SAND2016-7542PE



# Process Logic Verification & Analysis

Sarah Hostetler, Jonathan Van Houten,

Sam Mulder

U.S. DEPARTMENT OF ENERGY | NNSA

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2011-XXXXP

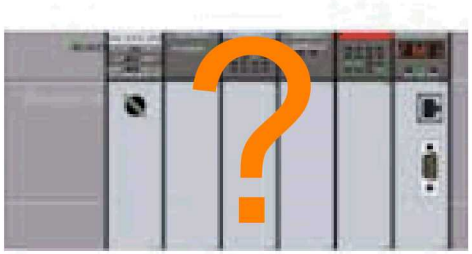
# Process logic is a PROGRAM which

Sandia National Laboratories

- Runs “in” the Programmable Logic Controller (PLC)
- Controls the physical process
  - Relays
  - Switches
  - Gauges

- Is NOT low-level PLC firmware
- Is NOT PLC embedded Operating System
- Is NOT the business logic for the whole plant or process

# Motivation for new tools



How do we verify the condition of the PLC?

Would you ask a bank robber to verify the condition of the vault? No!

We don't trust tools running on a compromised network to verify the condition of the PLC. We need independent tools.

# Independent tools

Sandia National Laboratories

What are the options?

1. Clean copy of vendor software
2. Commercial tools
3. National Laboratories tools

- Clean copy of vendor software
  - $$$ expensive licensing
  - Big learning curve to know all the vendor tools
  - Vendor software is itself a target
    - Malware may know how to avoid or compromise it
  - No advanced or automated analysis capabilities

# Independent tools (continued)

Sandia National Laboratories

- Commercial tools
  - $$ can be expensive and have yearly license
  - Not able to directly drive requirements
  - May not cover vendors of interest or multiple vendors

- National Laboratory developed tools
  - $ initial expense, but then shareable with no licenses
  - Leverage existing analysis tools and expertise from many tech fields
  - Can drive requirements and vendor selections

# Goal for process logic tool

Sandia National Laboratories

- One PLC extraction & analysis tool for many vendors
  - Requires no licensing from vendors
  - Extracts PLC program over network (no physical access)
  - Runs on any Python-capable Operating System
  - Integrates with existing advanced logic analysis tools
    - Oxide / Inquisit
  - Not a common target for attack
    - Harder to acquire than vendor software
  - Same user interface, different PLCs
    - Easy to use
    - Does not require user to learn different tools for each vendor

# Ethernet access to process logic

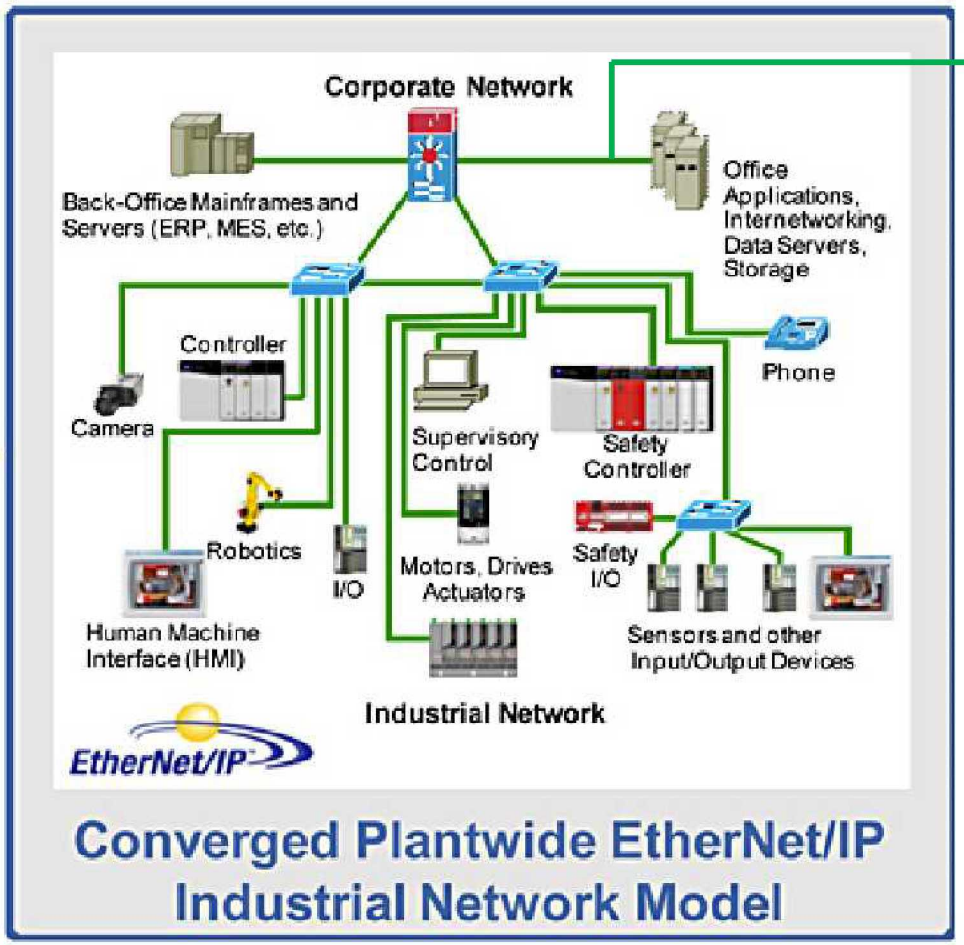



Image Source: Rockwell Automation whitepaper


PLCs are programmed over the network, thus Process Logic programs are accessible via Ethernet

Protocol used depends on vendor

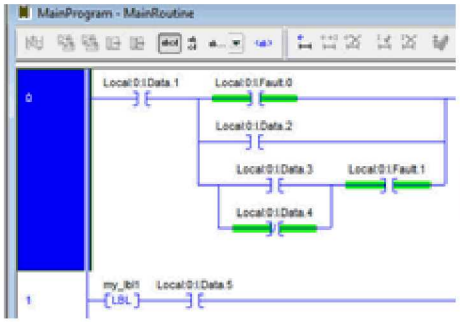
- Profinet for Siemens
- Common Industrial Protocol (CIP) over EtherNet/IP for Allen-Bradley

# There and back again

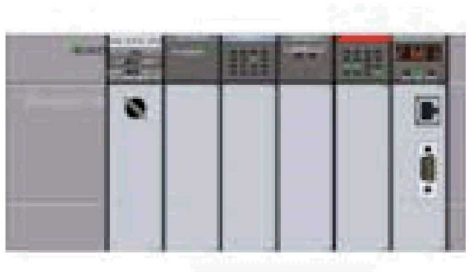
Sandia National Laboratories

**Human readable --> Binary code --> Human readable**

- The process logic program starts as human readable code
- It is converted into a binary executable and stored in the PLC
- PLC Extraction & Analysis tool (PEAT) extracts this binary
- PEAT puts the binary back into human readable format
- Analysis and comparison to original can now be performed




Compile



Decompile

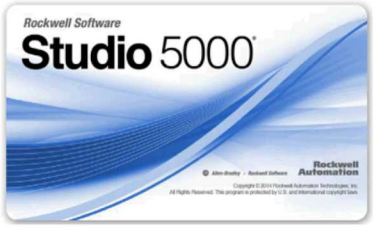
python™
PEAT

Vendor Design & Programming Tool

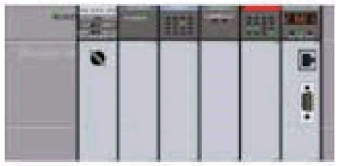
Process Logic Program Binary

PLC Extraction & Analysis Tool

# Example AB Relay Ladder Logic (RLL)




Compile



Decompile

python™
PEAT

```
<Routine Name="MainRoutine" Type="RLL">
<RLLContent>
<Rung Number="0" Type="N">
<Text>
<![CDATA[XIC(Local:0:I.Data.1)XIC(Local:0:I.Data.0)XIO(Local:0:I.Data.2)
OTE(Local:4:O.Data.0);]]>
</Text>
</Rung>
</RLLContent>
```

RLL in XML (.L5X) file

```
48E10978830F00000C4280410C42
80400C42804AFC4080602085277A
```

RLL bytecode in PLC

```
Decompiled:
[0x00278504]  7809e148  RUNG:
[0x0027850c]  4180420c
XIC(Local:0:I+0x00000004.1)
[0x00278510]  4080420c
XIC(Local:0:I+0x00000004.0)
[0x00278514]  4a80420c
XIO(Local:0:I+0x00000004.2)
[0x00278518]  608040fc
OTE(Local:4:O+0x00000000.0)
```

RLL output of PLC
Extraction & Analysis
“PEAT” program

# Process logic takes several forms

Sandia National Laboratories

- Siemens S7 – All encoded to MC7 bytecode (binary)
  - Structured Text
  - Ladder Logic
  - Sequential Function Charts
- Allen-Bradley ControlLogix – Both binary and string
  - Binary
    - Relay Ladder Logic
  - Encoded String
    - Sequential Function Charts
    - Structured Text
    - Function Block Diagrams
- Based on IEC 61131-3

# Siemens progress

Sandia National Laboratories

- PLC extraction tool complete
  - Supports Siemens S7-300, S7-400
  - Leverages open source Snap7 library to extract MC7 bytecode over Ethernet
  - Available as plug-in for Inquisit
- PLC module for Inquisit complete
  - Automatically creates collections based on “block type”
  - Immediately determines which blocks are different between two sets of extracted logic
  - Converts extracted binary into readable “Structured Text”

# Allen Bradley and others progress

Sandia National Laboratories

- Allen Bradley PLC extraction tool complete
  - Supports Logix5000 controllers
  - Uses Common Industrial Protocol (CIP) to extract binary
  - Currently using CIP over EtherNet/IP
  - CIP works over DeviceNet and ControlNet so method is portable
- Allen Bradley module for Inquisit in progress
- GE PLC extraction tool planned

# Why is this work important to ICS-CERT?

- Reduces cost of analyzing process logic from multiple vendors
- Provides independent verification
  - Out-of-band solution in case vendor SW is compromised
- Interfaces with analysis tools (Oxide/Inquisit) so anomalies can be found efficiently
- Displays metadata not available in some vendor's software

# How does this work fit into the ICS-CERT workflow?

Sandia National Laboratories

- Enables analysis of process logic programs actually running on PLCs in ICS-Network
- Provides comparison to original “golden master” program through human-readable code
- Integrates with analysis tools such as Oxide/Inquisit for quick triage of collected PLC code
- Vendors supported can be selected to meet ICS-CERT assessment needs

# Discussion Topics

Sandia National Laboratories

- Leverage PC world techniques for PLCs
- Research dangerous code constructs in logic binaries
  - Catalog
    - PC examples: “use after free”, “execute in non-code space”
    - Use results from industry collectives - Digital Bond / S4
    - Can we access others?
  - Analyze
    - Identify bugs that lead to vulnerabilities
    - Is a pattern of quick iterations (set/change/set/change) a red flag or is it normal operations most of the time?
    - Are some patterns bad for one industry (water treatment) but okay for another (power plant)?
- Can we automate discovery of malicious indicators?

# Next – Feasibility of Automating Malware Indicators Discovery

Sandia National Laboratories

- Scope the questions
  - What has been done to characterize ladder logic indicators?
  - What ladder logic malware examples can we collect?
- Investigate industry initiatives
- Look for Lab partners whose research we could apply to this problem set

# Questions?

Sandia National Laboratories