** Task group: server security **

Context

A major attack vector on web application servers are injections, like SLQ injection, LDAP injections. To understand how these injections work, we use a deliberately insecure web application called WebGoat.

Deliverables

Show in screencast that you've executed all WebGoat injection attacks

Task

Perform all injection attacks on WebGoat

Subtask 1

1. Start the BBT VM
2. Log in as scrt/ict.se.scrt
3. run ./startWebGoat.sh and wait a minute
4. Check with "'netstat -ant" if the webGoat service are listening on port tcp/8080 and tcp/9090.

Alternative: DIY

1. Download and start the latest version of WebGoat. See https://github.com/WebGoat/WebGoat and the OWASP site for documentation. WebGoat runs in Tomcat, a java application server.
    - You need at least version 11 of java from http://openjdk.java.net
    - Check with `java -version`
2. Run in a command shell `java -jar webgoat-8.0.0.MXX.jar`
3. Watch for the line *Started Webgoat in ....*
4. WebGoat listens to localhost:8080 (check with `netstat -ant`)
5. WebGoat can be stopped via ˆC (CNTRL C)

Subtask 2

1. Start Firefox and browse to: http://localhost:8080/WebGoat and follow the instructions
    - Register as a new user

Subtask 3

1. Execute all the “General” and “Injection flaws” labs on WebGoat
    - Use ZAP (configure the right Local Proxy port in ZAP and proxy settings in Firefox) to intercept and manipulate the HTTP traffic

Done