

Create a backend

As we said, a backend is the combination of a transport and a strategy. That way, you can create a complete strategy exactly fitting your needs.

For this, you have to use the `AuthenticationBackend` class.

```
from fastapi_users.authentication import AuthenticationBackend, BearerTransport, JWTStrategy

SECRET = "SECRET"

bearer_transport = BearerTransport(tokenUrl="auth/jwt/login")

def get_jwt_strategy() -> JWTStrategy:
    return JWTStrategy(secret=SECRET, lifetime_seconds=3600)

auth_backend = AuthenticationBackend(
    name="jwt",
    transport=bearer_transport,
    get_strategy=get_jwt_strategy,
)
```

As you can see, instantiation is quite simple. It accepts the following arguments:

- `name (str)`: Name of the backend. Each backend should have a unique name.
- `transport (Transport)`: An instance of a `Transport` class.
- `get_strategy (Callable[..., Strategy])`: A dependency callable returning an instance of a `Strategy` class.

Next steps

You can have as many authentication backends as you wish. You'll then have to pass those backends to your `FastAPIUsers` instance and generate an auth router for each one of them.

