

1 Grundlegendes

Im Rahmen dieser Live-Aufgabe soll mit Hilfe des Visitor-Patterns ein „Hacking-Tool“ für unterschiedliche Router-Typen erstellt werden:

2 Erstellen Sie folgende Klassen/Interfaces

2.1 Interface Router

Ein Router enthält die Methode „void sendData(String data);“.

2.2 Drei Router-Implementierungen

Erstellen Sie drei Router-Implementierungen (z.B. benannt nach unterschiedlichen Herstellern). Diese implementieren die #sendData(String)-Methode zunächst identisch:

Die übergebenen Daten werden auf der Konsole ausgegeben.

2.2.1 Sicherheitslücken

Jeder der drei Router-Implementierungen erhält nun eine Sicherheitslücke: Wenn ein bestimmter (von Ihnen frei gewählter) String an den Router geschickt wird, gilt dieser als gehackt. Dies wird durch eine entsprechende Ausgabe auf der Konsole verdeutlicht.

Erweitern Sie also die Methoden-Implementierungen der drei Router-Klassen und geben Sie jeweils eine entsprechende Ausgabe auf der Konsole aus, sofern der herstellerspezifischer String übergeben wird.

3 Visitor-Pattern

3.1 HackingVisitor

Mit Hilfe des Visitor-Patterns soll nun „HackingVisitor“ erstellt werden. Dieser Visitor übergibt je nach Hersteller den bestimmten String an die sendData(String)-Methode und hackt somit diesen Router.

3.2 Unit-Test / Main-Methode

Erstellen Sie einen Unit-Test oder eine Main-Klasse welches das Verhalten des Visitors verdeutlicht.