工控系统蜜罐建设与协议仿真技术分享

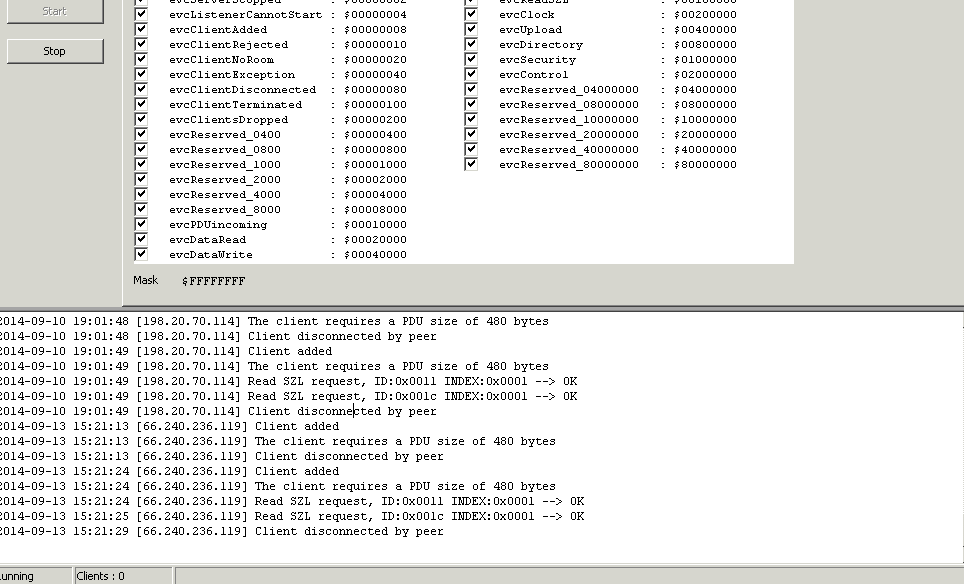
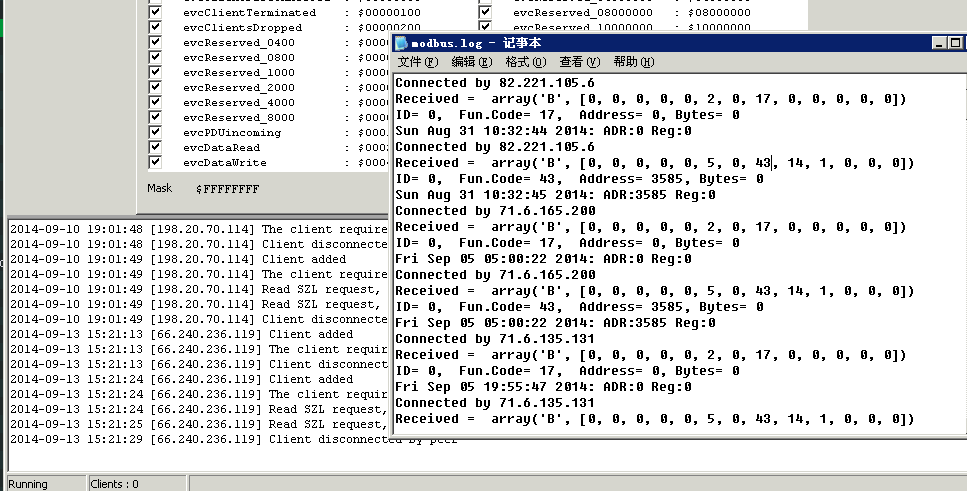
[2014/09/14](https://plcscan.org/blog/2014/09/icsscada-honeypot-and-ics-protocol-simulation-technology/)[Z-0ne](https://plcscan.org/blog/author/admin/)[Leave a comment](https://plcscan.org/blog/2014/09/icsscada-honeypot-and-ics-protocol-simulation-technology/#respond) 20576 views

**0x1、简介**

随着Eripp、Shodan、Zoomeye类似的网络空间搜索引擎先后的出现，网络扫描技术的发展和探测的增多，工控设备、物联网设备、基础设施等作为互联网的一部分，已逐渐被攻击者所重视。从2010年的震网到如今的Havex，工控网络作为一个相对封闭的网络则出现了越来越多更具有针对性的攻击事件和恶意程序。有针对的模拟工控系统某些特征的蜜罐出现也变得必要，蜜罐作为一种相对主动的安全检测手段，相信在未来使用的会越来越广泛。

**0x2、工控蜜罐日志分析**

**Link:**[**工控蜜罐日志收集项目链接地址**](https://plcscan.org/blog/dataanalysis/icsscada-honeypot-log/)

博主于6月低在香港和大陆外网节点搭建了设备协议仿真程序，用于发现主动扫描和来源IP的操作行为，根据积累的日志得出了如下结论：  
A、目前针对特定端口（tcp/102、tcp/502）的扫描和识别的源IP均来自于国外。  
  
B、对协议的识别使用的是已知公开的技术。（由ScadaStrangeLove发布的plcscan和Digitalbond移植的基于nmap的nse识别枚举脚本，这一点可以根据目前接收到的交互报文断定，如读取西门子PLC设备时使用的是SZL请求，识别Modbus设备型号信息使用了43号功能码）。  
  
C、协议仿真程序并未收到恶意攻击和写入指令（例如西门子PLC CPU的stop指令和数据写入指令等，modbus的05、06等具有写入功能的功能码，详请参考日志）。

**0x3、快速实现协议仿真**

**案例一**

**Modbus协议介绍**

Modbus协议是工控网络中最常见的一种协议之一，该协议由莫迪康于1979年，为使用可编程逻辑控制器（PLC）而发表，因为简单易用至今使用已经相当广泛，如RTU、PLC、传感器等，协议可以适用于以太网、串口。同样因为使用广泛，公网暴露数量较多（[Shodan查询连接](http://www.shodanhq.com/search?q=port%3A502)）。

**Modbus协议服务端实例脚本**

**Link:**[**Modbus协议资料**](http://wenku.baidu.com/link?url=OGUWIyVZEQ6RXppHvbSwQsg91_44q-fAvGiLIe8ONTH4v_ZuycYZ7dcNSjXoD-ROjcz08KSzjFNgPS4wnjrDFBdzIRBpyUwVaJSB2kfIUzK)

#!/usr/bin/python

# version 2.7

# Source Code form https://github.com/tecpal/PyModbus

# change Z-0ne

import os

import sys

import socket,thread

from array import array

from time import sleep, ctime

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

s.bind(('',502))

s.listen(10)

F = open('c:\modbus.log','a',0)

sys.stdout = F

def TCP(conn,addr,F):

buffer = array('B',[0]\*300)

while 1:

try:

conn.recv\_into(buffer)

ID = buffer[6]

FC = buffer[7]

mADR = buffer[8]

lADR = buffer[9]

ADR = mADR\*256+lADR

LEN = buffer[10]\*256+buffer[11]

BYT = LEN\*2

print "Received = ",buffer[0:13+buffer[12]]

if (FC < 5 and FC > 0): #Read Inputs or Registers

DAT = array('B')

if FC < 3:

BYT = (lambda x: x/8 if (x%8==0) else x/8+1)(LEN) #Round off the no. of bytes

v = 85 #send 85,86.. for bytes.

for i in range(BYT):

DAT.append(v)

v = (lambda x: x+1 if (x<255) else 85)(v)

else:

for i in range(LEN): #Sends back the address as data

DAT.append(mADR)

DAT.append(lADR)

if (lADR == 255):

lADR = 0

mADR = mADR + 1

else: lADR = lADR + 1

print "ID= %d, Fun.Code= %d, Address= %d, Length= %d" %(ID, FC, ADR, LEN)

conn.send(array('B', [0,0,0,0,0, BYT+3, ID, FC, BYT]) + DAT )

elif (FC == 15 or FC == 16 or FC == 6 or FC == 43 or FC == 17): #Write Registers

BYT = buffer[12]

conn.send(array('B', [0,0,0,0,0, 6, ID, FC, mADR, lADR, buffer[10], buffer[11] ] ) )

buf = buffer[13:(13+BYT)]

message = ': ADR:'+str(ADR)+' '

if FC == 15:

print "ID= %d, Fun.Code= %d, Address= %d, Length= %d, Bytes= %d" %(ID, FC, ADR, LEN, BYT)

for j in range(BYT): message = message+('Byte:'+str(j)+'='+str(buf[j])+', ')

elif FC == 16:

print "ID= %d, Fun.Code= %d, Address= %d, Length= %d, Bytes= %d" %(ID, FC, ADR, LEN, BYT)

for j in range(BYT/2): message = message+('Reg:'+str(j)+'='+str((buf[j\*2]<<8)+(buf[j\*2+1]))+', ')

elif FC == 6:

print "ID= %d, Fun.Code= %d, Address= %d, Bytes= %d" %(ID, FC, ADR, LEN)

message = message+('Reg:'+str(LEN))

elif FC == 43:

print "ID= %d, Fun.Code= %d, Address= %d, Bytes= %d" %(ID, FC, ADR, LEN)

message = message+('Reg:'+str(LEN))

conn.send(bytes(bytearray([0x00, 0x00,

0x00, 0x00,

0x00, 0x32,

0x00,

0x2b, #43 FC

0x0e, 0x01, 0x81, 0x00, 0x00, 0x03,

0x00, 0x14, 0x53, 0x63, 0x68, 0x6e, #Schneider Electric

0x65, 0x69, 0x64, 0x65, 0x72, 0x20,

0x45, 0x6c, 0x65, 0x63, 0x74, 0x72,

0x69, 0x63, 0x20, 0x20, 0x01, 0x0c,

0x42, 0x4d, 0x58, 0x20, 0x50, 0x33, #BMX P34 20 20

0x34, 0x20, 0x32, 0x30, 0x32, 0x30,

0x02, 0x04, 0x76, 0x32, 0x2e, 0x32 #V2.2

])))

elif FC == 17:

print "ID= %d, Fun.Code= %d, Address= %d, Bytes= %d" %(ID, FC, ADR, LEN)

message = message+('Reg:'+str(LEN))

conn.send(array('B', [0,0,0,0,0,3,FC,171,1] ) )#Illegal function

F.write(ctime() + message + "\n")

else:

conn.send(array('B', [0,0,0,0,0,3,FC,171,1])) #Illegal function

print "Funtion Code %d Not Supported" %FC

F.write(ctime() + message + "\n")

exit()

sleep(1)

except Exception, e:

print e, "\nConnection with Client terminated"

F.write(ctime() + "\n")

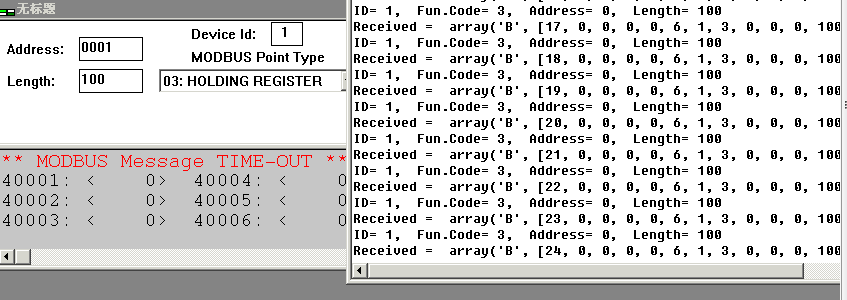
exit()

while 1:

conn, addr = s.accept()

print "Connected by", addr[0]

thread.start\_new\_thread(TCP,(conn,addr,F))

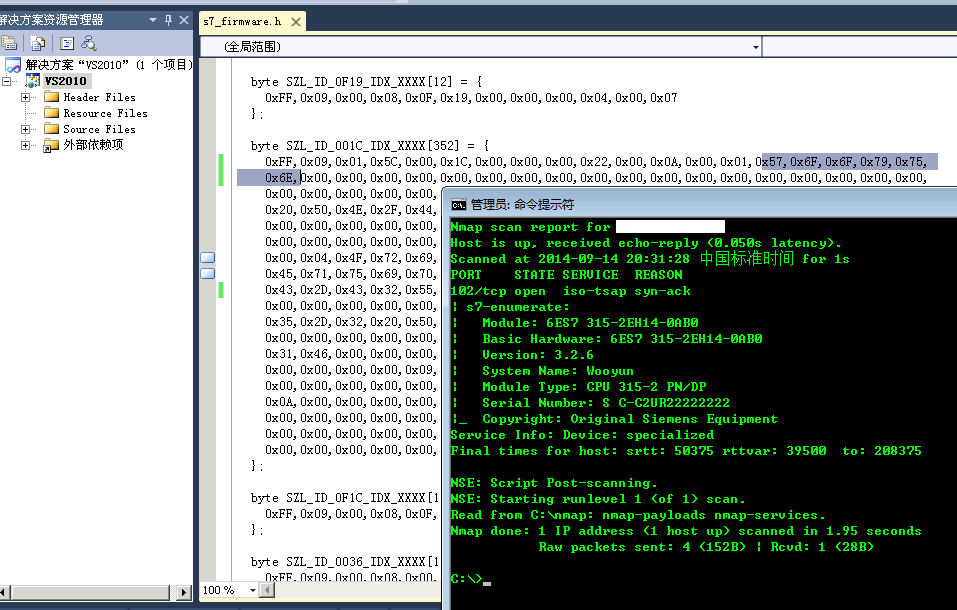


**案例二**

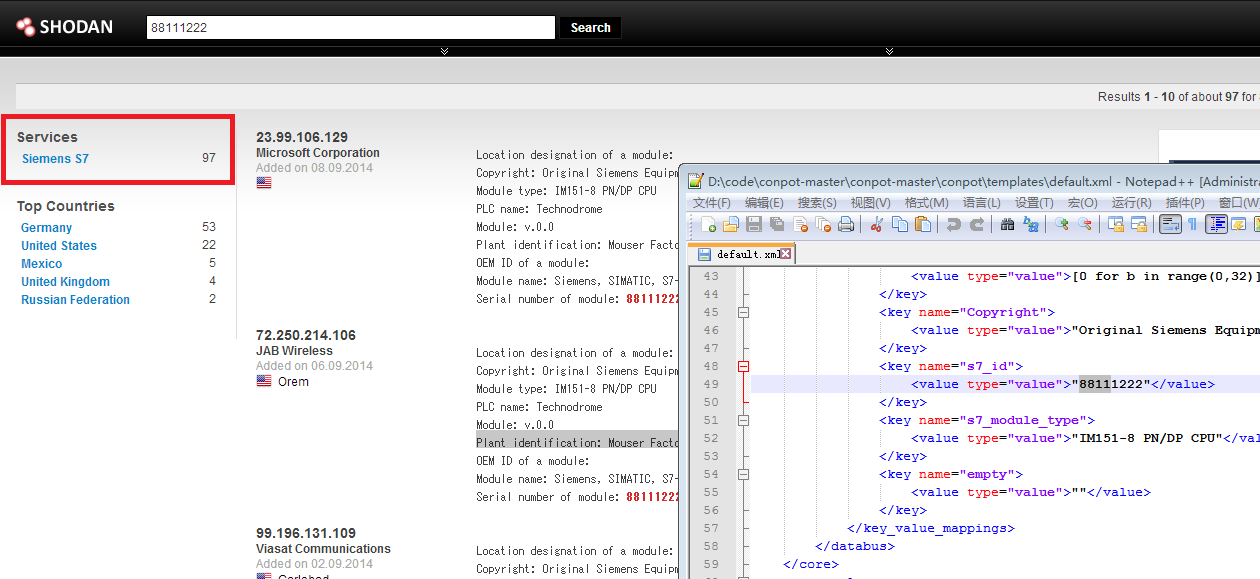
**西门子S7协议介绍**

西门子S7系列或CP模块使用以太网通信时主要基于ISO TCP (RFC1006)和西门子自有S7协议实现，并且协议详细实现由厂商自行持有，官方并未公开，外部应用与西门子S7系列PLC使用以太网通讯时可以使用西门子官方组件或者非官方实现的的开源通讯库实现，目前第三方开源的通讯库同样也实现了较多功能，如libnodave，snap7等均已解码了西门子PLC的较多操作功能。

**西门子S7协议服务端介绍**

西门子S7协议实现较为较为复杂，并且功能较多，实现高交互需要利用西门子的开源协议栈实现，以SNAP7项目为例，该项目中提供了PLC服务端的部分功能仿真，使用时只需对固定特征进行修改即可实现PLC服务端的功能。  


**0x4、注意事项**

1、 使用[Conpot](http://conpot.org/)类似开源工控蜜罐系统需要注意本地化，使用默认配置是不可取的。如下图  


Chia sẻ công nghệ mô phỏng giao thức và xây dựng hệ thống điều khiển công nghiệp honeypot

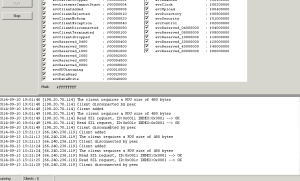
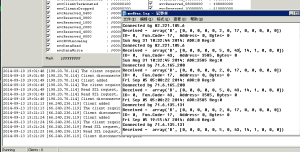
[14/09/2014](https://plcscan.org/blog/2014/09/icsscada-honeypot-and-ics-protocol-simulation-technology/)[Z-0ne](https://plcscan.org/blog/author/admin/)[Để lại một bình luận](https://plcscan.org/blog/2014/09/icsscada-honeypot-and-ics-protocol-simulation-technology/#respond) 20576 lượt xem

**0x1, Giới thiệu**

Với sự xuất hiện của các công cụ tìm kiếm không gian mạng như Eripp, Shodan và Zoomeye, sự phát triển của công nghệ quét mạng và sự gia tăng khả năng phát hiện, thiết bị điều khiển công nghiệp, thiết bị IoT, cơ sở hạ tầng, v.v., như một phần của Internet, dần dần được đánh giá cao bởi những kẻ tấn công. Từ Stuxnet năm 2010 đến Havex ngày nay, các mạng điều khiển công nghiệp, với tư cách là một mạng tương đối khép kín, ngày càng chứng kiến ​​nhiều cuộc tấn công có chủ đích và các chương trình độc hại. Sự xuất hiện của các honeypot có mục tiêu mô phỏng một số đặc điểm nhất định của hệ thống điều khiển công nghiệp cũng trở nên cần thiết. Là một phương pháp phát hiện bảo mật tương đối tích cực, honeypot được cho là sẽ được sử dụng ngày càng rộng rãi hơn trong tương lai.

**0x2. Phân tích nhật ký honeypot điều khiển công nghiệp**

**Liên kết: [Địa chỉ liên kết dự án thu thập nhật ký honeypot kiểm soát công nghiệp](https://plcscan.org/blog/dataanalysis/icsscada-honeypot-log/" \t "_blank)**

Vào tháng 6, blogger này đã xây dựng một chương trình mô phỏng giao thức thiết bị trên các nút mạng bên ngoài ở Hồng Kông và Trung Quốc đại lục để khám phá các hành vi hoạt động quét và vận hành IP nguồn đang hoạt động. Dựa trên nhật ký tích lũy, các kết luận sau đã được rút ra: A. Hiện đang nhắm mục tiêu vào các cổng cụ thể (  
tcp/102, tcp/502), các IP nguồn được quét và xác định đều đến từ nước ngoài. B. Việc xác định thỏa thuận sử dụng công nghệ công cộng đã biết. (Kịch bản liệt kê nhận dạng NSE dựa trên nmap được cấy ghép bởi plcscan và Digitalbond do ScadaStrangeLove phát hành có thể được đánh giá dựa trên các thông báo tương tác hiện nhận được. Ví dụ: khi đọc các thiết bị PLC của Siemens, các yêu cầu SZL được sử dụng để xác định thông tin kiểu thiết bị Modbus. Mã chức năng. Số 43 được sử dụng). C. Chương trình mô phỏng giao thức chưa nhận được các cuộc tấn công độc hại và hướng dẫn ghi (chẳng hạn như lệnh dừng và lệnh ghi dữ liệu của CPU Siemens PLC, v.v., và mã 05, 06 và các mã chức năng khác của modbus có chức năng ghi, vui lòng tham khảo nhật ký để biết chi tiết).  
[](https://plcscan.org/blog/wp-content/uploads/2014/09/s7comm_sim1.png)  
  
[](https://plcscan.org/blog/wp-content/uploads/2014/09/s7comm_modbus_sim1.png)

**0x3. Triển khai nhanh chóng mô phỏng giao thức**

**Trường hợp số một**

**Giới thiệu giao thức Modbus**

Giao thức Modbus là một trong những giao thức phổ biến nhất trong các mạng điều khiển công nghiệp. Giao thức này được Modcon xuất bản vào năm 1979 để sử dụng cho các bộ điều khiển logic khả trình (PLC). Bởi vì nó đơn giản và dễ sử dụng nên đã được sử dụng rộng rãi, chẳng hạn như như RTU. , PLC, cảm biến, v.v., giao thức có thể được áp dụng cho cổng Ethernet và cổng nối tiếp. Cũng vì tính sử dụng rộng rãi của nó, một số lượng lớn các lượt hiển thị trên mạng công cộng ( [kết nối truy vấn Shodan](http://www.shodanhq.com/search?q=port%3A502" \t "_blank) ).

**Tập lệnh ví dụ về máy chủ giao thức Modbus**

**Link:**[**Thông tin giao thức Modbus**](http://wenku.baidu.com/link?url=OGUWIyVZEQ6RXppHvbSwQsg91_44q-fAvGiLIe8ONTH4v_ZuycYZ7dcNSjXoD-ROjcz08KSzjFNgPS4wnjrDFBdzIRBpyUwVaJSB2kfIUzK)

#!/usr/bin/python

# phiên bản 2.7

# Mã nguồn dạng https://github.com/tecpal/PyModbus

# thay đổi Z-0ne

import os

import sys

import socket,thread

from array import array

from time import sleep, ctime

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

s.bind(('',502))

s.listen(10)

F = open('c:\modbus.log','a',0)

sys .stdout = F

def TCP(conn,addr,F):

buffer = array('B',[0]\*300)

while 1:

try:

conn.recv\_into(buffer)

ID = buffer[6]

FC = buffer[7 ]

mADR = bộ đệm[8]

lADR = bộ đệm[9]

ADR = mADR\*256+lADR

LEN = buffer[10]\*256+buffer[11]

BYT = LEN\*2

print "Received = ",buffer[0:13+buffer[12]]

if (FC < 5 và FC > 0): #Read Đầu vào hoặc Các thanh ghi

DAT = array('B')

if FC < 3:

BYT = (lambda x: x/8 if (x%8==0) else x/8+1)(LEN) #Làm tròn số byte

v = 85 #gửi 85,86.. cho byte.

for i in range(BYT):

DAT.append(v)

v = (lambda x: x+1 if (x<255) else 85)(v)

else:

for i in range(LEN): #Gửi lại địa chỉ dưới dạng dữ liệu

DAT.append(mADR)

DAT.append(lADR)

if (lADR == 255):

lADR = 0

mADR = mADR + 1

else: lADR = lADR + 1

print "ID= %d, Fun.Code= %d, Địa chỉ= %d, Độ dài= %d" %(ID, FC, ADR, LEN)

conn .send(array('B', [0,0,0,0,0, BYT+3, ID, FC, BYT]) + DAT )

elif (FC == 15 hoặc FC == 16 hoặc FC == 6 hoặc FC == 43 hoặc FC == 17): #Write Thanh ghi

BYT = buffer[12]

conn.send(array('B', [0,0,0,0,0, 6, ID, FC, mADR, lADR, buffer[10], buffer[11] ] ) )

buf = buffer[13:(13+BYT)]

message = ': ADR:'+str(ADR)+' '

if FC == 15:

print "ID = %d, Fun.Code= %d, Địa chỉ= %d, Độ dài= %d, Byte= %d" %(ID, FC, ADR, LEN, BYT)

for j in range(BYT): message = message+('Byte:'+str(j)+'='+str(buf[j])+', ') elif FC == 16: print "ID=

%

d , Fun.Code= %d, Địa chỉ= %d, Độ dài= %d, Bytes= %d" %(ID, FC, ADR, LEN, BYT) for j in range(BYT/2): message = message+(

' Reg:'+str(j)+'='+str((buf[j\*2]<<8)+(buf[j\*2+1]))+', ') elif FC == 6

:

print "ID= %d, Fun.Code= %d, Địa chỉ= %d, Bytes= %d" %(ID, FC, ADR, LEN) message = message+('Reg:'+str(LEN)) elif

FC

= = 43:

print "ID= %d, Fun.Code= %d, Địa chỉ= %d, Bytes= %d" %(ID, FC, ADR, LEN)

message = message+('Reg:'+str(LEN))

conn.send(byte(bytearray([0x00, 0x00,

0x00, 0x00,

0x00, 0x32,

0x00,

0x2b, #43 FC

0x0e, 0x01, 0x81, 0x00, 0x00, 0x03, 0x00, 0x14, 0x53, 0x63, 0x68, 0x6e

, #Schneider Electric

0x65, 0x69, 0x6 4, 0x65, 0x72, 0x20 ,

0x45, 0x6c, 0x65, 0x63, 0x74, 0x72, 0x69, 0x63, 0x20, 0x20, 0x01, 0x0c, 0x42, 0x4d, 0x58, 0x20, 0x50, 0x33

,

#BMX P34 20 20 0x34, 0x20, 0x32, 0x30 ,

0x32 , 0x30,

0x02, 0x04, 0x76, 0x32, 0x2e, 0x32 #V2.2

])))

elif FC == 17:

print "ID= %d, Fun.Code= %d, Địa chỉ= %d, Byte = % d" %(ID, FC, ADR, LEN)

tin nhắn = tin nhắn+('Reg:'+str(LEN))

conn.send(array('B', [0,0,0,0,0,3,FC,171,1] ) )#Illegal function

F.write(ctime() + message + "\n")

else :

conn.send(array('B', [0,0,0,0,0,3,FC,171,1])) #Illegal function

print "Funtion Code %d Not Supported" %FC

F.write( ctime() + message + "\n")

exit()

sleep(1)

ngoại trừ Ngoại lệ, e:

print e, "\nKết nối với máy khách bị chấm dứt"

F.write(ctime() + "\n")

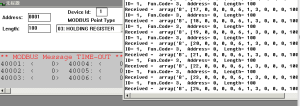
exit()

while 1:

conn, addr = s.accept()

print "Được kết nối bởi", addr[0]

thread.start\_new\_thread(TCP,(conn,addr,F))

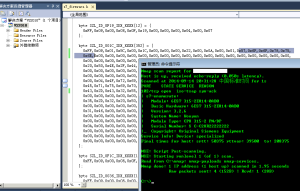
[](https://plcscan.org/blog/wp-content/uploads/2014/09/modbus_sim1.png)

**Trường hợp 2**

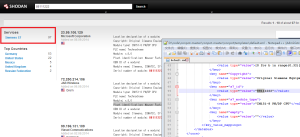
**Giới thiệu giao thức Siemens S7**

Khi các mô-đun CP hoặc dòng S7 của Siemens sử dụng giao tiếp Ethernet, chúng chủ yếu dựa trên giao thức ISO TCP (RFC1006) và S7 của Siemens, việc triển khai chi tiết giao thức này do nhà sản xuất nắm giữ và chưa được tiết lộ chính thức. PLC dòng S7 sử dụng Ethernet.Giao tiếp mạng có thể được thực hiện bằng cách sử dụng các thành phần chính thức của Siemens hoặc các thư viện giao tiếp nguồn mở không chính thức.Hiện tại, các thư viện giao tiếp nguồn mở của bên thứ ba cũng triển khai nhiều chức năng, chẳng hạn như libnodave, snap7, v.v., đã giải mã nhiều hoạt động Chức năng của PLC Siemens. .

**Giới thiệu về máy chủ giao thức Siemens S7**

Việc triển khai giao thức Siemens S7 tương đối phức tạp và có nhiều chức năng. Để đạt được tính tương tác cao, nó cần được triển khai bằng cách sử dụng ngăn xếp giao thức nguồn mở của Siemens. Lấy dự án SNAP7 làm ví dụ. Dự án này cung cấp mô phỏng một phần chức năng của PLC máy chủ. Khi sử dụng nó, chỉ cần có các tính năng cố định. Các chức năng của máy chủ PLC có thể được thực hiện bằng cách thực hiện các sửa đổi.  
[](https://plcscan.org/blog/wp-content/uploads/2014/09/s7comm_sim21.png)

**0x4.Ghi chú**

1. Khi sử dụng [Conpot](http://conpot.org/" \t "_blank) , tương tự như hệ thống honeypot điều khiển công nghiệp nguồn mở, bạn cần chú ý đến việc bản địa hóa, không nên sử dụng cấu hình mặc định. Như hình dưới đây  
[](https://plcscan.org/blog/wp-content/uploads/2014/09/conpot_s7comm_sim1.png)

追踪ICS扫描者（Trace ICS Scanner）

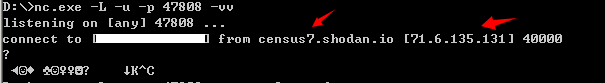
[2014/11/10](https://plcscan.org/blog/2014/11/trace-ics-scanner/)[Z-0ne](https://plcscan.org/blog/author/admin/)[Leave a comment](https://plcscan.org/blog/2014/11/trace-ics-scanner/#respond) 26754 views

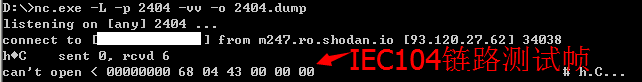
**简介**

如今针对互联网的探测性扫描越发增多，暴露在公网的PLC、嵌入式设备等作为互联网的一部分，也会因为一些识别方式的公开和研究的人员不断增多，导致针对设备的探测也会越来越多。在扫描识别这块国外的Shodan是较早着眼针对工控协议（如西门子S7PLC的TCP102端口，MODBUS设备的TCP502端口，DNP3设备的20000端口等）进行探测的组织，而国内这块CNCERT似乎已经开始了首轮的探测性扫描。如下便主要分享一些快捷监听方式和一些扫描组织的情报。

**tips分享**

在不使用类似conpot这种工控蜜罐和协议仿真程序的情况下我们可以使用短小精干的netcat来完成无交互的端口监听，完成对方扫描时发送的第一帧报文的接收分析，这样也可以帮助我们了解对方使用的识别方式和扫描来源，同样也可以辅助我们收集对方识别的方式。

如下图我们可以使用nc循环监听特定的工控协议端口，收取对方的测试报文（建议使用-o参数输出hex文件）。  


如下为IEC104协议运行在的TCP2404端口，并收到了测试帧的请求，根据IEC104的协议标准如果是真实设备收到图中报文后则会回复68 04 83 00 00 00，同我们可以新建Soketserver，并填充报文，进到下一个对方发送逻辑，辅助我们完成交互逻辑。  


**谁在扫描？**

通过对工控协议运行端口的监听、或是使用协议仿真器收集到的一些扫描日志，我们可以遍历到一些来自Shodan节点的IP，根据[ICS/SCADA Honeypot Log](https://plcscan.org/blog/dataanalysis/icsscada-honeypot-log/)收集到的扫描识别日志来看与下面列出的各个节点的IP都还是有较高的匹配度的。  
（如下IP建议黑名单）

census1.shodan.io 198.20.69.74

census2.shodan.io 198.20.69.98

census3.shodan.io 198.20.70.114

census4.shodan.io 198.20.99.130

census5.shodan.io 93.120.27.62

census6.shodan.io 66.240.236.119

census7.shodan.io 71.6.135.131

census8.shodan.io 66.240.192.138

census9.shodan.io 71.6.167.142

census10.shodan.io 82.221.105.6

census11.shodan.io 71.6.165.7

census12.shodan.io 71.6.165.200

rim.census.shodan.io 85.25.43.94

pacific.census.shodan.io 85.25.103.50

atlantic.census.shodan.io 188.138.9.50

scanner1.labs.rapid7.com 198.143.173.162

scanner2.labs.rapid7.com 71.6.216.34

如下日志显示了在9月底收到了来自国内的扫描，这也是自6六月底在外网发布TCP102端口以来首次收到国内的识别请求，近期国内IP如下：

2014-09-26 03:18:04 [202.108.211.63] Client added

2014-09-26 03:18:04 [202.108.211.63] Client added

2014-09-26 03:18:04 [202.108.211.63] Client disconnected by peer

2014-09-26 03:18:04 [202.108.211.63] The client requires a PDU size of 480 bytes

2014-09-26 03:18:04 [202.108.211.63] Client disconnected by peer

2014-09-26 03:18:04 [202.108.211.63] Client added

2014-09-26 03:18:04 [202.108.211.63] The client requires a PDU size of 480 bytes

2014-09-26 03:18:04 [202.108.211.63] Read SZL request, ID:0x0011 INDEX:0x0001 --> OK

2014-09-26 03:18:04 [202.108.211.63] Read SZL request, ID:0x001c INDEX:0x0001 --> OK

2014-09-26 03:18:04 [202.108.211.63] Client disconnected by peer

2014-11-07 23:52:49 [114.113.55.198] Client added

2014-11-07 23:52:49 [114.113.55.198] The client requires a PDU size of 480 bytes

2014-11-07 23:52:50 [114.113.55.198] Client disconnected by peer

2014-11-07 23:52:50 [114.113.55.198] Client added

2014-11-07 23:52:50 [114.113.55.198] The client requires a PDU size of 480 bytes

2014-11-07 23:52:51 [114.113.55.198] Read SZL request, ID:0x0011 INDEX:0x0001 --> OK

2014-11-07 23:52:51 [114.113.55.198] Read SZL request, ID:0x001c INDEX:0x0001 --> OK

2014-11-07 23:52:51 [114.113.55.198] Client disconnected by peer

2014-11-08 14:26:14 [114.113.55.198] Client added

2014-11-08 14:26:14 [114.113.55.198] The client requires a PDU size of 480 bytes

2014-11-08 14:26:14 [114.113.55.198] Client disconnected by peer

2014-11-08 14:26:14 [114.113.55.198] Client added

2014-11-08 14:26:15 [114.113.55.198] The client requires a PDU size of 480 bytes

2014-11-08 14:26:15 [114.113.55.198] Read SZL request, ID:0x0011 INDEX:0x0001 --> OK

2014-11-08 14:26:15 [114.113.55.198] Read SZL request, ID:0x001c INDEX:0x0001 --> OK

2014-11-08 14:26:15 [114.113.55.198] Client disconnected by peer

2014-11-09 20:00:50 [202.108.211.63] Client added

2014-11-09 20:00:50 [202.108.211.63] Client disconnected by peer

2014-11-09 20:00:51 [202.108.211.63] Client added

2014-11-09 20:00:51 [202.108.211.63] The client requires a PDU size of 480 bytes

2014-11-09 20:00:51 [202.108.211.63] Client disconnected by peer

2014-11-09 20:00:52 [202.108.211.63] Client added

2014-11-09 20:00:52 [202.108.211.63] The client requires a PDU size of 480 bytes

2014-11-09 20:00:52 [202.108.211.63] Read SZL request, ID:0x0011 INDEX:0x0001 --> OK

2014-11-09 20:00:52 [202.108.211.63] Read SZL request, ID:0x001c INDEX:0x0001 --> OK

2014-11-09 20:00:53 [202.108.211.63] Client disconnected by peer

Connected by 202.108.211.62

Received = array('B', [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])

Connected by 202.108.211.62

Received = array('B', [0, 0, 0, 0, 0, 5, 0, 43, 14, 1, 0, 0, 0])

ID= 0, Fun.Code= 43, Address= 3585, Bytes= 0

Fri Nov 07 16:13:02 2014: ADR:3585 Reg:0

**乌龙？**

lab.cert.org.cn 202.108.211.124

追影高级威胁检测系统 202.108.211.26

应急修复系统 202.108.211.34:8080

高级可持续威胁（APT）安全监测系统 202.108.211.92

SSL VPN 202.108.211.98

Máy quét dấu vết ICS

[2014/11/10](https://plcscan.org/blog/2014/11/trace-ics-scanner/)[Z-0ne](https://plcscan.org/blog/author/admin/)[Để lại một bình luận](https://plcscan.org/blog/2014/11/trace-ics-scanner/#respond) 26754 lượt xem

**Giới thiệu**

Ngày nay, ngày càng có nhiều hoạt động quét phát hiện trên Internet. PLC, thiết bị nhúng, v.v. tiếp xúc với mạng công cộng là một phần của Internet. Do sự tiết lộ của một số phương pháp nhận dạng và số lượng nhà nghiên cứu ngày càng tăng, việc phát hiện thiết bị cũng sẽ tăng thêm. Trong lĩnh vực quét và nhận dạng, Shodan nước ngoài là tổ chức trước đó tập trung phát hiện các giao thức điều khiển công nghiệp (như cổng TCP102 của Siemens S7PLC, cổng TCP502 của thiết bị MODBUS, cổng 20000 của thiết bị DNP3, v.v.), và CNCERT trong nước dường như đã bắt đầu vòng quét thăm dò đầu tiên. Sau đây chủ yếu chia sẻ một số phương pháp giám sát nhanh và một số thông tin về tổ chức quét.

**chia sẻ mẹo**

Không cần sử dụng honeypot điều khiển công nghiệp và các chương trình mô phỏng giao thức như conpot, chúng tôi có thể sử dụng netcat ngắn và gọn để hoàn thành việc giám sát cổng không tương tác cũng như hoàn tất việc tiếp nhận và phân tích tin nhắn khung đầu tiên do bên kia gửi trong quá trình quét. giúp chúng tôi hiểu các phương pháp nhận dạng và nguồn quét được bên kia sử dụng. Nó cũng có thể giúp chúng tôi thu thập các phương pháp nhận dạng của bên kia.

Như trong hình bên dưới, chúng ta có thể sử dụng nc để nghe một cổng giao thức điều khiển công nghiệp cụ thể trong một vòng lặp và thu thập các thông báo kiểm tra của bên kia (nên sử dụng tham số -o để xuất tệp hex).  
[nc-honeypot-mode1](https://plcscan.org/blog/wp-content/uploads/2014/11/nc-honeypot-mode1.png)

Sau đây là cổng TCP2404 nơi giao thức IEC104 chạy và nhận được yêu cầu khung thử nghiệm. Theo tiêu chuẩn giao thức IEC104, nếu một thiết bị thực nhận được tin nhắn trong hình, nó sẽ trả lời 68 04 83 00 00 00. Trong Tương tự như vậy, chúng ta có thể tạo một Soketserver mới, điền vào tin nhắn, đi đến logic gửi tiếp theo của bên kia và giúp chúng tôi hoàn thành logic tương tác.  
[](https://plcscan.org/blog/wp-content/uploads/2014/11/nc-honeypot-mode21.png)

**Ai đang quét?**

Bằng cách giám sát cổng đang chạy của giao thức điều khiển công nghiệp hoặc sử dụng một số nhật ký quét được thu thập bởi trình mô phỏng giao thức, chúng tôi có thể duyệt qua một số IP từ các nút Shodan. Theo nhật ký nhận dạng quét được thu thập bởi Nhật ký Honeypot của ICS/SCADA, các mục sau đây giống nhau [.](https://plcscan.org/blog/dataanalysis/icsscada-honeypot-log/) IP của mỗi nút ngoài kia vẫn có mức độ khớp cao.  
(Danh sách đen được đề xuất IP sau đây)

điều tra dân số1.shodan.io 198.20.69.74 điều tra dân

số2.shodan.io 198.20.69.98 điều tra

dân số3.shodan.io 198.20.70.114 điều tra dân số4.shodan.io 198.20.99.130 điều tra

dân số5.shodan.io 93.120.27.62 điều tra

dân

số6.shodan.io 66.240 .236.119

điều tra dân số7. shodan.io 71.6.135.131 điều tra

dân số8.shodan.io 66.240.192.138 điều tra dân

số9.shodan.io 71.6.167.142 điều

tra dân số10.shodan.io 82.221.105.6 điều tra

dân số11.shodan.io 71.6.165.7 điều tra

dân số12.shodan.io 71. 6,16 5,200

rim.điều tra dân số . shodan.io 85.25.43.94

pacific.census.shodan.io 85.25.103.50

atlantic.census.shodan.io 188.138.9.50

scanner1.labs.rapid7.com 198.143.173.162

scanner2.labs.rapid7.com 71.6.216.34

Nhật ký sau đây cho thấy bản quét trong nước đã được nhận vào cuối tháng 9. Đây cũng là yêu cầu nhận dạng trong nước đầu tiên nhận được kể từ khi cổng TCP102 được phát hành trên mạng bên ngoài vào cuối tháng 6. Các IP trong nước gần đây như sau:

2014-09-26 03:18:04 [202.108.211.63] Khách hàng đã thêm

2014-09-26 03:18:04 [202.108.211.63] Khách hàng đã thêm

2014-09-26 03:18:04 [202.108.211.63] Khách hàng Máy khách bị ngắt kết nối bởi ngang hàng

2014-09-26 03:18:04 [202.108.211.63] Máy khách yêu cầu kích thước PDU là 480 byte

2014-09-26 03:18:04 [202.108.211.63] Máy khách bị ngắt kết nối bởi ngang hàng

2014-09 - 26 03:18:04 [202.108.211.63] Khách hàng đã thêm

2014-09-26 03:18:04 [202.108.211.63] Khách hàng yêu cầu kích thước PDU là 480 byte

2014-09-26 03:18:04 [202.108 . 211.63] Đọc yêu cầu SZL, ID:0x0011 INDEX:0x0001 --> OK

2014-09-26 03:18:04 [202.108.211.63] Đọc yêu cầu SZL, ID:0x001c INDEX:0x0001 --> OK

2014-09- 26 03:18:04 [202.108.211.63] Máy khách bị ngắt kết nối ngang hàng

2014-11-07 23:52:49 [114.113.55.198] Khách hàng đã thêm

2014-11-07 23:52:49 [114.113.55.198] Khách hàng yêu cầu kích thước PDU là 480 byte

2014-11-07 23:52: 50 [114.113.55.198] Máy khách bị ngắt kết nối ngang hàng

2014-11-07 23:52:50 [114.113.55.198] Máy khách được thêm vào 2014-11-07

23:52:50 [114.113.55.198] Máy khách yêu cầu kích thước PDU là 480 byte

2014-11-07 23:52:51 [114.113.55.198] Đọc yêu cầu SZL, ID:0x0011 INDEX:0x0001 --> OK

2014-11-07 23:52:51 [114.113.55.198] Đọc yêu cầu SZL, ID :0x001c INDEX:0x0001 --> OK

2014-11-07 23:52:51 [114.113.55.198] Máy khách bị ngắt kết nối bởi ngang hàng

2014-11-08 14:26:14 [114.113.55.198] Máy khách được thêm vào

2014-11-08 14:26:14 [114.113.55.198] Máy khách yêu cầu kích thước PDU là 480 byte

2014-11-08 14:26:14 [114.113.55.198] Máy khách bị ngắt kết nối bởi ngang hàng

2014-11-08 14:26:14 [114.113.55.198] Máy khách được thêm vào

2014-11-08 14:26:15 [114.113.55.198 ] Máy khách yêu cầu kích thước PDU là 480 byte

2014-11-08 14:26:15 [114.113.55.198] Đọc yêu cầu SZL, ID:0x0011 INDEX:0x0001 --> OK

2014-11-08 14:26:15 [ 114.113.55.198] Đọc yêu cầu SZL, ID:0x001c INDEX:0x0001 --> OK

2014-11-08 14:26:15 [114.113.55.198] Máy khách bị ngắt kết nối bởi ngang hàng

2014-11-09 20:00:50 [202.108. 211.63] Khách hàng đã thêm

2014-11-09 20:00:50 [202.108.211.63] Khách hàng đã bị ngắt kết nối bởi ngang hàng

2014-11-09 20:00:51 [202.108.211.63] Khách hàng đã thêm

2014-11-09 20:00:51 [202.108.211.63] Máy khách yêu cầu kích thước PDU là 480 byte

2014-11-09 20:00:51 [202.108.211.63] Máy khách bị ngắt kết nối bởi ngang hàng

2014-11-09 20:00:52 [202.108.211.63] Máy khách được thêm vào

2014-11-09 20:00:52 [202.108.211.63 ] Máy khách yêu cầu kích thước PDU là 480 byte

2014-11-09 20:00:52 [202.108.211.63] Đọc yêu cầu SZL, ID:0x0011 INDEX:0x0001 --> OK

2014-11-09 20:00:52 [ 202.108.211.63] Đọc yêu cầu SZL, ID:0x001c INDEX:0x0001 --> OK

2014-11-09 20:00:53 [202.108.211.63] Máy khách bị ngắt kết nối ngang hàng

Được kết nối bởi 202.108.211.62

Đã nhận = mảng('B', [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]) Được kết nối bởi 202.108.211.62 Đã nhận =

mảng

( 'B', [0, 0, 0, 0, 0, 5, 0, 43, 14, 1, 0, 0, 0]) ID= 0, Fun.Code= 43, Địa chỉ= 3585, Byte= 0

Thứ

Sáu Ngày 07 tháng 11 16:13:02 2014: ADR:3585 Reg:0

**Ô long?**

lab.cert.org.cn 202.108.211.124

Hệ thống phát hiện mối đe dọa nâng cao Chaiying 202.108.211.26

Hệ thống sửa chữa khẩn cấp 202.108.211.34:8080

Hệ thống giám sát bảo mật mối đe dọa liên tục nâng cao (APT) 202.108.211.92

SSL VPN 202.108.211.98

# 西门子S7-1200 PLC识别指南与工具脚本分享（ICS Discovery Tools Releases）

[2014/11/14](https://plcscan.org/blog/2014/11/s7-plc-discovery-tools-releases/)[Z-0ne](https://plcscan.org/blog/author/admin/)[Leave a comment](https://plcscan.org/blog/2014/11/s7-plc-discovery-tools-releases/#respond) 26554 views

# 简介

S7-1200是西门子SIMATIC S7系列的一款小型、紧凑、模块化的PLC，西门子S7-1200系列PLC的CPU模块一般自集成了以太网接口，硬件形态如下图。1200支持以太网进行管理，同时支持较多的以太网服务，如WEB服务器，SNMP，PROFINET等，这为设备的识别和发现都提供了很好的基础，同样为攻击者也开放了更多的入口，根据[ICS-CERT和CVE漏洞库](https://ics-cert.us-cert.gov/advisories/ICSA-14-079-02)曾经的多个漏洞报告显示s7-1200上一些应用均存在过拒绝服务等。因为s7-1200对以太网的支持，免不了因为一些原因暴露在公网，在互联网上针对该设备的查找可以关键字和协议在[ISO-TSAP（TCP/102端口）](http://www.shodanhq.com/search?q=port%3A102)、[WEB服务（默认80端口）](http://www.shodanhq.com/search?q=Location%3A+%2FDefault.mwsl)，[SNMP服务（UDP/161）](http://www.shodanhq.com/search?q=CPU-1200)进行检索。  
[](https://plcscan.org/blog/wp-content/uploads/2014/11/s7-1200-jpg.jpg)

# 服务介绍

## TCP/102 ISO-TSAP

西门子S7-1200的TCP/102端口主要用于设备管理和数据通讯，因为一些特性的改变导致1200在通讯协议上和西门子S7-300、400有所不同，如rack和slot(机架和cpu号)默认为0，1，导致在COTP组包时目的地址不一样导致通讯失败，例如使用以前公开过的plcscan和nmap下的s7-enumerate.nse无法再获取到设备信息。读取S7-1200的SZL信息需要构造如下请求报文：

---

-- add S7-1200 packet

-- by Z-0ne plcscan.org

-- Based on S7COMM Protocol analysis plugin.

--

---

-- S7-1200 PLC usage Rack 0 Slot 1

local COTP\_0x0000 = bin.pack("H","0300001611e00000000100c0010ac1020100c2020301")

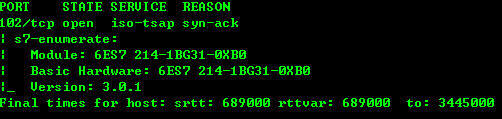
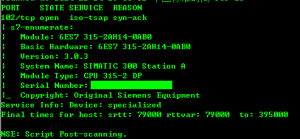
-- Setup communication 0xf0

local Setup\_comm = bin.pack("H","0300001902f080320100000c0000080000f0000001000101e0")

-- Request SZL functions Read SZL ID=0X0011

local Req\_SZL\_0x0011 = bin.pack("H","0300002102f080320700000d00000800080001120411440100ff09000400110000")

-- response is used to collect the packet responses

NMAP脚本下载：[s7-enumerate.nse](https://plcscan.org/blog/wp-content/uploads/2014/11/s7-enumerate.nse_.txt)（在Digital Bond原件脚本上增加了对S7-1200的支持，减少原有判断，增加模糊识别S7系列其他型号，S7-1200、S7-300测试通过）  
识别S7-1200如下图：  
  
识别S7-300如下：  
  
规则识别未知S7设备如下图：  
s7-enumerate-s7unkonew  
在TCP/102端口上还有另一种极不推荐的识别方式如下：

--

-- Based on TIA Portal software.

-- soft handshake Methods one

local connectpack = bin.pack("H","030000231ee00000000600c1020600c20f53494d415449432d524f4f542d4553c0010a")

-- soft handshake Methods two

-- local connectpack = bin.pack("H","0300001611e00000000800c1020600c2020600c0010a")

-- send local connection packet(IE NIC and session)

local gethwinfo = bin.pack("H","030000e502f080720100d631000004ca00000001"..

"00000120360000011d00040000000000a1000000"..

"d3821f0000a38169001515536572766572536573"..

"73696f6e5f31433943333932a3822100152c313a"..

"3a3a362e303a3a5443502f4950202d3e2042726f"..

"6164636f6d204e65744c696e6b2028544d29202e"..

"2e2ea38228001500a38229001500a3822a001516"..

"4846504654385246375052474837595f34313831"..

"3731a3822b000401a3822c001201c9c392a3822d"..

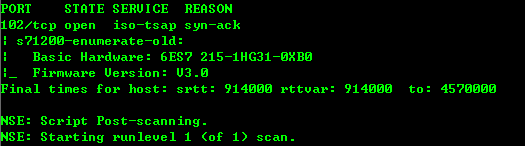
"001500a1000000d3817f0000a381690015155375"..

"62736372697074696f6e436f6e7461696e6572a2"..

"a20000000072010000")

NMAP脚本下载：[s71200-enumerate-old.nse](https://plcscan.org/blog/wp-content/uploads/2014/11/s71200-enumerate-old.nse_.txt)（注意同时间内对同一设备使多次使用该脚本扫描将会导致PLC连接堵塞，无响应或其他意想不到的情况）

识别S7-1200如下图：



## SNMP服务

西门子S7-1200的SNMP服务同样主要用于对设备的状态、连接等进行监控，可以通过对UDP的161端口进行探测，设备在sysDescr系统描述上会有详细的型号标注和固件版本以及串号等。  
特征：

sysDescr:.1.3.6.1.2.1.1.1.0

SNMP Output:

Siemens, SIMATIC S7, CPU-1200, 6ES7 212-1HD30-0XB0 SZVA3YUXXXXXX , 1, V.1.0.1, SZVA3YUXXXXXX

查询样例：[Shodan](http://www.shodanhq.com/search?q=CPU-1200)

## WEB服务

西门子S7-1200系统内嵌的WEB服务器主要用于设备的状态监控，如缓冲区日志、设备运行状态等，网页CGI程序后缀为.mwsl，比较好辨别。  
特征：

GET / HTTP/1.1

HTTP/1.1 302 Object Moved

Content-Type:text/html

Content-Length: 0

Location: /Default.mwsl

查询样例：[Shodan](http://www.shodanhq.com/search?q=Location%3A+%2FDefault.mwsl)

## PROFINET

西门子S7-1200支持PROFINET，扫描计算机可以使用组播方式请求网络内PROFINET设备，同网络内的西门子设备会主动响应。  
工具地址：[Github](https://github.com/atimorin/PoC2013/blob/master/profinet/profinet_scanner.noscapy.py)

# Hướng dẫn nhận dạng PLC và chia sẻ tập lệnh công cụ Siemens S7-1200 (Bản phát hành Công cụ Khám phá ICS)

[14/11/2014](https://plcscan.org/blog/2014/11/s7-plc-discovery-tools-releases/)[Z-0ne](https://plcscan.org/blog/author/admin/)[Để lại một bình luận](https://plcscan.org/blog/2014/11/s7-plc-discovery-tools-releases/#respond) 26555 lượt xem

# Giới thiệu

S7-1200 là PLC mô-đun nhỏ, gọn thuộc dòng SIMATIC S7 của Siemens. Mô-đun CPU của PLC dòng Siemens S7-1200 thường tích hợp giao diện Ethernet, dạng phần cứng như hình dưới đây. 1200 hỗ trợ Ethernet để quản lý và hỗ trợ nhiều dịch vụ Ethernet, chẳng hạn như máy chủ WEB, SNMP, PROFINET, v.v. Điều này cung cấp nền tảng tốt cho việc nhận dạng và khám phá thiết bị, đồng thời mở ra nhiều cơ hội hơn cho những kẻ tấn công . từ [thư viện lỗ hổng ICS-CERT và CVE, một số ứng dụng trên s7-1200 đã gặp phải tình trạng từ chối dịch vụ.](https://ics-cert.us-cert.gov/advisories/ICSA-14-079-02" \t "_blank)Vì s7-1200 hỗ trợ Ethernet nên chắc chắn nó sẽ bị lộ ra mạng công cộng vì một số lý do.Việc tìm kiếm thiết bị này trên Internet có thể sử dụng từ khóa và giao thức trong [ISO-TSAP (cổng TCP/102)](http://www.shodanhq.com/search?q=port%3A102) và [dịch vụ WEB (cổng mặc định 80)](http://www.shodanhq.com/search?q=Location%3A+%2FDefault.mwsl" \t "_blank) , [dịch vụ SNMP (UDP/161)](http://www.shodanhq.com/search?q=CPU-1200" \t "_blank) được truy xuất.  
[](https://plcscan.org/blog/wp-content/uploads/2014/11/s7-1200-jpg.jpg)

# Giới thiệu dịch vụ

## TCP/102 ISO-TSAP

Cổng TCP/102 của Siemens S7-1200 chủ yếu được sử dụng để quản lý thiết bị và truyền dữ liệu. Do có sự thay đổi ở một số tính năng nên giao thức truyền thông của 1200 khác với giao thức của Siemens S7-300 và 400, chẳng hạn như giá đỡ và khe cắm ( rack và số CPU). ) mặc định là 0, 1, dẫn đến lỗi giao tiếp do các địa chỉ đích khác nhau trong quá trình tập hợp gói COTP. Ví dụ: không thể lấy được thông tin thiết bị bằng cách sử dụng plcscan và s7-enumerate.nse đã tiết lộ trước đó trong nmap. Đọc thông tin SZL của S7-1200 yêu cầu xây dựng thông báo yêu cầu sau:

---

-- thêm gói S7-1200

-- bởi Z-0ne plcscan.org

-- Dựa trên plugin phân tích Giao thức S7COMM.

--

---

-- S7-1200 PLC sử dụng Rack 0 Slot 1

COTP\_0x0000 cục bộ = bin.pack("H","0300001611e00000000100c0010ac1020100c2020301")

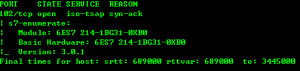
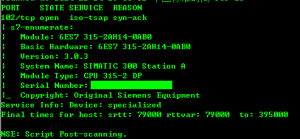
--Thiết lập giao tiếp 0xf0

Local Setup\_comm = bin.pack("H","0300001902f080320100000c0000080000f0000001000101e0")

-- Yêu cầu chức năng SZL Đọc SZL ID=0X0011

Req\_SZL\_0x0011 cục bộ = bin.pack("H","0300002102f080320700000d00000800080001120411440100ff09000400110000")

-- phản hồi được sử dụng để thu thập các phản hồi gói

Tải xuống tập lệnh NMAP: [s7-enumerate.nse](https://plcscan.org/blog/wp-content/uploads/2014/11/s7-enumerate.nse_.txt) (Thêm hỗ trợ cho S7-1200 vào tập lệnh Digital Bond gốc, giảm các phán đoán ban đầu và thêm khả năng nhận dạng mờ của các mẫu khác của dòng S7. Các bài kiểm tra S7-1200 và S7-300 đã vượt qua)  
S7 -1200 như dưới đây: Việc xác định S7-300 như sau: Các quy tắc xác định các thiết bị S7 không xác định như dưới đây: Có một phương pháp nhận dạng khác rất được khuyến khích trên cổng TCP/102 như sau:  
[](https://plcscan.org/blog/wp-content/uploads/2014/11/s7-enumerate-s71200.png)  
  
  
  
s7-enumerate-s7unkonew

--

-- Dựa trên phần mềm TIA Portal.

-- bắt tay nhẹ nhàng Phương pháp 1

gói kết nối cục bộ = bin.pack("H","030000231ee00000000600c1020600c20f53494d415449432d524f4f542d4553c0010a")

-- cái bắt tay nhẹ nhàng Phương pháp hai

-- gói kết nối cục bộ = bin.pack("H","0300001611e00000000800c1020600c2020600c0010a")

-- gửi gói kết nối cục bộ (IE NIC và phiên)

gethwinfo cục bộ = bin.pack("H","030000e502f080720100d631000004ca00000001"..

"00000120360000011d00040000000000a1000000"..

"d3821f0000a38169001515536572766572536573"..

"73696f6e5f31433943333932a3822100152c313a"..

"3a3a362e303a3a5443502f4950202d3e2042726f"..

"6164636f6d204e65744c696e6b2028544d29202e"..

"2e2ea38228001500a38229001500a3822a001516"..

"4846504654385246375052474837595f34313831"..

"3731a3822b000401a3822c001201c9c392a3822d"..

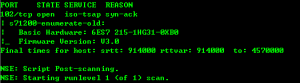
"001500a1000000d3817f0000a381690015155375"..

"62736372697074696f6e436f6e7461696e6572a2"..

"a20000000072010000")

Tải tập lệnh NMAP: [s71200-enumerate-old.nse](https://plcscan.org/blog/wp-content/uploads/2014/11/s71200-enumerate-old.nse_.txt) (Lưu ý việc sử dụng tập lệnh này để quét cùng một thiết bị nhiều lần cùng lúc sẽ gây tắc nghẽn kết nối PLC, không phản hồi hoặc các tình huống không mong muốn khác)

Xác định S7-1200 như hình dưới đây:

[](https://plcscan.org/blog/wp-content/uploads/2014/11/s71200-enumerate-1.png)

## dịch vụ SNMP

Dịch vụ SNMP của Siemens S7-1200 cũng chủ yếu được sử dụng để theo dõi trạng thái, kết nối, v.v. của thiết bị, có thể được phát hiện thông qua cổng UDP 161. Thiết bị sẽ có nhãn chi tiết model, phiên bản firmware, số sê-ri, v.v. trên mô tả hệ thống sysDescr. .  
tính năng:

sysDescr:.1.3.6.1.2.1.1.1.0

Đầu ra SNMP:

Siemens, SIMATIC S7, CPU-1200, 6ES7 212-1HD30-0XB0 SZVA3YUXXXXXX, 1, V.1.0.1, SZVA3YUXXXXXX

Ví dụ truy vấn: [Shodan](http://www.shodanhq.com/search?q=CPU-1200)

## dịch vụ WEB

Máy chủ WEB được nhúng trong hệ thống Siemens S7-1200 chủ yếu được sử dụng để theo dõi trạng thái thiết bị, chẳng hạn như nhật ký bộ đệm, trạng thái chạy thiết bị, v.v. Hậu tố chương trình CGI của trang web là .mwsl, dễ nhận biết hơn.  
tính năng:

NHẬN/HTTP/1.1

Đối tượng HTTP/1.1 302 đã được di chuyển

Loại nội dung:text/html

Độ dài nội dung: 0

Vị trí: /Default.mwsl

Ví dụ truy vấn: [Shodan](http://www.shodanhq.com/search?q=Location%3A+%2FDefault.mwsl)

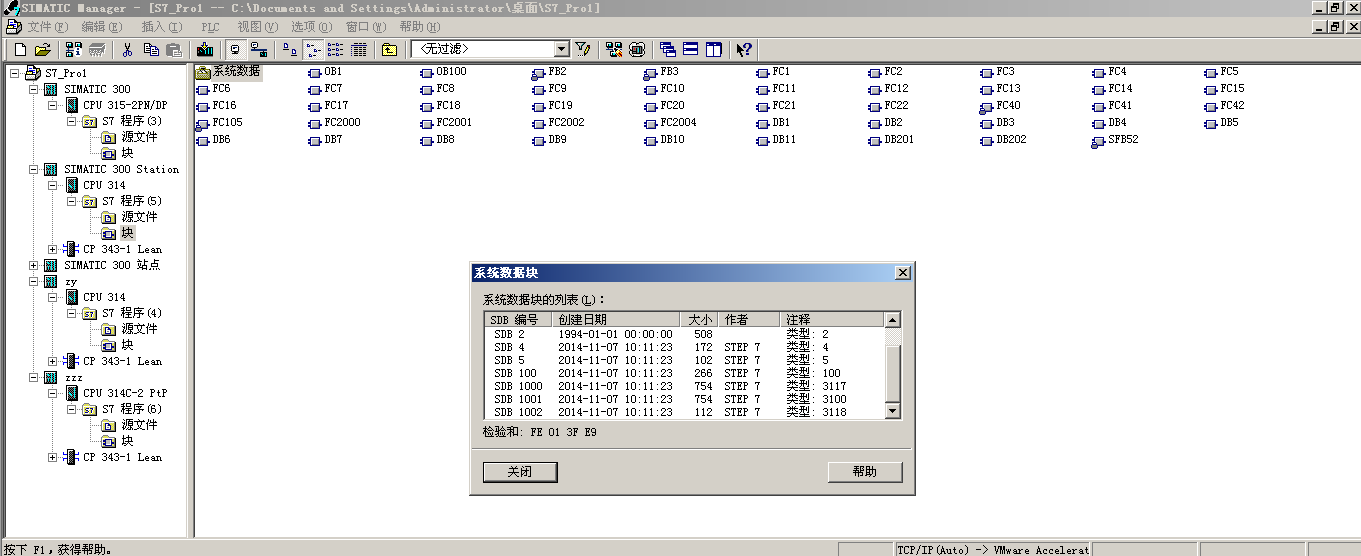
## PROFINET

Siemens S7-1200 hỗ trợ PROFINET, máy quét có thể sử dụng chế độ multicast để yêu cầu các thiết bị PROFINET trong mạng và các thiết bị Siemens trong cùng mạng sẽ chủ động phản hồi.  
Địa chỉ công cụ: [Github](https://github.com/atimorin/PoC2013/blob/master/profinet/profinet_scanner.noscapy.py" \t "_blank)

对西门子S7 PLC块(Blocks)的探索（ICS Discovery Tools Releases）

[2014/11/27](https://plcscan.org/blog/2014/11/s7-plc-list-blocks-scan/)[Z-0ne](https://plcscan.org/blog/author/admin/)[Leave a comment](https://plcscan.org/blog/2014/11/s7-plc-list-blocks-scan/#respond) 20548 views

**简介**

在西门子S7系列PLC中，用户在通过STEP7以及TIA Portal软件定义的PLC一些功能，在PLC内部主要是以不同的区块存在，已知的区块有如下几种，这里做个简单的介绍：  
**组织块（OB）（主程序块负责所有FC程序块的调用）**  
**数据块（DB）（用于存放用户和系统定义的变量数据）**  
**程序块（FC）（由用户编写的程序块）**  
**功能块（FB）（由用户编写的专用数据块）**  
**系统程序块（SFC）（调用系统某些功能时自动创建）**  
**系统功能块（SFB）（调用系统某些数据功能时自动创建）**  
**系统数据块（SDB）(由编程软件自动生成主要存放PLC的硬件组态等信息，用户无法直接打开和更改)**  
同样在工程中也有比较直观的表示，如下图：  
  
具体详细讲解可以[点击这里](http://www.dqjsw.com.cn/dianqi/zhuanti/ximenziPLC/110207.html)

**“监测”中的发现**

早在9月中旬，部署在香港的西门子PLC协议仿真服务端（基于SNAP7西门子通讯库）上，收到了来自土耳其IP比较诡异的读取设备信息请求，捕获的日志如下：

2014-09-16 17:06:33 [178.211.45.210] Client added

2014-09-16 17:06:34 [178.211.45.210] The client requires a PDU size of 480 bytes

2014-09-16 17:06:35 [178.211.45.210] Read SZL request, ID:0x0f1c INDEX:0x0000 --> OK

2014-09-16 17:06:35 [178.211.45.210] Read SZL request, ID:0x001c INDEX:0x0000 --> OK

2014-09-16 17:06:36 [178.211.45.210] Client disconnected by peer

2014-09-16 17:06:36 [178.211.45.210] Client added

2014-09-16 17:06:37 [178.211.45.210] The client requires a PDU size of 240 bytes

2014-09-16 17:06:38 [178.211.45.210] Read SZL request, ID:0x0132 INDEX:0x0004 --> OK

2014-09-16 17:06:38 [178.211.45.210] Client disconnected by peer

2014-09-16 17:06:39 [178.211.45.210] Client added

2014-09-16 17:06:39 [178.211.45.210] The client requires a PDU size of 480 bytes

2014-09-16 17:06:40 [178.211.45.210] Block info requested SDB 2000 --> NOT AVAILABLE

2014-09-16 17:06:40 [178.211.45.210] Client disconnected by peer

2014-09-16 17:21:45 [178.211.45.210] Client added

2014-09-16 17:21:45 [178.211.45.210] The client requires a PDU size of 480 bytes

2014-09-16 17:21:46 [178.211.45.210] Read SZL request, ID:0x0f1c INDEX:0x0000 --> OK

2014-09-16 17:21:47 [178.211.45.210] Read SZL request, ID:0x001c INDEX:0x0000 --> OK

2014-09-16 17:21:47 [178.211.45.210] Client disconnected by peer

2014-09-16 17:21:48 [178.211.45.210] Client added

2014-09-16 17:21:49 [178.211.45.210] The client requires a PDU size of 240 bytes

2014-09-16 17:21:50 [178.211.45.210] Read SZL request, ID:0x0132 INDEX:0x0004 --> OK

2014-09-16 17:21:50 [178.211.45.210] Client disconnected by peer

2014-09-16 17:21:51 [178.211.45.210] Client added

2014-09-16 17:21:51 [178.211.45.210] The client requires a PDU size of 480 bytes

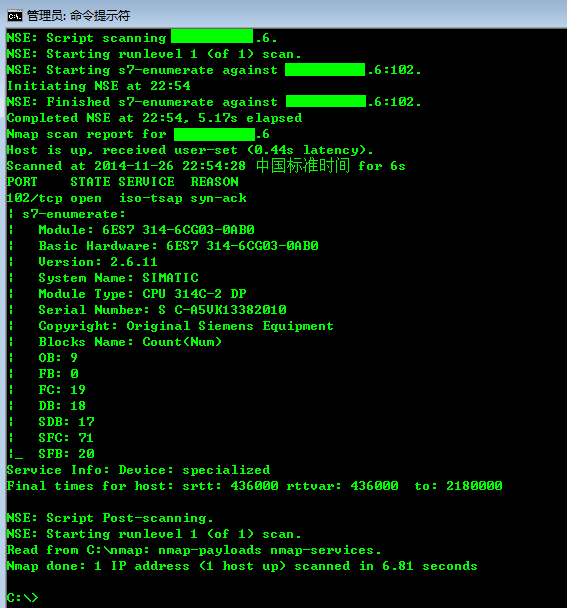
2014-09-16 17:21:52 [178.211.45.210] Block info requested SDB 2000 --> NOT AVAILABLE

2014-09-16 17:21:52 [178.211.45.210] Client disconnected by peer

根据时间顺序交互的逻辑可以判断，该情况不是来自于STEP7以及TIA Portal的这种西门子官方客户端软件，当然明显更像传统公开的识别S7 PLC硬件信息的手段（如PLCScan.py和s7-enumerate.nse），具有不同的地方是对方增加了针对S7 PLC特定的区块进行枚举，该手段不得不说是探测PLC内部状态和支持设备及功能的另一种方法，而LOG中对方想获取的SDB 2000，如果该PLC内有PROFIBUS slaves则SDB 2000块就会存在，同样在列SDB块的LIST时也会发现存在该块，那么就就能获取到SDB2000的Block info。  
这种枚举的方式可以在黑盒以及无法解密SDB块（系统数据）的MC7二进制数据的情况下，有效判断PLC内是否存在某些特定子功能、子模块，对于识别紧靠工控协议仿真的蜜罐（c[onpot](https://github.com/glastopf/conpot)）更是必杀。

**关于优化S7系列PLC识别的构思和行动**

因为digitalbond的[Redpoint](https://github.com/digitalbond/Redpoint)项目的发起，原先由scadastrangelove开源的基于python的S7系列PLC的识别工具（[PLCScan](https://code.google.com/p/plcscan/)）被移植成了基于nmap的nse脚本，而正因为这些的出现使其批量探测变得也不是那么神秘，甚至越来越多的组织加入到了工控资源的挖掘中，而已digitalbond开源s7-enumerate.nse为例，我们可以在原有基础上，构建对S7系列的其他型号支持和符合S7协议的模糊匹配（因为协议握手差异导致S7-1200无法使用原有的s7-enumerate.nse获取到信息需要重新构建握手报文，其他S7设备需要针对协议进行模糊识别），甚至包括如上的更深层次的探测，例如枚举PLC内各区块的数量等，这样可以有效快速识别当前PLC工作运行逻辑的复杂程度和定位蜜罐。  
如下图，在原有基础上增加对块的数量统计(支持S7-300/400)：



**NMAP NSE脚本获取**

[Github\_ICS Discovery Tools](https://github.com/Z-0ne/ICS-Discovery-Tools/blob/master/s7-enumerate.nse)

Khám phá Khối PLC Siemens S7 (Bản phát hành Công cụ Khám phá ICS)

[27/11/2014](https://plcscan.org/blog/2014/11/s7-plc-list-blocks-scan/)[Z-0ne](https://plcscan.org/blog/author/admin/)[Để lại một bình luận](https://plcscan.org/blog/2014/11/s7-plc-list-blocks-scan/#respond) 20548 lượt xem

**Giới thiệu**

Trong PLC dòng Siemens S7, một số chức năng PLC do người dùng xác định thông qua phần mềm STEP7 và TIA Portal chủ yếu tồn tại trong các khối khác nhau bên trong PLC.Các khối đã biết như sau.Dưới đây là phần giới thiệu ngắn gọn:Khối tổ chức (OB) (khối chương trình chính  
**) chịu trách nhiệm gọi tất cả các khối chương trình FC)**  
**khối dữ liệu (DB) (được sử dụng để lưu trữ dữ liệu biến do người dùng và hệ thống xác định)**  
**khối chương trình (FC) (khối chương trình được viết bởi người dùng)**  
**khối chức năng ( FB) (Khối dữ liệu đặc biệt được ghi bởi người dùng)**  
**Khối chương trình hệ thống (SFC) (Được tạo tự động khi một số chức năng nhất định của hệ thống được gọi)**  
**Khối chức năng hệ thống (SFB) (Được tạo tự động khi một số chức năng dữ liệu nhất định của hệ thống được gọi)**  
**Khối dữ liệu hệ thống (SDB) (Nó là được tạo ra tự động bởi phần mềm lập trình và chủ yếu lưu trữ cấu hình phần cứng và các thông tin khác của PLC. Người dùng không thể trực tiếp mở và thay đổi nó.) Ngoài ra**  
còn có một bản trình bày tương đối trực quan trong dự án, như hình dưới đây: Để giải thích chi tiết , [hãy nhấp vào đây](http://www.dqjsw.com.cn/dianqi/zhuanti/ximenziPLC/110207.html" \t "_blank)  
[](https://plcscan.org/blog/wp-content/uploads/2014/11/S7_Pro.png)

**Những phát hiện trong "Giám sát"**

Ngay từ giữa tháng 9, máy chủ mô phỏng giao thức PLC của Siemens (dựa trên thư viện truyền thông SNAP7 của Siemens) được triển khai tại Hồng Kông đã nhận được một yêu cầu khá lạ là đọc thông tin thiết bị từ một IP Thổ Nhĩ Kỳ. Nhật ký thu được như sau:

2014-09-16 17:06:33 [178.211.45.210] Khách hàng đã thêm

2014-09-16 17:06:34 [178.211.45.210] Khách hàng yêu cầu kích thước PDU là 480 byte

2014-09-16 17:06: 35 [178.211.45.210] Đọc yêu cầu SZL, ID:0x0f1c INDEX:0x0000 --> OK

2014-09-16 17:06:35 [178.211.45.210] Đọc yêu cầu SZL, ID:0x001c INDEX:0x0000 --> OK

2014 -09-16 17:06:36 [178.211.45.210] Máy khách bị ngắt kết nối bởi ngang hàng

2014-09-16 17:06:36 [178.211.45.210] Máy khách được thêm vào

2014-09-16 17:06:37 [178.211.45.210] Máy khách yêu cầu kích thước PDU là 240 byte

2014-09-16 17:06:38 [178.211.45.210] Đọc yêu cầu SZL, ID:0x0132 INDEX:0x0004 --> OK

2014-09-16 17:06:38 [178.211 .45.210] Máy khách bị ngắt kết nối ngang hàng

2014-09-16 17:06:39 [178.211.45.210] Đã thêm máy khách

2014-09-16 17:06:39 [178.211.45.210] Máy khách yêu cầu kích thước PDU là 480 byte

2014-09-16 17:06:40 [178.211.45.210] Yêu cầu thông tin khối SDB 2000 --> KHÔNG CÓ SẴN

2014 -09-16 17:06:40 [178.211.45.210] Máy khách bị ngắt kết nối ngang hàng

2014-09-16 17:21:45 [178.211.45.210] Máy khách được thêm vào 2014-09-16 17:21:45

[178.211.45.210] Máy khách yêu cầu kích thước PDU là 480 byte

2014-09-16 17:21:46 [178.211.45.210] Đọc yêu cầu SZL, ID:0x0f1c INDEX:0x0000 --> OK

2014-09-16 17:21:47 [178.211 .45.210] Đọc yêu cầu SZL, ID:0x001c INDEX:0x0000 --> OK

2014-09-16 17:21:47 [178.211.45.210] Máy khách bị ngắt kết nối ngang hàng 2014-09-16

17:21:48 [178.211.45.210 ] Khách hàng đã thêm

2014-09-16 17:21:49 [178.211.45.210] Khách hàng yêu cầu kích thước PDU là 240 byte

2014-09-16 17:21:50 [178.211.45.210] Đọc yêu cầu SZL, ID:0x0132 INDEX:0x0004 --> OK

2014-09-16 17:21:50 [178.211.45.210] Máy khách bị ngắt kết nối bởi ngang hàng

2014- 09-16 17:21:51 [178.211.45.210] Khách hàng đã thêm

2014-09-16 17:21:51 [178.211.45.210] Khách hàng yêu cầu kích thước PDU là 480 byte

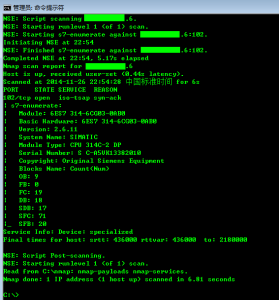
2014-09-16 17:21:52 [ 178.211.45.210] Chặn thông tin được yêu cầu SDB 2000 --> KHÔNG CÓ SẴN

2014-09-16 17:21:52 [178.211.45.210] Máy khách bị ngắt kết nối ngang hàng

Theo logic tương tác theo trình tự thời gian, có thể đánh giá rằng tình huống này không đến từ phần mềm máy khách Siemens chính thức của STEP7 và TIA Portal. Tất nhiên, nó rõ ràng giống với các phương tiện công cộng truyền thống để xác định thông tin phần cứng S7 PLC (chẳng hạn như như PLCScan.py và s7-enumerate .nse), điểm khác biệt là bên kia đã thêm các khối cụ thể để S7 PLC liệt kê, phương pháp này phải nói là một cách khác để phát hiện trạng thái bên trong của PLC và các thiết bị hỗ trợ và chức năng, còn bên kia muốn lấy LOG.SDB 2000, nếu có PROFIBUS nô lệ trong PLC thì khối SDB 2000 sẽ tồn tại.Tương tự, khi liệt kê DANH SÁCH khối SDB, bạn cũng sẽ thấy rằng khối đó tồn tại, sau đó bạn có thể lấy thông tin Khối của SDB2000.  
Phương pháp liệt kê này có thể xác định một cách hiệu quả liệu có một số chức năng phụ và mô-đun phụ nhất định trong PLC hay không trong trường hợp hộp đen và dữ liệu nhị phân MC7 không thể giải mã các khối SDB (dữ liệu hệ thống [)](https://github.com/glastopf/conpot) . .

**Ý tưởng và hành động tối ưu hóa nhận dạng PLC dòng S7**

Do sự ra mắt của dự án [Redpoint](https://github.com/digitalbond/Redpoint) của Digitalbond , công cụ nhận dạng PLC dòng S7 dựa trên python ( [PLCScan](https://code.google.com/p/plcscan/" \t "_blank) ) ban đầu có nguồn mở bởi scadastrangelove đã được cấy vào tập lệnh nse dựa trên nmap, và chính xác là do sự xuất hiện của những công cụ này, việc phát hiện hàng loạt đã trở nên ít hơn Điều bí ẩn là ngày càng có nhiều tổ chức tham gia khai thác tài nguyên điều khiển công nghiệp.Lấy mã nguồn mở digitalbond s7-enumerate.nse làm ví dụ, chúng ta có thể xây dựng hỗ trợ cho các mẫu khác của dòng S7 và kết hợp mờ phù hợp với S7 (Do sự khác biệt về bắt tay giao thức, S7-1200 không thể sử dụng s7-enumerate.nse ban đầu để lấy thông tin và cần xây dựng lại thông báo bắt tay. Các thiết bị S7 khác cần thực hiện nhận dạng mờ cho giao thức), thậm chí bao gồm cả khả năng phát hiện sâu hơn như trên, chẳng hạn như Việc liệt kê số khối trong PLC có thể xác định một cách hiệu quả và nhanh chóng mức độ phức tạp của logic vận hành công việc PLC hiện tại và xác định vị trí các honeypot.  
Như thể hiện trong hình bên dưới, số khối được thêm vào cơ sở ban đầu (hỗ trợ S7-300/400):

[](https://plcscan.org/blog/wp-content/uploads/2014/11/scan_s7_plc_block1.png)

**Thu thập tập lệnh NMAP NSE**

[Công cụ khám phá Github\_ICS](https://github.com/Z-0ne/ICS-Discovery-Tools/blob/master/s7-enumerate.nse)