GENROCKET CASE STUDY

Test Data Generation for Financial Services

BACKGROUND

A global payment processing and Point of Sale (POS) technology company needed a secure and efficient test data provisioning process. The company serves small, medium and large-scale businesses and processes over 79 billion transactions annually for ecommerce and point-of-sale environments in 118 countries.

Their vision is to help their clients grow through innovation and the company leads by example with innovations in mobile solutions, advanced POS systems, data analytics and security solutions.

When the QA team asked GenRocket to help streamline their process for preparing test data, they chose GenRocket's *Test Data Generation* solution based on its speed, simplicity, data quality and use of synthetic data to ensure compliance with GDPR security regulations.

GDPR defines the security requirements for any company doing business in the European Union (EU). For financial services companies, compliance is particularly important as violations may carry penalties up to 4% of global revenue. GDPR ensures data privacy and protection for all vendors and consumers and serves to build customer loyalty and trust. As a result, financial services companies around the world are acting to comply with this important standard.

THE TECHNICAL CHALLENGE

To provision its test data, the QA team would mine data from the production database based on the results of functional testing to access merchant and authorization data and then manually add personal data (e.g., usernames and addresses) to avoid the possibility of exposing private information. The test data was then populated in an excel spreadsheet to be used by the automation framework for inputs and validation assertions.

Using this process, data preparation was taking approximately 6 hours and was repeated every 2 weeks. The QA team was looking to replace this inefficient process with a fully automated solution for provisioning the test data required by their test automation framework.

GENROCKET'S TEST DATA GENERATION SOLUTION

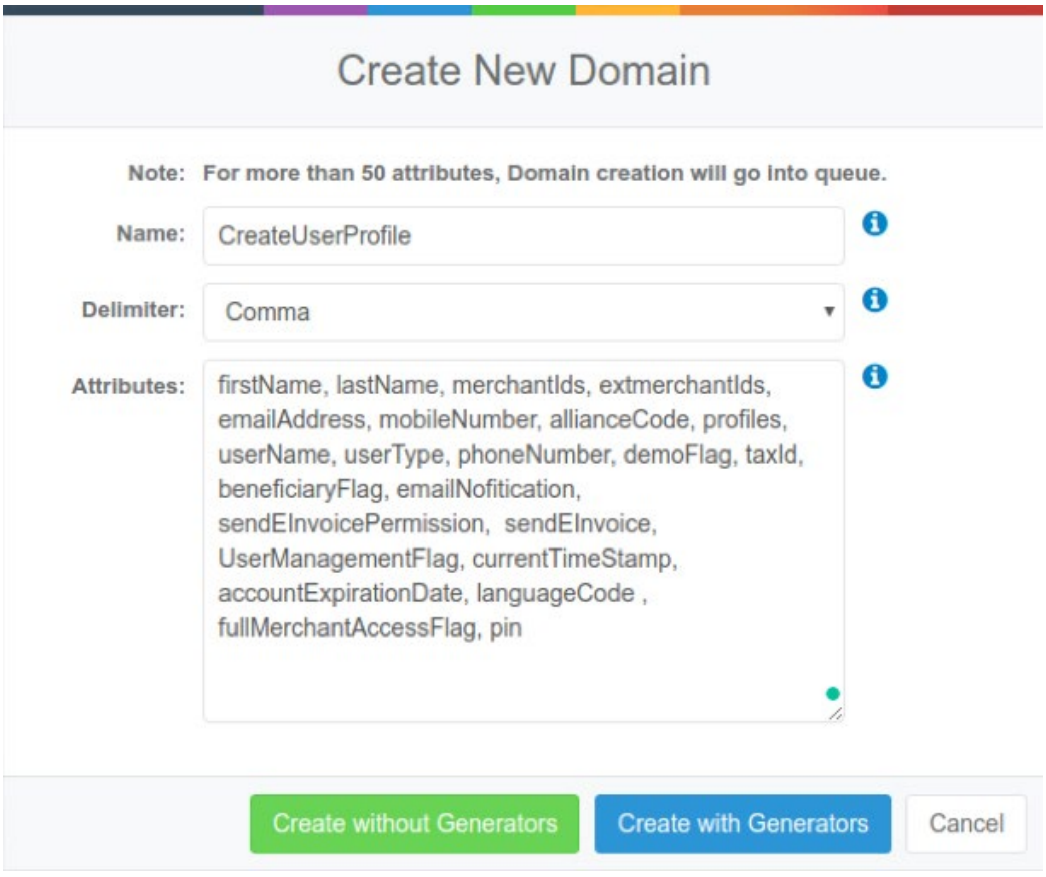
The QA team initially requested GenRocket to provide a solution that would automatically populate the Excel spreadsheet with test data – eliminating a manual data creation process. A script would be created to read the Excel spreadsheet and convert the generated data into a JSON file that can be consumed by the test automation tool.

However, it was determined that multiple Excel spreadsheets, each with many columns, would be required to accommodate the variety of test cases and their test data requirements. Together, GenRocket and the QA team decided this approach would not fully achieve the goal of building a fully automated, time-efficient solution.

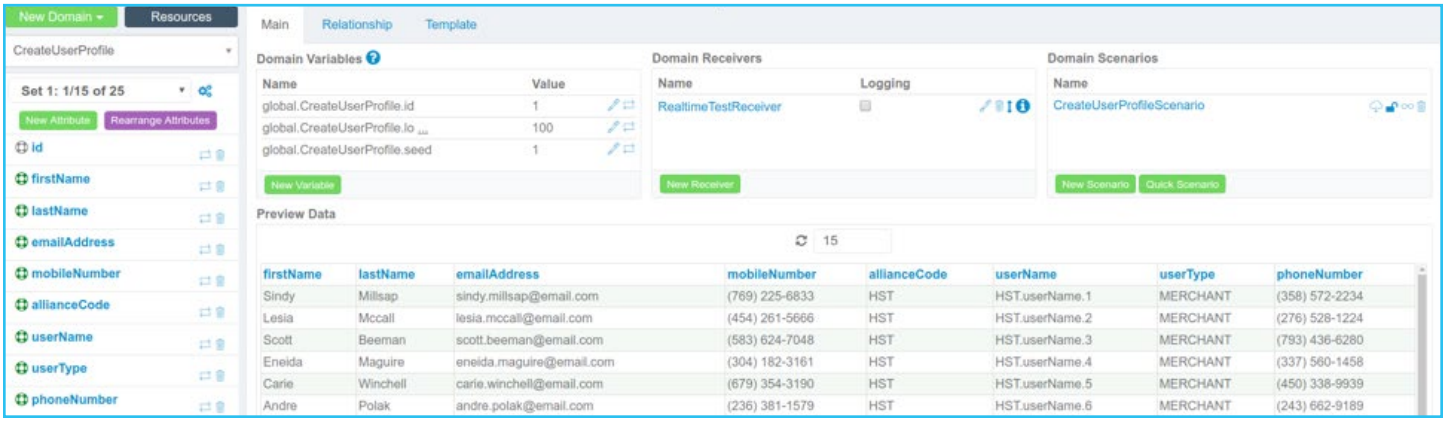
GenRocket developed an alternate approach using the *RealTimeTestReceiver*, a component of the GenRocket *Test Data Generation* platform. GenRocket *Receivers* format the data produced by GenRocket *Generators* according to test data *Scenarios*. The *RealTimeTestReceiver* has the ability to pass data via RESTful web services and inject test data directly into the automation tool, thus eliminating the need for spreadsheets.

HOW IT WORKS

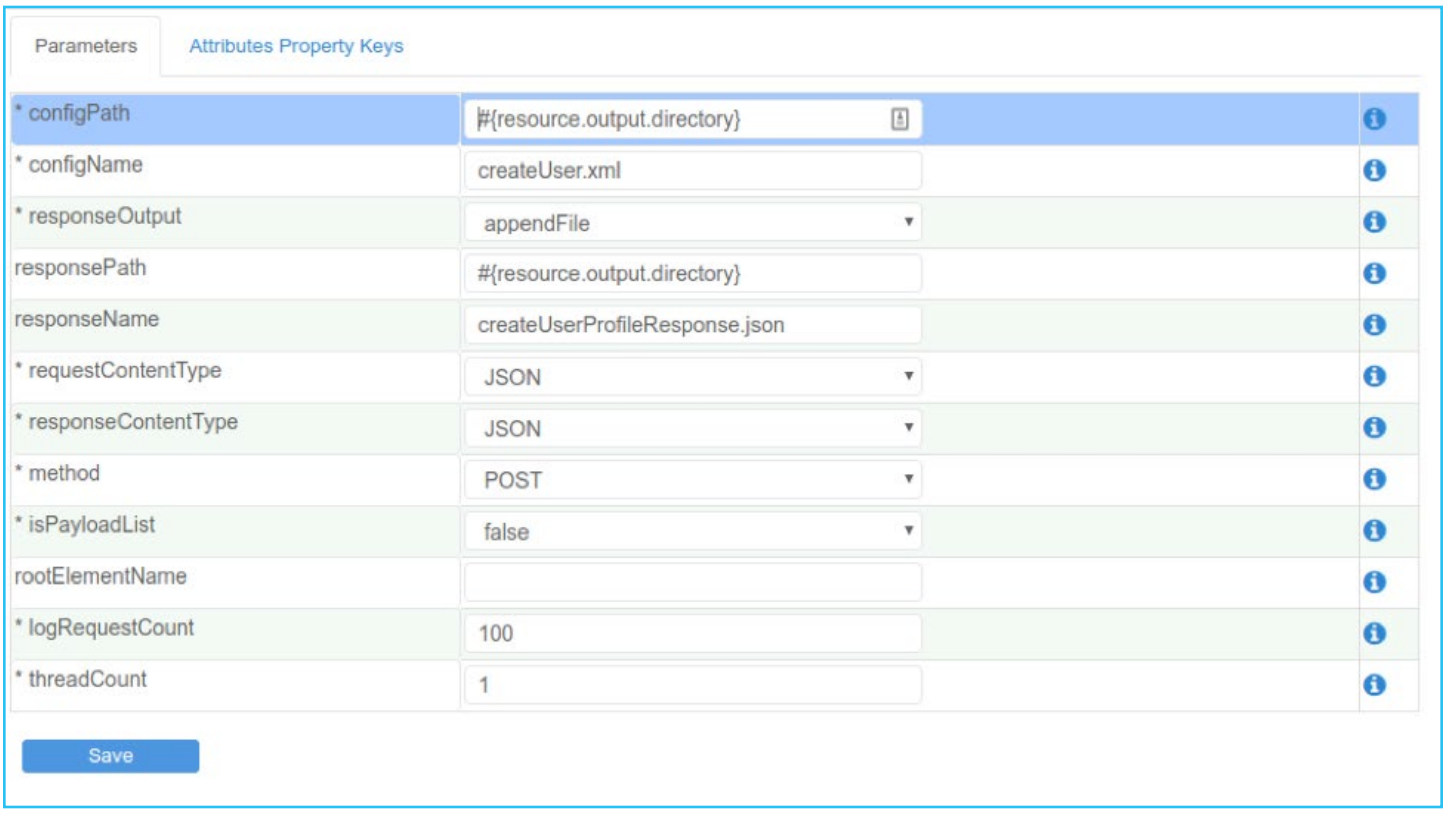
GenRocket's "Scratch Pad" option was used to quickly define a *Domain* and its required test data *Attributes* by dropping in a set of comma delimited values into the editor. In the screenshot below, the Domain "CreateUserProfile" is created using the Scratch Pad option.



The form below shows the *Domain* and *Attributes* created from the Scratchpad. The appropriate *Generators* were added to the Domain's Attributes to generate the desired data.



The form below shows the *RealTimeTestReceiver* Parameters that define the behavior of the REST calls. The *RealTimeTestReceiver* is associated with the Domain to make REST calls.



The file below shows a sample configuration file. When the *Scenario* is loaded by the GenRocket Runtime Engine, the *Receiver* reads a configuration file, saved on the local computer, to get the *Endpoint* for the request.

```
<config>
<requestURL>https://localhost:8443/home/test1</requestURL>
</config>
```

The JSON envelope below shows a sample output generated by the RealTimeTestReceiver.

```
{
 "fullMerchantAccessFlag": 0,
 "profiles": [
  "BUSINESSOWNER"
 ],
 "firstName": "EMS.John",
 "lastName": "Smith",
 "userName": "ems.john.smith",
 "emailAddress": "ems.john.smith@test.com",
 "mobileNumber": "31990099001",
 "phoneNumber": "31990099002",
 "boUserManagementFlag": 1,
 "userType": "MERCHANT",
 "allianceCode": "HST",
 "merchantIds": [
  10000000
 ]
}
```

IMPACT

Using GenRocket's *Test Data Generation* platform, this financial services company was able to meet all of their test data challenges and realized the following important benefits:

- Reduced a six-hour data generation process to a few seconds.
- Move from a manual process to an automated "zero touch" process
- Ensure that all test cases and data formats are fully supported
- Maintain total security for compliance with data privacy regulations (GDPR).