News from Camp 4

Reid Wightman

wightman@digitalbond.com

AppSecDC, 2012
Today

- Quick Recap
- New ‘sploits
- Dear Vendor
What’s a PLC

• **Programmable Logic Controller**

• **Inputs and Outputs**
  – Input example: thermometer in a mash tun
  – Output example: heater element and pump motor on a mash tun

• **Program sez:**
  – Keep the temperature @ 153-156F for one hour
  – After the timer expires, turn on the pump on to move the mash to fermentation tank

• **PLC reports to HMI:** How is the beer coming?
Quick recap

- GE D20
  - Security via one protocol (TELNET) but not another (TFTP, LogicLinx)
  - Bad guys get full access (read/write) to configuration, plus plaintext passwords, ability to write new ladder logic, etc
Quick Recap 2

• Schneider Modicon
  - Security via one protocol (HTTP) but not others (FTP/TELNET/Modbus)
  - Bad guys get full access (read/write) to configuration, plus plaintext passwords
Quick Recap 3

- **Koyo ECOM100**
  - Security via one protocol (HAP) but not another (HTTP)
  - HAP protocol features small password space, easy to bruteforce
Quick Recap 4

- Rockwell ControlLogix
  - Security via one protocol (EIP) but not another (err...EIP)
  - Bad guys can kill controller remotely
<3 Metasploit

- Building as many vulnerability demonstrations as possible into MSF
- Let everybody see just how easy it is to kill controllers
D20 Modules

- **d20tftpbd** – provides asynchronous command line via TFTP
- **d20pass** – retrieves configuration via TFTP, parses config, stores usernames + passwords as loot
- **d20_tftp_overflow** – triggers buffer overflow in TFTP service. Currently DoS, likely RCE.
Modicon Modules

• modicon_password_recovery – retrieve passwords
  - HTTP, Write Password are plaintext
  - FTP Password uses vxworks loginDefaultEncrypt() – easily reversed

• Two new modules today…wait for it.
Allen-Bradley ControlLogix

• multi_cip_command – Three payloads derived from Rubén Santamarta’s C code
  – STOP the CPU
  – Crash the CPU
  – Crash the Ethernet card
Koyo

- koyo_login – Brute-force Koyo ECOM passwords
Inconsistent Security
To view this page, you must log in to this area on 192.168.63.253:80:

Schneider Web

Your password will be sent unencrypted.

Name: 

Password: 

[ ] Remember this password in my keychain

[Cancel] [Log In]
<table>
<thead>
<tr>
<th>User</th>
<th>Time Form</th>
<th>Date Sep</th>
<th>Date For</th>
<th>Application Control</th>
<th>Monitor Access Level</th>
<th>User Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>🕒 24 hr</td>
<td>🕒</td>
<td>🕒 YYYY M</td>
<td>🕒</td>
<td>🕒 Read/Write</td>
<td>westronic</td>
</tr>
<tr>
<td>2</td>
<td>🕒 24 hr</td>
<td>🕒</td>
<td>🕒 YYYY M</td>
<td>🕒</td>
<td>🕒 Read/Write</td>
<td>reid</td>
</tr>
<tr>
<td>3</td>
<td>🕒 24 hr</td>
<td>🕒</td>
<td>🕒 YYYY M</td>
<td>🕒</td>
<td>🕒 Read Only</td>
<td>readonly</td>
</tr>
<tr>
<td>4</td>
<td>🕒 24 hr</td>
<td>🕒</td>
<td>🕒 YYYY M</td>
<td>🕒</td>
<td>🕒 Maintenance</td>
<td>maintenance</td>
</tr>
</tbody>
</table>

*Time display format.*

Port Configuration | Modem Command Strings | **User Configuration** | Localized Text Strings
To view this page, you must log in to area "GoAhead" on 192.168.63.240:80.
Your login information will be sent securely.

Name:

Password:

☐ Remember this password in my keychain

Cancel  Log In
The search for more exploits
Stuxnet + Beresford Fun

- Stuxnet showed ladder-logic ‘hooking’
- Beresford showed weakness in Siemens controllers that ‘bad guys’ could use
- Most PLCs have the ‘Beresford/Stuxnet’ vuln
  - Download ladder logic without authentication
  - Upload ladder logic without authentication
  - Code-hooking can combine the two with a simple Langner-style ‘logic time bomb’
Modicon

- Modbus used for Engineering Access
- Special Function Code 90: “Unity”
  - Lets us STOP the CPU
  - Lets us retrieve/overwrite ladder logic
from the Automation department...

Quantum Modbus TCP Communication

Posted by hungdnq on 24 November, 2010 - 2:43 am

Try to write my own code to connect Quantum PLC, 140 NOE 771 01 Ethernet Module, using Modbus TCP/IP. I used Analyser to capture the packets when PLC and Unity Pro connected. every packets look likes:
Ethernet part/ 80 2C 00 00 00 05 00/ 5A 00 01 00

In Modbus 5A must be the function code? What does it mean? anyone can decode the ADU of the Modbus message?

Thank you in advance.

hung

Reply to this post...
Modicon

- No authentication for any operation, but...
- ...Unity == Chatty
- Quick Python code to isolate packets
  - Walk through .pcap, find unique packets
  - Analyze in Wireshark
  - Find Ladder Logic Upload/Download
  - Find CPU STOP
- Replay commands
Modicon - CPU STOP

- ~100 packets to initialize conversation
- One packet to STOP CPU
  - FC 90
  - Payload 0x01, 0x41, 0xff, 0x00
  - (Start is 0x01, 0x40, 0xff, 0x00)
Project Browser

Structural view

Project
  Configuration
    1: Local Bus
  Derived Data Types
  Derived FB Types
  Variables & FB instances
    Elementary Variables
    Derived Variables
    10 Derived Variables

{MASTSECTION1: [MAST]} : 0 error(s), 0 warning(s)
{Evt0 <I/O Event>} : 0 error(s), 0 warning(s)
Process succeeded : 0 Error(s), 0 Warning(s)
0 warning(s)
Modicon - Logic Upload

- ~100 packets to initialize (same)
- Split into ~240 byte blocks (max size of Modbus packet + some overhead bytes)
- Second block must be sent twice
  - No idea why
  - Repeated testing
Modicon – Logic Upload

…But…what file do we transfer?
Great question!
<table>
<thead>
<tr>
<th>Offset</th>
<th>00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000000</td>
<td>50 4B 03 04 14 00 00 00 08 00 B3 71 86 3F 05 7A</td>
</tr>
<tr>
<td>00000010</td>
<td>56 F1 90 7C 00 00 00 10 02 00 09 00 00 00 41 53</td>
</tr>
<tr>
<td>00000020</td>
<td>50 52 4F 47 2E 64 62 EC 5B 79 78 14 55 B6 3F 55</td>
</tr>
<tr>
<td>00000030</td>
<td>09 A4 B3 80 09 20 84 3D 2C 8A 84 00 21 40 08 04</td>
</tr>
</tbody>
</table>

*Ke...*  
*Vñ.*  
*PROG*
Back to the PCAP

Block 7 contains strings to search for

| 0000 | 00 50 56 f0 fc 13 00 0c | 29 7a 52 bd 08 00 45 00 | .PV..... )zR...E. | ...
| 0010 | 01 2c 03 ba 40 00 80 06 | 81 3f c0 a8 b3 84 c0 a8 | ...@..... ?...... |
| 0020 | 3f fd 04 14 01 f6 f9 1b | 2d e7 97 c9 82 a1 50 18 | ...j;..... Z.1 |
| 0030 | f9 90 6a 3b 00 00 00 bf | 00 00 00 fe 00 5a 01 31 | ........ Projec |
| 0040 | 00 01 06 00 f4 00 00 00 | 00 00 50 72 6f 6a 65 63 | t...GIKY .....V4. |
| 0050 | 74 00 00 00 00 47 49 4b 59 | 00 00 00 00 00 56 34 2e | l........... |
| 0060 | 31 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | ................. |
| 0070 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | ................. |
| 0080 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | ................. |
| 0090 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | ................. |
| 00a0 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | ................. |
| 00b0 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | ................. |
| 00c0 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | ................. |
| 00d0 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | ................. |
| 00e0 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | ................. |
Files

- APX == STL (Statement List)
- APB == FBD (Function Block Diagram)
- Multiple blocks become one file
- Both types may be used at once
Simple attack: Overwrite

- Overwrite a remote Modicon to do nothing
- Alt: randomly operate outputs
- Metasploit module shows how it’s done
More complicated: Stuxnet

- Retrieve logic from remote system
- Parse it and wrap it
  - Parsing the output probably the hardest step
  - Alt: Just use Unity to edit the file (it’s what the pros would do)
- Re-upload
Time spent

• Level of difficulty: miniscule
• < 8 hours from first packet capture to successful file upload/download
Password (Un)protection

- Password protection applies to APX and APB files
- Does not prevent overwrite of existing files
- Does prevent Unity from opening the ‘source code’
- Protection is really crappy
Password (Un)protection

Properties of Project

General | Protection | Identification | Comment
---|---|---|---

Sections

Protection active

Change password ...

Clear password ...

OK | Cancel | Apply | Help
Password (Un)protection

[Image of a password protection dialog box with fields for old password, new password (with an option to encrypt), entry, and confirmation]
Back to the PCAP

Block 7 contains strings to search for

0000 00 50 56 f0 fc 13 00 0c 29 7a 52 bd 08 00 45 00
0010 01 2c 03 ba 40 00 80 06 81 3f c0 a8 b3 84 c0 a8
0020 3f fd 04 14 01 f6 f9 1b 2d e7 97 c9 82 a1 50 18
0030 f9 90 6a 3b 00 00 00 bf 00 00 00 fe 00 5a 01 31
0040 00 01 06 00 f4 00 00 00 00 00 50 72 6f 6a 65 63
0050 74 00 00 00 47 49 4b 59 00 00 00 00 00 56 34 2e
0060 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0090 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00a0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00b0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00c0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00d0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00e0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

.PV..... )zR...E.
...@... ?????....
????? ------- P.
..j; ...... Z:1
............. Projec
t...GIKY .......V4.
l..............
..............
..............
..............
..............
..............
..............
..............
349 DLLs!
Password (Un)protection

• Password stored in .APX and .APB files
• Offset ~0x4C8
• Password in plaintext is...plaintext
• Password ‘encrypted’:
  - aaaa => YCOG
  - aaab => 5BB1
  - aaba => 5BDA
  - abaa => 5U1B
  - baaa => 5BBU
Password (Un)protection

- Project is not encrypted, only password is
- Change password to known-value (hexedit)
- Modicon “Password Proxy” could strip password
Password (Un)protection

Modicon password is dumb no matter who you are:
- Hate security? It’s annoying!
- Love security? It doesn’t do anything!
‘modiconstux’ and ‘modiconstop’ available today
  - Overwrite the ladder logic in that pesky controller
  - STOP or RUN your (least) favorite controller

Password proxy stripper TBA

Aside from that, it’s been p0wn3d enough

What I want:
Schneider -- give me a security roadmap so that I can start recommending your products.
More New Stuff: WAGO

• Russian group DsecRG released vulns with Basecamp
  - CSRF
  - default credentials
• Lots of other vulns + backdoors + FUN!
WAGO

- My model: IPC 758-870
- 266Mhz x86, 32MB flash, 32MB ram
- Linux 2.4.31
WAGO

• Hard-Coded user accounts
  - guest/guest
  - user/user00
  - root/ko2003wa (requires su)

• Open telnet, ftp services
  - Upload files and run them
  - Just like any other Linux box
  - Successfully compiled tinyproxy, tor
Macintosh-3:~ krweightm$ telnet 192.168.63.240
Trying 192.168.63.240...
Connected to 192.168.63.240.
Escape character is '^]'.

Linux 2.4.31-adeos (192.168.63.200) (pts/0)

10.0.0.201 login: user
Password:
   -sh-3.00$ su
Password:
   -sh-3.00# passwd
Changing password for root
Enter the new password (minimum of 5, maximum of 8 characters)
Please use a combination of upper and lower case letters and numbers.
Enter new password:
Re-enter new password:
passwd: An error occurred updating the password file.
   -sh-3.00#
WAGO Ladder Logic

3S-Software CoDeSys
- Most amazing ladder logic implementation ever
- Used by hundreds of manufacturers
- Security--
CoDeSys - How it works

1) Engineer writes their logic
2) Engineering software compiles binary
3) Binary transferred to PLC (no authentication!)
4) PLC loads binary into memory, jumps inside
Remember PLC Notes?

- Very few PLCs use MMU
- WAGO does (Linux on x86, yay)
- ...But the CoDeSys process runs as root
CoDeSys Project Format

• Header
• X86 binary
• Footer
• Don’t really need to understand it to exploit it
CoDeSys Project Format

- World’s longest NOP-sled?
- ~750kb of NOPs followed by a bind shell
- Uploaded to WAGO
- Sadly, FAIL – CRC failure
- Need to RE the CRC (32-bit CRC, stored as .CHK file on filesystem)
- Expect an update and metasploit poc in a few weeks
Macintosh-3:~ krwightm$ telnet 192.168.63.240
Trying 192.168.63.240...
Connected to 192.168.63.240.
Escape character is '^['.

Linux 2.4.31-adeos (192.168.63.200) (pts/0)

10.0.0.201 login: user
Password:

- sh-3.00$ ls
- 1
- 20
- 268435455
- sh-3.00$ hexdump DEFAULT.CHK
0000000 1796 03e3
0000004
- sh-3.00$
Dear 3S-Software

• You are in an amazing position to promote secure ladder logic transfer
• A little goes a long way in this area
Basecamp responses

• A-B decent
  – gave quick mitigation information
  – provided Snort signatures
  – …still waiting for long-term fix for CIP

• Schneider has said little since Rubén’s backdoor disclosure
  – “We take security seriously…”
  – (Nevermind the backdoors + other flaws)

• GE has shared nothing

• Koyo has shared nothing