

## פרויקט סיכום משותף עם חברת אוטוריו

במשימה זו אתם תתמקדו בלהזליג מידע רגיש ממפעל קריטי בלי שישימו לב (בלי התראות ב HMI).

0. תיצרו pcap ותחקרו את הפרוטוקול -תבינו איך המידע עובר ומה עובר. (10 נקודות

1. משימה ראשונה [50 נקודות] להזליג : 50 \* "otorio Rocks" (50 פעם)

מדדי ההצלחה לכל המשימות:

א. אין התרעות ב HMI

ב. הסטטוס של המפעל הוא תקין והמנורות עובדות כמצופה.

2. תבחרו תמונה ותזליגו אותה [20 נקודות]

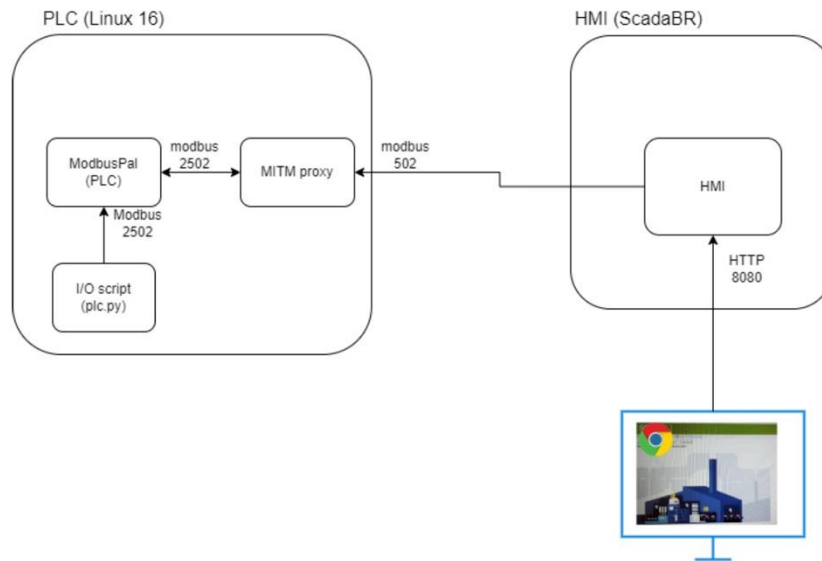
3. אנא תוסיפו coil ב offset 900 ב ModbusPal interface שאתם לא יכולים למחוק ותגבילו את גודל החבילה המקסימלית ל 1000. כלומר אתם מזליגים עכשיו את המידע של התמונה מ offset 900 ומוגבלים ל mtu של 1000.

הגשה:

א. קוד רק במערכת Moodle.

ב. בחינה פרונטלית.

ארכיטקטורה של המערכות:



Credentials

What	creds
VM - Linux16	matand:1
VM - ScadaBR	scadabr:scadabr
HMI web interface (8080)	admin:admin

לינקים ל VM:לפתוח עם virtualbox

[scada\\_br ovf.zip](#)

[Linux ovh.zip](#)

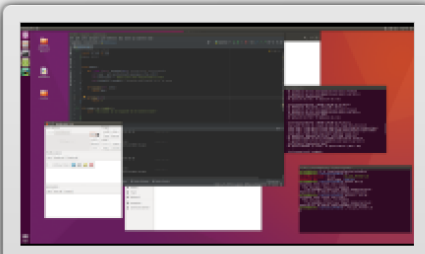
**מה מקבלים:**

לתרגיל זה מצורפים 2 מערכות vm:

1. הינה מערכת HMI אשר מצורפת לה ממשק גרפי של מפעל.
2. VM תכנות שבו יהיה לכם מערכת plc שרצה, ממשק modbus server וממשק זדוני של mitm שאתם תכנסו ותשפרו על מנת להזליג את המידע.

כל העבודה תהיה מעל virtualbox.

שימו לב להגדיר את ה VM בזמן ה import בצורה הבאה: linux 2.6 64 bit (זה חשוב)

<b>General</b> Name: vm 1 Operating System: Linux 2.6 / 3.x / 4.x / 5.x (64-bit)	<b>Preview</b> 
<b>System</b> Base Memory: 8192 MB Processors: 4 Boot Order: Floppy, Optical, Hard Disk Acceleration: Nested Paging, PAE/NX, KVM Paravirtualization	
<b>Display</b> Video Memory: 16 MB Graphics Controller: VBoxVGA Remote Desktop Server: Disabled Recording: Disabled	
<b>Storage</b>	

שימו לב הגדרה זו נעשת לאחר ה import ולאחר שבחרת את ה vm הרצוי.

Import Virtual Appliance

### Appliance to import

Please choose the source to import appliance from. This can be a local file system to import OVF archive or one of known cloud service providers to import cloud VM from.

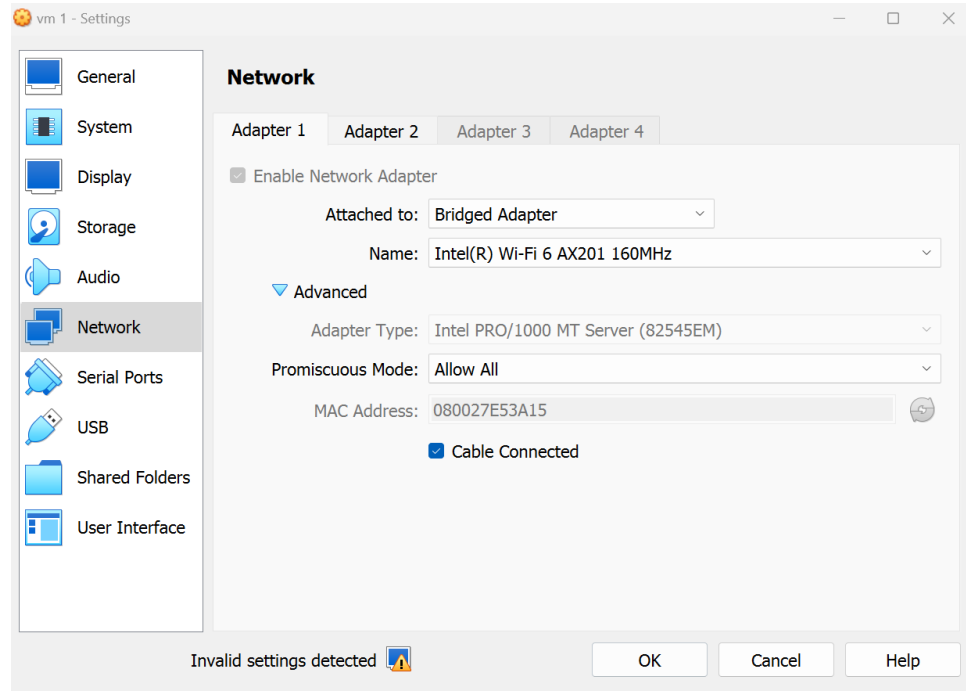
Source: Local File System

Please choose a file to import the virtual appliance from. VirtualBox currently supports importing appliances saved in the Open Virtualization Format (OVF). To continue, select the file to import below.

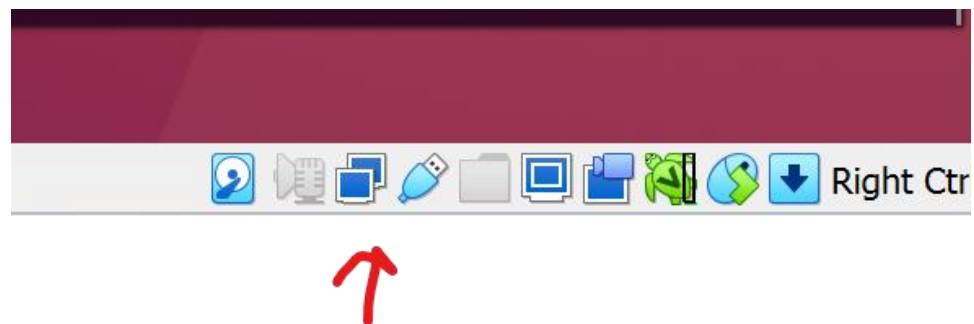
File:

Help
Expert Mode
Back
Next
Cancel

שימו לה להגדרות הרשת:



כאשר אתם טוענים את ה vm של HMI אם הכל מוגדר כהלכה ויש לכם רשת:



שימו לב שיש לכם 8.8.8.8 ping אם הגדרתם את הרשת ל bridge לפני ההתחלה אז צריך להיות לכן.

HMI:

פעילות תקינה שלו הינה כאשר הוא עולה אתם רואים שיש לו כתובת IP:

```
Welcome to ScadaBR! To use ScadaBR, open your browser and navigate to:
http://10.0.0.16:8080/ScadaBR
Have Fun!
scadabr login:
Welcome to ScadaBR! To use ScadaBR, open your browser and navigate to:
http://10.0.0.16:8080/ScadaBR
Have Fun!
scadabr login: _
```



אם זה לא קרה אז אנא תעשו את השלבים הבאים:

/sbin/ifconfig

Take the interface name: in my case enp0s3

```
scadabr@scadabr:~$ /sbin/ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 10.0.0.16  netmask 255.255.255.0  broadcast 10.0.0.255
    inet6 fe80::a00:27ff:fe9a:b881  prefixlen 64  scopeid 0x20<link>
    inet6 2a10:8012:15:2da2:a00:27ff:fe9a:b881  prefixlen 64  scopeid 0x0<global>
obal>
```

Edit:

Sudo nano /etc/network/interfaces (cotrl+O and later enter) will save it. Then you can restart and check that everything is working and you have an IP as in my example [HTTP://10.0.0.16:8080/ScadaBR](http://10.0.0.16:8080/ScadaBR)

```
# and how to activate them. For more information, see interfaces(5).
source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s3
iface enp0s3 inet dhcp
```

התוצר של הפעולה הינה מערכת ה HMI שאליה נדבר בהמשך.

MITM:

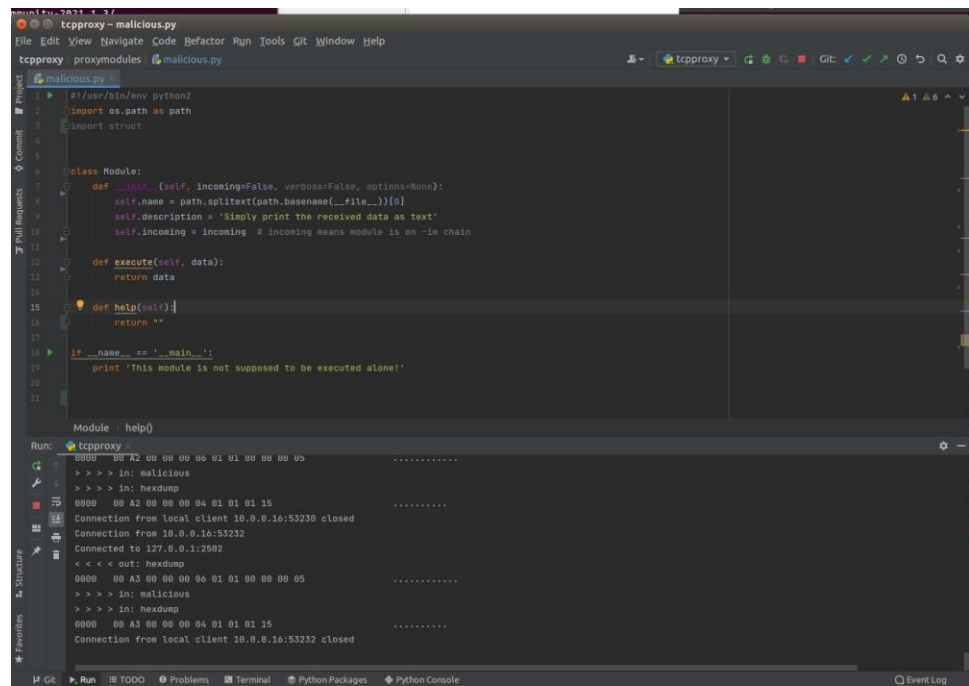
ל VM זה יש ממשק גרפי (UBUNTU) ואליו אתם מפתחים:

א. תבדקו שיש לכם רשת .

ב. תעלו את pycharm (לא לשכוח: sudo)

תפתחו טרמינל תגשו ל desktop ושם בתוך pycharm/bin תריצו ./pycharm.sh sudo

התוצאה תהיה pycharm וקטע הקוד הבא:



```

1 #!/usr/bin/env python2
2 import os.path as path
3 import struct
4
5 class Module:
6     def __init__(self, incoming=False, verbose=False, options=None):
7         self.name = path.splitext(path.basename(__file__))[0]
8         self.description = 'Simply print the received data as text'
9         self.incoming = incoming # incoming means module is on -in chain
10
11     def execute(self, data):
12         return data
13
14     def help(self):
15         return ""
16
17 if __name__ == '__main__':
18     print 'This module is not supposed to be executed alone!'
19
20
21
Module : help()

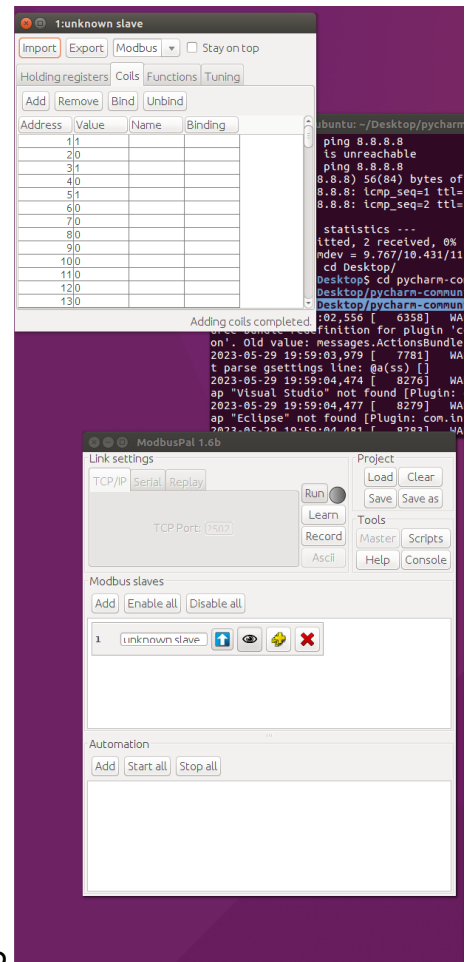
```

```

Run: tcp-proxy
0000 00 A2 00 00 00 00 01 01 00 00 00 00 00 00 00 00 .....
> > > in: malicious
> > > in: hexdump
0000 00 A2 00 00 00 00 04 01 01 01 15 .....
Connection from local client 10.0.0.16:53230 closed
Connection from 10.0.0.16:53232
Connected to 127.0.0.1:2502
< < < out: hexdump
0000 00 A3 00 00 00 00 06 01 01 00 00 00 05 .....
> > > in: malicious
> > > in: hexdump
0000 00 A3 00 00 00 00 04 01 01 01 15 .....
Connection from local client 10.0.0.16:53232 closed

```

ג. תפתחו את תקיית ה modbuspal תאזינו לפורט 2502 (בהתאם לצירור) ואם אין לכם slave אז תיצרו לכם:



כאשר ה coils זה מ 1 עד כמה שאפשר.

ותלחצו על run אתם אמורים לראות את run מהבהב בירוק

ד. נפעיל את המפעל כל הזמן:

לכו ל `home/matand/projects/modbus/` ותריצו

`./run_plc_forever.sh`

בטרמינל

## בחזרה ל: HMI

א. תעדכנו את כתובת ה IP של ה HMI ב datasource ב UI: (הכתובת הינה הכתובת של vmx אשר מחזיק את ה.mitm proxy).

**Current alarms**  
No active alarms for this data source

**Modbus IP properties**

Name: Copy of LinuxVModbus  
Export ID (XID): DS\_572605  
Update period: 1 second(s)  
Quantize: ☐  
Timeout (ms): 500  
Retries: 2  
Contiguous batches only: ☐  
Create slave monitor points: ☒  
Max read bit count: 2000  
Max read register count: 125  
Max write register count: 120  
Transport type: TCP  
Host: 10.0.0.16  
Port: 502  
Encapsulated: ☐

**Event alarm levels**

Data source exception: Urgent  
Point read exception: Urgent  
Point write exception: Urgent

**Modbus node scan**  
Scan for nodes Cancel  
Nodes found:

**Modbus read data**  
Slave id: 1  
Register range: Coil status  
Offset (0-based): 0  
Number of registers: 100  
Read data

**Point locator test**  
Slave id: 1  
Register range: Coil status  
Modbus data type: Binary  
Offset (0-based): 0  
Bit: 0  
Number of registers: 0  
Character encoding: ASCII  
Read Add point

**Points**

Name	Data type	Status	Slave	Range	Offset (0-based)
light1	Binary		1	Coil status	0
light2	Binary		1	Coil status	1
light3	Binary		1	Coil status	2
light4	Binary		1	Coil status	3
light5	Binary		1	Coil status	4

**Data sources** BACnet I/P

Name	Type	Connection	Status
LinuxVModbus	Modbus IP	10.0.0.15:502	

Page 1 of 1 (1 - 1 of 1 rows) **1**

ב. ננקה את כל ההתראות: על ידי לחיצה על. acknowledge all.

**SCADA BR**  
1.8.02 - Community Edition

**Pending alarms**

Id	Alarm level	Time	Message	Inactive time
2839		12:23:57	"LinuxVModbus": com.serotonin.modbus4j.exception.ModbusTransportException: java.net.SocketTimeoutException: connect timed out	Active

**Event search**

Id:   
Event source type: All  
Status: All  
Alarm level: All  
Keywords:   
Max results:   
Search

©2009-2011 Fundação Certi, MCA Sistemas, Unis Sistemas, Conetec. All rights reserved.

הקוד ב MITM:

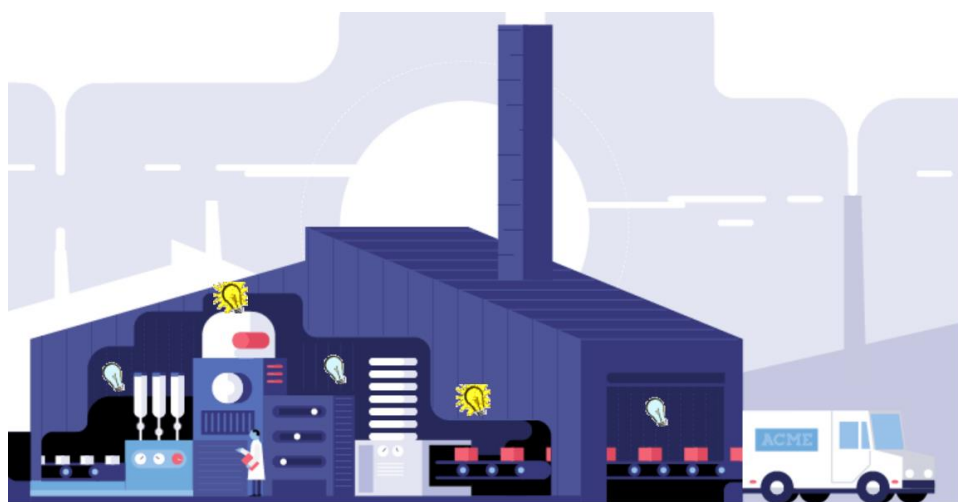
פונקציה execute הינה הפונקציה שבה אתם משחקים.



מערכת עובדת נראת ככה:



אורות מתחלפים כל 3 שניות:



ללא סימני אזהרה ליד הנורות  
ה:pycharm



```
tcpdump
0000  00 EA 00 00 00 00 01 01 01 01 01 01 01 01 01 01  .....
> > > in: malicious
> > > in: hexdump
0000  00 EA 00 00 00 00 04 01 01 01 01 15  .....
Connection from local client 10.0.0.16:54478 closed
Connection from 10.0.0.16:54480
Connected to 127.0.0.1:2582
< < < out: hexdump
0000  00 EB 00 00 00 00 06 01 01 00 00 00 05  .....
> > > in: malicious
> > > in: hexdump
0000  00 EB 00 00 00 00 04 01 01 01 15  .....
Connection from local client 10.0.0.16:54480 closed
```

בהצלחה 😊

אוטוריו ורן