** Task group: secure communication **

Context

To avoid Confidentiality and Integrity issues, all HTTP trafic between client (browser) en server must be encrypted via TLS (Transport Layer Security), so called HTTPS. The HTTPS communication is terminated on the Apache web server reverse proxy.

Task

Create a PKI certificate and configure the Apache web server reverse proxy

Deliverables

- Show a working HTTPS connection to your Debian host in the acceptance environment.
- Show a running smoketest application on your acceptance environment. (https://145.44.a.b)
- Show an updated ASVS excel sheet

Subtask 1: hostname at hbo-ict.org

1. Propose a sane hostname, .hbo-ict.org, for your acceptance environment. Visit http://hbo-ict.org/proposename It might take a while before we will activate your hostname in DNS. This is a manual procedure.

Subtask 2: self-signed certificate.

Create a self-signed certificate.

1. In a command shell use the `openssl` command to create a key and a self-signed certificate
   - See sheets of the lecture with this topic
2. Use for the Common Name, your given hostname.

Subtask 3: Configure Apache

Configure Apache to use the certificate and private key

1. Enable the ssl module

   ```
   $ sudo a2enmod ssl
   ```

2. Enable the default ssl host

   ```
   $ sudo a2ensite default-ssl
   ```

3. In a command shell edit the `default-ssl.conf` file

   - See sheets of the lecture with this topic
   - See https://httpd.apache.org/docs/2.4/mod/mod_ssl.html

4. Restart the apache server

   ```
   $ sudo systemctl restart apache2
   ```

5. Check for listening ports (should be tcp/80 and tcp/443)

   ```
   $ netstat -ant
   ```

6. Now you should be able to connect with a browser to your web application server

   - https://YOUR_HOSTNAME.hbo-ict.org/

Subtask 4: Configure Apache Continued

Configure the Apache reverse proxy, so that connection on port tcp/443 will be proxied to your dotnet application running on http://localhost:5001

1. Update the file '/etc/apache2/sites-available/default-ssl.conf'. follow the instructions on the security powerpoint slides of week 2.

   - Add the relevant proxy directives
     - SSLProxyEngine On
     - SSLProxyCheckPeerCN Off
     - ProxyPass / https://127.0.0.1:5001/
     - ProxyPassReverse / https://127.0.0.1:5001/
   - See https://httpd.apache.org/docs/2.4/mod/mod_proxy.html

Subtask 5: Verify

Verifying and debugging

1. Use openssl on Kali to debug the HTTPS connection

   ```
   openssl s_client -connect <IP of Debian server>:443
   ```

2. Use Wireshark on Kali to sniff the TLS handshake

Subtask 6: promote the Smoketest application to the Acceptance environment

See https://docs.microsoft.com/en-us/dotnet/core/deploying/ for using a Framework Dependent Executable (FDE).

1. On your Test- or Development server create a FDE
2. Secure copy the generated executable to the Acceptance environment via ssh (e.g. `scp` or WinSCP).
3. Deploy the executable on the Acceptance environment
    1. Run the executable on the acceptance environment

The smoketest application will validate your installation and configuration of the minimal viable set of software components (apache, mod-ssl, dotnet-runtime, docker, mssql) on your acceptance environment.

Done