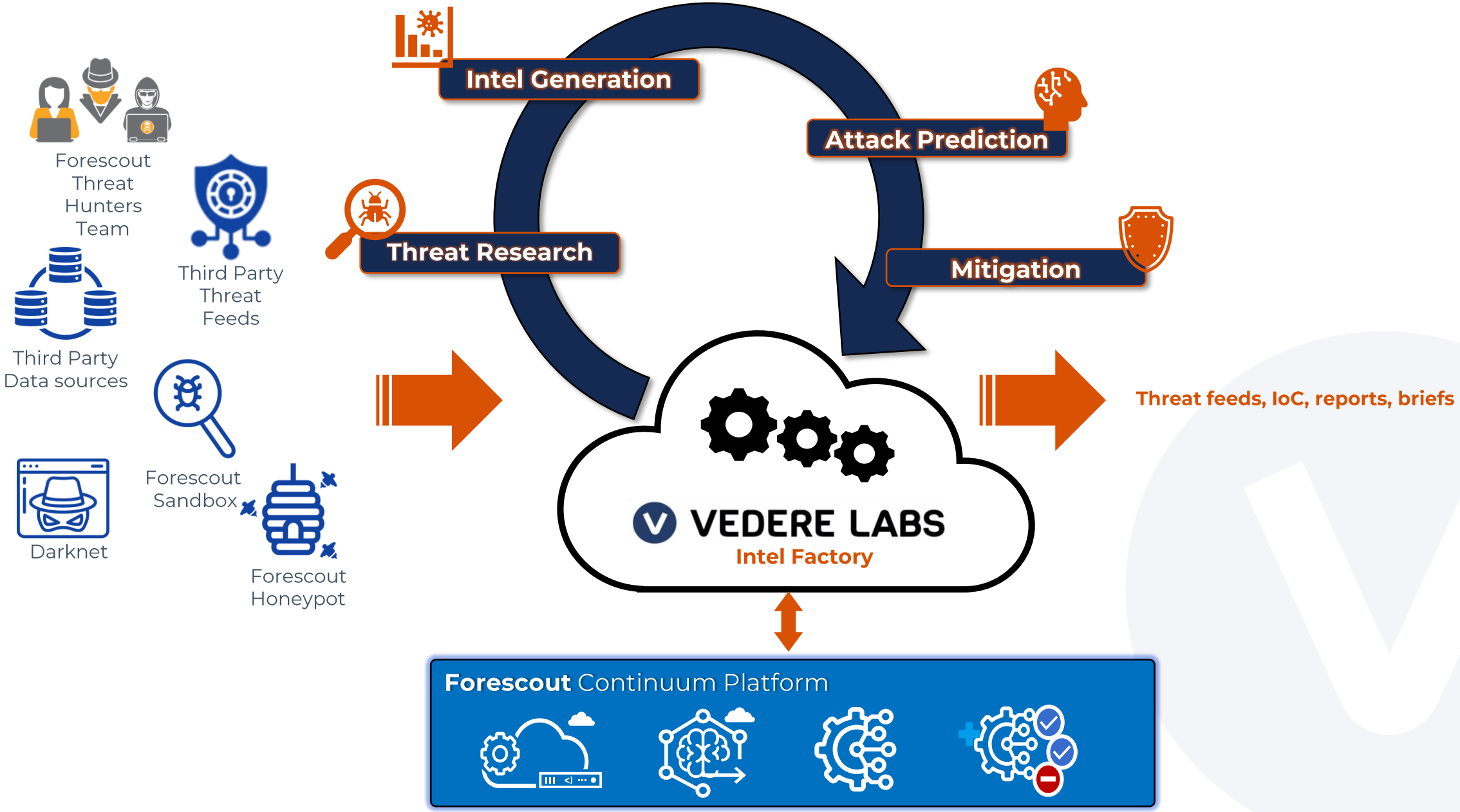
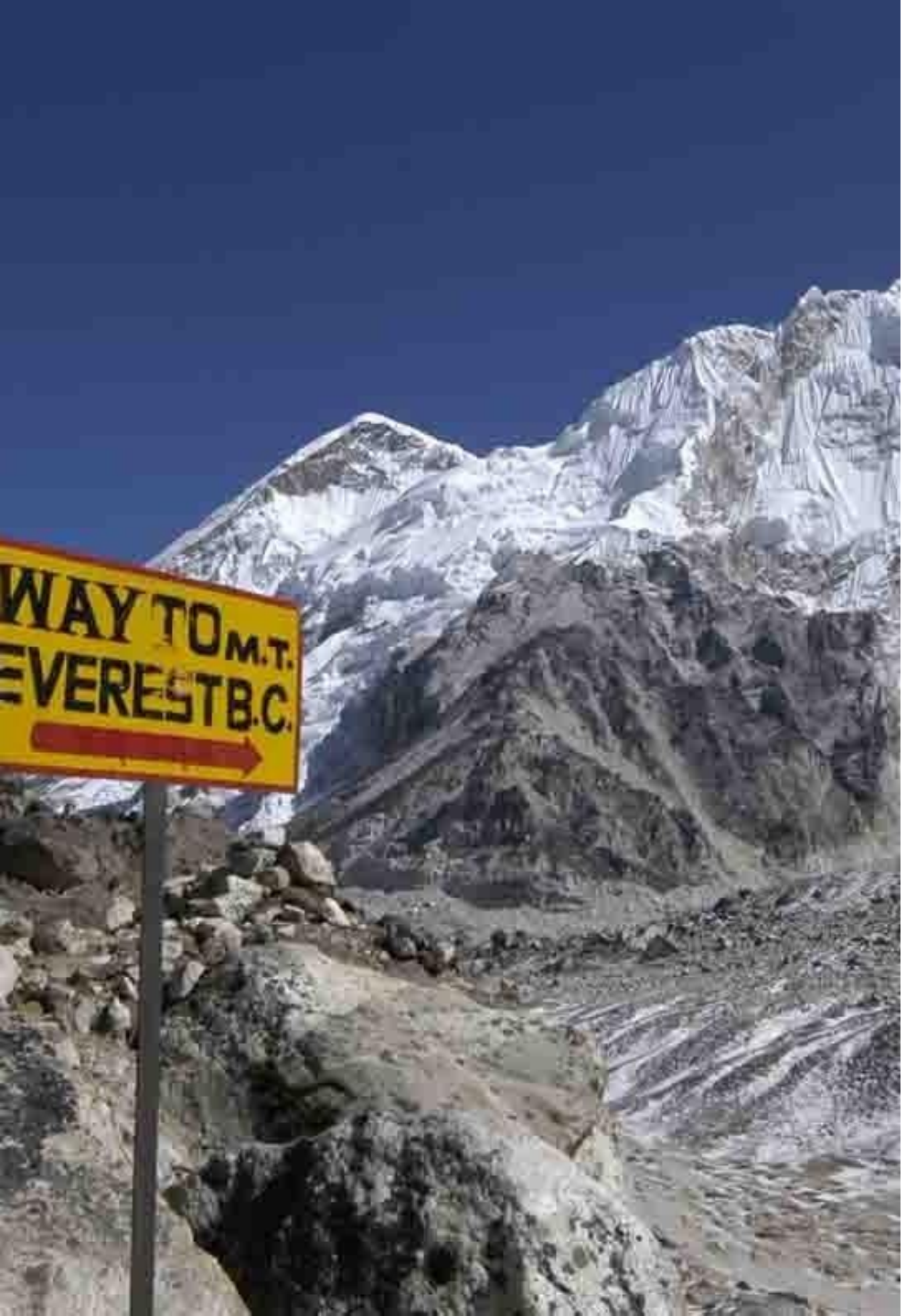
OT:ICEFALL

Revisiting a decade of insecure-by-design practices in OT

Jos Wetzels
Security Researcher, Forescout

Vedere Labs: The Threat Intelligence Arm of Forescout





The long climb ahead

- 10+ years ago, Digital Bond's Project Basecamp[1], modeled after Firesheep, showed pervasiveness of **insecure-by-design** in ICS equipment
- Lack of basic security controls → historical deployment in trusted, air-gapped networks
- Advent of standards-driven security efforts
  - IEC 62443, NERC CIP, NIST SP 800-82, etc.
- OT:ICEFALL[2] (after next stop on Mt. Everest) aims to be quantitative **checkup of progress** made in **active production environments**
  - Evaluated systems selected based on customer asks to look into & support specific OT systems

[1] https://github.com/digitalbond/Basecamp
[2] https://www.forescout.com/resources/ot-icefall-report/

Insecure-by-design is a well-known issue, why revisit it?

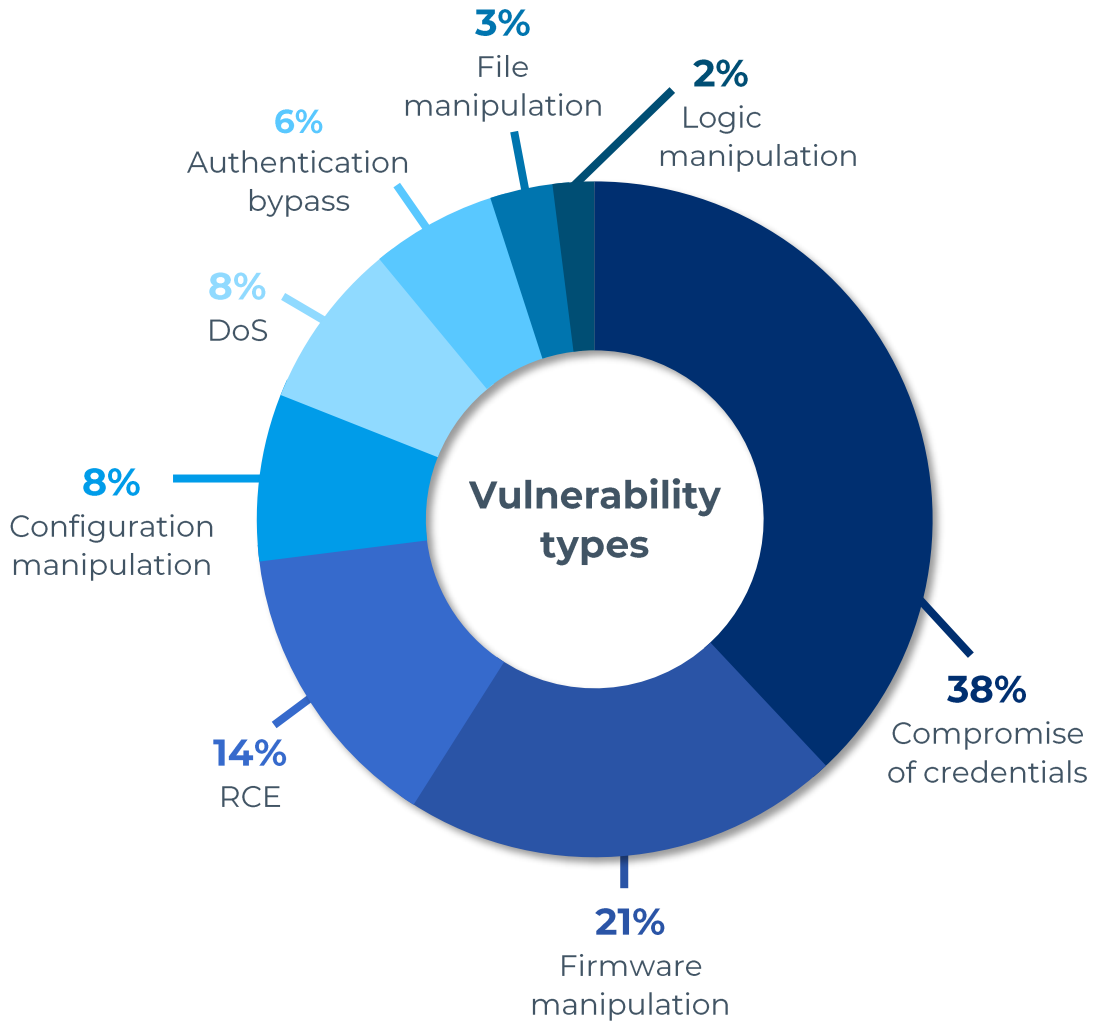
- Risk management is complicated by opacity
  - Are we making progress in install base?
  - What's under the hood of those security controls?
- Cannot assume every proprietary system is & remains equally broken
  - Security mechanisms are sometimes retrofitted
  - Authentication implementations change over time
- Not enough to know thing is 'insecure', need to know in what way
  - Big difference between changing setpoint and RCE
  - Need to justify compensating controls
- Can't make informed decisions based on speculation

Overview

56 CVEs affecting 10+ vendors

Vendor	Model	Type
Bently Nevada	3700 / TDI	Condition Monitoring
Emerson	DeltaV	DCS
Emerson	Ovation	DCS
Emerson	OpenBSI	Engineering Workstation
Emerson	ControlWave, ROC	RTU
Emerson	FANUC / PACsystems	PLC
Honeywell	Trend IQ	Building Controller
Honeywell	Safety Manager / FSC	SIS
Honeywell	Experion LX	DCS
Honeywell	ControlEdge	RTU
Honeywell	Saia Burgess PCD	PLC
JTEKT	Toyopuc	PLC
Motorola	MOSCAD IP Gateway	Gateway
Motorola	MDLC	Protocol
Motorola	ACE1000	RTU
Motorola	MOSCAD Toolbox	Engineering Workstation
Omron	SYSMAC Cx/Nx	PLC
Phoenix Contact	ProConOS/eCLR	Runtime
Siemens	WinCC OA	SCADA
Yokogawa	STARDOM	PLC

Full overview: https://www.forescout.com/research-labs/ot-icefall/

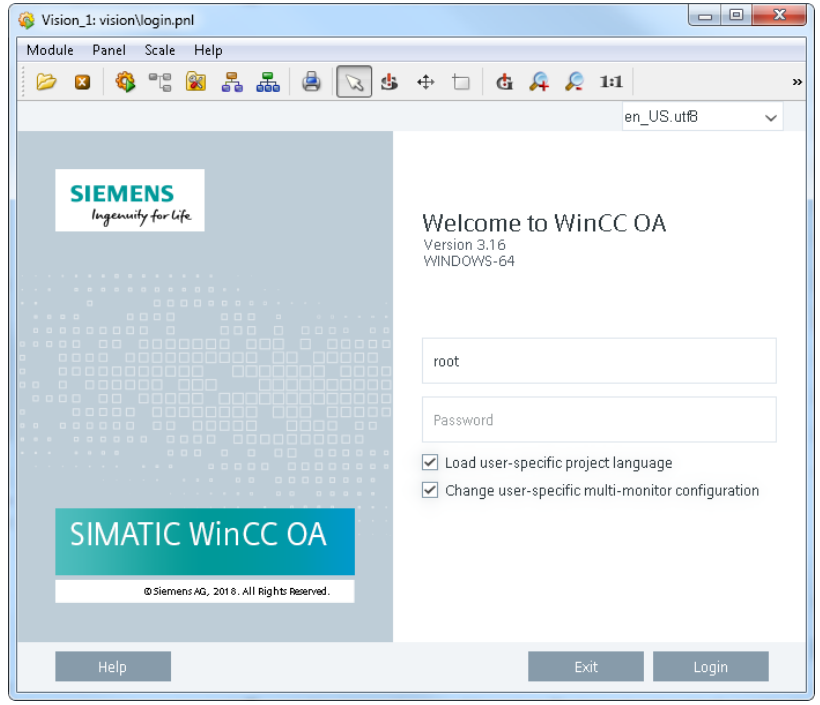


Disclosure

- Disclosed issues to CISA/vendors 90+ days ahead of publication
  - Some vendors started in-depth investigation very late
  - Some issues and responses still in disclosure
  - Some vendors wanted details to be restricted to product bulletins
  - In some cases, invigorated secure protocol development efforts
- Affected versions & detailed mitigations in CISA / vendor advisories
  - https://www.cisa.gov/uscert/ncas/current-activity/2022/06/22/cisa-releases-security-advisories-related-oticefall-insecure
- Will not disclose full technical details

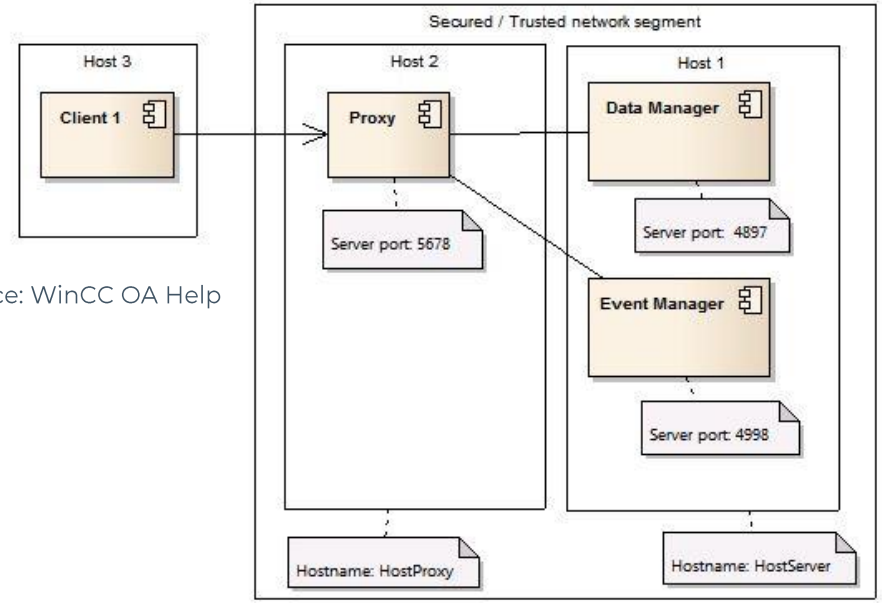
Broken Authentication Schemes & Improper Fixes

Siemens WinCC OA SCADA
(CVE-2022-33139)



- Operator UI talks to proxy
  - Proxy wraps proprietary PVSS protocol in TLS

- Authentication Modes
  - Client-Side Authentication (CSA): default pre-3.17
    - Preferred for SSO integration
  - Server-Side Authentication (SSA): default 3.17+
  - Kerberos Authentication



Source: WinCC OA Help

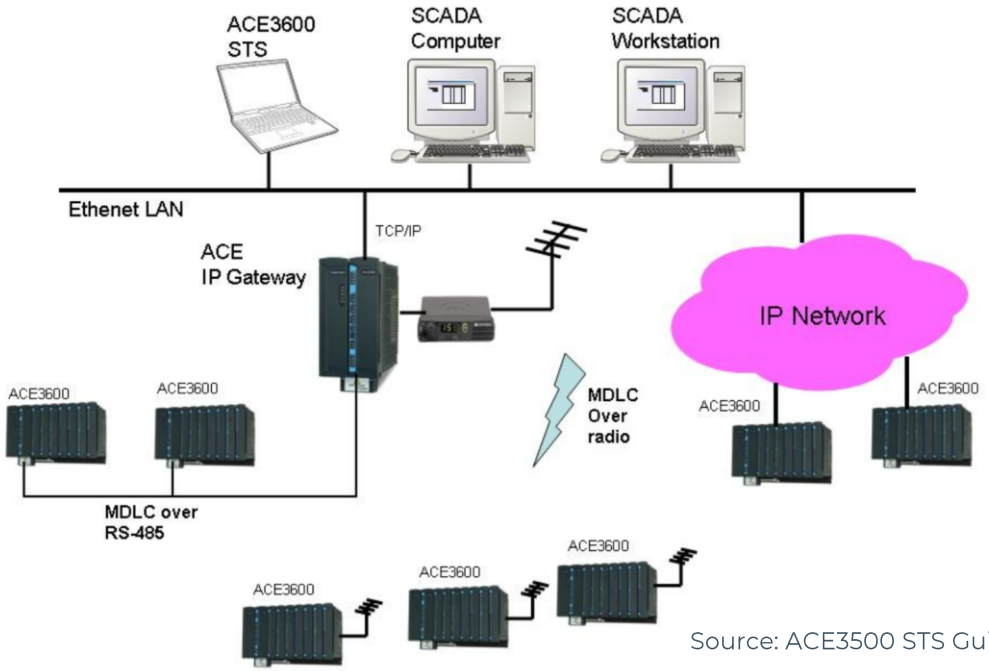
- CSA stores creds as database points
- Client sends PVSS request for creds, validates locally (e.g. against AD)
  - Attacker can just write malicious client, no need for auth

Motorola MDLC
(CVE-2022-30273)

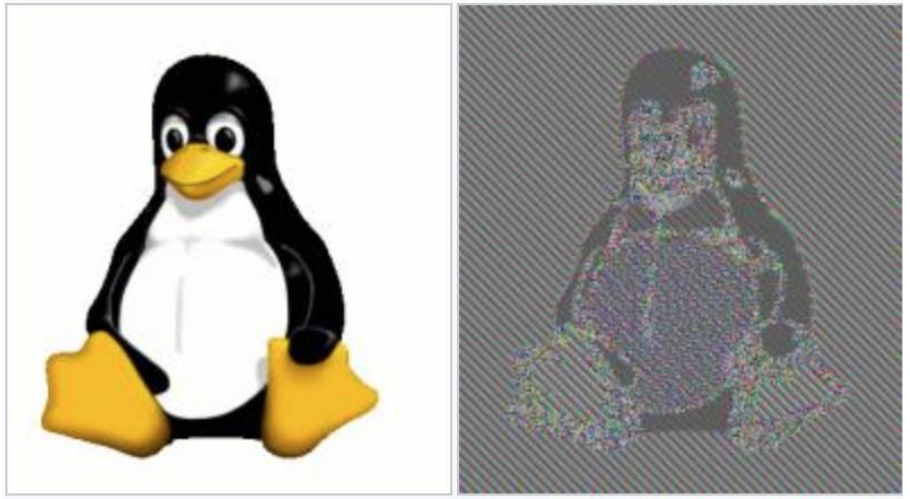
- SCADA ↔ RTU WAN protocol
  - (over IP, serial, radio, microwave, etc.)

- PSK-based Encryption Modes
  - AES256: default in newer RTUS (e.g. ACE3600)
  - TEA-ECB: default in older RTUs (e.g. MOSCAD/ACE1000)

    Supported by ACE3600 until 2022 (backward compatibility/mixed networks)
    - Notorious block cipher mode of operation
    - Known plaintext attacks, block-swapping, etc.

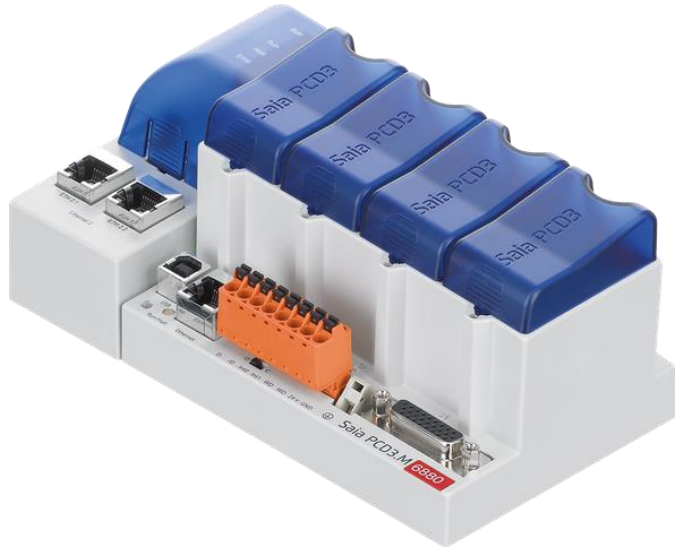


Source: ACE3500 STS Guide



Original image | Encrypted using ECB mode

Saia Burgess PG5 PCD PLC
(CVE-2022-30320)



- Uses S-Bus protocol (5050/UDP)
  - Master-slave protocol, historically RS485
- Password auth for engineering operations
  - S-Bus write to static address
  - 32-bit 'hash' derived from CRC-16/XMODEM over password without nonce
- Trivially insecure
  - Collisions
  - Replay
  - UDP with MAC/IP whitelist

```
public static uint PasswordToUInt32(string password)
{
  ushort crc1 = 0;
  foreach (char c in password)
  {
    if (!char.IsDigit(c) && !char.IsUpper(c))
      return 0;
    crc1 = SaiaLib.Crc16(crc1, Convert.ToByte(c));
  }
  ushort crc2 = crc1;
  foreach (char c in password)
  {
    if (!char.IsDigit(c) && !char.IsUpper(c))
      return 0;
    crc2 = SaiaLib.Crc16(crc2, Convert.ToByte(c));
  }
  return ((uint) crc2 << 16) + (uint) crc1;
}
```

Emerson ControlWave PLC/RTU
(CVE-2022-29954, CVE-2022-29955, CVE-2022-29956)

- Uses BSAP/IP protocol (1234/UDP)
  - Password auth for engineering operations
  - But: UDP with MAC/IP whitelist

- Authentication Modes
  - Simple (legacy): 1-6 character plaintext password
  - Secure (legacy): PLC sends 8-bit key K, EWS responds with E(pass, K)
  - Secure 2 (undocumented, current): PLC sends 64-bit key K, EWS responds with E(pass, K)

- Bad design, 3 times over
  - Can just decrypt the credentials
  - Fundamental misunderstanding of challenge-response

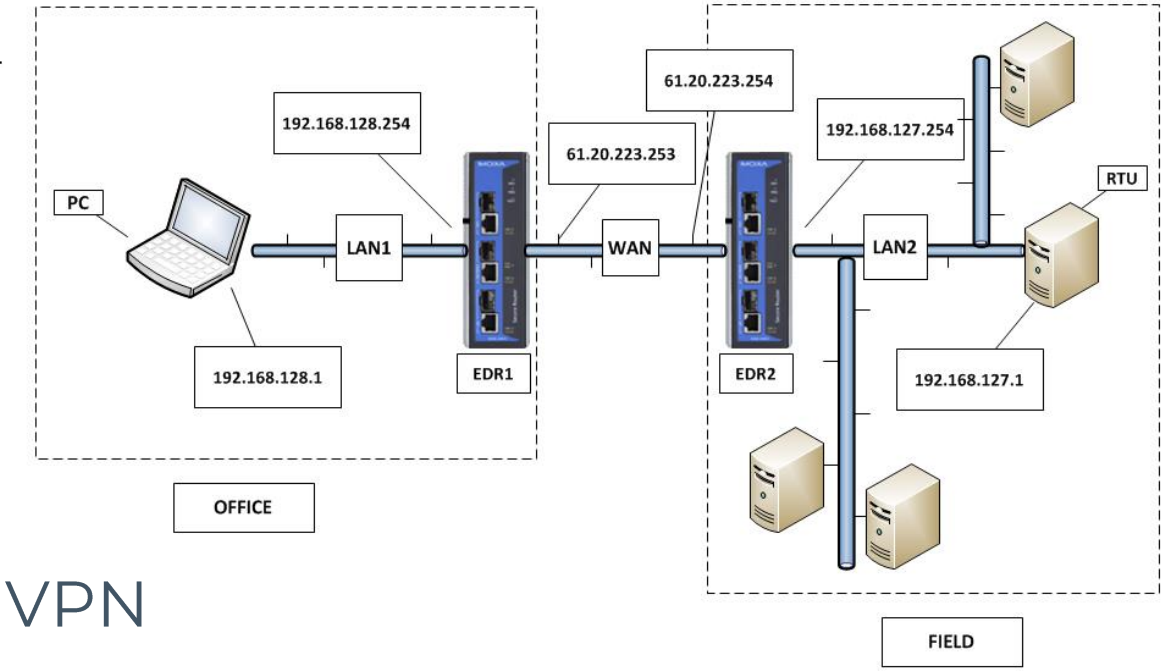
“*Not a vulnerability*” according to Emerson

- They feel this is adequately addressed by ControlWave manual
  - Basically: install VPN + firewall
  - “*Enhanced security*” implies existing controls offer “*basic security*”

1.8 Secure Gateway

For enhanced data security when using an IP/Ethernet connection, Emerson Remote Automation Solutions recommends adding an industrial router with VPN and firewall security. Recommended solutions include the MOXA EDR-810, the Hirschman Eagle One, or the Phoenix mGuard rs4000 (or equivalents). An example of how to install one of these devices to the RTU can be found in the Emerson Remote Automation Solutions *MOXA® Industrial Secure Router Installation Guide* (part number D301766X012). For further information, contact your Local Business Partner or the individual vendor’s website.

Source: ControlWave Micro Instruction Manual



- Reference recommends site-to-site VPN
  - No protection at site level

Source: MOXA Industrial Secure Router Installation Guide (D301766X012)

Yokogawa STARDOM PLC
(CVE-2022-30997)

- Hardcoded credentials for Telnet maintenance interface
  - Duplex controllers only up to R4.31
- Multiple prior CVEs for hardcoded creds on *same* interface
  - CVE-2018-10592, CVE-2018-17896
- Indicates bug-fixing is not followed up with variant-hunting

```
strcpy(password, [redacted]);
password[9] = 0;
*(_WORD *)&password[10] = 0;
if ( loginUserVerify([redacted] password) )
{
  if ( !dword_593E58 && DuoLoginEncrypt(password, &s_encrypted) )
```

Emerson DeltaV DCS controllers
(CVE-2022-29962, CVE-2022-29963, CVE-2022-29964, CVE-2022-29965)

- "Read-only Telnet" (18550/TCP) hardcoded creds
  - S/P-controllers, CIOC, EIOC up to at least v14.3.1
  - Other hardcoded reported for "*disabled FTP/SSH*" but could be used for LPE
- Maintenance Telnet + shell access (23/TCP) insecure auth algorithm
  - M/S/P-controllers, SIS up to at least v14.3.1.7283
    - Silently patched some point after
  - Homebrew algorithm without secret, using predictable seed < 16 bits
- Not first time DeltaV suffered from these issues
  - CVE-2014-2350 (hardcoded creds on Telnet 706/TCP)
  - Again shows bug-fixing without subsequent variant-hunting

OT Product Security Certification

Vulnerable products are often certified

74%

of the product families affected by the found vulnerabilities have some form of security certification

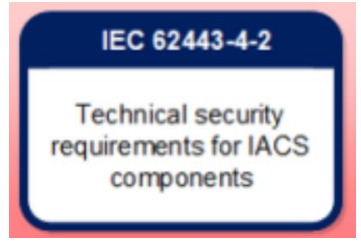
Factors contributing to this problem include:

- (Re)certification effort
- Limited targets for evaluations
- Opaque security definitions
- Focus on **functional testing**

Certifications among affected product families

Advisories serve as reference for cert lab auditors without SME knowledge

Example: IEC-62443-4-2



- Even at SL1 we need to meet CR 1.1, CR 3.4, EDR 2.4
  - How do we rhyme this with ubiquitous unauthenticated interfaces, logic downloads, software tampering, etc.?

5.3 CR 1.1 – Human user identification and authentication

5.3.1 Requirement

Components shall provide the capability to identify and authenticate all human users according to IEC 62443-3-3 SR 1.1 on all interfaces capable of human user access. This capability shall enforce such identification and authentication on all interfaces that provide human user access to the component to support segregation of duties and least privilege in accordance with applicable security policies and procedures. This capability may be provided locally by the component or by integration into a system level identification and authentication system.

13.2 EDR 2.4 – Mobile code

13.2.1 Requirement

In the event that an embedded device utilizes mobile code technologies, the embedded device shall provide the capability to enforce a security policy for the usage of mobile code technologies. The security policy shall allow, at a minimum, the following actions for each mobile code technology used on the embedded device:

a) Control execution of mobile code;

b) Control which users (human, software process, or device) are allowed to upload mobile code to the device;

c) Control the execution of mobile code based on the results of an integrity check prior to the code being executed.

7.6 CR 3.4 – Software and information integrity

7.6.1 Requirement

Components shall provide the capability to perform or support integrity checks on software, configuration and other information as well as the recording and reporting of the results of these checks or be integrated into a system that can perform or support integrity checks.

7.6.2 Rationale and supplemental guidance

Integrity verification methods are employed to detect, record, report and protect against software and information tampering that may occur if other protection mechanisms (such as authorization enforcement) have been circumvented. Components should employ formal or recommended integrity mechanisms (such as cryptographic hashes). For example, such

Example: IEC-62443-4-1

IEC 62443-4-1

Secure product development lifecycle requirements

10.4 DM-3: Assessing security-related issues

10.4.1 Requirement

A process shall be employed for analysing security-related issues in the product to include:

a) assessing their impact with respect to:
   1) the actual security context in which they were discovered;
   2) the product's security context (see Clause 6); and
   3) the product's defense in depth strategy (see Clause 7);

b) severity as defined by a vulnerability scoring system (for example, CVSS);

c) identifying all other products/product versions containing the security-related issue (if any);

d) identifying the root causes of the issue; and

e) identifying related security issues.

EMERSON	Emerson Automation Solutions	Secure Product Development Process	Austin, TX, USA	SDLA 2.0.0
EMERSON	Emerson Automation Solutions	Secure Product Development Process	Manila, Philippines	SDLA 2.0.0
YOKOGAWA	Yokogawa Electric Corporation	Secure Development Life Cycle	Musashino, Tokyo, Japan	SDLA 2.0.0
YOKOGAWA	Yokogawa Electric Corporation	Secure Development Life Cycle	Singapore	SDLA 3.0.0

https://isasecure.org/en-US/End-Users/IEC-62443-4-1-Certified-Development-Organizations

Example: IEC-62443-4-1, GE Achilles ACC

IEC 62443-4-1

Secure product development lifecycle requirements

NUCLEUS RTOS

9.4 SVV-3: Vulnerability testing

9.4.1 Requirement

A process shall be employed for performing tests that focus on identifying and characterizing potential security vulnerabilities in the product. Known vulnerability testing shall be based upon, at a minimum, recent contents of an established, industry-recognized, public source for known vulnerabilities. Testing shall include:

a) abuse case or malformed or unexpected input testing focused on uncovering security issues. This shall include manual or automated abuse case testing and specialized types of abuse case testing on all external interfaces and protocols for which tools exist. Examples include fuzz testing and network traffic load testing and capacity testing;

Achilles Grammars

Achilles Grammars test for protocol boundary conditions in the device communications. They systematically iterate over each field and combinations of fields to produce repeatable, quantifiable tests of the common types of implementation errors.

Achilles Grammars send invalid, malformed or unexpected packets to the Device Under Test (DUT) to test for vulnerabilities in specific layers of the protocol stack.

Product (ACC SL2)	TCP/IP stack vulns
Nucleus RTOS	NUCLEUS:13 NAME:WRECK
VxWorks 7 RTOS	Urgent/11 NAME:WRECK
Siemens SENTRON PAC4200	Amnesia:33
Schneider ATV6000	Ripple20
Rockwell ControlLogix	Urgent/11 NAME:WRECK

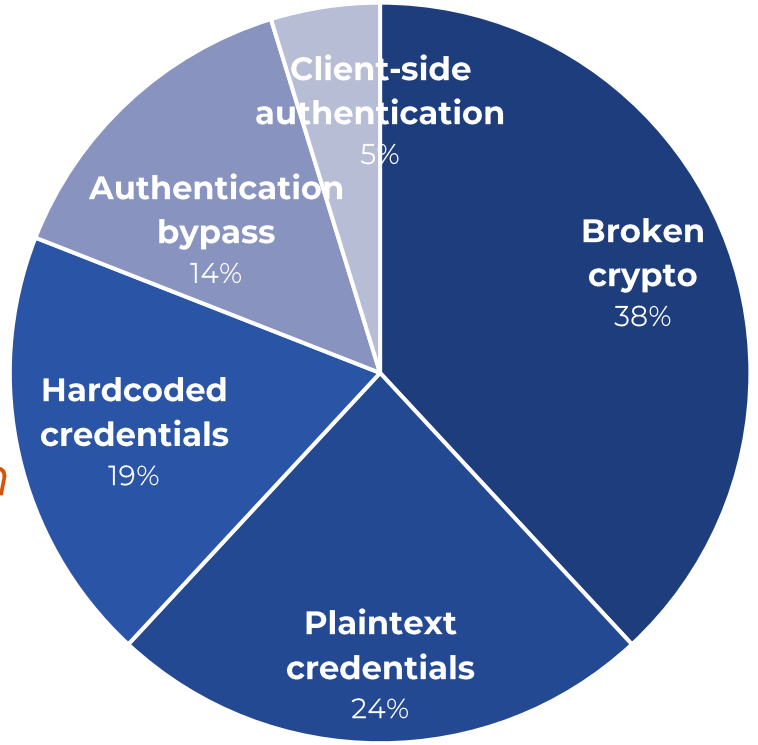
https://www.ge.com/digital/applications/achilles-communications-certified-products, https://www.ge.com/digital/sites/default/files/download_assets/achilles_test_platform.pdf

When is something ‘secure-by-design’?

- Subpar controls → false sense of security which is worse than clear sense of insecurity
  - 22 CVEs in OT:ICEFALL related to broken auth
  - 28 CVEs in prior work with similar root causes

- What to expect of product security standards?
  - High level requirement, little guidance on *robust design*
  - Only functional testing is borderline useless
  - Who tests the testers? What’s the QA metric on auditor expertise? Fuzzing coverage? Etc.
  - Why do certified SDLCs remain immature?

- Secure-by-design is not enough
  - Need secure-by-default
    (no more “*there’s hardening guidance in the manual*”)



Impact & Nuance

Nuance: Supply Chains & Collisions

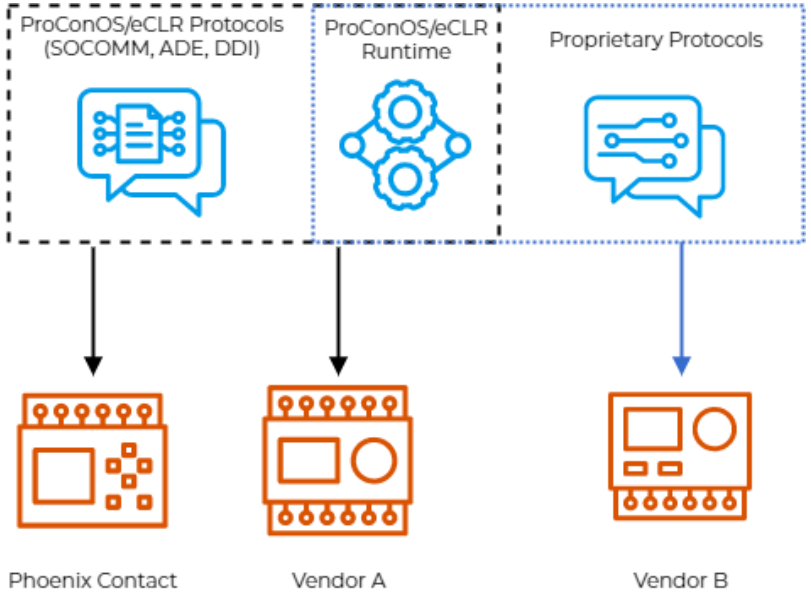
- See more supply chain vulns across tech stack
  - RTOSes, SDKs and standard libraries[1]
  - Protocol stacks[2]
  - IEC 61131-3 runtimes[3]
  - Remote access solutions[4]

- Vulns discovered in particular product rarely make their way ‘up & down the chain’ to other affected products
  - Discoverers unaware vuln is in 3rd party component
  - 3rd party vendor don’t have complete overview of supply chain (intermediate integrators, white label vendors, etc.)

- Leads to vuln collisions & risk blindness

[1] BadAlloc, uClibc DNS
[2] Urgent/11, Ripple20, Amnesia:33, INFRA:HALT, NUCLEUS:13, RTA 499ES ENIP, Siemens PROFINET
[3] CODESYS, ProConOS, ISaGRAF
[4] Access:7

Example: ProConOS runtime



- IEC 61131-3 runtime by KW soft (now Phoenix Contact)
  - Used by many OEMs, integrators, white label vendors
  - Different integration conditions
    - Runtime : ProConOS vs eCLR
    - Protocol : SOCOMM vs ADE vs custom (eg Emerson ControlWave)

- History of vulns
  - Unauthenticated protocols, RCE via unsigned logic
  - CVE-2022-31800/1 known but never assigned CVEs
  - Other CVEs only for Phoenix Contact products
  - Public PoCs available for years

- Lack of SBOMs leads to vuln rediscovery
  - CVE-2014-9195 (Phoenix) == CVE-2016-4860 (Yokogawa)
- We identified additional affected parties
  - Unfortunately, info wasn't backpropagated to original CVEs as suggested

Vendor	Product
Phoenix Contact	AXC, ILC, RFC, FC
Emerson	ControlWave
ABB	RTU 520/540/560
Advantech	ADAM, APAX, AMAX, UNO
KUKA	KUKA.PLC
ICP DAS	KinCon-8xxx
Yaskawa	Mpiec
Schleicher	XCx
Hilscher	netPLC
Luetze	DIOLINE PLC
Delta	DMXC
ISH	SIS, SIC, uPLC
Yokogawa	STARDOM

Nuance: Firmware Updates

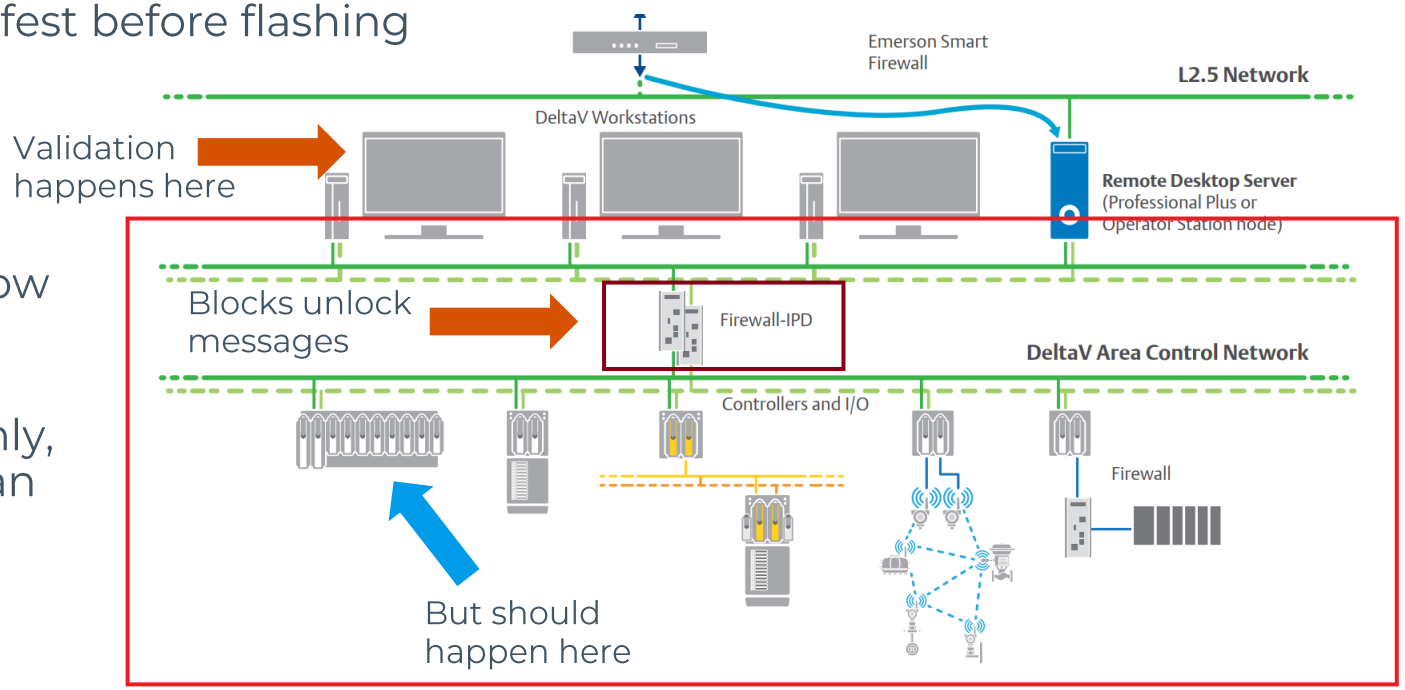
- Malicious FW updates
  - Are *noisy*: controller reboot + process interrupt likely triggers alarms
  - But *powerful*: persist, mass-brick controllers, etc.
- Only 51% had update authentication, only 22% FW signing
- Majority of updates over Ethernet
  - But: some via SD/USB/serial ← risk reduced to compromised EWS / converters
- Caveat: if you sign, need to do it right
  - I.e.: end-to-end & asymmetric

Example: Emerson DeltaV DCS

- Various unauthenticated, proprietary protocols (FW UPGD, PnP, SIS, Hawk SVC)
  - Impact: FW manipulation, config/strategy changes, service shutdown, etc.

- Controller firmwares are unsigned pre-14.3, only use CRC
- Emerson considers this *resolved* in 14.3 and *mitigated* elsewhere
  - 14.3+: Update tool validates sig in manifest before flashing
  - Firewall-IPD blocks unlock messages

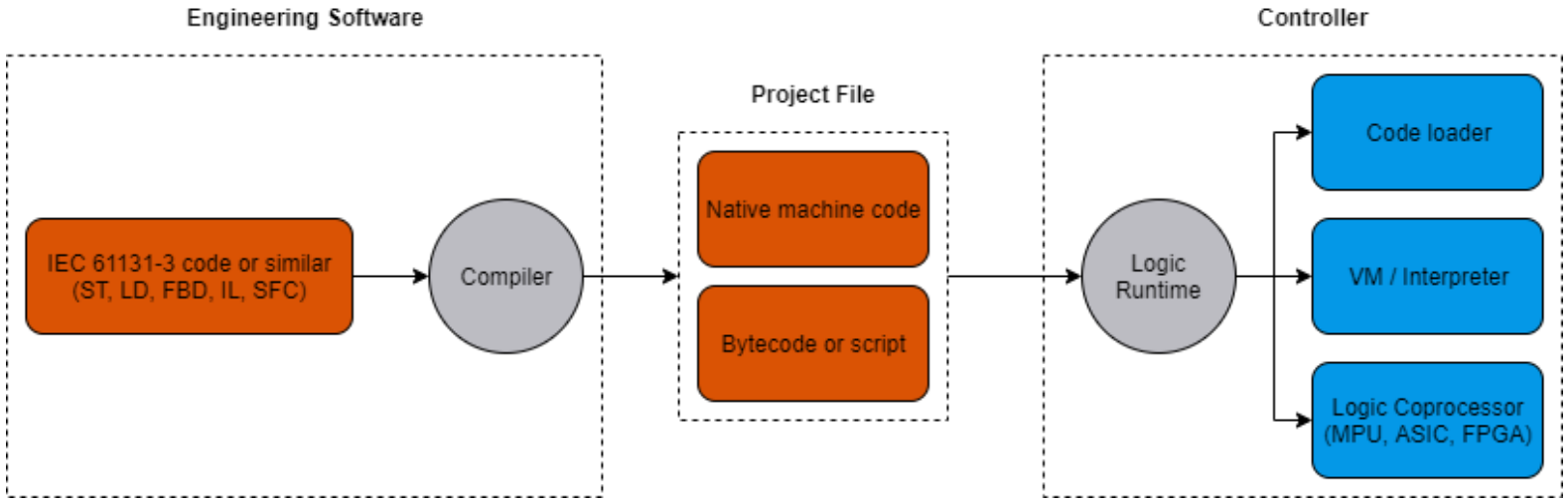
- But
  - Attacker can target legit update window with own tooling
  - In case IPD restricts comms to EWS only, attacker who can compromise EWS can still push malicious FW

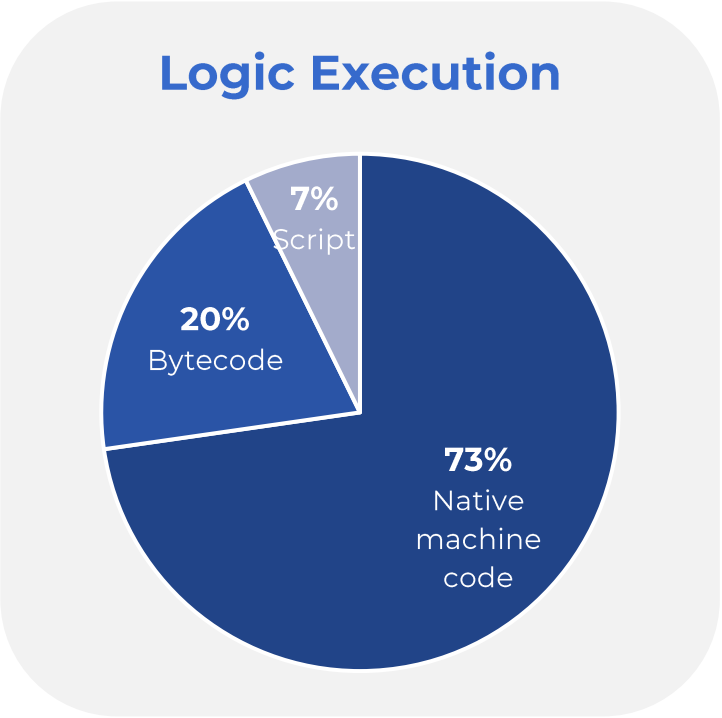


Source: DeltaV Remote Client Product Data Sheet (corresponds to reference architecture in DeltaV Security Manual V5.0.0)

Nuance: Logic Execution Model

- Majority but not all logic is executed as native machine code
  - No logic signing, only handful used sandboxing
  - Often on PLCs without RTOS/MMU support for memory & privilege separation
  - Leads to '*execute my shellcode please*' scenarios
  - Bytecode VMs are bigger hurdle (see MC7P in S7-1200)*
  - Bytecode ASICs / FPGAs even more robust

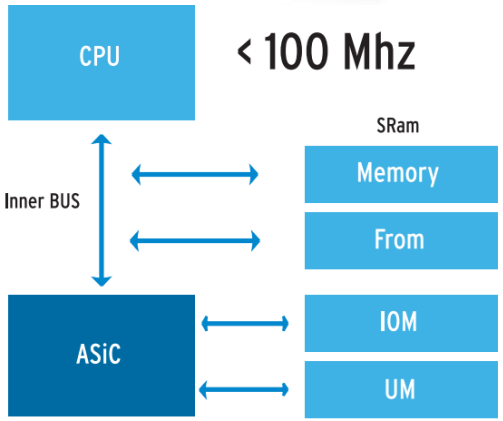


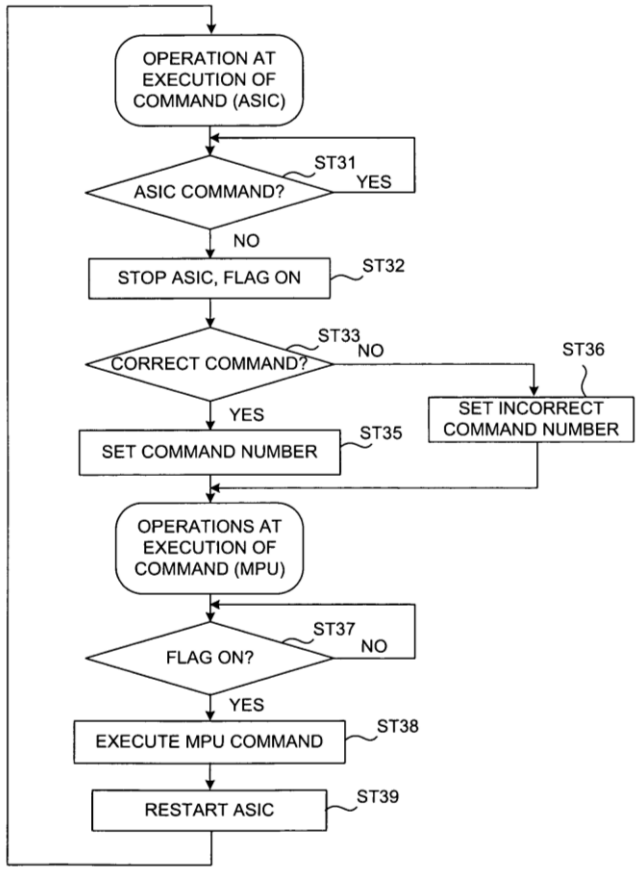


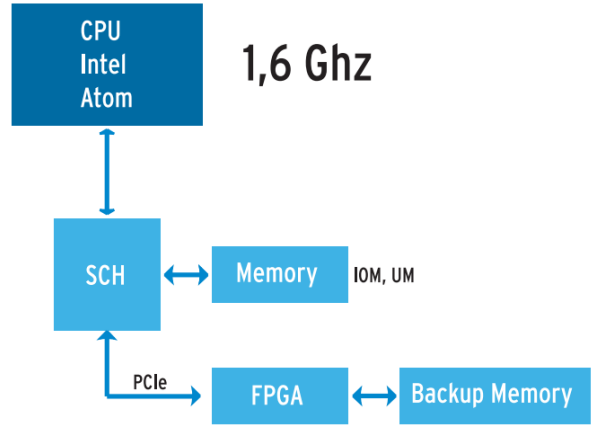
* See: S. Brizinov - The Race to Native Code Execution in PLCs

Example: Omron SYSMAC CS/CJ/CP vs NJ/NX PLCs

- Proprietary ASIC bytecode vs x86 machine code
  - Difference is constrained vs unconstrained, low-level access



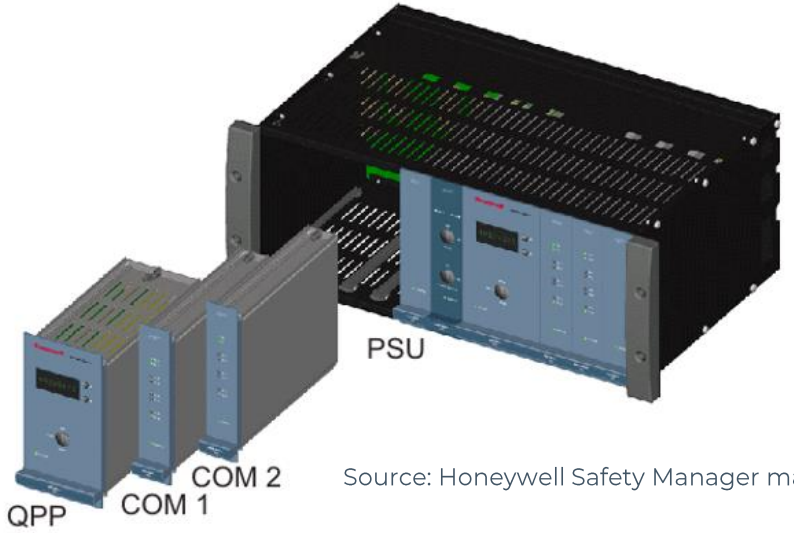




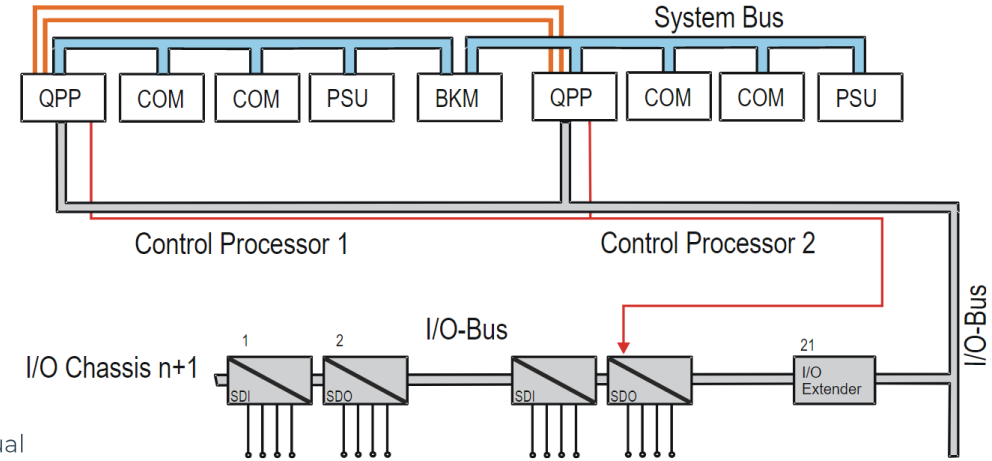
Source: Omron Sysmac NJ customer reference, Omron Patent US7243260B2

Example: Honeywell Safety Manager (SC) SIS

- Safety Manager
  - Safety Station compiles SIF FLDs to unsigned, native machine code
  - Downloads projects using unauthenticated Safety Builder protocol
  - USI module sends logic via backplane to QPP CPU which executes SIF logic



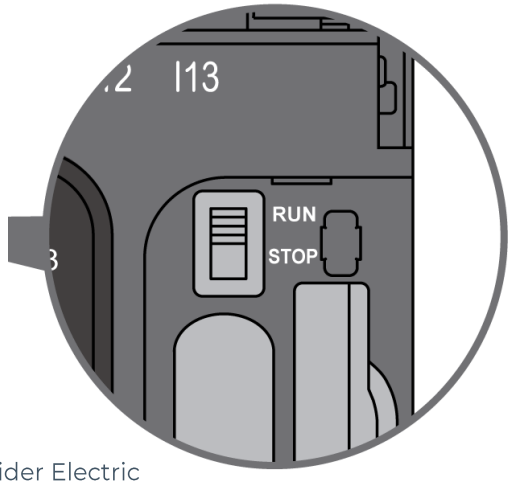
Source: Honeywell Safety Manager manual



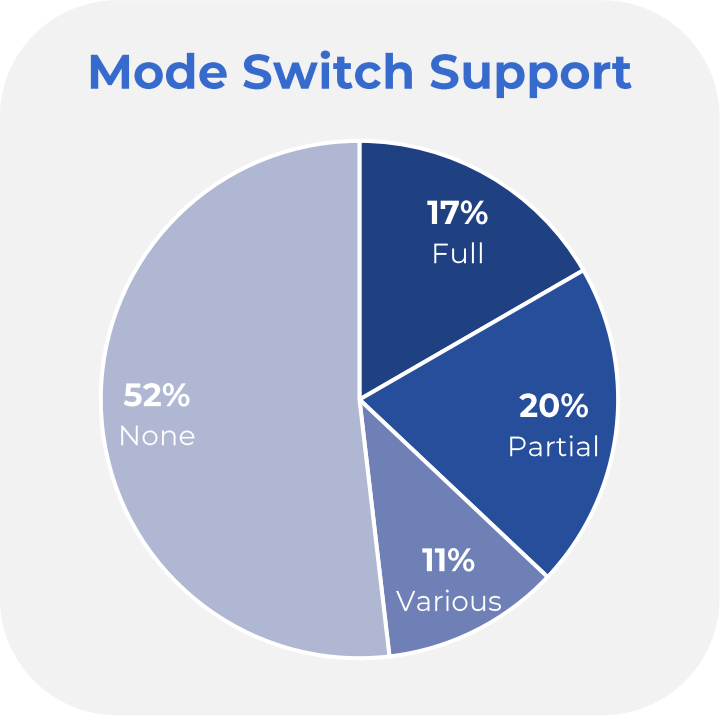
- Safety Manager SC (S300)
  - SIF FLDs compiled to mnemonic bytecode instead

Nuance: Mode Switches

- ‘Traditional’ defense against logic downloads & FW updates
  - Physical switch to set RUN/REM/PROG/IDLE modes



Source: Cyberark, Schneider Electric



- Pitfalls
  - *Distinctness*: want distinct RUN / PROG modes
  - *Virtual switches*: just a setting in EWS software
  - *Exception modes*: instruct PLC to ignore switch settings
  - *Defaults*: not all sensitive ops require switch settings by default

Example: Emerson ControlWave PLC/RTU

- Unsigned firmware updates (CVE-2022-30262)
  - Troubling combined with BSAP/IP auth bypasses

- "*Not a vulnerability*" according to Emerson since
  - Keyswitch can be set to RUN, config setting can disable remote changes

_APPLICATION_LOCKED	%MX 3.103.0 : BOOL	4.50	When set TRUE, prevents external control changes to project via ControlWave Designer. Also prevents project downloads.

Source: ControlWave Micro Instruction Manual, ControlWave Designer Programmer's Handbook

- However
  - Keyswitch has REMOTE mode, Attacker can still wait for legitimate window …
  - _APPLICATION_LOCKED not set by default, prohibits *all* remote changes

Example: Honeywell Safety Manager SIS

- QPP module keyswitch must be IDLE *before* logic download
- BKM RESET keyswitch must be triggered *after* logic download

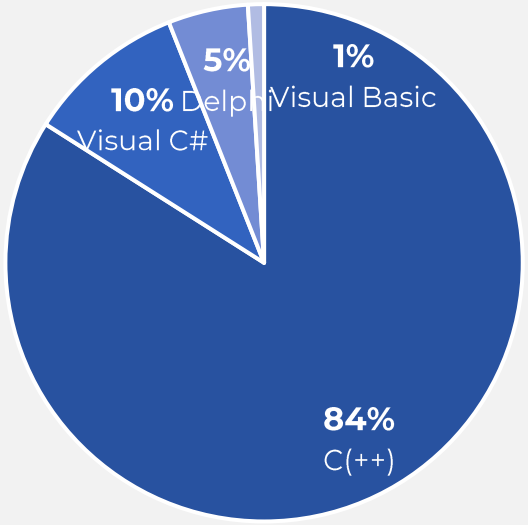
- Except when "*remote load/reset*" are enabled!
  - Physical key needed to enable feature
  - But if enabled for historical reasons and not documented in ISMS or overlooked, might lead to blind spot

Reverse Engineering Effort

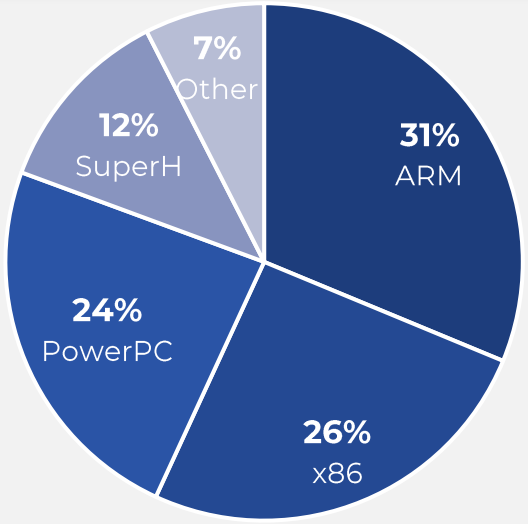
Reverse Engineering

For offensive OT capability development

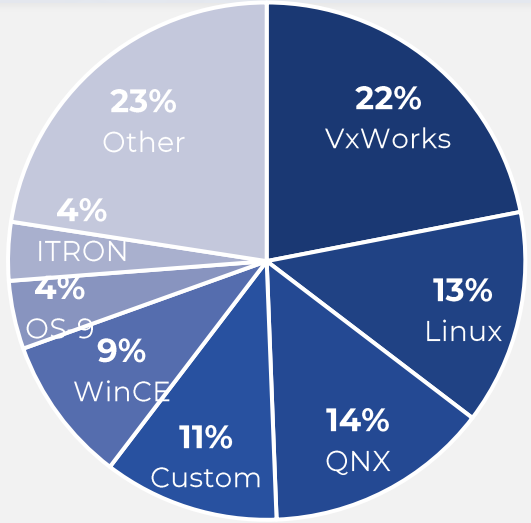
Dev. Languages



CPU Architectures



RTOSes

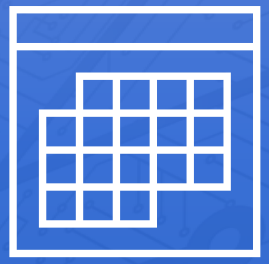


- Windows software packages are typically huge (GBs) & complex
  - 100s of DLLs, MFC, ATL, COM, RPC, Qt

- Devices match typical non-consumer embedded systems
  - Regional outliers (OS-9/ITRON + SuperH in Asia)

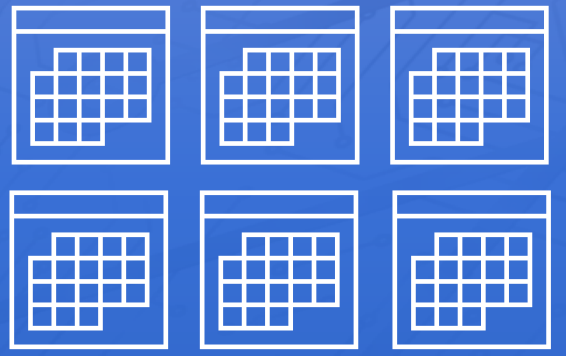
Offensive Capabilities are Feasible to Develop

Reverse engineering a single proprietary protocol



- Took between 1 day and 2 man-weeks

Reverse engineering a complex, multi-protocol system



- Took 5 to 6 man-months

- Basic offensive cyber capabilities leading to the development of OT-focused malware or cyberattacks could be developed by a small but skilled team at a reasonable cost

Conclusions

Conclusions

- Insecure-by-design continues to persist in production install base despite decade+ of hardening efforts

- Need to get clearer on what *secure-by-design* actually means
  - Many security controls turn out to be trivially broken
  - Products with broken controls continue to be certified
  - Vulns sometimes dismissed "because VPN+FW", even if risk not fully controlled
  - Security retrofits sometimes miss the point (e.g. IP ACL on UDP protocol)
  - Fixes frequently don't address root cause
  - Lack of variant hunting suggests immature SDLCs
  - Should be secure-by-default, not "there's options somewhere in the manual"

CTA

- **Device manufacturers** – Properly secure OT devices and protocols
- **Asset owners** – Actively procure for secure-by-design products
- Wider **security community** – Ensure that security controls are robust

Thank you.

VEDERE LABS