

Building secure software for operational technologies applications with forensics readiness in mind

Trial lecture - Associate professor position in Information Security

Victor Morel

Chalmers University of Technology and University of Gothenburg

August 15, 2025

Kristiania University of Applied Sciences

Froosty Goop 🍦



Malware discovered in April 2024

- Targets the Modbus protocol (OT)
- Disrupted heating systems in the Ukrainian city of Lviv
- 600 buildings with no heat ... in January 🤖

Securing OT infrastructure is **critical!**

What's OT doc? 🥕

Operational Technology (OT)

Refers to hardware and software **systems** used to **monitor and control** physical devices, processes, and events in industrial and critical infrastructure environments.

Found in various sectors



Manufacturing



Energy



Utilities



Transportation

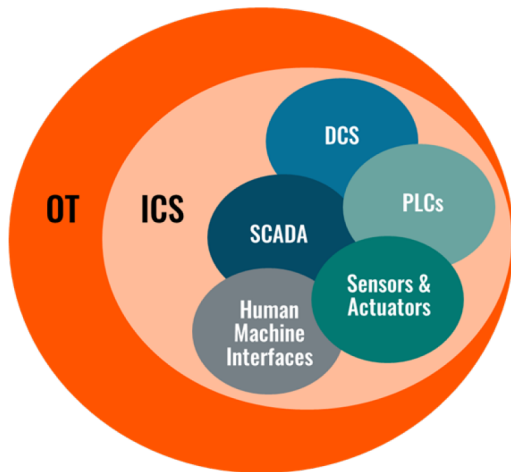


Healthcare

OT vs. IT (Information Technology)

In a nutshell: **IT** deals with an organization's *front-end* informational activity while **OT** focuses on the *back-end* production (machines).

Main technical components



- › Industrial Control Systems
 - › Supervisory Control and Data Acquisition
 - › Distributed Control Systems
 - › Programmable Logic Controllers
- And many more ...

Quiz time!



Secure software in a nutshell

Security is a process, not an end

Secure software refers to software that is designed and developed with **security in mind**, aiming to **protect against** unauthorized access, use, disclosure, disruption, modification, or destruction. The goal is to **minimize vulnerabilities** and ensure that the software can continuously **withstand and respond** to security threats.

A framework for secure software (by NIST)

- › *Prepare the Organization* (PO) (define requirements, implement roles...)
- › *Protect the Software* (PS) (secure repositories...)
- › *Produce Well-Secured Software* (PW) (design, review, test...)
- › *Respond to Vulnerabilities* (RV) (identify, prioritize, remedy, analyze root cause...)

Secure coding

What does it consist in?

Following coding standards and guidelines that help prevent common vulnerabilities such as SQL injection, cross-site scripting (XSS), and **buffer overflows**.

Buffer overflow

- Happens when a process attempts to store data **beyond** a fixed-length **buffer**.
- In an OT context, an attacker can overpass the **safety values** of industrial machines, and/or execute payloads.
- It can lead to remote control, **disruption**, or destruction.

Quiz time!



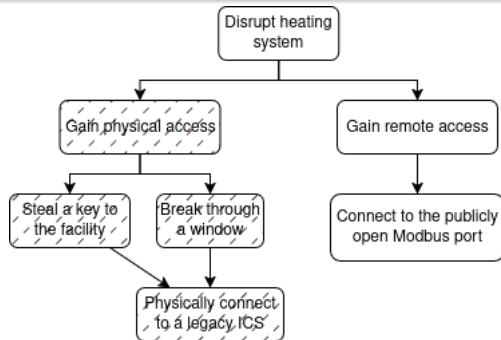
Threat modeling

What does it consist in?

Identifying potential threats and vulnerabilities early in the development process and designing mitigations for them.


Old school: attack trees

- 0 Define adversaries
- 1 Create an attack tree
- 2 Assess risks
- 3 Keep viable attack vectors
- 4 Devise countermeasures
- 5 Optimize countermeasures




Quiz time!



An important question remains 

Whodunit?

The attack has not been officially
attributed nor claimed...
(but we have pointers )

But also

And how we can efficiently defend against
it next time?

For that, we need forensics readiness!

What is forensics science?



Historically

Scientific investigations for matters of criminal and civil law.

DNA, fingerprints, ballistics...

What does it mean in a digital context?

Forensics readiness refers to the ability of an organization to efficiently and effectively **collect, preserve, and analyze digital evidence** in the event of a security incident or breach. It involves **preparing and planning** for potential forensic investigations to ensure that **evidence is available and admissible** in legal proceedings.

Logging and monitoring



Logging



Monitoring

```
#separator \x09
#set_separator ,
#empty_field (empty)
#unset_field -
#path modbus
#open 2020-07-07-18-00-43
#fields ts uid id.orig_h id.orig_p id.resp_h id.resp_p func exception
#types time string addr port addr port string string
1594166443.370237 CKJw923A52zHnHyd33 10.1.2.38 39058 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166443.510211 CbG3na1QXT7tyhzTqj 10.1.2.38 39060 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166443.670235 CGyqta3kN9273U2REd 10.1.2.38 39062 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166443.810233 Ctx4iz31uZjA4MZpx4 10.1.2.38 39064 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166443.970180 CRJAdAxlauxCj4Daa 10.1.2.38 39066 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166444.110246 CuJ3P24HLuXWlQhe6h 10.1.2.38 39068 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166444.250243 Ce20u74Ch1ZjcxapFh 10.1.2.38 39070 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166457.460197 CrIK7tmfV65lrHHc1 10.1.2.9 52959 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166457.600215 CrIK7tmfV65lrHHc1 10.1.2.9 52959 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166457.760227 CrIK7tmfV65lrHHc1 10.1.2.9 52959 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166457.900181 CrIK7tmfV65lrHHc1 10.1.2.9 52959 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166502.920249 CRyVoU2hkwtYuIWM1j 10.1.2.38 39086 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166503.060229 CL0GUC3Hg9GwYv10a6 10.1.2.38 39088 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166503.220206 Csk3jx1Qc2vE1Wh4Xb 10.1.2.38 39090 10.1.2.30 502 READ_HOLDING_REGISTERS -
1594166503.380233 CfLALc2VN2098k8nAj 10.1.2.38 39092 10.1.2.30 502 READ_HOLDING_REGISTERS -
```

Figure 57: Zeek Modbus logs that identify attacker IP address.

Data integrity



Log integrity can be preserved through cryptographic hashes.

But also:

- › Digital signatures
- › Append-only secure ledger

The goals

Implement mechanisms to preserve log data and other forensic evidence in a secure and tamper-evident manner.

- › Prevent attackers from covering their tracks
- › Help make the evidence admissible in a legal procedure

Quiz time!



Summary

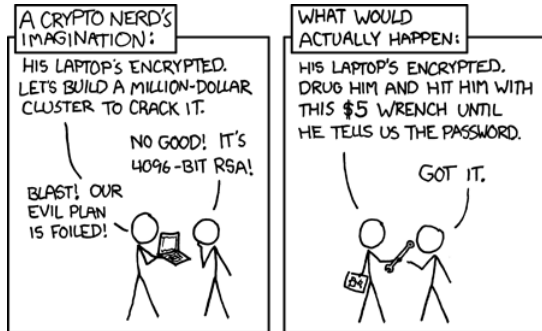
Following the example of the Froosty Goop attack in April 2024.

What we saw today

- › An overview of Operational Technology (OT)
- › How to secure OT
 - Via threat modeling
 - And secure coding
- › And how to make it in a forensics-ready way
 - Using logging/monitoring
 - And data integrity
- › Securing OT matters, but it is not trivial

Many more things to consider for **secure OT software**! Network security, incident response planning... And so is **forensics-readiness**! Compliance, internal training...

The weakest link



Technology can be hacked, but so can humans

Human factors in security

- How do we make secure solutions more usable?
- How do we ensure acceptance?
- How do we train staff for incident response? etc.