# Trip.com Persisted Query Analysis and SHA-256 Hash Generation

Assignment Report

—--

**Submitted By:**

Prajwal Naik

naikprajwal089@gmail.com

—--

## Overview:

This report details the analysis of the persisted query generation process used by Trip.com and provides a Python script for generating and validating the SHA-256 hash used in the persisted query. The report includes a step-by-step breakdown of the approach, methodology, and code implementation.

**Table of Contents:**

## Report: Generating the Persisted Query on Trip.com

### Objective

You need to understand how Trip.com creates a special code (SHA-256 hash) for their flight search data and write a Python script to generate and validate this code.

### Steps:

1. **Look at the cURL Command:**
    - The cURL command is a way to request information from Trip.com.
    - It includes headers and a data payload in JSON format.
    - The key part to focus on is the data payload, which contains flight search details.
2. **Understand the Data Payload:**
    - The payload includes details like departure city, arrival city, and trip type.
    - Inside the payload, there is an extensions section with a persistedQuery that has a sha256Hash.
3. **Generate the SHA-256 Hash:**
    - The sha256Hash is created from the request part of the payload.
    - To generate this hash, we need to:
        1. Convert the request section into a JSON string.
        2. Ensure the JSON string is formatted consistently (no extra spaces and keys sorted).
        3. Use the SHA-256 algorithm to create a hash from this JSON string.
4. **Compare and Validate the Hash:**
    - Compare the generated hash with the provided sha256Hash.
    - If they match, the data is valid; if not, something is different.

`

**Python Script Explanation:**

1. **Import Necessary Modules**:
   - The code starts by importing two modules: hashlib and json.
   - hashlib is used to create secure hash values.
   - json is used to convert Python dictionaries into JSON strings.

2. **Define Function to Generate SHA256 Hash**:
   - A function named generate_sha256_hash is created.
   - This function takes a dictionary as input.
   - Inside the function:
     - The dictionary is turned into a JSON string using json.dumps.
     - The JSON string is formatted to have no extra spaces and the keys are sorted.
     - The JSON string is then converted to bytes.
     - These bytes are hashed using the SHA256 algorithm from hashlib.
     - The resulting hash is turned into a hexadecimal string and returned.

3. **Define the Data Payload**:
   - A dictionary called data_payload is created.
   - This dictionary contains details like:
     - The name of the operation ("routeInfo").
     - Variables related to searching for a route.
     - Search criteria including departure and arrival cities.

4. **Generate the SHA256 Hash for the Data Payload**:
   - The function generate_sha256_hash is called with data_payload as the argument.
   - The resulting hash is stored in a variable named generated_hash.
   - This generated hash is then printed to the console.

5. **Compare the Generated Hash with a Provided Hash**:
    - A predefined SHA256 hash called provided_hash is given.
    - The code checks if the generated hash matches this provided hash.
    - The result of the comparison is stored in a variable called is_valid.
    - If the hashes match, is_valid will be True; otherwise, it will be False.
    - A message is printed to show whether the generated hash is valid or not.

## Conclusion

In this assignment, I reverse-engineered the `sha256Hash` used in Trip.com's flight search queries. By analyzing network requests, I identified key parameters and created a Python script to generate and validate the hash. This process helped me understand how API interactions work and how to ensure their security.

## References

- hashlib — Secure hashes and message digests
    - https://docs.python.org/3/library/hashlib.html
- API
    - https://www.geeksforgeeks.org/what-is-an-api/
- Browers developer tools
    - https://www.geeksforgeeks.org/browser-developer-tools/