

Caso S1:

1. Clonar projeto (<https://github.com/tecnico-distsys/A37-SD18Proj>)
2. Entrar na diretoria do projeto
3. Instalar kerby-lib
 - 3.1. `cd kerby`
 - 3.2. `mvn clean install -DskipTests`
4. Instalar station-ws-cli
 - 4.1. `cd ../stations-ws-cli`
 - 4.2. `mvn clean install -DskipTests`
5. Instalar os *handlers*
 - 5.1. `cd ../ws-handlers`
 - 5.2. `mvn install -DskipTests`
6. Correr uma estação
 - 6.1. `mvn exec:java`
7. Abrir outro terminal, aceder à diretoria do projeto e correr o binas
 - 7.1. `cd binas-ws`
 - 7.2. `mvn compile exec:java`
8. Abrir outro terminal, aceder à diretoria do projeto e correr o binas-ws-cli
 - 8.1. `cd binas-ws-cli`
 - 8.2. `mvn compile exec:java`

Caso S2:

Proteção contra alterações de conteúdo do *body* da mensagem SOAP:

1. Manter a estação a correr
2. Descomentar a linha 91 do ficheiro
`./ws-handlers/src/main/java/example/ws/handler/EveSimulatorHandler.java`
3. Voltar a instalar os ws-handlers
4. Voltar a correr o binas
5. Voltar a correr o binas-ws-cli
6. Verificar que foi emitida uma `SOAPFaultException` por invalidação no `MACHandler`

Proteção contra alterações de *headers* da mensagem SOAP:

1. Manter a estação a correr
2. Comentar a linha 91 do ficheiro
`./ws-handlers/src/main/java/example/ws/handler/EveSimulatorHandler.java`
3. Descomentar a linha 92
4. Voltar a instalar os ws-handlers
5. Voltar a correr o binas
6. Voltar a correr o binas-ws-cli
7. Verificar que foi emitida uma `SOAPFaultException` por invalidação no `MACHandler`

Proteção contra pedidos de um email diferente do discriminado no *Ticket*:

1. Manter a estação a correr
2. Descomentar a linha 93 do ficheiro

`./ binas-ws-cli/src/main/java/org/binas/ws/cli/BinasClientApp.java`

3. Voltar a correr o binas
4. Voltar a correr o binas-ws-cli (recompilando)
5. Verificar que foi emitida uma `SOAPFaultException` por invalidação no `BinasAuthorizationHandler`