INDUSTRIAL CONTROL SYSTEMS
Pentesting PLCs 101
WHO AM I?

ARNAUD SOULLIE

Senior security auditor

INTERESTS

- Windows Active Directory
  Can a Windows AD be secured? JSSI 2013
  (French, sorry)

- SCADA stuff

- Wine tasting
  (we’re not going to talk about it today)

@arnaudsoullie

solucom
management & IT consulting
LAB PREREQUISITE
WHAT’S IN THE LAB VM?

LAB EXERCISES

KALI LINUX

ADDITIONAL TOOLS
- MODBUSPAL
- MBTGET
- PLCSCAN
- SNAP7
- ...

SCRIPTS AND FILE EXAMPLES
- PCAP SAMPLES
- SCRIPTS
- SKELETONS
- ...

-
WHAT IS AN **INDUSTRIAL CONTROL SYSTEM (ICS)**?

**Corporate network**
- ERP server
- Production management
- Corporate IT

**Supervision network / SCADA**
- Supervision consoles
- Maintenance laptops
- Data Historian / Scada server

**Production network**
- RTUs
- PLCs
- Wireless industrial networks

---

**Corporate IS handle data** ≠ **ICS handle interfaces data with physical world**
ICS (Industrial Control System) = IACS (Industrial Automation and Control Systems) \sim= SCADA (Supervisory Control And Data Acquisition) \sim= DCS (Distributed Control System) Nowdays, people tend to say “SCADA” for anything related to ICS
ICS COMPONENTS

- **Sensors and actuators**: allow interaction with the physical world (pressure sensor, valves, motors, ...)

- **Local HMI**: Human-Machine Interface, permits the supervision and control of a subprocess

- **PLC**: Programmable Logic Controller: manages the sensors and actuators

- **Supervision screen**: remote supervision of the industrial process

- **Data historian**: Records all the data from the production and Scada networks and allows exporting to the corporate IS (to the ERP for instance)
Who cares?

SCADA SECURITY AWARENESS TIMELINE (SIMPLIFIED)

AINT NOBODY GOT TIME

<2011

Who cares?

FO DAT
SCADA SECURITY AWARENESS TIMELINE (SIMPLIFIED)

Who cares?
OMG!
OMG!
STUXNET!!!
2011
<2011
Who cares?

September 5, 2014
<2011
Who cares?
OMG!
OMG!
STUXNET!!!

2011
Under control
One day?

9/5/14
WHAT IS **WRONG** WITH CURRENT ICS SECURITY?

- **Organization & Awareness**
- **Network Segmentation**
- **Vulnerability Management**
- **Security in Protocols**
- **Third Party Management**
- **Security Supervision**
ICS-CERT listed over **250 attacks** on ICS in 2013

- **59%** of attacks targeted the energy sector
- **79** attacks successfully compromised the target
- **57** attacks did not succeed in compromising the target
- **120** attacks were not identified/investigated
WHAT IS A PLC?

- Real-time digital computer used for automation
- Replaces electrical relays
- Lots of analogue or digital inputs & outputs
- Rugged devices (immune to vibration, electrical noise, temperature, dust, ...)

WHAT'S INSIDE?

PROGRAMMABLE LOGIC CONTROLLER

- Applications (web server, FTP, snmp, ...)
- Programmable logic
- Middleware
- Firmware / OS
- Hardware

SIEMENS S7-1200

POWER

INPUTS

NETWORK CONNECTIVITY

OUTPUTS
“Ladder Logic” was the first programming language for PLC, as it mimics the real-life circuits.

IEC 61131-3 defines 5 programming languages for PLCs:
- **LD**: Ladder Diagram
- **FBD**: Function Block Diagram
- **ST**: Structured Text
- **IL**: Instruction List
- **SFC**: Sequential Function Chart

```plaintext
(* simple state machine *)
TxtState := STATES[StateMachine];

CASE StateMachine OF
  1: ClosingValve();
ELSE
  ;; BadCase();
END_CASE;
```

<table>
<thead>
<tr>
<th>LD</th>
<th>Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>GT</td>
<td>1000</td>
</tr>
<tr>
<td>JMPNC</td>
<td>VOLTS_OK</td>
</tr>
<tr>
<td>LD</td>
<td>Volts</td>
</tr>
<tr>
<td>VOLTS_OK</td>
<td>LD</td>
</tr>
<tr>
<td>ST</td>
<td>%Q75</td>
</tr>
</tbody>
</table>
Shodan is a search engine dedicated to find devices exposed to the Internet. It regularly scans the whole Internet IPV4 range (~4.3 billions IPs). Results are partially free (you have to pay to export the results).

**WHAT CAN YOU FIND?**

- All kinds of connected devices
  - PLCs
  - Webcams
  - Smart-things (fridge, TV, ...)
- Things you can’t even imagine...
- Example ICS report:  
  [https://www.shodan.io/report/l7VjfVKc](https://www.shodan.io/report/l7VjfVKc)

**ALTERNATIVES?**

- Scan the Internet yourself (Zmap, Massscan)
- Other online services/surveys
FUNNY THINGS YOU CAN FIND ON TEH INTERWEBS

It’s not just webcams.

THIS IS A CREMATORIUM. ON THE INTERNET.
Serial communication protocol invented in 1979 by Schneider Electric
Developed for industrial application
Royalty-free
Now one of the standards for industrial communications

**HOW IT WORKS**

- Master / Slave protocol
- Master must regularly poll the slaves to get information
- Modbus addresses are 8 bits long, so only 247 slaves per master
- There is no object description: a request returns a value, without any context or unit

**SECURITY ANYONE?**

- Clear-text
- No authentication
- Modbus was originally made for serial communications
- However it is now often used over TCP

### MODBUS/TCP FRAME FORMAT

<table>
<thead>
<tr>
<th>Name</th>
<th>Length</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transaction identifier</td>
<td>2</td>
<td>For synchronization between server &amp; client</td>
</tr>
<tr>
<td>Protocol identifier</td>
<td>2</td>
<td>Zero for Modbus/TCP</td>
</tr>
<tr>
<td>Length field</td>
<td>2</td>
<td>Number of remaining bytes in this frame</td>
</tr>
<tr>
<td>Unit identifier</td>
<td>1</td>
<td>Slave address (255 if not used)</td>
</tr>
<tr>
<td>Function code</td>
<td>1</td>
<td>Function codes as in other variants</td>
</tr>
<tr>
<td>Data bytes or command</td>
<td>n</td>
<td>Data as response or commands</td>
</tr>
</tbody>
</table>
The most common Modbus functions allow to read and write data from/to a PLC
Other functions, such as file read and diagnostics functions also exist
Undocumented Modbus function codes can also be used to perform specific actions

**COMMONLY USED MODBUS FUNCTION CODES**

<table>
<thead>
<tr>
<th>Function name</th>
<th>Function code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read coils</td>
<td>1</td>
</tr>
<tr>
<td>Write single coil</td>
<td>5</td>
</tr>
<tr>
<td>Read holding registers</td>
<td>3</td>
</tr>
<tr>
<td>Write single register</td>
<td>6</td>
</tr>
<tr>
<td>Write multiple registers</td>
<td>16</td>
</tr>
<tr>
<td>Read/Write multiple registers</td>
<td>23</td>
</tr>
</tbody>
</table>
# Modbus Protocol

## All Documented Modbus Function Codes (From Wikipedia)

<table>
<thead>
<tr>
<th>Function type</th>
<th>Function name</th>
<th>Function code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit access</td>
<td>Physical Discrete Inputs</td>
<td>Read Discrete Inputs</td>
</tr>
<tr>
<td></td>
<td>Internal Bits or Physical Coils</td>
<td>Read Coils</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Write Single Coil</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Write Multiple Coils</td>
</tr>
<tr>
<td>Data Access</td>
<td>Physical Input Registers</td>
<td>Read Input Registers</td>
</tr>
<tr>
<td>16-bit access</td>
<td></td>
<td>Read Holding Registers</td>
</tr>
<tr>
<td></td>
<td>Internal Registers or Physical Output Registers</td>
<td>Write Single Register</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Write Multiple Registers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Read/Write Multiple Registers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Mask Write Register</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Read FIFO Queue</td>
</tr>
<tr>
<td></td>
<td>File Record Access</td>
<td>Read File Record</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Write File Record</td>
</tr>
<tr>
<td>Diagnostics</td>
<td>Read Exception Status</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Diagnostic</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Get Com Event Counter</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Get Com Event Log</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Report Slave ID</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>Read Device Identification</td>
<td>43</td>
</tr>
<tr>
<td>Other</td>
<td>Encapsulated Interface Transport</td>
<td>43</td>
</tr>
</tbody>
</table>
LAB SESSION #1: ANALYZING A MODBUS COMMUNICATION WITH WIRESHARK

- Launch Wireshark
- Open "modbus1.pcap"
- Try to understand what’s going on
- What’s the value of register #123 at the end?
- Modbuspal is a modbus simulator
  
  $ > java -jar ModbusPal.jar

- Add a modbus slave

- Set some register values

- Query it with:
  - MBTGET Perl script
  - Metasploit module

- Analyze traffic with Wireshark
- Mbget is a perl script to perform Modbus/tcp queries
  $ > cd toolz
  $ > ./mbget -h

- Read requests
  - Coils (1 bit)
    $ > ./mbget -r1 -a 0 -n 8 127.0.0.1
  - Words (8 bits)
    $ > ./mbget -r3 -a 0 -n 8 127.0.0.1

- Write requests
  - Coils (1 bit)
    $ > ./mbget -w3 #{VALUE} -a 0 -n 8 127.0.0.1
  - Words (8 bits)
    $ > ./mbget -w6 #{VALUE} -a 0 -n 8 127.0.0.1
A simple modbus client that I developed
Can perform read and write operations on coils and registers
Included in msf’s trunk so you already have it 😊

Launch msf console
$ > msfconsole
msf > use auxiliary/scanner/scada/modbusclient
msf auxiliary(modbusclient) > info

Play!
msf auxiliary(modbusclient) > set ACTION
WARNING

The following show features stunts performed either by professionals or under the supervision of professionals. Never do this on live production systems.
Objective: Identify all exposed services on a device or a range of devices
Often the first step in a pentest

We will use two tools
- **Nmap**: The world’s finest port scanner
- **PLCSCAN**: A reconnaissance tool dedicated to PLCs

PLCs IP addresses
- **192.168.0.50**: Siemens S7-1200
- **192.168.0.5**: Schneider m340
The de-facto tool for port scanning
- Can be really dangerous on ICS

Two stories from NIST SP800-82
- A ping sweep broke for over 50 000$ in product at a semi-conductor factory
- The blocking of gas distribution for several hours after a pentester went slightly off-perimeter during an assessment for a gas company

Nmap useful setup for ICS scanning
- Reduce scanning speed! Use `--scan-delay=1` to scan one port at a time
- Perform a TCP scan instead of a SYN scan
- Do not perform UDP scan
- Do not use fingerprinting functions, and manually select scripts (do not use `–sC`)

Nmap output for 192.168.0.50
- Host is up (0.013s latency), Not shown: 65532 filtered ports
- Port STATE SERVICE
  - 80/tcp open http
  - 135/tcp open iso.tsap
- MAC Address: 00:1C:06:18:03:62 (Siemens Numerical Control, Nanjing)
- Nmap done: 1 IP address (1 host up) scanned in 129.66 seconds
Scans for ports 102 (Siemens) and 502 (Modbus) and tries to pull information about the PLC (modules, firmware version,...)

Not exhaustive since not all PLCs use Modbus or are Siemens

https://code.google.com/p/plcscan/
by SCADAStrangeLove (http://scadastrangelove.org/)
Most PLCs have standard interfaces, such as **HTTP** and **FTP**. Let’s say security was not the first thing in mind when introducing these features ...

**Schneider M340**
- Connect to the webservice
- Default password
- Hardcoded password?
- Take a look at Java applets!
LAB SESSION #5: ATTACKING ICS PROTOCOLS

- Modbus
  - Scan for registry values using `mbtget`
  - Python / Ruby / Perl / PHP, your call!

- Unauthenticated actions
  - STOP/RUN
    
    ```
    msf > use auxiliary/admin/scada/modicon_command
    ```

  - Logic download/upload
    
    ```
    msf > use auxiliary/admin/scada/modicon_stux_transfer
    ```
WHAT CAN WE DO ABOUT IT?

It’s difficult, but not all hope is lost.

NETWORK SEGMENTATION
- Do not expose your ICS on the Internet
- Do not expose all of your ICS on your internal network
- Use DMZ / Data diodes to export data from ICS to corporate network

PATCH WHEN YOU CAN
- Patching once a year during plant maintenance is better than doing nothing

APPLY CORPORATE BEST PRACTICES
- Change default passwords
- Disable unused services

SECURITY SUPERVISION
- IPS have signatures for ICS
- Create your own signatures, it is not that difficult

Y U NO SECURE ICS?

THE COST IS TOO DAMN HIGH!
The power of simplicity

«Ce qui est simple est fort»