УТВЕРЖДАЮ

Генеральный директор ООО «Азон»

_____ / Е.А. Гуцало

«17» августа 2026 г.

МП

# Описание программного интерфейса и форматов данных

программного обеспечения

«Чистый интернет»

**МОСКВА, 2026**

## 1. Общие сведения

1.1. Настоящий документ описывает программный интерфейс и форматы данных программного обеспечения «Чистый интернет» (далее — Комплекс) и разработан в соответствии с п. 10.5 Технического задания на разработку программного обеспечения «Чистый интернет».

1.2. Документ предназначен для специалистов, выполняющих интеграцию смежных систем с Комплексом, а также для администраторов Комплекса.

1.3. Программный интерфейс построен по архитектурному стилю REST. Обмен выполняется по протоколу HTTP поверх защищённого соединения TLS. Тело запросов и ответов, за исключением явно указанных случаев, передаётся в формате JSON в кодировке UTF-8.

1.4. Интерактивная спецификация в формате OpenAPI публикуется Комплексом по пути `/swagger/` и входит в состав поставки в каталогах документации соответствующих служб.

## 2. Схема доступа и версионирование

2.1. Единой точкой входа для внешних потребителей является шлюз программного интерфейса, публикуемый обратным прокси-сервером.

2.2. Базовый адрес методов: `https://<адрес Комплекса>/api/v1`.

2.3. Версия интерфейса указывается в пути запроса. Текущая версия — `v1`. При выпуске несовместимой версии интерфейса она будет опубликована по отдельному префиксу пути; версия `v1` при этом сохраняется на согласованный период.

2.4. Внутренние службы Комплекса за пределы контура не публикуются. Обращение к ним из смежных систем не предусмотрено.

2.5. Для интеграции смежных систем предназначены методы, перечисленные в разделе 6 (отчёты и перечень блокировок), поскольку они содержат явное указание версии схемы данных.

## 3. Аутентификация и авторизация

### 3.1. Механизм

3.1.1. Аутентификация выполняется службой идентификации Комплекса. Клиенту выдаётся маркер доступа в формате JWT, подписанный по алгоритму RS256, и маркер обновления.

3.1.2. Маркер доступа передаётся в каждом защищённом запросе в заголовке:

```
Authorization: Bearer <маркер доступа>
```

3.1.3. Маркер обновления передаётся в заголовке `X-Refresh-Token`.

3.1.4. Подлинность маркера доступа проверяется шлюзом программного интерфейса и службой аналитики по открытому ключу службы идентификации. При отсутствии, истечении срока действия либо неверной подписи маркера возвращается код 401.

### 3.2. Методы аутентификации

| Метод | Путь | Тело запроса | Ответ |
|---|---|---|---|
| POST | `/api/v1/auth/sign-in` | `{"login": "...", "password": "..."}` | Маркеры доступа |
| POST | `/api/v1/auth/refresh` | — (заголовок `X-Refresh-Token`) | Новые маркеры доступа |
| POST | `/api/v1/auth/sign-out` | — (заголовки `Authorization` и `X-Refresh-Token`) | Сообщение о завершении сеанса |

3.2.1. Структура ответа методов выдачи маркеров:

```
{
  "access_token": "строка",
  "refresh_token": "строка",
  "expires_in": 86400,
  "grafana_session": "строка",
  "grafana_session_expiry": "строка"
}
```

3.2.2. Поле `expires_in` содержит срок действия маркера доступа в секундах. Поля сеанса службы мониторинга используются для сквозного входа в служебные панели.

3.2.3. Частота обращений к методу входа ограничена. Значение по умолчанию — 20 попыток в минуту с одного сетевого адреса; при превышении возвращается код 429.

### 3.3. Разграничение доступа

3.3.1. Полномочия определяются ролями, содержащимися в маркере доступа.

| Роль в маркере | Смысл | Область данных |
|---|---|---|
| `SA` | Системный администратор | Все узлы |
| `CA-<идентификатор узла>` | Администратор узла | Данные закреплённого узла |
| `CO-<идентификатор узла>` | Оператор узла | Данные закреплённого узла |

3.3.2. Для роли администратора узла и оператора служба аналитики принудительно ограничивает выборку идентификатором закреплённого узла. Значение параметра `hostname`, отличное от закреплённого узла, не расширяет область видимости.

3.3.3. Методы работы с нейросетевыми выявлениями, изменения решения по ресурсу и выгрузки перечня блокировок доступны только роли `SA`. При обращении с иной ролью возвращается код 403.

3.3.4. Методы управления параметрами автоматического решения доступны ролям `SA` и `CA`.

### 3.4. Межсайтовые запросы

3.4.1. Перечень разрешённых источников межсайтовых запросов задаётся параметром конфигурации шлюза программного интерфейса.

3.4.2. Разрешённые методы: GET, POST, PUT, PATCH, DELETE, OPTIONS, HEAD. Разрешённые заголовки: `Origin`, `Content-Type`, `Accept`, `Authorization`, `X-Refresh-Token`, `X-Request-ID`. Передача учётных данных в межсайтовых запросах разрешена.

### 3.5. Служебные ключи

3.5.1. Для синхронного обращения к службе извлечения содержимого применяется служебный ключ, передаваемый в заголовке `X-API-Key`. Значение ключа задаётся параметром конфигурации соответствующей службы. При незаданном ключе метод недоступен и возвращает код 503. Метод является внутренним и за пределы контура не публикуется.

3.5.2. Межсервисная запись в журнал бизнес-событий сопровождается заголовками `X-Caller-Service` и `X-Request-ID`.

## 4. Общие соглашения

### 4.1. Параметры отбора

4.1.1. Периоды задаются параметрами `from` и `to` в формате RFC 3339. При их отсутствии применяются значения по умолчанию, указанные для каждого метода.

4.1.2. Постраничный вывод задаётся параметрами `page` (номер страницы, с единицы) и `limit` либо `count` (число записей на странице) в зависимости от метода.

4.1.3. Порядок вывода задаётся параметрами `order_by` (имя поля) и `order_dir` (`asc` либо `desc`).

4.1.4. Отбор по узлу межсетевого экрана задаётся параметром `hostname`.

4.1.5. Поиск по текстовому представлению записи задаётся параметром `search`.

### 4.2. Формат ошибки

4.2.1. Тело ответа при ошибке имеет вид:

```
{"error": "текстовое описание ошибки"}
```

4.2.2. Применяемые коды состояния:

| Код | Условие возникновения |
|---|---|
| 400 | Недопустимые параметры запроса либо тело запроса |
| 401 | Отсутствует, повреждён либо просрочен маркер доступа |
| 403 | Полномочий роли недостаточно для выполнения операции |
| 404 | Запрашиваемый объект не найден |
| 409 | Объект с такими реквизитами уже существует |
| 413 | Превышен допустимый размер передаваемого содержимого |
| 429 | Превышено ограничение частоты обращений |
| 500 | Внутренняя ошибка обработки |
| 502 | Нижестоящая служба недоступна либо вернула некорректный ответ |
| 503 | Функция не сконфигурирована |

### 4.3. Трассировка

4.3.1. Заголовок `X-Request-ID` передаётся клиентом либо формируется шлюзом и сопровождает запрос по всей цепочке обработки. Значение фиксируется в журналах служб и применяется при разборе обращений.

## 5. Перечень методов

### 5.1. Справочники

| Метод | Путь | Параметры | Ответ |
|---|---|---|---|
| GET | `/api/v1/analytics/categories` | — | Массив наименований категорий |
| GET | `/api/v1/analytics/devices` | — | Массив имён узлов межсетевого экрана |

### 5.2. Показатели и панели

| Метод | Путь | Параметры (значения по умолчанию) |
|---|---|---|
| GET | `/api/v1/analytics/dashboards/traffic` | `from`, `to`, `count` (20), `hostname` |
| GET | `/api/v1/analytics/dashboards/requests` | `from`, `to`, `request_type` (`pending`), `count` (10), `hostname` |
| GET | `/api/v1/analytics/dashboards/top-categories` | `from`, `to`, `count` (5), `type`, `hostname` |
| GET | `/api/v1/analytics/dashboards/top-unresolved_detections` | `from`, `to`, `count` (5), `hostname` |
| GET | `/api/v1/analytics/dashboards/anomalies` | `from`, `to`, `hostname` |
| GET | `/api/v1/analytics/dashboards/devices` | `from`, `to`, `count` (20) |
| GET | `/api/v1/analytics/dashboards/proh-activity` | `year`, `hostname` |

5.2.1. Допустимые значения параметра `request_type`: `allowed`, `blocked`, `before_block`, `pending`.

5.2.2. Допустимые значения параметра `type`: `Разрешен`, `Заблокирован`, `VPN`.

### 5.3. Сессии

| Метод | Путь | Параметры (значения по умолчанию) |
|---|---|---|
| GET | `/api/v1/analytics/sessions` | `from`, `to`, `page` (1), `count` (10), `order_by` (`datetime_utc`), `order_dir` (`desc`), `hostname`, `category`, `type`, `status`, `search` |

5.3.1. Допустимые значения параметра `status`: `Разрешен`, `Запрещен`, `Ожидает`, `Аномалия`.

5.3.2. Ответ содержит массив записей и сведения о постраничном выводе: общее число записей и число страниц.

### 5.4. Выявления

| Метод | Путь | Параметры / тело | | |
|---|---|---|---|---|
| GET | `/api/v1/analytics/detections` | `from`, `to`, `page` (1), `limit` (10), | | |

| | | | | |
|---|---|---|---|---|
| | | order_by (categorized_at), order_dir (desc), hostname, status, category, action, search | | |
| GET | /api/v1/analytics/detections_v2 | Те же параметры; ответ дополнен перечнем категорий с оценками | | |
| GET | /api/v1/analytics/detections/stat | from, to, hostname, category | | |
| PATCH | /api/v1/analytics/detections/act | Тело: `{"action": "allow" \ | "deny" \ | "unsolved", "path": "<домен>"}` |

5.4.1. Допустимые значения параметра action при отборе: Разрешено, Заблокировано, Не решено.

5.4.2. Метод изменения решения доступен только роли SA. Значение unsolved возвращает выявление в состояние нерешённого.

5.4.3. Ответ метода получения показателей выявлений:

```
{
  "detected": 0,
  "accepted": 0,
  "denied": 0,
  "unresolved": 0,
  "ml_ngfw_mismatch": 0
}
```

### 5.5. Управление учётными записями и ролями

| Метод | Путь | Назначение |
|---|---|---|
| GET | /api/v1/users | Перечень учётных записей; параметры page (1), limit (10), role, contextID, search |
| POST | /api/v1/users | Создание учётной записи; ответ 201 |
| GET | /api/v1/users/{userId} | Сведения об учётной записи |
| PUT | /api/v1/users/{userId} | Изменение сведений об учётной записи |
| DELETE | /api/v1/users/{userId} | Удаление учётной записи |
| PUT | /api/v1/users/{userId}/pass/otp | Назначение временного пароля |
| PUT | /api/v1/users/{userId}/roles | Назначение роли |
| DELETE | /api/v1/users/{userId}/roles | Отзыв роли |
| GET | /api/v1/users/profile | Сведения о текущем пользователе |
| PUT | /api/v1/users/profile/pass | Смена собственного пароля |
| PUT | /api/v1/users/profile/avatar | Загрузка изображения профиля |
| DELETE | /api/v1/users/profile/avatar | Удаление изображения профиля |
| GET | /api/v1/users/count | Общее число учётных записей |
| GET | /api/v1/users/count-by-role | Число учётных записей в разрезе ролей |
| GET | /api/v1/roles | Перечень ролей; параметры page (1), limit (10), search |
| GET | /api/v1/roles/count | Общее число ролей |

### 5.6. Журнал бизнес-событий

| Метод | Путь | Назначение |
|---|---|---|
| GET | `/api/v1/blog/logs` | Записи журнала; параметры `page` (1), `limit` (10), `role`, `contextID`, `search` |
| POST | `/api/v1/blog/add` | Регистрация события; применяется службами Комплекса |

### 5.7. Параметры автоматического решения

| Метод | Путь | Тело / ответ | |
|---|---|---|---|
| GET | `/api/v1/settings/auto-decision` | `{"mode": "on" \ | "off"}` |
| PUT | `/api/v1/settings/auto-decision` | `{"mode": "on" \ | "off"}` |

5.7.1. Методы доступны ролям `SA` и `CA`.

### 5.8. Служебные методы

| Метод | Путь | Назначение | Аутентификация |
|---|---|---|---|
| GET | `/api/v1/healthcheck` | Проверка работоспособности; ответ `{"message": "ready"}` | Не требуется |
| GET | `/api/v1/version` | Версия сборки | Не требуется |
| GET | `/metrics` | Метрики в формате системы сбора метрик | Не требуется; публикуется во внутреннем контуре |
| GET | `/swagger/` | Интерактивная спецификация | Не требуется |

5.8.1. Собственные методы проверки работоспособности предоставляют также службы аналитики, ETL, управления пользователями, журнала бизнес-событий, извлечения содержимого и нейросетевой классификации. Они применяются средой контейнеризации и средствами мониторинга и за пределы контура не публикуются.

## 6. Методы для интеграции смежных систем

### 6.1. Отчёт в машиночитаемом виде

| Метод | Путь | Параметры |
|---|---|---|
| GET | `/api/v1/reports` | `from`, `to` (по умолчанию — последние сутки) |
| GET | `/api/v1/reports/{hostname}` | `from`, `to` |

6.1.1. Указанные методы возвращают документ отчёта с явным указанием версии схемы данных и предназначены для интеграции.

6.1.2. Методы `/api/v1/analytics/reports` и `/api/v1/analytics/reports/{hostname}` предназначены для нужд веб-интерфейса и не гарантируют наличия поля версии схемы. Для интеграции их применять не следует.

6.1.3. Структура документа отчёта:

```
{
  "schema_version": "1.0",
  "from": "2026-09-10T12:00:00Z",
  "to": "2026-09-11T12:00:00Z",
  "narrative": "аналитическая справка",
  "main_activity_page": {
    "top_categories": [
      {"category": "...",
       "stat": {"all": 0, "before_block": 0, "pending": 0, "after_block": 0}}
    ],
    "top_resources": [
      {"resource": "...", "categories": ["..."], "stat": {"all": 0}}
    ],
    "traffic": {"data": {"time": [], "input": [], "output": []}, "count": 0}
  },
  "device_analytics_page": {
    "analytics": [
      {"hostname": "...",
       "traffic": {"input": 0, "output": 0},
       "requests": 0,
       "anomaly_block_stat": {"anomalies": 0, "blocks": 0, "all": 0},
       "detections": {"all": 0, "unresolved": 0, "blocked": 0, "allowed": 0}}
    ]
  },
  "anomalies_list_page": {"anomalies": []},
  "top_anomalies_page": {"device_anomaly": []},
  "top_categories_page": {"categories": []},
  "block_recommendations_page": {
    "items": [
      {"domain": "", "url": "", "ip": "", "category": "", "hostname": "",
       "neg_rate": 0.0, "request_count": 0, "reason": ""}
    ],
    "count": 0
  }
}
```

6.1.4. Отчёт по отдельному узлу дополнительно содержит поле `hostname` и раздел категорий вместо части общих разделов.

## 6.2. Перечень ресурсов для блокировки

| Метод | Путь | Параметры |
|---|---|---|
| GET | `/api/v1/block-list` | `format` (`json` по умолчанию либо `csv`), `from`, `to`, `hostname` |

6.2.1. Метод доступен только роли `SA`.

6.2.2. В перечень включаются ресурсы, по которым принято решение о блокировке, с отбором по времени принятия решения и, при указании, по узлу межсетевого экрана.

6.2.3. Формат JSON:

```
{
  "schema_version": "1.0",
  "generated_at": "2026-09-11T14:00:00Z",
  "count": 42,
  "items": [
    {
      "domain": "example.com",
      "url": "https://example.com/path",
      "ip": "1.2.3.4",
      "category": "наименование категории",
      "hostname": "имя узла",
      "blocked_at": "2026-09-11T10:00:00Z",
      "blocked_by": "автор решения"
    }
  ]
}
```

6.2.4. Формат CSV:

— тип содержимого — `text/csv; charset=utf-8`;

— имя файла — `block-list.csv`;

— строка заголовка — `domain,url,ip,category,hostname,blocked_at,blocked_by`;

— значения полей соответствуют одноимённым полям формата JSON;

— отметка времени приводится по всемирному координированному времени в формате RFC 3339.

6.2.5. Автоматическая передача перечня на межсетевой экран Комплексом не выполняется; перечень предназначен для загрузки администратором межсетевого экрана.

### 6.3. Форматы отчётных документов

6.3.1. Отчётные документы в форматах PDF, XLSX, DOCX и PPTX формируются в веб-интерфейсе на основании документа отчёта, описанного в п. 6.1.3. Отдельные серверные методы выдачи указанных форматов не предусмотрены.

6.3.2. Состав отчётов приведён в пп. 7.2.5 и 7.2.6 Руководства оператора.

## 7. Версионирование схемы данных

7.1. Документы, предназначенные для машинной обработки, содержат поле `schema_version` со строковым значением версии схемы. Текущее значение для документа отчёта и для перечня ресурсов блокировки — `1.0`.

7.2. Изменения, не нарушающие обратную совместимость (добавление новых полей, добавление новых значений в необязательные перечисления), выполняются без изменения значения версии схемы. Потребитель обязан игнорировать неизвестные ему поля.

7.3. Изменения, нарушающие обратную совместимость (удаление или переименование поля, изменение типа значения, изменение смысла поля), сопровождаются увеличением версии схемы.

7.4. Значение версии схемы следует контролировать на стороне потребителя перед разбором документа.

## 8. Внутренние контракты обмена сообщениями

8.1. Внутреннее взаимодействие служб обработки выполняется через брокер сообщений. Контракты приведены для целей сопровождения и диагностики; в качестве интерфейса для смежных систем они не предназначены.

8.2. Перечень очередей:

| Очередь | Отправитель | Получатель |
|---|---|---|
| Запросы на обработку адресов | Служба ETL | Служба извлечения содержимого |
| Результаты по метаданным адресов | Служба извлечения содержимого | Служба ETL |
| Содержимое для анализа | Служба извлечения содержимого | Служба нейросетевой классификации |
| Результаты классификации | Служба нейросетевой классификации | Служба ETL |
| Очередь необработанных сообщений | Служба ETL | — |

8.3. Ключом сообщения во всех очередях обработки является сквозной идентификатор запроса. Он формируется службой ETL при постановке задачи и сохраняется во всех последующих сообщениях цепочки, что обеспечивает корреляцию результатов с исходным событием.

8.4. Сообщение с запросом на обработку содержит идентификатор запроса, сведения об источнике и назначении (тип ресурса, значение ресурса, порт, протокол, идентификатор содержимого) и отметку времени.

8.5. Сообщение с метаданными содержит идентификатор запроса, состояние обработки, сведения о домене (имя, сетевой адрес, геолокация), применённую стратегию извлечения, оценку качества извлечения и отметку времени обработки.

8.6. Сообщение с содержимым для анализа дополнительно содержит извлечённый текст и идентификатор содержимого.

8.7. Сообщение с результатом классификации содержит идентификатор запроса, идентификатор содержимого, определённую категорию, оценку уверенности, перечень наиболее вероятных категорий с оценками и вектор оценок по полному перечню категорий.

8.8. Служба классификации подтверждает получение сообщения только после успешной обработки, отправка результата выполняется с подтверждением от всех реплик и повторными попытками. Сообщения, не поддающиеся разбору, направляются в очередь необработанных сообщений с указанием исходной очереди, раздела, позиции, текста ошибки и исходного содержимого.

8.9. Повторная обработка одного и того же ресурса не приводит к изменению ранее принятого решения по нему.

## 9. Модель данных

### 9.1. Сессия


```
{
  "id": 1,
  "datetime_utc": "2026-09-11T12:00:00Z",
  "type": "Разрешен",
  "status": "Разрешен",
  "domain": "example.com",
  "url": "https://example.com/",
  "proto": "https",
  "hostname": "имя узла",
```

```
  "src_ip": "10.0.0.1",
  "src_country": "RU",
  "username": "пользователь",
  "dst_ip": "93.184.216.34",
  "dst_port": 443,
  "dst_country": "US",
  "category": "наименование категории"
}
```

### 9.2. Выявление

```
{
  "domain": "example.com",
  "url": "https://example.com/",
  "ip": "1.2.3.4",
  "port": 443,
  "location": "US",
  "hostname": "имя узла",
  "category": "наименование категории",
  "ngfw_category": "категория межсетевого экрана",
  "ml_ngfw_mismatch": false,
  "description": "пояснение",
  "request_count": 100,
  "request_count_30d": 50,
  "neg_rate": 0.85,
  "action": "Не решено",
  "action_created_by": "автор решения",
  "categorized_at": "2026-09-11T12:00:00Z"
}
```

9.2.1. В расширенном представлении дополнительно передаётся массив `categories` с элементами вида `{"name": "...", "rate": 0.0}`.

9.2.2. Значение поля `action_created_by`, соответствующее служебному субъекту автоматической политики, означает, что решение принято автоматически.

### 9.3. Учётная запись

```
{
  "user_id": "идентификатор",
  "login": "логин",
  "email": "адрес электронной почты",
  "first_name": "имя",
  "last_name": "фамилия",
  "patronymic": "отчество",
  "role": "SA",
  "avatar": "изображение профиля",
  "created_at": "2026-09-11T12:00:00Z",
  "is_super_admin": false,
  "is_need_to_change_password": false
}
```

### 9.4. Запись журнала бизнес-событий

```
{
  "user_name": "пользователь",
```

```
  "user_role": "SA",
  "event_type": "CREATE",
  "entity": "user",
  "entity_id": "идентификатор объекта",
  "context": "контекст",
  "description": "описание",
  "old_value": "прежнее значение",
  "new_value": "новое значение"
}
```

9.4.1. Допустимые значения поля `event_type`: `CREATE`, `UPDATE`, `DELETE`.

### 9.5. Узел межсетевого экрана

9.5.1. Узел идентифицируется значением `hostname`. Отдельная сущность образовательной организации в текущей версии интерфейса не выделена; принадлежность данных организации определяется узлом межсетевого экрана и идентификатором контекста в наименовании роли.

## 10. Интеграция со смежными системами

10.1. Реализованные способы получения данных смежными системами:

| Способ | Состояние | Примечание |
|---|---|---|
| REST по защищённому соединению с аутентификацией по маркеру доступа | Реализован | Раздел 5 |
| Выгрузка отчёта в формате JSON с версией схемы | Реализован | П. 6.1 |
| Выгрузка перечня ресурсов блокировки в форматах JSON и CSV | Реализован | П. 6.2 |
| Выгрузка отчётных документов в форматах PDF, XLSX, DOCX, PPTX | Реализован в веб-интерфейсе | П. 6.3 |
| Приём событий по протоколу Syslog | Отнесён ко второй очереди реализации | П. 11.2 Технического задания |
| Доступ смежных систем по протоколам ODBC и JDBC | Отнесён ко второй очереди реализации | П. 11.2 Технического задания |
| Согласованный формат передачи данных в системы класса SIEM и DLP | Отнесён ко второй очереди реализации | П. 11.2 Технического задания |
| Автоматическая передача правил блокировки на межсетевой экран | Отнесена ко второй очереди реализации | П. 11.2 Технического задания |

10.2. Рекомендуемый порядок интеграции смежной системы:

— получить у администратора Комплекса учётную запись с необходимой ролью;

— получить маркер доступа методом входа и обновлять его методом обновления по истечении срока действия;

— выполнять периодические обращения к методу выгрузки отчёта либо перечня ресурсов блокировки за требуемый период;

— контролировать значение поля версии схемы данных перед разбором документа;

— обрабатывать коды состояния согласно п. 4.2.2, предусмотрев повторные обращения при кодах 429, 502 и 503.

| **Разработчик:** | |
|---|---|
| ООО «Азон» | _____/ _____ |