Design of a Unique ID Generator

Learn how to design a system that generates a unique ID.

We'll cover the following

In the previous lesson, we saw that we need unique identifiers for many use cases, such as identifying objects (for example, Tweets, uploaded videos, and so on) and tracing the execution flow in a complex web of services. Now, we'll formalize the requirements for a unique identifier and discuss three progressively improving designs to meet our requirements.

Requirements for unique identifiers

The requirements for our system are as follows:

- **Uniqueness**: We need to assign unique identifiers to different events for identification purposes.
- **Scalability**: The ID generation system should generate at least a billion unique IDs per day.

- **Availability**: Since multiple events happen even at the level of nanoseconds, our system should generate IDs for all the events that occur.
- **64-bit numeric ID**: We restrict the length to 64 bits because this bit size is enough for many years in the future. Let's calculate the number of years after which our ID range will wrap around.

  Total numbers available = 2^{64} = 1.8446744 x 10^{19}

  Estimated number of events per day = 1 billion = 10^9

  Number of events per year = 365 billion = 365×10^9

  Number of years to deplete identifier range = $\frac{2^{64}}{365 \times 10^9}$ = 50,539,024.8595 years

  64 bits should be enough for a unique ID length considering these calculations.

Let's dive into the possible solutions for the problem mentioned above.