

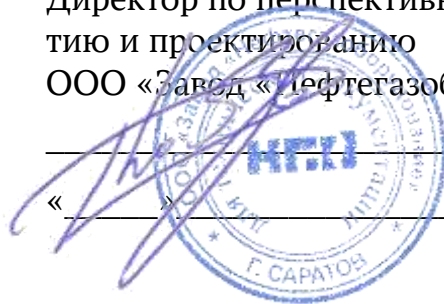
УТВЕРЖДАЮ

Директор по перспективному развитию и проектированию

ООО «Завод «Нефтегазоборудование»

Д.В. Кобзов

«\_\_\_\_\_» \_\_\_\_\_ 2023г.



**СИСТЕМА ОПЕРАТИВНО-ДИСПЕТЧЕРСКОГО  
УПРАВЛЕНИЯ  
НА БАЗЕ ПРОГРАММНОГО КОМПЛЕКСА СОДУ «ПОТОК»**

**Руководство системного программиста**

**Лист утверждения**

**НГО-СОДУ.28.99.39.190 РСП-ЛУ**

Инва. № подл.	Подпись и дата	Взам. инв. №	Инва. № дубл.	Подпись и дата

Начальник отдела АСУ ТП

И.П. Лазаренко И.П. Лазаренко

«\_\_\_\_\_» \_\_\_\_\_ 2023г.

Руководитель группы программных разработок отдела АСУ ТП

А.М. Полещук А.М. Полещук

«\_\_\_\_\_» \_\_\_\_\_ 2023г.

**УТВЕРЖДЕНО**

**НГО-СОДУ.28.99.39.190 РСП-ЛУ**

**СИСТЕМА ОПЕРАТИВНО-ДИСПЕТЧЕРСКОГО  
УПРАВЛЕНИЯ  
НА БАЗЕ ПРОГРАММНОГО КОМПЛЕКСА СОДУ «ПОТОК»**

**Руководство системного программиста**

**НГО-СОДУ.28.99.39.190 РСП-ЛУ**

**Листов 24**

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

## АННОТАЦИЯ

В данном документе описана установка и настройка программного комплекса (далее - ПК) системы оперативно-диспетчерского управления (далее - СОДУ).

В разделе «Общие сведения о программном комплексе» указаны назначение и функции программного комплекса и сведения о технических и программных средствах, обеспечивающих выполнение данной программы, а также требования к персоналу.

В разделе «Структура программного комплекса» приведены сведения о структуре программного комплекса, его составных частях, о связях между составными частями и о связях с другими программами.

В данном программном документе, в разделе «Настройка программного комплекса» приведено описание действий по настройке программ на условия конкретного применения (настройка на состав технических и программных средств, выбор функций и др.).

В разделе «Проверка программного комплекса» приведено описание способов проверки, позволяющих дать общее заключение о работоспособности программы (контрольные примеры, методы прогона, результаты).

В разделе «Сообщения системному программисту» указаны тексты сообщений, выдаваемых в ходе выполнения настройки, проверки программы, а также в ходе выполнения программы, описание их содержания и действий, которые необходимо предпринять по этим сообщениям.

Оформление программного документа «Руководство системного программиста» произведено по требованиям ЕСПД (ГОСТ 19.101-77 <sup>1)</sup>, ГОСТ 19.103-77 <sup>2)</sup>, ГОСТ 19.104-78\* <sup>3)</sup>, ГОСТ 19.105-78\* <sup>4)</sup>, ГОСТ 19.106-78\* <sup>5)</sup>, ГОСТ 19.503-79\* <sup>6)</sup>, ГОСТ 19.604-78\* <sup>7)</sup>).

---

<sup>1)</sup> ГОСТ 19.101-77 ЕСПД. Виды программ и программных документов

<sup>2)</sup> ГОСТ 19.103-77 ЕСПД. Обозначение программ и программных документов

<sup>3)</sup> ГОСТ 19.104-78\* ЕСПД. Основные надписи

<sup>4)</sup> ГОСТ 19.105-78\* ЕСПД. Общие требования к программным документам

<sup>5)</sup> ГОСТ 19.106-78\* ЕСПД. Общие требования к программным документам, выполненным печатным способом

<sup>6)</sup> ГОСТ 19.503-79\* ЕСПД. Руководство системного программиста. Требования к содержанию и оформлению

<sup>7)</sup> ГОСТ 19.604-78\* ЕСПД. Правила внесения изменений в программные документы, выполненные печатным способом

1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММНОМ КОМПЛЕКСЕ.....	5
1.1. Назначение программного комплекса .....	5
1.2. Функции программного комплекса.....	5
1.3. Минимальный состав технических средств.....	6
1.4. Минимальный состав программных средств .....	7
1.5. Требования к персоналу (системному программисту).....	8
2. СТРУКТУРА ПРОГРАММНОГО ПАКЕТА .....	8
2.1. Сведения о структуре программного пакета .....	8
2.2. Сведения о составных частях программы.....	9
2.3. Сведения о связях между составными частями ПК .....	11
2.4. Сведения о связях с другими программами.....	12
3. НАСТРОЙКА ПРОГРАММЫ .....	12
3.1. Настройка на состав технических средств .....	12
4. УСТАНОВКА ПРОГРАММНОГО КОМПЛЕКСА .....	13
4.1. Программы для серверных станций.....	13
4.2. Программы для АРМ.....	16
4.3. Программы для исторического сервера.....	17
4.4. Программы для станции развертывания .....	18
5. АДМИНИСТРИРОВАНИЕ ПРОГРАММНОГО КОМПЛЕКСА .....	22
5.1 Резервное копирование узлов СОДУ.....	23
5.1.1 Исторический сервер.....	23
5.1.2 Серверная станция.....	25
5.1.3 Станция АРМ.....	28
5.2 Резервное восстановление узлов СОДУ .....	29
5.2.1 Исторический сервер.....	29
5.2.2 Серверная станция.....	30
5.2.3 Станция АРМ.....	31
5.3 Синхронизация времени.....	31
6. ПРОВЕРКА ПРОГРАММНОГО КОМПЛЕКСА .....	32
6.1. Описание способов проверки .....	32
7. СООБЩЕНИЯ СИСТЕМНОМУ ПРОГРАММИСТУ.....	33
ПЕРЕЧЕНЬ ТЕРМИНОВ И СОКРАЩЕНИЙ.....	34
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ .....	35

## 1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММНОМ КОМПЛЕКСЕ

### 1.1. Назначение программного комплекса

1.1 Программный комплекс входит в состав СОДУ и является программным средством для сбора, обработки, представления и архивирования данных, поступающих с различных локальных автоматизированных систем управления технологическим процессом (далее - АСУ ТП).

СОДУ предназначена для оперативного, круглосуточного контроля и управления подконтрольными объектами. Обобщенная структура СОДУ приведена на рисунке 1.

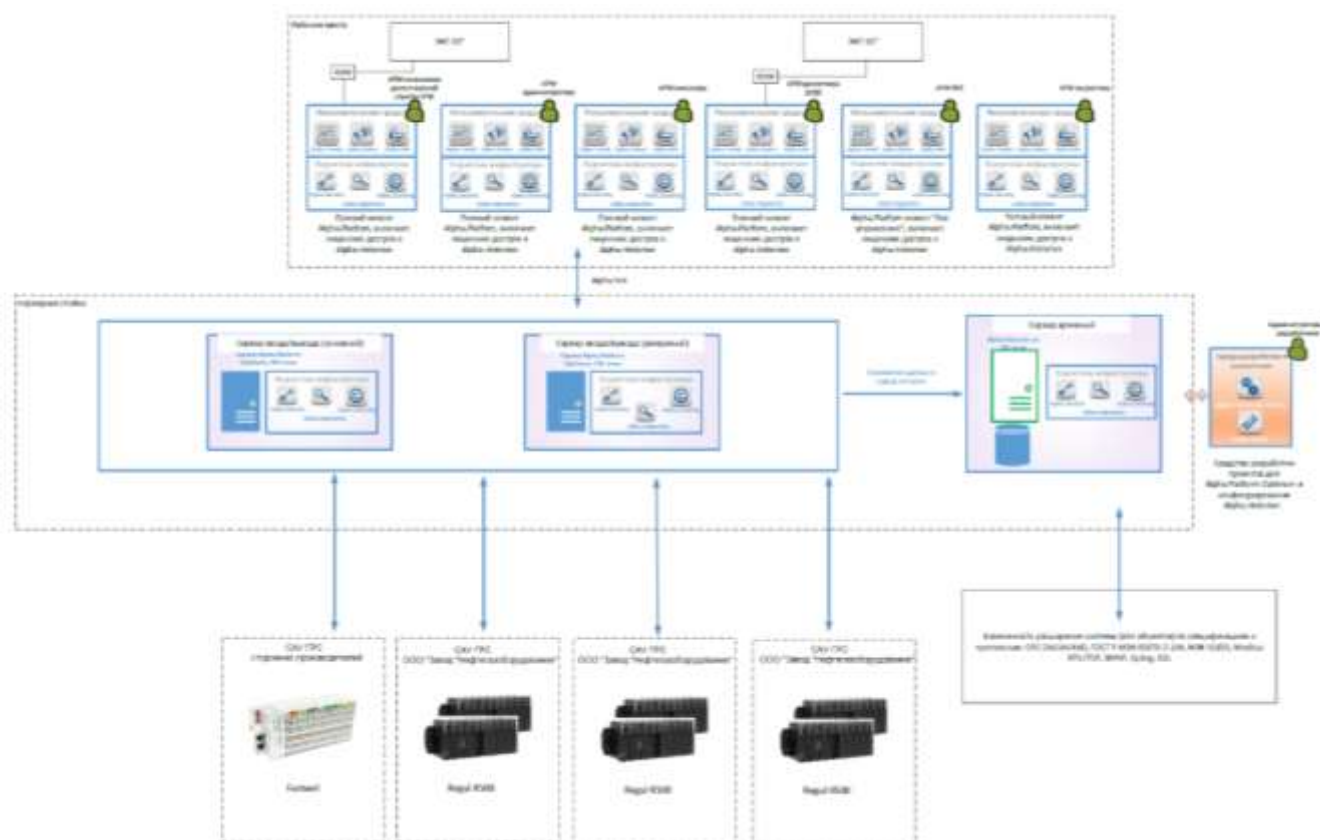


Рис. 1 Структура СОДУ

### 1.2. Функции программного комплекса

СОДУ осуществляет:

- 1) Ведение журналов событий;
- 2) Построение графиков технологических процессов;
- 3) Хранение и создание отчетной документации;
- 4) Хранение учетных документов (коммерческий и технологический учет газа);
- 5) Ведение баз данных (в т.ч. архивирование и пр.);
- 6) Интеграцию в системы управления предприятием;
- 7) Отображение текущей информации о состоянии контролируемых объектов (измерение, сигнализация);
- 8) Выдачу команд телеуправления (далее - ТУ) и регулирования посредством программного обеспечения (далее - ПО) локальных контроллеров подконтрольных объектов без участия ПО верхнего уровня (СОДУ);

**НГО-СОДУ.28.99.39.190 РСР**

- 9) Оповещения о предаварийных и аварийных ситуациях на объектах;
- 10) Доступ к СОДУ осуществляется согласно предустановленной системе доступов и паролей;
- 11) Резервирование на уровне серверов и баз данных.

**1.3. Минимальный состав технических средств**

## 1) Основной сервер.

Основной сервер является главным элементом системы и выполняет функции обработки и обмена информацией. Сервер представляет собой компьютер с установленным программным обеспечением Alpha Platform, который предназначен для обработки и хранения данных, а также для предоставления доступа к этим данным через сеть. Для полноценной работы системы резервирования требуется два таких сервера: основной и резервный.

Сервер выполнен по модульному принципу и содержит следующие блоки:

- две сетевые карты Ethernet или более;
- блок питания;
- процессор с тактовой частотой 2.1 ГГц или выше;
- оперативную память объемом 8 Гб или выше;
- жесткий диск объемом 32 Гб и выше;

## 2) Исторический сервер.

Исторический сервер (архивный) по аппаратной части аналогичен основному серверу, но с другим набором программ из пакета Alpha Platform. Основные функции – сбор и сохранение информации о технологическом процессе: ведение баз исторических данных, архива событий, сообщений. Для полноценной работы системы резервирования исторических данных требуется два таких сервера: основной и резервный исторические серверы.

## 3) Автоматизированное рабочее место оператора (далее - АРМ).

АРМ оператора – компьютер с монитором на базе операционной системы Astra Linux (Орел) и определенным набором программ из пакета Alpha Platform. Максимальное количество АРМ в системе составляет 100 шт.

4) Лицензионные ключи для программного обеспечения Alpha Platform для каждого сервера и АРМ. Могут быть программными или аппаратными.

## 5) Станция развертывания.

Станция развертывания (инженерная станция) предназначена для загрузки проекта СОДУ по всем узлам системы, а также внесения изменений в проект и настройки системы контроля доступа на АРМ.

## 6) Комплект сетевого оборудования

В комплект сетевого оборудования входит сетевой многопортовый управляемый маршрутизатор, канал(-ы) связи с САУ ГРС. Комплект обеспечивает физический обмен данными со всеми вышеописанными узлами сети и средним уровнем автоматизации – САУ ГРС.

#### 1.4. Минимальный состав программных средств

На все узлы СОДУ, кроме станции развертывания, необходима установка операционной системы Astra Linux (Орел) версии не ниже 2.12.44. Помимо операционной системы на каждый узел необходимо установить различный набор программ Alpha Platform в соответствии с его функционалом. Версии компонентов на различных узлах должны быть одинаковыми.

##### 1) Сервер:

- alpha.server (версия не ниже 5.11.0);
- alpha.tools (версия не ниже 1.6.3);
- alpha.domain (версия не ниже 1.2.3);
- openldap (версия не ниже 2.5.4).

##### 2) Исторический сервер:

- alpha.server (версия не ниже 5.11.0);
- alpha.domain (версия не ниже 1.2.3);
- alpha.historian (версия не ниже 3.6.9).

##### 3) АРМ оператора:

- alpha.domain (версия не ниже 1.2.3);
- alpha.accesspoint (версия не ниже 5.11.0);
- alpha.hmi (версия не ниже 1.10.4);
- alpha.hmi.security версия не ниже 1.1.2);

##### 4) Станция развертывания:

- alpha.devstudio (версия не ниже 3.20.0);
- alpha.server (версия не ниже 5.11.0);
- alpha.domain (версия не ниже 1.2.3);
- alpha.tools (версия не ниже 1.6.3);
- alpha.hmi.desktop (версия не ниже 1.9.2);
- alpha.hmi.charts (версия не ниже 1.1.2);
- alpha.hmi.tables (версия не ниже 1.2.3);
- alpha.hmi.alarms (версия не ниже 1.2.2);
- alpha.hmi.commonlib (версия не ниже 1.2.0);
- alpha.hmi.trends (версия не ниже 1.2.2);
- alpha.hmi.security версия не ниже 1.1.2);
- alpha.security (версия не ниже 1.4.1);
- openldap (версия не ниже 2.5.4);
- alpha.documentation (версия не ниже 22.6.4).

## НГО-СОДУ.28.99.39.190 РСП

### 1.5. Требования к персоналу (системному программисту)

#### 1.5.1 Системный программист должен:

- иметь опыт администрирования операционной системы (далее - ОС) семейства Linux;
- иметь опыт администрирования системы управления базами данных (далее - СУБД), PostgreSQL;
- иметь опыт работы с системами резервного копирования данных.

#### 1.5.2 В перечень задач, выполняемых системным программистом, входят:

- установка, настройка и поддержание работоспособности ОС и СУБД;
- установка, настройка и поддержание работоспособности П;
- создание и восстановление резервных копий узлов сети СОДУ.

## 2. СТРУКТУРА ПРОГРАММНОГО ПАКЕТА

### 2.1. Сведения о структуре программного пакета

Ключевой единицей инфраструктуры Alpha Platform является домен – совокупность вычислительных средств для исполнения проекта автоматизации. Серверные компоненты домена выполняют целевые функции проекта: сбор данных от нижестоящих систем, логическую обработку данных, предоставление данных вышестоящим системам, хранение исторической информации и прочее. К подсистеме исполнения относится Alpha.Server и Alpha.Historian.

Alpha.AccessPoint – единая точка доступа, объединяющая серверные компоненты и удалённые домены Alpha Platform (для построения распределённых систем), а также взаимодействие со сторонними приложениями и системами по стандартным протоколам и спецификациям.

Пользовательская среда обеспечивает работу с визуальной частью проекта автоматизации. К компонентам пользовательской среды относятся Alpha.HMI, Alpha.Alarms и Alpha.Trends.

Среда разработки и управления служит для создания, тестирования и отладки приложений и состоит из продуктов DevStudio и Alpha.Tools. Среда используется также для выполнения централизованных задач по настройке и обслуживанию домена Alpha Platform.

К подсистеме инфраструктуры относятся продукты Alpha.Security (регламентирует безопасность и разграничение прав внутри домена), Alpha.Diagnostics (выполняет функции комплексной диагностики, мониторинга и аудита), Alpha.Licensing (обеспечивает лицензирование продуктов внутри домена).



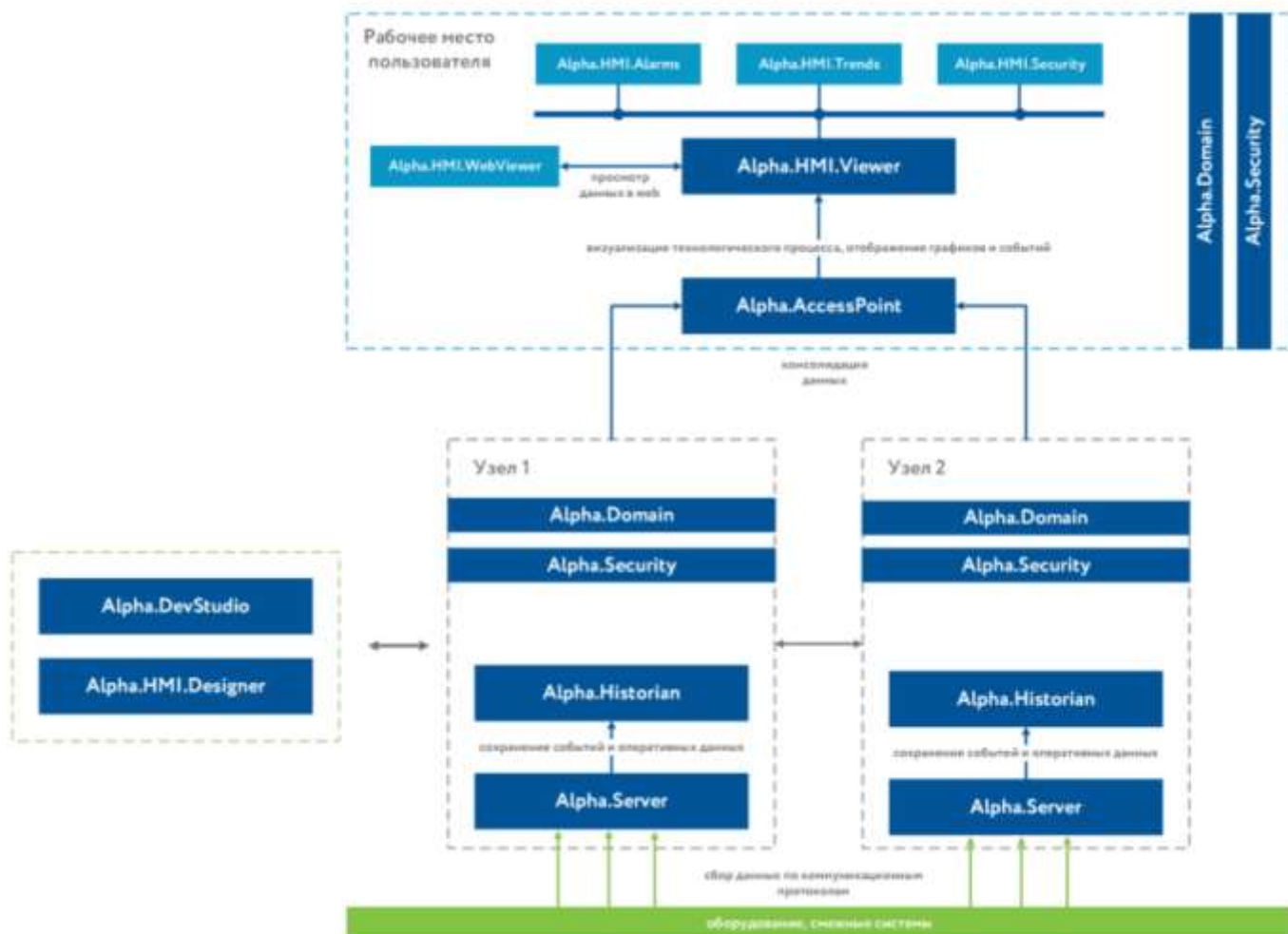


Рис. 2 Структура программного пакета.

## 2.2. Сведения о составных частях программы

Alpha.Server – компонент Alpha Platform, выполняющий следующие задачи:

- Сбор данных с устройств в ходе мониторинга контролируемых объектов.
- Предоставление данных клиентам по различным протоколам и спецификациям.
- Повышение надежности проекта за счёт резервирования.
- Логическая обработка данных в режиме реального времени.
- Генерация событий и тревог на основе полученных данных.

Alpha.Historian – компонент Alpha Platform для сбора и сохранения информации о технологическом процессе.

Возможности Alpha.Historian:

- Сбор и хранение оперативных значений параметров технологического процесса.
- Сбор и хранение истории событий и тревог технологического процесса.
- Предоставление исторических данных клиентам.

Модуль истории в составе Alpha.Server выполняет временное хранение данных на стороне сервера и передачу информации в хранилище Alpha.Historian.

## НГО-СОДУ.28.99.39.190 РСР

Alpha.AccessPoint – компонент реализует функции сервера приложений и межуровневого транспорта. Предназначен для снижения нагрузки на технологические серверы и для транзитной передачи данных между доменами Альфа платформы.

Возможности Alpha.AccessPoint:

- Объединение сигналов различных OPC DA-источников в единое дерево сигналов;
- Объединение событий и тревог от различных OPC AE-источников;
- Поддержание связи с источниками данных при разрыве соединения;
- Передача OPC DA и OPC AE данных в виде ТСР/ІР-трафика в условиях различных сетевых топологий и работа в режиме каскадирования;
- Поддержка сбора данных по файловому интерфейсу;
- Доступ к данным Alpha.AccessPoint осуществляется по спецификациям OPC DA, OPC AE, OPC UA.

Alpha.DevStudio – среда разработки и администрирования проектов автоматизации.

Возможности DevStudio:

- Сквозное описание физической структуры проекта автоматизации от уровня ПЛК до верхнего уровня;
- Сквозное описание логической структуры проекта автоматизации, а именно функций и данных объектов автоматизации применительно к средствам автоматизации различной функциональной направленности (сервера сбора данных, сервера истории, сервера межуровневого транспорта);
- Представление схемы развертывания проекта автоматизации на исполняющих компонентах;
- Компиляция и сборка конфигураций исполняющих компонентов Альфа платформы;
- Управление развёртыванием конфигураций проекта в среде исполнения Альфа платформы.

Alpha.HMI.Designer – среда разработки и исполнения визуальной части проектов автоматизации. Alpha.HMI позволяет представлять объекты технологического процесса в виде статических и анимированных объектов мнемосхемы для мониторинга за ходом процесса и управления им.

Возможности Alpha.HMI:

- Визуальный редактор со стандартной библиотекой компонентов для построения мнемосхем;
- Взаимодействие с OPC DA источниками данных;
- Объектно-ориентированный подход при разработке проектов и возможность создания собственных типов графических объектов;
- Поддержка скриптовых языков Alpha.Om и JavaScript;
- Взаимодействие с подсистемой безопасности Alpha.Security;
- Взаимодействие с файловой системой, сетевым окружением и оборудованием компьютера;
- Компоненты автоматической разметки элементов мнемосхемы;
- Компоненты организации динамики на мнемосхеме;
- Встраиваемые компоненты Alpha.Trends и Alpha.Alarms.

## НГО-СОДУ.28.99.39.190 РСП

Alpha.Security – часть инфраструктурной подсистемы Alpha.Domain, которая обеспечивает окружение для работы других подсистем Альфа платформы.

Alpha.Security – система, обеспечивающая разграничение доступа пользователей проекта автоматизации с помощью их должностных инструкций. Система построена на протоколе доступа к каталогам – LDAP-сервер.

Alpha.Security состоит из следующих компонентов:

- Программа Alpha.Security.Configurator;
- Служба Alpha.Security.Agent.

Alpha.Tools – набор инструментов, предназначенных для сервисных и диагностических целей при работе с проектами автоматизации. В состав дистрибутива Alpha.Tools входят приложения:

- Приложение EventLogViewer предназначено для удобного просмотра журналов событий с целью диагностики работоспособности системы;
- Приложение OpsExplorer предназначено для использования в пунктах автоматизации технологических процессов. Применяется для просмотра и изменения значений сигналов, мониторинга событий, возникающих при изменении состояний технологических объектов и для графического отображения изменения значений сигналов.

### 2.3. Сведения о связях между составными частями ПК

На рисунках 1 и 2 изображен информационный обмен между составными частями программного комплекса СОДУ. Серверные узлы (ввода-вывода) обмениваются со средним уровнем автоматизации (САУ ГРС) по различным протоколам обмена, в частности с САУ ГРС производства ООО «Завод «Нефтегазоборудование», по спецификации OPC UA. Межузловой обмен осуществляется по протоколу TCP.

**2.4. Сведения о связях с другими программами**

Связь программного комплекса СОДУ с другим программным обеспечением отсутствует.

**3. НАСТРОЙКА ПРОГРАММЫ****3.1. Настройка на состав технических средств**

Необходимо убедиться в наличии аппаратных USB ключей (если приобретена аппаратная лицензия программного комплекса) на всех узлах сети СОДУ (серверы и АРМы), кроме станции развертывания. Для всех узлов сети нужно убедиться в правильности задания IP адресов и при необходимости их скорректировать:

- Основная серверная станция -10.30.200.11;
- Резервная серверная станция - 10.30.200.16;
- Станция развертывания - 10.30.200.50;
- Исторический сервер – 10.30.200.21.

IP адреса АРМов могут быть произвольными в рамках той же подсети, что и серверные станции.

В остальном программный комплекс СОДУ не требует каких-либо настроек на состав технических средств.

#### 4. УСТАНОВКА ПРОГРАММНОГО КОМПЛЕКСА

Весь пакет программного комплекса СОДУ и документация хранится на станции развертывания, которые необходимо выгрузить на любой flash носитель. В зависимости от функционала узла сети СОДУ требуется установить соответствующий набор программ. Для каждого узла необходимо соблюдать описанную ниже последовательность установки программ. Указанные файлы расположены в одноименных каталогах на flash носителе. В каждом каталоге находятся программы для разных операционных систем.

##### 4.1. Программы для серверных станций

Набор программ для обеих серверных станций (основная и резервная) одинаков. Отличия в настроечных файлах \*.xml, описанных ниже.

Для серверной станции из дистрибутива программного комплекса СОДУ необходимо выбирать установочные пакеты для операционной системы Astra Linux (расширение файла \*.deb). Для установки программного обеспечения нужно обладать правами Суперпользователя.

Устанавливаем пакеты командой «sudo dpkg -I alpha.server» в терминале «fly» в следующей последовательности:

1. alpha.server;
2. alpha.domain;
3. alpha.security;
4. alpha.hmi.security;
5. alpha.licensing.

После установки требуется настроить службы Alpha.Domain, Alpha.Net и Alpha.Security для серверной станции. Для этого в каталоге «/opt/Automiq/Alpha.Domain» вносим изменения в файл «alpha.domain.agent.xml» следующим образом:

```
<?xml version="1.0" encoding="utf-8" ?>
<Alpha.Domain.Agent Name="SDKU ADA">
  <EntryPointNetAgent Name="Node_SERVER1" Port="1010"/>
  <InstalledComponents>
    <Alpha.Server Name="local" ServiceName="alpha.server.service" DefaultActivation="1" />
  </InstalledComponents>
  <Server>
    <Components StoragePath="/usr/local/DomainStorage/cache/server">
      <Component InstalledName="local" Name="AlphaServer" StorageLimitSize="0" StorageLimitNum="0"/>
    </Components>
  </Server>
  <Options LoggerLevel="2"/>
</Alpha.Domain.Agent>
```

Рис. 3 Содержимое файла «alpha.domain.agent.xml» для серверной станции

## НГО-СОДУ.28.99.39.190 РСП

Для резервного сервера в файле «alpha.domain.agent.xml» вместо «Node\_SERVER1» указываем «Node\_SERVER2».

Далее вносим изменения в файл «alpha.net.agent.xml» следующим образом:

```
<Alpha.Net.Agent Name="Node_SERVER1" ParentAgentPort="1111" NetEnterPort="1010">
  <Options LoggerLevel="2"/>
</Alpha.Net.Agent>
```

Рис. 4 Содержимое файла «alpha.net.agent.xml» для основной серверной станции

```
<Alpha.Net.Agent Name="Node_SERVER2" NetEnterPort="1010" ParentAgentPort="1112">
  <Options LoggerLevel="2"/>
</Alpha.Net.Agent>
```

Рис. 5 Содержимое файла «alpha.net.agent.xml» для резервной серверной станции

Далее вносим изменения в файл «alpha.security.agent.xml» следующим образом:

```
<Alpha.Security.Agent>
  <SecurityAgentPort Address="127.0.0.1" Port="1010"/>
  <LDAPHosts>
    <LDAPHost1 hostname="20.30.100.11" port="389"/>
    <LDAPHost2 hostname="20.30.200.18" port="389"/>
  </LDAPHosts>
  <LDAPUser value="cn=admin,dc=domcom,dc=com"/>
  <LDAPPassword value="B0tS2kewZf4snt49C5+/P97ap529y9L+1+62jUC9644M6012B4L1312B2u382z8/P8gtrV9j1q2v8+8K028g4L0LV7M60T81T1c442a29y9x982kaf8eagMTw88V3z0qTK0C29y3k4tN8A824P1340mC2B1+82310W3y"/>
  <LDAPSecure value="False"/>
  <Security value="cn=AlphaSecurity,dc=domcom,dc=com"/>
  <Certificate value="admin"/>
  <CertificatePassword value="PUN2+54W070g5C1E+034+82Mab4.7TMeq37W3yPqtYx6.7B3C+13h044675M42+08e+Q2B+881TM+2y2UC0W442B4n2E99013PylawL8y9v7P70Bp8M54e+e2V7J0x91370779yM0CE4-qw03M47w0Mv13aw41sQ0129e9Qm"/>
  <CertificateDisplayname value="quack"/>
  <Certificate value=""/>
  <Options LoggerLevel="2" LogLevel="1" EnableLogging="0" EnableDebug="0"/>
</Alpha.Security.Agent>
```

Рис. 6 Содержимое файла «alpha.security.agent.xml» для серверной станции

Содержимое файла «alpha.security.agent.xml» одинаково для обоих серверов.

После внесения изменений необходимо перезапустить соответствующие три службы в Astra Linux командами в терминале «fly»:

«sudo systemctl restart Alpha.Net.service»

«sudo systemctl restart Alpha.Domain.service»

«sudo systemctl restart Alpha.Security.service»

Затем при наличии интернета на станции устанавливаем службу Ldap командой «sudo apt install slapd ldap-utils».

Создаем файл в корневом каталоге командой «sudo nano ldaprootpasswd.ldif»

Добавляем в файл текст:

dn: olcDatabase={0}config,cn=config

changetype: modify

add: olcRootPW

olcRootPW: {SSHA}строка\_символов

Пишем команду для настройки БД: sudo dpkg-reconfigure slapd

В открывшемся окне жмем «нет», затем пишем текст «maxsrc.com», потом указываем название организации, например, «Automiq». Вводим пароль «secret» два раза. Тип базы –«MDB». Выбираем «удалять старую» и «переместить».

Перезапускаем службу командой «sudo systemctl restart slapd».

Структурируем дерево каталогов: переходим в каталог

«/opt/Automiq/Alpha.Security». Пишем команду «sudo sh ./alpha.security.schema.export.sh»

## НГО-СОДУ.28.99.39.190 РСП

Создаем файл `sudo nano access.ldif` и вставляем в него текст:

```
dn: olcDatabase={1}mdb,cn=config
changetype: modify
replace: olcAccess
olcAccess: {0}to * by users write by * read
```

Сохраняем и закрываем файл.

Пишем команду «`sudo ldapadd -Y EXTERNAL -H ldapi:/// -f access.ldif`».

На этом настройка Ldap сервера на серверной станции закончена.

В компоненте Alpha.server поддерживается режим горячего резервирования. При горячем резервировании основной и резервный сервер образуют резервную пару. В резервной паре основной сервер работает в полнофункциональном режиме, а резервный компонент находится в режиме ожидания. При прекращении работы основного компонента автоматически выполняется резервный переход, резервный компонент переключается в полнофункциональный режим и будет выполнять уже все функции основного компонента. Клиенты в процессе работы могут подключаться к тому компоненту резервной пары, который является основным. Основным преимуществом горячего резервирования является то, что резервный компонент начинает работать автоматически при потере связи с основным и горячее резервирование происходит практически мгновенно.

Настройка резервирования серверов осуществляется в инструментальной среде «Alpha.DevStudio» и заключается в указании второму серверу функции «Redundancy»<sup>8)</sup>. Эта операция производится на этапе разработки проекта развертывания и имеет следующие параметры для редактирования:

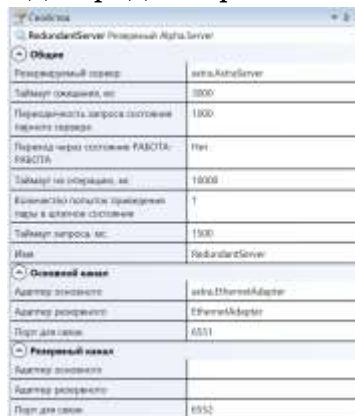


Рис. 7 Окно настроек резервирования в среде Alpha.DevStudio

В окне свойств резервирования отображены настроечные параметры:

- «Таймаут ожидания, мс» - таймаут ожидания установления связи с парным сервером;
- «Таймаут на операцию, мс» - если парная реплика не смогла достичь требуемого состояния за указанное время, считается ее состояние аварийным;
- «Таймаут запроса, мс» - время ожидания ответа на команду от парного сервера, после истечения которого связь по каналу считается потерянной.

Названия остальных параметров однозначно определяют их функционал.

<sup>8)</sup> <https://automiq.ru/ru/docs/Index.htm#Subsystems/DevStudio/Develop/Redundancy.htm>

## 4.2. Программы для АРМ

Набор программ для всех АРМ одинаков.

Для АРМ из дистрибутива программного комплекса СОДУ необходимо выбрать установочные пакеты для операционной системы Astra Linux (расширение файла \*.deb). Для установки программного обеспечения нужно обладать правами Суперпользователя.

Устанавливаем пакеты командой «sudo dpkg -I alpha.server» в терминале «fly» в следующей последовательности:

1. alpha.domain;
2. alpha.security;
3. alpha.hmi.security;
4. alpha.hmi.webviewer;
5. alpha.accesspoint;
6. alpha.licensing.

После установки требуется настроить службы Alpha.Domain, Alpha.Net и Alpha.Security для серверной станции. Для этого в каталоге «/opt/Automiq/Alpha.Domain» вносим изменения в файл «alpha.domain.agent.xml» следующим образом:

```
<?xml version="1.0" encoding="utf-8"?>
<Alpha.Domain.Agent Name="NDA">
  <EntryPointNetAgent Name="local" Address="127.0.0.1" Port="1010"/>
  <InstalledComponents>
    <Alpha.Server Name="Server_1" ServiceName="Alpha.Server" />
  </InstalledComponents>
  <Server>
    <Components StoragePath="/usr/local/DomainStorage/cache/server">
      <Component InstalledName="Server_1" Name="Server" />
    </Components>
  </Server>
  <Options LoggerLevel="2"/>
</Alpha.Domain.Agent>
```

Рис. 8 Содержимое файла «alpha.net.agent.xml» для АРМ

Далее вносим изменения в файл «alpha.net.agent.xml» следующим образом:

```
<Alpha.Net.Agent Name="local" NetEnterPort="1010" ParentAgentPort="1020">
  <Options LoggerLevel="2"/>
</Alpha.Net.Agent>
```

Рис. 9 Содержимое файла «alpha.net.agent.xml» для АРМ

После внесения изменений необходимо перезапустить соответствующие три службы в Astra Linux командами в терминале «fly»:

«sudo systemctl restart Alpha.Net.service»

«sudo systemctl restart Alpha.Domain.service».

Запустить проект визуализации «integration.hmi», расположенный на flash носителе, с помощью пакета «alpha.hmi.viewer».



### 4.3. Программы для исторического сервера

Для исторического сервера<sup>9)</sup> из дистрибутива программного комплекса СОДУ необходимо выбирать установочные пакеты для операционной системы Astra Linux (расширение файла \*.deb). Для установки программного обеспечения нужно обладать правами Суперпользователя.

Устанавливаем пакеты командой «sudo dpkg -I alpha.domain» в терминале «fly» в следующей последовательности:

1. alpha.domain;
2. alpha.historian;
3. alpha.licensing.

После установки требуется настроить службы Alpha.Domain и Alpha.Net для исторического сервера. Для этого в каталоге «/opt/Automiq/Alpha.Domain» вносим изменения в файл «alpha.domain.agent.xml» следующим образом:

```
<?xml version="1.0" encoding="utf-8"?>
<Alpha.Domain.Agent Name="NDA">
  <EntryPointNetAgent Name="local" Address="127.0.0.1" Port="1010"/>
  <InstalledComponents>
    <Alpha.Server Name="Server_1" ServiceName="Alpha.Server" />
  </InstalledComponents>
  <Server>
    <Components StoragePath="/usr/local/DomainStorage/cache/server">
      <Component InstalledName="Server_1" Name="Server" />
    </Components>
  </Server>
  <Options LoggerLevel="2"/>
</Alpha.Domain.Agent>
```

Рис. 10 Содержимое файла «alpha.net.agent.xml» для исторического сервера

Далее вносим изменения в файл «alpha.net.agent.xml» следующим образом:

```
<Alpha.Net.Agent Name="local" NetEnterPort="1010" ParentAgentPort="1020">
  <Options LoggerLevel="2"/>
</Alpha.Net.Agent>
```

Рис. 11 Содержимое файла «alpha.net.agent.xml» для исторического сервера

```
<?xml version="1.0" encoding="utf-8"?>
<Alpha.Historian.Server StatPort="3388" DefaultPrimaryDir="/opt/Alpha.Historian/Databases" MaxPoolMemorySize="512" EnableDCOM="0" >
  <tcp-server default-port="4949" idle-sessions-count=1 idle-sessions-timeout=15>
    <server-endpoint host="0.0.0.0" />
  </tcp-server>
  <Bases>
    <Base Alias="History" PreferredCommonCacheLimit="128" ActiveStorageDepth="3" StorageDepth="365" />
  </Bases>
</Alpha.Historian.Server>
```

Рис. 12 Содержимое файла «Alpha.Historian.Server.xml» для исторического сервера

После внесения изменений необходимо перезапустить соответствующие три службы в Astra Linux командами в терминале «fly»:

«sudo systemctl restart Alpha.Net.service»  
 «sudo systemctl restart Alpha.Domain.service».

<sup>9)</sup> <https://automiq.ru/ru/docs/Index.htm#Subsystems/Historian/Admin-manual/Work-principles.htm>

## НГО-СОДУ.28.99.39.190 РСП

## 4.4. Программы для станции развертывания

Станция развертывания используется в СОДУ только для загрузки проекта развертывания на все узлы сети и для диагностики работоспособности. Станция развертывания не принимает участие в обмене данными в процессе работы СОДУ. Для установки программного обеспечения нужно обладать правами Администратора. Устанавливаем программы в следующей последовательности:

1. alpha.server;
2. alpha.devstudio;
3. alpha.tools;
4. alpha.domain;
5. alpha.hmi.desktop;
6. alpha.hmi.charts;
7. alpha.hmi.tables;
8. alpha.hmi.alarms;
9. alpha.hmi.commonlib;
10. alpha.hmi.trends;
11. alpha.security;
12. alpha.hmi.security.

После установки требуется настроить службы Alpha.Domain, Alpha.Net и Alpha.Security для станции развертывания. Для этого в каталоге «C:\Program Files\Automatic\Alpha.Domain» вносим изменения в файл «alpha.domain.agent.xml» следующим образом:

```
<?xml version="1.0" encoding="utf-8"?>
<Alpha.Domain.Agent Name="SDKU ADA">
  <EntryPointAlphaNetAgent Name="Node_Domain" Port="1010" />
  <InstalledComponents>
    <Alpha.Server Name="InstalledAlphaSvr1" ServiceName="Alpha.Server" />
    <Alpha.Server Name="InstalledAlphaSvr2" ServiceName="Alpha.AccessPoint" />
  </InstalledComponents>
  <Domain>
    <WorkstationRoles>
      <WorkstationRole Name="THIN_CLIENT">
        <KnownWorkstations>
        </KnownWorkstations>
      </WorkstationRole>
    </WorkstationRoles>
    <ConfigurationCache Path="C:/DomainStorage"/>
  </Domain>
  <Server>
    <Components StoragePath="C:/DomainStorage">
      <Component InstalledName="InstalledAlphaSvr1" Name="AlphaServer"/>
      <Component InstalledName="InstalledAlphaSvr2" Name="AccessPoint"/>
    </Components>
  </Server>
</Alpha.Domain.Agent>
```

Рис. 13 Содержимое файла «alpha.domain.agent.xml» для станции развертывания

## НГО-СОДУ.28.99.39.190 РСП

Далее вносим изменения в файл «alpha.net.agent.xml» следующим образом:

```
<?xml version="1.0" encoding="utf-8"?>
<Alpha.Net.Agent Name="Node_Domain" ParentAgentPort="1000" NetEnterPort="1010">
  <ChildAgents>
    <ChildAgent Name="Node_SERVER1" Address="10.30.200.11" Port="1111" />
    <ChildAgent Name="Node_SERVER2" Address="10.30.200.16" Port="1112" />
  </ChildAgents>
</Alpha.Net.Agent>
```

Рис. 14 Содержимое файла «alpha.net.agent.xml» для станции развертывания

Затем вносим изменения в файл «alpha.security.agent.xml» следующим образом:

```
<Alpha.Security.Agent>
  <EntryPointNetAgent Address="127.0.0.1" Port="1010"/>
  <LogHost>
    <LogServer Address="10.30.200.32" Port="309"/>
  </LogHost>
  <LogUser value="cmeManager.dcomacro.dcom"/>
  <LogPassword value="E91e20xts41ajqE3L1a77a98Q7MU+/23GmhJw3JbTV398d1w1L5889W4Nq/7001763N7g8d4d4n6Fqz2U0vY70w113u0Q42aA20H9Q174/3ta/QuRq6J74+0DpX1d8U7a95041ag09811x75H28E209yk"/>
  <LogSecure value="False"/>
  <SecurityDb value="msAlphaSecurity.dcomacro.dcom"/>
  <DefaultUser value=""/>
  <DefaultNetPassword value=""/>
  <HostDisplayName value=""/>
  <netPrefix value=""/>
  <AuditLogConnection TracesOnly="0">
    <Server Start="127.0.0.1" Type="DB" Protocol="AF_OECD&Server" TCPServerPort="4388" HostTcpServer="" HostTcpServerClient="">
      <SecurityMap>
        <Severity Category="Critical" Value="300"/>
        <Severity Category="Important" Value="200"/>
        <Severity Category="Info" Value="100"/>
        <Severity Category="Debug" Value="0"/>
      </SecurityMap>
      <SignalMap>
        <Signal Name="DnsEvents.NormalDnsSignal" Node="DynamicEvent" Type="Normal"/>
        <Signal Name="DnsEvents.AdminDnsSignal" Node="DynamicEvent" Type="Admin"/>
        <Signal Name="DnsEvents.UserNameDnsSignal" Node="DynamicEvent" Type="UserName"/>
        <Signal Name="DnsEvents.DisplayDnsSignal" Node="DynamicEvent" Type="DisplayName"/>
        <Signal Name="DnsEvents.GroupNameDnsSignal" Node="DynamicEvent" Type="GroupName"/>
        <Signal Name="DnsEvents.WorkstationNameDnsSignal" Node="DynamicEvent" Type="WorkstationName"/>
        <Signal Name="DnsEvents.NormalMessage" Node="Value" Type="Normal"/>
        <Signal Name="DnsEvents.AdminMessage" Node="Value" Type="Admin"/>
        <Signal Name="DnsEvents.UserNameMessage" Node="Value" Type="UserName"/>
        <Signal Name="DnsEvents.DisplayNameMessage" Node="Value" Type="DisplayName"/>
        <Signal Name="DnsEvents.GroupNameMessage" Node="Value" Type="GroupName"/>
        <Signal Name="DnsEvents.WorkstationNameMessage" Node="Value" Type="WorkstationName"/>
      </SignalMap>
    </Server>
  </AuditLogConnection>
  <Options LogFile="E:\netlogs\1" NetName="1" NetIP="10.30.200.32" NetPort="309" NetHost="127.0.0.1" NetLogConnection="0" />
</Alpha.Security.Agent>
```

Рис. 15 Содержимое файла «alpha.security.agent.xml» для станции развертывания

После внесения изменений необходимо перезапустить соответствующие три службы:

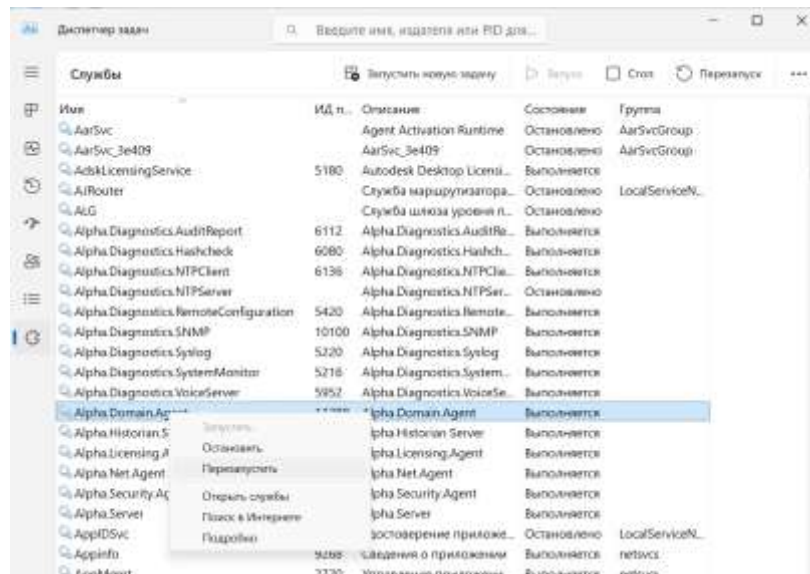


Рис. 16 Перезапуск служб Alpha.Domain.Agent, Alpha.Net.Agent и Alpha.Security.Agent

## НГО-СОДУ.28.99.39.190 РСП

Затем запускаем программу «SecurityConfigurator»<sup>10)</sup> из меню «Пуск». В строке «сервер» вводим IP адрес основной серверной станции, отключаем «TLS». При первом входе пользователь «root», пароль - любой. При запросе «Создать новую» отвечаем «да». Вводим логин «admin», пароль «secret».

После успешного входа в программе ждем «Приложения», «Добавить». Пишем название «НМІ».

Кликаем на «Логическое право», даем название «Оренпрорир» и ждем ввод. Ждем на него, сохраняем. Даем описание – «Право на открытие формы управления». Кликаем на дискету(сохранить) и возвращаемся назад.

Создаем группы: нажимаем на вкладку «Группы», «Добавить». Даем название «Диспетчеры», сохраняем. Тоже самое для группы «Операторы».

Выделить группу «Диспетчеры», нажать «Править», «Добавить права». НМІ -> «Право на открытие формы управления». Установить галочку «Разрешить». Сохранить и назад.

Выделить группу «Операторы», нажать «Править», «Добавить права». НМІ -> «Право на открытие формы управления». Установить галочку «Запретить». Сохранить и назад.

Создаем пользователей или подтягиваем из «ActiveDirectory»: Нажимаем «Пользователи», «Добавить». Выбираем средний значок «Локальная учетная запись Security», пишем логин «Dispatcher», пароль(7 символов), ФИО если надо. В поле «Группы» указываем «Диспетчеры». Снимаем галку «Требовать смены пароля при следующем входе в систему». Сохраняем. Создаем еще пользователей, процесс аналогичен. Сохраняем.

В диспетчере задач перезапускаем службу «Alpha.Security.Agent».

Для загрузки проекта в подготовленные узлы сети СОДУ необходимо запустить программу «Alpha.DevStudio»<sup>11)</sup> из меню «Пуск». Затем открыть решение(проект), расположенный на flash носителе «NGO.solution» и перейти к развертыванию решения на узлы сети как показано на рисунке 16:

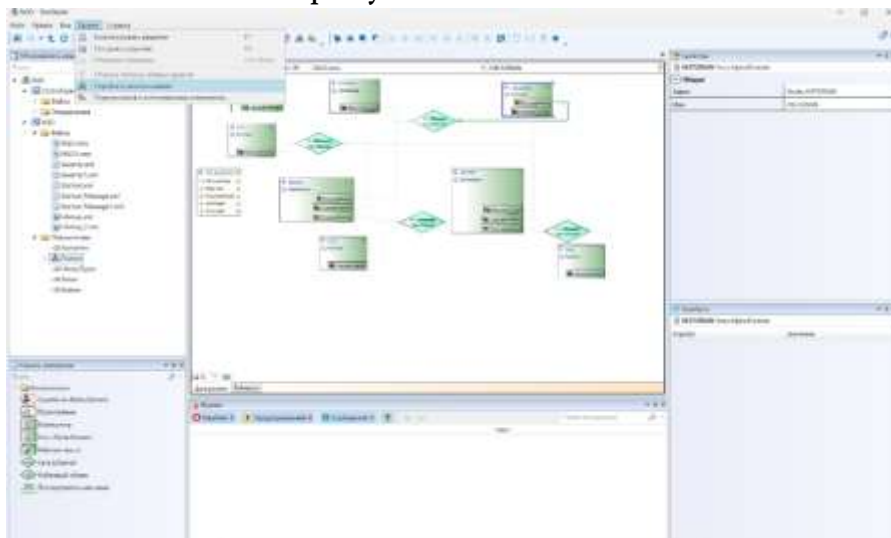


Рис. 17 Среда разработки «Alpha.DevStudio»

<sup>10)</sup> <https://automiq.ru/ru/docs/Index.htm#Subsystems/Security/Admin-manual/Configurator/About.htm>

<sup>11)</sup> <https://automiq.ru/ru/docs/Index.htm#Subsystems/DevStudio/About.htm>

## НГО-СОДУ.28.99.39.190 РСП

В открывшемся окне выделить все станции и кликнуть на иконку «Применить все»

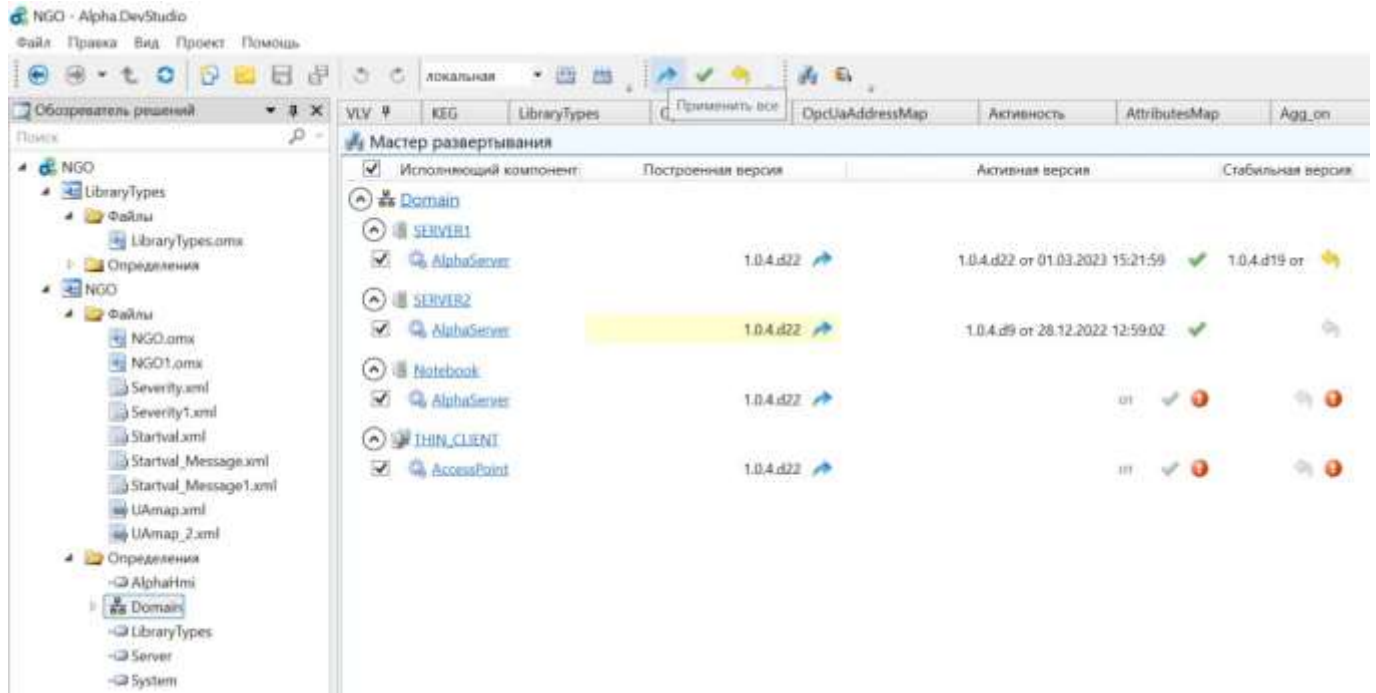


Рис. 18 Экран развертывания решения на станции

## 5. АДМИНИСТРИРОВАНИЕ ПРОГРАММНОГО КОМПЛЕКСА

Резервное копирование и восстановление СОДУ возможно производить целиком, либо частично и оно заключается в резервном копировании/восстановлении каждого узла системы. Резервное копирование серверных станций осуществляется последовательно для сохранения режима работы 24/7. Резервное копирование БД исторического сервера возможно производить без остановки его служб благодаря наличию нескольких стадий хранения и порционной записи данных в архивную область БД.

Резервное копирование узлов системы СОДУ целиком выполняется с помощью специализированного программного обеспечения – системы резервного копирования (далее СРК).

Система резервного копирования — совокупность программного и аппаратного обеспечения, выполняющее задачу создания копии данных на носителе, предназначенном для восстановления информации в оригинальном месте их расположения в случае их повреждения или разрушения.

Таблица 1. Системы резервного копирования

Название	Реестр российского ПО	Сертификат ФСТЭК	Кроссплатформенность
Кибер Бэкап (Acronis)	Да	Да	Да
RuBackup	Да	Нет	Да
Handy Backup	Да	Нет	Да

Для применения в программном комплексе СОДУ рекомендуется использовать СРК «Кибер Бэкап»<sup>12)</sup>.

Система «Кибер Бэкап» состоит из сервера управления и агентов защиты. Сервер управления — это центр управления всеми резервными копиями. Сервер управления отвечает за обмен данными с агентами и выполняет общие функции управления планом резервирования. Перед выполнением любого действия защиты агенты обращаются к серверу управления для проверки предварительных требований. Агент защиты устанавливается на каждой машине, для которой планируется создать резервную копию. Агент защиты осуществляет процесс резервирования/восстановления в соответствии с заранее составленным скриптом команд.

Для исторического сервера при частичном резервном копировании далее описывается процесс создания копии базы данных исторических событий и данных, процедура восстановления этой БД из копии.

Для серверных станций помимо полного резервирования средствами СРК описана процедура копирования служб и каталогов LDAP.

<sup>12)</sup> <https://docs.cyberprotect.ru/ru-RU/CyberBackup/16/user/>

## 5.1 Резервное копирование узлов СОДУ

### 5.1.1 Исторический сервер

Модуль Alpha.Server, расположенный на обоих серверах, обеспечивает сбор, фильтрацию и сохранение данных в БД Alpha.Historian, расположенную на историческом сервере, через модуль истории. Модуль истории в составе Alpha.Server выполняет временное хранение данных на стороне сервера и передачу информации в хранилище Alpha.Historian на историческом сервере.



Рис. 19 Принцип сохранения данных в БД Alpha.Historian

База данных Alpha.Historian является собственной разработкой «Атомик Софт». Представлена в виде нереляционной СУБД. Получить доступ к данным базы можно получить только через визуализацию проекта развертывания в Alpha.HMI.

Данные в БД хранятся в закрытом бинарном формате. Сервер допускает лишь добавление новых записей. В сервере истории данные хранятся в фрагментах - файлах, содержащих данные за сутки. После поступления в сервер истории, данные последовательно проходят несколько стадий хранения: в активной области и в архивной области. Длительность хранения фрагментов в активной области задаётся в настройках БД<sup>13)</sup> (в файле «Alpha.Historian.Server.xml» параметр «ActiveStorageDepth». По умолчанию 3 суток). Фрагменты, нижняя граница которых оказывается левее предела времени хранения в активной области, переносятся в архивную область. Глубина хранения данных ограничена размерами дискового пространства.

Буфер временного хранения данных на стороне серверов располагается на жестком диске в каталоге «/opt/Automiq/Alpha.Server/Queues», что предотвращает потерю данных при аварийном отключении компьютера сервера исторических данных. При следующем старте исторического сервера непереданные данные будут повторно отправлены в Alpha.Historian. Буферизация данных позволяет также сгладить пиковые нагрузки при большой интенсивности получения данных.

Далее описаны два варианта создания резервной копии: только базы данных и полное резервирование средствами СРК. Копирование БД исторического сервера можно производить без остановки его работы.

<sup>13)</sup> <https://automiq.ru/ru/docs/Index.htm#Subsystems/Historian/Admin-manual/Set-up.htm>

## НГО-СОДУ.28.99.39.190 РСР

### 5.1.1.1 Резервирование базы данных

В файловой системе в каталоге «/opt/Alpha.Historian/Databases/History» база данных содержит следующие папки и файл:

- /active
- /archive
- /imported
- /stx
- db.index

Папки и файл создаются автоматически, когда сервер истории создает базу данных. Их расположение в файловой системе зависит от описания базы данных в файле конфигурации «Alpha.Historian.Server.xml» в каталоге «opt/Automiq/Alpha.Historian» (атрибут «DefaultPrimaryDir» сервера истории).

Для резервного копирования БД исторического сервера необходимо:

1. Скопировать в пустую папку файл «db.index» из структуры базы данных.
2. Скопировать в нее папку «archive» из структуры базы данных.  
В ней данные сгруппированы в подпапки по датам и источникам.

Чтобы дополнить(обновить) созданную ранее копию архивных данных нужно:

1. Скопировать с заменой файл «db.index» из базы данных в копию архива.
2. В копию архива скопировать подпапки с данными от интересующих источников и за нужные даты.

### 5.1.1.2 Полное резервирование средствами СРК

При запуске процедуры резервирования исторического сервера агент СРК должен управлять службами Alpha Platform в следующем порядке:

1. Остановить работу служб на этом историческом сервере командами:  
«sudo systemctl stop alpha.Net.service»,  
«sudo systemctl stop alpha.Domain.service»,  
«sudo systemctl stop alpha.historian.server.service»;
2. Создать бэкап системы средствами СРК;
3. Произвести запуск ранее остановленных служб командами:  
«sudo systemctl start alpha.Net.service»,  
«sudo systemctl start alpha.Domain.service»,  
«sudo systemctl start alpha.historian.server.service».



### 5.1.2 Серверная станция

Резервное копирование серверных станций осуществляется последовательно для сохранения режима работы 24/7. Процедура резервного копирования серверных станций идентична и описана в данном подразделе. Помимо полного резервирования станции средствами СРК описана процедура копирования служб и каталогов LDAP.

#### 5.1.2.1 Резервирование средствами СРК

При запуске процедуры резервирования серверной станции агент СРК должен управлять службами Alpha Platform в следующем порядке:

1. Остановить работу служб на сервере командами:

```
«sudo systemctl stop alpha.net.service»,  
«sudo systemctl stop alpha.domain.service»,  
«sudo systemctl stop alpha.server.service»,  
«sudo systemctl stop alpha.security.service»;  
«sudo service nslcd stop»;  
«sudo service nscd stop»;  
«sudo service slapd stop».
```

После остановки служб сервер будет переведен в режим резервного без потери данных. Резервный сервер станет основным.

2. Создать бэкап системы средствами СРК;
3. Произвести запуск ранее остановленных служб командами:

```
«sudo systemctl start alpha.net.service»,  
«sudo systemctl start alpha.domain.service»,  
«sudo systemctl start alpha.server.service»,  
«sudo systemctl start alpha.security.service»;  
«sudo service nslcd start»;  
«sudo service nscd start»;  
«sudo service slapd start».
```

Сервер останется в режиме «горячего резервирования».

#### 5.1.2.2 Резервное копирование сервера LDAP

На серверной станции возможно сделать резервную копию содержимого базы данных LDAP, конфигурации сервера LDAP и общесистемной конфигурации сервера. База данных и конфигурация сервера LDAP могут быть выгружены в LDIF-файлы с помощью «slapcat». Удобно запаковать их вместе в один tar-файл. Общесистемная конфигурация — это всё остальное, например, набор ключей и файл */etc/defaults/slapd*. Создадим файл для скрипта резервного копирования, а также каталоги, куда будем складировать данные:

```
«mkdir -p /root/scripts /root/backup /var/log/backup»  
«touch /root/scripts/backup.slapd.sh»  
«chmod +x /root/scripts/backup.slapd.sh»
```

**НГО-СОДУ.28.99.39.190 РСП**

И запишем в файл *«backup slapd.sh»* скрипт:

```
#!/bin/sh
# Копируем данные и конфигурацию LDAP в сжатые LDIF-файлы.
# А так же резервируем весь каталог LDAP и конфигурацию демона slapd.
umask 022
PATH="/bin:/usr/bin:/usr/local/bin:/sbin:/usr/sbin:/usr/local/sbin"
export PATH
DATE=`date +%Y%m%d`
BACKUP_DIR="/root/backup/slapd"
BACKUP_DATA_FILENAME="slapd.data.${DATE}.ldif"
BACKUP_CONFIG_FILENAME="slapd.config.${DATE}.ldif"
BACKUP_TAR_FILENAME="slapd.directory.${DATE}.tar.gz"
CA_TLS_CERT="/etc/ssl/certs/rootca.crt"
DIT_CONFIG="cn=config"
DIT_SUFFIX="dc=example,dc=com"

SLAPD_CONFIG_FILENAME="/etc/default/slapd"
SLAPD_DIR="/etc/ldap"
SLAPD_LOG_ROTATION="/etc/logrotate.d/slapd"
SLAPD_TLS_CERT="/etc/ldap/ssl/ldap-srv.example.com.crt"
SLAPD_TLS_KEY="/etc/ldap/ssl/ldap-srv.example.com.key"
SLAPCAT_OPTIONS="-F /etc/ldap/slapd.d"

LOGFILE="/var/log/backup/slapd.log"
KEEP="30"

# Убедимся, что скрипт запущен от имени суперпользователя
if [ `id -u` -ne "0" ]; then
    echo "ERROR: only root can run this script." | tee -a ${LOGFILE}
    exit 1
fi

# Проверим, есть ли у нас файл для журнала
if [ ! -f ${LOGFILE} ]; then
    touch ${LOGFILE}

    if [ "$?" -ne "0" ]; then
        echo "ERROR: could not create the log file."
        exit 1
    fi
fi

# Убедимся, что у нас есть каталог для резервных копий
if [ ! -d ${BACKUP_DIR} ]; then
    mkdir -p ${BACKUP_DIR}

    if [ "$?" -ne "0" ]; then
        echo "ERROR: could not create the backup directory." | tee -a ${LOGFILE}
        exit 1
    fi
fi
```

## НГО-СОДУ.28.99.39.190 РСП

```

# Убедимся, что в нашем каталоге с резервными копиями не свалено слишком много файлов
# и удалим все резервные копии кроме последних ${KEEP} копий.
FILES=`find ${BACKUP_DIR} -type f -name "slapd.*" -print | wc -l`

if [ "${FILES}" -gt "${KEEP}" ]; then
  OVER=`echo ${FILES}-${KEEP} | bc`
  RMFILES=`find ${BACKUP_DIR} -type f -name "slapd.*" -print | sort -r | tail -${OVER}`
  echo "NOTE: removing ${RMFILES} from the backup directory." >> ${LOGFILE}
  rm ${RMFILES}
fi

# Создаём резервную копию данных из DIT
slapcat ${SLAPCAT_OPTIONS} -b ${DIT_SUFFIX} -l ${BACKUP_DIR}/${BACKUP_DATA_FILE-
NAME} >/dev/null 2>&1

if [ "$?" -eq "0" ]; then
  gzip -f ${BACKUP_DIR}/${BACKUP_DATA_FILENAME} 2>&1 >> ${LOGFILE}

  if [ "$?" -ne "0" ]; then
    echo "ERROR: dump file compression problem." | tee -a ${LOGFILE}
    exit 1
  fi
else
  echo "ERROR: problem running slapcat(8C) for the DIT data backup." | tee -a ${LOGFILE}
  rm ${BACKUP_DIR}/${BACKUP_DATA_FILENAME}
  exit 1
fi

# Сохраняем конфигурацию DIT в виде LDIF-файла
slapcat ${SLAPCAT_OPTIONS} -b ${DIT_CONFIG} -l ${BACKUP_DIR}/${BACKUP_CON-
FIG_FILENAME} >/dev/null 2>&1

if [ "$?" -eq "0" ]; then
  gzip -f ${BACKUP_DIR}/${BACKUP_CONFIG_FILENAME} 2>&1 >> ${LOGFILE}

  if [ "$?" -ne "0" ]; then
    echo "ERROR: dump file compression problem." | tee -a ${LOGFILE}
    exit 1
  fi
else
  echo "ERROR: problem running slapcat(8C) for the DIT config backup." | tee -a ${LOGFILE}
  rm ${BACKUP_DIR}/${BACKUP_CONFIG_FILENAME}
  exit 1
fi

# Делаем резервную копию файлов каталога и конфигурации
BACKUP_FILES_LIST="${CA_TLS_CERT} ${SLAPD_CONFIG_FILENAME} ${SLAPD_DIR}
${SLAPD_LOG_ROTATION} ${SLAPD_TLS_CERT} ${SLAPD_TLS_KEY}"

tar zcf ${BACKUP_DIR}/${BACKUP_TAR_FILENAME} ${BACKUP_FILES_LIST} >/dev/null
2>&1

```

**НГО-СОДУ.28.99.39.190 РСП**

```
if [ "$?" -ne "0" ]; then
  echo "ERROR: problem running config directory tar." | tee -a ${LOGFILE}
  rm ${BACKUP_DIR}/${BACKUP_TAR_FILENAME}
  exit 1
fi
```

По этому правилу резервное копирование можно запускать вручную или создать задание для планировщика «cron», например, с ежедневным запуском. Копируйте файлы на другую машину в сети.

**5.1.3 Станция АРМ**

При запуске процедуры резервирования станции АРМ агент СРК должен управлять службами Alpha Platform в следующем порядке:

1. Остановить работу служб на сервере командами:  
«sudo systemctl stop alpha.net.service»,  
«sudo systemctl stop alpha.domain.service»,  
«sudo systemctl stop alpha.accesspoint.service»,  
«sudo systemctl stop alpha.security.service»;
2. Создать бэкап системы средствами СРК;
3. Произвести запуск ранее остановленных служб командами:  
«sudo systemctl start alpha.net.service»,  
«sudo systemctl start alpha.domain.service»,  
«sudo systemctl start alpha.accesspoint.service»,  
«sudo systemctl start alpha.security.service».

## 5.2 Резервное восстановление узлов СОДУ

### 5.2.1 Исторический сервер

Алгоритм восстановления зависит от масштаба повреждения исторического сервера. Если перестали отображаться тренды и журнал истории на мнемосхеме АРМ, то требуется восстановить архивные данные на историческом сервере.

Для восстановления сохраненных архивных данных нужно:

1. В файл конфигурации сервера истории «Alpha.Historian.Server.xml» из каталога «opt/Automiq/Alpha.Historian» добавить новую базу данных. Для нее не указывать глубину хранения (по умолчанию – не ограничена) или указать неограниченную глубину хранения: `StorageDepth="0"`. В противном случае сервер истории при запуске может удалить данные, имеющие большую глубину относительно текущей даты.
2. Перезапустить сервис «Alpha.Historian» командой «`sudo systemctl restart astra.historian.server.service`».

В результате в файловой системе будут созданы файл и папки для новой базы данных.

3. Остановить сервис «Alpha.Historian» командой «`sudo systemctl stop alpha.historian.server.service`».
4. Заменить в папке базы данных файл «db.index» одноименным файлом из сохраненного архива.
5. Из папки «archive» сохраненного архива скопировать папки интересующих источников и за нужные для просмотра даты в одноименную папку базы данных.
6. Запустить сервис «Alpha.Historian» командой «`sudo systemctl start alpha.historian.server.service`».

После этого восстановленные архивные данные можно запрашивать из добавленной базы данных.

Если архивный сервер вышел из строя, то первым делом нужно найти другой или починить этот. Дальнейший процесс восстановления можно сделать средствами СРК при наличии полной резервной копии, либо самостоятельно.

При самостоятельном восстановлении после установки ОС Astra Linux требуется установить пакеты программ Alpha Platform в соответствии с пунктом 4.3 настоящего руководства. Потом скопировать на неё файлы и папки из резервной копии БД(созданную в пункте 5.1.1.1) и выполнить процедуру восстановления сохраненных архивных данных, описанную выше.

При необходимости восстановления средствами СРК исторического сервера целиком необходимо обратиться к документации выбранной СРК.

### 5.2.2 Серверная станция

Сценарий восстановления зависит от типа решаемой проблемы:

1. Полная потеря сервера;
2. Проблемы со списком управления доступа (ACL);
3. Повреждение данных (или человеческий фактор).

1. Если серверная станция сломалась, то первым делом нужно найти другую или починить эту. Дальнейший процесс восстановления можно сделать средствами СРК при наличии полной резервной копии, либо самостоятельно.

При самостоятельном восстановлении после установки ОС Astra Linux и пакетов программ Alpha Platform в соответствии с пунктом 4.1 настоящего руководства, необходимо скопировать на неё файлы из резервной копии и выполнить следующие команды:

```
# apt-get install -y slapd ldap-utils krb5-kdc-ldap krb5-pkinit krb5-admin-server libnss-ldapd libpam-ldapd wamerican sudo-ldap
# mv /var/lib/ldap /val/lib/ldap.install
# mkdir /var/lib/ldap
# zcat /root/backup/slapd/slapd.data.20231005.ldif.gz > /tmp/slapd.data.ldif
# slapadd -v < /tmp/slapd.data.ldif
# chown -R openldap:openldap /var/lib/ldap
# mv /etc/ldap /etc/ldap.install
# cd / && tar zxvf /tmp/slapd.directory.20231005.tar.gz
# update-rc.d slapd defaults
# service slapd start
```

2. Проблемы со списком управления доступа (ACL) могут возникнуть при настройке сервера. При изменении ACL можно закрыть самому себе любой доступ к нему. Поправить ситуацию можно выполнив команды:

```
# service nslcd stop
# service nscd stop
# service slapd stop
# mv /etc/ldap /etc/ldap.broken
# cd / && tar zxvf /root/backup/slapd/slapd.directory.20231005.tar.gz
# service slapd start
# service nscd start
# service nslcd start
```

3. Самый простой способ исправления ошибки, связанной с повреждением данных — остановить демон «slapd» и вернуть данные из резервной копии с помощью «slapadd»:

```
# service nslcd stop
# service nscd stop
# service slapd stop
# mv /var/lib/ldap /var/lib/ldap.broken
# mkdir /var/lib/ldap
# zcat /root/backup/slapd/slapd.data.20231005.ldif.gz > /tmp/slapd.data.ldif
# slapadd -v < /tmp/slapd.data.ldif
# chown -R openldap:openldap /var/lib/ldap
# service slapd start
# service nscd start
# service nslcd start
```

### 5.2.3 Станция АРМ

Если станция АРМ вышла из строя, то первым делом нужно найти другой ПК или починить этот. Дальнейший процесс восстановления можно сделать средствами СРК при наличии полной резервной копии станции АРМ, либо самостоятельно. При самостоятельном восстановлении необходимо руководствоваться пунктом 4.2.

### 5.3 Синхронизация времени

Все узлы сети СОДУ имеют синхронизацию с сервером точного времени NTP, расположенным в локальной сети по IP адресу 10.20.30.2.

Для настройки клиента NTP в ОС Astra Linux необходимо изменить содержимое файла «/etc/ntp.conf».

Затем необходимо перезапустить службу командой: «sudo service ntp restart».

Для автоматического запуска при загрузке ОС: «sudo systemctl enable ntp»

Для проверки состояния сервиса необходимо воспользоваться командами:

«systemctl status ntp» или «sudo ntpq -c sysinfo».





## **НГО-СОДУ.28.99.39.190 РСР**

Если на основном экране мнемосхемы отсутствует выпадающее меню со списком пользователей, значит не корректно работает служба Alpha.Security. Самой частой проблемой может стать истекший срок действия паролей для пользователей различных групп. Для устранения необходимо внести новые пароли с инженерной станции с помощью программы «SecurityConfigurator».

### **7. СООБЩЕНИЯ СИСТЕМНОМУ ПРОГРАММИСТУ**

При инсталляции пакетов программного комплекса возможно появление сообщений о невозможности установки из-за отсутствия каких-то элементов. Для установки недостающего необходимо загрузить их и установить. В Astra Linux можно выполнить команды «`sudo apt update`» и «`sudo apt upgrade`».

## ПЕРЕЧЕНЬ ТЕРМИНОВ И СОКРАЩЕНИЙ

<b>Сокращение</b>	<b>Полное наименование</b>
АРМ	Автоматизированное рабочее место
АСУ ТП	Автоматизированные системы управления технологическим процессом
ГРС	Газораспределительная станция
САУ ГРС	Система автоматического управления газораспределительной станцией
АСУ	Автоматизированная система управления
БД	База данных
КИПиА	Контрольно-измерительные приборы и автоматика
ПК	Программный комплекс
ПО	Программное обеспечение
СОДУ	Система оперативно-диспетчерского управления
ТИ	Телеизмерение
ОС	Операционная система
СУБД	Система управления базами данных
ПК	Программный комплекс
СРК	Система резервного копирования, например, «Кибер Бэкап»

