

Hepia

Haute École du Paysage, d'Ingénierie et d'Architecture

Ingénierie et Systèmes de Communication

Année académique 2020/2021

Hacking et pentesting

Série 6 – Proxy

Genève, 16 Mai 2021

Étudiant :

Sergio Guarino

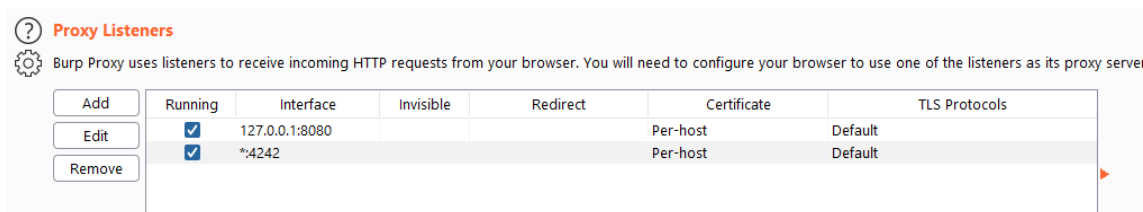
Professeur :

Stéphane Küng

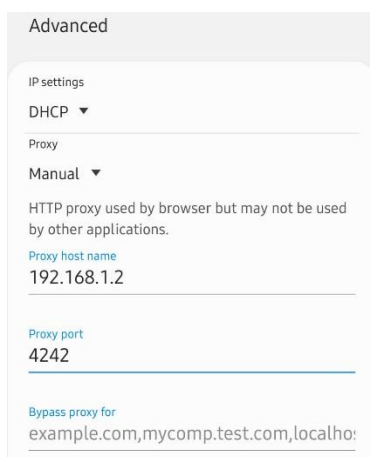
1. Proxy1

Pour cet exercice, une application météo préinstallée sur un smartphone Android a été choisie. Le proxy utilisé est Burp Suite.

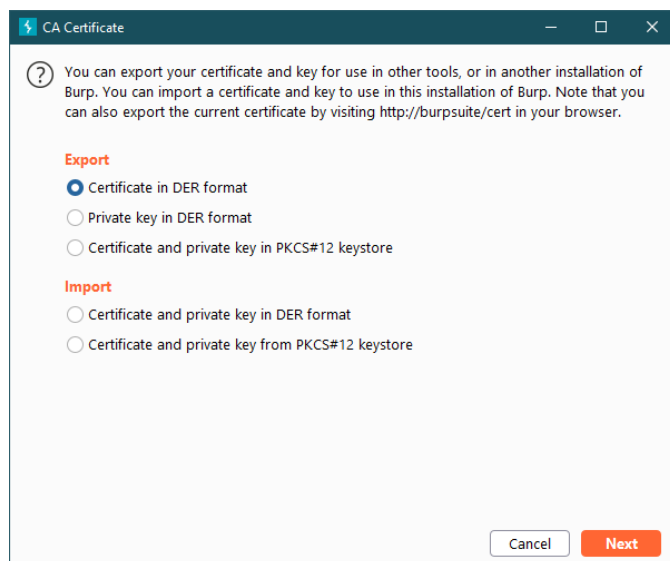
La première étape est d'indiquer au proxy sur quel port et adresse IP écouter. L'interface **127.0.0.1:8080** est celle par défaut, on peut la modifier ou en ajouter une nouvelle. Dans ce cas on a ajouté ***:4242**, qui écoute tout le trafic passant sur le port 4242.



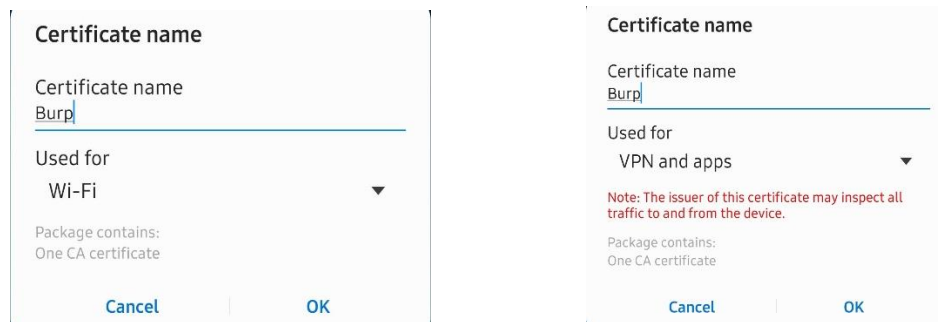
Ensuite il faut paramétrer le smartphone et indiquer l'adresse IP et le port du proxy :



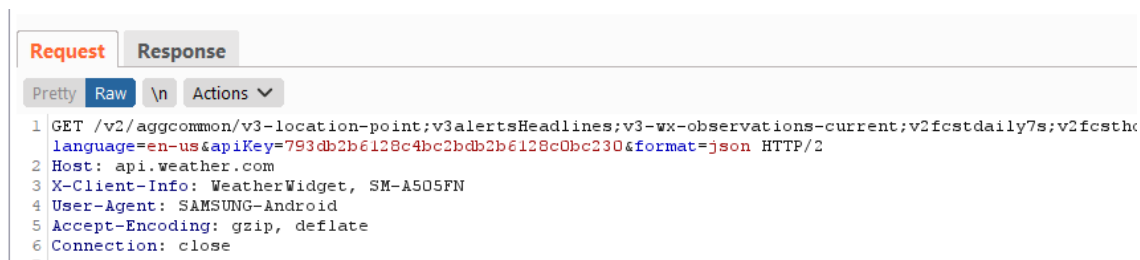
Pour pouvoir intercepter le trafic via le proxy, il faut également ajouter le certificat de sécurité de Burp Suite dans le smartphone. Il peut être généré depuis l'interface de Burp :



Il faut ensuite le transférer vers le smartphone et, à l'ouverture, il est installé automatiquement. Il faut l'installer deux fois pour qu'il soit accepté par le Wi-Fi et les applications :



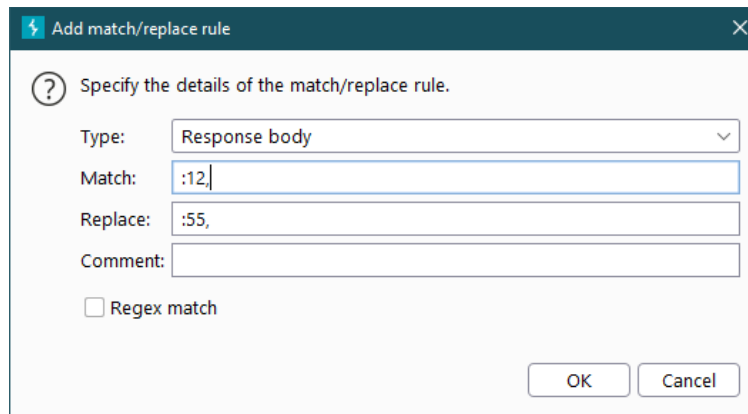
Le setup est terminé. Maintenant, si on ouvre une application ou un site web, le trafic est intercepté par Burp :



On a la possibilité d'intercepter la réponse à cette requête (option présente dans le menu Actions) :

```
{
  "obsQualifierCode": null,
  "obsQualifierSeverity": null,
  "precip1Hour": 0.76,
  "precip6Hour": 0.76,
  "precip24Hour": 1.02,
  "pressureAltimeter": 1011.18,
  "pressureChange": 0.68,
  "pressureMeanSeaLevel": 1011.2,
  "pressureTendencyCode": 1,
  "pressureTendencyTrend": "Rising",
  "relativeHumidity": 71,
  "snow1Hour": 0.0,
  "snow6Hour": 0.0,
  "snow24Hour": 0.0,
  "sunriseTimeLocal": "2021-05-14T06:03:44+0200",
  "sunriseTimeUtc": 1620965024,
  "sunsetTimeLocal": "2021-05-14T21:00:35+0200",
  "sunsetTimeUtc": 1621018835,
  "temperature": 12,
  "temperatureChange24Hour": 3,
  "temperatureDewPoint": 7,
  "temperatureFeelsLike": 11,
  "temperatureHeatIndex": 12,
  "temperatureMax24Hour": 15,
  "temperatureMaxSince7Am": 12,
  "temperatureMin24Hour": 5,
  "temperatureWindChill": 11,
  "uvDescription": "Low",
  "uvIndex": 2,
  "validTimeLocal": "2021-05-14T10:31:13+0200",
  "validTimeUtc": 1620981073,
  "visibility": 11.27,
  "windDirection": 230,
  "windDirectionCardinal": "SW",
  "windGust": null,
  "windSpeed": 8,
  "wxPhraseLong": "Rain",
  "wxPhraseMedium": "Rain",
  "wxPhraseShort": "Rain"
}
```

La réponse n'est rien d'autre que du json avec toutes les informations sur les données météorologiques. On peut alors utiliser la fonction de remplacement de Burp pour substituer les valeurs reçues avec d'autres valeurs de notre choix. Dans ce cas, on souhaite remplacer la valeur de la température quand ça vaut 12° et afficher 55 à la place.



Specify the details of the match/replace rule.

Type: Response body

Match: :12

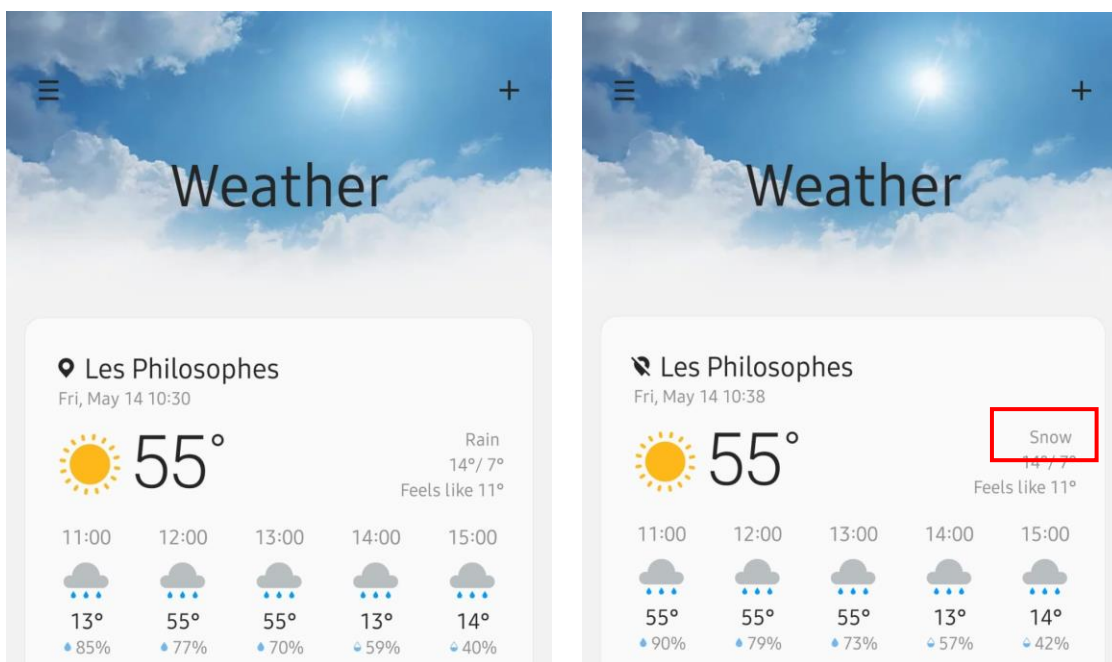
Replace: :55

Comment:

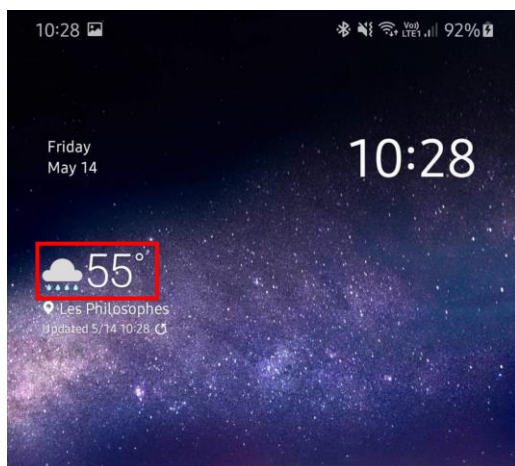
☐ Regex match

OK Cancel

Depuis l'application, on peut voir que la modification a fonctionné correctement (image de gauche). On peut aussi s'amuser à modifier d'autres valeurs, par exemple remplacer le mot « Rain » par « Snow » et faire croire qu'il va neiger (image de droite).



La chose intéressante est que ces fausses informations sont également transmises aux différents widgets ou applications qui se basent sur les données provenant de la même source, par exemple ci-dessous on a des captures d'écran de la page d'accueil du smartphone et de l'application sur un smartwatch :



2. Proxy2

Pour ce deuxième exercice, l'application choisie a été une app de news norvégiennes, appelé NRK. Comme pour l'exercice précédent, on a utilisé Burp Suite pour intercepter le trafic et récupérer l'API HTTPS utilisé dans l'application.

Ci-dessous la requête et la réponse :

```

Request
Pretty Raw \n Actions ▼
1 GET /vestland/skalabu-stengt-pa-ubestemt-tid-_-gjelek-ikkje-a-fryse-i-hel-veggedyra-1.15495965?header=off&viewContext=no.nrk.mobil.app&bridgeVersion=1.0.0 HTTP/2
2 Host: www.nrk.no
3 Cookie: nrkbucket1=3; i00=0000609f77e7c6e70000; _gid=GA1.2.1453823254.1621063657; _ga=GA1.2.b4b0356d-2c20-4579-ae41-9dfcdc144a75; _MBL=47Bt22ut22t3A422AhiLRBZxQG%22%2C%22t%22%3A1621063663%7D; _lp4_u=92fkrN3Kgg; _gat_ca_kurator=1; data-core-analytics-sc-referrer={%22id%22:%22pp:1.15495965%22%2C%22name%22:%22DNT-hytte420blei420ikkje%20kvitt%20vegge%22%2C%22addyra%20%22%80%93%20stengt%20p%2C%22A5%20ubestemt%20tid%22%2C%22list%22:%22nrkno_forside_floor_4%22%2C%22position%22:3%2C%22variant%22:%22N/A%22%2C%22brand%22:%22kur:12%22%2C%22category%22:%22kur-room--size-33%22}
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (Linux; Android 10; SM-A505FN Build/QP1A.190711.020; wv) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/90.0.4430.210 Mobile Safari/537.36
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
7 X-Requested-With: no.nrk.mobil.app
8 Sec-Fetch-Site: none
9 Sec-Fetch-Mode: navigate
10 Sec-Fetch-User: ?1
11 Sec-Fetch-Dest: document
12 Accept-Encoding: gzip, deflate
13 Accept-Language: en-US,en;q=0.9,fr-CH;q=0.8,fr;q=0.7,it-CH;q=0.6,it;q=0.5
14 Connection: close

```

Response

Pretty Raw Render \n Actions

```
1 HTTP/2 200 OK
2 Cache-Control: no-cache, max-age=0, must-revalidate, no-store
3 Content-Type: text/html; charset=UTF-8
4 X-UA-Compatible: IE=Edge
5 Vary: Accept-Encoding
6 Accept-Ranges: bytes
7 Date: Sat, 15 May 2021 07:29:26 GMT
8 Set-Cookie: _nrkbucket1=3; Path=/; max-age=15552000;
9 Strict-Transport-Security: max-age=15768000
10
11 <!doctype html>
12 <!--[if IEMobile 7 ]><html class="no-js iem7 " lang="nn-NO"><![endif]-->
13 <!--[if lt IE 7 ]><html class="no-js ie6 " lang="nn-NO"><![endif]-->
14 <!--[if IE 7 ]><html class="no-js ie7 " lang="nn-NO"><![endif]-->
15 <!--[if IE 8 ]><html class="no-js ie8 " lang="nn-NO"><![endif]-->
16 <!--[if IE 9 ]><html class="no-js ie9 " lang="nn-NO"><![endif]-->
17 <!--[if (gt IE 9)|!(gt IEMobile 7)|!(IEMobile)|!(IE)]><!--><html class="no-js " lang="nn-NO">
18 <!--<![endif]-->
19 <head>
20 <meta charset="UTF-8" />
21 <title>
22 Skålabu stengt på ubestemt tid - gjekk ikkje å fryse i hel veggedyra - NRK Vestland
23 </title>
24 <meta name="viewport" content="width=device-width,initial-scale=1.0" />
25 <meta name="msapplication-tap-highlight" content="no" />
26 <meta http-equiv="X-UA-Compatible" content="IE=edge" />
27 <meta name="description" content="Håpet var at vinteren skulle ta knekken på veggedyra på Skålabu i Nordfjord. -
28 <meta name="author" content="Ole Kristian Svalheim"/>
29 <meta name="dc.date.issued" content="2021-05-14"/>
30 <link rel="apple-touch-icon-precomposed" sizes="114x114" href="https://static.nrk.no/nrkno/serum/2.0.448/type/page/
31 <link rel="apple-touch-icon-precomposed" sizes="72x72" href="https://static.nrk.no/nrkno/serum/2.0.448/type/page/
32 <link rel="apple-touch-icon-precomposed" href="https://static.nrk.no/nrkno/serum/2.0.448/type/page/img/apple-tou
33 <link rel="shortcut icon" href="https://static.nrk.no/nrkno/serum/2.0.448/type/page/img/nrk.ico" />
34 <!-- Custom address bar color for Chrome, Firefox OS and Opera -->
35 <meta name="theme-color" content="#2e2e2e">
36 <!-- Custom address bar color for Windows Phone -->
37 <meta name="msapplication-navbutton-color" content="#2e2e2e">
38 <!-- Custom address bar color for iOS Safari -->
39 <meta name="apple-mobile-web-app-status-bar-style" content="#2e2e2e">
40 <meta property="fb:app_id" content="726018114135848" />
41 <meta property="fb:pages" content="172533639487543,165167636714,389088897795,11222655995,1474569816109652,4477861
42 8811,258697507625,194077312500,596161207086401,68480421097,163083014696,7186751742,352840908242419,20142313670928
43 <meta name="p:domain_verify" content="ec4318a079405ca7cd9055634d626a64"/>
```

Un programme en Golang (proxy2.go en annexe) a été écrit pour effectuer cette requête GET et afficher la réponse sur une page web. Le programme est basé sur les exemples du site officiel de Golang : <https://golang.org/pkg/net/http/#pkg-examples>

Pour vérifier le bon fonctionnement du programme, on a testé également l'API de l'application météo du premier exercice et aussi celui d'une autre application pour afficher les vols au départ de Bruxelles (détails en commentaire dans le programme).

Pour exécuter le programme, lancer la commande **go run proxy2.go**. Ensuite, aller à la page <http://localhost:4200/> pour afficher le résultat.

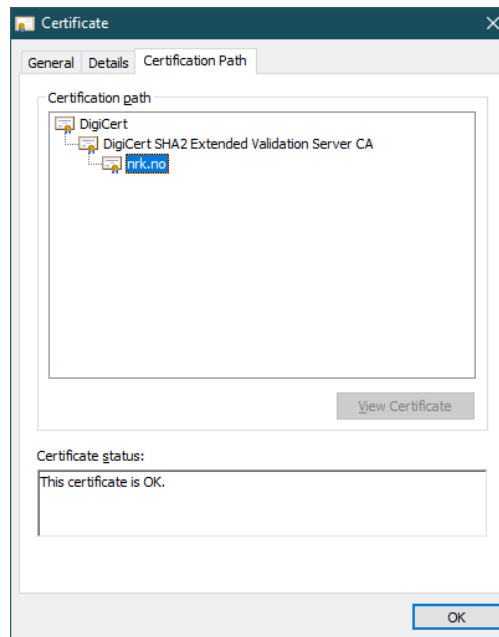
3. Proxy3

Pour ce troisième exercice il a fallu modifier le code pour utiliser une connexion TLS avec l'API de l'exercice précédent.

Tout d'abord, il faut télécharger le certificat du serveur qu'on souhaite atteindre. Il peut être exporté avec n'importe quel navigateur. Dans le cas où il ne serait pas en format **.pem** il faut le convertir. Pour cela, on peut utiliser une commande **openssl** :

```
openssl x509 -inform der -in cert.cer -outform pem -out cert.pem
```

Quand on est dans le navigateur, on retrouve plusieurs certificats :



Pour cet exercice on a testé avec le certificat root et celui SSL, mais aussi celui intermédiaire irait bien.

Après avoir exporté et converti les certificats, on peut exécuter notre programme et, si on accède à la page <http://localhost:42443>, on peut voir qu'on reçoit bien les données depuis le site web.

En revanche, si on configure un proxy (toujours Burp Suite dans ce cas), le programme ne valide pas le certificat issu par Burp :

```
Terminal - sergi@ubuntu:~/Documents/Hacking/6.proxy
File Edit View Terminal Tabs Help
6.proxy [master] go run proxy3.go
Getting data from website...
2021/05/16 09:11:50 Get "https://www.nrk.no/vestland/skalabu-stengt-pa-ubestemt-
tid-_gjeikk-ikkje-a-fryse-i-hel-veggedyra-1.15495965?header=off&viewContext=no.n
rk.mobil.app&bridgeVersion=1.0.0": x509: certificate signed by unknown authority
exit status 1
6.proxy [master]
```

Le programme est **proxy3.go** en annexe et il peut être exécuté avec la commande **go run proxy3.go**. Avant exécution, il faut contrôler le parcours des certificats (en annexe) et l'adresse du proxy.