|  |  |  |  |
| --- | --- | --- | --- |
|  | | Утвержден |  |
| RU.64509942.00213-01-ЛУ |  |
| ПС «Атом.Мост»  Руководство администратора  RU.64509942.00213-01 99 01  Листов 116 | |
| *Подпись и дата* |  |
| *Инв № дубл.* |  |
| *Взам инв. №* |  |
| *Подпись и дата* |  |
| *Инв № подл.* | *B-IB5-1-0023* |
| Литера О1 | |
| 2021 | |

Аннотация

Настоящий документ является руководством системного администратора программного средства «Атом.Мост» (далее по тексту — ПС «Атом.Мост»).

Руководство содержит общие сведения о ПС «Атом.Мост», включающие: назначение, функционал, установку, конфигурацию, настройки идентификации и аутентификации пользователей, политик доступа и другие сведения о платформе.

Руководство предназначено должностным лицам, осуществляющим и обеспечивающим эксплуатацию программного средства.

СОДЕРЖАНИЕ

[1. Общие сведения о программе 6](#_Toc82176048)

[2. Структура программы 8](#_Toc82176049)

[3. Установка программы 11](#_Toc82176050)

[3.1. Установка 11](#_Toc82176051)

[3.2. Конфигурация портов 12](#_Toc82176052)

[3.3. Рекомендации по взаимодействию с другими средствами защитами информации 12](#_Toc82176053)

[3.4. Дополнительные условия установки в ОС 13](#_Toc82176054)

[3.5. Установка среды Java 13](#_Toc82176055)

[3.5.1. ГосJava 13](#_Toc82176056)

[3.5.2. Liberica 14](#_Toc82176057)

[4. Конфигурация безопасности 15](#_Toc82176058)

[4.1. Набор средств генерации TLS 17](#_Toc82176059)

[4.1.1. Standalone 18](#_Toc82176060)

[4.1.2. Client/Server 20](#_Toc82176061)

[4.2. Использование существующего центра сертификации 23](#_Toc82176062)

[4.3. Использование сертификатов с внешней подписью 30](#_Toc82176063)

[4.4. Дополнительные команды сертификации 33](#_Toc82176064)

[5. Идентификация и аутентификация 36](#_Toc82176065)

[5.1. Аутентификация пользователя 36](#_Toc82176066)

[5.1.1. Lightweight Directory Access Protocol 37](#_Toc82176067)

[5.1.2. Kerberos 41](#_Toc82176068)

[5.1.3. Аутентификация через OpenId Connect 42](#_Toc82176069)

[5.2. Мультитентантная авторизация 43](#_Toc82176070)

[5.3. Настройка Authorizers.xml 44](#_Toc82176071)

[5.3.1. FileUserGroupProvider 45](#_Toc82176072)

[5.3.2. LdapUserGroupProvider 45](#_Toc82176073)

[5.3.3. ShellUserGroupProvider 49](#_Toc82176074)

[5.3.4. Составные реализации 50](#_Toc82176075)

[5.3.5. FileAccessPolicyProvider 52](#_Toc82176076)

[5.3.6. StandardManagedAuthorizer 54](#_Toc82176077)

[5.3.7. FileAuthorizer 54](#_Toc82176078)

[5.3.8. Первоначальная идентификация администратора 55](#_Toc82176079)

[6. Политики доступа 67](#_Toc82176080)

[6.1. Глобальные политики доступа 69](#_Toc82176081)

[6.2. Политики доступа на уровне компонентов 71](#_Toc82176082)

[6.3. Наследование политики доступа 73](#_Toc82176083)

[6.4. Создание пользователей и групп 74](#_Toc82176084)

[6.5. Просмотр политик для пользователей 76](#_Toc82176085)

[6.6. Примеры настройки политики доступа 78](#_Toc82176086)

[6.6.1. Перемещение процессора 80](#_Toc82176087)

[6.6.2. Редактирование процессора 81](#_Toc82176088)

[6.6.3. Создание подключения 83](#_Toc82176089)

[6.6.4. Редактирование соединения 84](#_Toc82176090)

[6.7. Идентификационные данные узла кластера 85](#_Toc82176091)

[7. Кластеризация 88](#_Toc82176092)

[7.1. Особенности терминологии 90](#_Toc82176093)

[7.1.1. «Координатор» кластера 90](#_Toc82176094)

[7.1.2. Узлы 90](#_Toc82176095)

[7.1.3. Первичный узел 90](#_Toc82176096)

[7.1.4. Изолированные процессоры 91](#_Toc82176097)

[7.1.5. Такты 91](#_Toc82176098)

[7.2. Обмен информацией внутри кластера 92](#_Toc82176099)

[7.3. Управление узлами 93](#_Toc82176100)

[7.3.1. Отключение узлов 93](#_Toc82176101)

[7.3.2. Выгрузка узлов 93](#_Toc82176102)

[7.3.3. Удаление узла 94](#_Toc82176103)

[7.3.4. Вывод узла из эксплуатации 94](#_Toc82176104)

[7.4. «Выборы» потока 95](#_Toc82176105)

[7.5. Базовая настройка кластера 97](#_Toc82176106)

[8. Обновление системы 100](#_Toc82176107)

[8.1. Предварительные шаги 101](#_Toc82176108)

[8.1.1. Сохранение пользовательских процессоров 101](#_Toc82176109)

[8.1.2. Подготовка кластера 101](#_Toc82176110)

[8.2. Общие шаги 105](#_Toc82176111)

[9. Контроль целостности 107](#_Toc82176112)

[9.1. Утилита генерации хэш-сумм DFSumGenerator 107](#_Toc82176113)

[9.2. Утилита проверки хэш-сумм DFSumChecker 108](#_Toc82176114)

[10. Правила приёмки 109](#_Toc82176115)

[10.1. Общие положения 109](#_Toc82176116)

[10.2. Предъявительские испытания 111](#_Toc82176117)

[10.3. Приёмо-сдаточные испытания 112](#_Toc82176118)

# Общие сведения о программе

ПС «Атом.Мост» – интеграционная платформа для организации и управления потоками данных между информационными системами, информационными ресурсами и т.п. Программное средство позволяет собирать, анализировать и обрабатывать сведения об информационных потоках в режиме реального времени, локально или в облаке с помощью визуального интерфейса.

Программное средство поддерживает разрозненные и распределенные источники данных, различных форматов, схем, протоколов и размеров, позволяет организовать обмен данными, который не зависит от источника данных.

1. Веб-интерфейс пользователя:

* возможность визуально проектирования, управления и мониторинга;
* мультитенантный пользовательский интерфейс;

1. Отслеживание потока данных;
2. Надежность и производительность:

* устойчивость к потерям;
* низкая задержка и высокая пропускной способности;
* динамическая приоритезация;
* внесение изменений в потоки во время выполнения;
* поддержка сжатия;
* масштабируемость с использованием модели кластеризации;

1. Поддержка расширений – возможность создания собственных процессоров на языке Java;
2. Безопасность:

* использование SSL, SSH, HTTPS, а также шифрования;
* подключаемая детальная аутентификация / авторизация на основе ролей;
* настройка управления определенными частями потока и совместного использования разными группами пользователей.

Программное средство дает множество преимуществ, которые помогают «Атом.Мост» быть эффективной интеграционной платформой, позволяющей в режиме реального времени создавать, управлять, изменять и потоки данных в том числе между высоконагруженными информационными ресурсами. Некоторые из этих преимуществ:

* интуитивно понятные средства визуального создания и управления ориентированными графами процессоров;
* асинхронное взаимодействие, что обеспечивает высокую пропускную способность и естественную буферизацию, даже если обработка и скорость потока не являются стабильными;
* поддержка модели с высокой степенью параллелизма, при этом разработчику не нужно беспокоиться о типичных сложностях таких взаимодействий;
* поддержка связанных и слабосвязанных компонентов, которые затем могут быть использованы многократно в разных сценариях обмена данными;
* эффективная обработка ошибок.

# Структура программы

ПС «Атом.Мост» состоит из взаимодействующих друг с другом модулей:

1. Web-сервер

Модуль выступает в качестве точки входа в систему и решает следующие задачи:

* обеспечивает поддержку транспортных протоколов HTTP и HTTPS.
* Web-сервер предоставляет пользовательский интерфейс (UI)
* обеспечивает доступ к API через прикладные REST, SAP XI, SOAP протоколы

1. Модуль управления потоками (Flow Control). Модуль выделяет ресурсы виртуальной машины для запуска расширений, управления планировщиком расширений и непосредственно запуском расширений. Модуль управления потоками решает следующие задачи:

* гарантированная доставка данных;
* буферизация с управляемой связью;
* управлением приоритетами очереди;
* запись маршрута потока;
* перезапуск потока с любого этапа выполнения;
* поддержка очередности заданий для распределения нагрузки;
* объединение данных из нескольких источников в единый поток;
* комплексная трансформация с поддержкой обработки скриптовыми языками;
* валидация данных на соответствие схемы;
* просмотр данных интеграционного потока;
* поддержка библиотеки компонентов и типовых шаблонов интеграционных сценариев.

1. Процессоры-расширения. Это отдельные программные модули для обработки данных, передаваемых через шину. Процессоры выступают в качестве «коннекторов» к сторонним (относительно шины) системам.
2. Хранилище файлов потока (FlowFile Repository). Место, где система отслеживает состояние о файле потока, который в настоящее время находится в потоке. Это подключаемый репозиторий. Постоянный журнал предварительно записи, расположенный на выбранном разделе диска.
3. Хранилище данных (Content Repository). Место, где хранятся файлы потоков. Является подключаемым репозиторием. Данные в репозитории хранятся блоками на файловой системе. Имеется возмоность подключать несколько хранилищ данных в разных местах файловой системы, для того чтобы равномерно распределить нагрузку на дисковую подсистему.
4. Репозиторий происхождения (Provenance Repository). Репозиторий, который хранит историю о каждом потоковом файле и операциях с ним (создание, изменение и т.д.). Подключаемый репозиторий, может размещаться как на одном, так и на нескольких физических дисках. В рамках каждого данные о событиях индексируются и доступны для поиска.
5. Модуль аутентификации и авторизации. Модуль предназначен для аутентификациии пользователей и разграничения доступа к ресурсам системы при помощи внешних систем, при этом учетные данные (логины пароли, контакты) в системе не хранятся. Модуль решает следующие задачи:

* предоставляет средства по аутентификации пользователей при помощи подключаемого механизма для аутентификации пользователей. Поддерживаются механизмы аутентификации по логину паролю через LDAP Kerberos, OpenID и пользовательским сертификатам x.509;
* реализует протокол межсервисной авторизации OAuth2;
* выполняет разграничение доступа к интеграционным потокам;
* выполняет контроль целостности компонентов безопасности;
* выполняет защиту доступа от анонимного доступа к узлу.

1. Модуль авторизации. Позволяет производить настройку конфигурации пользователей, групп и политик как в пользовательском интерфейсе, так и на глобальном уровне.
2. Модуль политики доступа. Модуль управления пользователями и группами платформы с помощью политик доступа.

# Установка программы

## Установка

Для установки ПС «Атом.Мост» необходимо распаковать файл дистрибутива в желаемый каталог.

Далее необходимо внести изменения в файлы, находящиеся в папке <installdir> / conf.

В каталоге <installdir> / bin выполните команды, набрав ./nifi.sh <command> в терминале:

* start: запускает ПС «Атом.Мост» в фоновом режиме;
* stop: останавливает ПС «Атом.Мост», работающее в фоновом режиме;
* status: показывает текущий статус ПС «Атом.Мост»;
* run: запускает ПС «Атом.Мост» на переднем плане и ждет, пока <Ctrl-C> инициирует выключение ПС «Атом.Мост»;
* install: устанавливает ПС «Атом.Мост» как сервис, которым затем можно управлять через:
* сервис nifi start
* сервис nifi stop
* статус службы nifi.

При первом запуске ПС «Атом.Мост» создаются следующие файлы и каталоги:

* content\_repository
* database\_repository
* flowfile\_repository
* originance\_repository
* каталог work
* каталог logs
* в каталоге conf создается файл flow.xml.gz.

## Конфигурация портов

В таблице 1 перечислены порты по умолчанию, используемые ПС «Атом.Мост», и соответствующее свойство в файле nifi.properties.

1. Порты по умолчанию

|  |  |  |
| --- | --- | --- |
| Функция | Свойство | Значение по умолчанию |
| Web HTTP Forwarding Port | nifi.web.http.port.forwarding | none |
| HTTP Port | nifi.web.http.port | 8080 |
| HTTPS Port\* | nifi.web.https.port | 9443 |
| Remote Input Socket Port\* | nifi.remote.input.socket.port | 10443 |
| Cluster Node Protocol Port\* | nifi.cluster.node.protocol.port | 11443 |
| Cluster Node Load Balancing Port | nifi.cluster.node.load.balance.port | 6342 |

|  |
| --- |
|  |

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!** Порты, отмеченные звездочкой (\*), имеют значения свойств, которые по умолчанию пусты в файле nifi.properties. Значения, показанные в таблице, являются значениями по умолчанию для портов, после использования TLS Toolkit для создания файла nifi.properties для защищенного экземпляра ПС «Атом.Мост». Порт центра сертификации по умолчанию, используемый TLS Toolkit – 8443. |  |

## Рекомендации по взаимодействию с другими средствами защитами информации

Антивирусное программное обеспечение может долго сканировать большие каталоги и многочисленные файлы в них. Кроме того, если антивирусное программное обеспечение блокирует файлы или каталоги во время сканирования, эти ресурсы недоступны для процессов ПС «Атом.Мост», вызывая задержку или недоступность этих ресурсов в экземпляре / кластере. Чтобы предотвратить возникновение этих проблем с производительностью и надежностью, рекомендуется настроить антивирусное программное обеспечение так, чтобы оно пропускало сканирование в следующих каталогах:

* content\_repository
* flowfile\_repository
* logs
* provenance\_repository
* state.

## Дополнительные условия установки в ОС

В различных ОС в зависимости от их состава необходим выполнить некоторые дополнительные действия перед установкой ПС.

## Установка среды Java

В состав некоторых сертифицированных ОС по умолчанию не входит исполняемая среда для Java-приложений (например – Astra Linux и Альт Линукс).

Для функционирования ПС «Атом.Мост» необходимо установить в ОС дополнительные пакеты исполняемой среды.

### ГосJava

Для установки пакета ГосJava создайте файл /etc/apt/sources.list.d/gosjava.list:

deb http://packages.lab50.net/gosjava/se16 gosjava main

Далее вам необходимо добавить цифровой ключ подписи в APT:

wget -qO - http://packages.lab50.net/lab50.asc | sudo apt-key add -

Выполните aptitude update и установите пакет gosjava-jre.

### Liberica

Выполните следующие команды:

wget -q -O - https://download.bell-sw.com/pki/GPG-KEY-bellsoft | sudo apt-key add -

echo "deb [arch=amd64] https://apt.bell-sw.com/ stable main" | sudo tee /etc/apt/sources.list.d/bellsoft.list

Репозитории Liberica JDK содержат все архитектуры Linux, поддерживаемые в данной версии: amd64, i386, arm64, armhf. Если amd64 не является целевой архитектурой, замените amd64 в приведенной выше команде на целевую архитектуру.

Обновите репозитории и установите пакеты:

sudo apt-get update

sudo apt-get install bellsoft-java12

Доступны следующие пакеты:

* bellsoft-java12 – cодержит Liberica JDK, включая LibericaFX и различные JVM для платформ, которые его поддерживают;
* bellsoft-java12-lite – содержит Liberica JDK со сжатыми модулями и Server VM, без дополнительных пакетов.

Так же возможно скачать (https://libericajdk.ru/pages/java-11.0.9.1/) необходимый пакет и поставить его отдельно.

# Конфигурация безопасности

Для конфигурирования настроек безопасности ПС «Атом.Мост» предоставляет несколько различных настраиваемых параметров. Наиболее важными являются свойства под заголовком «Security properties» в файле nifi.properties. Установка свойств, необходимых для безопасной работы, приведена в таблице 2.

1. Установка свойств, необходимых для безопасной работы

|  |  |
| --- | --- |
| Свойство | Описание |
| nifi.security.keystore | Имя файла Keystore, содержащего закрытый ключ сервера |
| nifi.security.keystoreType | Тип Keystore. Должен быть либо PKCS12, либо JKS. JKS является предпочтительным типом, файлы PKCS12 загружаются библиотекой BouncyCastle |
| nifi.security.keystorePasswd | Пароль Keystore |
| nifi.security.keyPasswd | Пароль для сертификата в Keystore. Если значение не установлено, используется nifi.security.keystorePasswd |
| nifi.security.truststore | Имя файла Truststore для авторизации при подключении к DataFlow. Защищенный инстанс без Truststore отклоняет все входящие подключения |
| nifi.security.truststoreType | Тип Truststore. Должен быть либо PKCS12, либо JKS. JKS является предпочтительным типом, файлы PKCS12 загружаются библиотекой BouncyCastle |
| nifi.security.truststorePasswd | Пароль Truststore |

После настройки перечисленных свойств устанавливается доступ к пользовательскому интерфейсу через HTTPS. Это достигается путем установки свойств nifi.web.https.host и nifi.web.https.port.

Свойство nifi.web.https.host указывает, на каком хосте должен работать сервер. При доступности интерфейса HTTPS со всех сетевых интерфейсов следует использовать значение 0.0.0.0. Для того, чтобы администраторы могли настраивать приложение для работы только на определенных сетевых интерфейсах, следует указать свойства nifi.web.http.network.interface и nifi.web.https.network.interface.

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!** При включении HTTPS необходимо исключить свойство nifi.web.http.port, так как DataFlow поддерживает либо HTTP, либо HTTPS. |  |

Так же и с nifi.security.needClientAuth – веб-сервер может быть настроен на требование аутентификации на основе сертификатов у пользователей, обращающихся к интерфейсу. Для этого веб-сервер не должен поддерживать аутентификацию имени пользователя и пароля с помощью протокола LDAP или Kerberos, так как любой из этих параметров настраивает проверку подлинности клиента на основе сертификатов. Но если аутентификация по имени пользователя и паролю и анонимный доступ не настроены, то веб-сервер запрашивает аутентификацию клиента на основе сертификата.

После защиты пользовательского интерфейса следует обеспечить внутренние кластерные коммуникации и связь между сайтами. Это достигается установкой свойств nifi.remote.input.secure и nifi.cluster.protocol.is.secure в значение true.

Веб-сервер ПС «Атом.Мост» требует аутентификации клиента на основе сертификата для пользователей, осуществляющих доступ к пользовательскому интерфейсу, если не настроен альтернативный механизм аутентификации, который требует одностороннего SSL (например, LDAP, OpenId Connect и т. Д.). Включение альтернативного механизма аутентификации настроит веб-сервер для аутентификации клиента на базе сертификатов WANT. Это позволит ему поддерживать пользователей с сертификатами, а те пользователи, у которых нет сертификата, могут входить в систему по выделенным учетным данным. Подробнее см. раздел 3 документа.

После защиты пользовательского интерфейса, производится защита соединения site-to-site, а также связь внутри кластера. Это достигается установкой для свойств nifi.remote.input.secure и nifi.cluster.protocol.is.secure в значение true. Для этого всегда требуется подключение двустороннего SSL, так как узлы будут использовать свои настроенные хранилища ключей / доверенных сертификатов для аутентификации.

## Набор средств генерации TLS

Для упрощения установки файлов потоков и автоматического создания необходимых хранилищ ключей, доверительного хранилища и соответствующих файлов конфигурации можно использовать утилиту командной строки tls-toolkit, что так же обеспечит безопасность многочисленных узлов файлов потоков.

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!** Для keystores и truststores в ПС «Атом.Мост» рекомендуются JKS. Этот инструмент позволяет задавать другие типы хранилищ ключей в командной строке и игнорировать тип PKCS12 для использования в качестве доверительного хранилища, потому что данный формат имеет проблемы совместимости между реализациями BouncyCastle и Oracle |  |

Инструмент командной строки tls-toolkit имеет два основных режима работы:

* Standalone (автономный) – создает организацию сертификатов, хранилища ключей, доверительные хранилища и файлы nifi.properties в одной команде;
* Client/Server (Клиент/Сервер) – использует Certificate Authority Server, который принимает запросы на подписание сертификатов от клиентов, подписывает и отправляет обратно. И клиент, и сервер проверяют идентификацию друг друга через общий секрет.

### Standalone

Автономный режим вызывается запуском ./bin/tls-toolkit.sh standalone -h и отображает информацию об использовании с описаниями опций, которые могут быть указаны.

В автономном режиме с tls-toolkit можно использовать следующие параметры командной строки:

-a, --keyAlgorithm <arg> – алгоритм использования сгенерированных ключей (по умолчанию: RSA);

-B, --clientCertPassword <arg> – пароль сертификата клиента. Должно быть либо одно значение, либо одно для каждого DN клиента (если не задано, генерируется автоматически);

-c, --certificateAuthorityHostname <arg> – имя хоста NiFi Certificate Authority (по умолчанию: localhost);

-C, --clientCertDn <arg> – создание сертификата клиента, подходящего для использования в браузере, с указанным DN (может быть указан несколько раз);

-d, --days <arg> – количество дней, в течение которых выданный сертификат является действительным (по умолчанию: 1095);

-f, --nifiPropertiesFile <arg> – базовый файл nifi.properties для обновления (если не указан, используется встроенный файл, идентичный файлу по умолчанию при установке NiFi);

-g, --differentKeyAndKeystorePasswords – использование другого сгенерированного пароля для ключа и хранилища ключей;

-G, --globalPortSequence <arg> – использование последовательных портов, которые вычисляются для всех хостов в соответствии с предоставленными выражениями имен хостов (могут быть указаны несколько раз, но должны быть одинаковыми от запуска до запуска);

-h, --help – печать справки и выход;

-k, --keySize <arg> – количество бит для генерации ключей (по умолчанию: 2048);

-K, --keyPassword <arg> – пароль ключа. Либо одно значение, либо одинаковое для каждого хоста (если не задано, генерируется автоматически);

-n, --hostnames <arg> – список имен хостов через запятую;

--nifiDnPrefix <arg> – строка для добавления имени хоста (в начало) при определении DN (по умолчанию: CN=);

--nifiDnSuffix <arg> – строка для добавления имени хоста (в конец) при определении DN (по умолчанию: OU=NIFI);

-o, --outputDirectory <arg> – каталог для вывода keystore, truststore и config-файлов (по умолчанию: ../bin);

-O, --isOverwrite – перезапись существующего вывода хоста;

-P, --trustStorePassword <arg> – пароль truststore. Либо одно значение, либо одинаковое для каждого хоста (если не задано, генерируется автоматически);

-s, --signingAlgorithm <arg> – алгоритм подписи сертификатов (по умолчанию: SHA256WITHRSA);

-S, --keyStorePassword <arg> – пароль keytstore. Либо одно значение, либо одинаковое для каждого хоста (если не задано, генерируется автоматически);

--subjectAlternativeNames <arg> – разделенный запятыми список доменов для использования в качестве альтернативных имен в сертификате;

-T, --keyStoreType <arg> – тип создаваемого хранилища ключей (по умолчанию: jks).

Шаблоны имен хостов:

* для указания диапазона имен хостов используются квадратные скобки, например: [01-20];
* круглые скобки используются для определения, что на хосте (хостах) работает больше, чем один инстанс DataFlow, например: (5).

### Client/Server

Режим Клиент/Сервер опирается на Центр сертификации (Certificate Authority, СЦ) для выдачи сертификатов. Центр можно остановить, если узлы не подключены к сети.

1. Server

Сервер CA вызывается запуском ./bin/tls-toolkit.sh -h, который печатает информацию об использовании с описаниями опций, которые могут быть заданы.

В режиме сервера с tls-toolkit можно использовать следующие параметры командной строки:

-a, --keyAlgorithm <arg> – алгоритм использования сгенерированных ключей (по умолчанию: RSA);

--configJsonIn <arg> – место для чтения информации о конфигурации, подразумевает useConfigJson, если установлено (по умолчанию: значение configJson);

-d, --days <arg> – количество дней, в течение которых выданный сертификат является действительным (по умолчанию: 1095);

-D, --dn <arg> – DN для сертификата CA (по умолчанию: CN=YOUR\_CA\_HOSTNAME,OU=NIFI);

-f, --configJson <arg> – место записи информации о конфигурации (по умолчанию: config.json);

-F, --useConfigJson – флаг, указывающий, что вся конфигурация считывается из configJson (для облегчения автоматического использования, иначе в configJson производится только запись);

-g, --differentKeyAndKeystorePasswords – использование другого сгенерированного пароля для ключа и хранилища ключей;

-h, --help – печать справки и выход;

-k, --keySize <arg> – количество бит для генерации ключей (по умолчанию: 2048);

-p, --PORT <arg> – порт для прослушивания центром сертификации (по умолчанию: 8443);

-s, --signingAlgorithm <arg> – алгоритм подписи сертификатов (по умолчанию: SHA256WITHRSA);

-T, --keyStoreType <arg> – тип создаваемого хранилища ключей (по умолчанию: jks);

-t, --token <arg> – маркер для предотвращения MITM (должен быть таким же, как тот, что используется клиентами).

1. Client

Клиент может использоваться для запроса новых сертификатов из центра сертификации. Утилита клиента генерирует пару ключей и запрос подписи сертификата (CSR, Certificate Signing Request), после чего отправляет CSR в центр сертификации. Клиент вызывается запуском ./bin/tls-toolkit.sh client -h, который печатает информацию об использовании с описаниями опций, которые могут быть заданы.

В режиме клиента с tls-toolkit можно использовать следующие параметры командной строки:

-a, --keyAlgorithm <arg> – алгоритм использования сгенерированных ключей (по умолчанию: RSA);

-c, --certificateAuthorityHostname <arg> – имя хоста NiFi Certificate Authority (по умолчанию: localhost);

-C, --certificateDirectory <arg> – каталог записи сертификата CA (по умолчанию: .);

--configJsonIn <arg> – место для чтения информации о конфигурации, подразумевает useConfigJson, если установлено (по умолчанию: значение configJson);

-D, --dn <arg> – DN для сертификата клиента (по умолчанию: CN=<localhost name>, OU=NIFI, заполняется автоматически инструментом);

-f, --configJson <arg> – место записи информации о конфигурации (по умолчанию: config.json);

-F, --useConfigJson – флаг, указывающий, что вся конфигурация считывается из configJson (для облегчения автоматического использования, иначе в configJson производится только запись);

-g, --differentKeyAndKeystorePasswords – использование другого сгенерированного пароля для ключа и хранилища ключей;

-h, --help – печать справки и выход;

-k, --keySize <arg> – количество бит для генерации ключей (по умолчанию: 2048);

-p, --PORT <arg> – порт для прослушивания центром сертификации (по умолчанию: 8443);

--subjectAlternativeNames <arg> – разделенный запятыми список доменов для использования в качестве альтернативных имен в сертификате;

-T, --keyStoreType <arg> – тип создаваемого хранилища ключей (по умолчанию: jks);

-t, --token <arg> – маркер для предотвращения MITM (должен быть таким же, как тот, что используется клиентами).

В результате запуска клиента предоставляется сертификат СЦ, keystore, truststore и config.json с информацией о них, а также их пароли.

Сертификат клиента можно легко импортировать в браузер, указав: -T PKCS12.

## Использование существующего центра сертификации

В некоторых случаях может потребоваться метод использования уже предоставленных сертификатов, которые уже были подписан сертификационным центром (СЦ) организации. Этот «промежуточный» СЦ можно использовать для подписания сертификатов узлов, которые будут установлены на каждом узле ПС «Атом.Мост», или сертификатов клиентов, используемых для идентификации пользователей. Чтобы внедрить существующий сертификат подписи в процессы ПС «Атом.Мост», выполните следующие действия:

1. Сгенерируйте или получите подписанные промежуточные ключи СЦ в следующем формате (см. дополнительные команды ниже):

* публичный сертификат в формате PEM: nifi-cert.pem
* закрытый ключ в формате PEM: nifi-key.key

1. Поместите файлы в рабочий каталог toolkit ПС «Атом.Мост». Это каталог, в котором инструмент настроен для вывода подписанных сертификатов. Это не обязательно каталог, в котором находится или запускается бинарный файл.

Например, в следующем сценарии команду toolkit можно запускать из ее нахождения, если выходной каталог -o равен ../hardcoded/, и будут использоваться существующие nifi-cert.pem и nifi-key.key.

Если аргумент -o не указан, рабочий каталог по умолчанию (.) должен содержать nifi-cert.pem и nifi-key.key.

Пример структуры каталогов \* до \* выполнения команд выше:

🔓 0s @ 18:07:58 $ tree -L 2

.

├── hardcoded

│ ├── CN=myusername.hardcoded\_OU=NiFi.p12

│ ├── CN=myusername.hardcoded\_OU=NiFi.password

│ ├── nifi-cert.pem

│ ├── nifi-key.key

│ ├── node1.nifi.apache.org

│ ├── node2.nifi.apache.org

│ └── node3.nifi.apache.org

└── toolkit

├── LICENSE

├── NOTICE

├── README

├── bin

├── conf

├── docs

└── lib

Файлы nifi-cert.pem и nifi-key.key должны иметь кодировку ASCII (кодировка Base64 ASCII), содержащие открытый сертификат СЦ и закрытый ключ соответственно. Ниже приведены примеры файлов каждого из них, чтобы показать ожидаемый формат.

Файл nifi-cert.pem приведен ниже:

# The first command shows the actual content of the encoded file, and the second parses it and shows the internal values

.../certs $ more nifi-cert.pem

-----BEGIN CERTIFICATE-----

MIIDZTCCAk2gAwIBAgIKAWTeM3kDAAAAADANBgkqhkiG9w0BAQsFADAxMQ0wCwYD

VQQLDAROSUZJMSAwHgYDVQQDDBduaWZpLWNhLm5pZmkuYXBhY2hlLm9yZzAeFw0x

ODA3MjgwMDA0MzJaFw0yMTA3MjcwMDA0MzJaMDExDTALBgNVBAsMBE5JRkkxIDAe

BgNVBAMMF25pZmktY2EubmlmaS5hcGFjaGUub3JnMIIBIjANBgkqhkiG9w0BAQEF

AAOCAQ8AMIIBCgKCAQEAqkVrrC+AkFbjnCpupSy84tTFDsRVUIWYj/k2pVwC145M

3bpr0pRCzLuzovAjFCmT5L+isTvNjhionsqif07Ebd/M2psYE/Rih2MULsX6KgRe

1nRUiBeKF08hlmSBMGDFPj39yDzE/V9edxV/KGjRqVgw/Qy0vwaS5uWdXnLDhzoV

4/Mz7lGmYoMasZ1uexlH93jjBl1+EFL2Xoa06oLbEojJ9TKaWhpG8ietEedf7WM0

zqBEz2kHo9ddFk9yxiCkT4SUKnDWkhwc/o6us1vEXoSw+tmufHY/A3gVihjWPIGz

qyLFl9JuN7CyJepkVVqTdskBG7S85G/kBlizUj5jOwIDAQABo38wfTAOBgNVHQ8B

Af8EBAMCAf4wDAYDVR0TBAUwAwEB/zAdBgNVHQ4EFgQUKiWBKbMMQ1zUabD4gI7L

VOWOcy0wHwYDVR0jBBgwFoAUKiWBKbMMQ1zUabD4gI7LVOWOcy0wHQYDVR0lBBYw

FAYIKwYBBQUHAwIGCCsGAQUFBwMBMA0GCSqGSIb3DQEBCwUAA4IBAQAxfHFIZLOw

mwIqnSI/ir8f/uzDMq06APHGdhdeIKV0HR74BtK95KFg42zeXxAEFeic98PC/FPV

tKpm2WUa1slMB+oP27cRx5Znr2+pktaqnM7f2JgMeJ8bduNH3RUkr9jwgkcJRwyC

I4fwHC9k18aizNdOf2q2UgQXxNXaLYPe17deuNVwwrflMgeFfVrwbT2uPJTMRi1D

FQyc6haF4vsOSSRzE6OyDoc+/1PpyPW75OeSXeVCbc3AEAvRuTZMBQvBQUqVM51e

MDG+K3rCeieSBPOnGNrEC/PiA/CvaMXBEog+xPAw1SgYfuCz4rlM3BdRa54z3+oO

lc8xbzd7w8Q3

-----END CERTIFICATE-----

.../certs $ openssl x509 -in nifi-cert.pem -text -noout

Certificate:

Data:

Version: 3 (0x2)

Serial Number:

01:64:de:33:79:03:00:00:00:00

Signature Algorithm: sha256WithRSAEncryption

Issuer: OU=NIFI, CN=nifi-ca.nifi.apache.org

Validity

Not Before: Jul 28 00:04:32 2018 GMT

Not After : Jul 27 00:04:32 2021 GMT

Subject: OU=NIFI, CN=nifi-ca.nifi.apache.org

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

Public-Key: (2048 bit)

Modulus:

00:aa:45:6b:ac:2f:80:90:56:e3:9c:2a:6e:a5:2c:

bc:e2:d4:c5:0e:c4:55:50:85:98:8f:f9:36:a5:5c:

02:d7:8e:4c:dd:ba:6b:d2:94:42:cc:bb:b3:a2:f0:

23:14:29:93:e4:bf:a2:b1:3b:cd:8e:18:a8:9e:ca:

a2:7f:4e:c4:6d:df:cc:da:9b:18:13:f4:62:87:63:

14:2e:c5:fa:2a:04:5e:d6:74:54:88:17:8a:17:4f:

21:96:64:81:30:60:c5:3e:3d:fd:c8:3c:c4:fd:5f:

5e:77:15:7f:28:68:d1:a9:58:30:fd:0c:b4:bf:06:

92:e6:e5:9d:5e:72:c3:87:3a:15:e3:f3:33:ee:51:

a6:62:83:1a:b1:9d:6e:7b:19:47:f7:78:e3:06:5d:

7e:10:52:f6:5e:86:b4:ea:82:db:12:88:c9:f5:32:

9a:5a:1a:46:f2:27:ad:11:e7:5f:ed:63:34:ce:a0:

44:cf:69:07:a3:d7:5d:16:4f:72:c6:20:a4:4f:84:

94:2a:70:d6:92:1c:1c:fe:8e:ae:b3:5b:c4:5e:84:

b0:fa:d9:ae:7c:76:3f:03:78:15:8a:18:d6:3c:81:

b3:ab:22:c5:97:d2:6e:37:b0:b2:25:ea:64:55:5a:

93:76:c9:01:1b:b4:bc:e4:6f:e4:06:58:b3:52:3e:

63:3b

Exponent: 65537 (0x10001)

X509v3 extensions:

X509v3 Key Usage: critical

Digital Signature, Non Repudiation, Key Encipherment, Data Encipherment, Key Agreement, Certificate Sign, CRL Sign

X509v3 Basic Constraints:

CA:TRUE

X509v3 Subject Key Identifier:

2A:25:81:29:B3:0C:43:5C:D4:69:B0:F8:80:8E:CB:54:E5:8E:73:2D

X509v3 Authority Key Identifier:

keyid:2A:25:81:29:B3:0C:43:5C:D4:69:B0:F8:80:8E:CB:54:E5:8E:73:2D

X509v3 Extended Key Usage:

TLS Web Client Authentication, TLS Web Server Authentication

Signature Algorithm: sha256WithRSAEncryption

31:7c:71:48:64:b3:b0:9b:02:2a:9d:22:3f:8a:bf:1f:fe:ec:

c3:32:ad:3a:00:f1:c6:76:17:5e:20:a5:74:1d:1e:f8:06:d2:

bd:e4:a1:60:e3:6c:de:5f:10:04:15:e8:9c:f7:c3:c2:fc:53:

d5:b4:aa:66:d9:65:1a:d6:c9:4c:07:ea:0f:db:b7:11:c7:96:

67:af:6f:a9:92:d6:aa:9c:ce:df:d8:98:0c:78:9f:1b:76:e3:

47:dd:15:24:af:d8:f0:82:47:09:47:0c:82:23:87:f0:1c:2f:

64:d7:c6:a2:cc:d7:4e:7f:6a:b6:52:04:17:c4:d5:da:2d:83:

de:d7:b7:5e:b8:d5:70:c2:b7:e5:32:07:85:7d:5a:f0:6d:3d:

ae:3c:94:cc:46:2d:43:15:0c:9c:ea:16:85:e2:fb:0e:49:24:

73:13:a3:b2:0e:87:3e:ff:53:e9:c8:f5:bb:e4:e7:92:5d:e5:

42:6d:cd:c0:10:0b:d1:b9:36:4c:05:0b:c1:41:4a:95:33:9d:

5e:30:31:be:2b:7a:c2:7a:27:92:04:f3:a7:18:da:c4:0b:f3:

e2:03:f0:af:68:c5:c1:12:88:3e:c4:f0:30:d5:28:18:7e:e0:

b3:e2:b9:4c:dc:17:51:6b:9e:33:df:ea:0e:95:cf:31:6f:37:

7b:c3:c4:37

Файл nifi-key.key приведен ниже:

# The first command shows the actual content of the encoded file, and the second parses it and shows the internal values

.../certs $ more nifi-key.key

-----BEGIN RSA PRIVATE KEY-----

MIIEpAIBAAKCAQEAqkVrrC+AkFbjnCpupSy84tTFDsRVUIWYj/k2pVwC145M3bpr

0pRCzLuzovAjFCmT5L+isTvNjhionsqif07Ebd/M2psYE/Rih2MULsX6KgRe1nRU

iBeKF08hlmSBMGDFPj39yDzE/V9edxV/KGjRqVgw/Qy0vwaS5uWdXnLDhzoV4/Mz

7lGmYoMasZ1uexlH93jjBl1+EFL2Xoa06oLbEojJ9TKaWhpG8ietEedf7WM0zqBE

z2kHo9ddFk9yxiCkT4SUKnDWkhwc/o6us1vEXoSw+tmufHY/A3gVihjWPIGzqyLF

l9JuN7CyJepkVVqTdskBG7S85G/kBlizUj5jOwIDAQABAoIBAAdWRnV89oVBuT0Z

dvsXGmyLzpH8U9DMcO6DRp+Jf3XaY+WKCutgCCDaVbtHrbtIr17EAzav5QOifGGb

SbVCp6Q0aJdi5360oSpEUrJRRZ5Z4dxL1vimSwUGG+RnIEn9YYJ1GWJve+2PFnr7

KieLnL03V6UPzxoMJnhcnJNdTp+dBwzSazVQwye2csSJlVMk49t2lxBwce7ohuh+

9fL7G3HU5S9d08QT1brknMHahcw1SYyJd0KSjRJCB6wAxnAZmJYJ1jQCI8YICq0j

RX2rhxEXuEMXQcaiFQXzCrmQEXreKUISDvNeu/h7YU9UvJWPZSFGnEGgnMP2XvQm

EjK3rQECgYEA5+OkpLsiLNMHGzj72PiBkq82sTLQJ2+8udYp6PheOGkhjjXoBse5

YynyHlQt6CnVpJQ33mQUkJ+3ils0SMFtmI3rz3udzleek1so2L2J3+CI4kt7fFCb

FFbVXv+dLNrm+tOw68J48asyad8kEnHYq9Us+/3MLDmFJYTthkgzCpECgYEAu/ml

lQaWaZAQcQ8UuVeasxMYoN8zMmzfrkxc8AfNwKxF9nc44ywo4nJr+u/UVRGYpRgM

rdll5vz0Iq68qk03spaW7vDJn8hJQhkReQw1it9Fp/51r9MHzGTVarORJGa2oZ0g

iNe8LNizD3bQ19hEvju9mn0x9Q62Q7dapVpffwsCgYEAtC1TPpQQ59dIjERom5vr

wffWfTTIO/w8HgFkKxrgyuAVLJSCJtKFH6H1+M7bpKrsz6ZDCs+kkwMm76ASLf3t

lD2h3mNkqHG4SzLnuBD90jB666pO1rci6FjYDap7i+DC3F4j9+vxYYXt9Aln09UV

z94hx+LaA/rlk9OHY3EyB6ECgYBA/cCtNNjeaKv2mxM8PbjD/289d85YueHgfpCH

gPs3iZiq7W+iw8ri+FKzMSaFvw66zgTcOtULtxulviqG6ym9umk29dOQRgxmKQqs

gnckq6uGuOjxwJHqrlZHjQw6vLSaThxIk+aAzu+iAh+U8TZbW4ZjmrOiGdMUuJlD

oGpyHwKBgQCRjfqQjRelYVtU7j6BD9BDbCfmipwaRNP0CuAGOVtS+UnJuaIhsXFQ

QGEBuOnfFijIvb7YcXRL4plRYPMvDqYRNObuI6A+1xNtr000nxa/HUfzKVeI9Tsn

9AKMWnXS8ZcfStsVf3oDFffXYRqCaWeuhpMmg9TwdXoAuwfpE5GCmw==

-----END RSA PRIVATE KEY-----

.../certs $ openssl rsa -in nifi-key.key -text -noout

Private-Key: (2048 bit)

modulus:

00:aa:45:6b:ac:2f:80:90:56:e3:9c:2a:6e:a5:2c:

bc:e2:d4:c5:0e:c4:55:50:85:98:8f:f9:36:a5:5c:

02:d7:8e:4c:dd:ba:6b:d2:94:42:cc:bb:b3:a2:f0:

23:14:29:93:e4:bf:a2:b1:3b:cd:8e:18:a8:9e:ca:

a2:7f:4e:c4:6d:df:cc:da:9b:18:13:f4:62:87:63:

14:2e:c5:fa:2a:04:5e:d6:74:54:88:17:8a:17:4f:

21:96:64:81:30:60:c5:3e:3d:fd:c8:3c:c4:fd:5f:

5e:77:15:7f:28:68:d1:a9:58:30:fd:0c:b4:bf:06:

92:e6:e5:9d:5e:72:c3:87:3a:15:e3:f3:33:ee:51:

a6:62:83:1a:b1:9d:6e:7b:19:47:f7:78:e3:06:5d:

7e:10:52:f6:5e:86:b4:ea:82:db:12:88:c9:f5:32:

9a:5a:1a:46:f2:27:ad:11:e7:5f:ed:63:34:ce:a0:

44:cf:69:07:a3:d7:5d:16:4f:72:c6:20:a4:4f:84:

94:2a:70:d6:92:1c:1c:fe:8e:ae:b3:5b:c4:5e:84:

b0:fa:d9:ae:7c:76:3f:03:78:15:8a:18:d6:3c:81:

b3:ab:22:c5:97:d2:6e:37:b0:b2:25:ea:64:55:5a:

93:76:c9:01:1b:b4:bc:e4:6f:e4:06:58:b3:52:3e:

63:3b

publicExponent: 65537 (0x10001)

privateExponent:

07:56:46:75:7c:f6:85:41:b9:3d:19:76:fb:17:1a:

6c:8b:ce:91:fc:53:d0:cc:70:ee:83:46:9f:89:7f:

75:da:63:e5:8a:0a:eb:60:08:20:da:55:bb:47:ad:

bb:48:af:5e:c4:03:36:af:e5:03:a2:7c:61:9b:49:

b5:42:a7:a4:34:68:97:62:e7:7e:b4:a1:2a:44:52:

b2:51:45:9e:59:e1:dc:4b:d6:f8:a6:4b:05:06:1b:

e4:67:20:49:fd:61:82:75:19:62:6f:7b:ed:8f:16:

7a:fb:2a:27:8b:9c:bd:37:57:a5:0f:cf:1a:0c:26:

78:5c:9c:93:5d:4e:9f:9d:07:0c:d2:6b:35:50:c3:

27:b6:72:c4:89:95:53:24:e3:db:76:97:10:70:71:

ee:e8:86:e8:7e:f5:f2:fb:1b:71:d4:e5:2f:5d:d3:

c4:13:d5:ba:e4:9c:c1:da:85:cc:35:49:8c:89:77:

42:92:8d:12:42:07:ac:00:c6:70:19:98:96:09:d6:

34:02:23:c6:08:0a:ad:23:45:7d:ab:87:11:17:b8:

43:17:41:c6:a2:15:05:f3:0a:b9:90:11:7a:de:29:

42:12:0e:f3:5e:bb:f8:7b:61:4f:54:bc:95:8f:65:

21:46:9c:41:a0:9c:c3:f6:5e:f4:26:12:32:b7:ad:

01

prime1:

00:e7:e3:a4:a4:bb:22:2c:d3:07:1b:38:fb:d8:f8:

81:92:af:36:b1:32:d0:27:6f:bc:b9:d6:29:e8:f8:

5e:38:69:21:8e:35:e8:06:c7:b9:63:29:f2:1e:54:

2d:e8:29:d5:a4:94:37:de:64:14:90:9f:b7:8a:5b:

34:48:c1:6d:98:8d:eb:cf:7b:9d:ce:57:9e:93:5b:

28:d8:bd:89:df:e0:88:e2:4b:7b:7c:50:9b:14:56:

d5:5e:ff:9d:2c:da:e6:fa:d3:b0:eb:c2:78:f1:ab:

32:69:df:24:12:71:d8:ab:d5:2c:fb:fd:cc:2c:39:

85:25:84:ed:86:48:33:0a:91

prime2:

00:bb:f9:a5:95:06:96:69:90:10:71:0f:14:b9:57:

9a:b3:13:18:a0:df:33:32:6c:df:ae:4c:5c:f0:07:

cd:c0:ac:45:f6:77:38:e3:2c:28:e2:72:6b:fa:ef:

d4:55:11:98:a5:18:0c:ad:d9:65:e6:fc:f4:22:ae:

bc:aa:4d:37:b2:96:96:ee:f0:c9:9f:c8:49:42:19:

11:79:0c:35:8a:df:45:a7:fe:75:af:d3:07:cc:64:

d5:6a:b3:91:24:66:b6:a1:9d:20:88:d7:bc:2c:d8:

b3:0f:76:d0:d7:d8:44:be:3b:bd:9a:7d:31:f5:0e:

b6:43:b7:5a:a5:5a:5f:7f:0b

exponent1:

00:b4:2d:53:3e:94:10:e7:d7:48:8c:44:68:9b:9b:

eb:c1:f7:d6:7d:34:c8:3b:fc:3c:1e:01:64:2b:1a:

e0:ca:e0:15:2c:94:82:26:d2:85:1f:a1:f5:f8:ce:

db:a4:aa:ec:cf:a6:43:0a:cf:a4:93:03:26:ef:a0:

12:2d:fd:ed:94:3d:a1:de:63:64:a8:71:b8:4b:32:

e7:b8:10:fd:d2:30:7a:eb:aa:4e:d6:b7:22:e8:58:

d8:0d:aa:7b:8b:e0:c2:dc:5e:23:f7:eb:f1:61:85:

ed:f4:09:67:d3:d5:15:cf:de:21:c7:e2:da:03:fa:

e5:93:d3:87:63:71:32:07:a1

exponent2:

40:fd:c0:ad:34:d8:de:68:ab:f6:9b:13:3c:3d:b8:

c3:ff:6f:3d:77:ce:58:b9:e1:e0:7e:90:87:80:fb:

37:89:98:aa:ed:6f:a2:c3:ca:e2:f8:52:b3:31:26:

85:bf:0e:ba:ce:04:dc:3a:d5:0b:b7:1b:a5:be:2a:

86:eb:29:bd:ba:69:36:f5:d3:90:46:0c:66:29:0a:

ac:82:77:24:ab:ab:86:b8:e8:f1:c0:91:ea:ae:56:

47:8d:0c:3a:bc:b4:9a:4e:1c:48:93:e6:80:ce:ef:

a2:02:1f:94:f1:36:5b:5b:86:63:9a:b3:a2:19:d3:

14:b8:99:43:a0:6a:72:1f

coefficient:

00:91:8d:fa:90:8d:17:a5:61:5b:54:ee:3e:81:0f:

d0:43:6c:27:e6:8a:9c:1a:44:d3:f4:0a:e0:06:39:

5b:52:f9:49:c9:b9:a2:21:b1:71:50:40:61:01:b8:

e9:df:16:28:c8:bd:be:d8:71:74:4b:e2:99:51:60:

f3:2f:0e:a6:11:34:e6:ee:23:a0:3e:d7:13:6d:af:

4d:34:9f:16:bf:1d:47:f3:29:57:88:f5:3b:27:f4:

02:8c:5a:75:d2:f1:97:1f:4a:db:15:7f:7a:03:15:

f7:d7:61:1a:82:69:67:ae:86:93:26:83:d4:f0:75:

7a:00:bb:07:e9:13:91:82:9b

## Использование сертификатов с внешней подписью

Чтобы подписать сгенерированные сертификаты с помощью центра сертификации, созданного вне TLS Toolkit, нужно убедиться, что необходимые файлы находятся в правильном формате и в правильном каталоге. Например, некая организация «Рога и копыта» имеет внутренний СЦ (CN = ca.large.org, OU = Certificate Authority). Этот корневой СЦ отключен и используется только для подписи других внутренних СЦ. Некая большая ИТ-команда создает промежуточный СЦ (CN = nifi\_ca.large.org, OU = NiFi, OU = Certificate Authority), который будет использоваться для подписания всех сертификатов узлов ПС «Атом.Мост» (CN = node1.nifi.large.org, OU = NiFi, CN = node2.nifi.large.org, OU = NiFi и т. Д.).

Чтобы использовать набор инструментов для создания этих сертификатов и их подписания с помощью «промежуточного» СЦ, убедитесь, что присутствуют следующие файлы (см. Дополнительные команды сертификатов):

* публичный сертификат в формате PEM: nifi-cert.pem
* закрытый ключ в формате PEM: nifi-key.key

Если бы промежуточный СЦ был корневым СЦ, он был бы самоподписанным - подпись над сертификатом была бы выпущена с тем же ключом. В этом случае (так же, как и для СЦ, созданного с помощью toolkit), никаких дополнительных аргументов не требуется. Однако, поскольку промежуточный СЦ подписан корневым СЦ, для проверки подписи также необходимо предоставить открытый сертификат корневого СЦ. Параметр --additionalCACertificate используется для указания пути к общедоступному сертификату для подписи. Значение должно быть абсолютным путем к общедоступному сертификату корневого СЦ.

Пример такого файла:

# Generate cert signed by intermediate CA (which is signed by root CA) -- WILL FAIL

$ ./bin/tls-toolkit.sh standalone -n 'node1.nifi.apache.org' \

-P passwordpassword \

-S passwordpassword \

-o /opt/certs/externalCA \

-O

2018/08/02 18:48:11 INFO [main] org.apache.nifi.toolkit.tls.standalone.TlsToolkitStandaloneCommandLine: No nifiPropertiesFile specified, using embedded one.

2018/08/02 18:48:12 INFO [main] org.apache.nifi.toolkit.tls.standalone.TlsToolkitStandalone: Running standalone certificate generation with output directory /opt/certs/externalCA

2018/08/02 18:48:12 INFO [main] org.apache.nifi.toolkit.tls.util.TlsHelper: Verifying the certificate signature for CN=nifi\_ca.large.org, OU=Certificate Authority

2018/08/02 18:48:12 INFO [main] org.apache.nifi.toolkit.tls.util.TlsHelper: Attempting to verify certificate CN=nifi\_ca.large.org, OU=NiFi, OU=Certificate Authority signature with CN=nifi\_ca.large.org, OU=NiFi, OU=Certificate Authority

2018/08/02 18:48:12 WARN [main] org.apache.nifi.toolkit.tls.util.TlsHelper: Certificate CN=nifi\_ca.large.org, OU=NiFi, OU=Certificate Authority not signed by CN=nifi\_ca.large.org, OU=NiFi, OU=Certificate Authority [certificate does not verify with supplied key]

Error generating TLS configuration. (The signing certificate was not signed by any known certificates)

# Provide additional CA certificate path for signature verification of intermediate CA

$ ./bin/tls-toolkit.sh standalone -n 'node1.nifi.apache.org' \

-P passwordpassword \

-S passwordpassword \

-o /opt/certs/externalCA \

--additionalCACertificate /opt/certs/externalCA/root.pem \

-O

2018/08/02 18:48:44 INFO [main] org.apache.nifi.toolkit.tls.standalone.TlsToolkitStandaloneCommandLine: No nifiPropertiesFile specified, using embedded one.

2018/08/02 18:48:44 INFO [main] org.apache.nifi.toolkit.tls.standalone.TlsToolkitStandalone: Running standalone certificate generation with output directory /opt/certs/externalCA

2018/08/02 18:48:44 INFO [main] org.apache.nifi.toolkit.tls.util.TlsHelper: Verifying the certificate signature for CN=nifi\_ca.large.org, OU=NiFi, OU=Certificate Authority

2018/08/02 18:48:44 INFO [main] org.apache.nifi.toolkit.tls.util.TlsHelper: Attempting to verify certificate CN=nifi\_ca.large.org, OU=NiFi, OU=Certificate Authority signature with CN=ca.large.org, OU=Certificate Authority

2018/08/02 18:48:44 INFO [main] org.apache.nifi.toolkit.tls.util.TlsHelper: Certificate was signed by CN=ca.large.org, OU=Certificate Authority

2018/08/02 18:48:44 INFO [main] org.apache.nifi.toolkit.tls.standalone.TlsToolkitStandalone: Using existing CA certificate /opt/certs/externalCA/nifi-cert.pem and key /opt/certs/externalCA/nifi-key.key

2018/08/02 18:48:44 INFO [main] org.apache.nifi.toolkit.tls.standalone.TlsToolkitStandalone: Writing new ssl configuration to /opt/certs/externalCA/node1.nifi.apache.org

2018/08/02 18:48:44 INFO [main] org.apache.nifi.toolkit.tls.standalone.TlsToolkitStandalone: Successfully generated TLS configuration for node1.nifi.apache.org 1 in /opt/certs/externalCA/node1.nifi.apache.org

2018/08/02 18:48:44 INFO [main] org.apache.nifi.toolkit.tls.standalone.TlsToolkitStandalone: No clientCertDn specified, not generating any client certificates.

2018/08/02 18:48:44 INFO [main] org.apache.nifi.toolkit.tls.standalone.TlsToolkitStandalone: tls-toolkit standalone completed successfully

## Дополнительные команды сертификации

Чтобы преобразовать открытый сертификат, закодированный в DER (cert.der), в кодированный PEM (cert.pem):

1. Если файл DER содержит как открытый сертификат, так и закрытый ключ, удалите закрытый ключ с помощью этой команды:

perl -pe 'BEGIN {undef $ /;} s | ----- BEGIN PRIVATE KEY -----. \*? ----- END PRIVATE KEY ----- | наименование.der> cert.pem

1. Если файл DER содержит только общедоступный сертификат, используйте эту команду:

openssl x509 -inform der -in cert.der -out cert.pem

Чтобы преобразовать из хранилища ключей PKCS12 (keystore.p12), содержащего как открытый сертификат, так и закрытый ключ, в файлы с кодировкой PEM ($ PASSWORD – пароль хранилища ключей):

openssl pkcs12 -in keystore.p12 -nodes -clcerts -nokeys -out cert.pem -password "pass: $ PASSWORD"

openssl pkcs12 -in keystore.p12 -nodes -nocerts -out key.key -password "pass: $ PASSWORD"

Чтобы преобразовать из хранилища ключей Java (keystore.jks), содержащего закрытый ключ, в файлы с кодировкой PEM ($ P12\_PASSWORD - это пароль хранилища ключей PKCS12, $ JKS\_PASSWORD - пароль хранилища ключей Java, который вы хотите установить, а $ ALIAS может иметь любое значение - значение по умолчанию ПС «Атом.Мост» это NiFi-ключ):

keytool -importkeystore -srckeystore keystore.jks -destkeystore keystore.p12 -srcstoretype JKS -deststoretype PKCS12 -destkeypass "$ P12\_PASSWORD" -deststorepass "$ P12\_PASSWORD" -srcstorepass "$ JKS\_PcalASSWORD" -srcstorepass "$ JKS\_PcalASSWORD" -srcstorepass "$ JKS\_PcalASSWORD" -srcstorepass "$ JKS\_PcalASSWORD" -srcstorepass "$ JKS\_PcalASSWORD" -srcstorepass "$ JKS\_PcalASSWORD" -srcstorepass "" $ JKS\_PcalASSWORD "-srcstorepass"

Выполните указанные выше действия, чтобы преобразовать keystore.p12 в cert.pem и key.key.

Чтобы преобразовать формат PEM PKCS # 8 в формат PEM PKCS # 1: если закрытый ключ предоставлен в формате PKCS # 8 (**ВНИМАНИЕ!** файл начинается с ----- BEGIN PRIVATE KEY -----, а не ----- BEGIN RSA PRIVATE KEY -----), следующая команда преобразует его в формат PKCS # 1, переместит оригинал в nifi-key-pkcs8.key и переименует версию PKCS # 1 в nifi-key.key:

openssl rsa -in nifi-key.key -out nifi-key-pkcs1.key && mv nifi-key.key nifi-key-pkcs8.key && mv nifi-key-pkcs1.key nifi-key.key

Чтобы объединить закрытый ключ в формате PEM (private.key) и открытый сертификат в формате PEM (certificate.pem) в хранилище ключей PKCS12. Нижеприведенная команда создаст хранилище ключей PKCS12 (keystore.p12) из ​​двух независимых файлов. Хранилище ключей Java (JKS) не может быть сформировано непосредственно из файлов PEM:

openssl pkcs12 -export -out хранилище ключей.p12 -inkey private.key -in certificate.pem

Чтобы преобразовать хранилище ключей PKCS12 (keystore.p12) в хранилище ключей JKS (keystore.jks): команда ниже создаст хранилище ключей JKS (keystore.jks). Флаг -destalias не является обязательным, поскольку ПС «Атом.Мост» в настоящее время не читает из определенного псевдонима в хранилище ключей. Пользователю будет предложено ввести пароль хранилища ключей, который должен быть установлен и состоять минимум из 8 символов, и пароль ключа, который может быть таким же, как пароль хранилища ключей, или другим:

keytool -importkeystore -srckeystore keystore.p12 -srcstoretype pkcs12 -destkeystore keystore.jks -deststoretype jks -destalias nifi-key

# Идентификация и аутентификация

## Аутентификация пользователя

Аутентификация пользователей осуществляется с использованием внешнего источника аутентификации. В качестве внешних источников аутентификации могут выступать Lightweight Directory Access Protocol (LDAP), Kerberos и OpenId Connect. ПС «Атом.Мост» поддерживает аутентификацию пользователей с помощью сертификатов пользователей, идентификатора пользователя (логин) и пароля, а также через OpenId Connect.

Аутентификация по идентификатору пользователя / паролю выполняется через «идентификатор входа в систему». Идентификатор входа в систему – это подключаемый механизм для аутентификации пользователей с помощью их идентификаторов и пароля. Этот механизм и идентификационные данные для входа настраиваются в файле nifi.properties. ПС «Атом.Мост» предлагает проверку идентификатора и пароля с параметрами Login Identity Provider для Lightweight Directory Access Protocol (облегченный протокол доступа к каталогам, LDAP) и Kerberos.

В файле nifi.login.identity.provider.configuration.file указывается конфигурация Login Identity Provider. По умолчанию эти свойства установлены в ./conf/login-identity-providers.xml.

Свойство nifi.security.user.login.identity.provider указывает, какой из настроенных Login Identity Provider следует использовать. По умолчанию это свойство не настроено, что означает, что имя пользователя / пароль должны быть явно указаны.

Во время аутентификации OpenId Connect ПС «Атом.Мост» будет перенаправлять пользователей для входа в систему с логином и паролем, прежде чем вернуться в ПС «Атом.Мост». Затем ПС «Атом.Мост» обратится к провайдеру, чтобы идентифицировать пользователя.

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!** ПС «Атом.Мост» можно настроить только для идентификации по идентификатору пользователя / паролю или OpenId Connect. Она не поддерживает одновременный запуск каждого из них. ПС «Атом.Мост» потребует клиентские сертификаты для аутентификации пользователей по HTTPS, если ни один из них не настроен. |  |

### Lightweight Directory Access Protocol

Ниже приведены пример и описание настройки Login Identity Provider для входа в систему, который интегрируется с Directory Server для аутентификации пользователей.

Добавьте следующее в nifi.properties, чтобы включить аутентификацию по имени пользователя и паролю LDAP:

<provider>

<identifier>ldap-provider</identifier>

<class>org.apache.nifi.ldap.LdapProvider</class>

<property name="Authentication Strategy">START\_TLS</property>

<property name="Manager DN"></property>

<property name="Manager Password"></property>

<property name="TLS - Keystore"></property>

<property name="TLS - Keystore Password"></property>

<property name="TLS - Keystore Type"></property>

<property name="TLS - Truststore"></property>

<property name="TLS - Truststore Password"></property>

<property name="TLS - Truststore Type"></property>

<property name="TLS - Client Auth"></property>

<property name="TLS - Protocol"></property>

<property name="TLS - Shutdown Gracefully"></property>

<property name="Referral Strategy">FOLLOW</property>

<property name="Connect Timeout">10 secs</property>

<property name="Read Timeout">10 secs</property>

<property name="Url"></property>

<property name="User Search Base"></property>

<property name="User Search Filter"></property>

<property name="Identity Strategy">USE\_DN</property>

<property name="Authentication Expiration">12 hours</property>

</provider>

С помощью данной конфигурации аутентификация имени пользователя и пароля может быть активирована путем ссылки на провайдер в nifi.properties:

nifi.security.user.login.identity.provider=ldap-provider

Описание настроек Login Identity Provider для LDAP приведено в таблице 3.

1. Описание настроек Login Identity Provider для LDAP

|  |  |
| --- | --- |
| Свойство | Описание |
| Authentication Strategy | Аутентификация подключения к LDAP-серверу. Возможные значения: ANONYMOUS, SIMPLE, LDAPS или START\_TLS |
| Manager DN | DN менеджера, который используется для привязки к LDAP-серверу для поиска пользователей |
| Manager Password | Пароль менеджера, который используется для привязки к LDAP-серверу для поиска пользователей |
| TLS - Keystore | Путь к Keystore при подключении к LDAP с использованием LDAPS или START\_TLS |
| TLS - Keystore Password | Пароль для Keystore при подключении к LDAP с использованием LDAPS или START\_TLS |
| TLS - Keystore Type | Тип Keystore при подключении к LDAP с использованием LDAPS или START\_TLS (то есть JKS или PKCS12) |
| TLS - Truststore | Путь к Truststore при подключении к LDAP с использованием LDAPS или START\_TLS |
| TLS - Truststore Password | Пароль для Truststore при подключении к LDAP с использованием LDAPS или START\_TLS |
| TLS - Truststore Type | Тип Truststore при подключении к LDAP с использованием LDAPS или START\_TLS (то есть JKS или PKCS12) |
| TLS - Client Auth | Политика аутентификации клиента при подключении к LDAP с использованием LDAPS или START\_TLS. Возможные значения: REQUIRED, WANT, NONE |
| TLS - Protocol | Протокол при подключении к LDAP с использованием LDAPS или START\_TLS (TLS, TLSv1.1, TLSv1.2 и т.д.) |
| TLS - Shutdown Gracefully | Указывает, следует ли корректно завершать работу TLS перед закрытием целевого контекста. По умолчанию: false |
| Referral Strategy | Стратегия обработки рефералов. Возможные значения: FOLLOW, IGNORE, THROW |
| Connect Timeout | Время ожидания соединения (10 секунд) |
| Read Timeout | Время ожидания чтения (10 секунд) |
| Url | Разделенный пробелами список URL-адресов серверов LDAP (например, [ldap:/](https://notes.rosatom.local/ldap:/)/<hostname>:<port>) |
| User Search Base | Базовый DN для поиска пользователей (например, CN=Users,DC=example,DC=com) |
| User Search Filter | Фильтр для поиска пользователей в User Search Base (sAMAccountName={0}). Указанное пользователем имя вставляется в {0} |
| Identity Strategy | Стратегия идентификации пользователей. Возможные значения: USE\_DN и USE\_USERNAME. По умолчанию: - USE\_DN (для сохранения обратной совместимости). USE\_DN использует полный DN пользовательской записи (рекомендуется). USE\_USERNAME использует имя пользователя, под которым он вошел в систему |
| Authentication Expiration | Продолжительность действия проверки подлинности пользователя. Если пользователь никогда не выходит из системы, он должен будет снова войти в систему в течение указанного времени |
| Authentication Strategy | Аутентификация подключения к LDAP-серверу. Возможные значения: ANONYMOUS, SIMPLE, LDAPS или START\_TLS |
| Manager DN | DN менеджера, который используется для привязки к LDAP-серверу для поиска пользователей |
| Manager Password | Пароль менеджера, который используется для привязки к LDAP-серверу для поиска пользователей |
| TLS - Keystore | Путь к Keystore при подключении к LDAP с использованием LDAPS или START\_TLS |
| TLS - Keystore Password | Пароль для Keystore при подключении к LDAP с использованием LDAPS или START\_TLS |
| TLS - Keystore Type | Тип Keystore при подключении к LDAP с использованием LDAPS или START\_TLS (то есть JKS или PKCS12) |
| TLS - Truststore | Путь к Truststore при подключении к LDAP с использованием LDAPS или START\_TLS |
| TLS - Truststore Password | Пароль для Truststore при подключении к LDAP с использованием LDAPS или START\_TLS |
| TLS - Truststore Type | Тип Truststore при подключении к LDAP с использованием LDAPS или START\_TLS (то есть JKS или PKCS12) |

### Kerberos

Далее приведен пример с описанием настроек Login Identity Provider, который интегрируется с Kerberos Key Distribution Center (KDC) для аутентификации пользователей.

<provider>

<identifier>kerberos-provider</identifier>

<class>org.apache.nifi.kerberos.KerberosProvider</class>

<property name="Default Realm">NIFI.APACHE.ORG</property>

<property name="Kerberos Config File">/etc/krb5.conf</property>

<property name="Authentication Expiration">12 hours</property>

</provider>

С помощью данной конфигурации аутентификация имени пользователя и пароля может быть активирована путем ссылки на провайдер в nifi.properties:

nifi.security.user.login.identity.provider=kerberos-provider

Описание настроек Login Identity Provider для Kerberos приведено в таблице 4.

1. Описание настроек Login Identity Provider для Kerberos

|  |  |
| --- | --- |
| Свойство | Описание |
| Default Realm | Область по умолчанию для предоставления пользователю в случае, если пользователь вводит неполный пользовательский принципал (например, NIFI.APACHE.ORG) |
| Kerberos Config File | Абсолютный путь к файлу конфигурации клиента Kerberos |
| Authentication Expiration | Продолжительность действия проверки подлинности пользователя. Если пользователь никогда не выходит из системы, он должен будет снова войти в систему в течение указанного времени |

### Аутентификация через OpenId Connect

При аутентификации через OpenId Connect сервер ПС «Атом.Мост» перенаправляет пользователей для проверки подлинности в провайдер, а затем ПС «Атом.Мост» вызывает провайдер для получения идентификации пользователя.

Для включения аутентификации через OpenId Connect необходимо настроить свойства в nifi.properties, представленные далее в таблице 5.

1. Описание настроек для аутентификации через OpenId Connect

|  |  |
| --- | --- |
| Свойство | Описание |
| nifi.security.user.oidc.discovery.url | URL-адрес обнаружения необходимого OpenId Connect Provider (<http://openid.net/specs/openid-connect-discovery-1_0.html>) |
| nifi.security.user.oidc.connect.timeout | Время ожидания соединения при обмене данными с OpenId Connect Provider |
| nifi.security.user.oidc.read.timeout | Время ожидания чтения при обмене данными с OpenId Connect Provider |
| nifi.security.user.oidc.client.id | Идентификатор клиента для DataFlow после регистрации в OpenId Connect Provider |
| nifi.security.user.oidc.client.secret | Секрет клиента для DataFlow после регистрации в OpenId Connect Provider |
| nifi.security.user.oidc.preferred.jwsalgorithm | Предпочтительный алгоритм проверки токенов идентификации. Если значение свойства пустое, по умолчанию используется RS 256, поддерживаемое OpenId Connect Provider в соответствии со спецификацией. Если значение равно HS256, HS384 или HS512, NiFi пытается проверить защищенные токены HMAC, используя указанный секретный ключ. Если значение свойства равно none, NiFi пытается проверить незащищенные/простые токены. Иные значения для алгоритма анализируются как алгоритм RSA или EC, который используется совместно с JSON Web Key (JWK), предоставленным через jwks\_uri в метаданных URL-обнаружения |
| nifi.security.user.oidc.discovery.url | URL-адрес обнаружения необходимого OpenId Connect Provider (<http://openid.net/specs/openid-connect-discovery-1_0.html>) |

## Мультитентантная авторизация

Мультитентантная авторизация позволяет нескольким группам пользователей (арендаторов) управлять, контролировать и наблюдать за различными частями потока данных с различными уровнями авторизации. Когда аутентифицированный пользователь пытается просмотреть или изменить ресурс ПС «Атом.Мост», система проверяет, есть ли у пользователя права на выполнение этого действия. Разрешения определяются политикой безопасности, которые можно применять в масштабах всей системы или к отдельным компонентам.

Мультитентантная авторизация настраивается с помощью двух свойств в файле nifi.properties и предоставляет администраторам право управления пользователями и политикой безопасности, создавая предварительные авторизации при запуске.

Свойство nifi.authorizer.configuration.file указывает на файл конфигурации, в котором определена авторизация. По умолчанию выбран файл authorizers.xml, расположенный в корневом каталоге conf установки.

Свойство nifi.security.user.authorizer указывает, какая из настроенных авторизаций в файле authorizers.xml будет использоваться.

## Настройка Authorizers.xml

Файл authorizers.xml используется для определения и настройки доступных авторизаторов. Авторизатором по умолчанию является StandardManagedAuthorizer. Управляемый авторизатор состоит из UserGroupProvider и AccessPolicyProvider. Пользователи, группы и политики доступа будут загружены и при необходимости настроены через этих поставщиков. Управляемый авторизатор будет принимать все решения о доступе на основе предоставленных пользователей, групп и политики доступа.

Во время запуска выполняется проверка, чтобы убедиться, что нет пользователей и групп с одинаковым идентификатором или именем. Эта проверка выполняется независимо от настроенной реализации, поскольку именно так пользователи и группы идентифицируются и авторизуются при принятии решений о предоставлении доступа.

### FileUserGroupProvider

По умолчанию UserGroupProvider - это FileUserGroupProvider, однако вы можете разработать дополнительные UserGroupProvider как расширения. FileUserGroupProvider имеет следующие свойства:

1. Users File – файл, в котором FileUserGroupProvider хранит пользователей и группы. По умолчанию выбран файл users.xml в каталоге conf;
2. Legacy Authorized Users File – полный путь к существующему файлу authorized-users.xml, который будет автоматически использоваться для загрузки пользователей и групп в файл пользователей;
3. Initial User Identity - идентификация пользователей и систем для заполнения файла пользователей. Имя каждого свойства должно быть уникальным, например: "Initial User Identity A", "Initial User Identity B", "Initial User Identity C" или "Initial User Identity 1", "Initial User Identity 2", "Initial User Identity 3".

### LdapUserGroupProvider

Другой вариант для UserGroupProvider - это LdapUserGroupProvider. По умолчанию этот параметр закомментирован, но его можно настроить вместо FileUserGroupProvider. Данное действие синхронизирует пользователей и группы с сервера службы каталогов и представит их в пользовательском интерфейсе ПС «Атом.Мост» в формате только для чтения.

LdapUserGroupProvider имеет свойства, приведенные в таблице 6.

1. Свойства LdapUserGroupProvider

|  |  |
| --- | --- |
| Свойство | Описание |
| Authentication Strategy | Как аутентифицируется соединение с сервером LDAP. Возможные значения ANONYMOUS, SIMPLE, LDAPS, или START\_TLS. |
| Manager DN | DN менеджера, который используется для привязки к серверу LDAP для поиска пользователей. |
| Manager Password | Пароль менеджера, который используется для привязки к серверу LDAP для поиска пользователей. |
| TLS - Keystore | Путь к хранилищу ключей, который используется при подключении к LDAP с помощью LDAPS или START\_TLS. |
| TLS - Keystore Password | Пароль для хранилища ключей, который используется при подключении к LDAP с помощью LDAPS или START\_TLS. |
| TLS - Keystore Type | Тип хранилища ключей, которое используется при подключении к LDAP с помощью LDAPS или START\_TLS (т.е. JKSили PKCS12). |
| TLS - Truststore | Путь к хранилищу доверенных сертификатов, который используется при подключении к LDAP с помощью LDAPS или START\_TLS. |
| TLS - Truststore Password | Пароль для хранилища доверенных сертификатов, который используется при подключении к LDAP с помощью LDAPS или START\_TLS. |
| TLS - Truststore Type | Тип хранилища доверенных сертификатов, который используется при подключении к LDAP с помощью LDAPS или START\_TLS (т.е. JKSили PKCS12). |
| TLS - Client Auth | Политика аутентификации клиента при подключении к LDAP с помощью LDAPS или START\_TLS. Возможные значения REQUIRED, WANT, NONE. |
| TLS - Protocol | Протокол, используемый при подключении к LDAP с помощью LDAPS или START\_TLS. (то есть TLS, TLSv1.1, TLSv1.2и т.д.). |
| TLS - Shutdown Gracefully | Определяет, следует ли корректно завершить работу TLS перед закрытием целевого контекста. По умолчанию - false. |
| Referral Strategy | Стратегия работы с рефералами. Возможные значения FOLLOW, IGNORE, THROW. |
| Connect Timeout | Продолжительность тайм-аута подключения. (т.е. 10 secs). |
| Read Timeout | Продолжительность таймаута чтения. (т.е. 10 secs). |
| Url | Разделенный пробелами список URL-адресов серверов LDAP (т.е. ldap://<hostname>:<port>). |
| Page Size | Устанавливает размер страницы при получении пользователей и групп. Если не указано, подкачка не выполняется. |
| Group Membership - Enforce Case Sensitivity | Устанавливает, учитывается ли регистр при принятии решений о членстве в группе. При выводе пользователя или группы (путем не указания базы поиска пользователя или группы, атрибута идентификатора пользователя или атрибута имени группы) учитывается регистр, поскольку значение, используемое для идентификатора пользователя или имени группы, будет неоднозначным. По умолчанию - false. |
| Sync Interval | Продолжительность времени между синхронизацией пользователей и групп. (т.е. 30 mins). Минимально допустимое значение 10 secs. |
| User Search Base | Базовое DN для поиска пользователей (т.е. ou=users,o=nifi). Требуется для поиска пользователей. |
| User Object Class | Класс объекта для идентификации пользователей (т.е. person). Обязательно при поиске пользователей. |
| User Search Scope | Область поиска для поиска пользователей ( ONE\_LEVEL, OBJECTили SUBTREE). Обязательно при поиске пользователей. |
| User Search Filter | Фильтр для поиска пользователей против User Search Base(т.е. (memberof=cn=team1,ou=groups,o=nifi)). По желанию. |
| User Identity Attribute | Атрибут, используемый для извлечения идентификатора пользователя (т.е. cn). По желанию. Если не установлен, используется весь DN. |
| User Group Name Attribute | Атрибут, используемый для определения членства в группе (т.е. memberof). По желанию. Если не задано, членство в группе не будет рассчитываться через пользователей. Будет полагаться на членство в группе, определяемое, Group Member Attributeесли установлено. Значением этого свойства является имя атрибута в записи ldap пользователя, который связывает их с группой. Значением этого пользовательского атрибута может быть, например, dn или имя группы. Ожидаемое значение настраивается в файле User Group Name Attribute - Referenced Group Attribute. |
| User Group Name Attribute - Referenced Group Attribute | Если поле пусто, User Group Name Attributeожидается , что значение атрибута, определенного в , будет полным dn группы. Если не пусто, это свойство будет определять атрибут записи группы ldap, на которую User Group Name Attributeссылается значение атрибута, определенного в (т.е. name). Для использования этого свойства необходимо, чтобы Group Search Baseоно было настроено. |
| Group Search Base | Базовое DN для поиска групп (т.е. ou=groups,o=nifi). Требуется для поиска в группах. |
| Group Object Class | Класс объекта для идентификации групп (т.е. groupOfNames). Обязательно при поиске в группах. |
| Group Search Scope | Область поиска для поиска групп ( ONE\_LEVEL, OBJECTили SUBTREE). Обязательно при поиске в группах. |
| Group Search Filter | Фильтр для поиска групп по Group Search Base. По желанию. |
| Group Name Attribute | Атрибут, используемый для извлечения имени группы (т.е. cn). По желанию. Если не установлен, используется весь DN. |
| Group Member Attribute | Атрибут, используемый для определения членства в группе (т.е. member). По желанию. Если не задано, членство в группах не будет рассчитываться по группам. Будет полагаться на членство в группе, определяемое, User Group Name Attributeесли установлено. Значением этого свойства является имя атрибута в записи группы ldap, которая связывает их с пользователем. Значением этого атрибута группы может быть, например, dn или memberUid. Ожидаемое значение настраивается в файле Group Member Attribute - Referenced User Attribute. (т.е. member: cn=User 1,ou=users,o=nifivs. memberUid: user1) |

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!** Любые правила сопоставления идентификаторов, указанные в nifi.properties, также будут применяться к идентификаторам пользователей. Имена групп не отображаются. |  |

### ShellUserGroupProvider

ShellUserGroupProvider извлекает сведения о пользователях и группах из Unix-подобных систем с помощью shell-команд.

Этот провайдер представляет собой пайплайны shell с такими командами, как, например, Linux.

Поддерживаемые системы могут быть настроены для пользователей и групп из внешнего источника, такого как LDAP или NIS. В этих случаях команды оболочки возвращают этих внешних пользователей и группы. Это предоставляет для администратора еще один механизм для интеграции служб каталогов пользователей и групп.

ShellUserGroupProvider имеет свойства, приведенные в таблице 7.

1. Свойства ShellUserGroupProvider

|  |  |
| --- | --- |
| Свойство | Описание |
| Exclude Users | Регулярное выражение, используемое для исключения пользователей. По умолчанию "" – означает, что пользователи не исключены. |
| Initial Refresh Delay | Продолжительность начальной задержки перед обновлением первого пользователя и группы (т.е. 10 сек.). По умолчанию 5 мин. |
| Refresh Delay | Продолжительность задержки между обновлением каждого пользователя и группы (т.е. 10 сек.). По умолчанию 5 мин. |
| Exclude Groups | Регулярное выражение, используемое для исключения групп. По умолчанию "" – означает, что никакие группы не исключаются. |

Как и LdapUserGroupProvider, ShellUserGroupProvider закомментирован в файле authorizers.xml. См. Этот комментарий для примеров использования.

### Составные реализации

Другой вариант для UserGroupProvider – это составные реализации. Это означает, что можно настроить и составить несколько источников или реализаций. Например, администратор может настроить пользователей / группы для загрузки из файла и сервера каталогов. Существует две составные реализации: одна поддерживает несколько UserGroupProvider, а другая поддерживает несколько UserGroupProvider и один настраиваемый UserGroupProvider.

CompositeUserGroupProvider обеспечит поддержку для получения пользователей и групп из нескольких источников. CompositeUserGroupProvider имеет свойство User Group Provider [unique key] – это идентификатор поставщиков группы пользователей, из которых выполняется загрузка. Имя каждого свойства должно быть уникальным, например: "User Group Provider A", "User Group Provider B", "User Group Provider C" or "User Group Provider 1", "User Group Provider 2", "User Group Provider 3".

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!** Любые правила сопоставления идентификаторов, указанные в nifi.properties, не применяются в этой реализации. Это поведение необходимо применить в базовой реализации. |  |

CompositeConfigurableUserGroupProvider обеспечит поддержку для получения пользователей и групп из нескольких источников. Кроме того, требуется один настраиваемый поставщик группы пользователей. Пользователи из настраиваемого поставщика группы пользователей настраиваются, однако пользователей, загруженные из одного из UserGroupProvider [unique key], не будет.

CompositeConfigurableUserGroupProvider имеет свойства, приведенные в таблице 8.

1. Свойства CompositeConfigurableUserGroupProvider

|  |  |
| --- | --- |
| Свойство | Описание |
| User Group Provider [unique key] | Настраиваемый поставщик группы пользователей. |
| Configurable User Group Provider | Идентификатор поставщиков группы пользователей, из которых выполняется загрузка. Имя каждого свойства должно быть уникальным, например: "User Group Provider A", "User Group Provider B", "User Group Provider C" or "User Group Provider 1", "User Group Provider 2", "User Group Provider 3". |

### FileAccessPolicyProvider

По умолчанию AccessPolicyProvider – это FileAccessPolicyProvider, однако вы можете разработать дополнительные AccessPolicyProvider как расширения. FileAccessPolicyProvider имеет следующие свойства, приведенные в таблице 9.

1. Свойства FileAccessPolicyProvider

|  |  |
| --- | --- |
| Свойство | Описание |
| User Group Provider | Определенный выше идентификатор поставщика группы пользователей, который будет использоваться для доступа к пользователям и группам для использования в политиках управляемого доступа. |
| Authorizations File | Файл, в котором FileAccessPolicyProvider будет хранить политики. |
| Initial Admin Identity | Удостоверение первоначального пользователя-администратора, которому будет предоставлен доступ к пользовательскому интерфейсу и дана возможность создавать дополнительных пользователей, группы и политики. Значением этого свойства может быть DN при использовании сертификатов или LDAP, или принципал Kerberos. Это свойство будет использоваться только в том случае, если не определены другие политики. Если это свойство указано, то файл авторизованных пользователей прежних версий не может быть указан. |
| Legacy Authorized Users File | Полный путь к существующему авторизованному users.xml, который будет автоматически преобразован в новую модель авторизации. Если это свойство указано, то исходное удостоверение администратора не может быть указано, и это свойство будет использоваться только тогда, когда не определены другие пользователи, группы и политики. |
| Node Identity | Идентификатор узла кластера ПС «Атом.Мост». При кластеризации необходимо определить свойство для каждого узла, чтобы каждый узел знал о каждом другом узле. Если они не сгруппированы, эти свойства можно игнорировать. Имя каждого свойства должно быть уникальным, например, для кластера из трех узлов: "Node Identity A", "Node Identity B", "Node Identity C" or "Node Identity 1", "Node Identity 2", "Node Identity 3" |
| Node Group | Название группы, содержащей узлы кластера ПС «Атом.Мост». Обычно это используется, когда узлы динамически добавляются / удаляются из кластера. |

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!**  Удостоверения, настроенные в исходном идентификаторе администратора, свойствах идентификатора узла или обнаруженные в устаревшем файле авторизованных пользователей, должны быть доступны в настроенном поставщике группы пользователей.  Все пользователи в устаревшем файле пользователей должны быть найдены в настроенном Поставщике группы пользователей.  Любые правила сопоставления идентификаторов, указанные в nifi.properties, также будут применяться к идентификаторам узлов, поэтому значения должны быть неотображенными идентификаторами (то есть полным DN из сертификата). Этот идентификатор должен быть найден у настроенного поставщика группы пользователей. |  |

### StandardManagedAuthorizer

Авторизатором по умолчанию является StandardManagedAuthorizer, однако вы можете разработать дополнительные авторизаторы как расширения. StandardManagedAuthorizer имеет свойство Access Policy Provider – это идентификатор поставщика политики доступа.

### FileAuthorizer

FileAuthorizer был заменен более детальным подходом StandardManagedAuthorizer, описанным выше. Однако он по-прежнему доступен по причинам обратной совместимости. FileAuthorizer имеет свойства, представленные в таблице 10.

1. Свойства FileAuthorizer

|  |  |
| --- | --- |
| Свойство | Описание |
| Node Identity | Полный путь к существующему файлу authorized-users.xml, который автоматически преобразуется в многопользовательскую модель авторизации. Это свойство используется только тогда, когда не определены другие пользователи, группы и политики. |
| Authorizations File | Идентификатор узла кластера ПС «Атом.Мост». При кластеризации необходимо определить свойство для каждого узла, чтобы каждый узел знал о каждом другом узле. Если они не сгруппированы, эти свойства можно игнорировать. |
| Users File | Файл, в котором FileAuthorizer хранит политики. По умолчанию в каталоге выбран файл authorizations.xmlconf . |
| Initial Admin Identity | Файл, в котором FileAuthorizer хранит пользователей и группы. По умолчанию в каталоге выбран файл users.xmlconf . |
| Legacy Authorized Users File | Идентификатор первоначального пользователя-администратора, которому предоставлен доступ к пользовательскому интерфейсу и дана возможность создавать дополнительных пользователей, группы и политики. Это свойство используется только тогда, когда не определены другие пользователи, группы и политики. |

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!**  Любые правила сопоставления идентификаторов, указанные в nifi.properties, также будут применены к исходному идентификатору администратора, поэтому значение должно быть несопоставленным идентификатором.  Любые правила сопоставления идентификаторов, указанные в nifi.properties, также будут применяться к идентификаторам узлов, поэтому значения должны быть неотображенными идентификаторами (то есть полным DN из сертификата). |  |

### Первоначальная идентификация администратора

Если вы настраиваете защищенный экземпляр ПС «Атом.Мост» в первый раз, вы должны вручную указать “Initial Admin Identity” в файле authorizers.xml. Этому первоначальному администратору предоставляется доступ к пользовательскому интерфейсу и возможность создавать дополнительных пользователей, группы и политики. Значением этого свойства может быть DN (при использовании сертификатов или LDAP) или принципал Kerberos. Если вы являетесь администратором ПС «Атом.Мост», укажите себя как “Initial Admin Identity”.

После того, как вы отредактировали и сохранили файл authorizers.xml, перезапустите ПС «Атом.Мост». Во время перезапуска в файлы users.xml и authorizations.xml добавляются пользовательские и административные политики “Initial Admin Identity”. После запуска ПС «Атом.Мост» пользователь «Первоначальный идентификатор администратора» может получить доступ к пользовательскому интерфейсу и начать управлять пользователями, группами и политиками.

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!** Для нового безопасного потока предоставление «Initial Admin Identity» дает пользователю доступ к пользовательскому интерфейсу и управлению пользователями, группами и политиками. Но если этот пользователь хочет начать изменять поток, он должен предоставить себе политики для корневой группы процессов. Система не может сделать это автоматически, поскольку в новом потоке UUID корневой группы процессов не является постоянным до тех пор, пока не будет сгенерирован flow.xml.gz. Если экземпляр ПС «Атом.Мост» является обновлением существующего flow.xml.gz или экземпляра 1.x, переходящего от незащищенного к безопасному, то пользователю «Initial Admin Identity» автоматически предоставляются права на изменение потока. |  |

Далее описаны некоторые распространенные варианты использования.

* + - 1. На основе файлов (аутентификация LDAP)

Пример записи LDAP с именем Иван Петров (Ivan Petrov):

<authorizers>

<userGroupProvider>

<identifier>file-user-group-provider</identifier>

<class>org.apache.nifi.authorization.FileUserGroupProvider</class>

<property name="Users File">./conf/users.xml</property>

<property name="Legacy Authorized Users File"></property>

<property name="Initial User Identity 1">cn=Ivan Petrov,ou=people,dc=example,dc=com</property>

</userGroupProvider>

<accessPolicyProvider>

<identifier>file-access-policy-provider</identifier>

<class>org.apache.nifi.authorization.FileAccessPolicyProvider</class>

<property name="User Group Provider">file-user-group-provider</property>

<property name="Authorizations File">./conf/authorizations.xml</property>

<property name="Initial Admin Identity">cn=Ivan Petrov,ou=people,dc=example,dc=com</property>

<property name="Legacy Authorized Users File"></property>

<property name="Node Identity 1"></property>

</accessPolicyProvider>

<authorizer>

<identifier>managed-authorizer</identifier>

<class>org.apache.nifi.authorization.StandardManagedAuthorizer</class>

<property name="Access Policy Provider">file-access-policy-provider</property>

</authorizer>

</authorizers>

* + - 1. На основе файлов (проверка подлинности при помощи Kerberos)

Пример записи Kerberos с именем Иван Петров и областью NIFI.APACHE.ORG:

<authorizers>

<userGroupProvider>

<identifier>file-user-group-provider</identifier>

<class>org.apache.nifi.authorization.FileUserGroupProvider</class>

<property name="Users File">./conf/users.xml</property>

<property name="Legacy Authorized Users File"></property>

<property name="Initial User Identity 1">ivanpetrov@NIFI.APACHE.ORG</property>

</userGroupProvider>

<accessPolicyProvider>

<identifier>file-access-policy-provider</identifier>

<class>org.apache.nifi.authorization.FileAccessPolicyProvider</class>

<property name="User Group Provider">file-user-group-provider</property>

<property name="Authorizations File">./conf/authorizations.xml</property>

<property name="Initial Admin Identity">ivanpetrov@NIFI.APACHE.ORG</property>

<property name="Legacy Authorized Users File"></property>

<property name="Node Identity 1"></property>

</accessPolicyProvider>

<authorizer>

<identifier>managed-authorizer</identifier> <class>org.apache.nifi.authorization.StandardManagedAuthorizer</class>

<property name="Access Policy Provider">file-access-policy-provider</property>

</authorizer>

</authorizers>

* + - 1. Пользователи и группы на основе LDAP, ссылающиеся на DN пользователя

Пример загрузки пользователей и групп из LDAP. Членство в группе будет определяться атрибутом члена каждой группы. Авторизация по-прежнему будет использовать политики доступа на основе файлов:

dn: cn=User 1,ou=users,o=nifi

objectClass: organizationalPerson

objectClass: person

objectClass: inetOrgPerson

objectClass: top

cn: User 1

sn: User1

uid: user1

dn: cn=User 2,ou=users,o=nifi

objectClass: organizationalPerson

objectClass: person

objectClass: inetOrgPerson

objectClass: top

cn: User 2

sn: User2

uid: user2

dn: cn=admins,ou=groups,o=nifi

objectClass: groupOfNames

objectClass: top

cn: admins

member: cn=User 1,ou=users,o=nifi

member: cn=User 2,ou=users,o=nifi

<authorizers>

<userGroupProvider>

<identifier>ldap-user-group-provider</identifier>

<class>org.apache.nifi.ldap.tenants.LdapUserGroupProvider</class>

<property name="Authentication Strategy">ANONYMOUS</property>

<property name="Manager DN"></property>

<property name="Manager Password"></property>

<property name="TLS - Keystore"></property>

<property name="TLS - Keystore Password"></property>

<property name="TLS - Keystore Type"></property>

<property name="TLS - Truststore"></property>

<property name="TLS - Truststore Password"></property>

<property name="TLS - Truststore Type"></property>

<property name="TLS - Client Auth"></property>

<property name="TLS - Protocol"></property>

<property name="TLS - Shutdown Gracefully"></property>

<property name="Referral Strategy">FOLLOW</property>

<property name="Connect Timeout">10 secs</property>

<property name="Read Timeout">10 secs</property>

<property name="Url">ldap://localhost:10389</property>

<property name="Page Size"></property>

<property name="Sync Interval">30 mins</property>

<property name="Group Membership - Enforce Case Sensitivity">false</property>

<property name="User Search Base">ou=users,o=nifi</property>

<property name="User Object Class">person</property>

<property name="User Search Scope">ONE\_LEVEL</property>

<property name="User Search Filter"></property>

<property name="User Identity Attribute">cn</property>

<property name="User Group Name Attribute"></property>

<property name="User Group Name Attribute - Referenced Group Attribute"></property>

<property name="Group Search Base">ou=groups,o=nifi</property>

<property name="Group Object Class">groupOfNames</property>

<property name="Group Search Scope">ONE\_LEVEL</property>

<property name="Group Search Filter"></property>

<property name="Group Name Attribute">cn</property>

<property name="Group Member Attribute">member</property>

<property name="Group Member Attribute - Referenced User Attribute"></property>

</userGroupProvider>

<accessPolicyProvider>

<identifier>file-access-policy-provider</identifier>

<class>org.apache.nifi.authorization.FileAccessPolicyProvider</class>

<property name="User Group Provider">ldap-user-group-provider</property>

<property name="Authorizations File">./conf/authorizations.xml</property>

<property name="Initial Admin Identity">Ivan Petrov</property>

<property name="Legacy Authorized Users File"></property>

<property name="Node Identity 1"></property>

</accessPolicyProvider>

<authorizer>

<identifier>managed-authorizer</identifier>

<class>org.apache.nifi.authorization.StandardManagedAuthorizer</class>

<property name="Access Policy Provider">file-access-policy-provider</property>

</authorizer>

</authorizers>

* + - 1. Пользователи и группы на основе LDAP, ссылающиеся на атрибут пользователя

Пример загрузки пользователей и групп из LDAP. Членство в группе будет определяться атрибутом member uid каждой группы. Авторизация по-прежнему будет использовать политики доступа на основе файлов:

dn: uid=User 1,ou=Users,dc=local

objectClass: inetOrgPerson

objectClass: posixAccount

objectClass: shadowAccount

uid: user1

cn: User 1

dn: uid=User 2,ou=Users,dc=local

objectClass: inetOrgPerson

objectClass: posixAccount

objectClass: shadowAccount

uid: user2

cn: User 2

dn: cn=Managers,ou=Groups,dc=local

objectClass: posixGroup

cn: Managers

memberUid: user1

memberUid: user2

<authorizers>

<userGroupProvider>

<identifier>ldap-user-group-provider</identifier>

<class>org.apache.nifi.ldap.tenants.LdapUserGroupProvider</class>

<property name="Authentication Strategy">ANONYMOUS</property>

<property name="Manager DN"></property>

<property name="Manager Password"></property>

<property name="TLS - Keystore"></property>

<property name="TLS - Keystore Password"></property>

<property name="TLS - Keystore Type"></property>

<property name="TLS - Truststore"></property>

<property name="TLS - Truststore Password"></property>

<property name="TLS - Truststore Type"></property>

<property name="TLS - Client Auth"></property>

<property name="TLS - Protocol"></property>

<property name="TLS - Shutdown Gracefully"></property>

<property name="Referral Strategy">FOLLOW</property>

<property name="Connect Timeout">10 secs</property>

<property name="Read Timeout">10 secs</property>

<property name="Url">ldap://localhost:10389</property>

<property name="Page Size"></property>

<property name="Sync Interval">30 mins</property>

<property name="Group Membership - Enforce Case Sensitivity">false</property>

<property name="User Search Base">ou=Users,dc=local</property>

<property name="User Object Class">posixAccount</property>

<property name="User Search Scope">ONE\_LEVEL</property>

<property name="User Search Filter"></property>

<property name="User Identity Attribute">cn</property>

<property name="User Group Name Attribute"></property>

<property name="User Group Name Attribute - Referenced Group Attribute"></property>

<property name="Group Search Base">ou=Groups,dc=local</property>

<property name="Group Object Class">posixGroup</property>

<property name="Group Search Scope">ONE\_LEVEL</property>

<property name="Group Search Filter"></property>

<property name="Group Name Attribute">cn</property>

<property name="Group Member Attribute">memberUid</property>

<property name="Group Member Attribute - Referenced User Attribute">uid</property>

</userGroupProvider>

<accessPolicyProvider>

<identifier>file-access-policy-provider</identifier>

<class>org.apache.nifi.authorization.FileAccessPolicyProvider</class>

<property name="User Group Provider">ldap-user-group-provider</property>

<property name="Authorizations File">./conf/authorizations.xml</property>

<property name="Initial Admin Identity">Ivan Petrov</property>

<property name="Legacy Authorized Users File"></property>

<property name="Node Identity 1"></property>

</accessPolicyProvider>

<authorizer>

<identifier>managed-authorizer</identifier>

<class>org.apache.nifi.authorization.StandardManagedAuthorizer</class>

<property name="Access Policy Provider">file-access-policy-provider</property>

</authorizer>

</authorizers>

* + - 1. Составной – Пользователи и группы на основе файлов и LDAP

Пример составной реализации, загружающей пользователей и группы из LDAP и локального файла. Членство в группе будет определяться атрибутом члена каждой группы. Пользователи из LDAP будут доступны только для чтения, в то время как пользователи, загруженные из файла, будут настраиваться в пользовательском интерфейсе:

dn: cn=User 1,ou=users,o=nifi

objectClass: organizationalPerson

objectClass: person

objectClass: inetOrgPerson

objectClass: top

cn: User 1

sn: User1

uid: user1

dn: cn=User 2,ou=users,o=nifi

objectClass: organizationalPerson

objectClass: person

objectClass: inetOrgPerson

objectClass: top

cn: User 2

sn: User2

uid: user2

dn: cn=admins,ou=groups,o=nifi

objectClass: groupOfNames

objectClass: top

cn: admins

member: cn=User 1,ou=users,o=nifi

member: cn=User 2,ou=users,o=nifi

<authorizers>

<userGroupProvider>

<identifier>file-user-group-provider</identifier>

<class>org.apache.nifi.authorization.FileUserGroupProvider</class>

<property name="Users File">./conf/users.xml</property>

<property name="Legacy Authorized Users File"></property>

<property name="Initial User Identity 1">cn=nifi-node1,ou=servers,dc=example,dc=com</property>

<property name="Initial User Identity 2">cn=nifi-node2,ou=servers,dc=example,dc=com</property>

</userGroupProvider>

<userGroupProvider>

<identifier>ldap-user-group-provider</identifier>

<class>org.apache.nifi.ldap.tenants.LdapUserGroupProvider</class>

<property name="Authentication Strategy">ANONYMOUS</property>

<property name="Manager DN"></property>

<property name="Manager Password"></property>

<property name="TLS - Keystore"></property>

<property name="TLS - Keystore Password"></property>

<property name="TLS - Keystore Type"></property>

<property name="TLS - Truststore"></property>

<property name="TLS - Truststore Password"></property>

<property name="TLS - Truststore Type"></property>

<property name="TLS - Client Auth"></property>

<property name="TLS - Protocol"></property>

<property name="TLS - Shutdown Gracefully"></property>

<property name="Referral Strategy">FOLLOW</property>

<property name="Connect Timeout">10 secs</property>

<property name="Read Timeout">10 secs</property>

<property name="Url">ldap://localhost:10389</property>

<property name="Page Size"></property>

<property name="Sync Interval">30 mins</property>

<property name="Group Membership - Enforce Case Sensitivity">false</property>

<property name="User Search Base">ou=users,o=nifi</property>

<property name="User Object Class">person</property>

<property name="User Search Scope">ONE\_LEVEL</property>

<property name="User Search Filter"></property>

<property name="User Identity Attribute">cn</property>

<property name="User Group Name Attribute"></property>

<property name="User Group Name Attribute - Referenced Group Attribute"></property>

<property name="Group Search Base">ou=groups,o=nifi</property>

<property name="Group Object Class">groupOfNames</property>

<property name="Group Search Scope">ONE\_LEVEL</property>

<property name="Group Search Filter"></property>

<property name="Group Name Attribute">cn</property>

<property name="Group Member Attribute">member</property>

<property name="Group Member Attribute - Referenced User Attribute"></property>

</userGroupProvider>

<userGroupProvider>

<identifier>composite-user-group-provider</identifier>

<class>org.apache.nifi.authorization.CompositeConfigurableUserGroupProvider</class>

<property name="Configurable User Group Provider">file-user-group-provider</property>

<property name="User Group Provider 1">ldap-user-group-provider</property>

</userGroupProvider>

<accessPolicyProvider>

<identifier>file-access-policy-provider</identifier>

<class>org.apache.nifi.authorization.FileAccessPolicyProvider</class>

<property name="User Group Provider">composite-user-group-provider</property>

<property name="Authorizations File">./conf/authorizations.xml</property>

<property name="Initial Admin Identity">Ivan Petrov</property>

<property name="Legacy Authorized Users File"></property>

<property name="Node Identity 1">cn=nifi-node1,ou=servers,dc=example,dc=com</property>

<property name="Node Identity 2">cn=nifi-node2,ou=servers,dc=example,dc=com</property>

</accessPolicyProvider>

<authorizer>

<identifier>managed-authorizer</identifier>

<class>org.apache.nifi.authorization.StandardManagedAuthorizer</class>

<property name="Access Policy Provider">file-access-policy-provider</property>

</authorizer>

</authorizers>

# Политики доступа

Глобальные политики доступа приведены в таблице 11.

1. Глобальные политики доступа

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Admin | DataFlow-Manager | Monitor | Provenance | ПС «Атом.Мост» | Proxy |
| Просмотр UI | **◉** | **◉** | **◉** |  |  |  |
| Доступ к контроллеру (просмотр) | **◉** | **◉** | **◉** |  | **◉** |  |
| Доступ к контроллеру (изменение) |  | **◉** |  |  |  |  |
| Доступ к parameter contexts (просмотр) |  |  |  |  |  |  |
| Доступ к parameter contexts (измене-ние) |  |  |  |  |  |  |
| Источники запроса |  |  |  | **◉** |  |  |
| Доступ к restricted components |  | **◉** |  |  |  |  |
| Доступ к политикам доступа (просмотр) | **◉** |  |  |  |  |  |
| Доступ к политикам доступа (изменение) | **◉** |  |  |  |  |  |
| Доступ к пользователям/группам (просмотр) | **◉** |  |  |  |  |  |
| Доступ к пользователям/группам (изменение) | **◉** |  |  |  |  |  |
| Информация о межсайтовом взаимодействии |  |  |  |  | **◉** |  |
| Просмотр диагностики системы |  | **◉** | **◉** |  |  |  |
| Запросы пользователей через proxy |  |  |  |  |  | **◉** |
| Счетчики доступа |  |  |  |  |  |  |

Политики доступа к компонентам в группе корневых процессов приведены в таблице 12.

1. Политики доступа к компонентам в группе корневых процессов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Admin | DataFlow-Manager | Monitor | Provenance | ПС «Атом.Мост» | Proxy |
| Просмотр компонента | **◉** | **◉** | **◉** |  |  |  |
| Изменение компонента |  | **◉** |  |  |  |  |
| Просмотр данных |  | **◉** |  | **◉** |  | **◉** |
| Изменение данных |  | **◉** |  |  |  | **◉** |
| Просмотр происхождения |  |  |  | **◉** |  |  |

Администратор может управлять возможностью пользователей и групп просматривать или изменять ресурсы ПС «Атом.Мост» с помощью «политик доступа». К ресурсу можно применить два типа политик доступа:

* Просмотр - если для ресурса создана политика просмотра, только пользователи или группы, которые добавлены к этой политике, могут видеть детали этого ресурса;
* Изменение - если у ресурса есть политика изменения, только пользователи или группы, которые добавлены к этой политике, могут изменять конфигурацию этого ресурса.

Администратор может создавать и применять политики доступа как на глобальном, так и на компонентном уровне.

## Глобальные политики доступа

Политики глобального доступа управляют авторизациями системного уровня, приведенными в таблице 13.

1. Глобальные политики доступа

|  |  |  |  |
| --- | --- | --- | --- |
| Политика | Привилегия | Глобальное меню | Описание ресурса |
| view the UI | Разрешение пользователям просматривать UI | N/A | /flow |
| access the controller | Позволяет пользователям просматривать/изменять контроллер, включая задачи отчетности, службы контроллеров и узлы в кластере | Controller Settings | /controller |
| query provenance | Позволяет пользователям отправлять Provenance Search и запрашивать Event Lineage | Data Provenance | /provenance |
| access restricted components | Позволяет пользователям создавать/изменять ограниченные компоненты при условии наличия других разрешений. Ограниченные компоненты могут указывать, какие конкретные разрешения требуются. Разрешения могут предоставляться для определенных ограничений или независимо от них. Если разрешение предоставляется независимо от ограничений, пользователь может создавать/изменять все ограниченные компоненты | N/A | /restricted-components |
| access all policies | Позволяет пользователям просматривать/изменять политики для всех компонентов | Policies | /policies |
| access users/user groups | Позволяет пользователям просматривать/изменять пользователей и группы пользователей | Users | /tenants |
| retrieve site-to-site details | Позволяет другим инстансам NiFi извлекать информацию site-to-site | N/A | /site-to-site |
| view system diagnostics | Позволяет пользователям просматривать системную диагностику | Summary | /system |
| proxy user requests | Позволяет прокси отправлять запросы от имени других пользователей | N/A | /proxy |
| access counters | Позволяет пользователям просматривать/изменять счетчики доступа | Counters | /counter |

## Политики доступа на уровне компонентов

Политики доступа на уровне компонентов управляют полномочиями на уровне компонентов, приведенные в таблице 14.

1. Политики доступа на уровне компонентов

|  |  |  |
| --- | --- | --- |
| Политика | Привилегия | Глобальное меню |
| view the component | Позволяет пользователям просматривать детали конфигурации компонентов | resource=”/<component-type>/<component-UUID>” action=”R” |
| modify the component | Позволяет пользователям изменять детали конфигурации компонентов | resource=”/<component-type>/<component-UUID>” action=”W” |
| view provenance | Позволяет пользователям просматривать события происхождения, созданные компонентом | resource=”/provenance-data/<component-type>/<component-UUID>” action=”R” |
| view the data | Позволяет пользователям просматривать метаданные и содержимое компонента в очередях потока в исходящих соединениях и через события происхождения | resource=”/data/<component-type>/<component-UUID>” action=”R” |
| modify the data | Позволяет пользователям очищать очереди потоков в исходящих соединениях и повторно отправлять через события происхождения | resource=”/data/<component-type>/<component-UUID>” action=”W” |
| view the policies | Позволяет пользователям просматривать список пользователей, которые могут просматривать/изменять компонент | resource=”/policies/<component-type>/<component-UUID>” action=”R” |
| modify the policies | Позволяет пользователям изменять список пользователей, которые могут просматривать/изменять компонент | resource=”/policies/<component-type>/<component-UUID>” action=”W” |
| receive data via site-to-site | Позволяет порту получать данные из инстансов NiFi | resource=”/data-transfer/input-ports/<port-UUID>” action=”W” |
| send data via site-to-site | Позволяет порту отправлять данные из инстансов NiFi | resource=”/data-transfer/output-ports/<port-UUID>” action=”W” |

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!**  Политики доступа можно применять ко всем типам компонентов, кроме соединений. Разрешения на соединения определяются по индивидуальным политикам доступа к исходному и целевому компонентам соединения, а также по политике доступа группы процессов, содержащей компоненты.  Для доступа к List Queue и Delete Queue для соединения пользователю требуются политики “view the data” и “modify the data” на компоненте. Так же все узлы в кластерной среде должны быть добавлены к этим политикам, так как запрос пользователя может быть реплицирован через любой узел в кластере. |  |

## Наследование политики доступа

Администратору не обязательно вручную создавать политики для каждого компонента в потоке данных. Чтобы сократить время, которое администраторы тратят на управление доступом, политики наследуются от родительского ресурса к дочернему. Например, если пользователю предоставлен доступ для просмотра и изменения группы процессов, этот пользователь также может просматривать и изменять компоненты в группе процессов. Наследование политик позволяет администратору назначать политики одновременно и применять политики ко всему потоку данных.

Администратор так же может переопределить унаследованную политику. Переопределение политики удаляет унаследованную политику, разрывая цепочку наследования от родительского к дочернему, и создает заменяющую политику для добавления пользователей по желанию. Унаследованные политики и пользователей, привязанных к ним, можно восстановить, удалив замещающую политику.

## Создание пользователей и групп

В пользовательском интерфейсе выберите «Пользователи» в глобальном меню. Откроется диалоговое окно для создания пользователей и групп, а также управления ими (рисунок 1).

Изображение выглядит как текст

Автоматически созданное описание

1. Пользователи

Щелкните значок Добавить (). Чтобы создать пользователя, введите информацию «Identity», соответствующую методу аутентификации, выбранному для защиты вашего экземпляра ПС «Атом.Мост». Щелкните ОК (рисунок 2).

Изображение выглядит как текст

Автоматически созданное описание

1. Добавить пользователя

Чтобы создать группу, выберите переключатель «Группа», введите имя группы и выберите пользователей, которые будут включены в группу. Щелкните ОК (рисунок 3).

Изображение выглядит как текст

Автоматически созданное описание

1. Добавить группу пользователей

## Просмотр политик для пользователей

В пользовательском интерфейсе выберите «Пользователи» в глобальном меню. Откроется диалоговое окно «Пользователи NiFi» (рисунок 4).

Изображение выглядит как текст

Автоматически созданное описание

1. Пользователи

Щелкните значок Просмотр политик пользователей (, рисунок 5).

Изображение выглядит как стол

Автоматически созданное описание

1. Просмотр политик пользователей

В окне «Политики пользователей» отображаются глобальные политики и политики на уровне компонентов, установленные для выбранного пользователя. Щелкните значок «Перейти к» (  ), чтобы перейти к этому компоненту на холсте.

## Примеры настройки политики доступа

Самый эффективный способ понять, как создавать и применять политики доступа – это рассмотреть несколько распространенных примеров. В приведенных в главе сценариях предполагается, что User1 является администратором, а User2 - новым пользователем, которому был предоставлен доступ только к пользовательскому интерфейсу.

Начнем с двух процессоров на холсте в качестве отправной точки: GenerateFlowFile и LogAttribute (рисунок 6).

Изображение выглядит как стол

Автоматически созданное описание

1. Процессоры GenerateFlowFile и LogAttribute

User1 может добавлять компоненты в поток данных и может перемещать, редактировать и подключать все процессоры. Подробности и свойства корневой группы процессов и процессоров видны пользователю User1 (рисунок 7).

Изображение выглядит как текст, стол

Автоматически созданное описание

1. Управление компонентами User1

Пользователь User1 хочет сохранить свои текущие права доступа к потоку данных и его компонентам.

User2 не может добавлять компоненты в поток данных или перемещать, редактировать и подключать компоненты. Подробности и свойства корневой группы процессов и процессоров скрыты от User2 (рисунок 8).

Изображение выглядит как текст

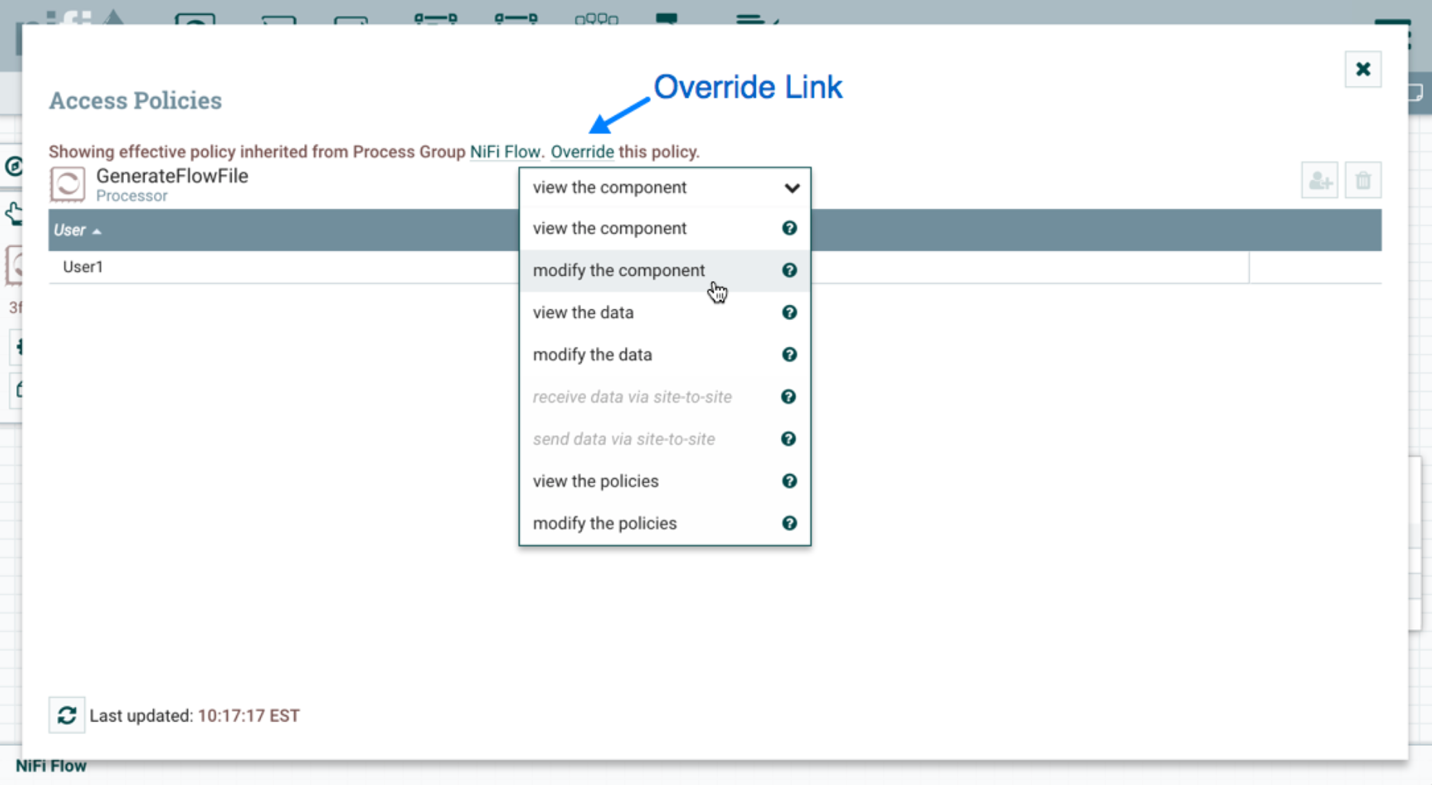
Автоматически созданное описание

1. Управление компонентами User2

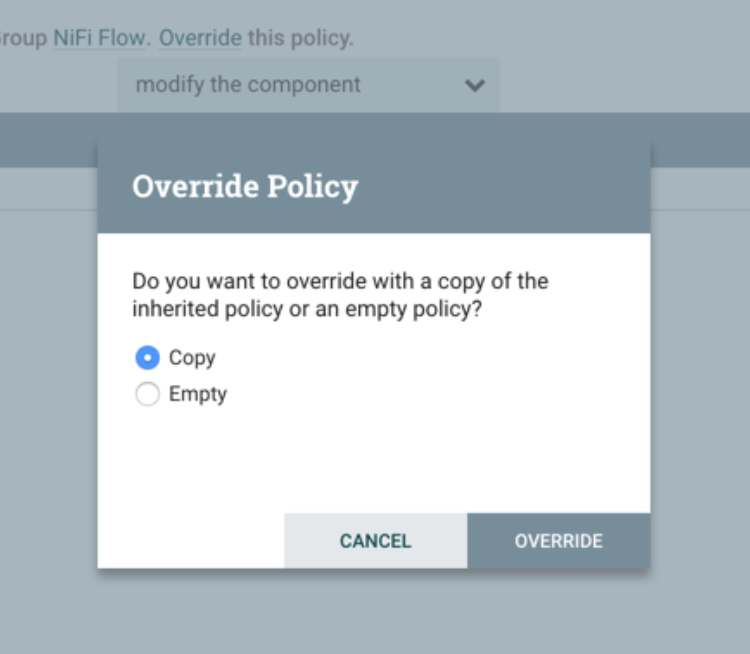
### Перемещение процессора

Чтобы позволить пользователю User2 перемещать, например, процессор GenerateFlowFile в потоке данных и только этот процессор, нужно выполнить следующие шаги:

1. Выберите процессор GenerateFlowFile, чтобы он был выделен;
2. Щелкните значок «Политики доступа» () на палитре «Operate», и откроется диалоговое окно «Политики доступа».
3. В раскрывающемся списке политики выберите «изменить компонент». Политика «изменить компонент», которая в настоящее время существует на процессоре, является политикой «изменить компонент», унаследованной от корневой группы процессов (родительской), на которую пользователь User1 имеет привилегии (рисунок 9).



1. Изменение компонентов политики
2. Выберите ссылку «Переопределить» в сообщении о наследовании политики. При создании политики замены вам предоставляется выбор: заменить ее копией унаследованной политики или пустой политикой. Нажмите кнопку «Переопределить», чтобы создать копию (рисунок 10).



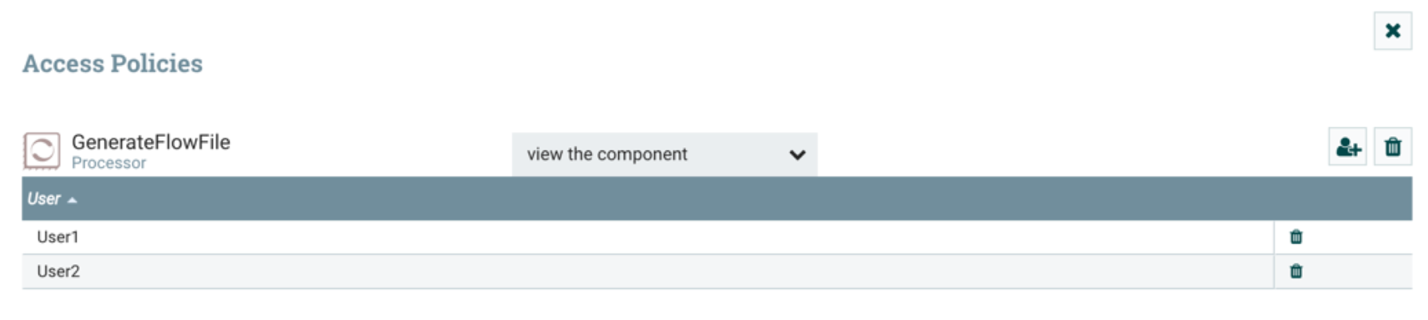
1. Переопределение
2. В созданной измененной политике щелкните значок «Добавить пользователя» (). Найдите или введите User2 в поле «User Identity» и нажмите OK. С этими изменениями User1 сохраняет возможность перемещать оба процессора на холсте. Пользователь User2 теперь может перемещать процессор GenerateFlowFile, но не может перемещать процессор LogAttribute.

### Редактирование процессора

В приведенном выше примере «Перемещение процессора» пользователь User2 был добавлен в политику «изменить компонент» для GenerateFlowFile. Без возможности просмотра свойств процессора User2 не может изменять конфигурацию процессора. Чтобы редактировать компонент, пользователь должен иметь доступ как к «просмотру компонента», так и на «изменение компонента». Для этого User1 должен выполнить следующие шаги:

1. Выберите процессор GenerateFlowFile.
2. Щелкните значок «Политики доступа» (Изображение выглядит как человек, часы, держит, мужчина

   Автоматически созданное описание) на палитре «Operate», и откроется диалоговое окно «Политики доступа».
3. В раскрывающемся списке политики выберите «просмотреть компонент». Политика просмотра компонента, которая в настоящее время существует на процессоре (дочернем), является политикой «просмотра компонента», унаследованной от корневой группы процессов (родительской), на которую пользователь User1 имеет привилегии (см рис. 6).
4. Выберите ссылку «Переопределить» в сообщении о наследовании политики, оставьте значение по умолчанию «Копировать политику» и нажмите кнопку «Переопределить».
5. В созданной политике переопределения щелкните значок Добавить пользователя (). Найдите или введите User2 в поле «User Identity» и нажмите OK. С этими изменениями User1 сохраняет возможность просматривать и редактировать процессоры на холсте. User2 теперь может просматривать и редактировать процессор GenerateFlowFile (рисунок 11).



1. Access Policies

### Создание подключения

С политиками доступа, настроенными, как описано в двух предыдущих примерах, пользователь User1 может подключить GenerateFlowFile к LogAttribute, а User2 не может установить соединение, потому что:

* Пользователь User2 не имеет прав на изменение группы процессов.
* Несмотря на то, что User2 имеет доступ для просмотра и изменения исходного компонента (GenerateFlowFile), User2 не имеет политики доступа для целевого компонента (LogAttribute).

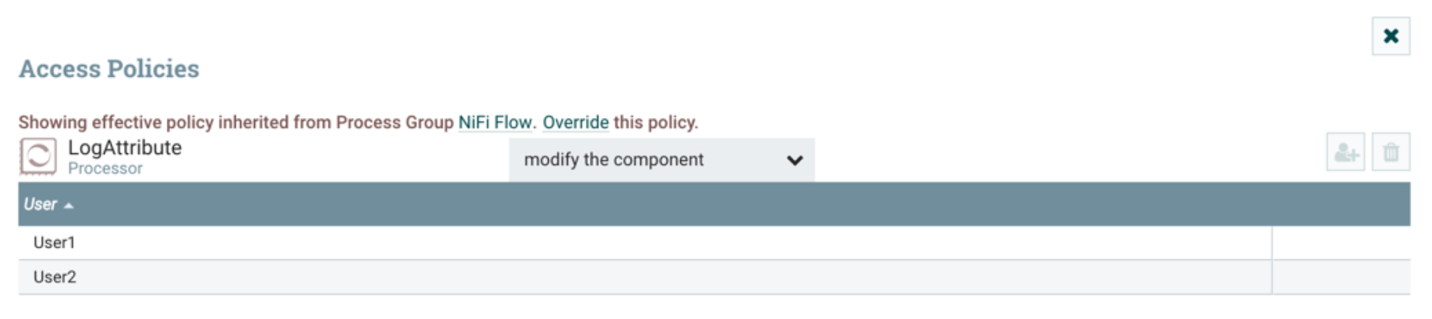
Чтобы позволить User2 подключать GenerateFlowFile к LogAttribute, нужно выполнить следующее:

1. Выберите корневую группу процессов. Палитра «Управление» обновлена ​​подробными сведениями о корневой группе процессов.
2. Щелкните значок «Политики доступа» (Изображение выглядит как человек, часы, держит, мужчина

   Автоматически созданное описание) на палитре «Operate», и откроется диалоговое окно «Политики доступа».
3. В раскрывающемся списке политики выберите «изменить компонент».
4. Щелкните значок Добавить пользователя (). Найдите или введите User2 и выберите OK.

Добавляя User2 в политику «изменить компонент» в группе процессов, User2 добавляется в политику «изменить компонент» в процессоре LogAttribute путем наследования политики. Чтобы подтвердить это, выделите процессор LogAttribute и выберите значок Политики доступа (Изображение выглядит как человек, часы, держит, мужчина

Автоматически созданное описание) на палитре «Operate» (рисунок 12).

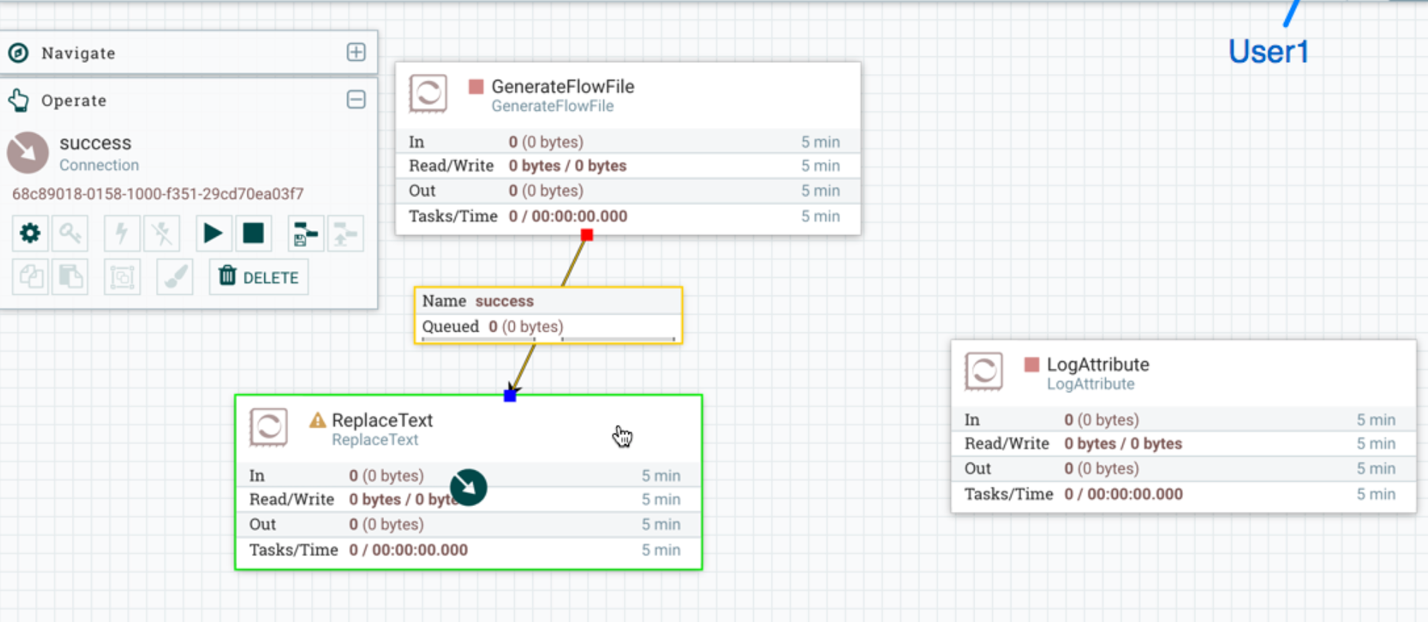


1. Изменение политики в процессоре LogAttribute

С этими изменениями пользователь 2 теперь может подключить процессор GenerateFlowFile к процессору LogAttribute.

### Редактирование соединения

Предположим, что пользователи User1 или User2 добавляют процессор ReplaceText в корневую группу процессов. Пользователь User1 может выбрать и изменить существующее соединение (между GenerateFlowFile и LogAttribute), чтобы теперь подключить GenerateFlowFile к ReplaceText (рисунок 13).

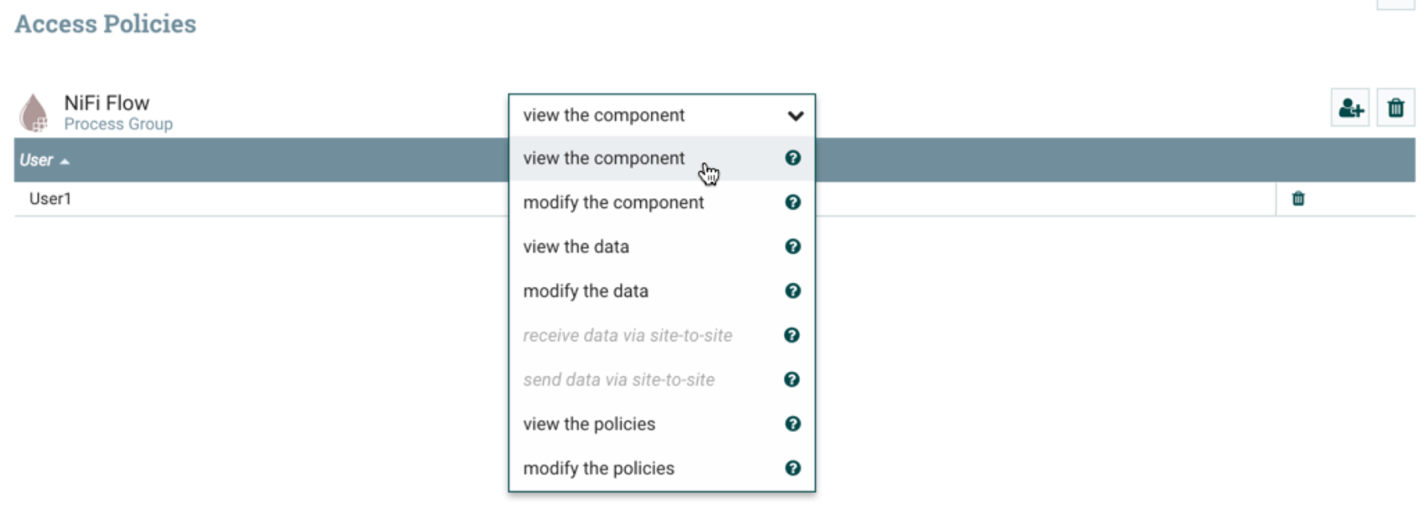


1. Подключение GenerateFlowFile к ReplaceText

User2 не может выполнить это действие. Чтобы позволить User2 подключать GenerateFlowFile к ReplaceText, необходимо:

1. Выберите корневую группу процессов. Палитра «Управление» обновлена ​​подробными сведениями о корневой группе процессов.
2. Щелкните значок Политики доступа (Изображение выглядит как человек, часы, держит, мужчина

   Автоматически созданное описание).
3. В раскрывающемся списке политики выберите «просмотреть компонент» (рисунок 14).



1. Просмотр компонента
2. Щелкните значок Добавить пользователя (). Найдите или введите User2 и выберите OK.

Будучи добавленным в политики просмотра и изменения для группы процессов, User2 теперь может подключить процессор GenerateFlowFile к процессору ReplaceText.

## Идентификационные данные узла кластера

Если вы используете ПС «Атом.Мост» в кластерной среде, вы должны указать идентификаторы для каждого узла. Политики авторизации, необходимые для связи узлов, создаются во время запуска.

Пример: если вы настраиваете кластер из 2 узлов со следующими DN для каждого узла.

cn=nifi-1,ou=people,dc=example,dc=com

cn=nifi-2,ou=people,dc=example,dc=com

<authorizers>

<userGroupProvider>

<identifier>file-user-group-provider</identifier>

<class>org.apache.nifi.authorization.FileUserGroupProvider</class>

<property name="Users File">./conf/users.xml</property>

<property name="Legacy Authorized Users File"></property>

<property name="Initial User Identity 1">ivanpetrov@NIFI.APACHE.ORG</property>

<property name="Initial User Identity 2">cn=nifi-1,ou=people,dc=example,dc=com</property>

<property name="Initial User Identity 3">cn=nifi-2,ou=people,dc=example,dc=com</property>

</userGroupProvider>

<accessPolicyProvider>

<identifier>file-access-policy-provider</identifier>

<class>org.apache.nifi.authorization.FileAccessPolicyProvider</class>

<property name="User Group Provider">file-user-group-provider</property>

<property name="Authorizations File">./conf/authorizations.xml</property>

<property name="Initial Admin Identity">ivanpetrov@NIFI.APACHE.ORG</property>

<property name="Legacy Authorized Users File"></property>

<property name="Node Identity 1">cn=nifi-1,ou=people,dc=example,dc=com</property>

<property name="Node Identity 2">cn=nifi-2,ou=people,dc=example,dc=com</property>

</accessPolicyProvider>

<authorizer>

<identifier>managed-authorizer</identifier>

<class>org.apache.nifi.authorization.StandardManagedAuthorizer</class>

<property name="Access Policy Provider">file-access-policy-provider</property>

</authorizer>

</authorizers>

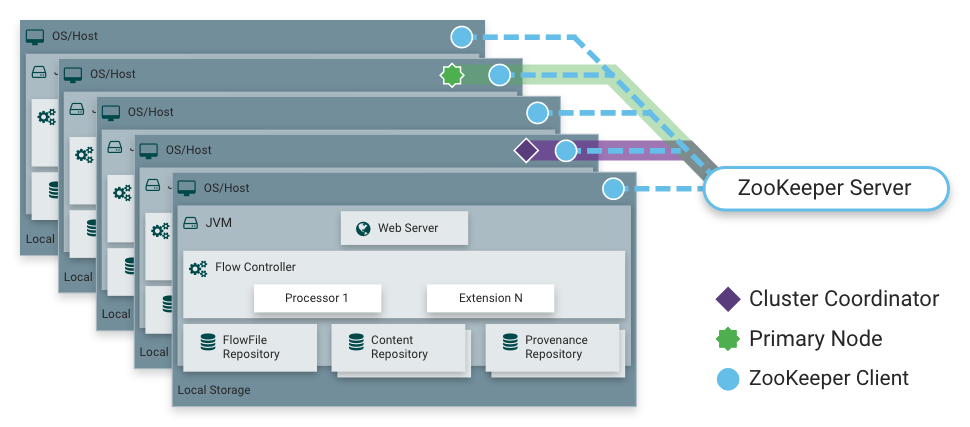
|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!** В кластере все узлы должны иметь одинаковые файлы authorizations.xml и users.xml. Единственное исключение - если узел имеет пустые файлы authorizations.xml и users.xml до присоединения к кластеру. В этом сценарии узел наследует их от кластера во время запуска. |  |

Теперь, когда созданы первоначальные авторизации, можно создавать дополнительные пользователи, группы и авторизации, а также управлять ими в пользовательском интерфейсе ПС «Атом.Мост».

# Кластеризация

ПС «Атом.Мост» использует парадигму кластеризации без «лидера». Это обеспечивает отказоустойчивость и дает возможность горизонтально масштабировать нагрузку. Каждый узел (Node) в кластере имеет идентичное определение потока и выполняет одни и те же задачи с данными, но каждый из узлов работают с уникальным набором данных. Кластер автоматически распределяет данные по всем активным узлам.

ПС «Атом.Мост» выбирает один из узлов автоматически (через Apache ZooKeeper) в качестве «координатора» кластера. Затем все узлы в кластере будут отправлять этому узлу информацию о состоянии, и этот узел будет отвечает за отключение от кластера узлов, которые не сообщают о своем состоянии в течение какого-то времени. Кроме того, когда новый узел решает присоединиться к кластеру, он должен сначала подключиться к выбранному в данный момент кластеру-«координатору», чтобы получить самую последнюю информацию. Если «координатор» кластера определяет, что узлу разрешено присоединиться (на основе настроенного конфигурационного файла), текущий поток предоставляется этому узлу, и он может присоединиться к кластеру, предполагая, что копия потока узла соответствует копии, которая предоставляется «координатором» кластера. Если версия конфигурации потока узла отличается от версии координатора кластера, узел не может присоединиться к кластеру (рис. 1).



1. Кластеризация ПС «Атом.Мост»

Выбор данной архитектуры кластеризации основывается на том, что использование одного экземпляра ПС «Атом.Мост» на одном сервере недостаточно для обработки имеющегося у них объема данных. Одно из решений - запустить один и тот же поток данных на нескольких серверах. Однако это создает проблему управления, потому что каждый раз, когда DataFlow-менеджеру нужно изменить или обновить поток данных, он должен внести эти изменения на каждом сервере, а затем контролировать каждый сервер индивидуально.

Путем кластеризации серверов ПС «Атом.Мост» можно получить масштабируемые возможности обработки в едином интерфейсе, через который можно вносить изменения в потоки данных и контролировать их. Кластеризация позволяет DataFlow-менеджеру вносить каждое изменение только один раз, а затем это изменение передается всем узлам кластера. Через этот же интерфейс DataFlow-менеджер может также контролировать работоспособность и статус всех узлов.

## Особенности терминологии

Кластеризация ПС «Атом.Мост» уникальна и наследует таковую в NiFi вместе терминологией. Перед созданием кластера важно понять нижеописанные термины.

### «Координатор» кластера

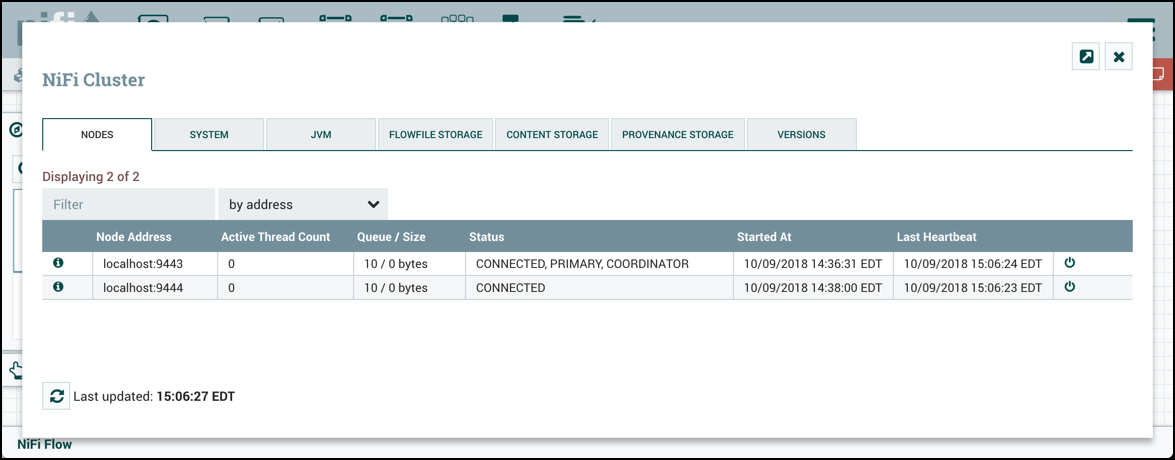
«Координатор» кластера – это узел в кластере ПС «Атом.Мост», который отвечает за выполнение задач по управлению узлами в кластере и обеспечение синхронизации потоков для вновь присоединяющихся узлов. Когда DataFlow-менеджер управляет потоком данных в кластере, они могут делать это через пользовательский интерфейс любого узла в кластере. Любое внесенное изменение затем копируется на все узлы кластера.

### Узлы

Каждый кластер состоит из одного или нескольких узлов. Узлы выполняют фактическую обработку данных.

### Первичный узел

У каждого кластера должен быть первичный узел, на котором можно запускать «изолированные процессоры». ZooKeeper автоматически выбирает узел и присваивает ему статус первичного. Если этот узел по какой-либо причине отключается от кластера, автоматически будет выбран новый основной узел. Администраторы и пользователи могут определить, какой узел в настоящее время выбран в качестве первичного узла, просмотрев страницу управления кластером пользовательского интерфейса (рис. 2).



1. Страница управления кластером

### Изолированные процессоры

В кластере ПС «Атом.Мост» один и тот же поток данных и каждый его компонент работает на всех узлах. Однако могут быть случаи, когда DataFlow-менеджер не хочет, чтобы каждый процессор работал на каждом узле. Наиболее распространенный случай – использование процессора, который взаимодействует с внешней службой по протоколу, который не может или плохо масштабируется. Таким образом, DataFlow-менеджер может настроить такой процессор на основном узле для работы изолированно, что означает, что он работает только на этом узле. При правильной конфигурации потока данных он может получать данные и распределять их нагрузку между остальными узлами кластера.

### Такты

Узлы сообщают о своем состоянии и статусе «координатору» кластера посредством «тактов», которые позволяют координатору знать, что они все еще подключены к кластеру и работают правильно.

По умолчанию узлы излучают контрольные сигналы каждые 5 секунд, и если «координатор» кластера не получает контрольных сигналов от узла в течение 40 секунд, он отключает узел из-за «отсутствия контрольных сигналов». Параметры 5 секунд и 8 раз настраиваются в файле nifi.properties. «Координатору» необходимо обеспечить синхронизацию каждого узла в кластере, и если от узла не поступают регулярные сигналы, «координатор» не может быть уверен, что он все еще синхронизирован с остальными кластерами, поэтому «координатор» отключает узел, который не подает сигналы о своем состоянии. Если по прошествии 40 секунд узел все же отправит новое контрольное сообщение, «координатор» автоматически запросит повторное присоединение узла к кластеру.

Информация как об отключении из-за отсутствия сигнала, так и о повторном подключении всегда отображается в пользовательском интерфейсе.

## Обмен информацией внутри кластера

Как уже упоминалось, узлы связываются с «координатором» кластера посредством «тактов». Когда выбран «координатор» кластера, он обновляет информацию о состоянии подключения, чтобы узлы понимали, куда отправлять контрольные сигналы. Если один из узлов выходит из строя, другие узлы в кластере не будут автоматически принимать нагрузку от отсутствующего узла. DataFlow-менеджер может настроить поток данных для аварийного переключения; однако это зависит от схемы потока данных и не происходит автоматически.

Когда DataFlow-менеджер вносит изменения в поток данных, узел, который получает запрос на изменение потока, передает эти изменения всем узлам и ожидает ответа от каждого из них, сигнализирую о том, что он внес изменения в свой локальный поток.

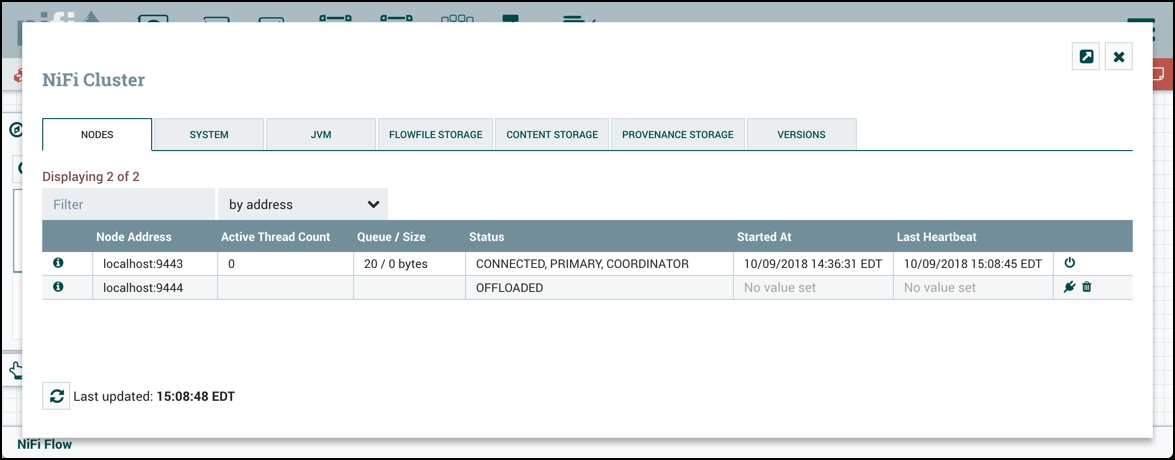
## Управление узлами

### Отключение узлов

DataFlow-менеджер может вручную отключить узел от кластера. Узел также может автоматически отключиться по другим причинам, например, из-за отсутствия подаваемых сигналов о состоянии («такт»). «Координатор» кластера отображает сводку в пользовательском интерфейсе, когда узел отключен. Пока причина отключенния узла не будет решена, DataFlow-менеджер не сможет вносить какие-либо изменения в поток данных. DataFlow-менеджеру или администратору потребуется устранить проблему с узлом и решить ее, прежде чем какие-либо новые изменения могут быть внесены в поток данных. Важно: то, что узел отключен, не означает, что он не работает. Отключение узла может произойти по нескольким причинам, например, когда узел не может связаться с координатором кластера из-за сетевых проблем.

### Выгрузка узлов

Файлы потоков, которые остаются на отключенном узле, могут быть перебалансированы на другие активные узлы в кластере с помощью выгрузки. DataFlow-менеджер или администратор может выгрузить данные из отключенного узла в пользовательском интерфейсе управления кластером (рис. 3). Это остановит все процессоры, завершит работу всех процессоров, прекратит передачу по всем группам удаленных процессов и перебалансирует потоковые файлы на другие подключенные узлы в кластере.



1. Выгрузка файлов потока

Узлы, которые остаются в состоянии «выгрузки» из-за обнаруженных ошибок (нехватка памяти, отсутствие сетевого подключения и т.д.), можно повторно подключить к кластеру, перезапустив ПС «Атом.Мост» на узле. Выгруженные узлы можно повторно подключить к кластеру (выбрав соответствующий пункт или перезапустить ПС «Атом.Мост» на узле в пользовательском интерфейсе) или удалить из кластера.

### Удаление узла

DataFlow-менеджер может принять решение полностью удалить узел из кластера. В диалоговом окне «Управление кластером» выберите значок «Удалить»  для отключенного или удаленного узла. После удаления узел нельзя повторно присоединить к кластеру, пока он не будет перезапущен.

### Вывод узла из эксплуатации

Чтобы вывести узел из эксплуатации и удалить его из кластера, выполните следующие действия:

* отключите узел;
* после завершения отключения разгрузите узел;
* после завершения разгрузки удалите узел;
* после завершения запроса на удаление остановите / удалите службу на хосте.

## «Выборы» потока

При первом запуске кластера ПС «Атом.Мост» должен определить, какой из узлов имеет «правильную» версию потока. Это делается путем «голосования» по потокам, которые есть у каждого из узлов. Когда узел пытается подключиться к кластеру, он предоставляет копию своего локального потока и (если дозволено настройками политики безопасности) его пользователей и групп «координатору» кластера. Если ни один поток еще не был выбран в качестве «правильного» потока, поток узла сравнивается с потоками каждого из узлов. Если поток другого узла совпадает с копией, «голос» отдается за этот поток. Если ни один другой узел еще не сообщил о таком же потоке, этот поток будет добавлен в пул возможных выбранных потоков с одним голосом. По прошествии некоторого времени (настроенного путем установки свойства nifi.cluster.flow.election.max.wait.time) или когда какой-то поток получает определенное количество «голосов» (настраивается путем установки параметра nifi.cluster.flow.election.max.candidates), поток выбирается как "правильная" копия потока.

Любой узел, а также потоки данных, пользователи и группы которых конфликтуют с «правильной» копией потока, будет создавать резервную копию всех конфликтующих ресурсов и заменять локальные ресурсы ресурсами из кластера.

Как выполняется резервное копирование, зависит от настроенной политики доступа и групп пользователей. Для поставщиков политик доступа на основе файлов резервная копия будет записана в тот же каталог, что и существующий файл и будет иметь то же имя, но с суффиксом "." и отметку времени. Например, если сам поток конфликтует с потоком кластера в 12:05:03 1 января 2020 г., файл flow.xml.gz узла будет скопирован как flow.xml.gz.2020-01-01-12-05-03, и поток кластера будет записан в flow.xml.gz. Аналогичным образом это произойдет с файлами users.xml и authorizations.xml. Это сделано для того, чтобы при необходимости можно было вручную вернуть поток, переименовав файл резервной копии, например, обратно в flow.xml.gz.

Важно отметить, что перед тем, как наследовать выбранный поток, ПС «Атом.Мост» сначала прочитает репозиторий файлов потоков и любые файлы подкачки, чтобы определить, какие очереди в потоке данных в настоящее время содержат данные. Если в потоке данных существует какая-либо очередь, содержащая файл потока, эта очередь также должна будет иметься в выбранном потоке данных. Если эта очередь не существует в выбранном потоке данных, узел не сможет унаследовать поток данных, пользователей и группы. Вместо этого ПС «Атом.Мост» регистрирует ошибки и не запускается. Такое поведение гарантирует, что даже если на узле есть данные, хранящиеся в соединении, а поток данных кластера отличается, перезапуск узла не приведет к потере данных.

Выборы проводятся в соответствии с «всеобщим голосованием» с оговоркой, что победителем никогда не будет «пустой поток». Это позволит администратору удалить файл flow.xml.gz узла и перезапустить узел, зная, что поток узла не будет признан «правильным» потоком, если не будет найден другой поток. Если есть два непустых потока, которые получают одинаковое количество голосов, будет выбран один из этих потоков. Методология, используемая для определения того, какой из этих потоков не определен, может измениться в любое время без предварительного уведомления.

## Базовая настройка кластера

Здесь для примера описывается настройка простого трехузлового незащищенного кластера, состоящего из трех экземпляров ПС.

Для каждого экземпляра необходимо обновить определенные свойства в файле nifi.properties. В частности, свойства сети и кластеризации должны быть оценены для конкретной ситуации и соответствующим образом скорректированы.

Для всех трех экземпляров общие свойства кластера можно оставить с настройками по умолчанию. Однако, если вы измените эти параметры, они должны быть одинаковыми для всех экземпляров в кластере.

Для каждого узла необходимо настроить следующие минимальные свойства.

В разделе «Веб-свойства» укажите порт HTTP или HTTPS, на котором должен работать узел. Также подумайте, нужно ли вам устанавливать свойство хоста HTTP или HTTPS. Все узлы в кластере должны использовать одинаковые настройки протокола.

В разделе «Управление состоянием» установите для свойства nifi.state.management.provider.cluster идентификатор поставщика состояния кластера. Убедитесь, что поставщик состояния кластера настроен в файле state-management.xml.

В разделе «Свойства узла кластера» установите следующие параметры:

nifi.cluster.is.node - установите значение true.

nifi.cluster.node.address - задайте здесь полное имя хоста узла. Если оставить поле пустым, по умолчанию используется localhost.

nifi.cluster.node.protocol.port - установите для этого открытого порта больше 1024 (для всех меньших требуется root).

nifi.cluster.node.protocol.threads - количество потоков, которые следует использовать для связи с другими узлами в кластере. По умолчанию это свойство равно 10. Пул потоков используется для репликации запросов на все узлы, и в пуле потоков никогда не будет меньше, чем это количество потоков. Он будет увеличиваться по мере необходимости до максимального значения, установленного свойством nifi.cluster.node.protocol.max.threads.

nifi.cluster.node.protocol.max.threads - максимальное количество потоков, которые следует использовать для связи с другими узлами в кластере. По умолчанию это свойство равно 50. Пул потоков используется для запросов репликации ко всем узлам, а размер пула потоков будет иметь «основной» размер, который настраивается свойством nifi.cluster.node.protocol.threads. Однако при необходимости пул потоков увеличит количество активных потоков до предела, установленного этим свойством.

nifi.zookeeper.connect.string - строка подключения, необходимая для подключения к Apache ZooKeeper. Это список пар имя хоста: порт, разделенных запятыми. Например, localhost: 2181, localhost: 2182, localhost: 2183. Он должен содержать список всех экземпляров ZooKeeper в кворуме ZooKeeper.

nifi.zookeeper.root.node - корневой ZNode, который следует использовать в ZooKeeper. ZooKeeper предоставляет структуру, похожую на каталог, для хранения данных. Каждый «каталог» в этой структуре называется ZNode. Это обозначает корневой ZNode или «каталог», который следует использовать для хранения данных. Значение по умолчанию - / root. Это важно установить правильно, поскольку к какому кластеру пытается присоединиться экземпляр NiFi, зависит от того, к какому экземпляру ZooKeeper он подключается, и от указанного корневого узла ZooKeeper.

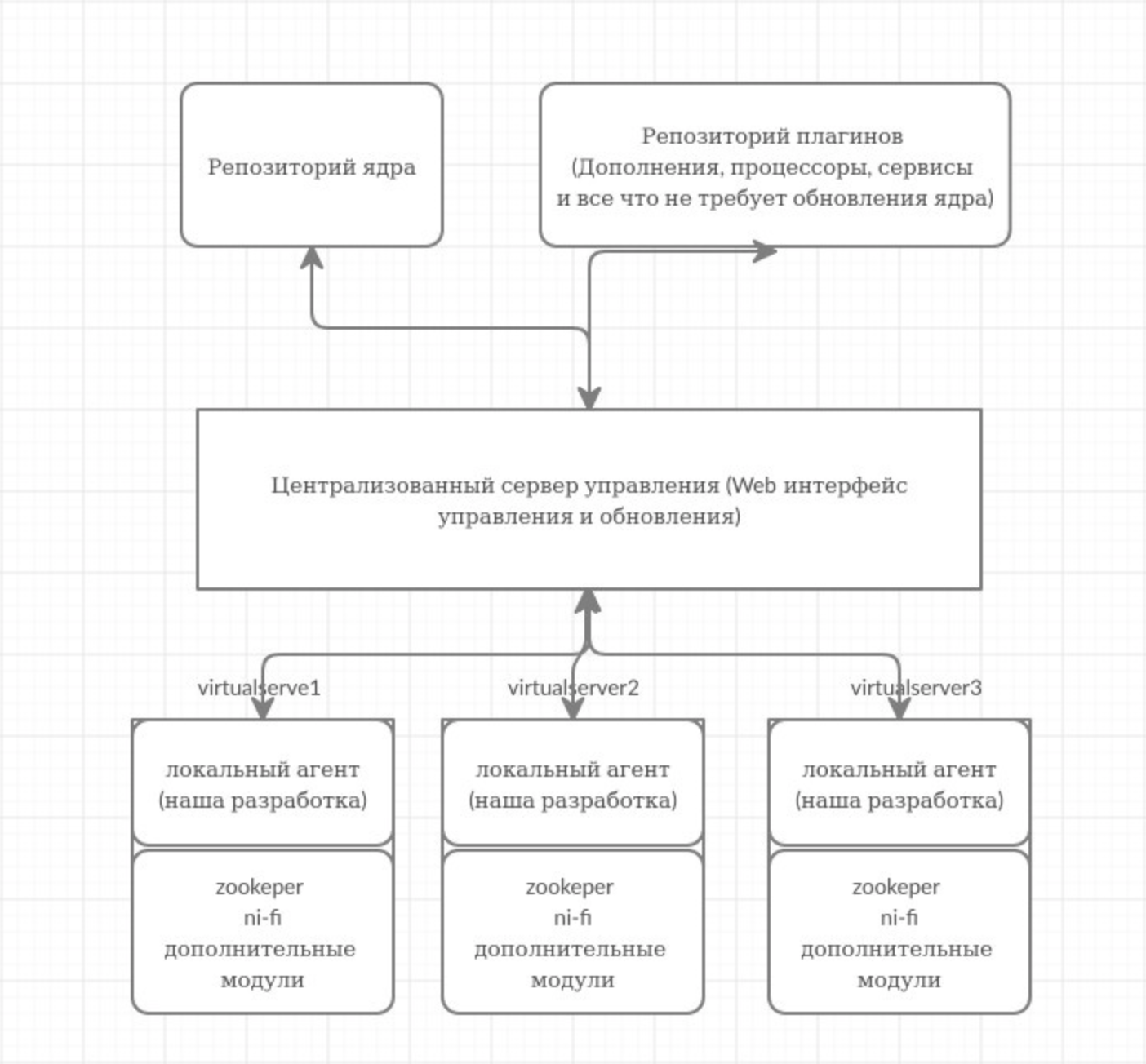
nifi.cluster.flow.election.max.wait.time - указывает время ожидания перед выбором потока в качестве «правильного» потока. Если количество проголосовавших узлов равно количеству, заданному свойством nifi.cluster.flow.election.max.candidates, кластер не будет ждать так долго. Значение по умолчанию - 5 минут. Обратите внимание, что отсчет времени начинается, как только подан первый голос.

nifi.cluster.flow.election.max.candidates - указывает количество узлов, необходимых в кластере, чтобы вызвать досрочный выбор потоков. Это позволяет узлам в кластере избежать длительного ожидания перед началом обработки, если мы достигнем хотя бы этого количества узлов в кластере.

Теперь можно запустить кластер. Не имеет значения, в каком порядке запускаются экземпляры.

# Обновление системы

В данном разделе описан процесс обновления ПС «Атом.Мост». При обновлении ПС «Атом.Мост» обновляться может как ядро, так процессоры и дополнения. При этом и ядро, и процессоры и дополнения являются обратно совместимыми после обновления. Схема обновления приведена на рисунке 16.



1. Схема обновления ПС «Атом.Мост»

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!** На всех узлах кластера должны быть одинаковые версии ПС «Атом.Мост». Запуск нескольких узлов ПС «Атом.Мост» на одной и той же хост-машине крайне нежелателен. |  |

## Предварительные шаги

### Сохранение пользовательских процессоров

Если вы написали какие-либо пользовательские NARs (процессоры, сервисы), сначала протестируйте их. Проверенные NAR могут быть сохранены во время обновления в отдельную директорию хранения нижеописанным образом.

1. Создайте второй каталог библиотеки, называемый custom\_lib.
2. Переместите свои пользовательские NARs в этот новый каталог lib.
3. Добавьте новую строку в список nifi.properties файл для указания этого нового каталога lib:

nifi.nar.library.directory=./lib

nifi.nar.library.directory.custom=/opt/configuration\_resources/custom\_lib

### Подготовка кластера

Подготовка кластера проходит по следующим этапам:

1. Остановите все процессоры, ответственные за прием новых данных.
2. Дождитесь пока ПС «Атом.Мост» обработает все текущие flowFile в потоке.
3. Завершите работу существующих экземпляров ПС «Атом.Мост».

Используйте конфигурационные файлы из существующей установки ИГЗИ:

<installation-directory>/conf

И обновите соответствующие свойства в новом развертывании NiFi согласно таблице 15.

1. Конфигурационные файлы

|  |  |
| --- | --- |
| Конфигурационный файл | Изменения |
| authorizers.xml | Если вы используете авторизатор "file-provider”” по умолчанию, убедитесь, что вы скопировали users.xml и authorizations.xml файлы из существующего в новый ПС «Атом.Мост».  Рекомендации по настройке рекомендуют создать отдельное расположение вне базового каталога ПС «Атом.Мост» для хранения таких конфигурационных файлов. Если вы храните эти файлы в отдельном каталоге, вам не нужно их перемещать. Вместо этого убедитесь, что новый ПС «Атом.Мост» указывает на те же файлы. Например: /opt/nifi/configuration-resources/ |
| bootstrap-notification-services.xml | Обновите значения на основе значений из существующего файла ПС «Атом.Мост». |
| bootstrap.conf | Используйте существующий ПС «Атом.Мост» bootstrap.conf-файл для обновления свойств в новом ПС «Атом.Мост». |
| logback.xml | Если вы добавили какие-либо пользовательские изменения ведения журнала в существующий ПС «Атом.Мост», внесите те же изменения в новый ПС «Атом.Мост». |
| login-identity-providers.xml | Если вы использовали значение поставщика в существующем ПС «Атом.Мост», скопируйте <provider>...</provider> конфигурация из существующего ПС «Атом.Мост». |
| nifi.properties | Используйте существующий файл для заполнения тех же свойств в новом ПС «Атом.Мост» nifi.properties файл.  Этот файл содержит большинство параметров конфигурации ПС «Атом.Мост», поэтому убедитесь, что вы скопировали значения правильно.  Если вы следовали рекомендациям ПС «Атом.Мост», то следующие свойства должны указывать на внешние каталоги за пределами базового пути установки ПС «Атом.Мост».  Если приведенные ниже свойства указывают на каталоги внутри базового пути установки ПС «Атом.Мост», необходимо скопировать целевые каталоги в новый ПС. Остановите существующую установку ПС «Атом.Мост», прежде чем делать это.  nifi.flow.configuration.file=  Если вы сохранили значение по умолчанию, (./conf/flow.xml.gz), копировать flow.xml.gz из существующего в новый каталог ПС «Атом.Мост».  Кроме того, вы можете скопировать его во внешнее расположение и обновить значение свойств.  nifi.flow.configuration.archive.dir=  То же самое относится и к приведенному выше, если вы хотите сохранить архивные копии flow.xml.gz.  nifi.database.directory=  Рекомендуется использовать внешнее расположение для каждого хранилища. Укажите новый ПС «Атом.Мост» в том же месте внешнего хранилища баз данных.  nifi.flowfile.repository.directory=  Рекомендуется использовать внешнее расположение для каждого хранилища. Укажите новый ПС «Атом.Мост» в том же месте внешнего хранилища flowfile.  Предупреждение: Вы можете испытать потеря данных, если репозитории flowfile недоступны для нового ПС «Атом.Мост».  nifi.content.repository.directory.default=  Рекомендуется использовать внешнее расположение для каждого хранилища. Укажите новый ПС «Атом.Мост» в том же месте внешнего хранилища контента.  В вашем существующем ПС «Атом.Мост» может быть определено несколько репозитариев контента. Убедитесь, что используются одни и те же имена свойств, и укажите соответствующие соответствующие местоположения репозитория контента. Например:  nifi.content.repository.directory.content1=  nifi.content.repository.directory.content2=  Предупреждение: Вы можете столкнуться с потерей данных, если хранилища контента недоступны для новых ПС «Атом.Мост».  Предупреждение: вы можете столкнуться с потерей данных, если имена свойств неверны или свойство указывает на неправильный репозиторий контента.  nifi.provenance.repository.directory.default=  Рекомендуется использовать внешнее расположение для каждого хранилища. Укажите новый ПС в том же месте внешнего хранилища провенанса.  У вашего существующего ПС «Атом.Мост» может быть определено несколько репозиториев провенанса. Убедитесь, что используются точно такие же имена свойств и указывают на соответствующие соответствующие местоположения репозиториев провенанса. Например:  nifi.provenance.repository.directory.provenance1=  nifi.provenance.repository.directory.provenance2=  Примечание: Вы можете быть не в состоянии запросить старые события, если репозиториев провенанса не перемещены правильно или свойства не обновлены правильно. |
| state-management.xml | Для значения "local-provider “проверьте расположение каталога” state/local".  Если вы сохранили расположение по умолчанию (./state/local), скопируйте все дерево каталогов в новый ПС «Атом.Мост». Существующий ПС «Атом.Мост» должен быть остановлен, если вы копируете этот каталог, потому что он может постоянно записывать данные в этот каталог во время работы.  Рекомендации по настройке рекомендуют переместить состояние во внешний каталог, например /opt/nifi/configuration-resources/, чтобы облегчить последующее обновление.  Для кластера ПС «Атом.Мост» поставщик кластера "ZooKeeper" строка подключения " должна использовать тот же внешний ZooKeeper, что и существующая установка ПС «Атом.Мост».  Для кластера ПС «Атом.Мост» убедитесь, что” поставщик кластера "ZooKeeper" корневой узел " точно соответствует значению, которое вы использовали в существующем ПС «Атом.Мост».  Если вы также настраиваете новый внешний ZooKeeper, обратитесь к разделу Zookeeper Migrator руководства системного администратора ПС «Атом.Мост» для получения инструкций о том, как переместить информацию ZooKeeper из одного кластера в другой и перенести права собственности на узел ZooKeeper. |
| Примечания:   1. Не копируйте конфигурационные файлы из старой версии ПС «Атом.Мост» в новую версию ПС «Атом.Мост». Более новые конфигурационные файлы могут ввести новые свойства, которые будут потеряны при этом. 2. Дважды проверьте все настроенные свойства на наличие опечаток. 3. Если вы шифруете свойства чувствительных компонентов в потоке данных с помощью ключа чувствительных свойств в nifi.properties файле, убедитесь, что тот же ключ используется при копировании поверх вашего flow.xml.gz. Если вам нужно изменить ключ, используйте процедуру миграции потока и инструмент ПС «Атом.Мост» Encrypt-Config для создания нового flow.xml.gz файла из существующего. | |

Запустите каждый из ваших новых экземпляров ПС «Атом.Мост». Проверьте:

1. Все ваши потоки данных вернулись в рабочее состояние. Некоторые процессоры могут иметь новые свойства, которые необходимо настроить, в этом случае они будут остановлены и помечены как “недопустимые”.
2. Все ваши службы контроллера и задачи отчетности снова запущены. Обратите внимание на службы контроллеров или задачи отчетности, помеченным как "недопустимые".
3. После подтверждения того, что ваши новые экземпляры ПС «Атом.Мост» стабильны и работают должным образом, старые экземпляры могут быть удалены.

Примечание – если исходный ПС «Атом.Мост» был настроен для запуска в качестве службы, обновите символические ссылки или сценарии службы, чтобы указать на новые исполняемые файлы версии ПС «Атом.Мост».

## Общие шаги

Обновление процессоров и сервисов:

1. Останавливаем входные процессоры на получение данных потоков;
2. Дожидаемся завершения обработки flowFile;
3. Заменяем необходимые nar файлы на нужные;
4. Запускаем узлы кластера;
5. Проверяем правильность функционирования кластера и интеграционных потоков.

Обновление ядра:

1. Останавливаем входные процессоры на получение данных потоков;
2. Дожидаемся завершения обработки flowFile;
3. Разархивируем новую версию ПС «Атом.Мост» на каждом узле;
4. Заменяем необходимые NAR-файлы на пользовательские из старой версии в новой;
5. Обновляем конфигурационные файлы в новой версии;
6. Запускаем узлы кластера;
7. Проверяем правильность функционирования кластера и интеграционных потоков.

# Контроль целостности

В ПС «Атом.Мост» для контроля целостности используется модуль DFSumChecker для проверки хэш-сумм запускаемого программного обеспечения.

DFSumChecker состоит из двух утилит:

1. Генерация хэш-сумм и их подпись закрытым ключом (DFSumGenerator);
2. Проверка подписи и проверка на соответствие хэш-суммам (DFSumChecker).

## Утилита генерации хэш-сумм DFSumGenerator

Запуск программы для генерации файла с хэш-суммами производится командой:

java -jar ./DFSumGenerator.jar --generate /path/to/directory/with/check/software

где:

--generate – указывает на режим генерации файла с хеш-суммами;

/path/to/directory/with/check/software – абсолютный, либо относительный путь для ПО, где требуется с генерировать файл с хеш-суммами.

Принцип работы утилиты заключается в рекурсивном обходе всего дерева начиная с /path/to/directory/with/check/software и подсчета хеш-сумм для каждого из файлов.

В результате работы создаются неизменяемый файл checksums.chk, и файл с ЭЦП checksums.sign к файлу checksums.chk.

Формат файла checksums.chk представляет собой текстовый файл, в котором каждая строчка является парой «ключ-значение», разделенной пробелом.

|  |  |  |
| --- | --- | --- |
|  | **ВАЖНО!**  В состав jar-архива DFSumGenerator входит хранилище ключей PKS12 с закрытым ключом сертификатом и открытым ключом.  В состав jar-архива DFSumChecker входит хранилище ключей PKS12 с сертификатом и открытым ключом.  Срок действия ключей 1 год. Если происходит изменения ключей, то необходимо провести переподписание файлов checksums.chk. |  |

## Утилита проверки хэш-сумм DFSumChecker

Запуск программы проверки на целостность производится командой:

java -jar ./DFSumChecker.jar --checkFile /path/to/file/with/checksums --checkDir /path/to/directory/with/check/software

где:

--checkFile /path/to/file/with/checksums – путь до файла с хэш-суммами

--checkDir /path/to/directory/with/check/software – абсолютный, либо относительный путь до проверяемого ПО.

Сначала проверяется ЭЦП файла /path/to/file/with/checksums, если все в порядке начинается проверка файлов на соответствия хэш-суммам.

# Правила приёмки

## Общие положения

* + 1. Для проверки правильности комплектования и функционирования изделия должны проводиться предъявительские и приемо-сдаточные испытания.
    2. Испытания проводятся силами и средствами предприятия-изготовителя на оборудовании, на которое имеется документация, подтверждающая его исправность.
    3. Основными документами при испытаниях и приемке являются:
* комплект ЭД;
* RU.64509942.00213-01 91 01 Технические условия.

Испытания изделия должны проводиться при нормальных климатических условиях:

* температуре от плюс 15° С до плюс 35° С;
* относительной влажности воздуха от 45% до 80%;
* атмосферном давлении от 84,0 до 106,7 кПа (от 630 до 800 мм рт.ст.).
  + 1. Изделие, предъявляемое на испытания, должно быть укомплектовано в соответствии с документом   
       RU.64509942.00213-01 30 01 Формуляр.
    2. Для проведения испытаний и приемки изделия предприятие-изготовитель предоставляет комплект документации, необходимые инструменты, материалы, средства измерения, испытательное и вспомогательное оборудование, описания и инструкции по пользованию ими.
    3. Изделие, предъявляемое на испытания и (или) приемку, должно быть полностью укомплектовано в соответствии с формуляром и договора на поставку.
    4. В процессе проведения испытаний изделия, по согласованию с представителем заказчика, разрешается изменять последовательность проверок и совмещать проверки по отдельным пунктам.

В процессе проведения испытаний разрешается делать перерывы между отдельными видами испытаний по согласованию с представителем заказчика сроком не более чем на 2 суток.

* + 1. Результаты испытаний считают положительными, а изделие выдержавшим испытания, если оно испытано в полном объеме и последовательности, которые установлены в технических условиях для данной категории испытаний, и соответствует всем требованиям технический условий, проверяемым при этих испытаниях.
    2. Результаты испытаний считают отрицательными, а изделие не выдержавшим испытания, если установлено несоответствие изделия хотя бы одному требованию технических условий для данной категории испытаний.
    3. Испытания и приемку изделия приостанавливают в следующих случаях, если:
* изделие, предъявлявшееся дважды на приемку, не выдержало испытаний оба раза;
* экземпляры изделий, последовательно один за другим первично предъявлявшиеся на испытания; не выдержали испытаний и были окончательно забракованы (без права их повторного предъявления на приемку) по результатам каждого из двух последовательно проведенных первичных испытаний.
  + 1. Приемка изделия может приостанавливаться также в других случаях по усмотрению ОТК и/или представителем заказчика, что должно быть отражено в документации в соответствии с системой качества, действующей на предприятии-изготовителе.
    2. Принятым считается изделие, которое выдержало приемосдаточные испытания, при соблюдении действия положительных результатов периодических испытаний, а также укомплектовано, промаркировано и упаковано в соответствии с пп. 1.3-1.5 ТУ.

## Предъявительские испытания

* + 1. Предъявительские испытания проводит ОТК с целью контроля готовых изделий на соответствие требованиям ТУ и определения готовности их для предъявления представителю заказчика.
    2. Предъявительские испытания изделия проводят в объеме, предусмотренном в таблице 3.
    3. Предъявление изделия на предъявительские испытания должно осуществляться в порядке, действующем на предприятии-изготовителе.
    4. Результаты предъявительских испытаний оформляются протоколом (ГОСТ 15.309-98, приложение В), который подписывается лицами, непосредственно проводившими испытания.
    5. Решение о возможности последующего использования изделия, не выдержавшей повторных предъявительских испытаний, принимает руководитель предприятия-изготовителя и начальник ОТК.
    6. После получения положительных результатов предварительных испытаний в формуляре на принятое изделие проставляется штамп ОТК, ставятся подпись и печать, свидетельствующие о приемке и годности изделия.
    7. Изделие, не выдержавшее предъявительских испытаний, может быть повторно подвергнуто испытаниям только после проведения мероприятий по устранению дефектов и причин их возникновения.
    8. Повторные предъявительские испытания проводят в объеме проверок, установленных для предварительных испытаний.

В зависимости от характера дефектов, выявленных при первичных испытаниях, в отдельных технически обоснованных случаях повторные предъявительские испытания могут проводить только в объеме тех проверок, по которым выявлено несоответствие изделия установленным требованиям, которые могли повлиять на возникновение дефектов и по которым испытания не проводились.

* + 1. Окончательно забракованное по результатам предварительных испытаний изделие изолируется от годных.

Решение об использовании окончательно забракованного изделия принимает предприятие-изготовитель. Принимаемые решения должны отвечать требованиям законодательства об охране окружающей среды, о безопасности и защите государственной тайны.

## Приёмо-сдаточные испытания

* + 1. Приемо-сдаточные испытания проводит представитель заказчика в присутствии представителя ОТК проверкой каждого экземпляра изделия в объеме и последовательности, приведенными в таблице 16 на соответствие требованиям, указанным в ТУ на изделие.

Приемо-сдаточные испытания проводят на изделии, выдержавшем предъявительские испытания.

1. Последовательность приёмо-сдаточных испытаний

| Наименование испытаний | Пункты ТУ | | Примечание |
| --- | --- | --- | --- |
| технических требований | методов испытаний |
| Проверка требований к комплектности | 1.6 | 3.1 |  |
| Проверка требований к ЭД | 1.1 | 3.2 |  |
| Правила требований к материалам | 1.4 | 3.3 |  |
| Проверка требований к маркировке | 1.7 | 3.4 |  |
| Проверка записи изделия на ЭН | 1.5 | 3.5 |  |
| Проверка требований к упаковке | 1.8 | 3.6 |  |

* + 1. Предъявление изделия на приемо-сдаточные испытания производит ОТК в соответствии с ГОСТ В 15.307-77.
    2. Результаты испытаний представитель заказчика должен оформить протоколом по форме в соответствии с ГОСТ 15.309-98. На основании протокола представитель заказчика в извещении составляет заключение о соответствии изделия требованиям ТУ.
    3. Результаты испытаний считают положительными, а изделие выдержавшим испытания, если изделие испытано в полном объеме и последовательности, которые установлены в ТУ, и соответствует всем требованиям.
    4. Если в процессе испытаний изделия будет выявлено несоответствие хотя бы по одному требованию ТУ, то проверяемое изделие считают не выдержавшим испытания и возвращают ОТК для выявления причин дефектов, а также для проведения мероприятий по их устранению и повторного предъявления. Причины несоответствия требованиям ТУ и принятые предприятием меры отражают в акте об анализе и устранении дефектов и их причин.
    5. Возвращенное представителем заказчика изделие после устранения дефектов, повторной проверки предприятием-изготовителем, при положительных результатах повторных предъявительских испытаний и приемки ОТК повторно предъявляется представителю заказчика.

Повторное предъявление осуществляется извещением, на котором должна быть пометка «Вторичное». Вторичное извещение для приемо-сдаточных испытаний и приемки изделия должно быть подписано руководителем предприятия (главным инженером) и начальником ОТК. К извещению должен быть приложен акт об анализе дефектов, их устранении с указанием причин дефектов и мер, принятых по устранению.

* + 1. Повторные испытания изделия, возвращенного представителем заказчика, проводятся в полном объеме, установленном для приемо-сдаточных испытаний.

В зависимости от результата анализа дефектов, обнаруженных при испытаниях, испытания возвращенных экземпляров изделия допускается по согласованию с представителем заказчика проводить только по требованиям, которым изделие не соответствовало, которые могут повлиять на возникновение дефектов и по которым испытания не проводились.

* + 1. Изделие, не выдержавшее повторных испытаний, окончательно забраковывается и изолируется от годных.
    2. Испытания и приемку изделия, изготовленного по той же технологической и нормативной документации, что и изделие, не выдержавшее повторных испытаний, или изделие, окончательно забракованное при двух последовательных первичных предъявлениях, приостанавливают.
    3. Решение о возобновлении испытаний и приемки принимает Заказчик после проведения согласованных с представительством заказчика мероприятий по устранению причин, вызвавших приостановку испытаний и приемки, и оформления соответствующего документа, согласованного с представительством заказчика.
    4. Решение о дальнейшем использовании окончательно забракованного изделия принимает Заказчик или по его указанию – представитель заказчика и руководитель (главный инженер) предприятия-изготовителя.
    5. Принятыми считают изделия, которые выдержали приемо-сдаточные испытания, укомплектованы, упакованы в соответствии с требованиями, указанными в ТУ, и опломбированы ОТК и представителем заказчика.

Принятые изделия подлежат отгрузке или сдаче на ответственное хранение предприятию-изготовителю.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Изм. | Номера листов (страниц) | | | | Всего листов  (страниц)  в доку-менте | Номер  доку-  мента | Входящий  номер сопроводи-тельного документа и дата | Подпись | Дата |
| изменен-ных | заменен-ных | новых | аннулиро-ванных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |