

# Технические условия

Программа «NORA — многопротокольный реестр артефактов»

Версия документа: 2.1 · Дата: 2026-06-05 · Правообладатель: ООО «ТАИАРС» · Версия программы: 1.0.0

## 1. Наименование и обозначение

---

Полное наименование: NORA — многопротокольный реестр артефактов.

Краткое наименование: NORA.

Обозначение: nora-registry.

## 2. Назначение

---

Программа предназначена для хранения, кэширования и распространения программных компонентов (артефактов), используемых при разработке, сборке и развёртывании программного обеспечения.

### 2.1. Область применения

- Организация внутренних репозиториев программных компонентов.
- Проксирование и кэширование общедоступных репозиториев (npmjs.org, PyPI, Maven Central, Docker Hub, crates.io, rubygems.org, pkg.go.dev и др.).
- Обеспечение доступности зависимостей в изолированных средах без доступа к сети Интернет (air-gapped).
- Контроль целостности и безопасности цепочки поставки программного обеспечения.
- Курирование зависимостей: блокировка, разрешение и изоляция пространств имён.

### 2.2. Класс программного обеспечения

Серверное и связующее программное обеспечение. Программа представляет собой серверное приложение, предоставляющее сетевые сервисы хранения и распространения программных компонентов другим компонентам инфраструктуры разработки.

Класс программного обеспечения по классификатору, утверждённому приказом Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 22.09.2020 № 486:

- Основной класс: **02.06 «Серверное и связующее программное обеспечение»**.

Код продукции по ОКПД2: **62.01.29** — оригиналы программного обеспечения прочие.

### 2.3. Хранение данных и размещение

В режиме прокси-кэша при первом обращении программа загружает артефакт из внешнего репозитория и **сохраняет (кэширует) его в локальное хранилище на файловой системе сервера**; последующие запросы к тому же артефакту обслуживаются непосредственно из локальной файловой системы без повторного обращения к внешнему источнику. В режиме хранилища (hosted) опубликованные пользователями артефакты также записываются в локальное хранилище. Помимо локальной файловой системы поддерживается S3-совместимое объектное хранилище. Все артефакты, метаданные и журналы аудита размещаются на технических средствах, находящихся на территории Российской Федерации.

## 3. Функциональные характеристики

### 3.1. Поддерживаемые протоколы

Протокол	Стандарт	Назначение
Docker / OCI	OCI Distribution Spec v1.0	Контейнерные образы
Helm OCI	OCI Distribution Spec v1.0	Helm-чарты (Kubernetes)
npm	npm Registry API	Библиотеки JavaScript / TypeScript
Maven	Maven Repository Layout	Библиотеки Java / Kotlin
PyPI	PEP 503 (Simple API)	Библиотеки Python
Cargo	Cargo Alternative Registry Protocol	Библиотеки Rust
Go	Go Module Proxy Protocol (GOPROXY)	Модули Go
NuGet	NuGet V3 API	Библиотеки .NET
RubyGems	RubyGems API	Библиотеки Ruby
Terraform	Terraform Provider Registry Protocol	Модули и провайдеры Terraform
Ansible Galaxy	Ansible Galaxy API	Коллекции и роли Ansible
Pub	Pub Repository Spec	Библиотеки Dart / Flutter
Conan	Conan v2 API	Библиотеки C / C++
APT	Debian Repository Format	Пакеты deb (Debian, Ubuntu)
YUM	RPM Repository Format (repomd)	Пакеты rpm (RHEL, CentOS, ALT Linux)
Raw	HTTP PUT/GET	Произвольные файлы

### 3.2. Режимы работы

- **Хранилище (hosted):** приём и хранение артефактов, опубликованных пользователями.
- **Прокси-кэш (proxy):** прозрачное проксирование запросов к внешним репозиториям с сохранением (кэшированием) полученных артефактов в локальное хранилище на файловой системе сервера.
- **Комбинированный:** одновременная работа в режимах хранилища и прокси-кэша (поиск сначала в локальном хранилище, затем во внешнем репозитории).
- **Виртуальный (group):** объединение нескольких репозиторий в единую точку доступа.

### 3.3. Управление доступом

- Аутентификация на основе htpasswd (алгоритм Argon2id).
- Единый вход через OpenID Connect (OIDC).
- Ролевая модель: read (чтение), write (чтение и запись), admin (полный доступ).
- Токены доступа с ограниченным сроком действия.

### 3.4. Безопасность и курирование

- Контроль целостности кэшированных артефактов (SHA-256).
- Защита от обхода каталогов (path traversal) при публикации.
- Проверка соответствия имени пакета в URL и теле запроса.
- Иммутабельность опубликованных версий.
- Аудит всех операций в формате JSON Lines.
- Поддержка TLS при размещении за обратным прокси-сервером.
- Механизм курирования: списки блокировки (blocklist), списки разрешений (allowlist), изоляция пространств имён (namespace isolation).
- Карантин новых пакетов (min-release-age).
- Автоматический выключатель (circuit breaker) при недоступности внешних реестров.

### 3.5. Эксплуатация

- Предварительное кэширование зависимостей (nora mirror) по файлам фиксации версий (lockfile).
- Сборка мусора (nora gc) — удаление осиротевших блобов.
- Резервное копирование и восстановление (nora backup, nora restore).
- Миграция между локальным хранилищем и S3-совместимым объектным хранилищем.
- Мониторинг: эндпоинты /health, /ready, /metrics (формат Prometheus), готовые дашборды Grafana.
- Политики жизненного цикла артефактов (retention, cleanup, max\_age, max\_versions).
- Миграция из Sonatype Nexus, JFrog Artifactory, GitLab Package Registry.
- Горячая перезагрузка конфигурации без перезапуска сервиса (hot reload).
- Поточковая загрузка артефактов (streaming uploads) для снижения потребления памяти.
- Веб-интерфейс для просмотра содержимого реестра (русский и английский языки).
- Документация API в формате OpenAPI 3.0.

## 4. Технические характеристики

### 4.1. Среда исполнения

<b>Язык реализации</b>	Rust
<b>Формат поставки</b>	Единый исполняемый файл, Docker-образ, Helm-чарт, deb- и rpm-пакеты
<b>Поддерживаемые ОС</b>	Astra Linux, РЕД ОС, ALT Linux, Debian, Red Hat Enterprise Linux, Ubuntu (ядро Linux 4.15 и новее)
<b>Архитектуры</b>	x86_64 (amd64), aarch64 (arm64)
<b>Контейнеризация</b>	Docker-образ на базе scratch, Helm-чарт для Kubernetes
<b>Системная интеграция</b>	systemd (файл сервиса в комплекте), deb/rpm-пакеты

## 4.2. Хранение данных

<b>Локальное хранилище</b>	Файловая система (ext4, XFS, ZFS)
<b>Объектное хранилище</b>	S3-совместимое API (MinIO, Yandex Object Storage, Selectel S3)
<b>Структура</b>	Иерархическая: {protocol}/{package}/{artifact}
<b>Аудит</b>	Append-only JSONL файл

## 4.3. Конфигурация

<b>Источник</b>	<b>Приоритет</b>
Переменные окружения (NORA_*)	Высший
Файл config.toml	Средний
Значения по умолчанию	Низший

## 4.4. Производительность

<b>Время запуска</b>	< 3 с
<b>Обслуживание из кэша</b>	< 2 мс (метаданные), < 10 мс (артефакты до 1 МБ)
<b>Параллельная обработка</b>	Асинхронный ввод-вывод (tokio runtime)
<b>Ограничение частоты</b>	Настраиваемое (по умолчанию 100 запросов/сек)
<b>Потребление памяти</b>	< 100 МБ при типичной нагрузке
<b>Размер бинарного файла</b>	< 23 МБ

## 5. Лицензирование

<b>Компонент</b>	<b>Лицензия</b>
NORA	MIT License

Полный перечень лицензий включённых библиотек приведён в файле SBOM (формат CycloneDX).

## 6. Комплектность

<b>Компонент</b>	<b>Описание</b>
nora	Исполняемый файл
nora.service	Файл systemd-сервиса
nora.env.example	Шаблон конфигурации
install.sh	Скрипт установки
nora.cdx.json	SBOM в формате CycloneDX

Компонент	Описание
nora.spdx.json	SBOM в формате SPDX
Руководство администратора	Настоящий комплект документации
Руководство пользователя	Настоящий комплект документации
Технические условия	Настоящий документ

## 7. Процессы поддержания жизненного цикла

---

Поддержание жизненного цикла программы обеспечивается правообладателем — ООО «ТАИАРС» — силами штатных сотрудников.

- **Гарантийное обслуживание, техническая поддержка и модернизация:** осуществляются силами штатных сотрудников правообладателя (2 человека) на территории Российской Федерации.
- **Устранение неисправностей:** приём обращений по каналам поддержки, диагностика, выпуск исправлений и обновлений.
- **Совершенствование:** развитие функциональности по плану развития продукта и обращениям пользователей.

## 8. Контактная информация

---

Правообладатель: ООО «ТАИАРС».

Сайт продукта: <https://getnora.io>

Документация: <https://getnora.io/docs>

Исходный код: <https://git.getnora.io/nora/nora>

Поддержка: <https://t.me/getnora>