New York University Abu Dhabi
Modern Microprocessor Architectures Lab
nyuad.nyu.edu/momalab

Hardware-based solutions for critical infrastructure security

Mihalis Maniatakos

Associate Professor, NYU Abu Dhabi

@realmomalab

NYU Abu Dhabi

NYU
at a glance

Global campus locations:

New York City
Abu Dhabi
Berlin
Buenos Aires
Florence
Ghana
London
Madrid
Paris
Prague
Shanghai
Sydney
Tel Aviv
Washington DC.

Critical Infrastructure Sectors

As defined by the US Department of Homeland Security

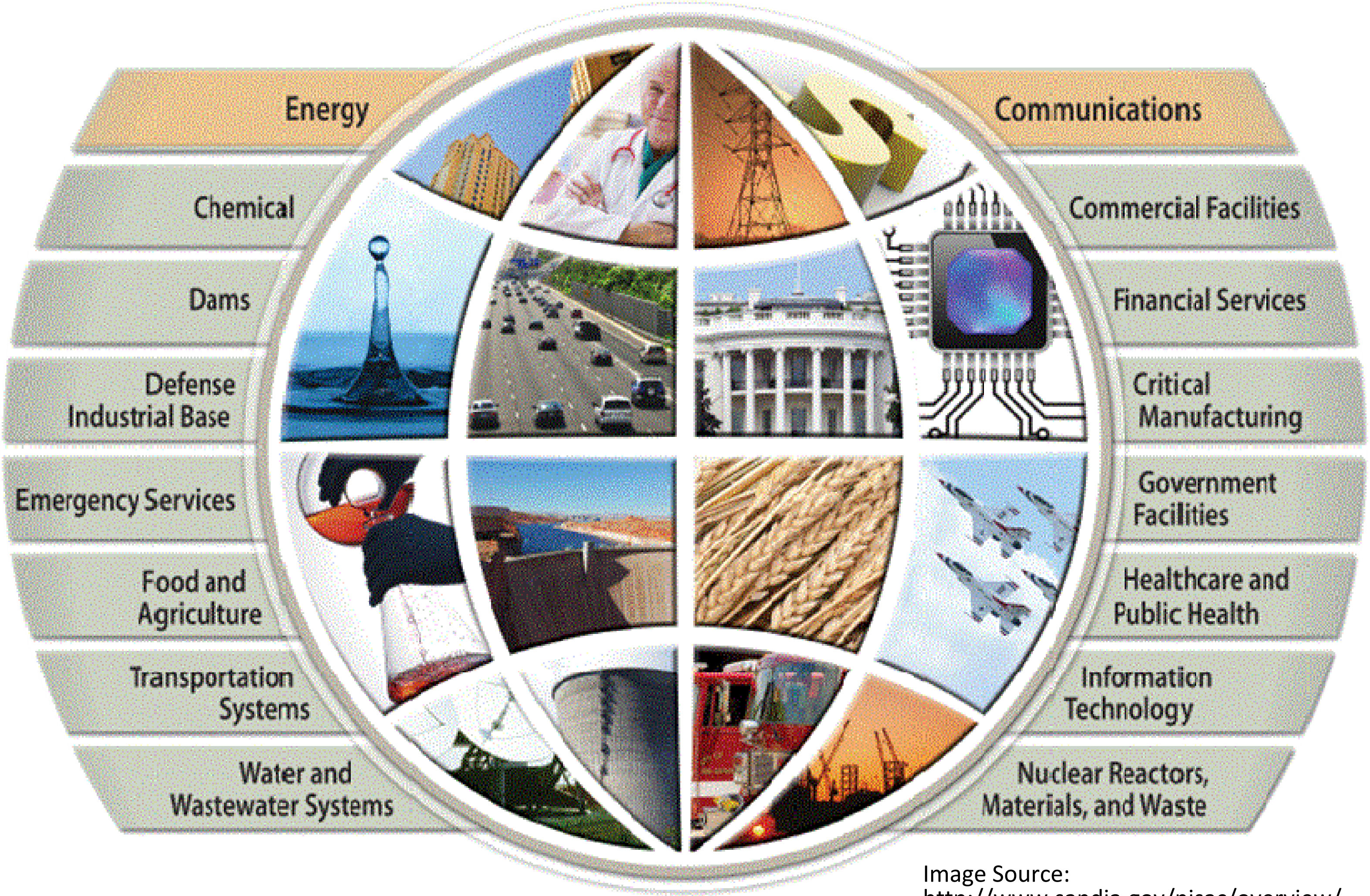
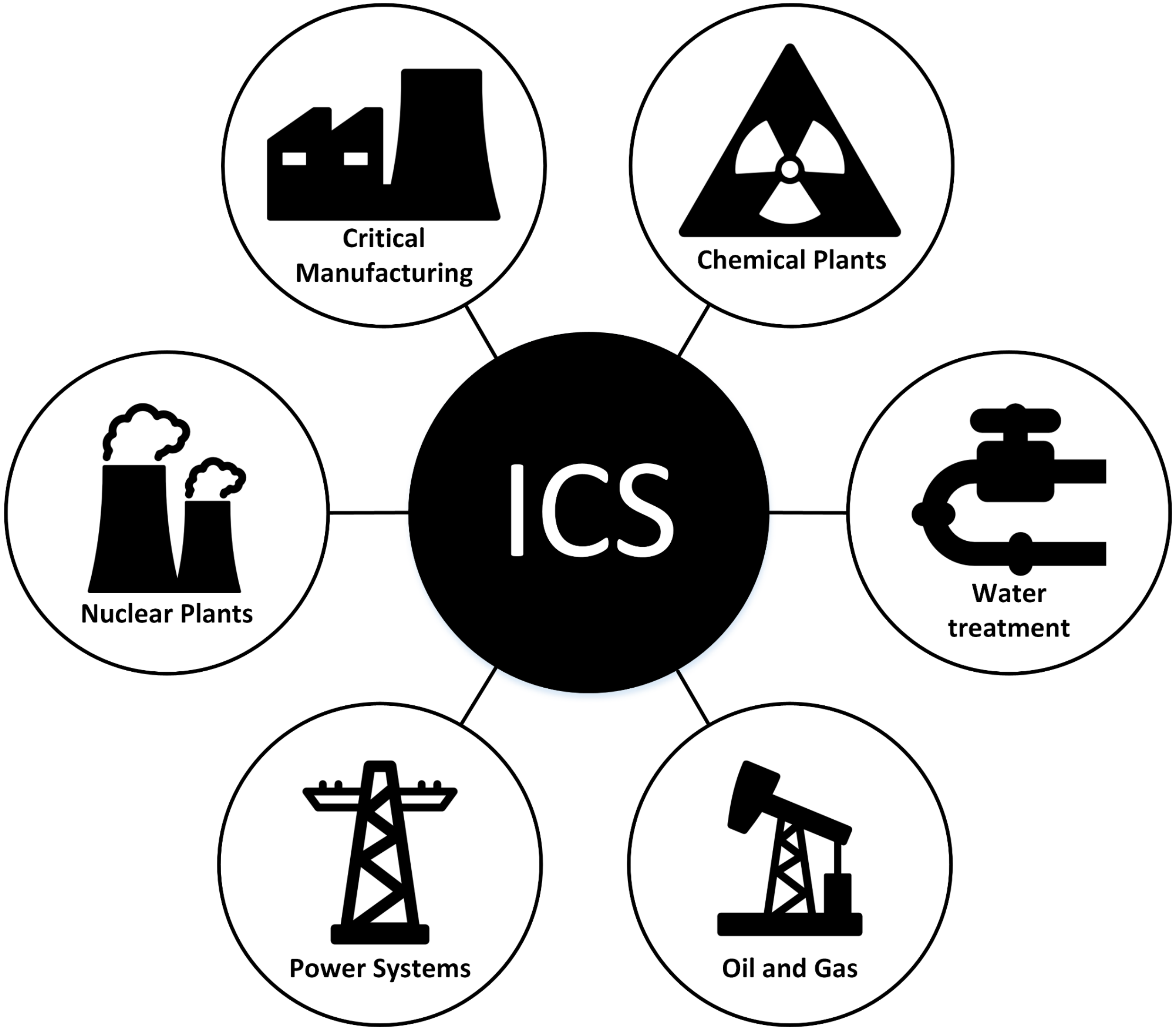
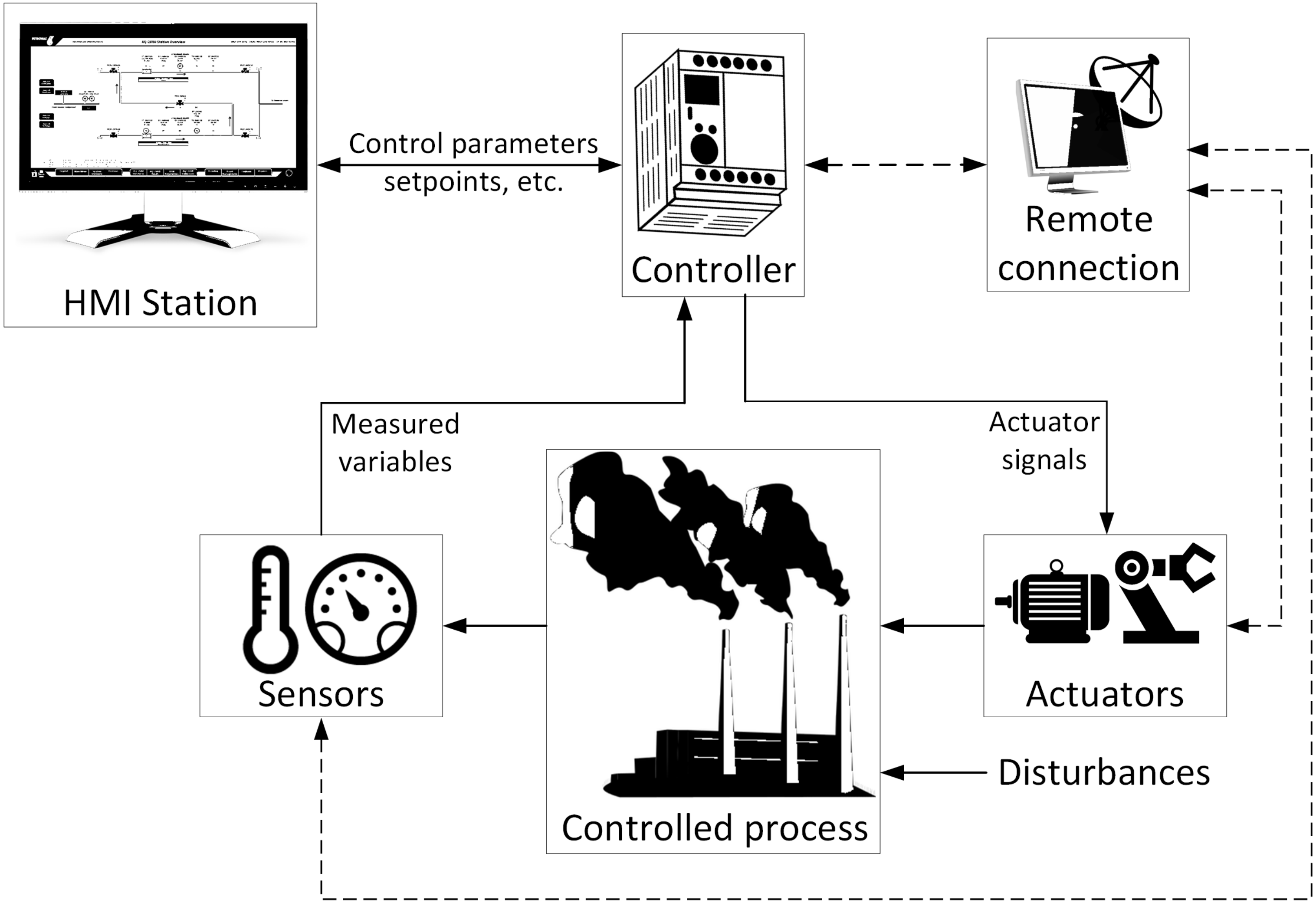


Image Source:
http://www.sandia.gov/nisac/overview/

Industrial Control Systems (ICS)

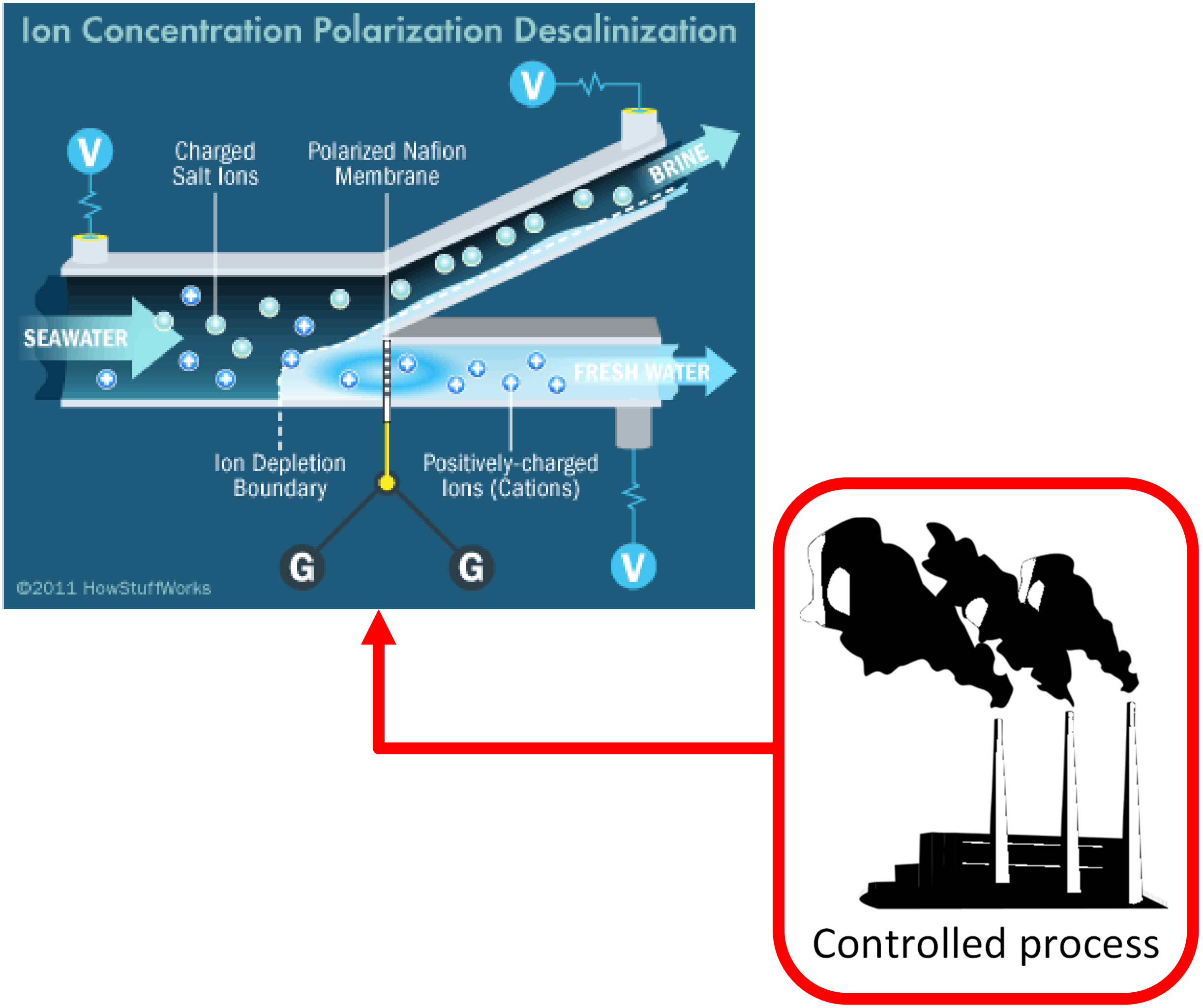


ICS architecture



Industrial Control Systems

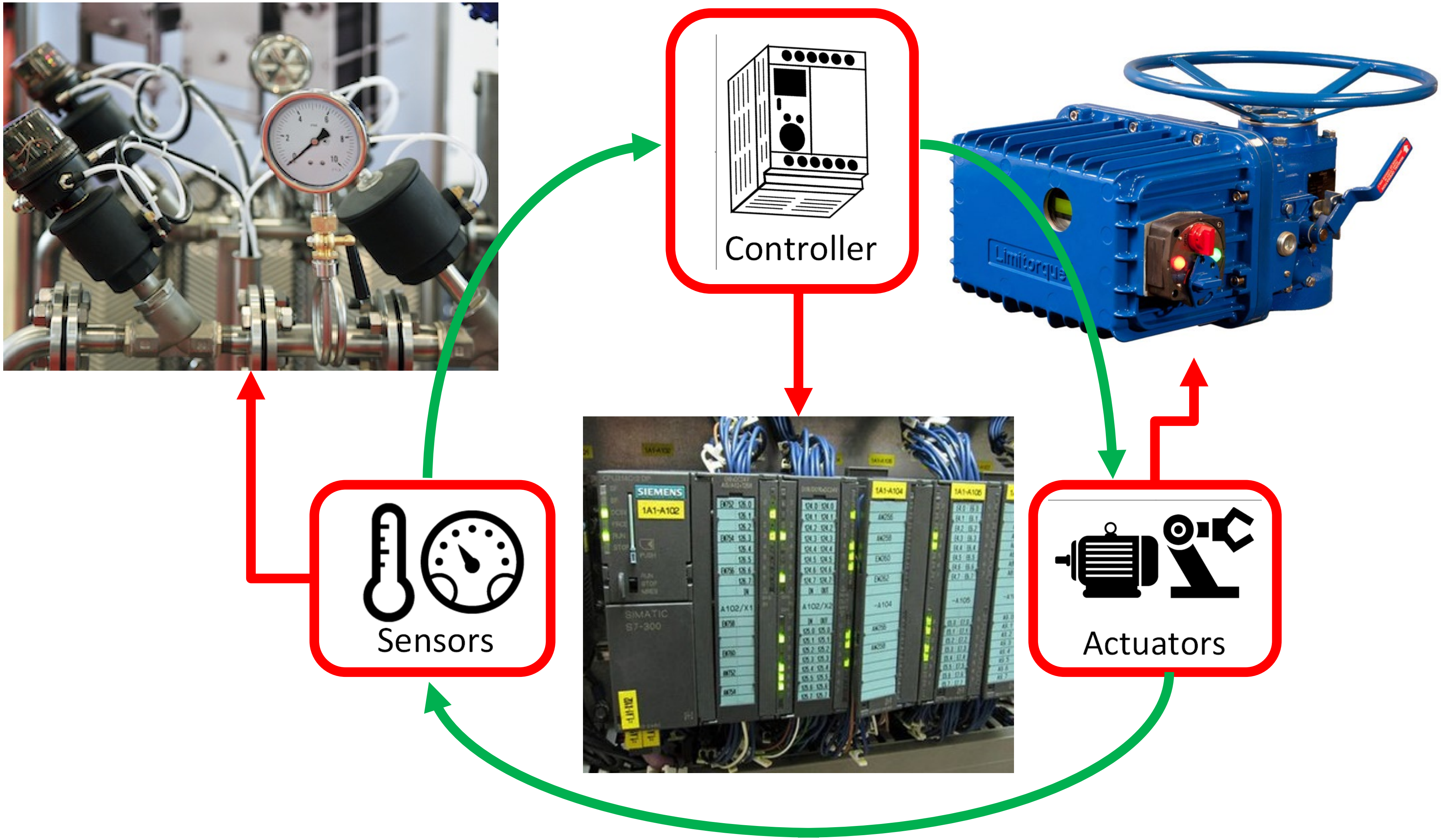
Physical process


Ion Concentration Polarization Desalinization
V
Charged Salt Ions
Polarized Nafion Membrane
BRINE
SEAWATER
FRESH WATER
Ion Depletion Boundary
Positively-charged Ions (Cations)
G
G
V
©2011 HowStuffWorks

Controlled process

Industrial Control Systems

Control loops



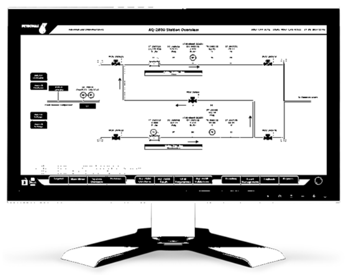
Controller

Sensors

Actuators

Industrial Control Systems

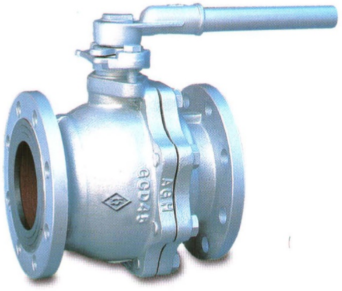
Human Machine Interface



HMI Station

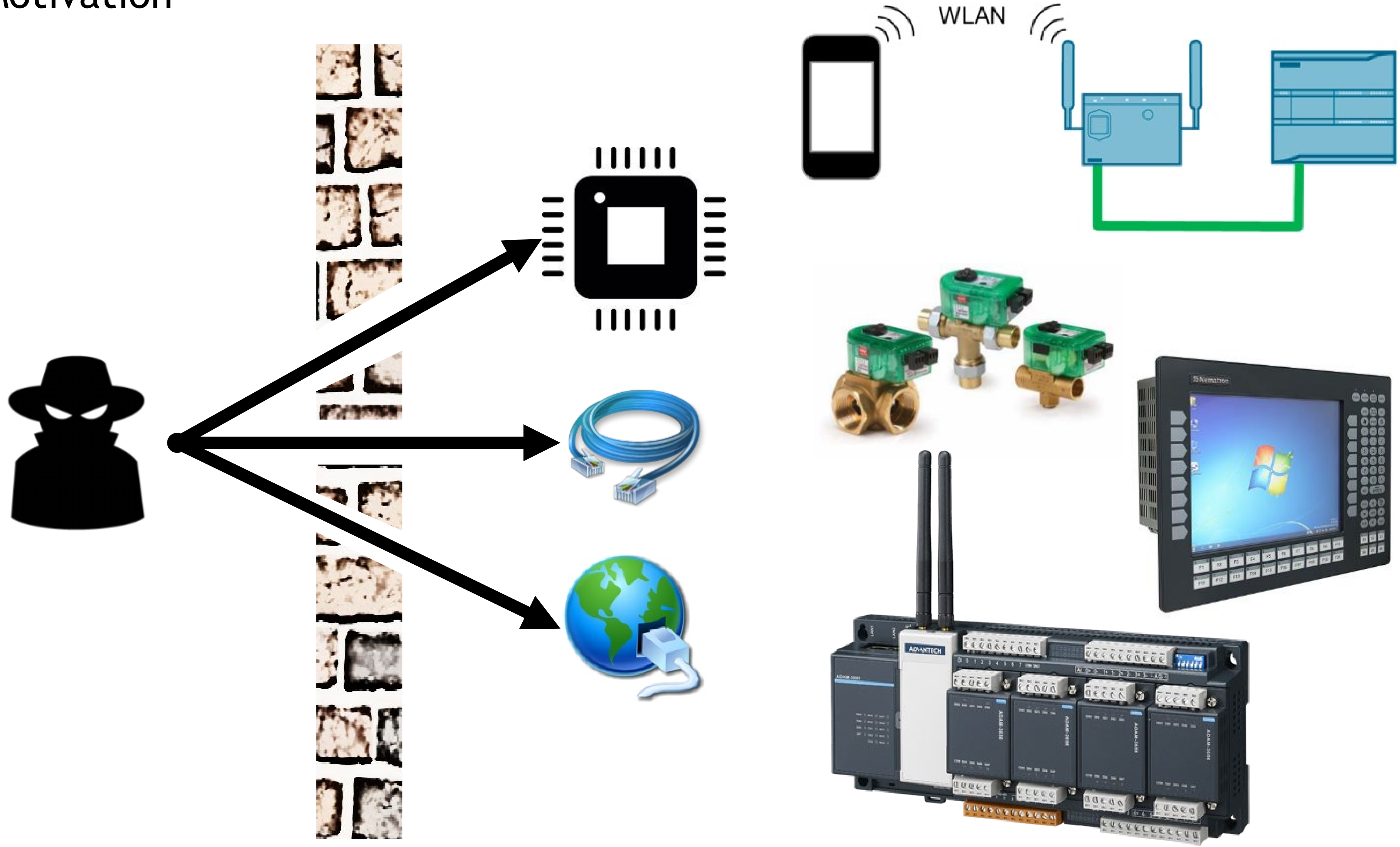
ICS threat landscape

Motivation

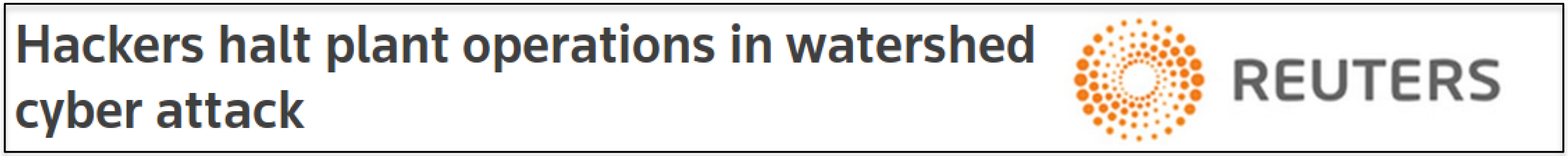


ICS threat landscape

Motivation

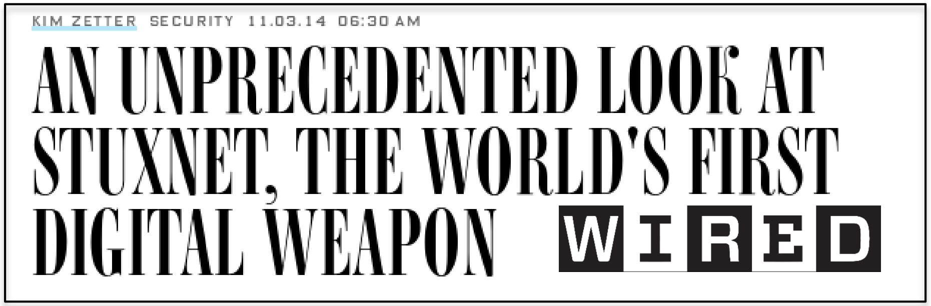


ICS cyberattacks are a reality



Hackers halt plant operations in watershed cyber attack

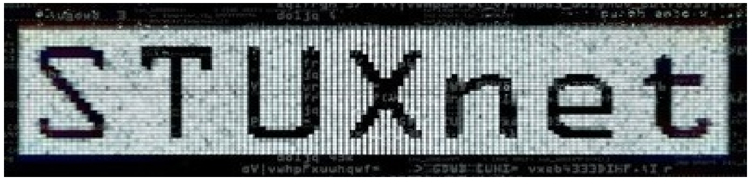
REUTERS

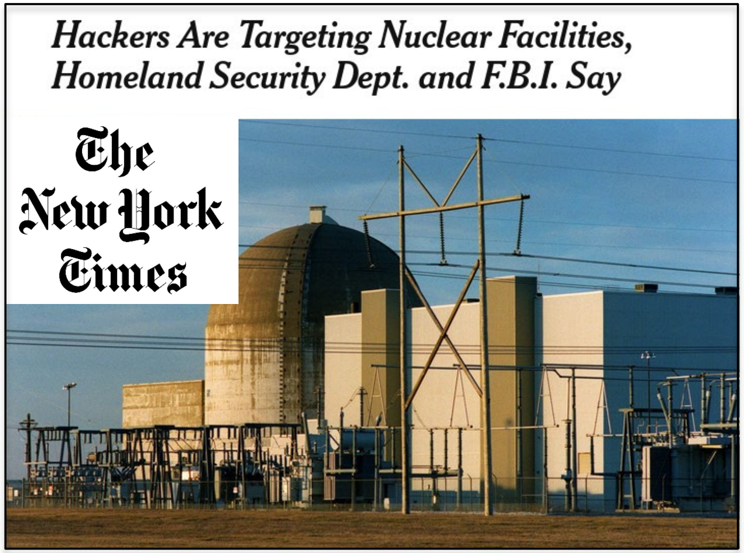


KIM ZETTER SECURITY 11.03.14 06:30 AM

AN UNPRECEDENTED LOOK AT STUXNET, THE WORLD'S FIRST DIGITAL WEAPON

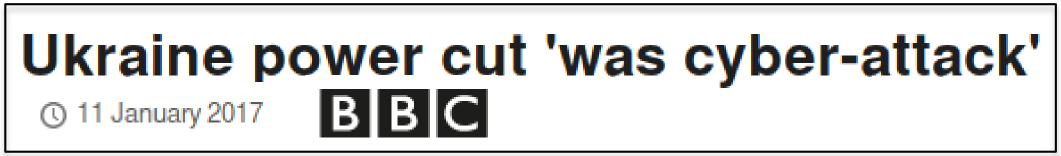
WIRED





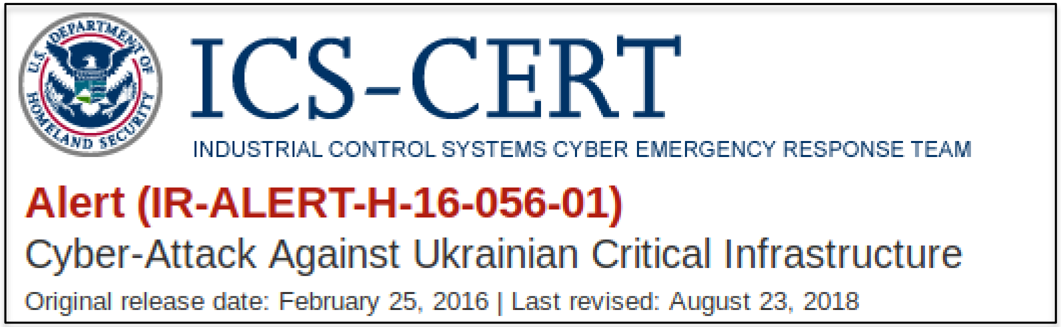
Hackers Are Targeting Nuclear Facilities, Homeland Security Dept. and F.B.I. Say

The New York Times



Ukraine power cut 'was cyber-attack'

11 January 2017 BBC



ICS-CERT

INDUSTRIAL CONTROL SYSTEMS CYBER EMERGENCY RESPONSE TEAM

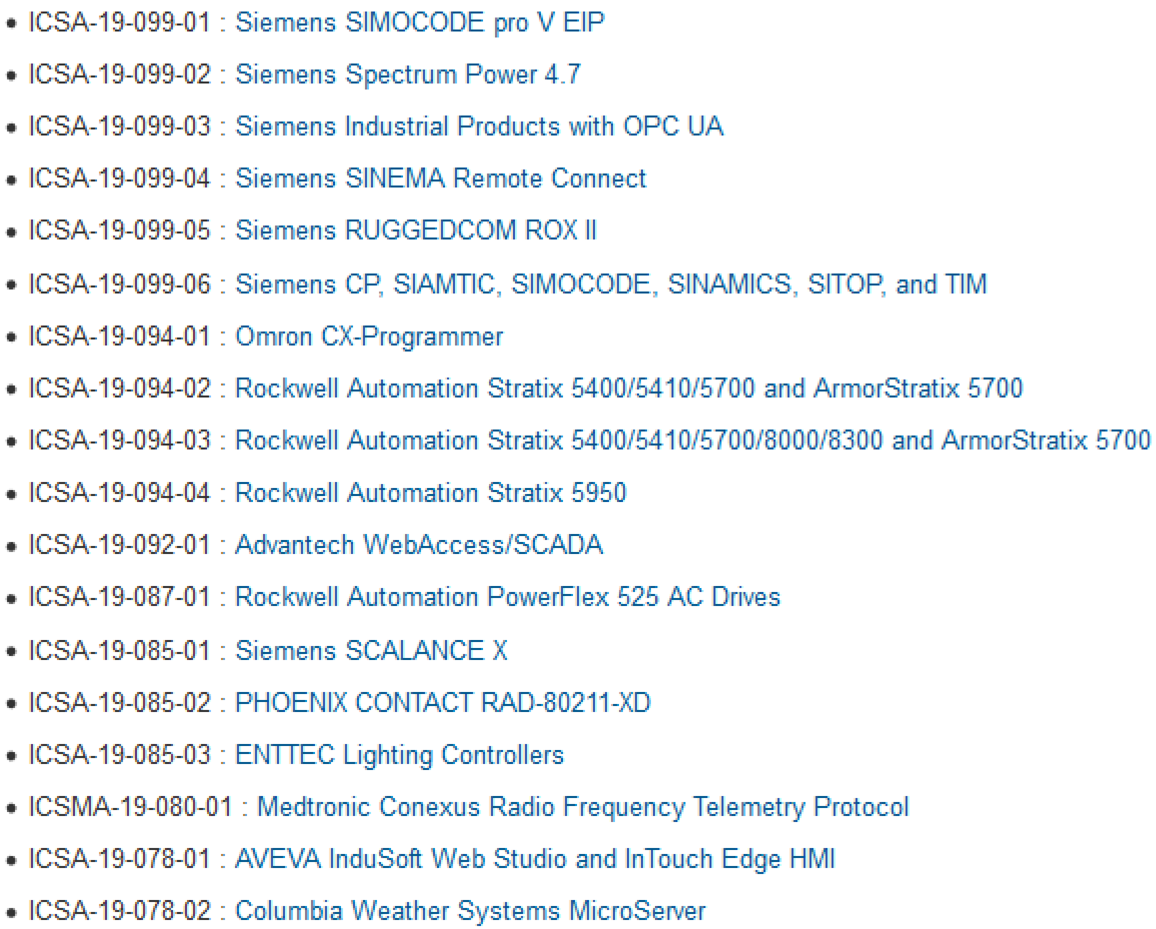
Alert (IR-ALERT-H-16-056-01)

Cyber-Attack Against Ukrainian Critical Infrastructure

Original release date: February 25, 2016 | Last revised: August 23, 2018

Is it getting worse?

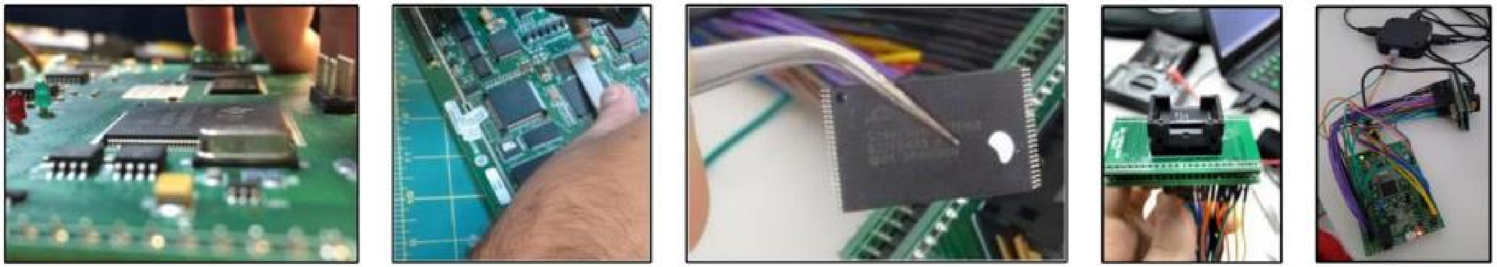
ICS-CERT advisories snapshot since 19th March 2019



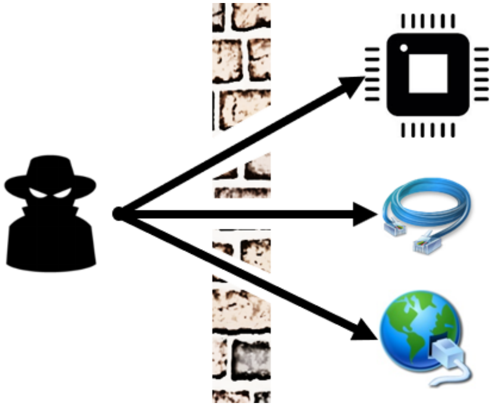
- ICSA-19-099-01 : Siemens SIMOCODE pro V EIP
- ICSA-19-099-02 : Siemens Spectrum Power 4.7
- ICSA-19-099-03 : Siemens Industrial Products with OPC UA
- ICSA-19-099-04 : Siemens SINEMA Remote Connect
- ICSA-19-099-05 : Siemens RUGGEDCOM ROX II
- ICSA-19-099-06 : Siemens CP, SIAMTIC, SIMOCODE, SINAMICS, SITOP, and TIM
- ICSA-19-094-01 : Omron CX-Programmer
- ICSA-19-094-02 : Rockwell Automation Stratix 5400/5410/5700 and ArmorStratix 5700
- ICSA-19-094-03 : Rockwell Automation Stratix 5400/5410/5700/8000/8300 and ArmorStratix 5700
- ICSA-19-094-04 : Rockwell Automation Stratix 5950
- ICSA-19-092-01 : Advantech WebAccess/SCADA
- ICSA-19-087-01 : Rockwell Automation PowerFlex 525 AC Drives
- ICSA-19-085-01 : Siemens SCALANCE X
- ICSA-19-085-02 : PHOENIX CONTACT RAD-80211-XD
- ICSA-19-085-03 : ENTTEC Lighting Controllers
- ICSMA-19-080-01 : Medtronic Conexus Radio Frequency Telemetry Protocol
- ICSA-19-078-01 : AVEVA InduSoft Web Studio and InTouch Edge HMI
- ICSA-19-078-02 : Columbia Weather Systems MicroServer

Why is it becoming worse?

- ⊙ More COTS hardware/software



- ⊙ Airgap illusion



- ⊙ Industrial protocols

Why hardware?

- Hardware is the root of trust
- Re-use existing hardware structures for intrusion detection purposes
  - Compatible with existing devices

Outline

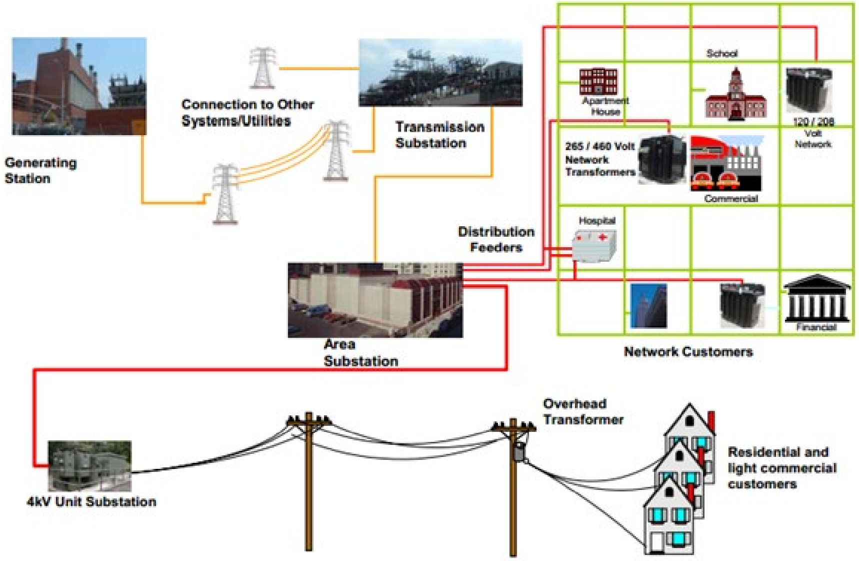
- Security for Critical Infrastructure
- **Testbed for security evaluation**
- Hardware solutions

What is a testbed?

- A collection of hardware, software, and networks enabling realistic analysis of a system’s **property** without fully replicating it
  - Example testbeds: Power flow optimization, Traffic lights control

- **Cybersecurity** testbed: A collection of hardware, software, and networks enabling realistic analysis of a system’s **cybersecurity** properties without fully replicating it

Sample Power Grid Testbed



Why cybersecurity testbed?

- Common belief: Cyber Security = Network Security
  - This is not true anymore (and maybe never was)
- We see attacks at all levels[1]
  - Control: Stuxnet, Crashoverride
  - Network: Night Dragon, Flame
  - Software: Stuxnet
  - Firmware: 2015 Ukraine Attack
  - Hardware: Side-channel attacks/leakage

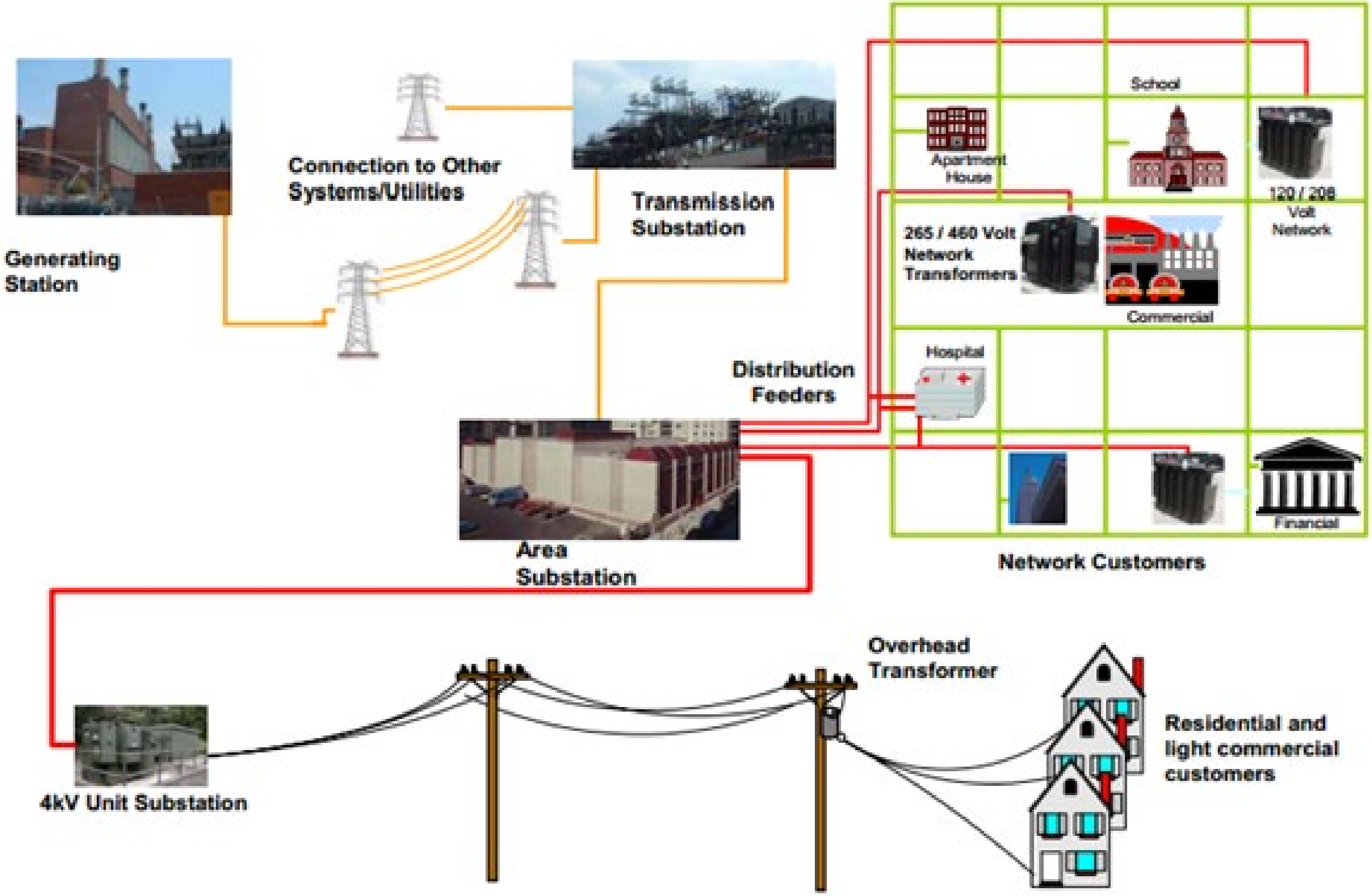
[1] **A. Keliris** et. al, "*Enabling Multi-Layer Cyber-Security Assessment of Industrial Control Systems through Hardware-in-the-Loop Testbeds*", Asia and South Pacific Design Automation Conference (ASPDAC), 2016

Why cybersecurity testbed?

- System replication prohibitive
- A testbed can be:
  - Realistic
  - Scalable (on budget),
  - Used for research, development, and training
    - R&D: New methodologies for protecting ICS
    - Training: Certification/Exposure to real-world scenarios
    - Inspire: Embedded Security Challenge

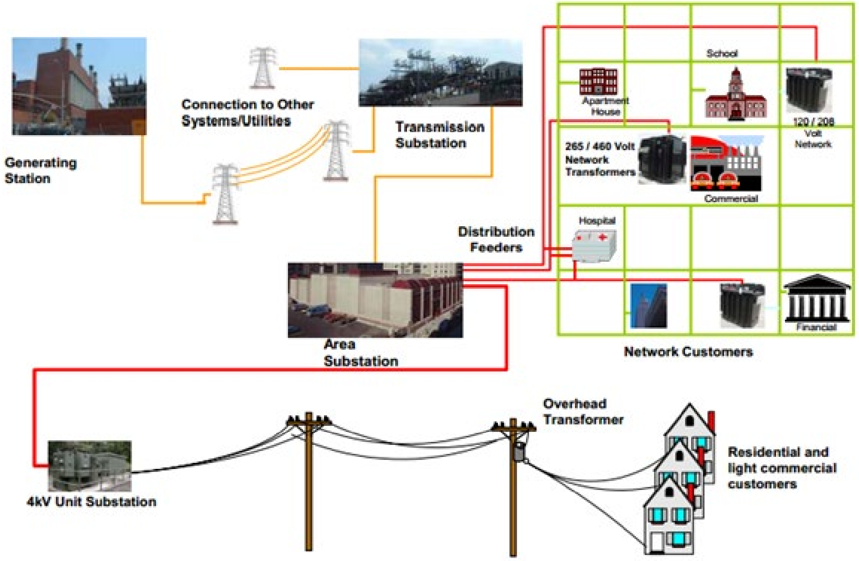
Typical power grid components

Generation, transmission, distribution, consumer



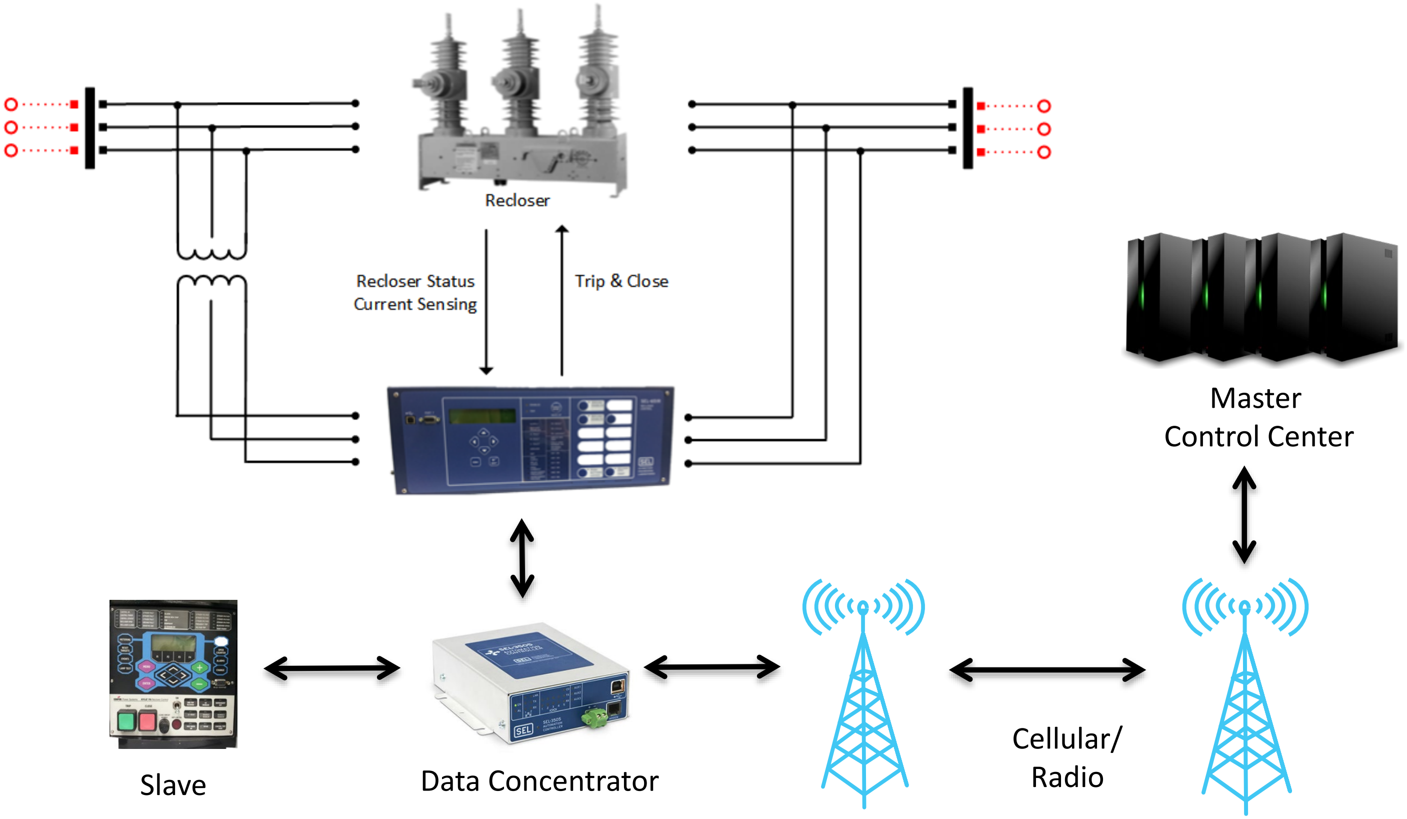
[1] Image Sources: Overview of Con Edison System and LIC Network, LIC Report, http://www.coned.com/

Step 0: Testbed creation



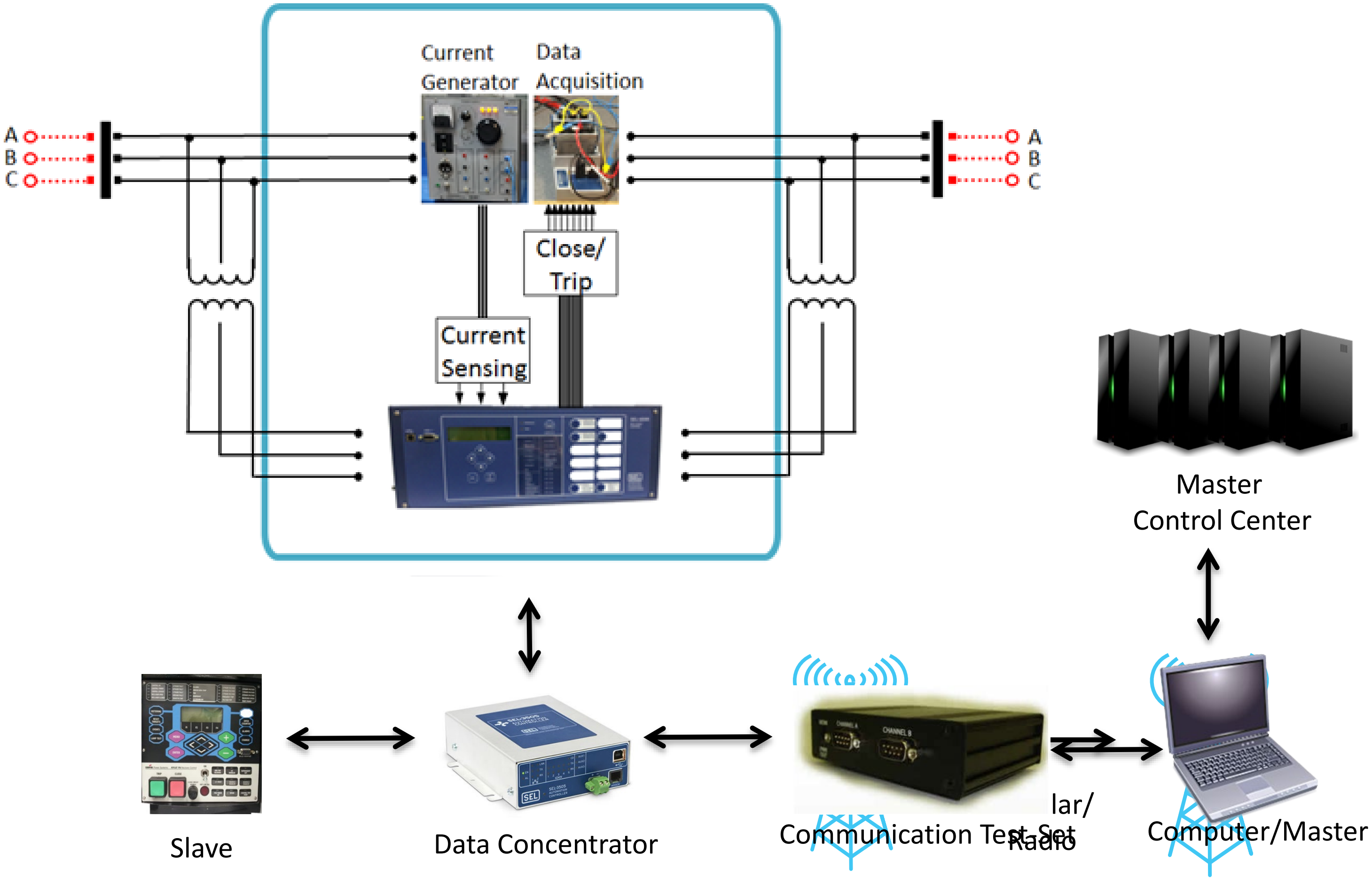
Testbed

Typical Power Grid Configuration



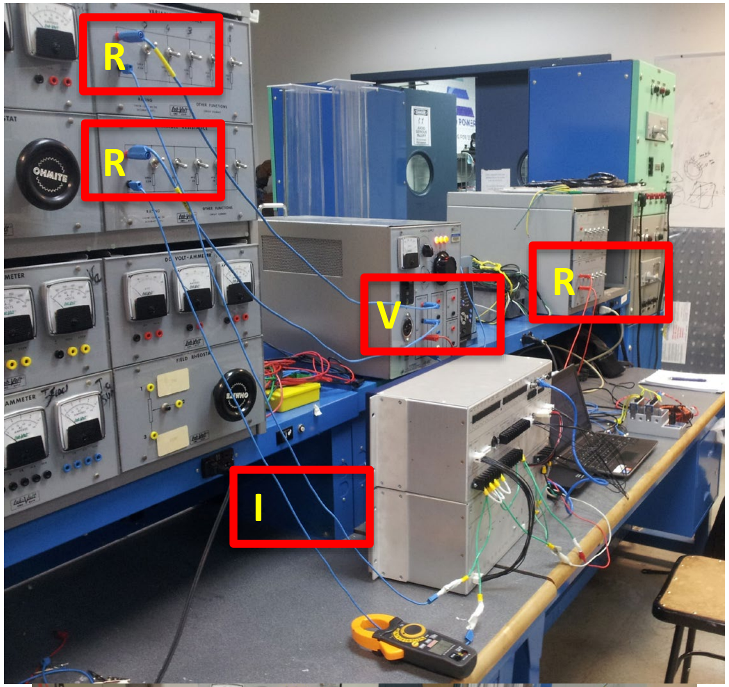
Testbed

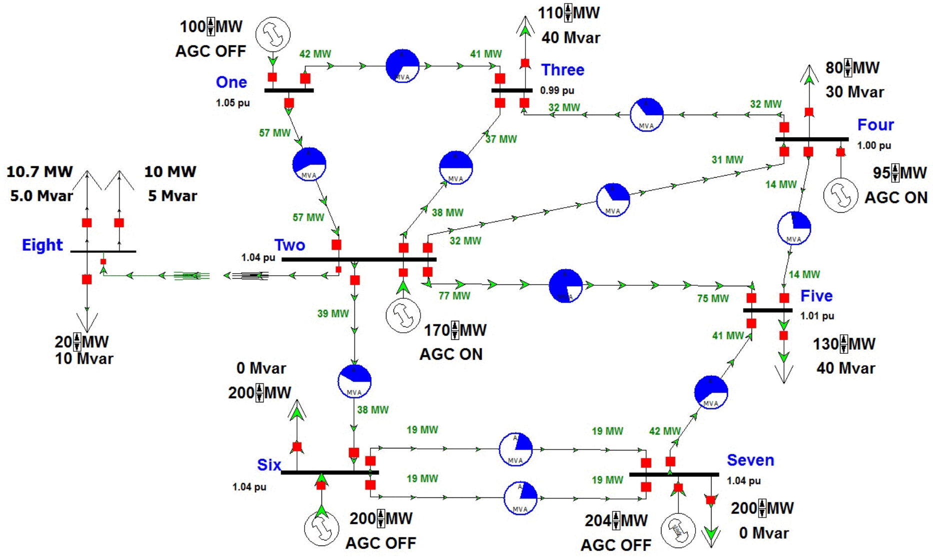
Typical Power Grid Configuration

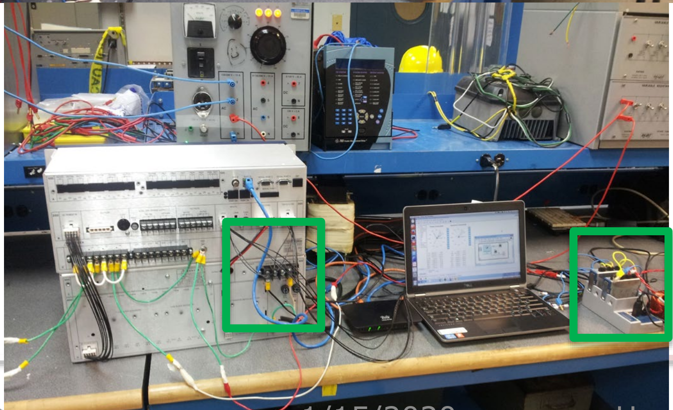


Testbed

Lab Setup: Real-time operation





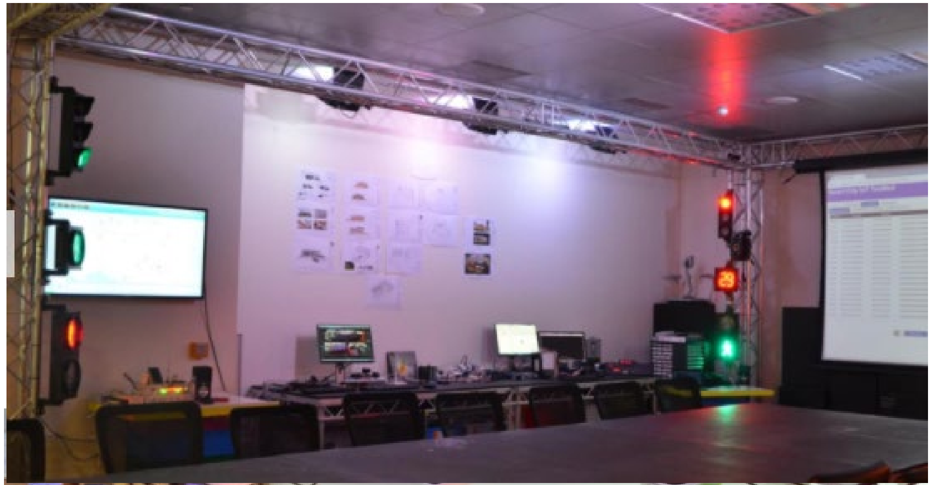


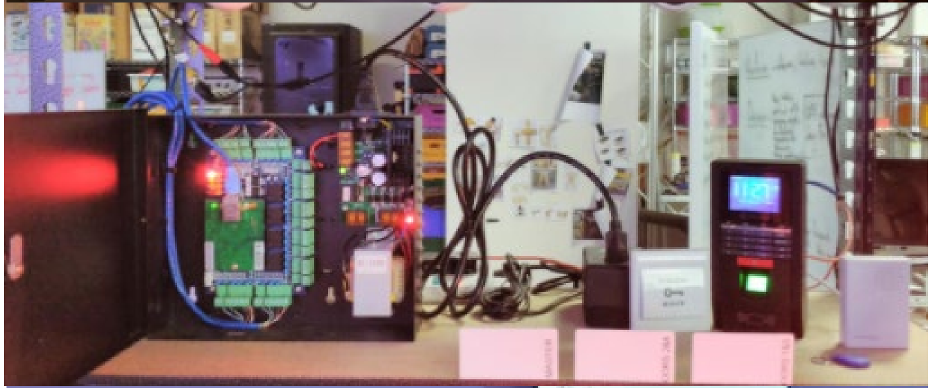
🟥 Power connections to simulate the current inputs to the devices (fine-tuned)

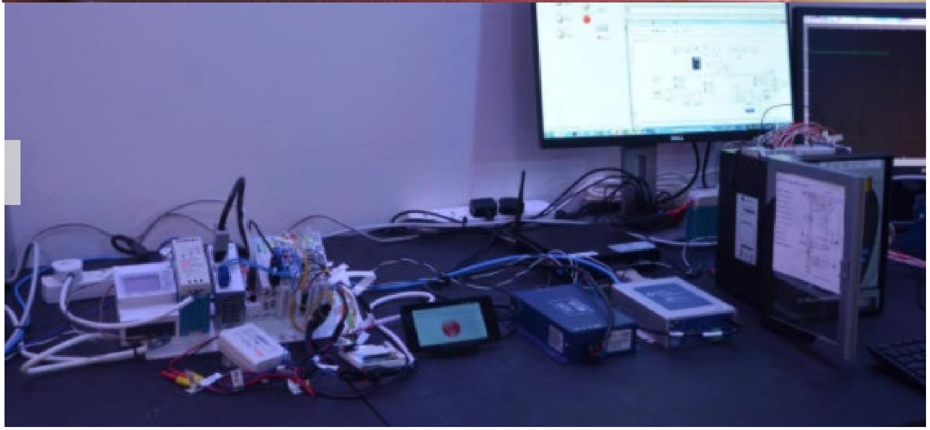
🟩 Data acquisition device connections to capture the controller output trip and close signals

NYUAD Smart-city testbed

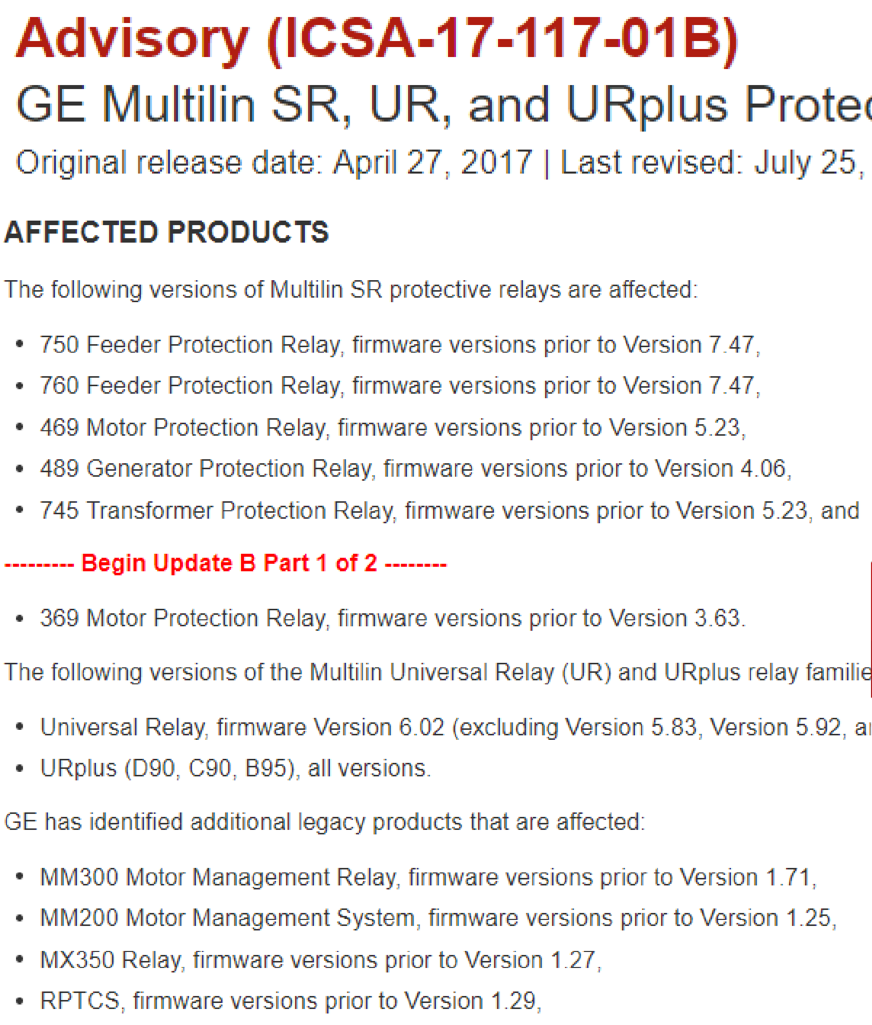
- Connecting various smart- processes
  - Smart-grid
  - Industrial IoT
    - Chemical plant
    - Desalination
  - Intelligent transportation
  - Smart house
  - Smart building
- “Come-and-hack” environment
- http://sites.nyuad.nyu.edu/ccs-ad/smart-city-testbed/







CVE-2017-7905



Advisory (ICSA-17-117-01B)

GE Multilin SR, UR, and URplus Protective Relays (Update B)

Original release date: April 27, 2017 | Last revised: July 25, 2017

AFFECTED PRODUCTS

The following versions of Multilin SR protective relays are affected:

- 750 Feeder Protection Relay, firmware versions prior to Version 7.47,
- 760 Feeder Protection Relay, firmware versions prior to Version 7.47,
- 469 Motor Protection Relay, firmware versions prior to Version 5.23,
- 489 Generator Protection Relay, firmware versions prior to Version 4.06,
- 745 Transformer Protection Relay, firmware versions prior to Version 5.23, and

--------- Begin Update B Part 1 of 2 --------

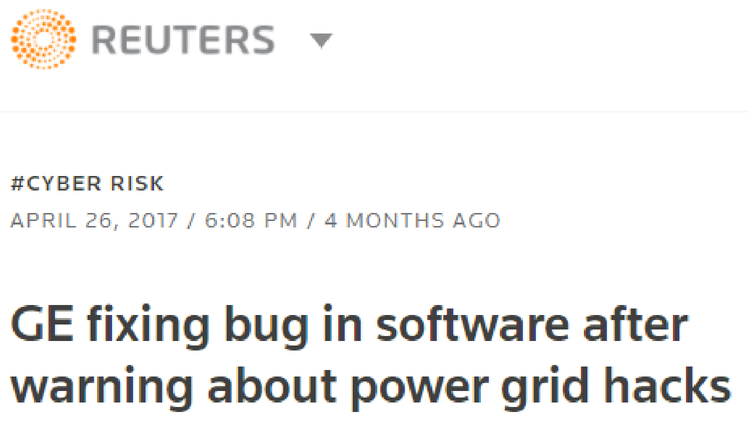
- 369 Motor Protection Relay, firmware versions prior to Version 3.63.

The following versions of the Multilin Universal Relay (UR) and URplus relay familie

- Universal Relay, firmware Version 6.02 (excluding Version 5.83, Version 5.92, a
- URplus (D90, C90, B95), all versions.

GE has identified additional legacy products that are affected:

- MM300 Motor Management Relay, firmware versions prior to Version 1.71,
- MM200 Motor Management System, firmware versions prior to Version 1.25,
- MX350 Relay, firmware versions prior to Version 1.27,
- RPTCS, firmware versions prior to Version 1.29,
- 350 Feeder Protection Relay, firmware v
- 345 Transformer Protection Relay, firmw
- 339 Motor Protection Relay, firmware ve
- T1000 Switch, firmware versions prior t

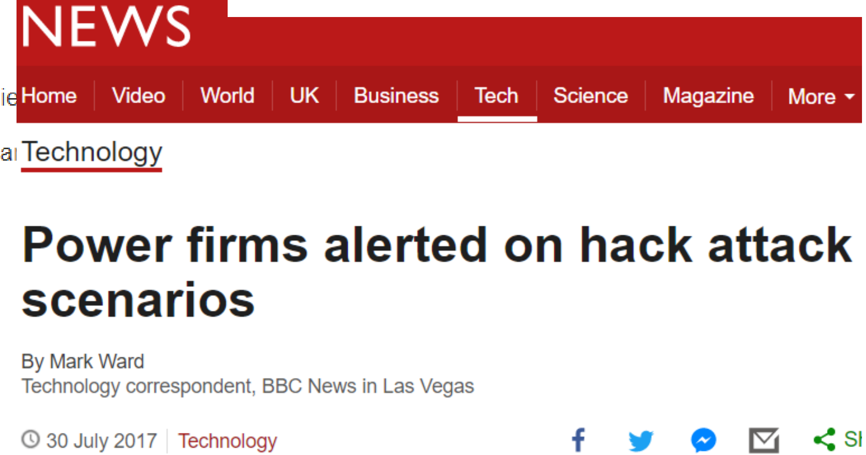


REUTERS

#CYBER RISK

APRIL 26, 2017 / 6:08 PM / 4 MONTHS AGO

GE fixing bug in software after warning about power grid hacks



Technology

Power firms alerted on hack attack scenarios

By Mark Ward
Technology correspondent, BBC News in Las Vegas

30 July 2017 | Technology

https://www.reuters.com/article/us-cyber-generalelectric-power-idUSKBN17S23Y
https://www.youtube.com/watch?v=A58DPrdSllM
https://it.slashdot.org/story/17/04/26/1839218/ge-fixing-bug-in-software-after-warning-about-power-grid-hacks
https://www.usnews.com/news/technology/articles/2017-04-26/ge-fixes-bug-in-power-software-as-researchers-warn-o
https://www.theregister.co.uk/2017/04/27/ge_rushing_patches_to_grid_systems_ahead_of_black_hat_demonstration
https://www.reddit.com/r/energy/comments/67qks9/ge_fixing_bug_in_software_after_warning_about/
https://uk.finance.yahoo.com/quote/GE?p=GE
http://www.bbc.com/news/technology-40766757
https://nakedsecurity.sophos.com/2017/05/02/ge-patches-flaws-allowing-attackers-to-disconnect-power-grid-at-will/
http://gulftoday.ae/portal/ae098790-8b50-43ef-a70b-b2c584954606.aspx
https://www.helpnetsecurity.com/2017/07/28/power-grid-cyberattacks/
https://www.eenews.net/energywire/2017/07/28/stories/1060058065
http://www.engerati.com/article/smart-grid-security-vulnerabilities-and-how-deal-them

Testbed in place! Now what?

- Hardware solutions can be explored
  - Anomaly detection using hardware performance counters
    - Funded by Consolidated Edison
  - Anomaly detection using external monitors
    - Funded by DARPA
  - Automated reverse engineering of Industrial Control Systems binaries
    - Funded by ONR

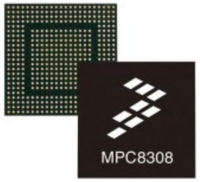
Testbed in place! Now what?

- Hardware solutions can be explored
  - Anomaly detection using hardware performance counters

- Research question:
  <u>Can we improve the security posture of legacy devices?</u>

Hardware Performance Counters

- A set of special-purpose registers that count low-level hardware events
  - Primarily targeting performance tuning
  - We repurpose them for security
- Included in some existing grid devices



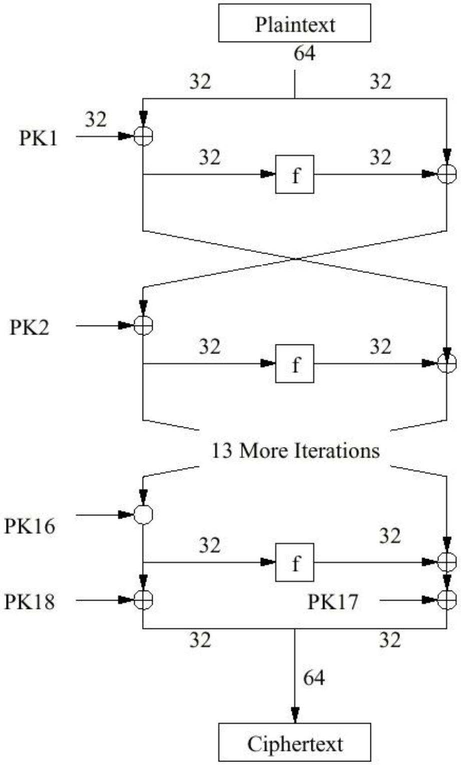
MPU POWERQUICC II PRO,
containing the e300c3 processor core

Name	Description
CPU_CLK	Cycles
COMPLETED_INSNS	Completed Instructions (0, 1, or 2 per cycle)
INSTRUCTION_FETCHES	Instruction fetches
PM_EVENT_TRANS	0 to 1 translations on the pm_event input
PM_EVENT_CYCLES	processor bus cycle
COMPLETED_BRANCHES	Branch Instructions completed
COMPLETED_LOAD_OPS	Load micro-ops completed
COMPLETED_STORE_OPS	Store micro-ops completed
BRANCHES_FINISHED	Branches finished
TAKEN_BRANCHES_FINISHED	Taken branches finished
BRANCHES_MISPREDICTED	Branch instructions mispredicted due to direction, target, or IAB prediction
DECODE_STALLED	Cycles the instruction buffer was not empty, but 0 instructions decoded
ISSUE_STALLED	Cycles the issue buffer is not empty but 0 instructions issued
CACHEINHIBITED_ACCESSES_TRANSLATED	Number of cache inhibited accesses translated
FETCHES	Counts the number of fetches that write at least one instruction to the instruction buffer

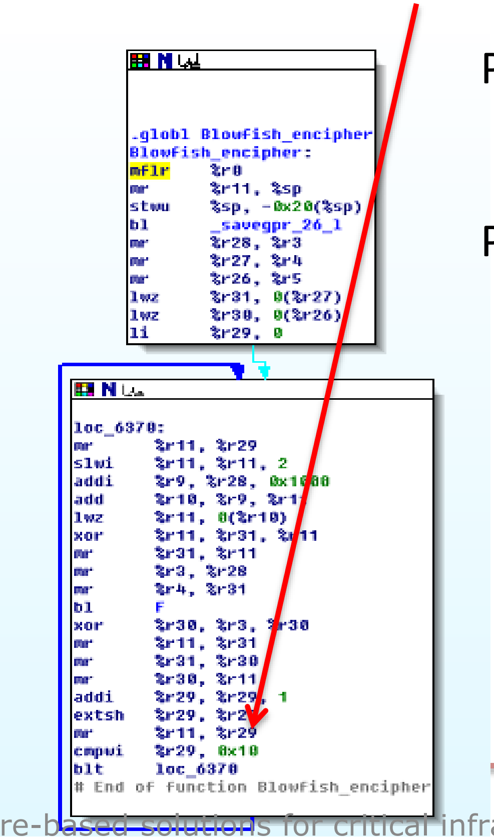
Toy example: Blowfish Cipher

Malicious actions will show up on a performance counter

- The valid execution flow runs **16** iter



- Modify ***cmpwi r29, 0x10*** to ***cmpwi r29, 0x0A*** to run less iterations



```
.globl Blowfish_encipher
Blowfish_encipher:
mflr      %r0
mr        %r11, %sp
stwu      %sp, -0x20(%sp)
bl        _savegpr_26_l
mr        %r28, %r3
mr        %r27, %r4
mr        %r26, %r5
lwz       %r31, 0(%r27)
lwz       %r30, 0(%r26)
li        %r29, 0
```

```
loc_6370:
mr        %r11, %r29
slwi      %r11, %r11, 2
addi      %r9, %r28, 0x1000
add       %r10, %r9, %r11
lwz       %r11, 0(%r10)
xor       %r11, %r31, %r11
mr        %r31, %r11
mr        %r3, %r28
mr        %r4, %r31
bl        F
xor       %r30, %r3, %r30
mr        %r11, %r31
mr        %r31, %r30
mr        %r30, %r11
addi      %r29, %r29, 1
extsh     %r29, %r29
mr        %r11, %r29
cmpwi     %r29, 0x10
blt       loc_6370
# End of function Blowfish_encipher
```

Profile of the valid path:

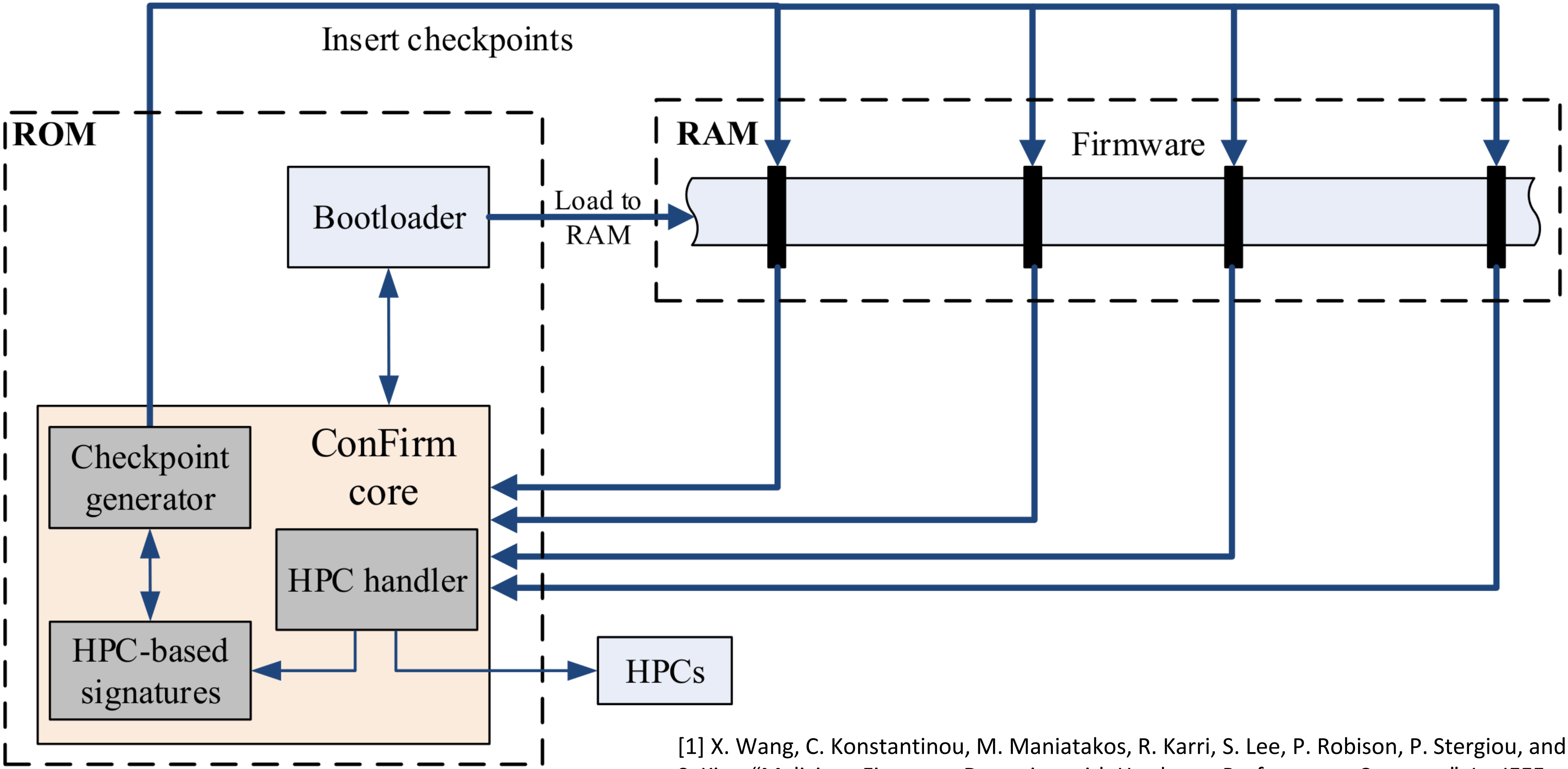
- # of instructions = 1143
- # of branches = 82

Profile of the malicious path:

- # of instructions = 723
- # of branches = 52

ConFirm [1]

Anomaly detection using HPCs

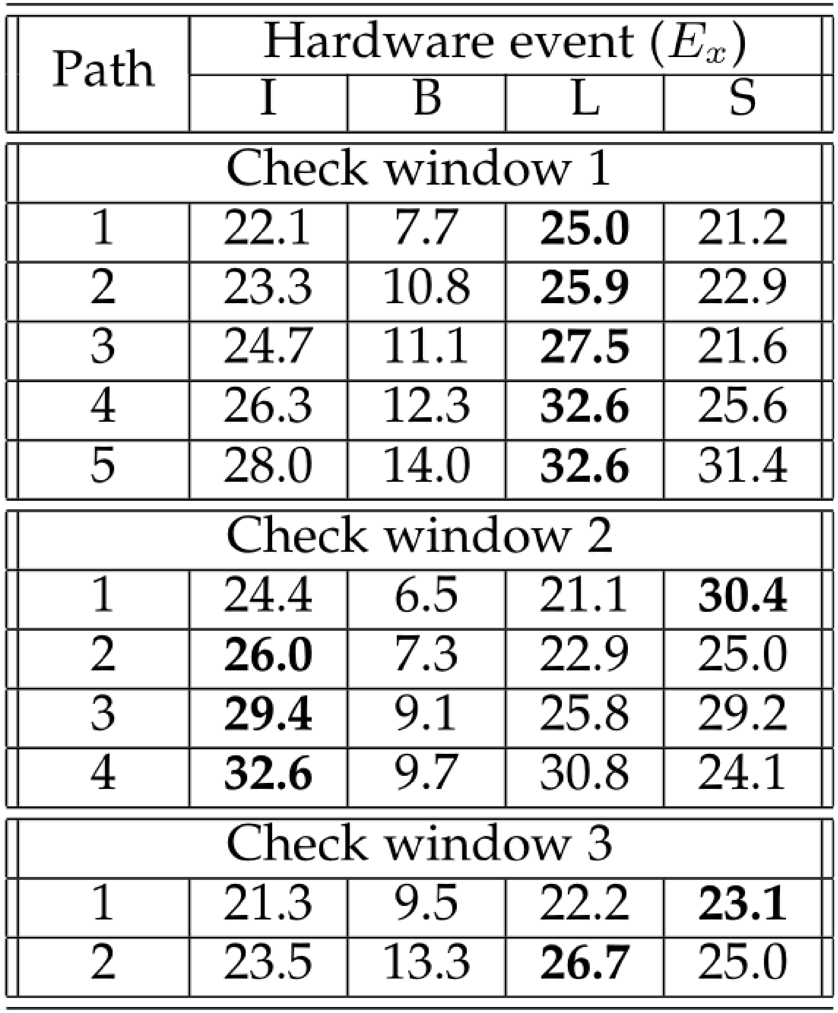


[1] X. Wang, C. Konstantinou, M. Maniatakos, R. Karri, S. Lee, P. Robison, P. Stergiou, and S. Kim. “Malicious Firmware Detection with Hardware Performance Counters”. In: IEEE Transactions on Multi-Scale Computing Systems 2.3 (2016), pp. 160–173

Case study: Attack detection

Man-In-The-Middle attack on PowerPC

- Simple thresholding
  - Instruction count, Branches taken, Load instructions, Store Instructions
- 100% detection when setting noise threshold > 5%
  - Accurate for superloop-type firmware



Path	Hardware event (E_x)			
	I	B	L	S
Check window 1				
1	22.1	7.7	**25.0**	21.2
2	23.3	10.8	**25.9**	22.9
3	24.7	11.1	**27.5**	21.6
4	26.3	12.3	**32.6**	25.6
5	28.0	14.0	**32.6**	31.4
Check window 2				
1	24.4	6.5	21.1	**30.4**
2	**26.0**	7.3	22.9	25.0
3	**29.4**	9.1	25.8	29.2
4	**32.6**	9.7	30.8	24.1
Check window 3				
1	21.3	9.5	22.2	**23.1**
2	23.5	13.3	**26.7**	25.0

Testbed in place! Now what?

- Hardware solutions can be explored
  - Anomaly detection using hardware performance counters
    - Funded by Consolidated Edison
  - Anomaly detection using external monitors
    - Funded by DARPA
  - Automated reverse engineering of Industrial Control Systems binaries
    - Funded by ONR

RADICS

- DARPA $77M program on protecting United States’ power grid
  - NYU participates in a team led by SRI
- Assumes a doomsday scenario
- Research question:
  Can we detect whether an attacker could still be in the system without prior instrumentation?

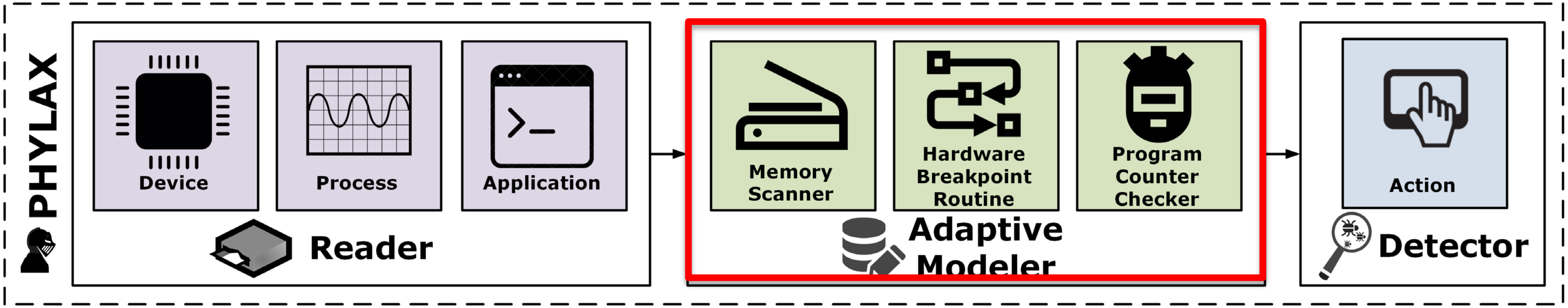
Leverage Hardware

Defenses: JTAG

- Detect intrusions in already installed real-time embedded devices via JTAG
  - JTAG: IEEE Std. 1149.1, used for boundary scan testing, storing firmware – programming modules, debugging embedded systems
- External monitoring tool
  - No code instrumentation
  - Adapt and prioritize based on:
    - Real-time requirements of the critical infrastructure process
    - Computing capabilities of the embedded system
  - Does not require any form of vendor collaboration

PHYLAX Architecture[1]

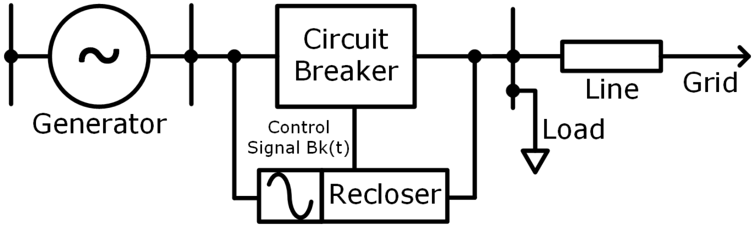
Defenses: JTAG



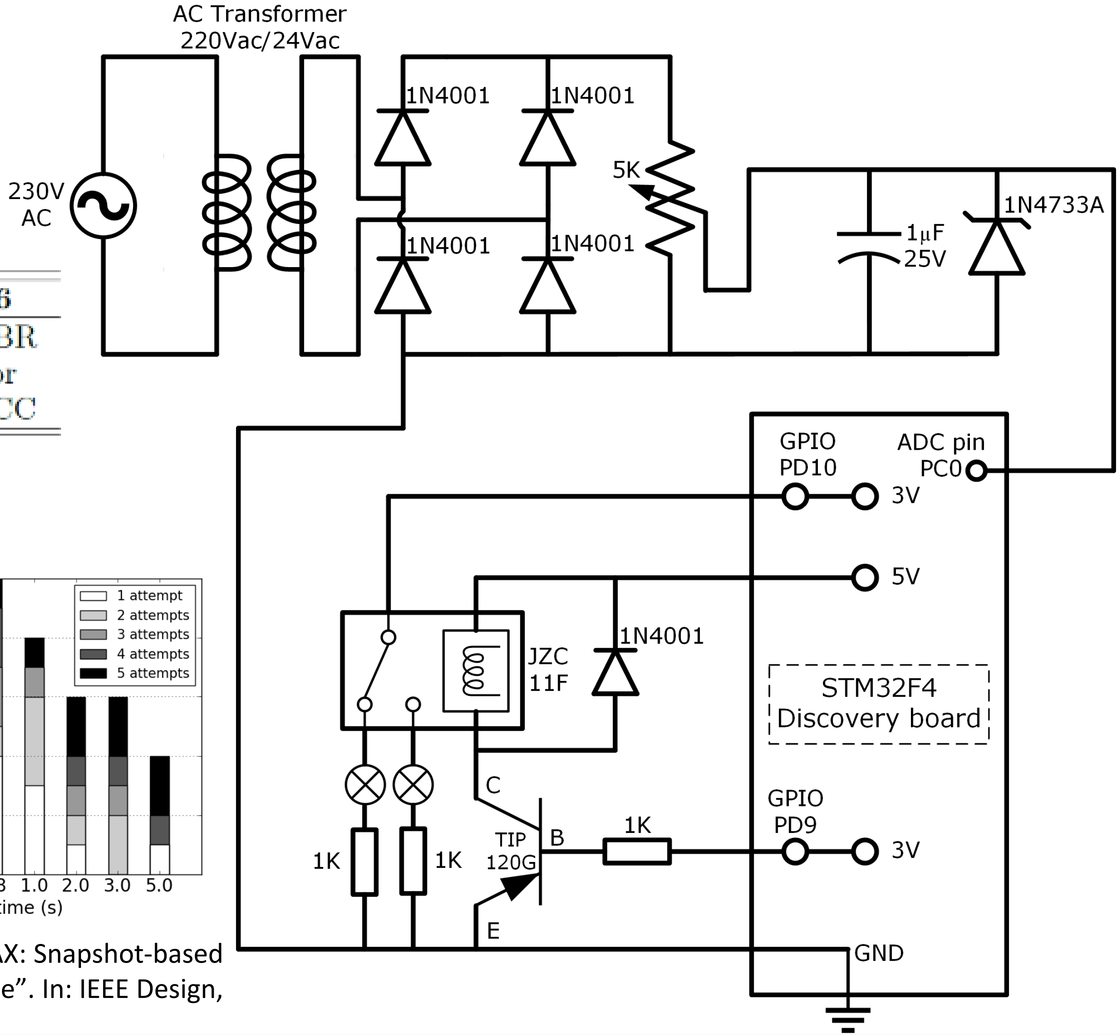
- Memory Scanner (MS)
  - Continuously extracts content from the device and inspects the run-time memory data
- Hardware Breakpoint Routine (HBR)
  - Triggered when the scanner identifies memory (e.g. stack) content that matches instructions
- Program Counter Checker (PCC)
  - Check execution area

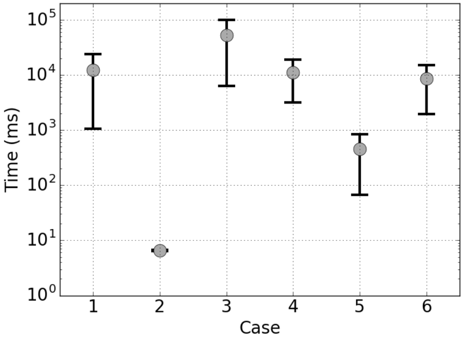
[1] C. Konstantinou, E. Chielle, and M. Maniatakos. "PHYLAX: Snapshot-based Profiling of Real-Time Embedded Devices via JTAG Interface". In: IEEE Design, Automation and Test in Europe (DATE). 2018, pp. 869?872

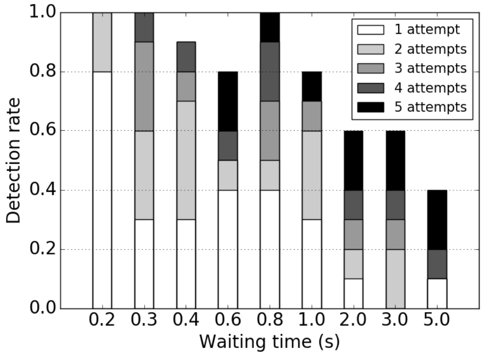
Case Study: Power Grid Monitor



Case	1	2	3	4	5	6
Detect by	MS	MS	HBR or PCC	MS	MS	HBR or PCC







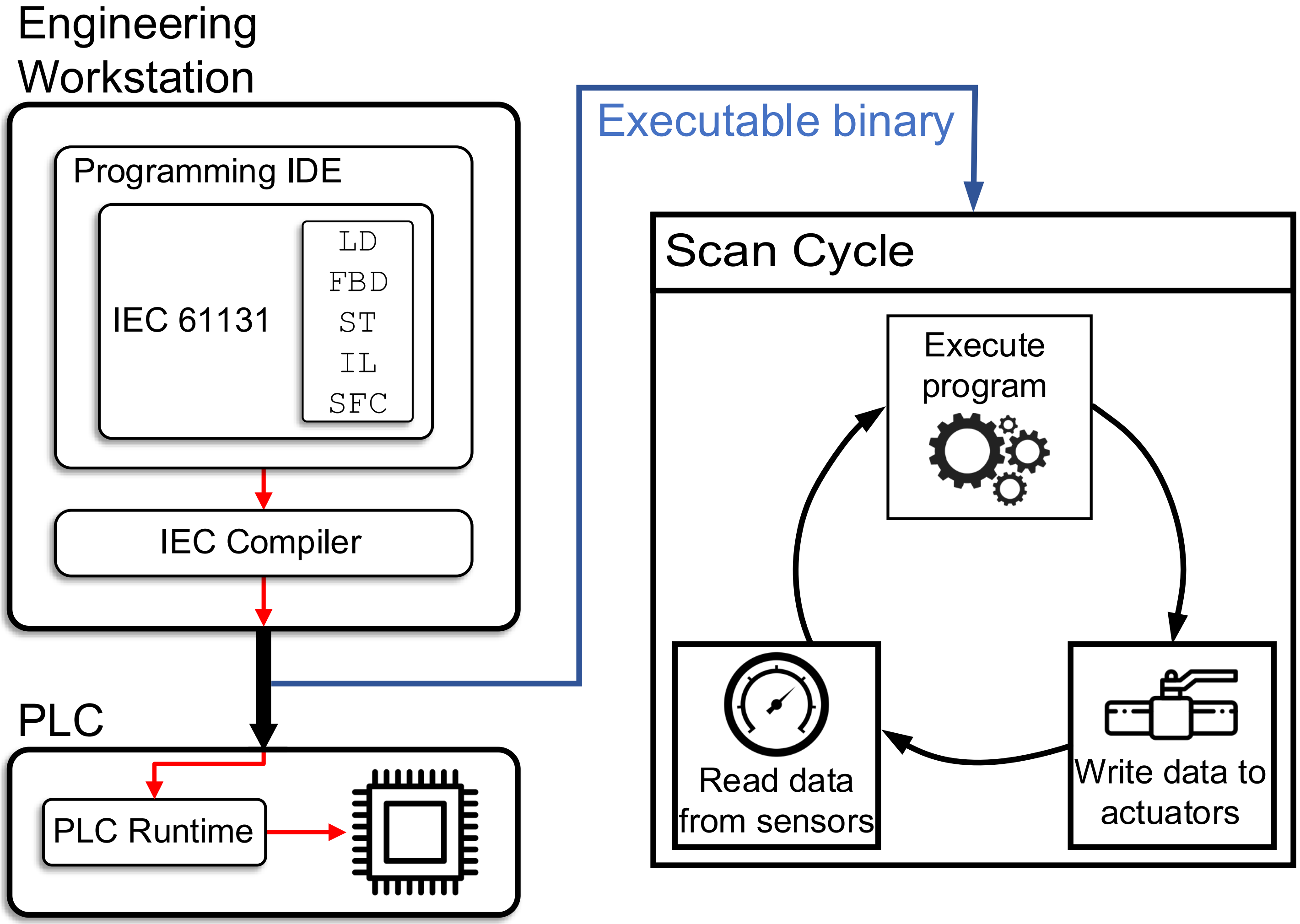
[1] C. Konstantinou, E. Chielle, and M. Maniatakos. "PHYLAX: Snapshot-based Profiling of Real-Time Embedded Devices via JTAG Interface". In: IEEE Design, Automation and Test in Europe (DATE). 2018, pp. 869-872

Testbed in place! Now what?

- Hardware solutions can be explored
  - Anomaly detection using hardware performance counters
    - Funded by Consolidated Edison
  - Anomaly detection using external monitors
    - Funded by DARPA
  - Automated reverse engineering of Industrial Control Systems binaries
    - Funded by ONR

PLC operation

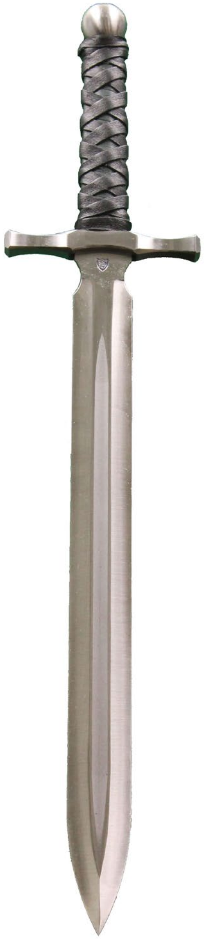
Payload design



Why reverse engineer ICS binaries?

- Analyze PLC malware
- Recover lost source code

- Dynamic payload generation
- No need for C2 server (air-gap)



Why are ICS binaries “special”?

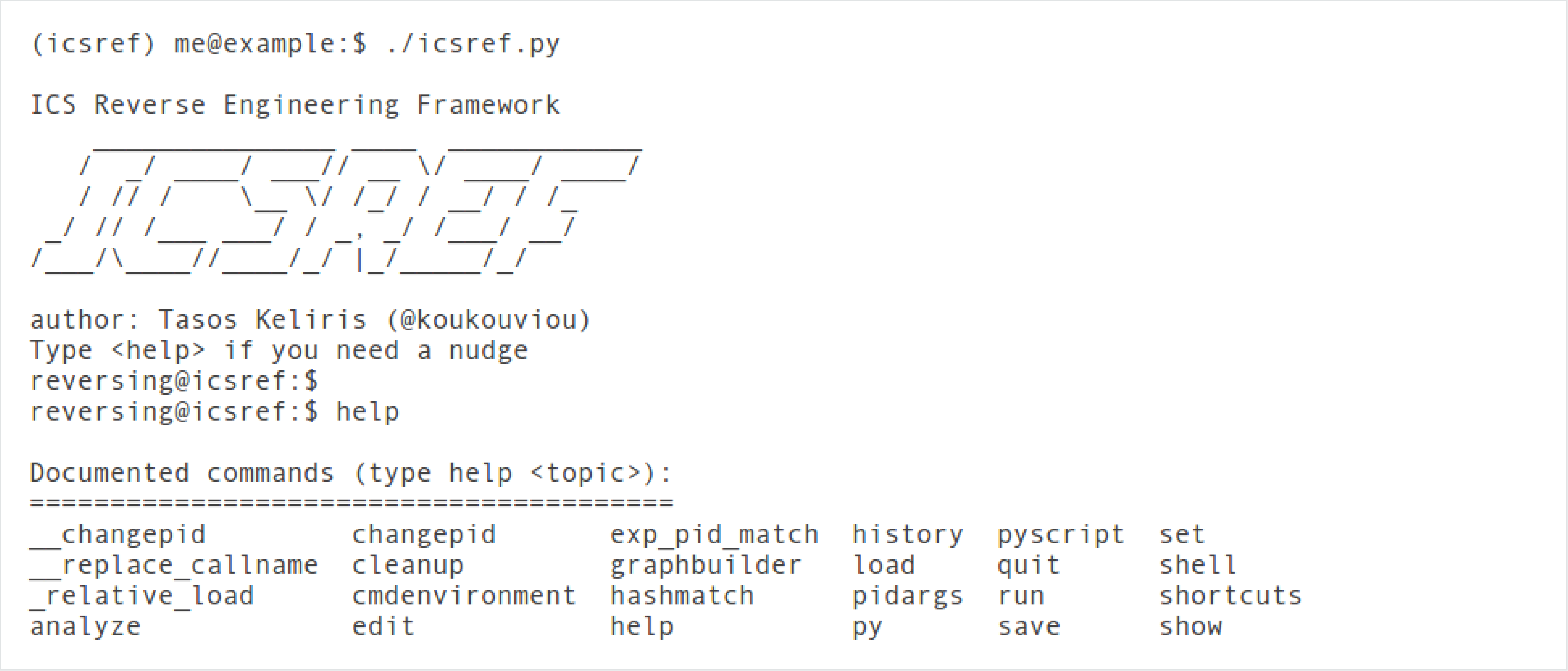
- Execution model
  - Scan cycle
- I/O operations
  - How and where are I/O operations?
- File formats
  - Custom & Proprietary
- Optimizations
  - Or lack thereof …

ICS RevEng Framework[1]

ICSREF: github.com/momalab/ICSREF

Payload design

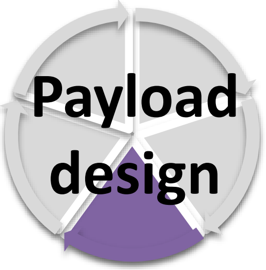
⊙ Methodology and Modular framework

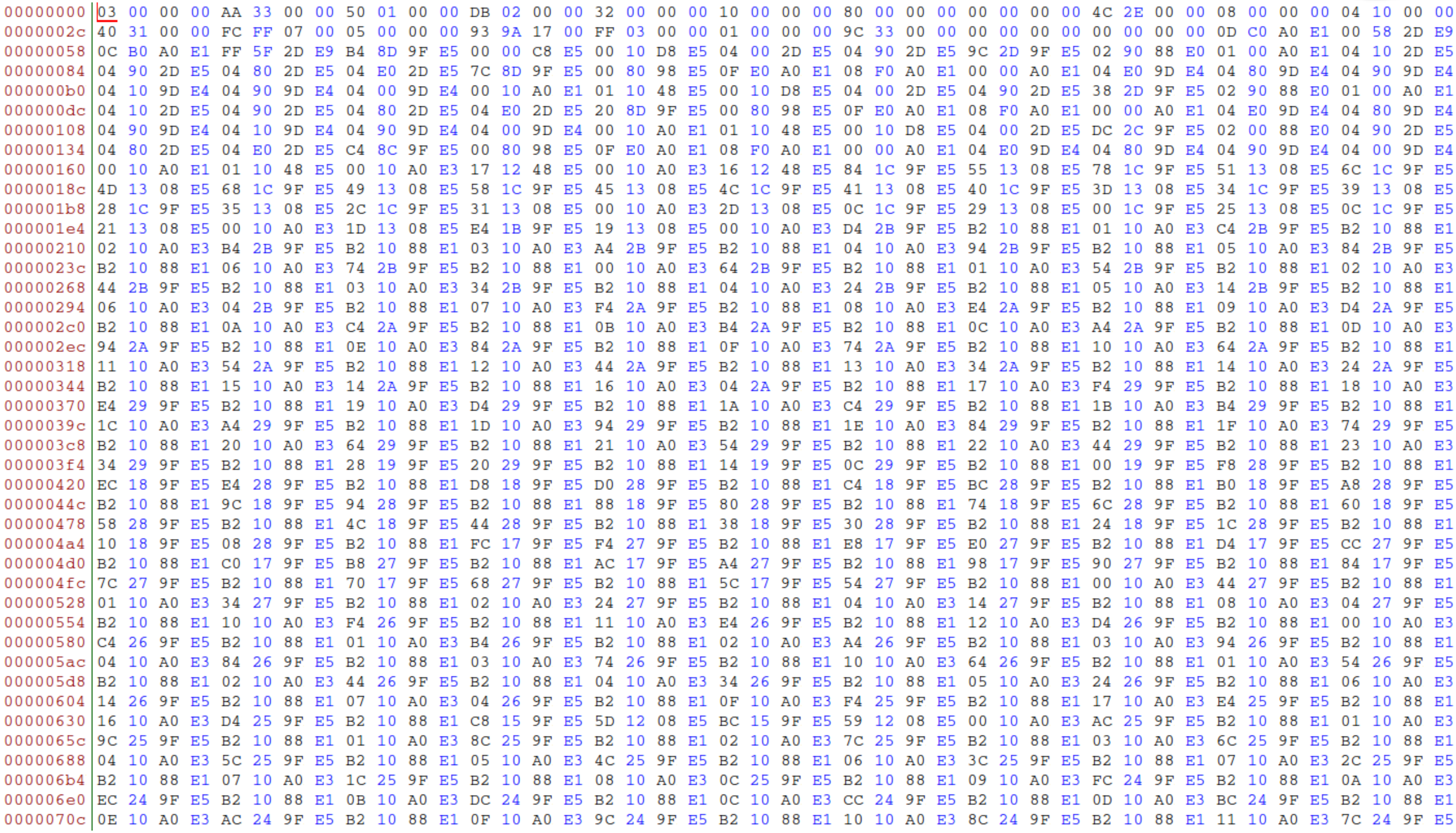


```
(icsref) me@example:$ ./icsref.py

ICS Reverse Engineering Framework
    ____________ ____  ________________
   /  _/ ____/ ___// __ \/ ____/ ____/
   / // /    \__ \/ /_/ / __/ / /_
 _/ // /___ ___/ / _, _/ /___/ __/
/___/\____//____/_/ |_/_____/_/

author: Tasos Keliris (@koukouviou)
Type <help> if you need a nudge
reversing@icsref:$
reversing@icsref:$ help

Documented commands (type help <topic>):
========================================
__changepid         changepid       exp_pid_match  history  pyscript  set
__replace_callname  cleanup         graphbuilder   load     quit      shell
_relative_load      cmdenvironment  hashmatch      pidargs  run       shortcuts
analyze             edit            help           py       save      show
```

[1] **A. Keliris**, M. Maniatakos, "*ICSREF: A Framework for Automated Reverse Engineering of Industrial Control Systems Binaries*", Network and Distributed System Security Symposium (NDSS), 2019.

Before ICSREF

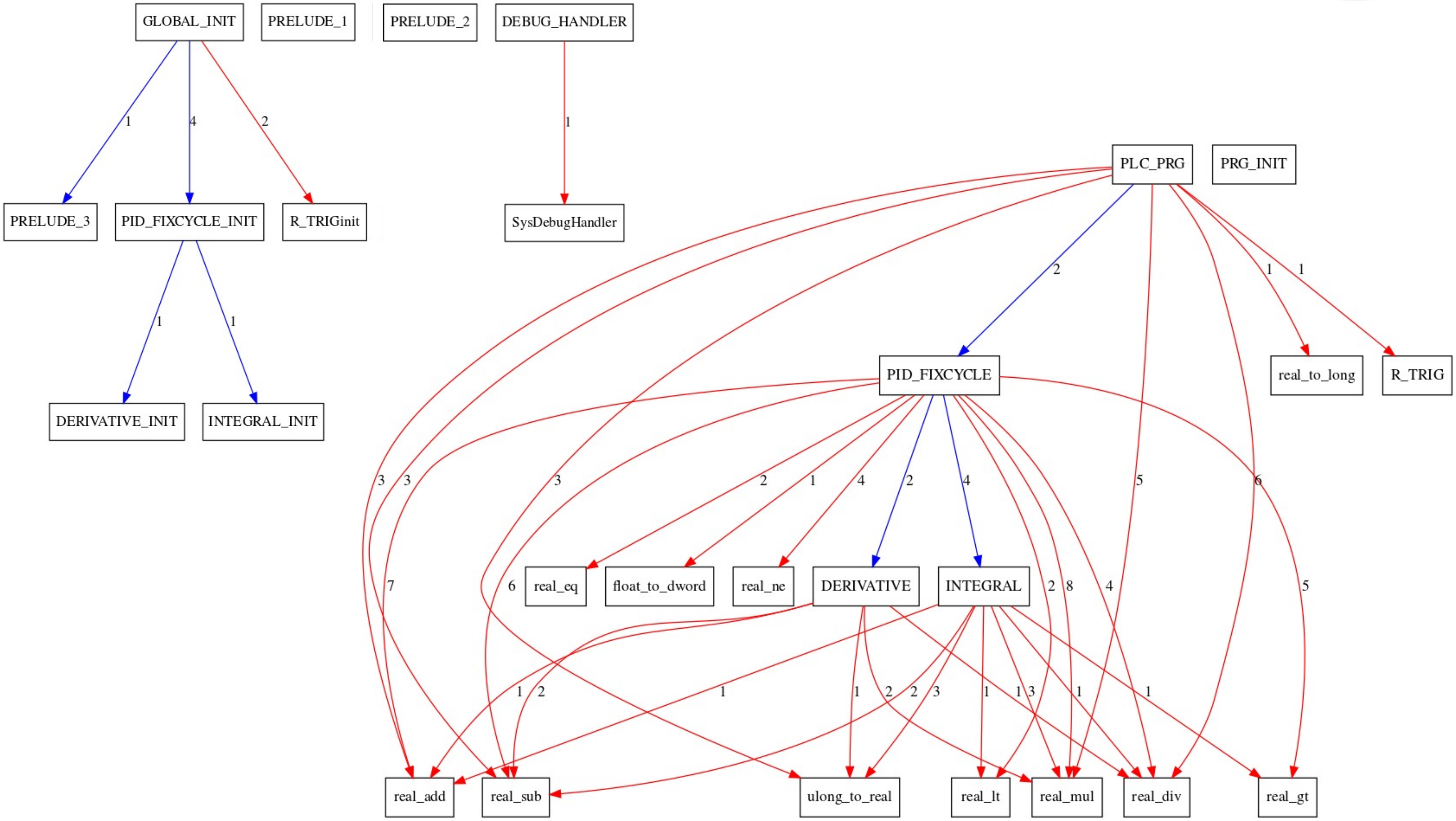




```
00000000 03 00 00 00 AA 33 00 00 50 01 00 00 DB 02 00 00 32 00 00 00 10 00 00 00 80 00 00 00 00 00 00 00 4C 2E 00 00 08 00 00 00 04 10 00 00
0000002c 40 31 00 00 FC FF 07 00 05 00 00 00 93 9A 17 00 FF 03 00 00 01 00 00 00 9C 33 00 00 00 00 00 00 00 00 00 00 0D C0 A0 E1 00 58 2D E9
00000058 0C B0 A0 E1 FF 5F 2D E9 B4 8D 9F E5 00 00 C8 E5 00 10 D8 E5 04 00 2D E5 04 90 2D E5 9C 2D 9F E5 02 90 88 E0 01 00 A0 E1 04 10 2D E5
00000084 04 90 2D E5 04 80 2D E5 04 E0 2D E5 7C 8D 9F E5 00 80 98 E5 0F E0 A0 E1 08 F0 A0 E1 00 00 A0 E1 04 E0 9D E4 04 80 9D E4 04 90 9D E4
000000b0 04 10 9D E4 04 90 9D E4 04 00 9D E4 00 10 A0 E1 01 10 48 E5 00 10 D8 E5 04 00 2D E5 04 90 2D E5 38 2D 9F E5 02 90 88 E0 01 00 A0 E1
000000dc 04 10 2D E5 04 90 2D E5 04 80 2D E5 04 E0 2D E5 20 8D 9F E5 00 80 98 E5 0F E0 A0 E1 08 F0 A0 E1 00 00 A0 E1 04 E0 9D E4 04 80 9D E4
00000108 04 90 9D E4 04 10 9D E4 04 90 9D E4 04 00 9D E4 00 10 A0 E1 01 10 48 E5 00 10 D8 E5 04 00 2D E5 DC 2C 9F E5 02 00 88 E0 04 90 2D E5
00000134 04 80 2D E5 04 E0 2D E5 C4 8C 9F E5 00 80 98 E5 0F E0 A0 E1 08 F0 A0 E1 00 00 A0 E1 04 E0 9D E4 04 80 9D E4 04 90 9D E4 04 00 9D E4
00000160 00 10 A0 E1 01 10 48 E5 00 10 A0 E3 17 12 48 E5 00 10 A0 E3 16 12 48 E5 84 1C 9F E5 55 13 08 E5 78 1C 9F E5 51 13 08 E5 6C 1C 9F E5
0000018c 4D 13 08 E5 68 1C 9F E5 49 13 08 E5 58 1C 9F E5 45 13 08 E5 4C 1C 9F E5 41 13 08 E5 40 1C 9F E5 3D 13 08 E5 34 1C 9F E5 39 13 08 E5
000001b8 28 1C 9F E5 35 13 08 E5 2C 1C 9F E5 31 13 08 E5 00 10 A0 E3 2D 13 08 E5 0C 1C 9F E5 29 13 08 E5 00 1C 9F E5 25 13 08 E5 0C 1C 9F E5
000001e4 21 13 08 E5 00 10 A0 E3 1D 13 08 E5 E4 1B 9F E5 19 13 08 E5 00 10 A0 E3 D4 2B 9F E5 B2 10 88 E1 01 10 A0 E3 C4 2B 9F E5 B2 10 88 E1
00000210 02 10 A0 E3 B4 2B 9F E5 B2 10 88 E1 03 10 A0 E3 A4 2B 9F E5 B2 10 88 E1 04 10 A0 E3 94 2B 9F E5 B2 10 88 E1 05 10 A0 E3 84 2B 9F E5
0000023c B2 10 88 E1 06 10 A0 E3 74 2B 9F E5 B2 10 88 E1 00 10 A0 E3 64 2B 9F E5 B2 10 88 E1 01 10 A0 E3 54 2B 9F E5 B2 10 88 E1 02 10 A0 E3
00000268 44 2B 9F E5 B2 10 88 E1 03 10 A0 E3 34 2B 9F E5 B2 10 88 E1 04 10 A0 E3 24 2B 9F E5 B2 10 88 E1 05 10 A0 E3 14 2B 9F E5 B2 10 88 E1
00000294 06 10 A0 E3 04 2B 9F E5 B2 10 88 E1 07 10 A0 E3 F4 2A 9F E5 B2 10 88 E1 08 10 A0 E3 E4 2A 9F E5 B2 10 88 E1 09 10 A0 E3 D4 2A 9F E5
000002c0 B2 10 88 E1 0A 10 A0 E3 C4 2A 9F E5 B2 10 88 E1 0B 10 A0 E3 B4 2A 9F E5 B2 10 88 E1 0C 10 A0 E3 A4 2A 9F E5 B2 10 88 E1 0D 10 A0 E3
000002ec 94 2A 9F E5 B2 10 88 E1 0E 10 A0 E3 84 2A 9F E5 B2 10 88 E1 0F 10 A0 E3 74 2A 9F E5 B2 10 88 E1 10 10 A0 E3 64 2A 9F E5 B2 10 88 E1
00000318 11 10 A0 E3 54 2A 9F E5 B2 10 88 E1 12 10 A0 E3 44 2A 9F E5 B2 10 88 E1 13 10 A0 E3 34 2A 9F E5 B2 10 88 E1 14 10 A0 E3 24 2A 9F E5
00000344 B2 10 88 E1 15 10 A0 E3 14 2A 9F E5 B2 10 88 E1 16 10 A0 E3 04 2A 9F E5 B2 10 88 E1 17 10 A0 E3 F4 29 9F E5 B2 10 88 E1 18 10 A0 E3
00000370 E4 29 9F E5 B2 10 88 E1 19 10 A0 E3 D4 29 9F E5 B2 10 88 E1 1A 10 A0 E3 C4 29 9F E5 B2 10 88 E1 1B 10 A0 E3 B4 29 9F E5 B2 10 88 E1
0000039c 1C 10 A0 E3 A4 29 9F E5 B2 10 88 E1 1D 10 A0 E3 94 29 9F E5 B2 10 88 E1 1E 10 A0 E3 84 29 9F E5 B2 10 88 E1 1F 10 A0 E3 74 29 9F E5
000003c8 B2 10 88 E1 20 10 A0 E3 64 29 9F E5 B2 10 88 E1 21 10 A0 E3 54 29 9F E5 B2 10 88 E1 22 10 A0 E3 44 29 9F E5 B2 10 88 E1 23 10 A0 E3
000003f4 34 29 9F E5 B2 10 88 E1 28 19 9F E5 20 29 9F E5 B2 10 88 E1 14 19 9F E5 0C 29 9F E5 B2 10 88 E1 00 19 9F E5 F8 28 9F E5 B2 10 88 E1
00000420 EC 18 9F E5 E4 28 9F E5 B2 10 88 E1 D8 18 9F E5 D0 28 9F E5 B2 10 88 E1 C4 18 9F E5 BC 28 9F E5 B2 10 88 E1 B0 18 9F E5 A8 28 9F E5
0000044c B2 10 88 E1 9C 18 9F E5 94 28 9F E5 B2 10 88 E1 88 18 9F E5 80 28 9F E5 B2 10 88 E1 74 18 9F E5 6C 28 9F E5 B2 10 88 E1 60 18 9F E5
00000478 58 28 9F E5 B2 10 88 E1 4C 18 9F E5 44 28 9F E5 B2 10 88 E1 38 18 9F E5 30 28 9F E5 B2 10 88 E1 24 18 9F E5 1C 28 9F E5 B2 10 88 E1
000004a4 10 18 9F E5 08 28 9F E5 B2 10 88 E1 FC 17 9F E5 F4 27 9F E5 B2 10 88 E1 E8 17 9F E5 E0 27 9F E5 B2 10 88 E1 D4 17 9F E5 CC 27 9F E5
000004d0 B2 10 88 E1 C0 17 9F E5 B8 27 9F E5 B2 10 88 E1 AC 17 9F E5 A4 27 9F E5 B2 10 88 E1 98 17 9F E5 90 27 9F E5 B2 10 88 E1 84 17 9F E5
000004fc 7C 27 9F E5 B2 10 88 E1 70 17 9F E5 68 27 9F E5 B2 10 88 E1 5C 17 9F E5 54 27 9F E5 B2 10 88 E1 00 10 A0 E3 44 27 9F E5 B2 10 88 E1
00000528 01 10 A0 E3 34 27 9F E5 B2 10 88 E1 02 10 A0 E3 24 27 9F E5 B2 10 88 E1 04 10 A0 E3 14 27 9F E5 B2 10 88 E1 08 10 A0 E3 04 27 9F E5
00000554 B2 10 88 E1 10 10 A0 E3 F4 26 9F E5 B2 10 88 E1 11 10 A0 E3 E4 26 9F E5 B2 10 88 E1 12 10 A0 E3 D4 26 9F E5 B2 10 88 E1 00 10 A0 E3
00000580 C4 26 9F E5 B2 10 88 E1 01 10 A0 E3 B4 26 9F E5 B2 10 88 E1 02 10 A0 E3 A4 26 9F E5 B2 10 88 E1 03 10 A0 E3 94 26 9F E5 B2 10 88 E1
000005ac 04 10 A0 E3 84 26 9F E5 B2 10 88 E1 03 10 A0 E3 74 26 9F E5 B2 10 88 E1 10 10 A0 E3 64 26 9F E5 B2 10 88 E1 01 10 A0 E3 54 26 9F E5
000005d8 B2 10 88 E1 02 10 A0 E3 44 26 9F E5 B2 10 88 E1 04 10 A0 E3 34 26 9F E5 B2 10 88 E1 05 10 A0 E3 24 26 9F E5 B2 10 88 E1 06 10 A0 E3
00000604 14 26 9F E5 B2 10 88 E1 07 10 A0 E3 04 26 9F E5 B2 10 88 E1 0F 10 A0 E3 F4 25 9F E5 B2 10 88 E1 17 10 A0 E3 E4 25 9F E5 B2 10 88 E1
00000630 16 10 A0 E3 D4 25 9F E5 B2 10 88 E1 C8 15 9F E5 5D 12 08 E5 BC 15 9F E5 59 12 08 E5 00 10 A0 E3 AC 25 9F E5 B2 10 88 E1 01 10 A0 E3
0000065c 9C 25 9F E5 B2 10 88 E1 01 10 A0 E3 8C 25 9F E5 B2 10 88 E1 02 10 A0 E3 7C 25 9F E5 B2 10 88 E1 03 10 A0 E3 6C 25 9F E5 B2 10 88 E1
00000688 04 10 A0 E3 5C 25 9F E5 B2 10 88 E1 05 10 A0 E3 4C 25 9F E5 B2 10 88 E1 06 10 A0 E3 3C 25 9F E5 B2 10 88 E1 07 10 A0 E3 2C 25 9F E5
000006b4 B2 10 88 E1 07 10 A0 E3 1C 25 9F E5 B2 10 88 E1 08 10 A0 E3 0C 25 9F E5 B2 10 88 E1 09 10 A0 E3 FC 24 9F E5 B2 10 88 E1 0A 10 A0 E3
000006e0 EC 24 9F E5 B2 10 88 E1 0B 10 A0 E3 DC 24 9F E5 B2 10 88 E1 0C 10 A0 E3 CC 24 9F E5 B2 10 88 E1 0D 10 A0 E3 BC 24 9F E5 B2 10 88 E1
0000070c 0E 10 A0 E3 AC 24 9F E5 B2 10 88 E1 0F 10 A0 E3 9C 24 9F E5 B2 10 88 E1 10 10 A0 E3 8C 24 9F E5 B2 10 88 E1 11 10 A0 E3 7C 24 9F E5
```

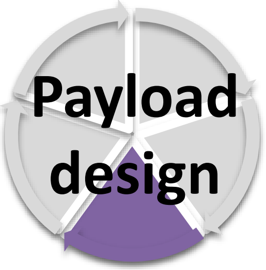
After ICSREF

Payload design



ICSREF capabilities/modules

For binaries compiled with CODESYS



- Binary subroutines
- Dynamic functions
- Static functions
- Physical I/O
- Call graph
- PID arguments
- Modify binaries

analysis
ap

🔴	Header	Offsets information
🔵	Global INIT	Initialization of global memory
🔵	Sub 1	Support subroutine
🔵	Sub 2	Support subroutine
🔵	Sub 3	Support subroutine
🔵	SYSDEBUG	Debugger handler
🔵	StaticLib	Statically linked library function 1
🔵	FB_n	User-defined Function Block n
🔵	FB_n INIT	User-defined Function Block n initialization
🔵	PLC_PRG	Main PLC Program (PRG)
🔵	Memory INIT	Program memory initialization
🔴	Data	Data
🔴	Dynamic libs	Dynamic library functions information
🔴	Data	Data

Legend
🔵 Code
🔴 Data

```
# Subroutine entry point
   MOV    R12, SP
   STMFD  SP!, {R11,R12,LR}
# Code
   ...
   ...
```

```
# Code
   ...
   ...
# Call other subroutine
   STR    Ri, [SP,#-4]!
   STR    LR, [SP,#-4]!
```

```
   ...
```

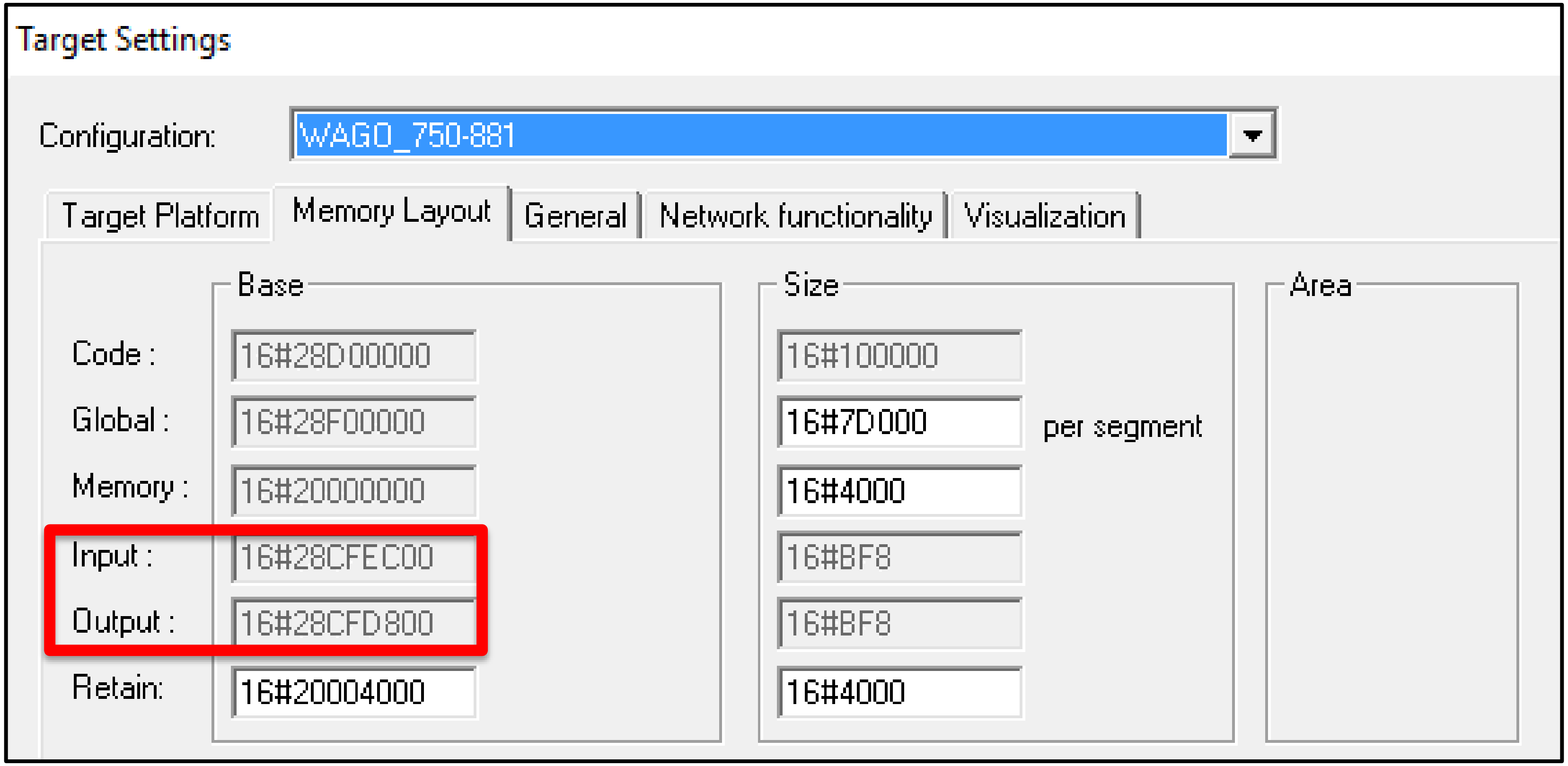
```
loc_Y:
# Code
   ...
   ...
# Subroutine exit
   LDMDB  R11, {R11,SP,PC}
# Data
   0xCAFEBABE
   0xDEADBEEF
   ...
   ...
```

☑ Statically analyze binary and find subroutines, static and dynamic calls

Extracting PLC memory maps

Payload design

- CODESYS uses *.TRG files that hold the particular controller memory maps



Target Settings

Configuration: WAGO_750-881

Target Platform | Memory Layout | General | Network functionality | Visualization

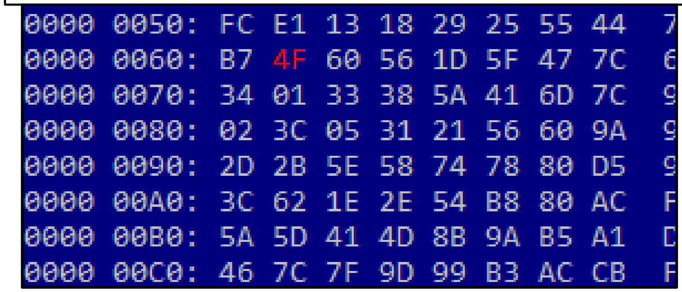
	Base	Size	
Code :	16#28D00000	16#100000	
Global :	16#28F00000	16#7D000	per segment
Memory :	16#20000000	16#4000	
Input :	16#28CFEC00	16#BF8	
Output :	16#28CFD800	16#BF8	
Retain:	16#20004000	16#4000	

Area

Extracting PLC memory maps

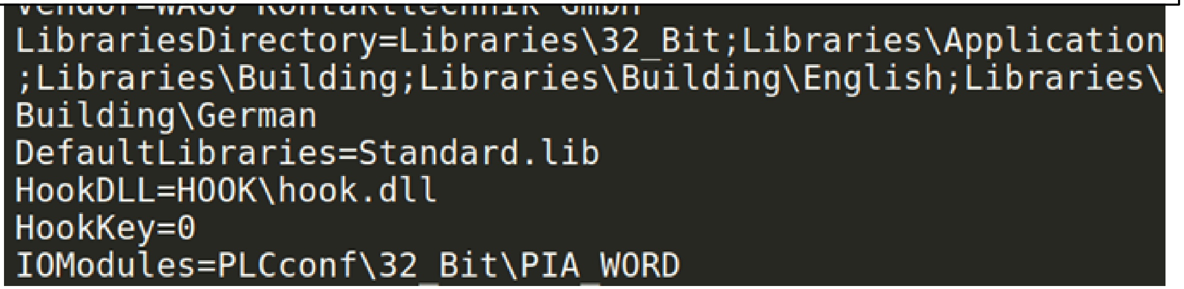
- Reverse engineering
  - Extracted 256-byte key

☑ Identify reads/writes from/to physical I/O



```
0000 0050: FC E1 13 18 29 25 55 44  7
0000 0060: B7 4F 60 56 1D 5F 47 7C  6
0000 0070: 34 01 33 38 5A 41 6D 7C  9
0000 0080: 02 3C 05 31 21 56 60 9A  9
0000 0090: 2D 2B 5E 58 74 78 80 D5  9
0000 00A0: 3C 62 1E 2E 54 B8 80 AC  F
0000 00B0: 5A 5D 41 4D 8B 9A B5 A1  D
0000 00C0: 46 7C 7F 9D 99 B3 AC CB  F
```

Before



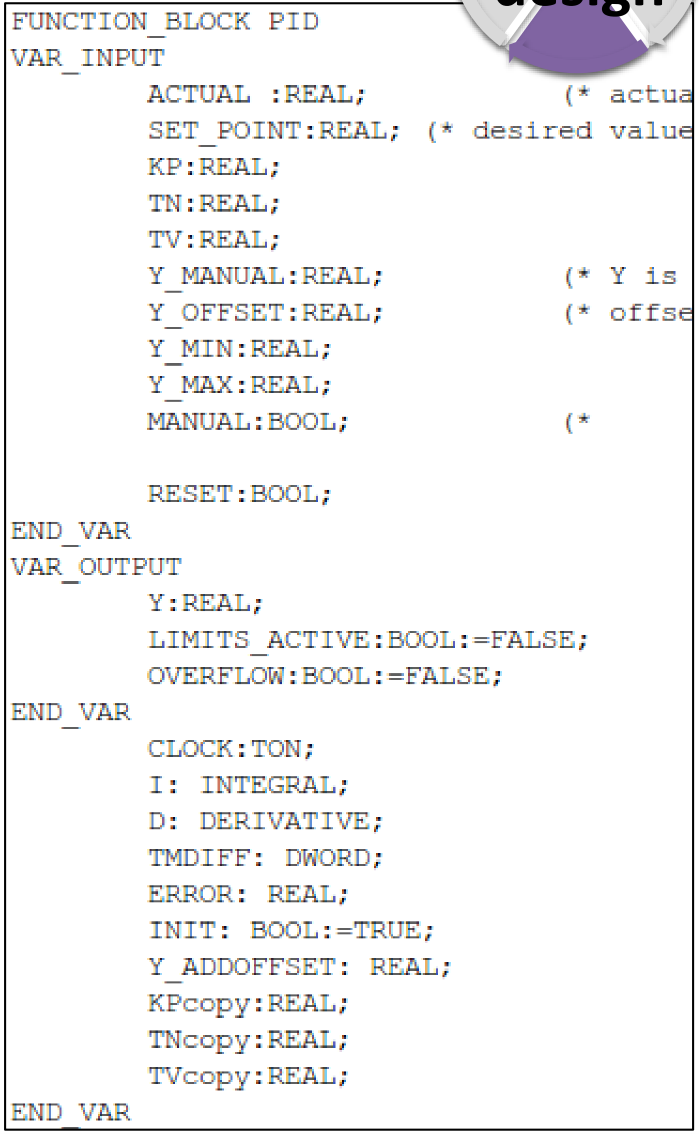
```
Vendor=WAGO Kontakttechnik GmbH
LibrariesDirectory=Libraries\32_Bit;Libraries\Application
;Libraries\Building;Libraries\Building\English;Libraries\
Building\German
DefaultLibraries=Standard.lib
HookDLL=HOOK\hook.dll
HookKey=0
IOModules=PLCconf\32_Bit\PIA_WORD
```

After

Standard library functions

Payload design

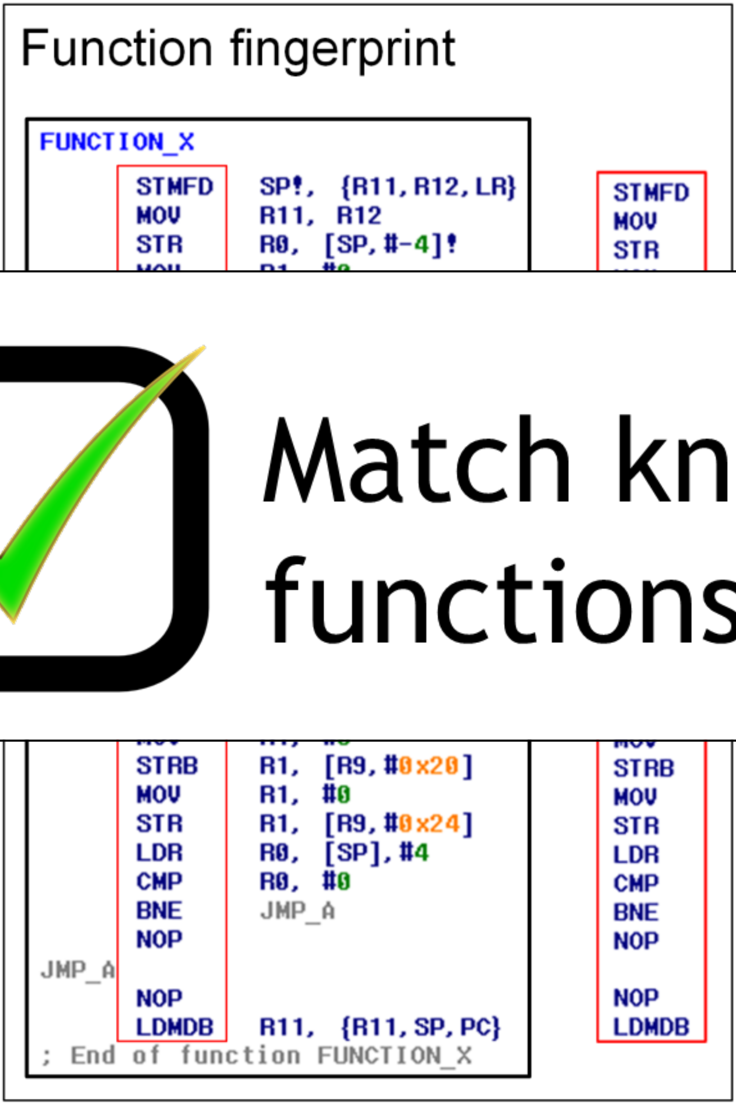
- Deobfuscated and decrypted library files
  - Same key as *.TRG files
  - Extracted source code for CODESYS libraries
- Built prototypes for all library functions
  - Name
  - Inputs/Outputs
  - Dependencies



```
FUNCTION_BLOCK PID
VAR_INPUT
        ACTUAL :REAL;           (* actua
        SET_POINT:REAL;  (* desired value
        KP:REAL;
        TN:REAL;
        TV:REAL;
        Y_MANUAL:REAL;          (* Y is
        Y_OFFSET:REAL;          (* offse
        Y_MIN:REAL;
        Y_MAX:REAL;
        MANUAL:BOOL;            (*

        RESET:BOOL;
END_VAR
VAR_OUTPUT
        Y:REAL;
        LIMITS_ACTIVE:BOOL:=FALSE;
        OVERFLOW:BOOL:=FALSE;
END_VAR
        CLOCK:TON;
        I: INTEGRAL;
        D: DERIVATIVE;
        TMDIFF: DWORD;
        ERROR: REAL;
        INIT: BOOL:=TRUE;
        Y_ADDOFFSET: REAL;
        KPcopy:REAL;
        TNcopy:REAL;
        TVcopy:REAL;
END_VAR
```

Opcode-based signatures

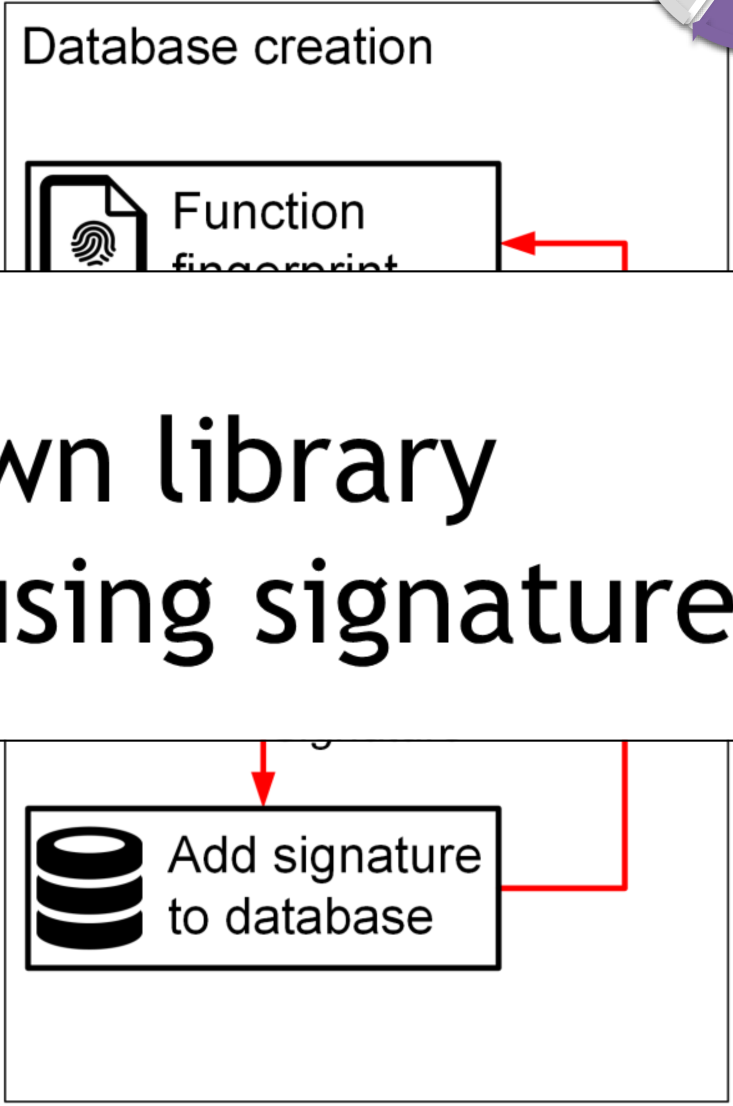
Payload design



Function fingerprint

```
FUNCTION_X
        STMFD   SP!, {R11,R12,LR}
        MOV     R11, R12
        STR     R0, [SP,#-4]!
```

```
        STRB    R1, [R9,#0x20]
        MOV     R1, #0
        STR     R1, [R9,#0x24]
        LDR     R0, [SP],#4
        CMP     R0, #0
        BNE     JMP_A
        NOP
JMP_A
        NOP
        LDMDB   R11, {R11,SP,PC}
; End of function FUNCTION_X
```



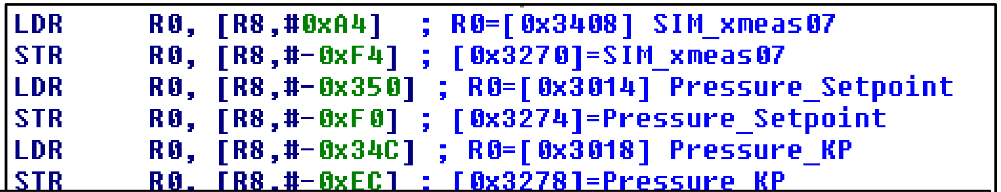
Database creation

Match known library functions using signatures

Finding function arguments

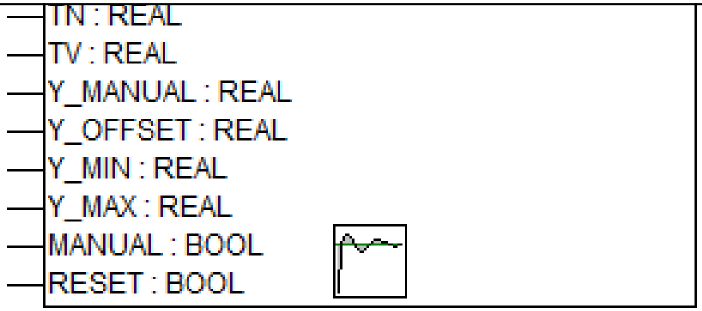
Payload design

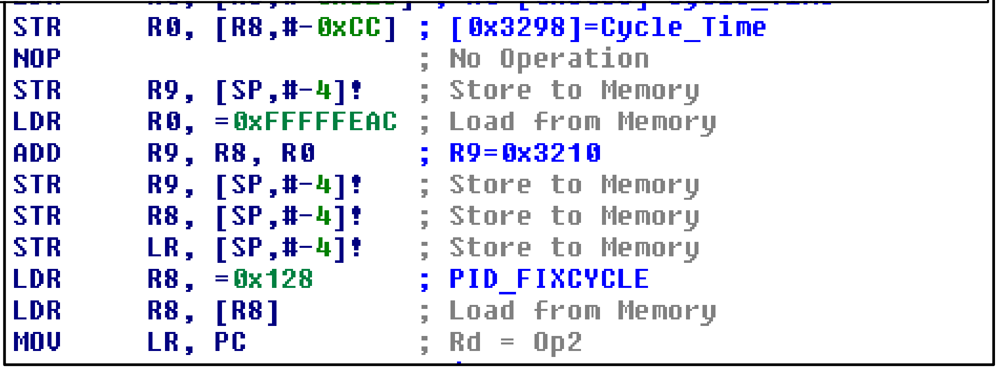
- Arguments passed on the stack



```
LDR      R0, [R8,#0xA4]   ; R0=[0x3408] SIM_xmeas07
STR      R0, [R8,#-0xF4] ; [0x3270]=SIM_xmeas07
LDR      R0, [R8,#-0x350] ; R0=[0x3014] Pressure_Setpoint
STR      R0, [R8,#-0xF0] ; [0x3274]=Pressure_Setpoint
LDR      R0, [R8,#-0x34C] ; R0=[0x3018] Pressure_KP
STR      R0, [R8,#-0xEC] ; [0x3278]=Pressure_KP
```

☑ Extract arguments of function calls (PID)

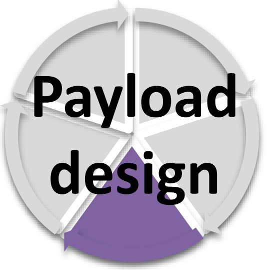




```
STR      R0, [R8,#-0xCC] ; [0x3298]=Cycle_Time
NOP                       ; No Operation
STR      R9, [SP,#-4]!    ; Store to Memory
LDR      R0, =0xFFFFFEAC  ; Load from Memory
ADD      R9, R8, R0       ; R9=0x3210
STR      R9, [SP,#-4]!    ; Store to Memory
STR      R8, [SP,#-4]!    ; Store to Memory
STR      LR, [SP,#-4]!    ; Store to Memory
LDR      R8, =0x128       ; PID_FIXCYCLE
LDR      R8, [R8]         ; Load from Memory
MOV      LR, PC           ; Rd = Op2
```

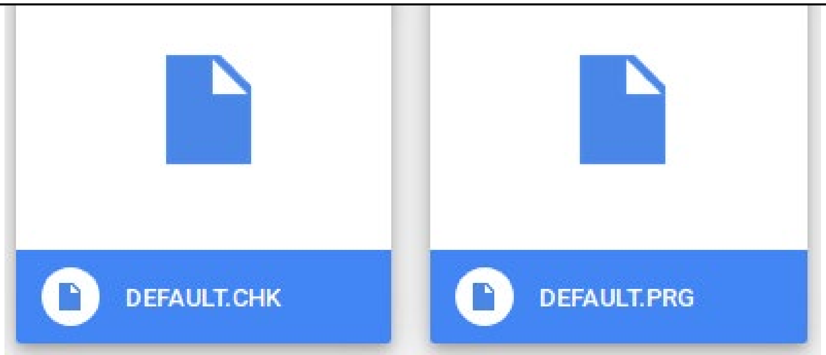
Binary modification

Checksum (*.CHK)

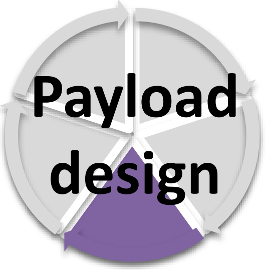


- Each compiled binary is uploaded to the PLC along with a checksum file

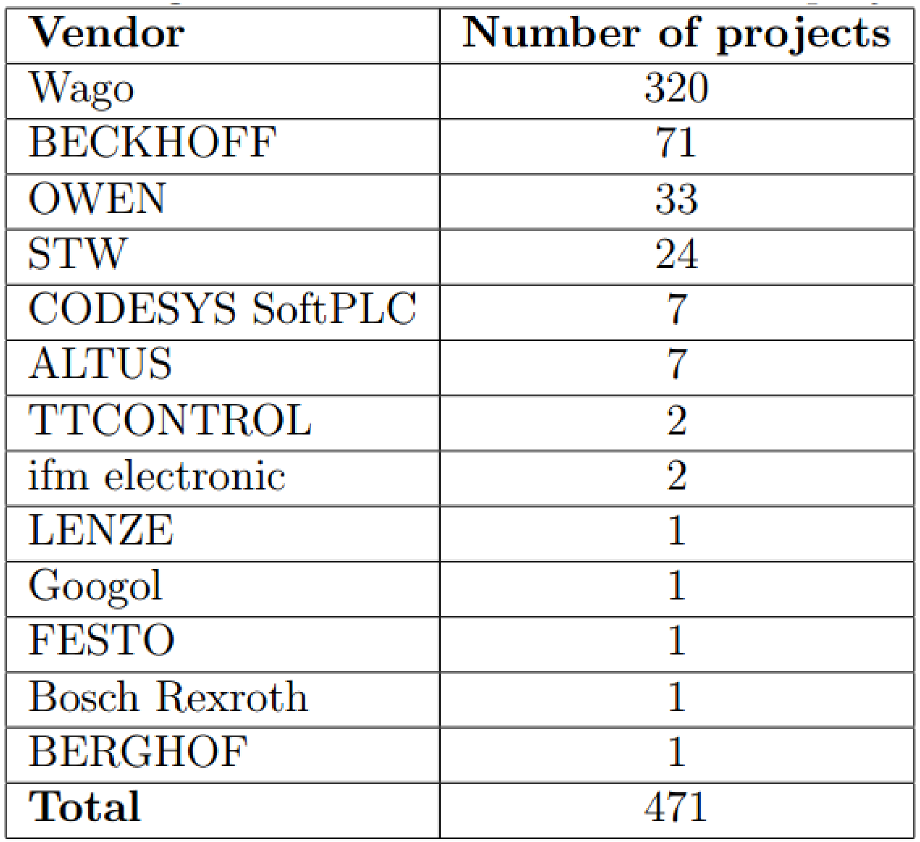
☑ Modify binary



ICSREF correctness evaluation



- In-house binaries
  - For HITL testbed
- Online code repositories (GitHub)
  - 55 users
  - 127 repositories
  - 471 source code and binaries
- 266 binaries used for testing
  - The other projects are code stubs or corrupted

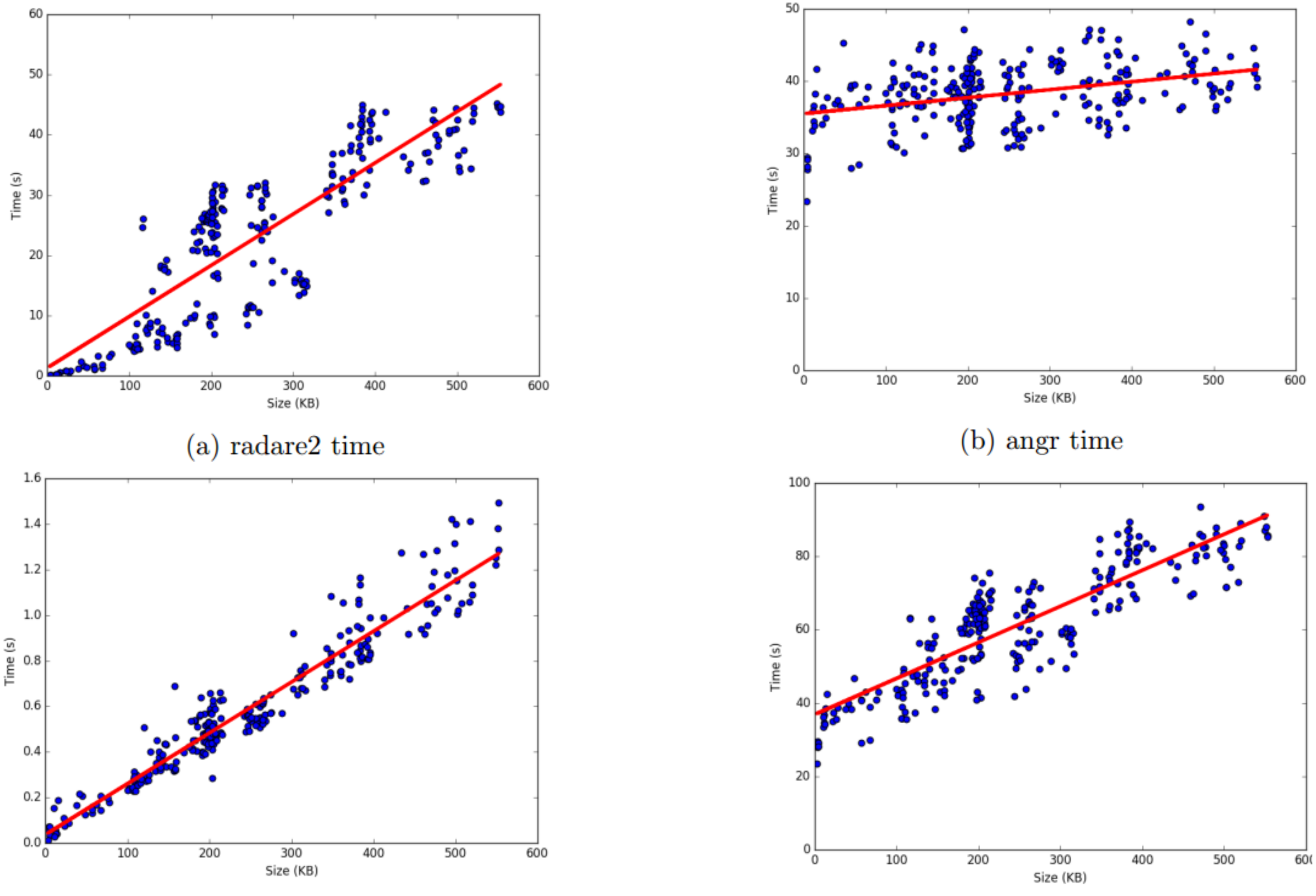


Vendor	**Number of projects**
Wago	320
BECKHOFF	71
OWEN	33
STW	24
CODESYS SoftPLC	7
ALTUS	7
TTCONTROL	2
ifm electronic	2
LENZE	1
Googol	1
FESTO	1
Bosch Rexroth	1
BERGHOF	1
Total	471

ICSREF performance

Dell XPS 9360: Intel i7-7500U CPU, 16 GB RAM, Ubuntu 16.04

Payload design



(a) radare2 time

(b) angr time

(c) Other operations

(d) Total time

Applications of ICSREF

- Forensics (Malware analysis)
- Parameters recovery (IP)
- Retrofitting security solutions
- Dynamic malware development

Current research

- ICS emulation
  - Allows in-field cybersecurity assessment
  - Current status: Can perform fuzzing on QEMU instance of Wago (TBP @ DATE20)
- JTAG-based fuzzing
  - Full visibility, slow speed
  - Can mostly be used for deep state exploration

Conclusions

All info can be found at: wp.nyu.edu/momalab

- ICS security is bad, and we should feel bad
  - Problem will stay for 20-30 years
  - Solutions are needed across the stack (not just network)
- Follow me @realmomalab (mostly lurker, working on it!)
- Good stuff:
  - ICSREF: github.com/momalab/ICSREF
  - NDSS talk: youtube.com/watch?v=kixDkd4z41s
  - BlackHat talk: wp.nyu.edu/momalab/2017/07/27/blackhat-talk/
- Please come visit us at NYU Abu Dhabi and see the testbed
- Questions?