

SmartTruckRoute External Routing API

Version: July 2, 2025

Base Endpoint: <https://main.smartcarroute.com/scripts/str-route-external/api.php>

1 Introduction

This document describes how third-party developers can request truck, RV, bus, car, or walking routes from the SmartTruckRoute (STR) External Routing wrapper script. All requests are made over HTTPS using a JSON POST body and require a publishable API key provided by SmartTruckRoute.

2 Quick Start

1. Obtain a ****publishable API key**** (prefix ``pk_``) from SmartTruckRoute support.
2. Send a JSON POST request to the endpoint with the ``Authorization: Bearer <your_key>`` header.
3. Parse the JSON response and extract the KML string containing the route.

Example cURL:

```
curl -X POST https://main.smartcarroute.com/scripts/str-route-external/api.php \
  -H "Content-Type: application/json" \
  -H "Authorization: Bearer pk_live_your_key_here" \
  -d @request.json
```

3 Authentication

- ****Header:**** ``Authorization: Bearer <publishable_api_key>``
- Keys beginning with ``sk_`` are private and will be rejected.
- Requests without this header will receive ****HTTP 400****.

4 Request Format

HTTP Method: ****POST**** (JSON payload)

Content-Type: ``application/json``

Body fields are divided into required and optional parameters.

5 Parameter Reference

Parameter	Type	Required	Description	Allowed / Range
action	string	Yes	Must be `GetRequest`	N/A
start_longitude	string (deg)	Yes	Start point longitude	-180 to 180
start_latitude	string (deg)	Yes	Start point latitude	-90 to 90
end_longitude	string (deg)	Yes	Destination longitude	-180 to 180
end_latitude	string (deg)	Yes	Destination latitude	-90 to 90
route_type	int	Yes	Routing profile code	See §6
route_size	int	Yes	Number of alternate routes to return	1 or 2
route_avoid_tolls	int	Yes	Toll avoidance level	0,1,2
route_avoid_tunnels	int	Yes	Tunnel restriction	0,1
route_avoid_weight_stations	int	Yes	Weigh-station avoidance	0-3

Vehicle Parameters

Parameter	Type	Required	Description	Allowed / Range
vehicle_height	object	Yes	Overall height	0-999 ft (in)
vehicle_length	object	Yes	Overall length	0-999 ft (in)
vehicle_width	object	Yes	Overall width	0-999 ft (in)

vehicle_weight	int (lbs)	Yes	Gross weight	1–1,000,000
vehicle_axles	int	Yes	Number of axles	2–33
vehicleTrailers	int	Yes	Trailer count	0–3
vehicle_fifth_wheel	int	Yes	Has 5th-wheel?	0 or 1
vehicle_hazmat	int	Yes	Hazmat class	0 or 1–9
vehicle_speed	int (mph)	Yes	Desired speed	0–199
vehicle_bearing	int	Yes	Initial heading; –1 to ignore	–1 or 0–360

Optional Parameters

Parameter	Type	Required	Description	Allowed / Range
route_via	array	No	Ordered waypoints	lat/lon
route_avoid	array	No	Points to avoid	lat/lon
route_area_avoid	array	No	Circular areas to avoid	lat/lon + radius m

6 Route Type Codes

Profile	Code
Truck Quickest	8
Truck Quickest Ferry	9
Truck Shortest	10
Truck Freeways	7
Truck Avoid Freeways	24
Truck Emergency	11
RV Quickest	13

RV Quickest Ferry	23
RV Shortest	14
RV Freeways	15
RV Avoid Freeways	20
RV Emergency	13
Bus Quickest	16
Bus Quickest Ferry	22
Bus Shortest	17
Bus Freeways	18
Bus Avoid Freeways	19
Bus Emergency	16
Car Quickest	5
Car Quickest Ferry	21
Car Shortest	2
Car Freeways	4
Car Avoid Freeways	3
Car Emergency	5
Walking Path	6

7 Sample Request JSON

```
{
  "action": "GetRequest",
  "start_longitude": "-86.792916",
  "start_latitude": "41.736715",
  "end_longitude": "-87.401552",
  "end_latitude": "41.572642",
  "route_type": "8",
  "route_size": "1",
  "route_avoid_tolls": "0",
  "route_avoid_tunnels": "0",
  "route_avoid_weigh_stations": "0",
  "vehicle_height": {"feet": "13", "inches": "6"},
  "vehicle_length": {"feet": "53", "inches": "0"},
}
```

```

"vehicle_width": {"feet": "8", "inches": "6"},
"vehicle_weight": "80000",
"vehicle_axles": "5",
"vehicle Trailers": "1",
"vehicle_fifth_wheel": "1",
"vehicle_hazmat": "0",
"vehicle_speed": "0",
"vehicle_bearing": "-1",
"route_via": [
  {"latitude": "41.569692", "longitude": "-87.389374"},
  {"latitude": "41.574811", "longitude": "-87.393835"}
],
"route_avoid": [
  {"latitude": "41.57534767037292", "longitude": "-87.40810220025094"}
],
"route_area_avoid": [
  {"latitude": "41.573940", "longitude": "-87.455485", "radius": "100"},
  {"latitude": "41.593010", "longitude": "-87.432888", "radius": "100"}
]
}

```

8 Response

Successful responses return ****HTTP 200**** with JSON:

```

{
  "code": 200,
  "message": "success",
  "data": {
    "kml": "<KML string>"
  },
  "copyright": "© 2025 SmartTruckRoute"
}

```

`kml` is a KML string beginning with `` that can be saved to a `.kml` file and opened in compatible GIS or mapping applications.

Sample KML file: <https://www.smarttruckroute.com/sample.kml>

9 Error Codes

Parameter	Type	Required	Description	Allowed / Range
400	Bad Request		Missing/invalid parameters or auth header	

402	Payment Required	Internal mis-configurati on
403	Forbidden	Subscription expired
429	Too Many Requests	Rate-limit (throttle) exceeded
500	Internal Server Error	Unexpected server error

10 Rate Limiting

Requests are throttled per API key. If you exceed the allowed rate, the server returns ****HTTP 429****. Contact support to increase your quota.

11 Code Examples

Python (requests):

```
import requests, json

url = "https://main.smartcarroute.com/scripts/str-route-external/api.php"
headers = {
    "Authorization": "Bearer pk_live_your_key",
    "Content-Type": "application/json"
}
payload = {...} # JSON from §7
r = requests.post(url, headers=headers, data=json.dumps(payload))
print(r.json())
```