**Guião de Demonstração**

**Instalação**

1. Obter o código do repositório Git com a tag “SD\_P3”:

(<https://github.com/tecnico-distsys/A47-SD18Proj>)

1. Aceder à diretoria base do código “\A47-SD18Proj\”
2. Correr o comando “mvn clean install -DskipTests”

**F1: Funcionamento normal da segurança**

* station-ws

1. Aceder à diretoria “\A47-SD18Proj\station-ws”
2. Abrir um novo terminal.
3. Correr comando “mvn exec:java”
4. Abrir um novo terminal.
5. Correr comando “mvn exec:java -Dws.i=2”
6. Abrir um novo terminal.
7. Correr comando “mvn exec:java -D ws.i=3”

* binas-ws

1. Aceder à diretoria “\A47-SD18Proj\binas-ws”
2. Correr comando “mvn clean compile exec:java”

* binas-ws-cli

1. Aceder à diretoria “\A47-SD18Proj\binas-ws-cli”
2. Correr comando “mvn clean compile exec:java”

**Caso F2: Resistência a ataque**

* + - 1. Aceder à diretoria “\A47-SD18Proj\binas-ws-cli\src\jaxws”
      2. Abrir ficheiro “binas-ws-cli\_handler”
      3. Descomentar a tag presente na linha 15.
* station-ws

1. Aceder à diretoria “\A47-SD18Proj\station-ws”
2. Abrir um novo terminal.
3. Correr comando “mvn exec:java”
4. Abrir um novo terminal.
5. Correr comando “mvn exec:java -Dws.i=2”
6. Abrir um novo terminal.
7. Correr comando “mvn exec:java -D ws.i=3”

* binas-ws

1. Aceder à diretoria “\A47-SD18Proj\binas-ws”
2. Correr comando “mvn clean compile exec:java”

* binas-ws-cli

1. Aceder à diretoria “\A47-SD18Proj\binas-ws-cli”
2. Correr comando “mvn clean compile exec:java”

O ataque neste caso é alteração de mensagem, pelo que vai falhar no MACHandler na parte do servidor. Se alterar o MaliciousHandler para antes do MACHandler na cadeia de handlers, é possível verificar a falha no BinasAuthorizationHandler.