным единицам (событие – это любой выход значений параметров любого компонента за допустимые пределы и /или расхождение в составе оборудования в ПУМ и ВС для формирования аварийного сообщения.

Модуль мониторинга оборудования ККП обеспечивает прием следующих данных из ВС:

Изн. № подл.	Подп. и дата	Взам. инв. №	Изн. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					9

- прием данных о результатах обработки событий в привязке к инцидентам и работам для информирования администратора ПУМ. ВС направляет в модуль мониторинга данные о всех работах и/или инцидентах, связанных с событиями, которые модуль мониторинга оборудования ранее отправил в ВС.

Для взаимодействия с ВС используется API интерфейс через брокер сообщений Kafka (служебный модуль коммутации данных) и Apache ServiceMix.

### 3.3. Подсистема технического учёта

#### 3.3.1. Модуль учета оборудования

Модуль учета оборудования решает следующие задачи:

- сбор данных учета оборудования;
- заполнение места расположения, координат, номеров коммерческих заказов и прочей дополнительной информации;
- составление базы знаний по возникающим проблемам;
- создание объектов, загруженных из ВС с возможностью внесения описания.

Для решения данных задач решено использовать заимствованную систему GLPI. Для обнаружения состава оборудования ККП и его загрузки в GLPI использовался плагин Fusioninventory.

В качестве языка разработки дополнений к программному обеспечению использовался python в среде Linux. Для создания объектов в GLPI, загруженных из ВС, использовался программный пакет для python «glpi-sdk- python».

Модуль учета оборудования взаимодействует с ВС, которая предназначена для обеспечения единого непрерывного жизненного цикла управления технологическими сетями передачи данных. Компоненты,

Изн. № подл.	Подп. и дата	Взам. инв. №	Изн. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					10

оборудование, соединения компонентов описываются во ВС. Все описания из ВС реплицируются в модуле учета оборудования. Модуль учета оборудования постоянно отслеживает реальный состав оборудования и соединений с базой оборудования и соединений и в случае расхождения формировать событие в модуле мониторинга оборудования ККП.

Модуль учета оборудования обеспечивает передачу следующих данных во ВС:

- регулярно передачу инвентаризационных данных, полученных с оборудования.

Модуль учета оборудования обеспечивает прием следующих данных из ВС:

- прием данных о типах, моделях и экземплярах конфигурационных единиц, соединениях между конфигурационными единицами.

Для взаимодействия с ВС используется API интерфейс через брокер сообщений Kafka (служебный модуль коммутации данных ВС) и Apache ServiceMix.

### 3.3.2. Модуль учета логических ресурсов

Модуль учета логических ресурсов решает следующие задачи:

- сбор данных учета логических ресурсов;
- заполнение дополнительной информации;
- получение актуального статуса установленного программного обеспечения и лицензий;
- создание объектов, загруженных из ВС с возможностью внесения описания.

Инв. № подл.	Подп. и дата
Взам. инв. №	Подп. и дата
Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					11



Для решения данных задач решено использовать заимствованную систему GLPI. Для обнаружения состава логических ресурсов и его загрузки в GLPI использовался плагин Fusioninventory.

В качестве языка разработки дополнений к программному обеспечению использовался python в среде Linux. Для создания объектов в GLPI, загруженных из BC, использовался программный пакет для python «glpi-sdk- python».

### 3.4. Подсистема взаимодействия с внешними системами

#### 3.4.1. Модуль экспорта данных во внешние системы

Модуль экспорта данных во внешние по отношению к ПУМ системы должен осуществлять средствами сетевого доступа:

- Периодический экспорт определённой части данных во BC (периодичность задается Администратором ПУМ) состава оборудования и логических ресурсов, аварийных сообщений, журналов логирования;
- Взаимодействие с BC должно быть обеспечено в необходимом объеме функций и гарантировать отсутствие влияния на безопасность ПУМ.

Экспорт данных аварийных сообщений и журналов логирования выполняется с использованием базы данных подсистемы мониторинга. Экспорт данных состава оборудования и логических ресурсов выполняется из базы данных подсистемы технического учёта. Экспорт данных журналов логирования выполняется из подсистемы контроля доступа.

В качестве языка разработки используется python в среде Linux.

Инв. № подл.	Подп. и дата
Взам. инв. №	Подп. и дата
Инв. № дубл.	Подп. и дата
Подп. и дата	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					12

### 3.4.2. Модуль обработки запросов API в реальном времени

Модуль обработки API в реальном времени должен работать по следующей схеме:

- Прием запроса в формате REST (HTTP GET/POST, JSON) от ВС.
- Обработка запроса с использованием диагностической информации с соответствующих подсистем и компонентов .
- Возврат полученных в ходе обработки данных инициатору запроса.

Модуль выполняет прием запросов от ВС, его обработку с использованием информации подсистем и компонентов ККП, возврат полученных в ходе обработки данных инициатору запроса. При этом взаимодействие осуществляется через очередь сообщений AMQP брокера RabbitMQ. Очередь входящих сообщений прослушивается модулем на предмет поступающих запросов, в теле которого содержится идентификатор ответной очереди, в которую направляется результат обработки.

В качестве языка разработки используется python 3 в среде Linux. Для взаимодействия с брокером RabbitMQ используется библиотека pika для python 3.

### 3.5. Подсистема контроля доступа

Подсистема контроля доступа ПУМ обеспечивает:

- аутентификацию, авторизацию и аудит учетных записей.
- валидацию доверенных устройств – взаимодействие между АРМ и серверами ПУМ, серверами ПУМ и серверами ПДУ, серверами ПДУ и аппаратной частью ПДУ, устройствами для консольного доступа к компонентам в аварийных.
- доступ пользователей в систему через графический интерфейс автоматизированного рабочего места;

Име. № подл.	Подп. и дата	Взам. инв. №	Име. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					13

- регистрацию пользователя (администратора) в системе с использованием подтверждения со стороны администратора и назначения группы, в которую включён пользователь. Данные пользователя автоматически вносятся в OpenLDAP;
- контроль учетных данных пользователей;
- внесение изменений в учётные данные пользователей через графический интерфейс;
- определение принадлежности пользователей к группам;
- запись событий входа и взаимодействия с системой в системные логи;
- блокировка и разблокировка пользователей;
- автоматизация создания учетных данных в системах OpenLDAP и Kerberos для дальнейшего использования устройствами системы;
- интеграция в модули управления и мониторинга квантовой коммуникационной платформы;
- единую точку входа по протоколу CAS 2.0, в которой используются данные из OpenLDAP.

### 3.6. ЛСУ и агенты ЛСУ

Операционная система на базе ОС Linux на ЛСУ решает задачи предоставления среды выполнения для агентов ЛСУ. Агенты ЛСУ решают задачи выполнения локального мониторинга, управления подсистемами ПДУ и КРК, инвенторизации оборудования а так же контроля доступа, и по сути являются компонентами соответствующих подсистем, выполняющимися на ЛСУ:

- Агент модуля управления ПДУ и КРК описан в п. 3.1.1
- Агент модуля мониторинга п 3.2.1 и 3.2.2
- Агент модуля технического учета описан в п 3.3.1 и 3.2.2
- Агент подсистемы контроля доступа описан в п 3.5

Инд. № подл.	Подп. и дата
Взам. инв. №	Инд. № дубл.
Подп. и дата	

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					14

### 3.7. ПО АРМ Администратора

Автоматизированное рабочее место администратора (АРМ администратора) реализует функции взаимодействия с администраторами с помощью графического интерфейса. Графический интерфейс реализуется в браузере администратора и может быть выполнен на рабочем месте – ПК, имеющем доступ к сети управления. Серверная часть ПО АРМ администратора выполняется на ЦУМ и позволяет производить авторизацию пользователей, отображать окна графического интерфейса с учётом доступных ролей в системе за счёт использования подсистемы контроля доступа.

ПО АРМ администратора обеспечивает:

- генерацию окон графического интерфейса
- отображение окон в браузере администратора по запросу
- отправку действий пользователя на исполнение подсистеме управления
- отображение результата выполнения действий подсистемой управления в графическом виде
- интеграцию с подсистемой мониторинга и технического учёта для отображения их данных

Изн. № подл.	Подп. и дата	Взам. инв. №	Изн. № дубл.	Подп. и дата

										Лист
Изм.	Лист	№ докум.	Подп.	Дата						15

## 4. Входные и выходные данные

### 4.1. Подсистема управления

#### 4.1.1. Модуль управления ПДУ и КРК

Входными данными модуля управления ПДУ и КРК являются данные, поступающие по протоколу AMQP со стороны интерфейса оператора (АРМ) а также ответы на запросы полученные от СПДУ

Выходными данными являются ответы на запросы со стороны интерфейса администратора (АРМ), отправляемые по протоколу AMQP, а также запросы к СПДУ.

#### 4.1.2. Модуль управления трафиком, топологией и технологическими параметрами ККП

Входные данные модуля представляют собой набор параметров, определяющих построение коммуникационного канала, передаваемые по протоколу HTTP со стороны интерфейса оператора (АРМ).

Выходными данными модуля являются ответы в контексте полученных запросов, и отображаемых в интерфейсе оператора (АРМ).

### 4.2. Подсистема мониторинга

#### 4.2.1. Модуль мониторинга ПДУ и КРК

Входные данные модуля представляют собой параметры команды на запрос состояния ПДУ и КРК, запросы от агента мониторинга и данные мониторинга ПДУ и КРК. Получив запрос от агента мониторинга, сервер отправляет ответ на этот запрос. Данные мониторинга ПДУ и КРК поступают на Zabbix сервер, запущенный в ВМ на ЦУМ. Zabbix сервер обрабатывает эти данные и заносит их в базу данных PostgreSQL.

Выходные данные модуля представляют собой результаты выполнения команд мониторинга ПДУ и КРК, ответы на запросы агента мониторинга, а

Име. № подл.	Подп. и дата	Взам. инв. №	Име. № дубл.	Подп. и дата
--------------	--------------	--------------	--------------	--------------

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					16

также оповещения о нештатных ситуациях, которые доставляются подписчикам на определенные события в системе. Получив запрос от агента мониторинга, сервер отправляет ответ на этот запрос. Zabbix сервер, запущенный в ВМ на ЦУМ, анализирует полученные данные мониторинга ПДУ и КРК и при отклонении от штатных значений формирует оповещения, отправляемые по электронной почте подписчикам этого события.

Кроме того, выходными данными являются события (событие – это любой выход значений параметров любого компонента за допустимые пределы и /или расхождение в составе оборудования в ПУМ и ВС), посылаемые в ВС через брокер сообщений Kafka (служебный модуль коммутации данных ВС) и Apache ServiceMix.

Входными данными являются данные о результатах обработки событий в привязке к инцидентам и работам для информирования администратора ПУМ, принятые из ВС через брокер сообщений Kafka (служебный модуль коммутации данных ВС) и Apache ServiceMix.

#### 4.2.2. Модуль мониторинга оборудования ККП

Входные данные модуля представляют собой параметры команды на запрос состояния оборудования ККП, запросы от агента мониторинга и данные мониторинга оборудования ККП. Получив запрос от агента мониторинга, сервер отправляет ответ на этот запрос. Данные мониторинга оборудования ККП поступают на Zabbix сервер, запущенный в ВМ на ЦУМ. Zabbix сервер обрабатывает эти данные и заносит их в базу данных PostgreSQL.

Выходные данные модуля представляют собой результаты выполнения команд мониторинга оборудования ККП, ответы на запросы агента мониторинга, а также оповещения о нештатных ситуациях, которые доставляются подписчикам на определенные события в системе. Получив запрос от агента мониторинга, сервер отправляет ответ на этот запрос. Zabbix сервер, запущенный в ВМ на ЦУМ, анализирует полученные данные мониторинга

Инв. № подл.	Подп. и дата				Лист
Инв. № инв.	Инв. № дубл.				17
Взам. инв. №	Подп. и дата				Формат А4
Инв. № подл.	Изм.	Лист	№ докум.	Подп.	Дата

оборудования ККП и при отклонении от штатных значений формирует оповещения, отправляемые по электронной почте подписчикам этого события.

Кроме того, выходными данными являются события (событие – это любой выход значений параметров любого компонента за допустимые пределы и /или расхождение в составе оборудования в ПУМ и ВС), посылаемые в ВС через брокер сообщений Kafka (служебный модуль коммутации данных ВС) и Apache ServiceMix.

Входными данными являются данные о результатах обработки событий в привязке к инцидентам и работам для информирования администратора ПУМ, принятые из ВС через брокер сообщений Kafka (служебный модуль коммутации данных ВС) и Apache ServiceMix.

### 4.3. Подсистема технического учёта

#### 4.3.1. Модуль учета оборудования

Входные данные модуля представляют собой параметры команды на запрос о состоянии и комплектности оборудования, запросы от агента инвентаризации и данные инвентаризации оборудования. Получив запрос от агента инвентаризации, сервер отправляет ответ на этот запрос. Данные учета оборудования поступают в систему GLPI, запущенную в ВМ на ЦУМ. GLPI обрабатывает эти данные и заносит их в базу данных MariaDB.

Выходные данные модуля представляют собой запросы к агенту инвентаризации оборудования, ответы на запросы агента инвентаризации, а также списки состава оборудования. Получив запрос от агента инвентаризации, сервер отправляет ответ на этот запрос. Для формирования списка состава оборудования соответствующие данные берутся из базы данных.

Кроме того, входными данными модуля являются описания оборудования, полученные от ВС через брокер сообщений Kafka (служебный модуль коммутации данных ВС) и Apache ServiceMix.

Все описания из ВС реплицируются в модуле.

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл.	

									Лист
Изм.	Лист	№ докум.	Подп.	Дата					18

Выходными данными модуля являются инвентаризационные данные, полученные с оборудования, регулярно передаваемые во ВС через брокер сообщений Kafka (служебный модуль коммутации данных ВС) и Apache ServiceMix.

### 4.3.2. Модуль учета логических ресурсов

Входные данные модуля представляют собой параметры команды на запрос о состоянии и комплектности логических ресурсов, запросы от агента инвентаризации и данные инвентаризации логических ресурсов. Получив запрос от агента инвентаризации, сервер отправляет ответ на этот запрос. Данные учета логических ресурсов поступают в систему GLPI, запущенную в ВМ на ЦУМ. GLPI обрабатывает эти данные и заносит их в базу данных MariaDB.

Выходные данные модуля представляют собой запросы к агенту инвентаризации логических ресурсов, ответы на запросы агента инвентаризации, а также списки состава логических ресурсов. Получив запрос от агента инвентаризации, сервер отправляет ответ на этот запрос. Для формирования списка состава логических ресурсов соответствующие данные берутся из базы данных.

Выходными данными модуля являются данные логических ресурсов, полученные с оборудования, регулярно передаваемые в ВС через брокер сообщений Kafka (служебный модуль коммутации данных ВС) и Apache ServiceMix.

## 4.4. Подсистема взаимодействия с внешними системами

### 4.4.1. Модуль экспорта данных во внешние системы

Входными данными модуля экспорта данных во ВС является конфигурация адресата отправки данных (адрес, порт), периодичность отправки данных, метрики для отправки. По результату обработки конфигурации производится настройка планировщика модуля.

Инв. № подл.	Подп. и дата
Взам. инв. №	Подп. и дата
Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					19



Выходными данными модуля экспорта данных во ВС являются:

- Файл конфигурации в текстовом формате, кодировка Unicode.
- Данные из Баз данных подсистемы мониторинга и подсистемы технического учета, передаваемые во ВС через брокер сообщений Kafka (служебный модуль коммутации данных ВС) и Apache ServiceMix, кодировка Unicode.
- Журнал логирования в текстовом формате, кодировка Unicode.

#### 4.4.2. Модуль обработки запросов API в реальном времени

Входными данными модуля обработки запросов API в реальном времени являются структуры формата JSON, описывающие тело запроса и идентификатор очереди, в которую необходимо отправить ответ.

Выходными данными модуля обработки запросов API в реальном времени являются сообщения формата JSON, поступающие через очередь сообщений Rabbitmq. Формат выходных данных модуля обработки запросов API в реальном времени – JSON, кодировка текста Unicode.

#### 4.5. Подсистема контроля доступа

Входные данные подсистемы контроля доступа содержат в себе идентификатор запроса для проверки сессии пользователя, либо логин и пароль для создания сессии. Формат входных данных подсистемы контроля доступа – запрос HTTPs, кодировка текста Unicode.

Выходными данными подсистемы контроля доступа выступают статус сессии пользователя и данные о группах пользователя в формате HTML, кодировка текста Unicode.

Подп. и дата
Инв. № дубл.
Взам. инв. №
Подп. и дата
Инв. № подл.

										Лист
										20
Изм.	Лист	№ докум.	Подп.	Дата						

#### 4.6. ЛСУ и агенты ЛСУ

Входными и выходными данными являются данные в форматах соответствующих подсистем, протоколы передачи данных так же описана в соответствующих подсистемах

- Агент модуля управления ПДУ и КРК описан в п. 4.1.1
- Агент модуля мониторинга п 4.2.1 и 4.2.2
- Агент модуля технического учета описан в п 4.3.1 и 4.3.2
- Агент подсистемы контроля доступа описан в п 4.5

#### 4.7. ПО АРМ Администратора

Входными данными ПО АРМ Администратора является тестовая строка запроса и поле данных запроса. По результатам обработки строки производится выбор страницы для генерации и отображения. Формат входных данных ПО АРМ Администратора – запрос HTTPs, кодировка текста Unicode.

Выходными данными ПО АРМ Администратора являются страницы формата HTML для отображения в браузере, кодировка текста Unicode.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Лист
Изм.	Лист	№ докум.	Подп.	Дата	

## Обозначения и сокращения

AMQP	Advanced Message Queuing Protocol
API	Application programming interface
CWDM	Coarse Wavelength Division Multiplexing
EVM	Extended Verification Module
IMA	Integrity Measurement Architecture
IMA/EVM	Integrity Measurement Architecture and Extended Verification Module
QBER	Quantum Bit Error Rate
QBER	Quantum Bit Error Rate
QKD	Quantum key distribution
SLA	Service Level Agreement
SNMP	Simple Network Management Protocol
TCP/IP	Transmission Control Protocol / Internet Protocol
YAML	Yet Another Markup Language
АРМ	Автоматизированное рабочее место
БД	База данных
ВМ	Виртуальная машина
КЗК	Квантово-защищенные ключи
ККП	Квантовая коммуникационная платформа цифровой экономики
КРК	Квантовое распределение ключей
ЛСУ	Локальный сервер управления
ПДУ	Подсистема организации доверенных опорных узлов квантовой связи
ПКРК	Подсистема квантового распределения ключей
ПО	Программное обеспечение

Подл. и дата

Инв. № дубл.

Взам. инв. №

Подл. и дата

Инв. № подл.

Изм.	Лист	№ докум.	Подп.	Дата

Лист

22

ПУМ	Подсистема управления и мониторинга квантовой сетью
СКЗИ	Система защиты информации
ЦУМ	Центр Управления и Мониторинга

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Изм.	Изм.	Изм.	Лист	Лист	Лист			
											Лист	Лист	Лист

