

רשתות מחשבים מתקדם – תרגיל כיתה

הקמת שרת HTTPS עם סרטיפיקט חתום

ברק גונן

בתרגיל מודרך זה ניצור שרת HTTPS. השרת כולל סוקט מוצפן. לטובת ההצפנה נצטרך מפתח וסרטיפיקט חתום, אותם כמובן ניצור בעצמנו ואף נחתום על הסרטיפיקט באמצעות CA. העבודה מצריכה WSL.

התוצאה תהיה אתר, שהדפדפן שלנו יציג בתור אתר "כשר" מבחינת אימות. כך עובדים סרטיפיקטים שניתנים לדוגמה במקומות עבודה, כדי לבצע Man In The Middle לתעבורת האינטרנט של העובדים. באופן זה ניתן לבצע פיקוח לאן העובדים גולשים.

שלבי העבודה:

1. ניצור Certificate Authority ונכניס את הסרטיפיקט שלו ל-`certmgr`, כך שהמחשב יכיר בו כבסיס לחתימות. שלבי העבודה:
 - a. יצירת זוג מפתחות ל-CA שלנו
 - b. ה-CA יחתום לעצמו על סרטיפיקט
 - c. נייבא את הסרטיפיקט לתוך `certmgr`
2. ניצור לדומיין שלנו סרטיפיקט חתום על ידי ה-CA. שלבי העבודה:
 - a. יצירת זוג מפתחות לדומיין שלנו
 - b. יצירת CSR
 - c. ה-CA יענה ל-CSR של הדומיין ויעניק לו סרטיפיקט חתום
3. ניצור שרת HTTPS שמוסר את הסרטיפיקט החתום ללקוח (הדפדפן). הדפדפן יוודא מול ה-`certmgr` שאכן הסרטיפיקט כשר.

יצירת Certificate Authority

1. צרו מפתח ל-CA:

```
openssl genrsa -out ca.key 4096
```

2. כדי להתבונן במפתח:

```
cat ca.key
```

מה שנקבל מטעה מעט, נראה שיש לנו רק מפתח פרטי. למעשה המידע במפתח הפרטי כולל גם את המודולוס ($P \times Q$) ואת $Q \times P$ עצמם. כזכור אם יש לנו אותם, אפשר לחשב את ה- $public\ key$. הפקודה הבאה תציג את כל המידע כולל חישוב המפתח הציבורי:

```
openssl rsa -in ca.key -text -noout
```

3. צרו סרטיפיקט חתום על ידי ה-CA לעצמו:

```
openssl req -new -x509 -days 3650 -sha256 -key ca.key -out ca.crt
```

בתהליך היצירה של הסרטיפיקט תתבקשו להזין נתונים שונים. למעט ה- $Common\ Name$, ניתן לדלג על כולם, באמצעות לחיצה על $enter$. באשר ל- $Common\ Name$. קיבעו אותו להיות $Root\ CA$, אפשר להוסיף לו את השם שלכם. אם תתבקשו להזין $Challenge\ Password$ הזינו סיסמה כלשהי.

4. הקליקו על קובץ ה- crt ועיקבו אחרי ההוראות עד שהוא ייובא לתוך $certmgr$.

יצירת סרטיפיקט חתום על ידי ה-CA

1. לפני הכל יש צורך להכין מספר קבצים בתיקה שבה נוצרים הסרטיפיקטים.

a. צרו קובץ בשם serial (ללא סיומת). באמצעות עורך טקסט, כיתבו ערך בעל 10 ספרות

הקדצימליות. לדוגמה ABCDEF0000.

b. צרו קובץ ריק בשם index.txt

c. העתיקו לתיקה את הקובץ CA-config.conf

d. העתיקו לתיקה את הקובץ CSR-config.conf

כעת אפשר להתחיל.

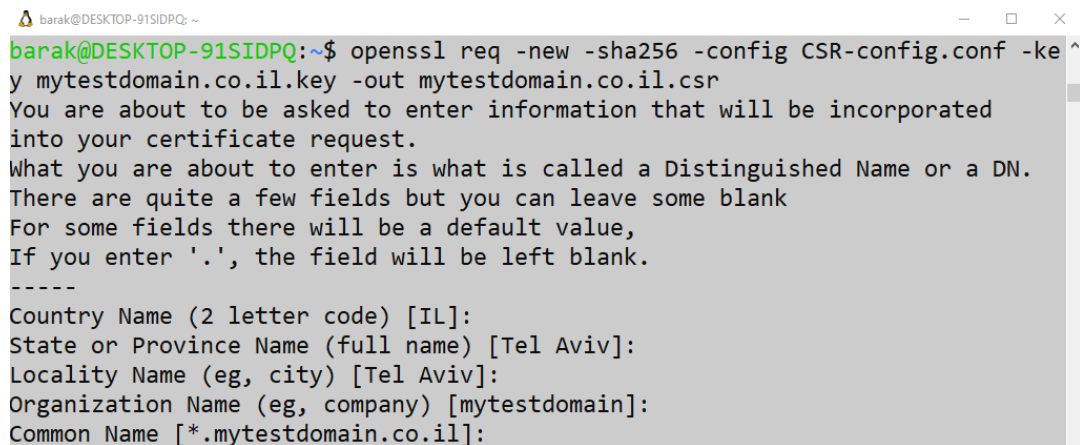
2. יצירת מפתח:

```
openssl genrsa -out mytestdomain.co.il.key 2048
```

3. יצירת Certificate Signing Request:

```
openssl req -new -sha256 -config CSR-config.conf -key mytestdomain.co.il.key -out
```

```
mytestdomain.co.il.csr
```



```
barak@DESKTOP-91SIDPQ: ~  
barak@DESKTOP-91SIDPQ:~$ openssl req -new -sha256 -config CSR-config.conf -key mytestdomain.co.il.key -out mytestdomain.co.il.csr  
You are about to be asked to enter information that will be incorporated into your certificate request.  
What you are about to enter is what is called a Distinguished Name or a DN.  
There are quite a few fields but you can leave some blank  
For some fields there will be a default value,  
If you enter '.', the field will be left blank.  
-----  
Country Name (2 letter code) [IL]:  
State or Province Name (full name) [Tel Aviv]:  
Locality Name (eg, city) [Tel Aviv]:  
Organization Name (eg, company) [mytestdomain]:  
Common Name [*mytestdomain.co.il]:
```

דלגו על כל הפרטים באמצעות enter. אם ברצונכם לשנות את שם הדומיין, פשוט עירכו את הקובץ CSR-

config.conf

צפו ב-CSR בעזרת הפקודה:

```
openssl req -in mytestdomain.co.il.csr -noout -text
```

במידה ולא ניתן לצפות עקב תקלה של 'unable to load x509 request', כיתבו:

```
openssl x509 -in mytestdomain.co.il.csr -noout -text
```

```
barak@DESKTOP-91SIDPQ:~$ openssl req -in mytestdomain.co.il.csr -noout -text
Certificate Request:
Data:
  Version: 1 (0x0)
  Subject: C = IL, ST = Tel Aviv, L = Tel Aviv, O = mytestdomain, CN = *.mytestdomain.co.il
```

4. השתמשו ב- RootCA כדי ליצור סרטיפיקט חתום ל-mytestdomain.co.il:

```
openssl ca -config CA-config.conf -cert ca.crt -keyfile ca.key
-in mytestdomain.co.il.csr -out mytestdomain.co.il.crt
```

```
barak@DESKTOP-91SIDPQ: ~
Using configuration from CA-config.conf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName             :PRINTABLE:'IL'
stateOrProvinceName     :ASN.1 12:'Tel Aviv'
localityName            :ASN.1 12:'Tel Aviv'
organizationName        :ASN.1 12:'mytestdomain'
commonName              :ASN.1 12:'*.mytestdomain.co.il'
Certificate is to be certified until Dec  8 21:04:28 2025 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

5. צפו בסרטיפיקט בעזרת הפקודה:

```
openssl x509 -in mytestdomain.co.il.crt -noout -text
```

6. את הסעיף להלן יש צורך לבצע רק אם בניתם שרשרת של יותר מסרטיפיקט אחד (בתרגיל זה לא עשינו כך). הסרטיפיקט של ה-Root CA נמצא ממילא במחשב. אם השתמשם ב-ICA, Intermediate CA, פיתחו את הסרטיפיקט באמצעות notepad. העתיקו לסופו את הסרטיפיקט של ה-ICA. התוצאה תכלול כיתוב של סיום סרטיפיקט אחד והתחלה של השני.

```
914=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIFSzCCAzOqAwIBAqIUd+20dvCY11x
```

שרת HTTPS

הקוד הבא יוצר שרת HTTPS:

```
import socketserver
import http.server
import ssl

httpd = socketserver.TCPServer(('0.0.0.0', 443), http.server.SimpleHTTPRequestHandler)
httpd.socket = ssl.wrap_socket(httpd.socket, certfile="mytestdomain.co.il.crt",
keyfile="mytestdomain.co.il.key", server_side=True)

httpd.serve_forever()
```

העתיקו את קובץ ה-key וקובץ ה-crt לתוך התיקיה שבה נמצא הסקריפט, לטובת פשטות (או שתשנו את הקוד כך שיכלול את הנתיב המלא לקבצים).

פיתחו את הקובץ hosts, שנמצא בנתיב c:\windows\system32\drivers\etc, במצב admin. הוסיפו לו את השורה:

127.0.0.1 www.mytestdomain.co.il

כעת כאשר תגלוש בדפדפן ל- <https://www.mytestdomain.co.il> תקבלו את תשובת השרת:

