Customer API Security Analysis

I’ve analysed the security of the Customer API and made appropriate fixes to the found issues. The issues are similar to the Operator API security analysis made before. This document will be updated if more issues are found later.

Firstly, the most important feature Customer API was lacking was authentication. Anyone who knew the URL of this API could just query data from all the clients, which is obviously a huge flaw. In order to fix this I’ve implemented Django authentication, where a token is needed to perform queries. A user gets the token only if he logs in to the query tool, which enhances the security and solves this problem.

Another problem I’ve noticed was that the query tool would display queried data even if the user was logged out. This is a huge problem because people who don’t have access to the query tool can still query data if they know the URL of the query. It is enough to look at someone’s browser history, or simply guess the URL and the data is leaked. I’ve fixed this by adding specific checks in the query tool, so that it only displays data if the user is authenticated.

Further issue was that there are no limits to the number of queries made to the Customer API. This is a potential problem as it could fall victim to the DoS (denial-of-service) attack. To tackle this problem I’ve implemented Django REST framework Throttling, which adds a feature to limit the requests made to the API per time unit. Currently I’ve made it 200 requests per day to match the Operator API, but this is a subject to change.

Regarding the man in the middle attack, it shouldn’t cause any trouble as the API is hosted on Pythonanywhere which is using HTTPS. If the man in the middle would be able to get the data, it would be safely encrypted and wouldn’t make much sense to them.

Another potential vulnerability is the SQL injection attack. However, Django generates SQL automatically using parameterization, and we’re not using raw SQL anywhere, so it shouldn’t cause any problems in that regard.