UNIVERSITY OF BIRMINGHAM

RITICS | CENTRE FOR CYBER SECURITY AND PRIVACY | UKRRIN UK RAIL RESEARCH AND INNOVATION NETWORK

Learning from Vulnerabilities – Categorising, Understanding and Detecting Weaknesses in ICS

Richard J. Thomas and Tom Chothia, University of Birmingham

R.J.Thomas@cs.bham.ac.uk

First… what is an Industrial Control System?

- Industrial Automation

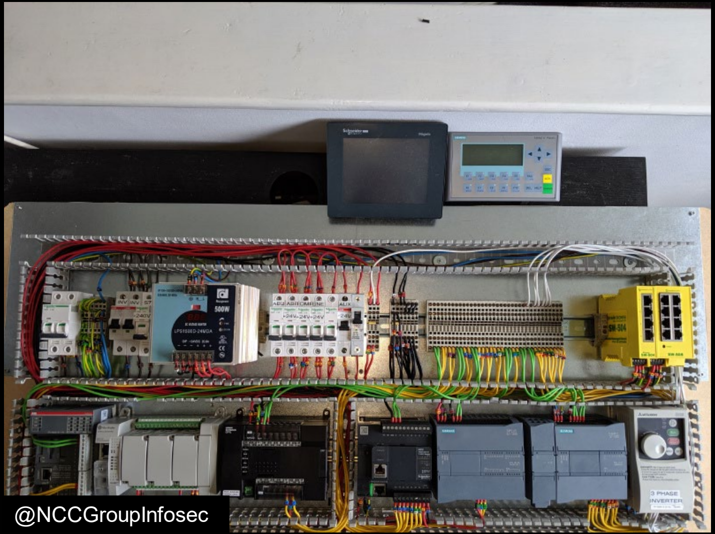
  Automating **people, plant and process**

- **Sensor-based** systems

  Actions defined through **logic-based** functions

- **Core** to ‘Industry 4.0’

- First PLC-based **Rail Signalling System** now deployed (Atkins and Alstom)



@NCCGroupInfosec

Industrial Control Systems as the New Target

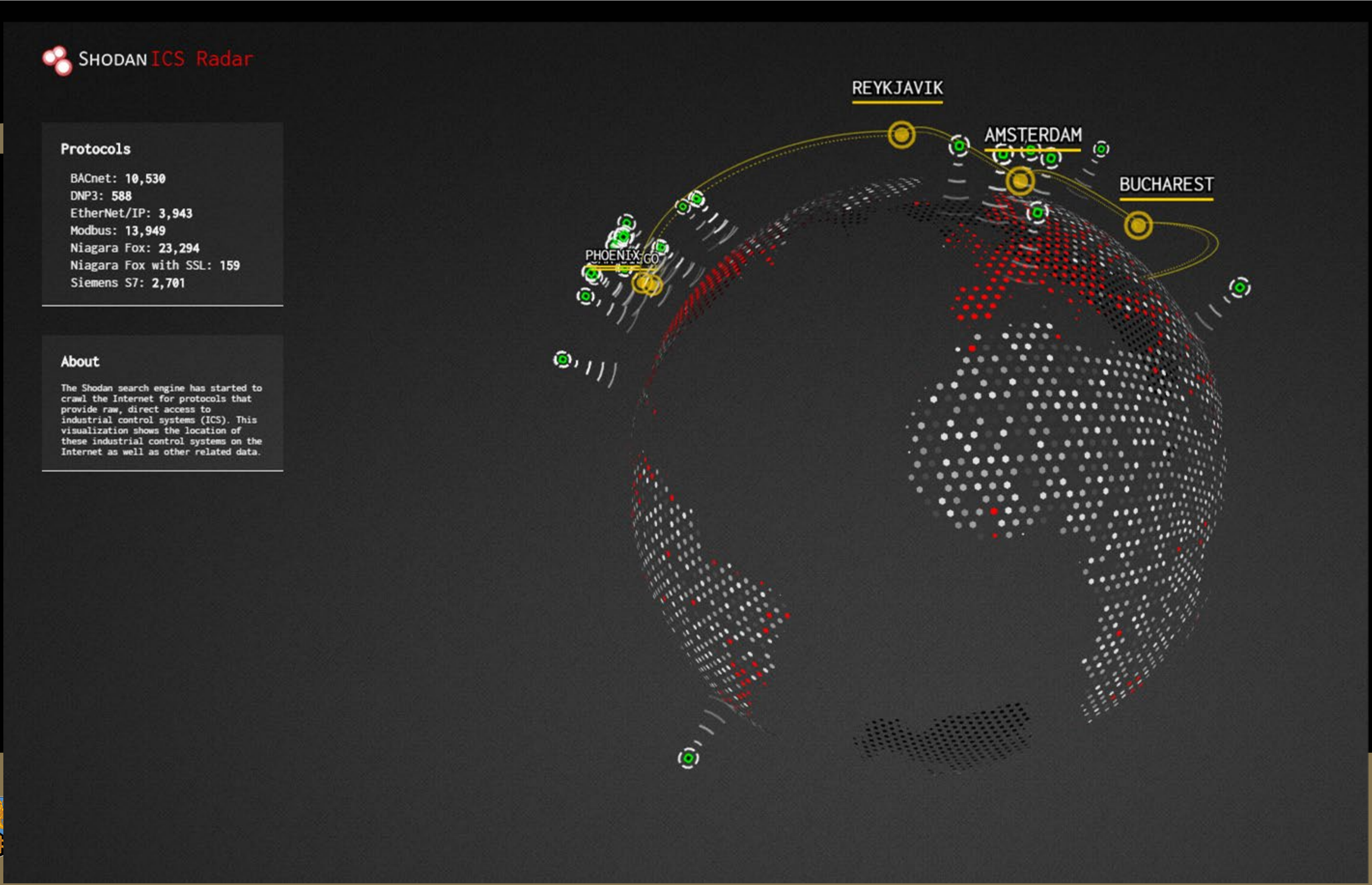
- **Legacy** Protocols – *Profinet, Modbus, BACnet, S7Comm, DNP3*

  Little/no authentication and encryption
- Design and Operational Lifespan is measured in ***decades***

  Commodity IT hardware ~5-10 years
- Traditional deployments were **bespoke**

  Small attack surface/scalability, now with **COTS** hardware, **attacks scale**
- Attacks are **evolving**

  Previously aimed at disruption (Stuxnet, Wannacry) but now attacking safety protection systems (Triton)



SHODAN ICS Radar

Protocols

BACnet:	10,530
DNP3:	588
EtherNet/IP:	3,943
Modbus:	13,949
Niagara Fox:	23,294
Niagara Fox with SSL:	159
Siemens S7:	2,701

About

The Shodan search engine has started to crawl the Internet for protocols that provide raw, direct access to industrial control systems (ICS). This visualization shows the location of these industrial control systems on the Internet as well as other related data.

The EU Network and Information Systems (NIS) Directive – EU 2016/1148

- In force since **May 2018**

  Member states responsible for own implementations, nominating *CAs*

- Aimed to deliver a *culture change* to secure **essential** services (Rail, Aviation, Roads, Water and power)
- Financial and Legislative consequences for lack of reporting

  €300 (Slovenia) - €17,000,000 (UK)

- **Emphasis** on supply chain *assurance*
- **Question**: *How do you do this?*

Previous/Relevant Work

- Previous work has focused on **vulnerability research** or **chronologies**

- Last ‘state of ICS’ report by ICS-CERT – **2016**

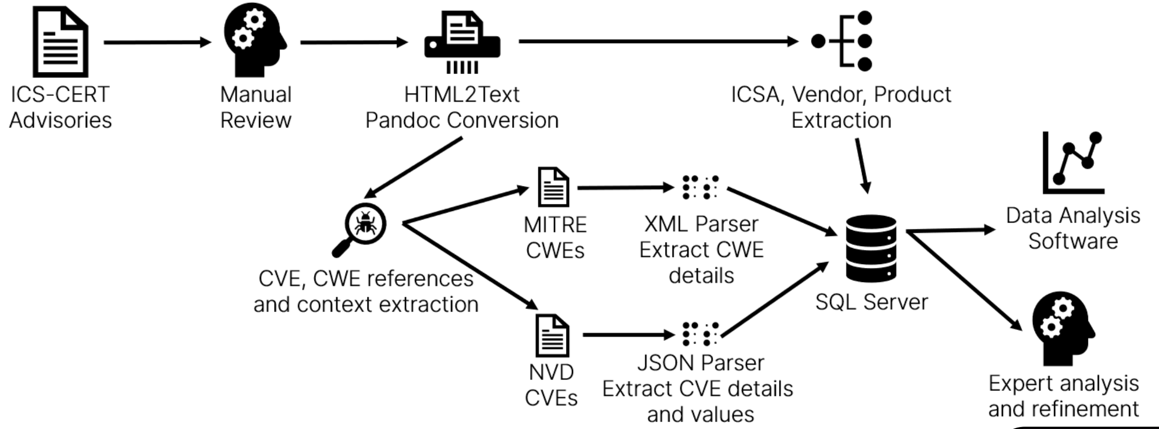
- **OpenCTI** project aims to make threat information more transparent to asset owners, but is not actionable and no ICS-CERT integrations

- **Jiang et al. (2019)** develops a correlated database, but their categories are not distinct or actionable

A Data-led Analysis of ICS Vulnerabilities

- Collect **9 years** worth of ICS vulnerability reports

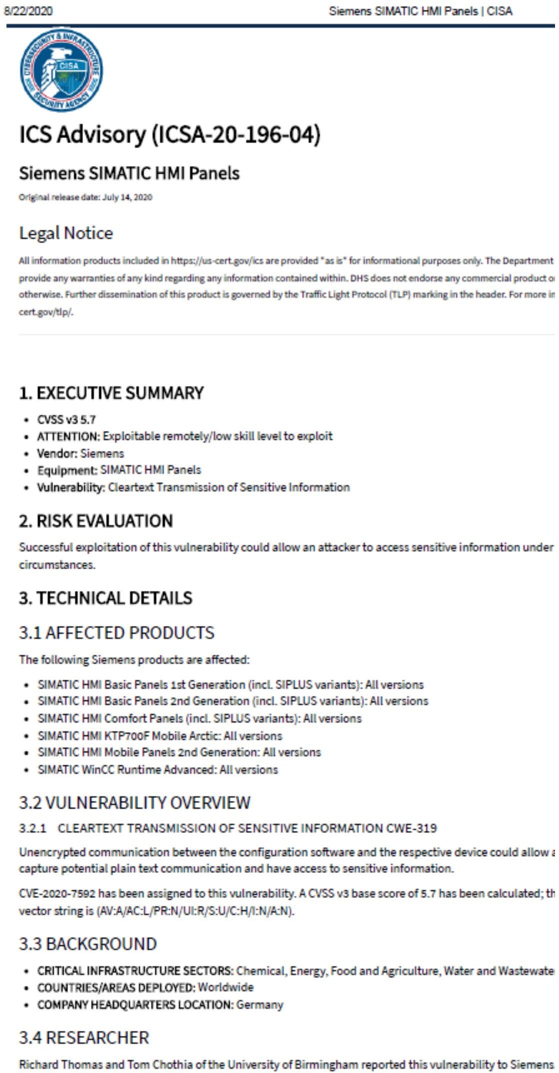
  Scrape ICS-CERT, convert to Markdown and automatically process


ICS-CERT Advisories
Manual Review
HTML2Text Pandoc Conversion
ICSA, Vendor, Product Extraction
CVE, CWE references and context extraction
MITRE CWEs
XML Parser Extract CWE details
NVD CVEs
JSON Parser Extract CVE details and values
SQL Server
Data Analysis Software
Expert analysis and refinement

- Our Dataset (through to August 2019):

1,114	ICS Advisories
283	Distinct CWEs
2,232	ICS CVEs

Dataset now live:
uob-ritics.github.io



8/22/2020 Siemens SIMATIC HMI Panels | CISA

ICS Advisory (ICSA-20-196-04)

Siemens SIMATIC HMI Panels

Original release date: July 14, 2020

Legal Notice

All information products included in https://us-cert.gov/ics are provided "as is" for informational purposes only. The Department provide any warranties of any kind regarding any information contained within. DHS does not endorse any commercial product o otherwise. Further dissemination of this product is governed by the Traffic Light Protocol (TLP) marking in the header. For more in cert.gov/tlp/.

1. EXECUTIVE SUMMARY

- CVSS v3 5.7
- **ATTENTION:** Exploitable remotely/low skill level to exploit
- **Vendor:** Siemens
- **Equipment:** SIMATIC HMI Panels
- **Vulnerability:** Cleartext Transmission of Sensitive Information

2. RISK EVALUATION

Successful exploitation of this vulnerability could allow an attacker to access sensitive information under circumstances.

3. TECHNICAL DETAILS

3.1 AFFECTED PRODUCTS

The following Siemens products are affected:

- SIMATIC HMI Basic Panels 1st Generation (incl. SIPLUS variants): All versions
- SIMATIC HMI Basic Panels 2nd Generation (incl. SIPLUS variants): All versions
- SIMATIC HMI Comfort Panels (incl. SIPLUS variants): All versions
- SIMATIC HMI KTP700F Mobile Arctic: All versions
- SIMATIC HMI Mobile Panels 2nd Generation: All versions
- SIMATIC WinCC Runtime Advanced: All versions

3.2 VULNERABILITY OVERVIEW

3.2.1 CLEARTEXT TRANSMISSION OF SENSITIVE INFORMATION CWE-319

Unencrypted communication between the configuration software and the respective device could allow a capture potential plain text communication and have access to sensitive information.

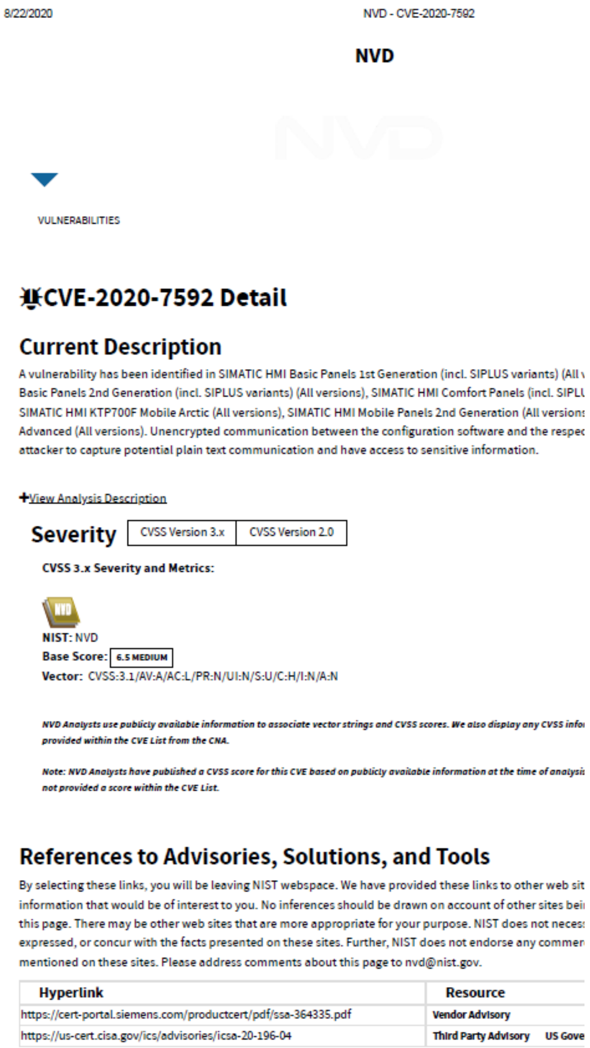
CVE-2020-7592 has been assigned to this vulnerability. A CVSS v3 base score of 5.7 has been calculated; th vector string is (AV:A/AC:L/PR:N/UI:R/S:U/C:H/I:N/A:N).

3.3 BACKGROUND

- **CRITICAL INFRASTRUCTURE SECTORS:** Chemical, Energy, Food and Agriculture, Water and Wastewate
- **COUNTRIES/AREAS DEPLOYED:** Worldwide
- **COMPANY HEADQUARTERS LOCATION:** Germany

3.4 RESEARCHER

Richard Thomas and Tom Chothia of the University of Birmingham reported this vulnerability to Siemens



8/22/2020 NVD - CVE-2020-7592

NVD

VULNERABILITIES

CVE-2020-7592 Detail

Current Description

A vulnerability has been identified in SIMATIC HMI Basic Panels 1st Generation (incl. SIPLUS variants) (All v Basic Panels 2nd Generation (incl. SIPLUS variants) (All versions), SIMATIC HMI Comfort Panels (incl. SIPLU SIMATIC HMI KTP700F Mobile Arctic (All versions), SIMATIC HMI Mobile Panels 2nd Generation (All version Advanced (All versions). Unencrypted communication between the configuration software and the respec attacker to capture potential plain text communication and have access to sensitive information.

+View Analysis Description

Severity

CVSS 3.x Severity and Metrics:

NIST: NVD
Base Score: 6.5 MEDIUM
Vector: CVSS:3.1/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

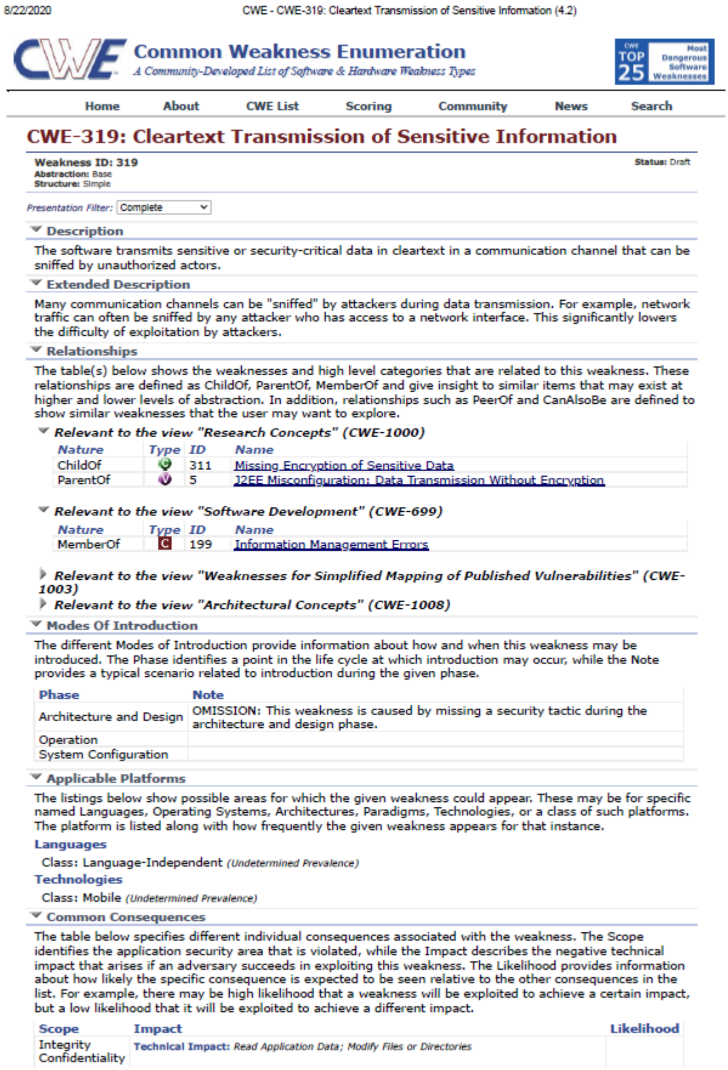
NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS infor provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis not provided a score within the CVE List.

References to Advisories, Solutions, and Tools

By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sit information that would be of interest to you. No inferences should be drawn on account of other sites bei this page. There may be other web sites that are more appropriate for your purpose. NIST does not neces expressed, or concur with the facts presented on these sites. Further, NIST does not endorse any commer mentioned on these sites. Please address comments about this page to nvd@nist.gov.

Hyperlink	Resource
https://cert-portal.siemens.com/productcert/pdf/ssa-364335.pdf	**Vendor Advisory**
https://us-cert.cisa.gov/ics/advisories/icsa-20-196-04	**Third Party Advisory** **US Gove**



8/22/2020 CWE - CWE-319: Cleartext Transmission of Sensitive Information (4.2)

Common Weakness Enumeration
A Community-Developed List of Software & Hardware Weakness Types

CWE-319: Cleartext Transmission of Sensitive Information

Weakness ID: 319
Abstraction: Base
Structure: Simple

Status: Draft

Presentation Filter: Complete

Description

The software transmits sensitive or security-critical data in cleartext in a communication channel that can be sniffed by unauthorized actors.

Extended Description

Many communication channels can be "sniffed" by attackers during data transmission. For example, network traffic can often be sniffed by any attacker who has access to a network interface. This significantly lowers the difficulty of exploitation by attackers.

Relationships

The table(s) below shows the weaknesses and high level categories that are related to this weakness. These relationships are defined as ChildOf, ParentOf, MemberOf and give insight to similar items that may exist at higher and lower levels of abstraction. In addition, relationships such as PeerOf and CanAlsoBe are defined to show similar weaknesses that the user may want to explore.

Relevant to the view "Research Concepts" (CWE-1000)

Nature	Type	ID	Name
ChildOf	●	311	Missing Encryption of Sensitive Data
ParentOf	V	5	J2EE Misconfiguration: Data Transmission Without Encryption

Relevant to the view "Software Development" (CWE-699)

Nature	Type	ID	Name
MemberOf	C	199	Information Management Errors

Relevant to the view "Weaknesses for Simplified Mapping of Published Vulnerabilities" (CWE-1003)

Relevant to the view "Architectural Concepts" (CWE-1008)

Modes Of Introduction

The different Modes of Introduction provide information about how and when this weakness may be introduced. The Phase identifies a point in the life cycle at which introduction may occur, while the Note provides a typical scenario related to introduction during the given phase.

Phase	Note
Architecture and Design	OMISSION: This weakness is caused by missing a security tactic during the architecture and design phase.
Operation	
System Configuration	

Applicable Platforms

The listings below show possible areas for which the given weakness could appear. These may be for specific named Languages, Operating Systems, Architectures, Paradigms, Technologies, or a class of such platforms. The platform is listed along with how frequently the given weakness appears for that instance.

Languages

Class: Language-Independent *(Undetermined Prevalence)*

Technologies

Class: Mobile *(Undetermined Prevalence)*

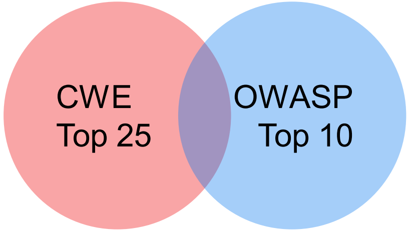
Common Consequences

The table below specifies different individual consequences associated with the weakness. The Scope identifies the application security area that is violated, while the Impact describes the negative technical impact that arises if an adversary succeeds in exploiting this weakness. The Likelihood provides information about how likely the specific consequence is expected to be seen relative to the other consequences in the list. For example, there may be high likelihood that a weakness will be exploited to achieve a certain impact, but a low likelihood that it will be exploited to achieve a different impact.

Scope	Impact	Likelihood
Integrity Confidentiality	**Technical Impact:** *Read Application Data; Modify Files or Directories*	

Understanding ICS Vulnerabilities

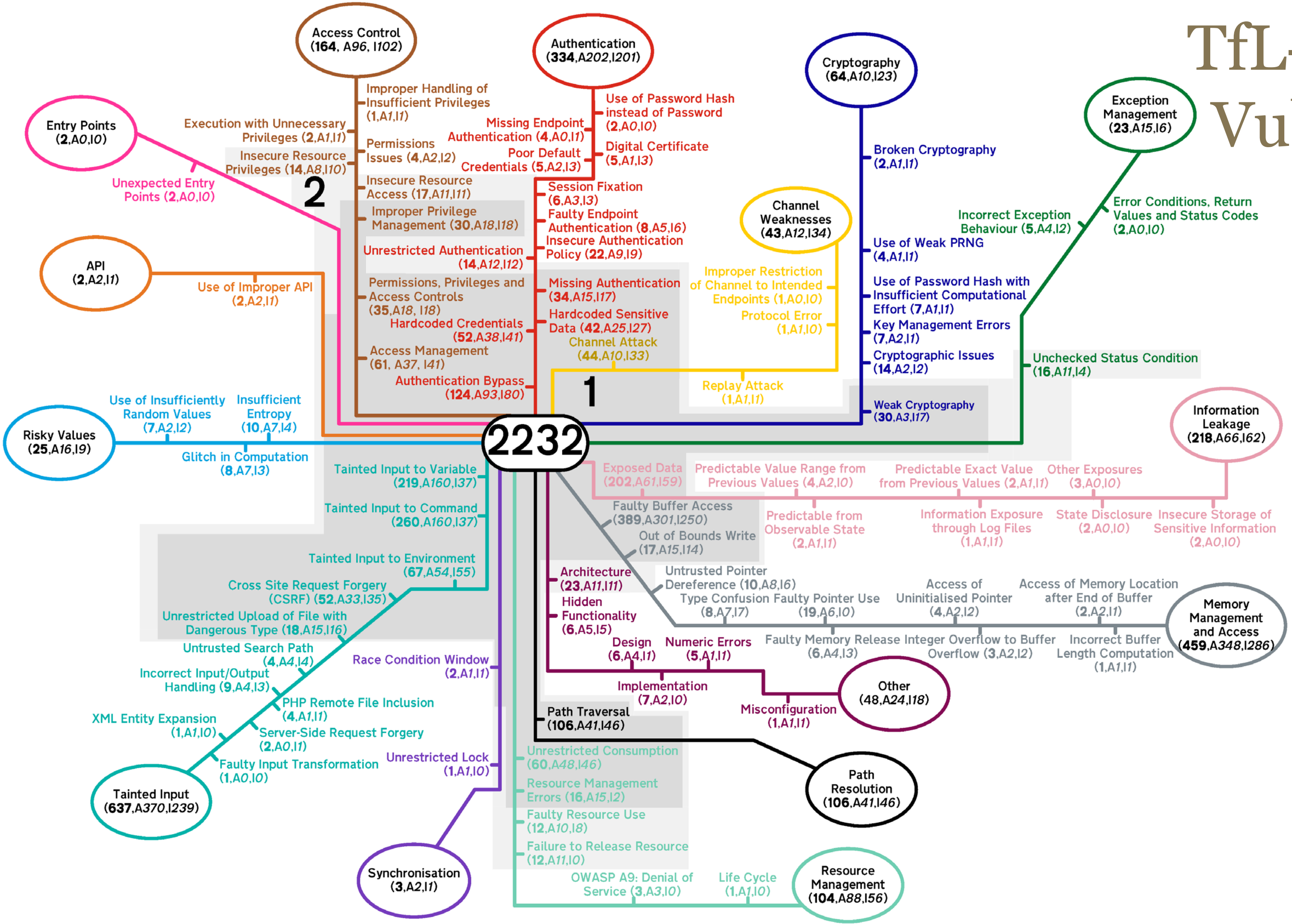
- Existing CWE Groupings did not have **complete coverage**
  Often an overlap between groupings



- **SFP Clusters** offered alternative groupings based on the nature of vulnerability
  e.g. Memory Access, Cryptography
  Offered **mutually exclusive** ways of grouping classes

- **1,801** CVEs could be mapped to **existing clusters**
  Remaining 431 were **manually assigned** to a cluster
  e.g. CSRF (web vulnerability) → Tainted Input
  Weak Password Hash → Cryptography

TfL-Map of ICS Vulnerabilities


2232
1
2
Access Control (164, A96, I102)
Improper Handling of Insufficient Privileges (1,A1,I1)
Execution with Unnecessary Privileges (2,A1,I1)
Permissions Issues (4,A2,I2)
Insecure Resource Privileges (14,A8,I10)
Insecure Resource Access (17,A11,I11)
Improper Privilege Management (30,A18,I18)
Unrestricted Authentication (14,A12,I12)
Permissions, Privileges and Access Controls (35,A18, I18)
Hardcoded Credentials (52,A38,I41)
Access Management (61, A37, I41)
Authentication Bypass (124,A93,I80)
Authentication (334,A202,I201)
Use of Password Hash instead of Password (2,A0,I0)
Missing Endpoint Authentication (4,A0,I1)
Digital Certificate (5,A1,I3)
Poor Default Credentials (5,A2,I3)
Session Fixation (6,A3,I3)
Faulty Endpoint Authentication (8,A5,I6)
Insecure Authentication Policy (22,A9,I9)
Missing Authentication (34,A15,I17)
Hardcoded Sensitive Data (42,A25,I27)
Channel Attack (44,A10,I33)
Cryptography (64,A10,I23)
Broken Cryptography (2,A1,I1)
Use of Weak PRNG (4,A1,I1)
Use of Password Hash with Insufficient Computational Effort (7,A1,I1)
Key Management Errors (7,A2,I1)
Cryptographic Issues (14,A2,I2)
Weak Cryptography (30,A3,I17)
Channel Weaknesses (43,A12,I34)
Improper Restriction of Channel to Intended Endpoints (1,A0,I0)
Protocol Error (1,A1,I0)
Replay Attack (1,A1,I1)
Exception Management (23,A15,I6)
Incorrect Exception Behaviour (5,A4,I2)
Error Conditions, Return Values and Status Codes (2,A0,I0)
Unchecked Status Condition (16,A11,I4)
Entry Points (2,A0,I0)
Unexpected Entry Points (2,A0,I0)
API (2,A2,I1)
Use of Improper API (2,A2,I1)
Risky Values (25,A16,I9)
Use of Insufficiently Random Values (7,A2,I2)
Insufficient Entropy (10,A7,I4)
Glitch in Computation (8,A7,I3)
Information Leakage (218,A66,I62)
Exposed Data (202,A61,I59)
Predictable Value Range from Previous Values (4,A2,I0)
Predictable Exact Value from Previous Values (2,A1,I1)
Other Exposures (3,A0,I0)
Predictable from Observable State (2,A1,I1)
Information Exposure through Log Files (1,A1,I1)
State Disclosure (2,A0,I0)
Insecure Storage of Sensitive Information (2,A0,I0)
Tainted Input (637,A370,I239)
Tainted Input to Variable (219,A160,I37)
Tainted Input to Command (260,A160,I37)
Tainted Input to Environment (67,A54,I55)
Cross Site Request Forgery (CSRF) (52,A33,I35)
Unrestricted Upload of File with Dangerous Type (18,A15,I16)
Untrusted Search Path (4,A4,I4)
Incorrect Input/Output Handling (9,A4,I3)
PHP Remote File Inclusion (4,A1,I1)
XML Entity Expansion (1,A1,I0)
Server-Side Request Forgery (2,A0,I1)
Faulty Input Transformation (1,A0,I0)
Memory Management and Access (459,A348,I286)
Faulty Buffer Access (389,A301,I250)
Out of Bounds Write (17,A15,I14)
Untrusted Pointer Dereference (10,A8,I6)
Type Confusion (8,A7,I7)
Faulty Pointer Use (19,A6,I0)
Access of Uninitialised Pointer (4,A2,I2)
Access of Memory Location after End of Buffer (2,A2,I1)
Faulty Memory Release (6,A4,I3)
Integer Overflow to Buffer Overflow (3,A2,I2)
Incorrect Buffer Length Computation (1,A1,I1)
Other (48,A24,I18)
Architecture (23,A11,I11)
Hidden Functionality (6,A5,I5)
Design (6,A4,I1)
Numeric Errors (5,A1,I1)
Implementation (7,A2,I0)
Misconfiguration (1,A1,I1)
Path Resolution (106,A41,I46)
Path Traversal (106,A41,I46)
Synchronisation (3,A2,I1)
Race Condition Window (2,A1,I1)
Unrestricted Lock (1,A1,I0)
Resource Management (104,A88,I56)
Unrestricted Consumption (60,A48,I46)
Resource Management Errors (16,A15,I2)
Faulty Resource Use (12,A10,I8)
Failure to Release Resource (12,A11,I0)
OWASP A9: Denial of Service (3,A3,I0)
Life Cycle (1,A1,I0)

Zone 1 = Classes with >=15 *A/I* CVEs

Zone 2 = Classes with >=10 *A/I* CVEs

Rank on prevalence based on **complete/high** integrity/availability impact

Assists **vendors**, but not **asset owners**

Improving ICS Vulnerability Classification with **Detectable** categories

- SFP Clusters do not guide analysts towards **detection** and contain **ambiguity**

  What is ‘*tainted input to variable*’/‘*Information Leakage*’?

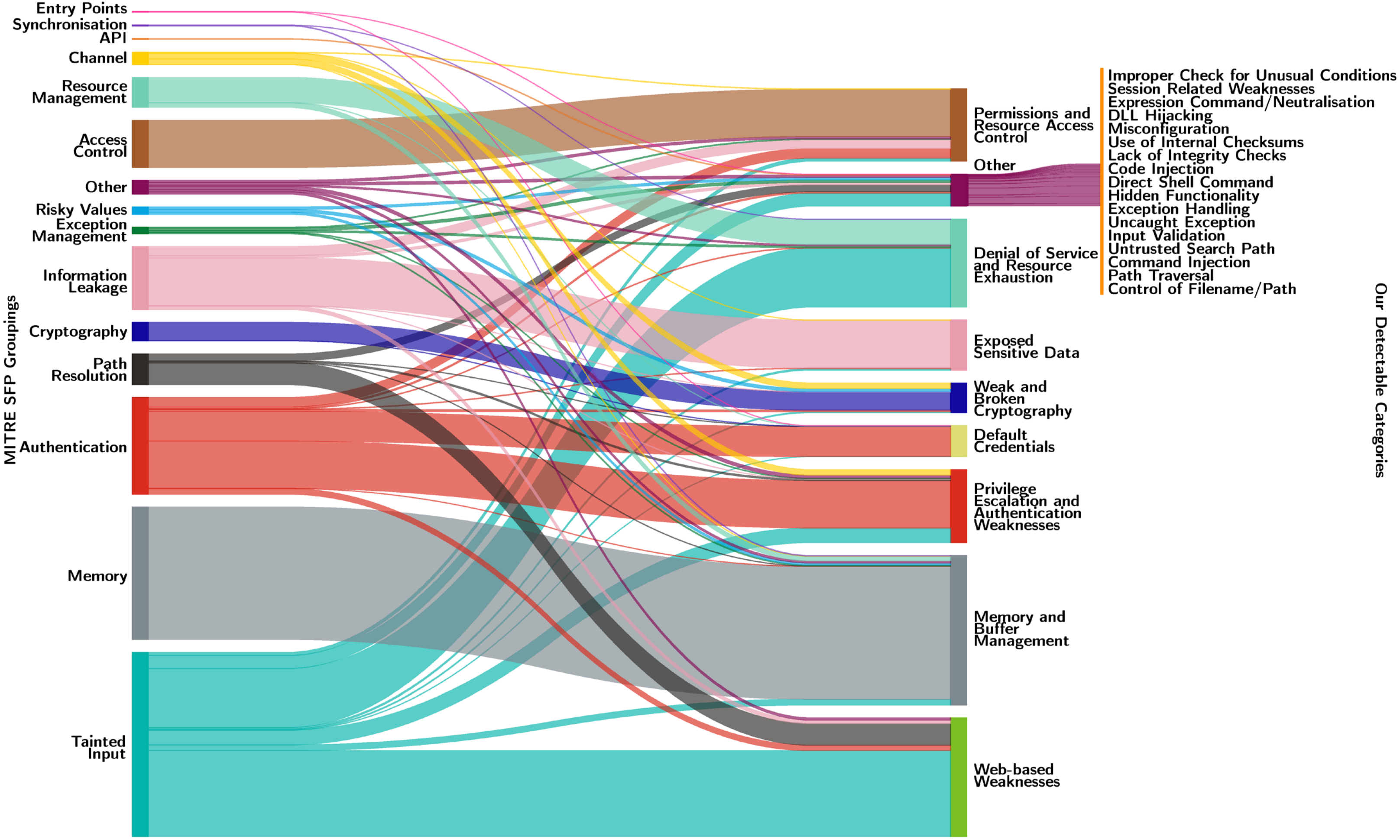
- **8 new *detectable*** vulnerability classes developed

  Evidence-driven, based on the way these types of vulnerability can be discovered

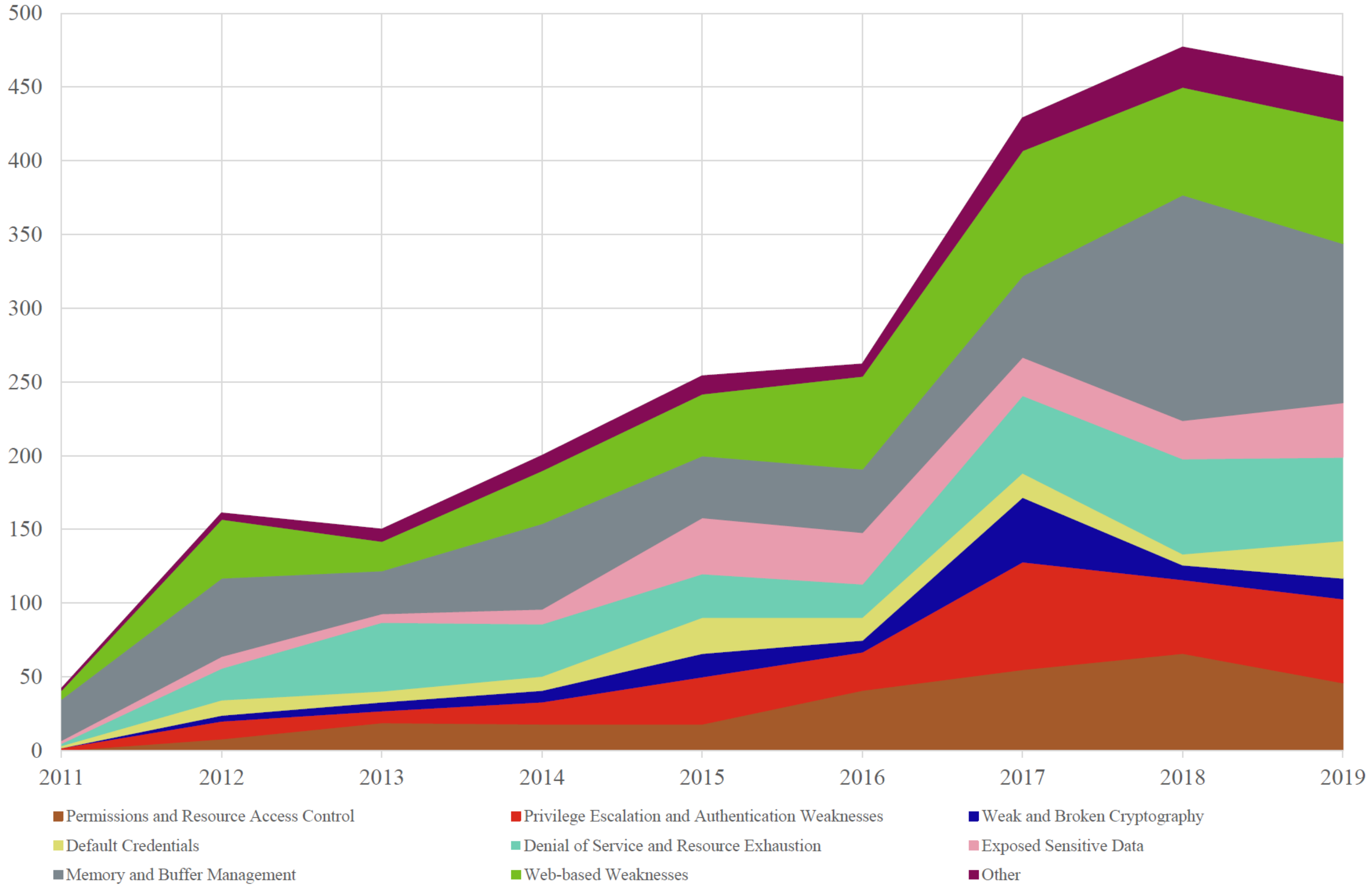
- **95% coverage** into the 8 detectable classes

- Each detectable class has a **precise definition**, removing ambiguity
- **Proven** to be dominant across the entire dataset

From SFP Cluster to Category

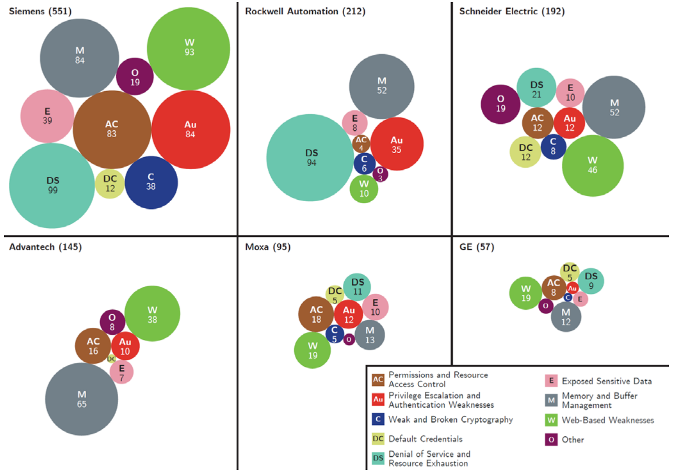


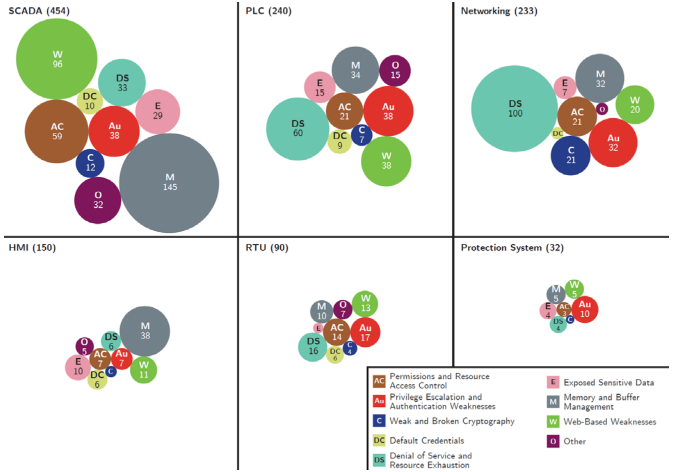
Dominance over Time



Applying these Categories to Vendors and Devices

- Based on **top 6** vendors/device types, distribution is **consistent**





- What will **new ICS CVEs** arise from?
- How do we **test** for these types of vulnerability?

What new ICS Vulnerabilities are on the Horizon?

- Dataset was ‘trained’ up to **August 2019**.

- How **effective** are our categories?

- Imported new ICS Advisories for **September 2019 – March 2020**

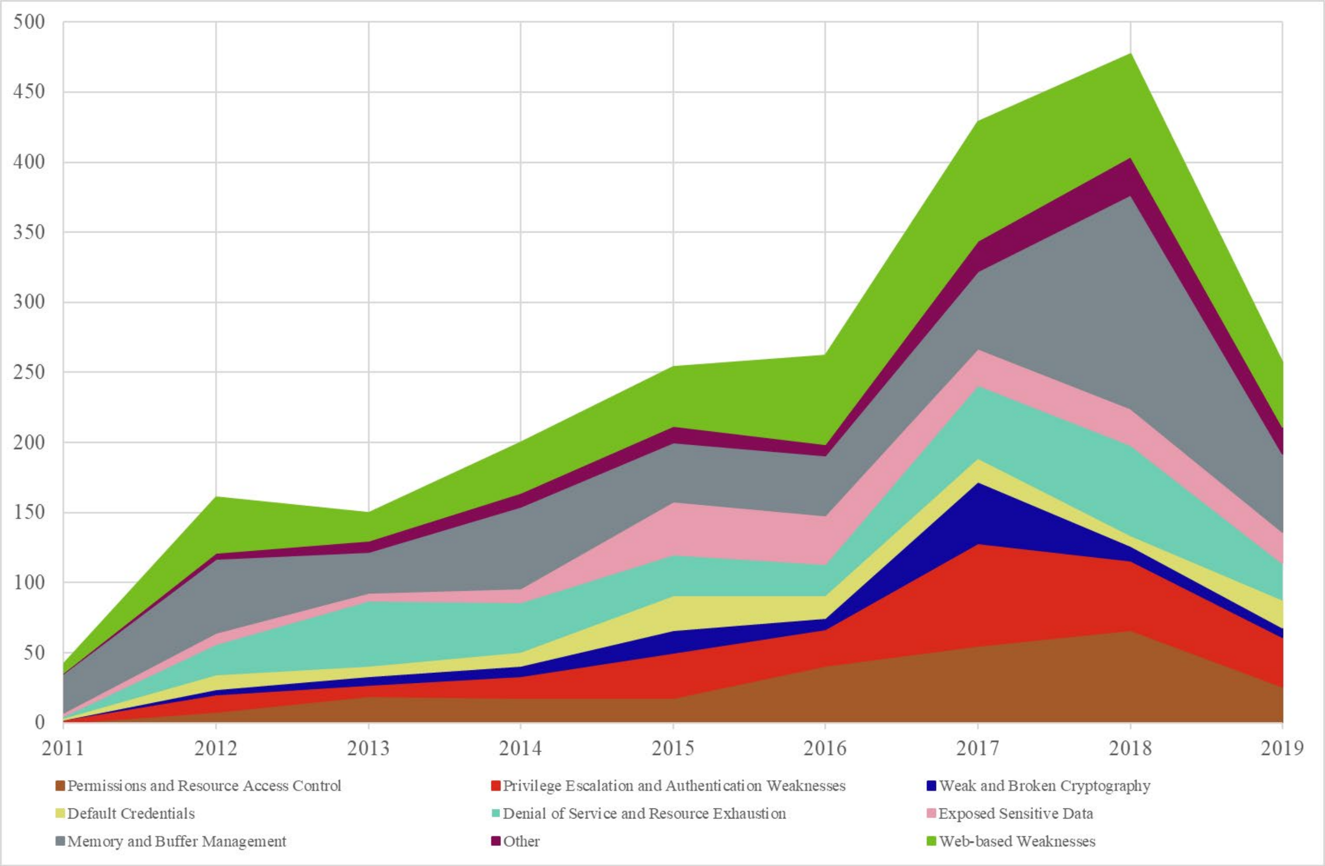
  126 new advisories

  334 CVEs

  Processed using the **same workflow**

- **96%** corresponded to one of our **8 categories**
- With **high confidence**, we can predict the types of new ICS vulnerabilities

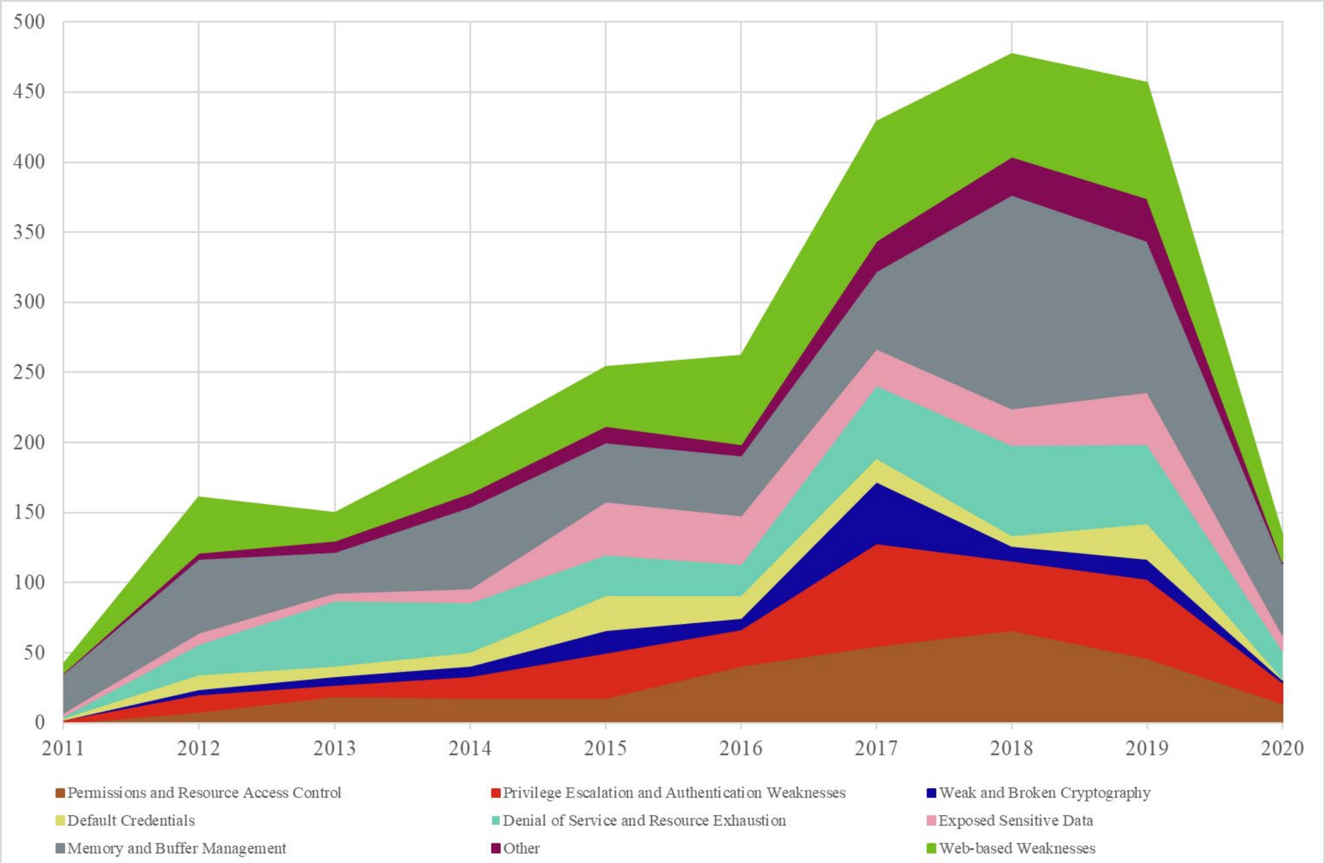
What new ICS Vulnerabilities are on the Horizon?



2020

vulnerabilities

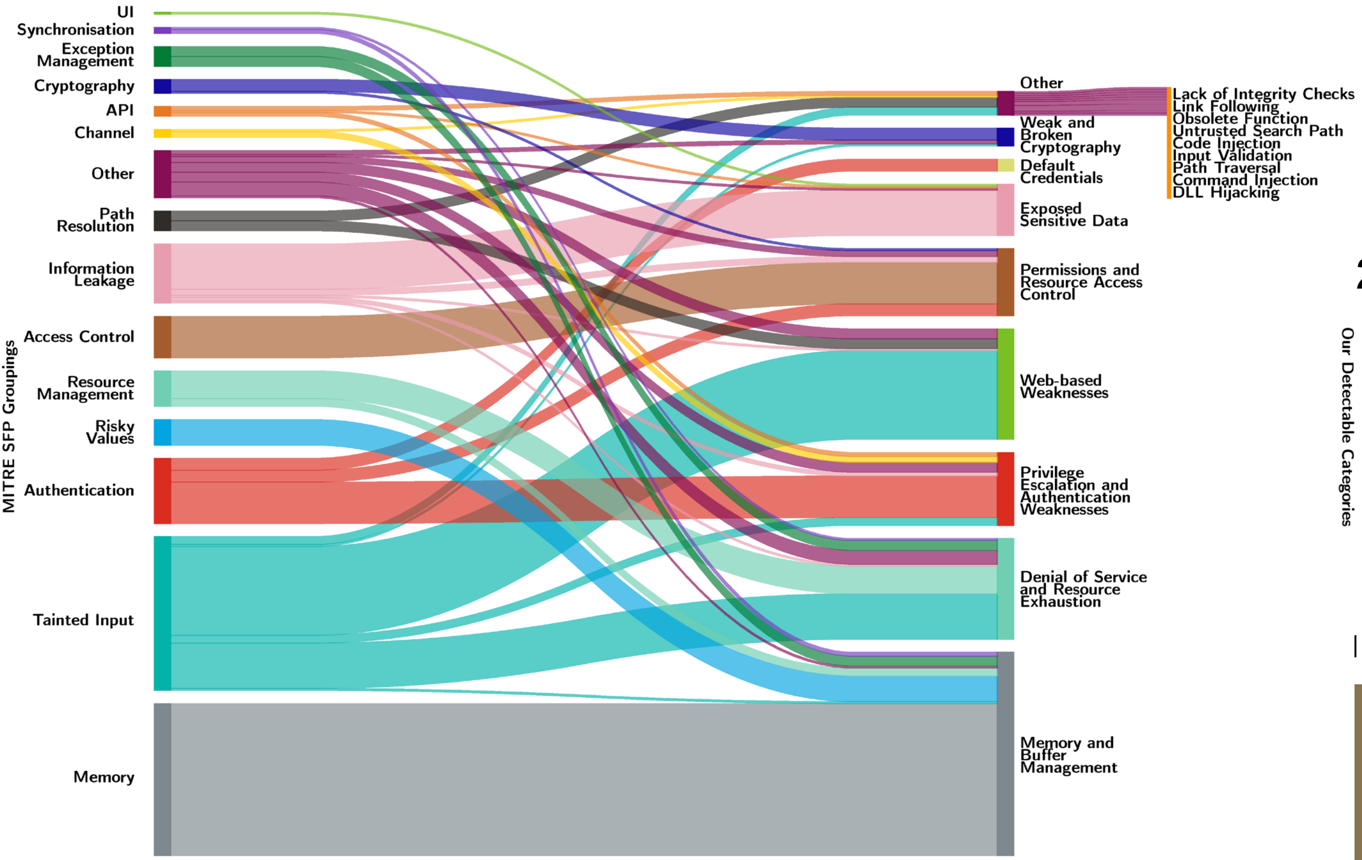
What new ICS Vulnerabilities are on the Horizon?



2020

vulnerabilities

What new ICS Vulnerabilities are on the Horizon?



20

lnerabilities

Defining testing strategies for our categories

- What is in the **capability** of an asset owner?

  Tools should be **automated and straight-forward** to use

- How can a **vendor** search for vulnerabilities?

  Preventing vulnerabilities being introduced in the first place

- What would **experts** apply?

  Reverse engineering of firmware, detailed trace analysis

- What **tooling already exists**?

  Allows an asset owner to baseline their estate and act

Defining testing strategies for our categories

- What is in the **capabili**

  Tools should be **auto**

- How can a **vendor** sea

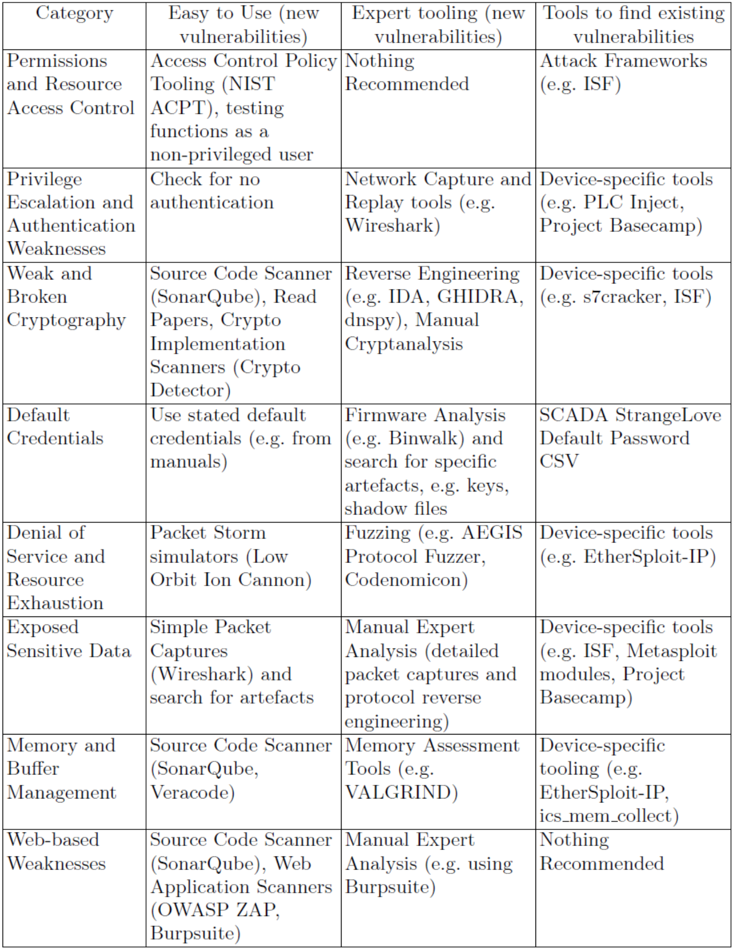
  Preventing vulnerab

- What would **experts** a

  Reverse engineering

- What **tooling already**

  Allows an asset owne



Category	Easy to Use (new vulnerabilities)	Expert tooling (new vulnerabilities)	Tools to find existing vulnerabilities
Permissions and Resource Access Control	Access Control Policy Tooling (NIST ACPT), testing functions as a non-privileged user	Nothing Recommended	Attack Frameworks (e.g. ISF)
Privilege Escalation and Authentication Weaknesses	Check for no authentication	Network Capture and Replay tools (e.g. Wireshark)	Device-specific tools (e.g. PLC Inject, Project Basecamp)
Weak and Broken Cryptography	Source Code Scanner (SonarQube), Read Papers, Crypto Implementation Scanners (Crypto Detector)	Reverse Engineering (e.g. IDA, GHIDRA, dnspy), Manual Cryptanalysis	Device-specific tools (e.g. s7cracker, ISF)
Default Credentials	Use stated default credentials (e.g. from manuals)	Firmware Analysis (e.g. Binwalk) and search for specific artefacts, e.g. keys, shadow files	SCADA StrangeLove Default Password CSV
Denial of Service and Resource Exhaustion	Packet Storm simulators (Low Orbit Ion Cannon)	Fuzzing (e.g. AEGIS Protocol Fuzzer, Codenomicon)	Device-specific tools (e.g. EtherSploit-IP)
Exposed Sensitive Data	Simple Packet Captures (Wireshark) and search for artefacts	Manual Expert Analysis (detailed packet captures and protocol reverse engineering)	Device-specific tools (e.g. ISF, Metasploit modules, Project Basecamp)
Memory and Buffer Management	Source Code Scanner (SonarQube, Veracode)	Memory Assessment Tools (e.g. VALGRIND)	Device-specific tooling (e.g. EtherSploit-IP, ics_mem_collect)
Web-based Weaknesses	Source Code Scanner (SonarQube), Web Application Scanners (OWASP ZAP, Burpsuite)	Manual Expert Analysis (e.g. using Burpsuite)	Nothing Recommended

Testing our Categories on ICS Equipment

- Tested **5** ICS Devices

  3 PLCs (Siemens and ABB), **2** HMIs (Phoenix Contact and Siemens)

- **Applied testing strategies** to detect **new** vulnerabilities

  CVE-2020-7592: Cleartext Transmission of Data in Siemens HMIs (Information Leakage)

  Open Redirect on a Siemens S7 PLC Web Administration Tool

  Denial of Service on a PLC Web Portal

  Authentication Bypass on a PLC

  Denial of Service and **Default Credentials** on a HMI

Responsible disclosure with the Vendors

- All testing was conducted in March 2020 – April 2020
- Vulnerabilities reported to the **vendors** in April 2020
- Siemens:
  - Open Redirect – leftover issue from a previous CVE (CVE-2015-1048) (GET vs POST) – **S7-1200**
  - “Users should follow the OT Security Guidance” – **Authentication Bypass**
  - No resolution yet – CVE-2020-7592 – **Siemens HMIs – Exposed Sensitive Data**
- Phoenix Contact: new CVE to be issued (**DoS**) and manual updated (**Default Credentials**)
- ABB still triaging
- Found additional flaws in MacOS and Firefox during impact analysis

Conclusion

Dataset now live:
uob-ritics.github.io

- ICS has **important differences** to standard IT
  Specifically the types of vulnerabilities and how they can be detected

- Analysed **9 years** of ICS vulnerability reports
- Carried out **trend analysis** and **defined 8 new detection-focused categories**

- Assess **testing strategies** to support asset owners
- Find **4 new critical vulnerabilities** in ICS equipment

- **Validate** our categories using 6 months of new data, **demonstrating** their effectiveness and **capability**

UNIVERSITY OF
BIRMINGHAM

RITICS

CENTRE FOR
CYBER SECURITY
AND PRIVACY

UKRRIN
UK RAIL RESEARCH AND
INNOVATION NETWORK

Learning from Vulnerabilities – Categorising, Understanding and Detecting Weaknesses in ICS

Richard J. Thomas and Tom Chothia, University of Birmingham
R.J.Thomas@cs.bham.ac.uk