Random number based UUID (IETF RFC 4122 version 4)

By default, Hibernate uses a random number based generation strategy. As always, you don't have to do much to get the default behaviour. You just need to add a *@GeneratedValue* annotation that references the ID generator to your primary key attribute and define the generator with one of Hibernate's *@GenericGenerator* annotations. The *@GenericGenerator* annotation requires 2 parameters, the name of the generator and the name of the class that implements the generator. In this case, it's the *org.hibernate.id.UUIDGenerator*.

```
@Entity
public class Book {


    @Id
    @GeneratedValue(generator = "UUID")
    @GenericGenerator(
        name = "UUID",
        strategy = "org.hibernate.id.UUIDGenerator",
    )
    @Column(name = "id",
            updatable = false,
            nullable = false)
    private UUID id;


    ...
}
```

IP and timestamp based UUID (IETF RFC 4122 version 1)

Hibernate can also generate a UUID based on IETF RFC 4122 version 1. Based on the specification, the UUID should be created with the MAC address and a timestamp. But Hibernate uses the IP address instead of the MAC address. In general, this is not an issue. But if the components of your distributed system are running in different networks you need to make sure that they use unique IP addresses.

The configuration of the IETF RFC 4122 version 1 based UUID generator requires an additional parameter that defines the generation strategy. You can see an example of it in the following code snippt. You just need to provide an additional *@Parameter* annotation with the name *uuid_gen_strategy_class* and the fully qualified classname of the generation strategy as the value.

```java
@Id
@GeneratedValue(generator = "UUID")
@GenericGenerator(

    name = "UUID",
    strategy = "org.hibernate.id.UUIDGenerator",
    parameters = {

        @Parameter(

            name = "uuid_gen_strategy_class",
            value =
            "org.hibernate.id.uuid.CustomVersionOneStrategy"

        )

    }

)
@Column(name = "id", updatable = false,

        nullable = false)
private UUID id;
```