Nevel

ВНЕШНЕЕ ИСПОЛЬЗОВАНИЕ API

Первый запрос к API

Как получить доступный каталог моделей и выполнить текстовый запрос через корпоративный API Nevel.

Облако / В контуре

https://docs.nevel.sh/api/quickstart/

API Nevel позволяет внутреннему приложению или агенту обращаться к моделям через корпоративный шлюз. Для первого запроса нужны адрес API вашей установки, ключ рабочего пространства и разрешённая модель. Этот сценарий отличается от подключения внешнего сервиса к чату: здесь именно ваше приложение вызывает модель.

Подготовьте параметры

Получите у администратора три значения и передайте их приложению через переменные окружения. Не сохраняйте ключ в исходном коде, браузерном приложении или общем документе.

Переменная	Значение
`NEVEL_BASE_URL`	Выданный базовый URL API с согласованным префиксом маршрута, без завершающего `/`
`NEVEL_API_KEY`	Действующий ключ нужного рабочего пространства
`NEVEL_MODEL`	Идентификатор разрешённой текстовой модели из каталога

В защищённом варианте относительный префикс маршрутов — `/secure/v1`. Например, к полному выданному базовому URL добавляется `/models` или `/chat/completions`. Не добавляйте второй `/v1`, если он уже входит в `NEVEL_BASE_URL`.

Маршруты с `/secure/v1` и обычные `/v1` различаются включением защитной обработки. Не меняйте префикс ради обхода ошибки. Допустимый маршрут и конфигурацию обработки определяет администратор; само название маршрута не подтверждает качество маскирования или допустимость конкретных данных. Подробнее — в разделе безопасности.

Получите каталог моделей

Следующая команда требует `curl`. Переменные должны быть заданы в текущем окружении. Проверки в начале предотвращают отправку запроса с отсутствующими параметрами.

```
: "${NEVEL_BASE_URL:?Укажите выданный базовый URL API}"
: "${NEVEL_API_KEY:?Укажите API-ключ}"

curl --fail-with-body --silent --show-error \
  "${NEVEL_BASE_URL%/}/models" \
  -H "Authorization: Bearer ${NEVEL_API_KEY}"
```

Успешный ответ имеет `object: "list"` и массив `data`. Используйте значение `id` разрешённой текстовой модели. Не выбирайте автоматически первый элемент: каталог может содержать модели другого назначения. Если нужной модели нет, проверьте конфигурацию пространства с администратором.

Выполните текстовый запрос

Пример использует Python 3 и стандартную библиотеку, поэтому не зависит от версии стороннего SDK. Он корректно формирует JSON и выводит текст первого ответа. Значение `NEVEL_MODEL` предварительно задаётся по полученному каталогу.

```
import json
import os
import urllib.error
import urllib.request

base_url = os.environ["NEVEL_BASE_URL"].rstrip("/")
payload = {
    "model": os.environ["NEVEL_MODEL"],
    "messages": [
        {"role": "user", "content": "Ответь одним словом: готово"}
    ],
    "stream": False,
}
request = urllib.request.Request(
    base_url + "/chat/completions",
    data=json.dumps(payload, ensure_ascii=False).encode("utf-8"),
    headers={
        "Authorization": "Bearer " + os.environ["NEVEL_API_KEY"],
        "Content-Type": "application/json",
    },
    method="POST",
)
try:
    with urllib.request.urlopen(request, timeout=60) as response:
        result = json.load(response)
    print(result["choices"][0]["message"]["content"])
except urllib.error.HTTPError as error:
    print("HTTP", error.code)
    print(error.read().decode("utf-8", errors="replace"))
    raise SystemExit(1)
```

Таймаут 60 секунд в примере — настройка клиента, а не гарантия времени ответа сервиса. Для своего приложения выберите таймаут по рабочему сценарию и обработайте прерывание запроса. Не используйте бесконечные автоматические повторения: они могут создавать дополнительную нагрузку и расход.

Подключение совместимого клиента

В клиенте с настройкой OpenAI-compatible API укажите выданный базовый URL, ключ и идентификатор модели. Начните с простого текстового запроса без инструментов, изображений и специальных параметров. После успешной проверки добавляйте необходимые возможности по одной: их поддержка зависит от модели и маршрута.

Для on-prem используйте выданный адрес API установленной у заказчика платформы. Этот адрес определяет точку входа, а поле `model` и политика — дальнейший маршрут. Модель может работать локально у заказчика, у российского провайдера, на серверах Nevel в РФ или у глобального провайдера, если такой путь разрешён. Доступные идентификаторы берите из каталога конкретного пространства, а размещение и получателей — из паспорта маршрута: имя модели само по себе их не доказывает.

Изолированный профиль использует только внутренние обработчики и модели. В любой конфигурации клиент и SDK не должны незаметно переключаться на другого провайдера при ошибке. Резервный маршрут допускается только как заранее разрешённый для этих данных путь с фиксацией переключения. Сверьте настройки клиента с матрицей вариантов и сетевой политикой установки.

Проверка результата

Убедитесь, что каталог доступен, текстовый запрос завершился успешно, а приложение корректно показывает ошибку с отключённым ключом. Зафиксируйте модель, время и HTTP-статус без сохранения секрета. Затем настройте лимиты и жизненный цикл ключа и добавьте обработку ошибок перед использованием рабочих данных.