EASY-RES

SUMMER SCHOOL "ENABLING DRES TO OFFER ANCILLARY SERVICES" 20TH – 24TH SEPTEMBER 2021

ICT Security as a First-Class Entity

Dr John Vidler & Dr Martin Bor // 23 September 2021

This project has received funding from the European Union´s Horizon 2020 Programme for research and innovation under Grant Agreement no 764090.

OR...

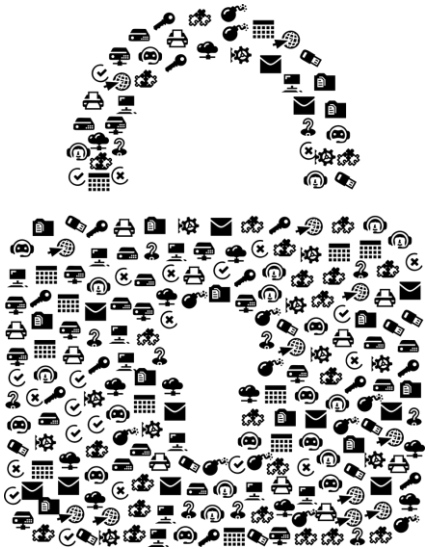
A Brief Introduction to Security for Non-Security Folks

(...a practical approach)

Agenda

- What to expect from this session
- Risk, Threat and Vulnerability
- Some history
- Thinking like an attacker
- Assessing a system
- Component Security
- Communication Security
- Information Security
- Summary
- Open Q&A session

What to Expect...



- We've seen lots of theory so far!
- This is going to be a bit different
  - More of a practical, interactive session
  - Feel free to post questions, and comments in the text chat
- Not expected to be a security expert by the end
  - ...but you should know where to be concerned!
- **Ask questions throughout – Martin is in the chat and will collect them for us to answer directly, or for the Q&A at the end**

Risk Analysis

What is the risk, threat, vulnerability?

- Bald car tyre
- …tied to a rope hanging from a tree branch,
- …the rope is frayed halfway through,
- …hanging over a 80-foot cliff with – sharp rocks at the bottom.

An Introduction to Factor Analysis of Information Risk (FAIR), Jack A. Jones

What is the risk, threat, vulnerability?

1. Assumptions
    - Who cares if an empty, old bald tyre falls to the rocks below?

2. Terminology
    - What do we exactly mean with “risk”, “threat” and “vulnerability”

3. No significant risk without significant loss
    - Who cares if an empty, old bald tyre falls to the rocks below?

4. Vulnerability != Risk

An Introduction to Factor Analysis of Information Risk (FAIR), Jack A. Jones

Terminology

- **Asset**
  - *Any data*, device, or other component of the environment that supports information-related activities, which can be illicitly accessed, used, disclosed, altered, destroyed, and/or stolen, *resulting in loss*.
- **Threat**
  - Anything capable of *acting against an asset* in a manner that can cause harm.
- **Vulnerability**
  - A condition in which a *threat capability* (force) is greater than the ability to *resist that force*. Or: A *weakness* that can be *exploited*
- **Risk**
  - The *probable frequency* and probable *magnitude of future loss*.
  - Or: risk = likelihood * impact

An Introduction to Factor Analysis of Information Risk (FAIR), Jack A. Jones

A Little History…

Where can this go wrong?

Stuxnet – A Nuclear Refinement Plant Attack

- External actors gained access to the control bus from an unsecured, networked machine
- Control computers were secure...
  - ...but the system *as a whole* was not
  - This included procedures and security training for staff
- Once in the network, the malware quickly gained a foothold and spread
  - Any unsecured equipment was infected, making removal difficult
  - Hidden by a 'rootkit' the software then waited for additional software from external sources
- Under external control, the attackers were able to spoof valid signals, while causing damage at the same time!

Further reading: https://www.businessinsider.com/stuxnet-was-far-more-dangerous-than-previous-thought-2013-11

Where can this go wrong?

Stuxnet – A Nuclear Refinement Plant Attack

"An infected device, such as a USB key, would be sufficient to infect one computer on the target network. Stuxnet would then spread through the network using peer-to-peer methods."

- ScienceDirect

https://www.sciencedirect.com/topics/computer-science/stuxnet

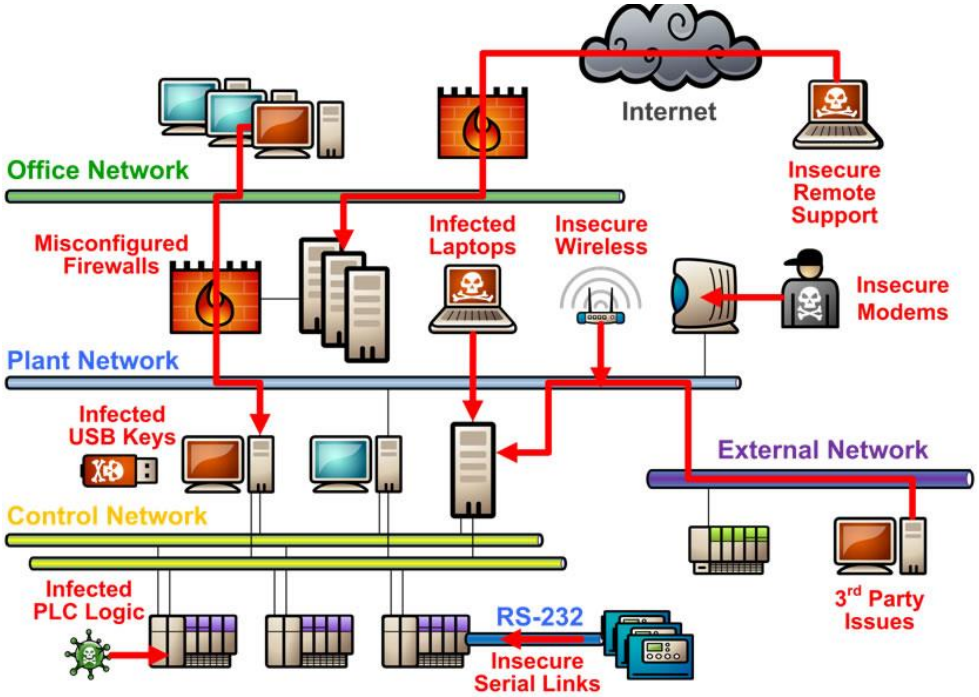


Image from: https://www.tofinosecurity.com/blog/air-gaps-won%E2%80%99t-stop-stuxnet%E2%80%99s-children

Where can this go wrong?

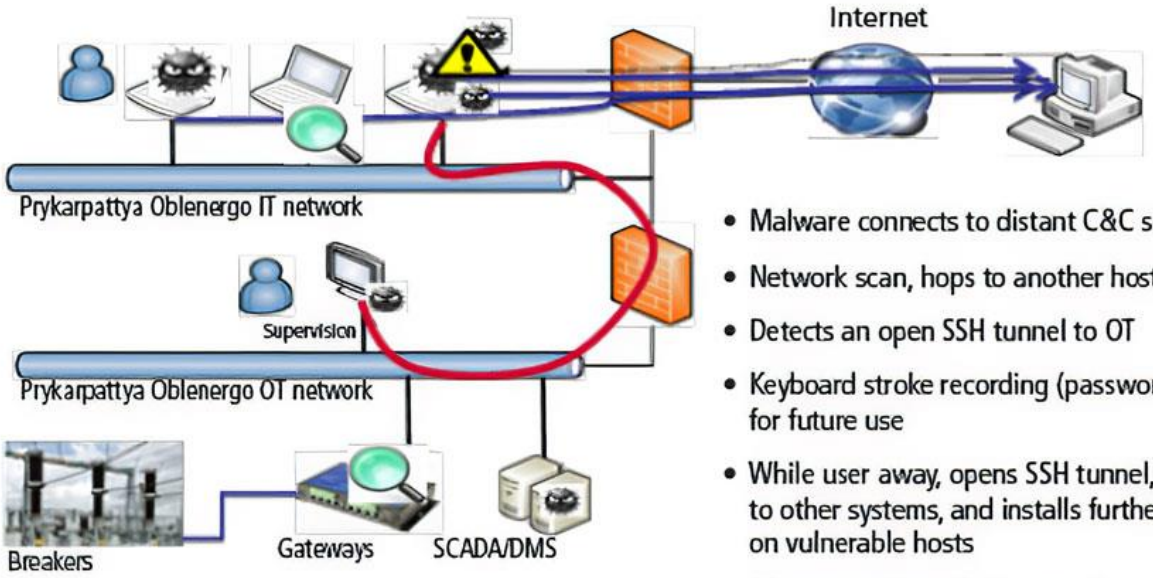
Ukraine Power Grid Hack

- Hackers compromised information systems of three energy distribution companies in Ukraine
- Only temporarily disruption to the electricity supply…
  - … but is the first known successful cyberattack on a power grid

- The cyberattack was complex:
  - Prior compromise of corporate networks using spear-phishing emails with BlackEnergy malware
  - Seizing SCADA under control, remotely switching substations off
  - Disabling/destroying IT infrastructure components (UPS, modems, RTUs, commutators)
  - Destruction of files stored on servers and workstations with the KillDisk malware
  - Denial-of-service attack on call-center to deny consumers up-to-date information on the blackout

Where can this go wrong?

Ukraine Power Grid Hack



- Malware connects to distant C&C server
- Network scan, hops to another host
- Detects an open SSH tunnel to OT
- Keyboard stroke recording (passwords, etc.) for future use
- While user away, opens SSH tunnel, connects to other systems, and installs further malware on vulnerable hosts
- Cleanup and installation on a low-profile persistent threat, ready for activation

Image from: https://blog.isa.org/lessons-learned-forensic-analysis-ukrainian-power-grid-cyberattack-malware

Where can this go wrong?

Ukraine Power Grid Hack

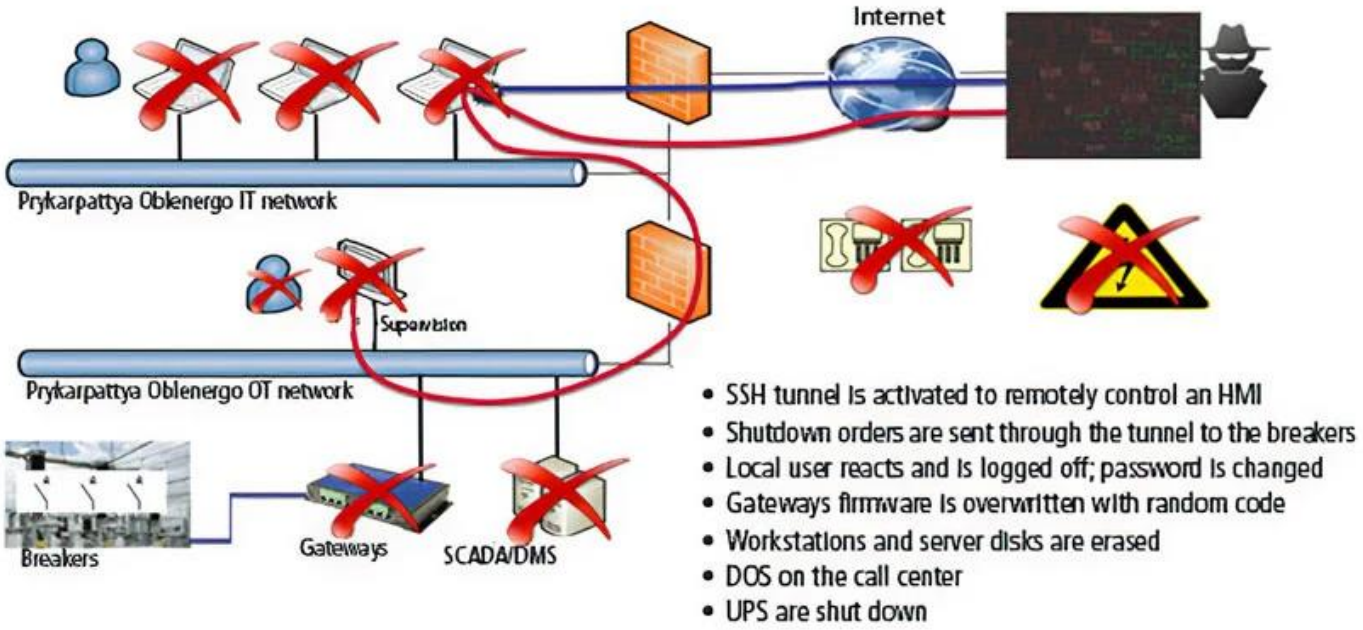


Image from: https://blog.isa.org/lessons-learned-forensic-analysis-ukrainian-power-grid-cyberattack-malware

How do we prevent this happening again?

Risk Analysis

Thinking Like an Attacker

- Who could be attacking our system?
  - Cyber Criminals
  - Insiders
  - Nation States
  - Hacktivists
  - Cyber-Fighters
  - Cyber-Terrorism
  - Script Kiddies

Risk Analysis

Security at *Every Layer*

- Security is a concern at every level in a system:
  - Business
  - Functional
  - Information
  - Communication
  - Component

- Far too much to cover in one session! So we're going to touch on 3 layers:
  - Component
  - Communication
  - Information

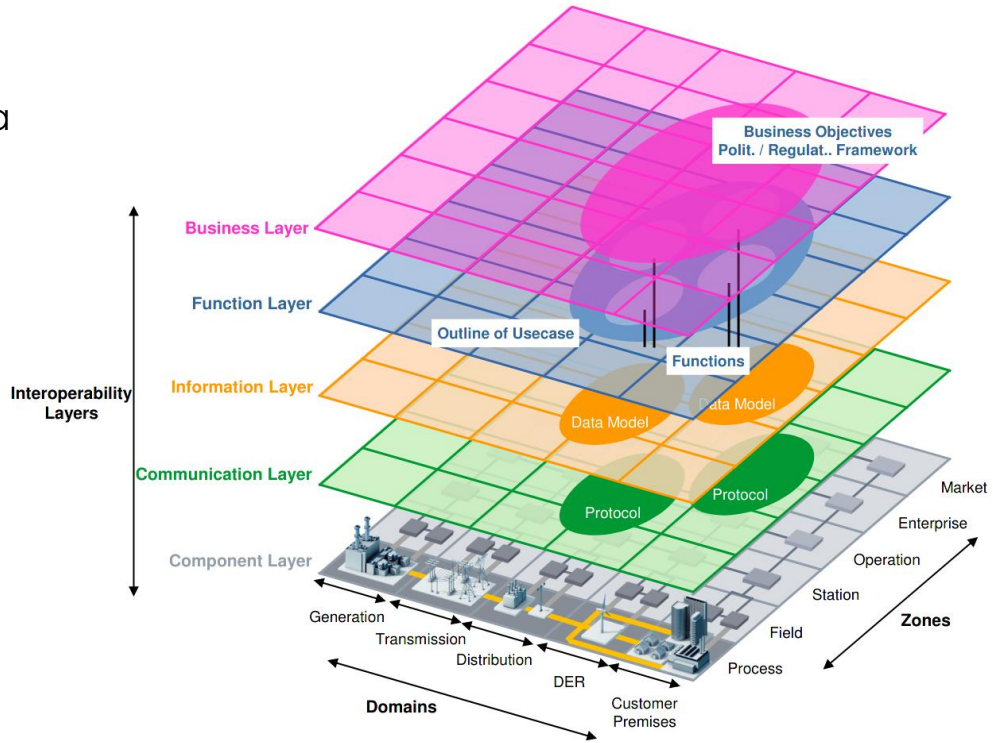


Image from: CEN-CENELEC-ETSI Smart Grid Coordination Group - Smart Grid Reference Architecture

Assessing a System

- Formal approaches do exist (for various levels of detail)

- Formalisms Example: RRA
  - Mozilla Rapid Risk Assessment: https://infosec.mozilla.org/guidelines/risk/rapid_risk_assessment

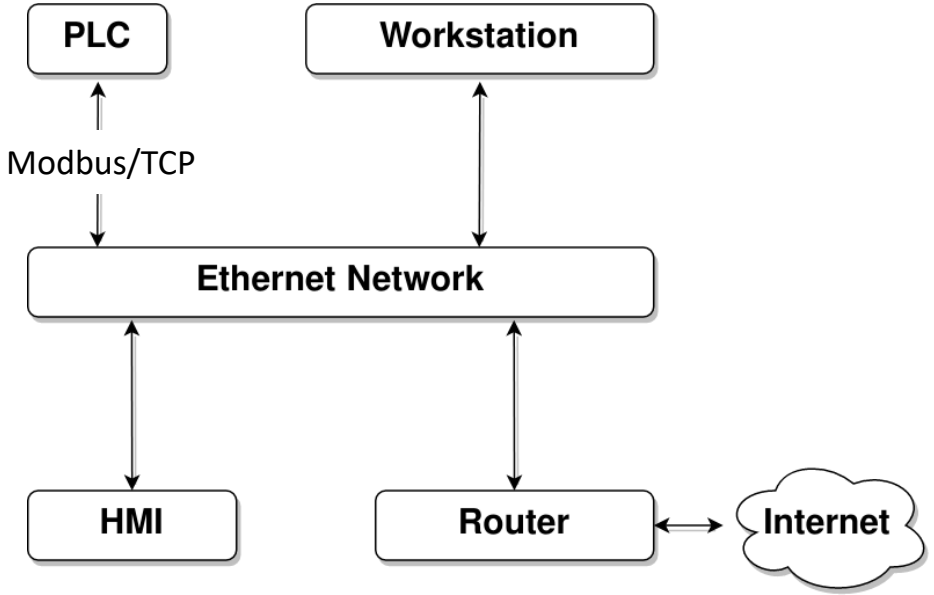
- But even without a full formal framework, we can cover many vulnerabilities with some simple steps.
  1. Treat every communication channel as if it were visible to an attacker
  2. Don't trust any data you don't directly produce
  3. Assume attackers know as much about the system as you

Some of my prior work was identifying such simple approaches for maximum effect in SME contexts: https://eprints.lancs.ac.uk/id/eprint/74598/4/SCC_2015_02_CS_Controls_Effectiveness.pdf

Component Security

A 'Toy' System

- Two PCs
  - An HMI
  - A workstation
- A router
  - For internet access
- A PLC
  - Controlled from the HMI
- All connected through an ethernet network switch

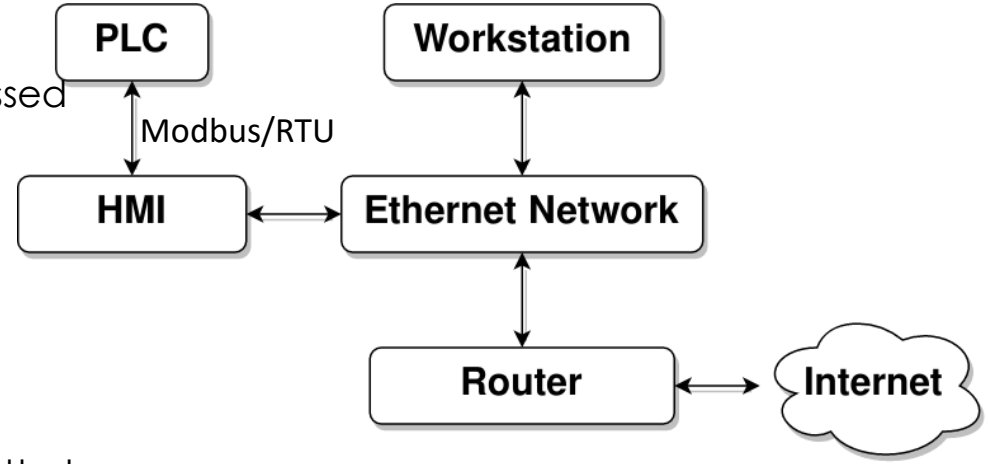


What is wrong here? How can we improve this situation?

Component Security

A 'Toy' System

- The PLC needs to only be accessed from a single PC, the HMI unit
- Isolate the PLC entirely from the network!
  - Changed to Modbus/RTU, perhaps over USB...

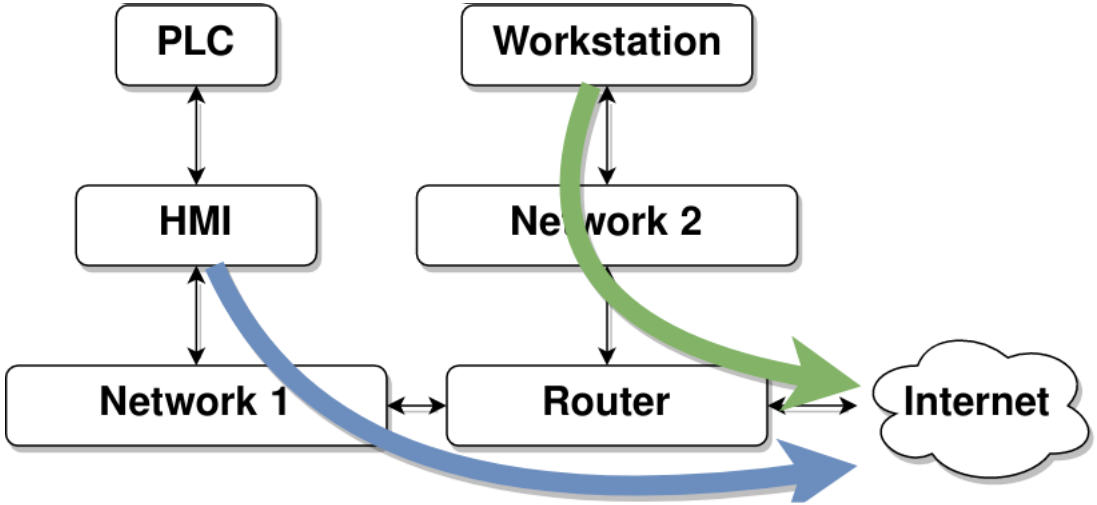


- Better... but we can do even better!
  - Is the HMI being used as a workstation too?
  - We still have a potential vulnerability from the workstation to the HMI!

Component Security

A 'Toy' System

- Keep the 'critical' away from the 'general' infrastructure
- Separate the network into two isolated sections
- This can be done much more simply than trying to plug any holes between the two PCs on an open network



Communication Security

- Touched on this slightly with the Toy System.
  - Moving the PLC to its own dedicated link to the HMI keeps the attack surface small

- But!
  - We actually moved from a potentially more secure connection (via TCP+SSL)
  - If the PLC is connected over RS485 (rather than USB), then this could prove to be a risk later if any other equipment is connected.
  - Must keep an eye on anything that could be used as a bridge to external networks!

Communication Security

- This can be easy if you have control of the whole network
  - LANs
  - Lab networks
  - Experimental sites

- Rather more difficult if you don't!
  - Internet connections
  - Wireless (WiFi, Zigbee, Bluetooth, etc.)
  - Cellular (3G, 4G, 5G, etc.)

- Let's explore these in reverse order...

Cellular Networking

- M2M connections often include network isolation
  - Devices can only talk to each other...
    - ... plus some specific addresses given access
- Cannot be relied upon alone!
  - Remember component security?
  - If an attacker gets hold of your devices, they have access to your entire network!
- This can be very hard to detect too!
- Can require changing deployed hardware! Might be very costly!

Wireless Networking

- M2M connections generally include some form of encryption
- WiFi, Zigbee and Bluetooth have 'handshakes' that perform an initial exchange to establish a secure link
  - Some of these technologies can be subverted at this stage!
  - Others can be inferred from easy-to-obtain information
  - WiFi routers used to use common terms, or MAC addresses as part of their keys, vastly limiting the possibilities the keys could be set to.
- Once again, however, relying on this alone is not enough
- Once an attacker gets access to any part of the network, they have access to *all* of the network

Internet Networking

- No isolation at all
- Requires that the client devices handle any encryption
- VPNs suffer from the same issues as M2M isolation
- Once into a VPN, you get access to everything (in general!)
- Exposure to any attacker in the world

Common Themes

- At best, you only get one layer of security!

- Access should not be used as authorisation
  - Just being on a network should not give you access to everything
  - Other forms of authorisation and encryption should also be used

- The security 'onion' :)
  - "Its layers all the way down, man..."

Information Security

- All based around minimising the possible exposure of ‘secrets’
  - Secrets are any password, key, or other credential
- Good practices around managing secrets can be key to stopping an attack

- As always – key reuse is a huge concern
  - Never use a single key to access multiple resources, or to access the same resource from multiple places
  - Once compromised the overall affects of the breach are magnified greatly

- This can quickly become cumbersome however, if you need to issue keys for *everything* each system does
  - Instead, consider per-device keys, which can easily be revoked if an attacker gains access to them

Information Security

- Indefinite keys are dangerous!
  - Any deployed keys or passwords should have a built-in lifetime
  - If valid forever, the keys remain a risk forever!

- A hierarchical structure for keys is beneficial
  - If a low level key is compromised, we can quickly re-issue this from a locally held mid-level key

Security Good Practices

- **Allowing Exceptions is Easier than Denial**
  - Avoid situations where you end up playing vulnerability whack-a-mole with your systems
  - Deny *all* access first, then allow just the expected connections through
    - Firewalls are the classic example of this
    - A good firewall should deny everything unknown, and only allow specific subsets of data through
- **Keep Systems Segmented**
  - Only allow devices required for operation onto a network
    - General purpose machines are *the most risky* devices, and should never be on any critical network connection
    - Consider placing 'gateways' between segments to allow functionality rather than connecting directly

Security Good Practices

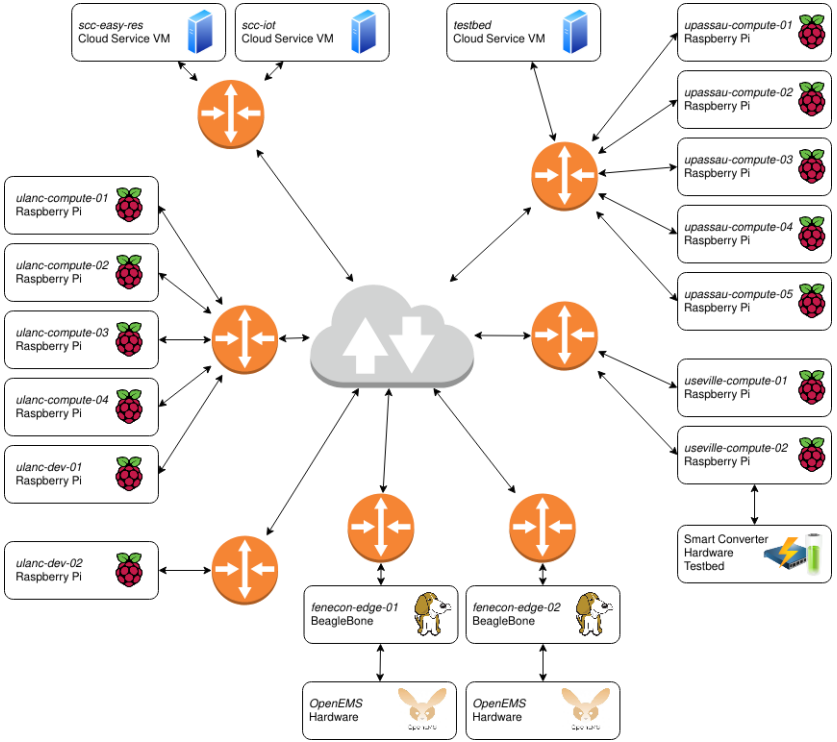
- **Don’t rely on any single mechanism to save you**
  - Security in depth
  - Wrap each access in ways that cause an attacker to halt

- **Avoid key reuse wherever possible**
  - Shared keys = shared risk
  - Minimise the effects of an attack

So how did we implement this ourselves?

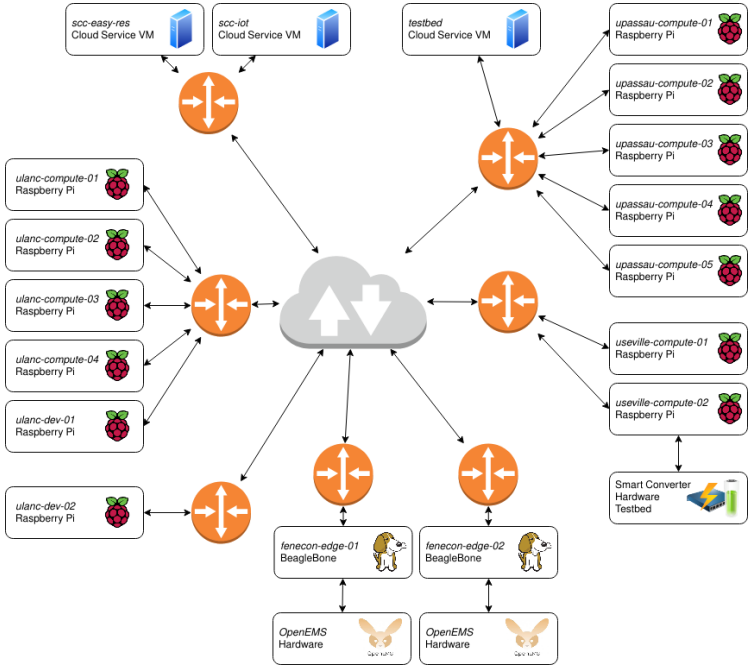
Security in the EASY-RES Testbed

The EASY-RES Testbed



The EASY-RES Testbed

- Built across 5 sites
- Lancaster University
  - Cloud VM Cluster
  - Physical Compute Cluster
  - Development Nodes
- University of Passau
  - Cloud VM
  - Physical Compute Cluster
- FENECON GmbH
  - BeagleBone-based Access Nodes



Component Security

- Edge nodes are connected to EMS and test hardware
  - Nodes act as 'gateways' to this hardware
  - No direct access from other devices to the EMSs
- Individual nodes have unique access keys
  - No globally shared secrets!
  - A compromised node can be evicted from the network easily
- Software is only deployed from a central management system
  - No direct access to 'external' components
  - Passau cannot directly access Lancaster equipment

Communication Security

- A Wireguard VPN spans the entire network
  - No segregation :(
  - Done this way for simplicity, as this is a research network after all!
  - Real deployments would have additional network separation
- Connections between services are controlled by overlay networks
  - Software needs to declare which connections it needs!
  - You have seen some of this in the previous slide deck
  - All deployable resources can be limited by the central control system (Nomad)

Information Security

- Any keys or passwords for use with deployments are stored in Vault
- Vault has access control to prevent external programs gaining access
- Locally nodes can have deployment details stored in virtual storage
  - In our testbed these *are* visible to someone with direct access to a node, but again, this is a research network, so we make some concessions...
- Program images have keys supplied to them via Nomad at *deployment* time
  - Gaining access to the software doesn't get you access to the keys

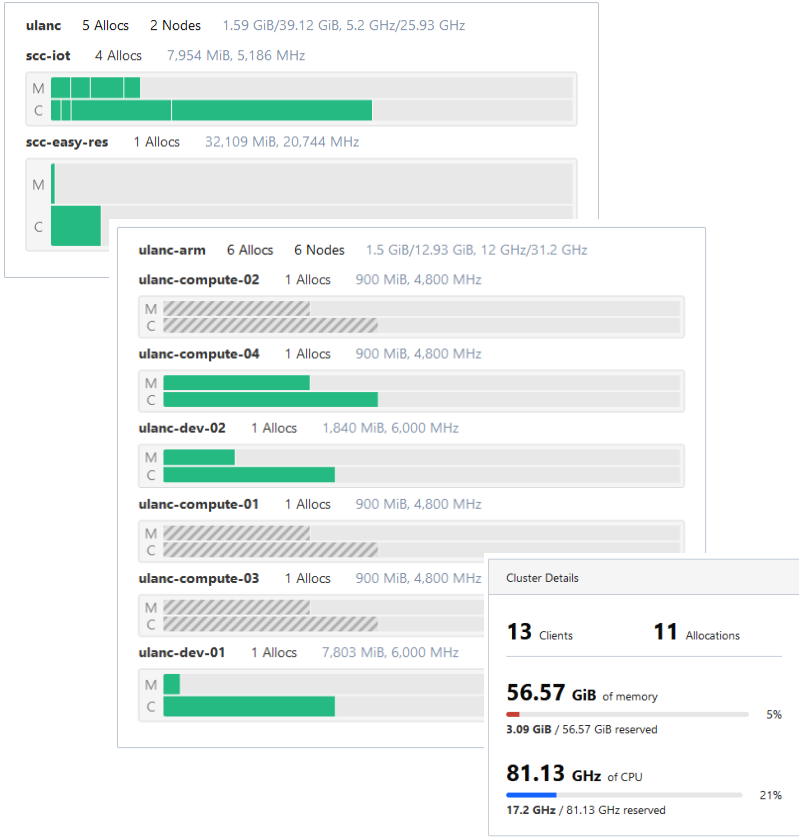
Information Security

- Users only have access to a limited view of the system
- Deployments must obey rules defined by both the deployment system (Nomad) and the edge nodes themselves
  - Devices can veto specific capabilities, refusing to run certain software.
- Containers are used to sandbox software
  - Prevent access to everything, then allow only specific things through
    - May also be useful for software security audits!

Security in the EASY-RES Testbed

Information Security


```
{"Affinities":null,"AllAtOnce":false,"Constraints":null,"ConsulNamespace":"","ConsulToken":""
,"CreateIndex":130011,"Datacenters":["ulanc"],"Dispatched":false,"ID":"thing-services"
,"JobModifyIndex":134200,"Meta":null,"ModifyIndex":134207,"Multiregion":null,"Name":"thing
-services","Namespace":"default","NomadTokenID":"","ParameterizedJob":null,"ParentID":""
,"Payload":null,"Periodic":null,"Priority":50,"Region":"global","Spreads":null,"Stable":true
,"Status":"running","StatusDescription":"","Stop":false,"SubmitTime":1625754891934085600
,"TaskGroups":[{"Affinities":null,"Constraints":null,"Consul":{"Namespace":""},"Count":1
,"EphemeralDisk":{"Migrate":false,"SizeMB":300,"Sticky":false},"Meta":null,"Migrate"
:{"HealthCheck":"checks","HealthyDeadline":300000000000,"MaxParallel":1,"MinHealthyTime"
:10000000000},"Name":"cloud","Networks":[{"CIDR":"","DNS":null,"Device":"","DynamicPorts":null
,"IP":"","MBits":0,"Mode":"","ReservedPorts":[{"HostNetwork":"default","Label":"http","To":0
,"Value":8081}]}],"ReschedulePolicy":{"Attempts":0,"Delay":30000000000,"DelayFunction"
:"exponential","Interval":0,"MaxDelay":3600000000000,"Unlimited":true},"RestartPolicy"
:{"Attempts":2,"Delay":15000000000,"Interval":1800000000000,"Mode":"fail"},"Scaling":null
,"Services":[{"AddressMode":"auto","CanaryMeta":null,"CanaryTags":null,"Checks":null,"Connect"
:null,"EnableTagOverride":false,"Meta":null,"Name":"thing-directory","Namespace":"default"
,"OnUpdate":"require_healthy","PortLabel":"http","Tags":null,"TaskName":""}],"ShutdownDelay":null
,"Spreads":null,"StopAfterClientDisconnect":null,"Tasks":[{"Affinities":null,"Artifacts":null
,"CSIPluginConfig":null,"Config":{"ports":["http"],"image":"linksmart/td:latest"},"Constraints"
:null,"DispatchPayload":null,"Driver":"docker","Env":null,"KillSignal":"","KillTimeout"
:5000000000,"Kind":"","Leader":false,"Lifecycle":null,"LogConfig":{"MaxFileSizeMB":10,"MaxFiles"
:10},"Meta":null,"Name":"directory","Resources":{"CPU":1000,"Cores":0,"Devices":null,"DiskMB":0
,"IOPS":0,"MemoryMB":512,"MemoryMaxMB":0,"Networks":null},"RestartPolicy":{"Attempts":2,"Delay"
:15000000000,"Interval":1800000000000,"Mode":"fail"},"ScalingPolicies":null,"Services":null
,"ShutdownDelay":0,"Templates":null,"User":"","Vault":null,"VolumeMounts":null}],"Update"
:{"AutoPromote":false,"AutoRevert":false,"Canary":0,"HealthCheck":"checks","HealthyDeadline"
:300000000000,"MaxParallel":1,"MinHealthyTime":10000000000,"ProgressDeadline":600000000000
,"Stagger":30000000000},"Volumes":null}],"Type":"service","Update":{"AutoPromote":false
,"AutoRevert":false,"Canary":0,"HealthCheck":"","HealthyDeadline":0,"MaxParallel":1
,"MinHealthyTime":0,"ProgressDeadline":0,"Stagger":30000000000},"VaultNamespace":"","VaultToken"
:"","Version":26}
```

Item 1: Text with bullet points

Summary

- By this stage you should understand:
    - ... how to look at a system and spot vulnerable points
    - ... how to isolate components of a system to add physical barriers to access
    - ... to wrap communications in layers of security
    - … how to identify each of the first three security layers
        - And what they cover!

Open Question & Answers

The EASY-RES Consortium

The Consortium

ARISTOTLE UNIVERSITY OF THESSALONIKI

UNIVERSIDAD DE SEVILLA

UNIVERSITÄT PASSAU

Lancaster University

TU Delft

stadtwerk haßfurt

ΑΔΜΗΕ
ΑΝΕΞΑΡΤΗΤΟΣ ΔΙΑΧΕΙΡΙΣΤΗΣ ΜΕΤΑΦΟΡΑΣ ΗΛΕΚΤΡΙΚΗΣ ΕΝΕΡΓΕΙΑΣ

elektro gorenjska

Stadtwerke LANDAU a.d.ISAR

fenecon
ENERGY ENGINEERING

ZD.B ZENTRUM DIGITALISIERUNG. BAYERN

This project has received funding from the European Union´s Horizon 2020 Programme for research and innovation under Grant Agreement no 764090.

Thank you!

Dr John Vidler & Dr Martin Bor

Affiliation: Lancaster University
Phone: +44 (0) 1524 510376
E-Mail: j.vidler@lancaster.ac.uk, m.bor@lancaster.ac.uk
EASY-RES website: http://www.easyres-project.eu/