Omnicomm Online

O OmniComm Online permite ao usuário controlar a operação de veículos e motoristas por meio da geração de relatórios integrados. Para acessar o Omnicomm Online, basta ter um computador pessoal conectado à Internet.

O sistema oferece ferramentas integradas especiais para a coleta e uso de dados processados em relatórios gerenciais e sistemas de monitoramento de frotas veiculares.

Este manual descreve a operação dos métodos de integração para a interação com sistema de terceiros.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api

Last update: **2026/11/16 13:41**

Conexão com o Webservice

Para obter a WSDL do webservice do Omnicomm Online, o cliente deve contatar a equipe de suporte e tecnologia da Omnicomm, através dos telefones 11 3042-7765 e 11 94265-2167.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/connection

Last update: **2026/11/16 13:41**

SOAP

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/definition

Last update: **2026/11/16 13:41**

Erros

Lista de erros retornados:

0: No errors – há sem erros

1: Signing in failed – Login/senha digitados incorretamente

2: Authorization required – autorização é necessária para acessar os dados

3: Dead session number – sessão expirou, re-autorização é necessária

4: Bad interval – intervalo incorreto inserido

5: Bad object – não há nenhum objeto com esse identificador

6: Admin login – alguém está tentando fazer login como usuário Administrador

7: Unusable object – o valor não pode ser calculado para o objeto com esse identificador

8: Bad event type – não há nenhum objeto com esse identificador

9: Access denied – sem autorização para acessar o objeto

10: Data not found – sem dados para os valores de entrada correspondentes

11: Blocked interval – o intervalo solicitado contém períodos de bloqueio de dados

12: Bad object type – o tipo de objeto especificado não existe

13: Invalid format – o formato está incorreto

14: Undefined error – o erro é não especificado

15: 404 – página não encontrada

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/definition/error

Last update: **2026/11/16 13:41**

Exemplo de consumo do Webservice

Importação de interfaces

wsimport -d bin -s src http://demo.omnicomm.ru:8000/AnalyticalServer/ws?wsdl

Código Java:

É necessário alterar as strings «user» e «pass» para os valores reais.

```
package ru.omnicomm.test.client;

import ru.omnicomm.analyticalserver.*;

import java.net.MalformedURLException;

import java.net.URL;

import java.util.List;

public class  ExampleClient {
    public static void main(String[] args) throws MalformedURLException {
        AnalyticalServer = new AnalyticalServer(new
URL("http://demo.omnicomm.ru:8000/AnalyticalServer/ws?wsdl"));

        AnalyticalServerWS port =
AnalyticalServer.getAnalyticalServerPort();

        AuthResponseEntry auth = port.signIn("user", "pass");

        String sessionId = auth.getsessionId ();

        System.out.println("auth sessionId: " + sessionId);

        ObjectSetResponseEntry objects = port.getObjectSet(sessionId);
        List<Vehicle> vehicles = objects.getVehicleList();

        for (Vehicle vehicle : vehicles) {
            System.out.printf("vehicle: %d / %s\n", vehicle.getVehicleID(),
vehicle.getRegNmb());
        }

    }
}
```

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/definition/example

Last update: **2026/11/16 13:41**

Lista de métodos SOAP

Para transmitir data e hora, é usado o formato UNIXTIME (em segundos). As tipagens de outros parâmetros são listadas abaixo

signIn – autorização

Valores de entrada
String login – nome de usuário no sistema **String password** – senha no sistema
Valores de retorno
Boolean **status** – verdadeiro/falso, verdadeiro em caso de autorização bem-sucedida String **sessionId** – no caso de autenticação bem-sucedida, a ID de sessão (mínimo de 16 caracteres) Unixtimestamp **dateTimeEnd** – em caso de autorização bem-sucedida, o término da sessão tempo após o qual você deve fazer login novamente) String **error** – mensagem de erro em caso de autorização indevida (nome de usuário e senha digitados incorretamente ou formato de dados incorretos)

getObjectSet – lista de objetos

Valores de entrada
String **sessionId** – ID de sessão obtida durante a autorização
Valores de retorno
Boolean **status** – verdadeiro/falso, verdadeiro em caso de operação bem-sucedida String **error** – mensagem de erro em caso de ocorrência de erro Dataset **objects** – lista de veículos disponíveis para o usuário — proprietário da sessão registrado (ou seja, somente os objetos que o usuário tem direito de visualizar): Integer **id** – identificador do objeto (o mesmo que o identificador da unidade) String **objectName** – nome do veículo String **objectType** – tipo do veículo String **GarageNumber** – número da oficina

getSmoothedFuel – nível de combustível suavizado para o período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo
Valores de retorno
Boolean **status** – verdadeiro/falso, verdadeiro em caso de operação bem-sucedida Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Dataset **fuel** – conjunto de dados (todos os dados arquivados para o período selecionado): Unixtimestamp **timeStamp** – hora do nível de combustível registrado Double **smothedFuel** – valor suavizado do combustível, litros, precisão até 0,1 l String **error** – mensagem em caso de ocorrência de erro

getFuelConsumption – consumo de combustível para o período

Valores de entrada
String **sessionId** - identificador de sessão obtido durante da autorização Integer **objectId** - identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp ** timeEnd** (segundos) - hora de término do intervalo

Valores de retorno
Boolean **status** - verdadeiro/falso, verdadeiro em caso de operação bem-sucedida. Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp ** timeEnd** (segundos) – hora de término do intervalo Double **fuelConsumption** – consumo de combustível para o período, litros, precisão de até 0,1 l String **error** – mensagem em caso de ocorrência de erro

getEvents – lista de eventos

Valores de entrada
String sessionId – identificador de sessão obtido durante a autorização **Integer objectId** – identificador do veículo/unidade. Parâmetro opcional, se o identificador não estiver presente, retorna os dados para todos os veículos. **Integer type** – tipo de evento, obrigatório. **Unixtimestamp timeBegin** (segundos) – hora de início do intervalo **Unixtimestamp timeEnd** (segundos) - hora final do intervalo

Valores de retorno
Boolean **status** – verdadeiro/falso, verdadeiro em caso de operação bem-sucedida. String **error** - mensagem de erro em caso de ocorrência de erro Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Dataset **objectEvents** – – conjunto de dados para cada evento: Unixtimestamp **timeStamp** – data e hora do evento Integer **objectId** – identificador do veículo/unidade String **type** – tipo do evento String **parameters** – parâmetros do evento String **eventAddress** – endereço do evento, se disponível String **iButton** – código iButton, em hexadecimal. Somente para eventos como 'Motorista' String **name** – nome da cerca eletrônica. Apenas para entrada e saída ou de eventos de cerca eletrônica.

getMileage – quilometragem para o período

Valores de entrada
String **sessionId** – identificador de sessão durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) - hora de término do intervalo

Valores de retorno
Boolean **status** - – verdadeiro/falso, verdadeiro em caso de operação bem-sucedida. Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Double **Mileage** – quilometragem em km em intervalo especificado, precisão de 0,1 km String **error** – mensagem em caso de ocorrência de erro

getEngineOnTime – tempo de funcionamento do motor para o período

Valores de entrada
String **sessionId** - identificador de sessão obtido durante a autorização Integer **objectId** - identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) - hora de término do intervalo

Valores de retorno
Boolean **status** - verdadeiro/falso, verdadeiro em caso de operação bem-sucedida Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Double **engineOnTime** – duração total de funcionamento do motor, em segundos String **error** – mensagem de erro em caso de ocorrência de erro

getVehiclesState – status atual do veículo

Valores de entrada
String **sessionId** – identificador de sessão obtido durante autorização VehiclesType **vehicles** – lista de IDs de veículos

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem de erro em caso de ocorrência de erro vehicleStatesType **states** – lista de parâmetros descrevendo cada status de veículo

getMileageSpeedExcess – quilometragem com excesso de velocidade para o período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo

Valores de retorno
Boolean **status** - verdadeiro/falso, verdadeiro em caso de operação bem-sucedida. Unixtimestamp ** timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Double **mileageSpeedExcess** – quilometragem com excesso de velocidade em km para o intervalo especificado, precisão de 0,1 km String **error** – mensagem de erro em caso de ocorrência de erro

getMovementTime – tempo em deslocamento para o período

Valores de entrada

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo

Valores de retorno
Boolean **status** - verdadeiro/falso, verdadeiro em caso de operação bem-sucedida. Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp ** timeEnd** (segundos) – hora de término do intervalo Double **movementTime** – tempo de deslocamento, segundos String **error** – mensagem de erro em caso de ocorrência de erro

getEngineOnTimeInMovement – tempo de funcionamento do motor em deslocamento para o período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo

Valores de retorno
Boolean **status** - verdadeiro/falso, verdadeiro em caso de operação bem-sucedida Unixtimestamp ** timeBegin** (segundos) – hora de início do intervalo Unixtimestamp ** timeEnd** (segundos) – hora de término do intervalo Double **engineOnTimeInMovement** – tempo de funcionamento do motor em deslocamento, segundos String **error** – mensagem em caso de ocorrência de erro

getEngineOnTimeWithoutMovement – tempo de funcionamento do motor em ociosidade para o período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp ** timeEnd** (segundos) – hora de término do intervalo

Valores de retorno
Boolean **status** – verdadeiro/falso, verdadeiro em caso de operação bem-sucedida Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Double **engineOnTimeWithoutMovement** – tempo de funcionamento do motor em ociosidade, segundos String **error** – mensagem de erro em caso de ocorrência de erro

getEngineOffTime – tempo com o motor desligado para o período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo

Valores de retorno
Boolean **status** - verdadeiro/falso, verdadeiro em caso de operação bem-sucedida Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Double **engineOffTime** – tempo de motor desligado para o período, segundos String **error** – mensagem em caso de ocorrência de erro

getFuelConsumptionInMovement – consumo de combustível em movimento para o período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo

Valores de retorno
Boolean **status** - verdadeiro/falso, verdadeiro em caso de operação bem-sucedida Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Double **fuelConsumptionInMovement** – consumo de combustível durante o movimento para o período, litros, precisão de 0,1 litro String **error** – mensagem em caso de ocorrência de erro

getFuelConsumptionWithoutMovement – consumo de combustível sem movimento para o período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo

Valores de retorno
Boolean **status** – verdadeiro/falso, verdadeiro em caso de operação bem-sucedida Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Double **fuelConsumptionWithout Movement** – consumo de combustível em ociosidade para o período, litros, precisão de 0,1 l String **error** – mensagem em caso de ocorrência de erro

getFuelConsumptionInMotohour – consumo de combustível por hora do motor

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo

Valores de retorno
Boolean **status** - verdadeiro/falso, verdadeiro em caso de operação bem-sucedida Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Double **fuelConsumptionIn Motohour** – consumo médio de combustível para a hora do motor para o período, litros, precisão de 0,1 l String **error** – mensagem em caso de ocorrência de erro

getFuelAtTime – nível de combustível em um determinado momento

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **time** (segundos) – horário do momento

Valores de retorno
Boolean **status** - verdadeiro/falso, verdadeiro em caso de operação bem-sucedida Unixtimestamp **time** (segundos) – horário do momento Double **fuelAtTime** – nível de combustível em um determinado momento, litros, precisão de 0,1 l String **error** – mensagem em caso de ocorrência de erro

getUserNotificationsByPeriod – notificações do usuário no período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante autorização Unixtimestamp **timeBegin** – hora de início do intervalo (UTC), segundos Unixtimestamp **timeEnd** – hora de término do intervalo (UTC), segundos. Se não definido, o horário de término é o horário da execução de solicitação pelo servidor Integer **page** – número da página solicitada. Se não definido, retorna a primeira página contém registros por página Integer **perPage** – quantidade de registros por página, se não definido, a quantidade não é limitada

Valores de retorno
Boolean **status** – status da operação. Verdadeiro no caso de operação bem-sucedida String **error** – mensagem de erro em caso de ocorrência de erro Unixtimestamp **timeBegin** – hora de início do intervalo (UTC), segundos Unixtimestamp **timeEnd** – hora de término do intervalo (UTC), segundos userNotificationsType **userNotifications** – matriz de retorno de conjuntos de parâmetros Integer **notificationsCount** – número total de notificações para todas as páginas. Se nenhuma notificação for encontrada, retorna o código de erro 10

getVisitedGeozonesByPeriod – geodelimitações visitadas no período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Unixtimestamp **timeBegin** – hora de início do intervalo (UTC), segundos Unixtimestamp **timeEnd** – hora de término do intervalo (UTC), segundos vehicleIdsType **vehicleId** – lista de IDs de veículos. Se ausente, todos os veículos disponíveis são usados para a solicitação

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem em caso de ocorrência de erro Unixtimestamp **timeBegin** – hora de início do intervalo (UTC), segundos Unixtimestamp **timeEnd** – hora de término do intervalo (UTC), segundos visitedGeozone **geozoneVisits** – blocos de matrizes de parâmetros de cercas eletrônicas visitadas

getVehiclesParams – lista de parâmetros disponíveis para o usuário

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem de erro em caso de ocorrência de erro Vehicle **vehicles** – parâmetros de veículos que o usuário atual tem o direito de visualizar

signOut – encerramento de sessão

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem de erro em caso de ocorrência de erro String **sessionId** – ID de sessão encerrada

getActiveNotificationRules – perfis de notificações ativas

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem em caso de ocorrência de erro rulesType **rules** – parâmetros para cada perfil de notificação

setDeviceIdToNotificationRules – atribuição de perfis de notificação para o veículo

Valores de entrada
String **sessionId** – ID de sessão obtida durante a autorização String **deviceId** – ID do dispositivo String **deviceTypeId** – identificador do tipo de dispositivo rulesType **rules** – identificadores de perfis de notificação, necessários para atribuir veículos

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem de erro em caso de ocorrência de erro

getFuelLevelsByTimeMoment – nível de combustível em um determinado momento

Valores de entrada
String **sessionId** – identificador de sessão obtido durante autorização Unixtimestamp **timeMoment** – hora de início do intervalo (UTC), segundos vehicleAndTankIdsType **vehicleAndTankIds** – lista de IDs de veículos e tanques de combustível. Na ausência da lista, a solicitação será executada para todos os veículos e tanques disponíveis para o usuário

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem em caso de ocorrência de erro Unixtimestamp **timeMoment** – horário do momento (UTC), segundos fuelData **fuelDataSet** – lista de parâmetros para cada veículo: int **vehicleId** – identificador do veículo; int **tankNumber** – número do tanque de combustível ; fuelLevelsType **fuelLevels** – dados de nível de combustível; activityPeriodsType **activityPeriods** – dados de funcionamento do motor; ignitionOffListType **ignitionOffList** – dados de ignição desligada; ignitionOnListType **ignitionOnList** – dados de ignição ligada; llsFailurePeriodsType **llsFailurePeriods** – dados sobre falhas de sensor de nível de combustível

getFuelLevelsByPeriod – nível de combustível no período de tempo

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Unixtimestamp **timeBegin** – hora de início do intervalo (UTC), segundos Unixtimestamp **timeEnd** – hora de término do intervalo (UTC), segundos. Se a hora de término não for indicado, a hora de término = a hora do sistema do início do lado do servidor processamento de solicitação VehicleAndTankIdsType **vehicleAndTankIds** – lista de IDs de veículos e tanques de combustível. Na ausência da lista, a solicitação será executada para todos os veículos e tanques disponíveis para o usuário Int **reduce** – desbaste: 0 = opcional 1 = obrigatório

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem de erro em caso de ocorrência de erro Unixtimestamp **timeBegin** – hora de início do intervalo (UTC), segundos Unixtimestamp **timeEnd** – hora de término do intervalo (UTC), segundos fuelData **fuelDataSet** – lista de parâmetros para cada veículo; int **vehicleId** – identificador do veículo; int **tankNumber** – número do tanque de combustível; fuelLevelsType **fuelLevels** – dados de nível de combustível; activityPeriodsType **activityPeriods** – dados de funcionamento do motor; ignitionOffListType **ignitionOffList** – dados de ignição desligada; ignitionOnListType **ignitionOnList** – dados de ignição ligada; llsFailurePeriodsType **llsFailurePeriods** – dados sobre falhas de sensor de nível de combustível

getSmoothedFuelLevelsByPeriod – níveis de combustível suavizados no período de tempo

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do objeto Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo vehicleAndTankIdsType **vehicleAndTankIds** – lista de IDs de veículos e tanques de combustível. Na ausência da lista, a solicitação será executada para todos os veículos e tanques disponíveis para o usuário Int **reduce** – desbaste: 0 = opcional 1 = obrigatório

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem de erro em caso de ocorrência de erro Unixtimestamp **timeBegin** – hora de início do intervalo (UTC), segundos Unixtimestamp **timeEnd** – hora de término do intervalo (UTC), segundos fuelData **fuelDataSet** – lista de parâmetros para cada veículo: int **vehicleId** – identificador do veículo; int **tankNumber** – número do tanque de combustível; fuelLevelsType **fuelLevels** – dados de nível de combustível; activityPeriodsType **activityPeriods** – dados de funcionamento do motor; ignitionOffListType **ignitionOffList** – dados de ignição desligada; ignitionOnListType **ignitionOnList** – dados de ignição ligada; llsFailurePeriodsType **llsFailurePeriods** – dados sobre falhas de sensor de nível de combustível

getRefuelingsAndDrainsByPeriod – operações de drenagem/reabastecimento operações no período de tempo

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo vehicleAndTankIdsType **vehicleAndTankIds** – lista de IDs de veículos e tanques de combustível. Na ausência da lista, a solicitação será executada para todos os veículos e tanques disponíveis para o usuário Integer **page** – número da página solicitada com dados Integer **perPage** – número de entradas por página; se não for predefinido, será considerado ilimitado String **sortname** – campo necessário para organizar os parâmetros retornados String **sortorder** – ordem de classificação: **asc** – crescente **desc** – decrescente

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem de erro em caso de ocorrência de erro Unixtimestamp **timeBegin** – hora de início do intervalo (UTC), segundos Unixtimestamp **timeEnd** – hora de término do intervalo (UTC), segundos Integer **entriesCounter** – número total de entradas por período de tempo RefuelingsAndDrainsType **RefuelingsAndDrains** – lista de parâmetros para cada veículo

getVehiclesProfiles – perfis do veículo correspondentes aos identificadores do veículo

Valores de entrada
String **sessionId** – identificador de sessão obtido durante autorização VehiclesType **vehicles** – lista de IDs de veículos

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem de erro em caso de ocorrência de erro **Vehicles** – lista de parâmetros para cada veículo

getCurrentObjectState – status atual do veículo

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do objeto

Valores de retorno
Boolean **status** – verdadeiro/falso. Verdadeiro no caso de operação bem-sucedida. String **error** - mensagem de erro em caso de ocorrência de erro String **lastGPS** – últimas coordenadas válidas. Contém valores de latitude e longitude, separados por ponto e vírgula Integer **lastGPSDir** – direção de deslocamento, graus de 0 a 359 Double **currentSpeed** – velocidade atual em determinado momento, em km/h, precisão de 0,1 km/h Double **currentFuel** – nível atual de combustível, em litros, precisão de 0,1 l Boolean **currentIgn** – status de ignição. Verdadeiro, se a ignição está LIGADA Boolean **speedExceed** – excesso de limite de velocidade. Verdadeiro no caso de limite de velocidade excedido Integer **lastGPSSat** – número de satélites, com as últimas coordenadas válidas Double **currentInputValue** – valor real de entrada universal. Atributos: Número inteiro – número de IU, nome da string – nome da IU

getReportData – relatório de equipamento auxiliar ao longo do período, TPMS, IQFreeze

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do objeto Long **timeBegin** – hora de início do intervalo (UTC), segundos Long **timeEnd** – hora de término do intervalo (UTC), segundos String **reportTemplateID** – identificador de modelo de relatório no Omnicomm Online. Valores possíveis: addEquipment, TPMS, refState, refWork

Valores de retorno
Boolean **status** – verdadeiro/falso. Verdadeiro no caso de operação bem-sucedida String **error** – mensagem de erro em caso de ocorrência de erro ReportDataType **reportData** – matriz incluindo dados de relatório

getSEOnTime – tempo de funcionamento do equipamento auxiliar para o período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador do veículo/unidade Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) - hora de término do intervalo

Valores de retorno
Boolean **status** - verdadeiro/falso, verdadeiro em caso de operação bem-sucedida Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Double **sEOnTime** (segundos) – tempo de funcionamento de equipamentos auxiliares para cada IU conectada. Atributos: Número inteiro – número de IU, nome da string – nome da IU

getStatisticsByPeriod – estatísticas para o período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante autorização Unixtimestamp **timeBegin** – hora de início do intervalo (UTC), segundos Unixtimestamp **timeEnd** – hora de término do intervalo (UTC), segundos. Se não definido, a hora de término = hora da execução da solicitação pelo servidor. int **objectType** – tipo de objeto: 0=veículo; 1=motorista; Se o tipo não existir, retorna o código de erro 12. objectIdsType **objectIds** – matriz do tipo objectIdsType, contendo a lista de parâmetros objectId do tipo inteiro. Caso não exista, a consulta é executada para todos o objectId do tipo correspondente, disponíveis para o usuário. requiredStatParamsType **requiredStatParams** – lista dos subgrupos necessárias dos parâmetros do relatório 'Estatísticas'. Se a lista estiver vazia, retorna todos os subgrupos com todos os parâmetros

Valores de retorno
movingAndWorkingParamsType **movementAndWorkingParams** – subgrupo dos parâmetros de deslocamento o e operação de veículos. Se a lista estiver vazia, retorna todos os parâmetros do subgrupo, caso contrário, retorna apenas os parâmetros enumerados. Se o subgrupo não for transmitido, não retorna o subgrupo todo. fuelParamsType **fuelParams** – subgrupo de parâmetros para combustível, drenagem e reabastecimentos. Se a lista estiver vazia, retorna todos os parâmetros do subgrupo, caso contrário, retorna apenas os parâmetros enumerados. Se o subgrupo não for transmitido, não retorna o subgrupo todo. optionalEquipmentParamsType **optionalEquipmentParams** – subgrupo de parâmetros do equipamento auxiliar. Se a lista estiver vazia, retorna todos os parâmetros do subgrupo, caso contrário, retorna apenas os parâmetros enumerados. Se o subgrupo não for transmitido, não retorna o subgrupo todo. CANDataParamsType** CANDataParams** – subgrupo dos parâmetros CAN. Se a lista estiver vazia, retorna todos os parâmetros do subgrupo, caso contrário, retorna apenas os parâmetros enumerados. Se o subgrupo não for transmitido, não retorna o subgrupo todo. addDataParamsType **addDataParams** – subgrupo de parâmetros estatísticos adicionais: TPMS, iQFreeze etc.

getTracksByPeriod – rastreamento do veículo para o período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Unixtimestamp **timeBegin** – hora de início do intervalo (UTC), segundos Unixtimestamp **timeEnd** – hora de término do intervalo (UTC), segundos. Se a hora de término não for indicado, a hora de término = a hora do sistema do início do lado do servidor processamento de solicitação VehiclesType **vehicles** – lista de ID de veículos Int **reduce** – desbaste: 0 = opcional 1 = obrigatório

Valores de retorno
Boolean **status** – status da operação String **error** – mensagem de erro em caso de ocorrência de erro Unixtimestamp **timeBegin** – hora de início do intervalo (UTC), segundos Unixtimestamp **timeEnd** – hora de término do intervalo (UTC), segundos trackDataSetType **trackDataSet** – parâmetros de rastreamento para cada veículo: **trackPoint** – parâmetros do ponto de rastreamento por um veículo unixtimestamp **timestamp** – hora do evento quando as coordenadas foram fixadas Integer **latitude** – latitude com precisão 0,0000001 grau Integer **longitude** – longitude com precisão de 0,0000001 grau Integer **direction** – direção, graus Integer **sattelitesCount** – número de satélites Double **speed** (km/hora) – velocidade Long **timeStamp** (segundos) – hora do evento. (UTC)

getTrack – rastreamento

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **objectId** – identificador de objeto Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo

Valores de retorno
Boolean **status** – verdadeiro/falso, verdadeiro em caso de operação bem-sucedida String **error** – mensagem de erro em caso de ocorrência de erro Unixtimestamp **timeBegin** (segundos) – hora de início do intervalo Unixtimestamp **timeEnd** (segundos) – hora de término do intervalo Dataset **trackEvents** – matriz de pontos de rastreamento: String **gpsPos** – coordenadas do evento. Contém valores de latitude e longitude, separados por ponto e vírgula Integer **gpsDir** – direção de movimento, graus de 0 a 359 Integer **sattelitesCount** – número de satélites Double **speed** – velocidade, em km/hora com precisão de até 0,1 km/hora Unixtimestamp **timeStamp** – ponto de data e hora

getEngineStatisticsByPeriod – estatísticas de operação do motor durante um período

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **vehicleId** – identificador do veículo Unixtimestamp **timeBegin** (segundos) – momento de início do intervalo Unixtimestamp **timeEnd** (segundos) – momento de término do intervalo

Valores de retorno
Boolean **status** – true/false Retorna 'true' (verdadeiro) se a operação foi bem sucedida String **error** – texto da mensagem de erro que aparece quando se dá o erro Unixtimestamp **timeBegin** (segundos) – momento de início do intervalo Unixtimestamp **timeEnd** (segundos) – momento de término do intervalo Unixtimestamp **lastDataTimestamp** (segundos) – registro de hora dos últimos dados processados (UTC) Unixtimestamp **operationStartDate** (segundos) – momento de início da operação para o período (UTC) Unixtimestamp **operationEndDate** (segundos) – momento de término da operação para o período (UTC) Integer **operationTime** (segundos) – tempo de operação para o período Integer **engineOffTime** (segundos) – duração com o motor desligado para o período Integer **engineOnTime** (segundos) – duração com o motor ligado para o período Integer **engineIdlingTime** (segundos) – duração da operação do motor em marcha lenta para o período Integer **engineOperationTimeNormalSpeed** (segundos) – duração da operação do motor sob carga normal para o período Integer **engineOperationTimeMaxSpeed** (segundos) – duração da operação do motor sob carga máxima para o período Integer **engineLoadTime** (segundos) – duração da operação do motor sob carga para o período Integer **dataAbsenceTime** (segundos) – duração da ausência de dados para o período

getShiftDataByPeriod – informação sobre os turnos

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Unixtimestamp **timeBegin** (segundos) – hora de início do turno agendada Unixtimestamp **timeEnd** (segundos) – hora de término do turno agendada Integer **devBeforeBegin** (segundos) – adiantamento permitido da hora de início do turno agendada Integer **devAfterBegin** (segundos) – atraso permitido da hora de início do turno agendada Integer **devBeforeEnd** (segundos) – adiantamento permitido da hora de término do turno agendada Integer **devAfterEnd** (segundos) – atraso permitido da hora de término do turno agendada Integer **vehicleId** – identificador do veículo

Valores de retorno
Boolean **status** – true/false Retorna 'true' (verdadeiro) se a operação foi bem sucedida String **error** – texto da mensagem de erro que aparece quando se dá o erro Integer **vehicleId** – identificador do veículo Integer **vehicleType** – tipo de veículo (0 - carro, 1 - caminhão-tanque) **shiftData**: Unixtimestamp **pointDate** (segundos) – momento efetivo do início/término do turno (UTC) Integer **engineOperationTime** (segundos) – tempo de operação do motor no momento do início/término do turno Integer **shiftEngineOperationTime** (segundos) – tempo de operação do motor para o turno Double **mileage** (km) – tempo de operação do motor no momento do início/término do turno Double **shiftMileage** (km) – quilometragem do turno **fuelData** – grupo de parâmetros de combustível: Double **fuelVolume** (l) – volume de combustível no momento do início/término do turno Double **fuelConsumption** (l) – consumo de combustível efetivo no momento do início/término do turno

getIntervalsInfo – informação dividida em intervalos

Valores de entrada
String **sessionId** – identificador de sessão obtido durante a autorização Integer **vehicleId** – identificador do veículo Unixtimestamp **timeBegin** (segundos) – momento de início do intervalo Unixtimestamp **timeEnd** (segundos) – momento de término do intervalo Boolean **partsFlag** – divisão em partes (true or false) Integer **interval** (min) – duração do intervalo Boolean **geocodingFlag** – resolução do endereço com base nas coordenadas (true or false) Boolean **additionalTankFlag** – retorna um conjunto de dados sobre o nível de combustível no tanque adicional (true or false)

Valores de retorno
Boolean **status** – true/false Retorna 'true' (verdadeiro) se a operação foi bem sucedida Integer **errorCode** – código de erro. Códigos de erro possíveis: 2: Authorization required - é necessária autorização para acessar os dados 3: Dead session number – a sessão expirou, é necessário autorizar novamente 4: Bad interval – inserido um intervalo incorreto 5: Bad object – não há nenhum veículo com esse identificador 7: Unusable object– o valor não pode ser calculado para um objeto com esse identificador. 9: Access denied – sem direitos de acesso ao objeto 10: Data not found – não há dados para os valores de entrada correspondentes (sem eventos de dados brutos para o [período de solicitação + um evento antes do início do período]) 11:Blocked interval – o intervalo solicitado contém períodos de bloqueio de dados 13: Invalid format – o formato está incorreto 14: Undefined error – o erro não foi especificado 19: Too many intervals: current N, allowed M - o número de intervalos por período (N) excede o limite (M) String **erorrDescription** – opcional Integer **vehicleId** – identificador do veículo Integer **vehicleType** – tipo de veículo (0 - carro, 1 - caminhão-tanque) Integer **tanksNumber** – número de tanques de combustível Integer **numberOfIntervals** – número de intervalos no período solicitado **periodData** – dados definidos para o período solicitado **intervalsData** – dados para o intervalo: Integer **intervalNumber** – número da sequência de intervalo Unixtimestamp **startTime** – data de início do intervalo Unixtimestamp **endTime** – data de término do intervalo **statData** – dados definidos para as estatísticas **gpsData** – dados definidos para o GPS **engineData** – dados definidos para a operação do motor **fuelData** – dados definidos para o nível de combustível **seData** – dados definidos para a operação do equipamento auxiliar **canData** – dados do barramento CAN

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/definition/method

Last update: **2026/11/16 13:41**

Tipos de eventos

Evento tipo número	Significado	Valores de parâmetro, comentários
1	Início do reabastecimento (para caminhões-tanque de reabastecimento – abastecimento)	Valor de reabastecimento, precisão de 0,1 litro
2	Fim do reabastecimento (para caminhões-tanque de reabastecimento – abastecimento)	Valor de reabastecimento, precisão de 0,1 litro
3	Início de drenagem	Valor de drenagem, precisão de 0,1 litro
4	Término da drenagem	Valor de drenagem, precisão de 0,1 litro
5	Ignição LIG	Tempo desde o último desligamento da ignição, minutos
6	Ignição DESL	
7	Fonte de alimentação externa LIG	
8	Bateria LIG	
9	Autorização do motorista	Código chave iButton, em HEX
10	Transição para roaming	
11	Saída de roaming	
12	Excesso de velocidade instantâneo	Valor máximo de velocidade, precisão de 0,1 km/h
13	Tempo de ociosidade	
14	Início de excesso de velocidade	
15	Início de grupos de transação (para reabastecimento de caminhões-tanque)	Volume de combustível antes do início de grupos de transações, precisão de 0,1 litro
16	Término de grupos de transação (para reabastecimento de caminhões-tanque)	Volume de combustível após encerramento dos grupos de transações, precisão de 0,1 litro
17	Início do abastecimento durante grupos de transações (para reabastecimento de caminhões-tanque)	
18	Término do abastecimento durante grupos de transações (para reabastecimento de caminhões-tanque)	
19	Início de drenagem durante grupos de transações (para reabastecimento de caminhões-tanque)	
20	Término de drenagem durante grupos de transações (para reabastecimento de caminhões-tanque)	
21	Conexão estabelecida	
22	Equipamento adicional LIG	
23	Equipamento adicional DESL	
24	Equipamento auxiliar excedendo limites máx	

Evento tipo número	Significado	Valores de parâmetro, comentários
25	Voltar aos valores normais de equipamento auxiliares valores normais	
31	Pressionamento do botão de pânico	
32	Sobrecarga de equipamento auxiliar	
33	LIGADO	
34	RPM excedido instantâneo	
35	Entrada em cerca eletrônica	
36	Saída de cerca eletrônica	
38	DESLIGADO	
42	Início da parada	
43	Término da parada	
44	Início da aceleração	
45	Término da aceleração	
46	Entrada digital LIG	
47	Entrada digital DESL	
48	Aceleração instantânea	
49	Motorista desconhecido	
52	Adulteração de dispositivo	
53	Autorização de motorista encerrada	
54	iButton aplicado	
55	Registro de motorista excluído	

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/definition/type

Last update: **2026/11/16 13:41**

Como testar a conexão com o Webservice

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/demo

Last update: **2026/11/16 13:41**

REST API

Use o seguinte para acesso de demonstração:

endereço: http://online.omnicomm.ru

nome de usuário: rudemoru

senha: rudemo123456

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/demo/rest

Last update: **2026/11/16 13:41**

SOAP

Se necessário, para testar a conexão para webservices (para verificar o aplicativo sem usar uma conta real do Omnicomm Online, ou se houver dúvida sobre as configurações de rede), você pode usar o servidor de demonstração de webservices da Omnicomm.

Endereço da WSDL (não para o sistema 1C):

http://demo.omnicomm.ru:8000/AnalyticalServer/v2/ws?wsdl

Endereço da WSDL para o sistema 1C:

http://demo.omnicomm.ru:8001/AnalyticalServer/v2/ws?wsdl

Login: rudemoru Senha: rudemo123456

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/demo/soap

Last update: **2026/11/16 13:41**

Informações gerais

A integração com a Omnicomm Online é usada para ampliar a funcionalidade de sistemas de terceiros e automatizar o recebimento de dados obtidos pelo Omnicomm Online nos sistemas de gestão de frotas, combustível, ERP’s e TMS.

Estão disponíveis os seguintes métodos de integração:

- REST API
- por meio do protocolo SOAP

Recomendamos usar o REST API para fazer a integração. Não é possível expandir os recursos de integração com o protocolo SOAP.

O módulo de integração para o sistema Omnicomm deve ser desenvolvido e implementado pelos desenvolvedores de sistemas dos terceiros.

A Omnicomm não realiza a delegação a especialistas nem o desenvolvimento do módulo de integração para sistemas de terceiros.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/general-information

Last update: **2026/11/16 13:41**

Principais recursos

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/opportunity

Last update: **2026/11/16 13:41**

REST API

REST API facilita a automatização de:

- gestão de usuários
- gerenciamento do perfil de veículos
- gerenciamento de cercas eletrônicas
- recepção de relatórios
- recepção de notificações
- uso do subsistema OVMS (serviço de vídeo)

A REST API possibilita o início rápido da coleta, processamento e monitoramento de dados dos veículos.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/opportunity/rest

Last update: **2026/11/16 13:41**

SOAP

Dados baixados do sistema Omnicomm Online permitem agregar valor agregado no que tange gestão de combustível e frotas aos sistemas de terceiros. As seguintes tarefas podem ser executadas automaticamente:

- Inserção de dados sobre consumo de combustível e quilometragem nas funcionalidades relacionadas a desempenho operacional, rota, estatísticas e dentre outras
- Estatística de quilometragem e tempo de uso, para fins de cálculo de serviço, remuneração de motoristas etc.
- Estatística de equipamento acessório, para fins de cálculo do escopo do trabalho executado
- Estatística de quilometragem, horas de funcionamento do motor e equipamento auxiliar, para fins de planejamento de manutenção
- Análise do combustível consumido
- Conferência de volume de reabastecimento documentado com os dados de uso reais para fins de evitar roubo de combustível
- Geração de relatórios de violação
- Planejamento x análise real do trabalho executado, combustível utilizado etc.
- Uso da localização atual do veículo para determinação da unidade ideal para execução de ordem de serviço
- Visualização de rastreamento no mapa, usando software de terceiros

Diversos relatórios podem ser gerados usando os parâmetros veiculares baixados do Omnicomm Online.

O sistema de terceiro ao se integrar com o webservice do Omnicomm Online será capaz de exibir as ações ilícitas que os colaboradores que trabalham diretamente com a frota possam vir a cometer, tal como o roubo de combustível e a utilização não autorizada de veículos.

Os principais sistemas de destino para integração são sistemas contábeis, sistemas ERP, sistemas de contabilidade do setor.

O protocolo SOAP não suporta o uso com o subsistema OVMS (serviço de vídeo). Use o REST API para uma integração ágil com o subsistema OVMS.

Os webservices proporcionam a obtenção dos seguintes parâmetros:

- Quilometragem
- Consumo de combustível
- Tempo de funcionamento do motor
- Tempo de funcionamento de equipamentos auxiliares
- Gráfico de volume de combustível
- Quilometragem com velocidade excedida*
- Duração de deslocamento*
- Tempo de funcionamento do motor em deslocamento*
- Funcionamento do motor em ociosidade*
- Tempo de «motor desligado»*

- Nível de combustível em um determinado momento*
- Consumo de combustível em deslocamento*
- Consumo de combustível em ociosidade*
- Taxa de consumo de combustível por uma hora de operação do motor*
- Rastreamento*
- Localização real*

* – Parâmetros marcados estão disponíveis na versão 2.4.2.

Além desses parâmetros, é possível receber todos os eventos exibidos no relatório «Eventos» no Omnicomm Online.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/opportunity/soap

Last update: **2026/11/16 13:41**

REST API

A especificação do método API pode ser encontrada em https://developers.omnicomm-world.com

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/rest_api

Last update: **2026/11/16 13:41**

Obtenção dos direitos de uso do REST API

Por favor, entre em contato com o suporte técnico Omnicomm escrevendo por e-mail para support@omnicomm-world.com para obter os direitos de uso do REST API

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/rest_api/access

Last update: **2026/11/16 13:41**

Gestão de usuários

A gestão de usuários abrange a adição, exclusão e bloqueio dos usuários no Omnicomm Online, além da obtenção de uma lista de todos os usuários do Omnicomm Online.

Um usuário autorizado pode acessar os dados do veículo.

Um sistema externo pode desempenhar as tarefas em nome de um usuário após fazer login com os seus dados de acesso (após obter o JWT com o direito do usuário).

Descrição dos métodos de gestão do usuário: https://developers.omnicomm-world.com/#/Users/

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/rest_api/add_user

Last update: **2026/11/16 13:41**

Gerenciamento de veículos

É necessário adicionar um perfil do veículo para habilitar o processamento dos dados do veículo na Omnicomm Online.

Um veículo é identificado por meio de um identificador único universal (UUID), o qual é atribuído quando ao adicionar-se um novo perfil de veículo no Omnicomm Online, ou através do número de identificação do terminal.

Os veículos podem ser adicionados a grupos, e o mesmo veículo pode pertencer a mais de um grupo. Também é possível configurar o acesso aos grupos de veículos pelos usuários. Os grupos de veículos são criados ao adicionar os usuários, bem como na interface do Omnicomm Online.

Descrição dos métodos de gerenciamento dos veículos:
https://developers.omnicomm-world.com/#/Vehicles/

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/rest_api/add_vehicle

Last update: **2026/11/16 13:41**

Autorização

Deve ser indicado um token JWT no cabeçalho de autorização ao acessar os métodos REST API (exceto para os métodos de autorização). Este token JWT concede o direito de uso do REST API.

Formato do JWT: JWT<espaço><JWT recebido pelo método de autorização>

Exemplo:

Autorização: JWT eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MTU3MDM1OTEsImxvZ2luIjoiYWR0…4ifQ.0I0CXcwWWxZWARE0eUEPOAvKd0prW_Uf0jbOMLnd5SI

O dia de expiração do JWT é indicado no atributo payload exp no formato de registro de hora Unix em UTC. Após a expiração aparecerá o erro 401 ao tentar acessar o método.

O JWT pode ser obtido usando o método POST /auth/login?jwt=1 ou, ao expirar, post /auth/refresh indicando o cabeçalho renovado de autorização obtido pelo JWT durante a autorização de post /auth/login?jwt=1.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/rest_api/authorization

Last update: **2026/11/16 13:41**

Gerenciamento de cercas eletrônicas

As cercas eletrônicas são áreas virtuais no mapa criadas pelos usuários no Omnicomm Online. Ao criar uma cerca eletrônica, especifique uma forma (polígono, círculo ou linha) e suas coordenadas geográficas.

As cercas eletrônicas são usadas para monitorar a localização do veículo (entrando ou saindo da cerca eletrônica) e outros parâmetros de operação, como a velocidade do veículo.

Descrição dos métodos de gerenciamento das cercas eletrônicas:
https://developers.omnicomm-world.com/#/Geozones/

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/rest_api/geofence

Last update: **2026/11/16 13:41**

Tratamento das notificações

As notificações são usadas para notificar imediatamente os usuários sobre os eventos gravados.

Descrição dos métodos de tratamento das notificações:
https://developers.omnicomm-world.com/#/Notifications/

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/rest_api/notifications

Last update: **2026/11/16 13:41**

Relatórios

Os relatórios são usados para obter diversas informações sobre a operação do veículo.

Descrição dos métodos de obtenção dos relatórios:
https://developers.omnicomm-world.com/#/Reports/

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/rest_api/report

Last update: **2026/11/16 13:41**

Restrições

Há restrições sobre a intensidade das solicitações que podem ser enviadas para o REST API do Onmicomm Online para protegê-lo contra ataques DoS e erros de sistemas de terceiros.

Quando essas restrições são extrapoladas, todas as solicitações feitas ao REST API provenientes do endereço IP ou usuário correspondente serão bloqueadas.

Tentativas de autorização malsucedidas

Não superiores a 10 por minuto pelo mesmo endereço IP.

Chamadas autorizadas

Não superiores a 180 por minuto do mesmo usuário.

Chamadas não autorizadas

Não superiores a 60 por minuto pelo mesmo endereço IP.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/rest_api/restriction

Last update: **2026/11/16 13:41**

Serviço de vídeo Omnicomm

O serviço de vídeo Omnicomm abrange o gerenciamento de terminais de vídeo e fornece ao usuário material em vídeo.

Características principais do serviço:

- receber, armazenar, modificar e fornecer dados dos perfis dos terminais de vídeo
- receber, armazenar, modificar e fornecer parâmetros da tarefa para o download de arquivos de vídeo
- executar tarefas de download de arquivos de vídeo

Descrição dos métodos para gestão do serviço de vídeo:
https://developers.omnicomm-world.com/#/VideoService/

Obtendo um trecho de vídeo

Esta seção descreve como usar o serviço de vídeo num cenário típico.

Para obter um trecho de vídeo:

1. Faça login na conta de um revendedor ou usuário com direitos sobre o veículo e direitos de uso do serviço de vídeo: POST /auth/login?jwt=1

2. Obter o perfil do veículo a partir do ID do terminal do veículo: GET /service/ovms/api/profiles

3. Obter o arquivo de vídeo:

- Crie uma tarefa para obter o trecho de vídeo com base no perfil de vídeo recebido: POST /service/ovms/api/tasks
- Verificar periodicamente o status da tarefa: GET /service/ovms/api/tasks/{task_id}
- Após receber o status «pronto» da tarefa, solicite o arquivo de vídeo:GET /service/ovms/api/tasks/file/{task_id}

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/pt/omnicomm_online-integration_api/rest_api/video

Last update: **2026/11/16 13:41**