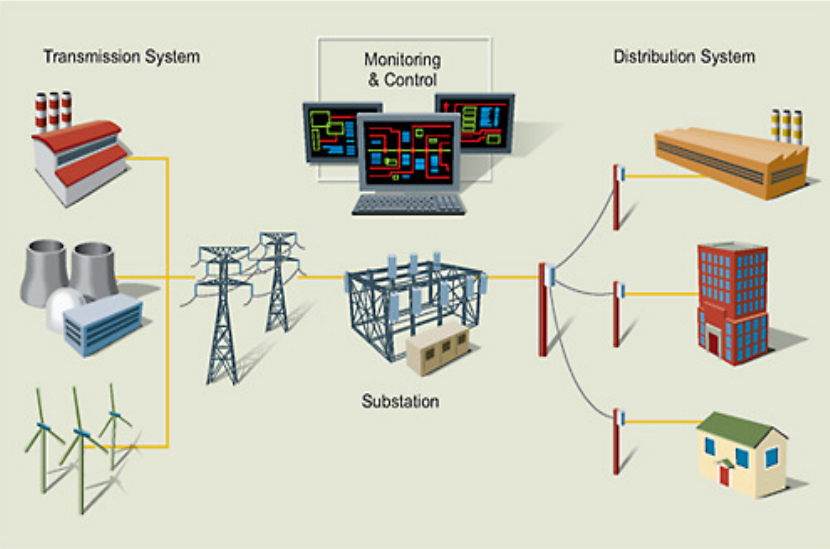
Evaluating Effectiveness of an Embedded System Endpoint Security Technology on EDS

Michael Siegel, Gregory Falco, Keman Huang, Weilian Chu, Elizabeth Reilly, Mayukha Vadari

CREDC

Digitization of Industrial Sector



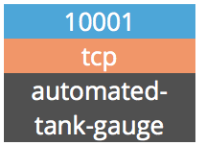
- Increased demand on utilities industry
- More optimized distribution required
- Digitization of system endpoints
- Two-way communication between consumer & distributor

Industrial IoT Endpoint Devices

- Single user device, interacts with larger system of devices
- Interacts with people, usually has IP address
- Smart meters, gas pipes, oil tanks, wind turbines
- Vulnerable to malicious access & tampering

Example - Automated Gas Storage Tank

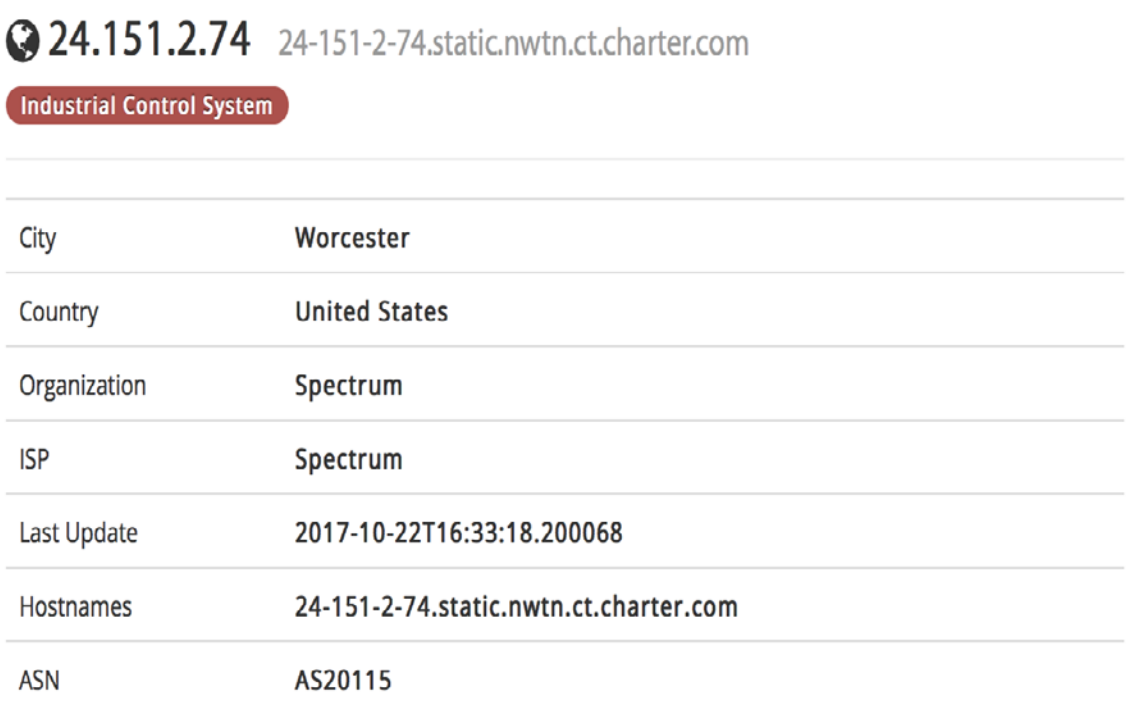
- Protocol & port # available
- Exact address
- Database information & timestamp




```
I20100
OCT 22, 2017 11:28 AM

828194 GAS STOP
47 CHURCH HILL RD
NEWTOWN CT
06470

IN-TANK INVENTORY

TANK PRODUCT               VOLUME TC VOLUME   ULLAGE   HEIGHT    WATER     TEMP
  1  DIESEL                  2529       2522     1481    57.92     0.00    65.86
  2  KEROSENE                 576        575     1429    31.65     0.00    64.55
  3  SUPER                   2978       2969     7022    32.48     0.00    63.66
  4  REGULAR                 4059       4043     5941    40.88     0.00    65.47
```

Example - Automated Gas Storage Tank

- ASN revealed
- Many devices have open SSH ports that allow for public access
- IP address vulnerable to ssh entry through password crackers



Consequences of Security Compromise

- Information & power theft
- Possibility of malicious control
- Disruption of distribution service to consumers
- Physical and technological infrastructure damage
- User security compromised

Why are Industrial IoT endpoints hard to secure?

- Can’t defend against users with malicious intent
- Industrial endpoints low in memory and storage
- Not enough computation power for conventional IoT security measures

- Certificate Verification

- Encrypted IoT network

Our Project - Overview

Lightweight Security Architecture

Blockchain Server

Lightweight Security Architecture

- Software enforce security policies from within device
- Written in C & Bash
- Lock down endpoint OS to limit its capabilities
- Prevents unauthorized programs from running in OS
- Small footprint -> works within the kernel -> doesn’t require network access
- Intensive computations are performed in the cloud

Blockchain Technology

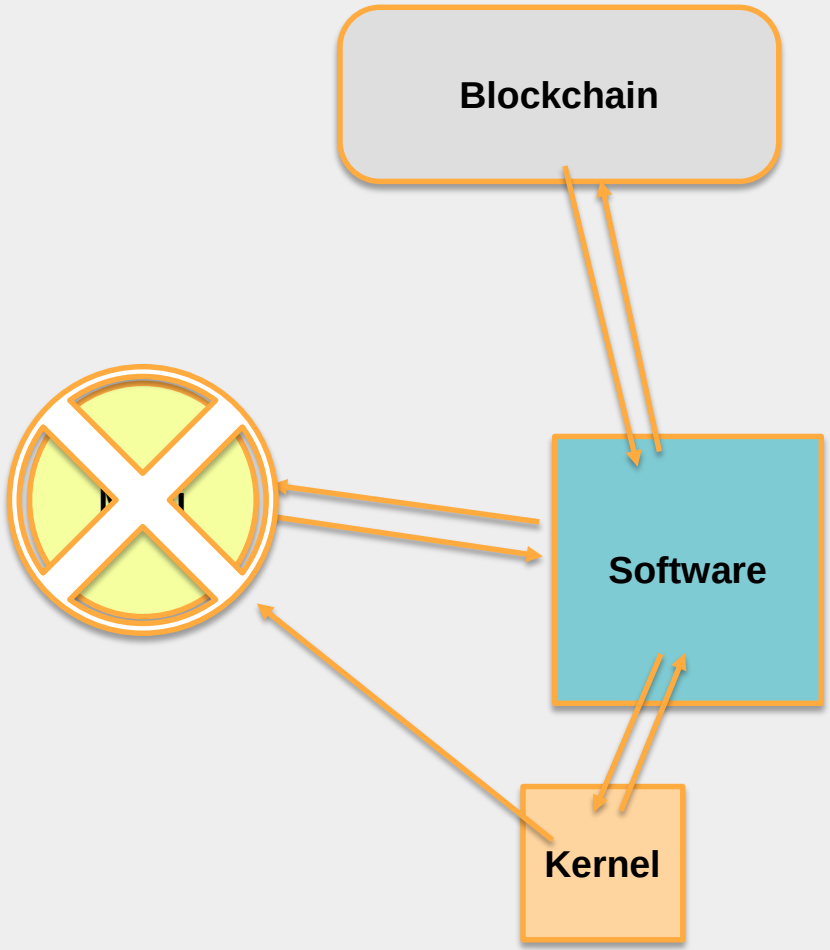
- Foundation for command & control
- Sends security updates, stores them in secure & decentralized channel
- Provided by Bitcoin Blockchain
- Controls applications that are black/whitelisted
- Does not interfere with firmware -> no system downtime during updates

Project Demo: Mirai

- Mirai is a famous malware botnet that targets Linux routers
- Ran open source software OpenWRT on linux virtual machine to simulate a router
- Compiled our security software and installed onto OpenWRT VM
- Attempted to run Mirai botnet on the VM

Project Demo: Mirai

- Software constantly checks for traces of Mirai
- Software has kernel privilege within OS
- Any process outside core system is verified over blockchain
- Any program that doesn’t pass through the black/whitelist is killed



Timeline - Future goals

Dec 2017	March 2018	April 2018	May 2018
Port existing software onto router	Run software on smart meters	Clustering algorithms to detect malicious IP addresses	Port software onto Windows-based devices
Software updates via VPN	Develop heuristics for machine learning analysis		

Industry Partners

CYBER RESILIENT ENERGY DELIVERY CONSORTIUM

http://cred-c.org

@credcresearch

facebook.com/credcresearch/

Funded by the U.S. Department of Energy and the U.S. Department of Homeland Security