3. Data Hiding

Design a class called **Employee** that represents an employee. Implement encapsulation and data hiding to protect the attributes of the employee. The class should include the following methods and attributes:

Attributes (private):

- __name (string): representing the name of the employee.
- __salary (float): representing the salary of the employee.

Methods:

- __init__(self, name, salary): Initialize the name and salary attributes.
- get_name(self): Return the name of the employee.
- get_salary(self): Return the salary of the employee.
- set_name(self, new_name): Update the name of the employee.
- set_salary(self, new_salary): Update the salary of the employee.

```
employee = Employee("John Doe", 5000.0)
print(employee.get_name())  # Output: "John Doe"
print(employee.get_salary())  # Output: 5000.0

employee.set_name("Jane Smith")
employee.set_salary(6000.0)
print(employee.get_name())  # Output: "Jane Smith"
print(employee.get_salary())  # Output: 6000.0

# Accessing the private attributes directly raises an AttributeError
print(employee.__name)
# Output: AttributeError: 'Employee' object has no attribute '__name'
print(employee.__salary)
# Output: AttributeError: 'Employee' object has no attribute '__salary'
```