CySER The Complexities of Hierarchical Software Quality Assurance Models

March 4, 2024
Dr. Clemente Izurieta
Professor of Computer Science
Software Engineering and Cybersecurity Laboratory (SECL)
Montana State University

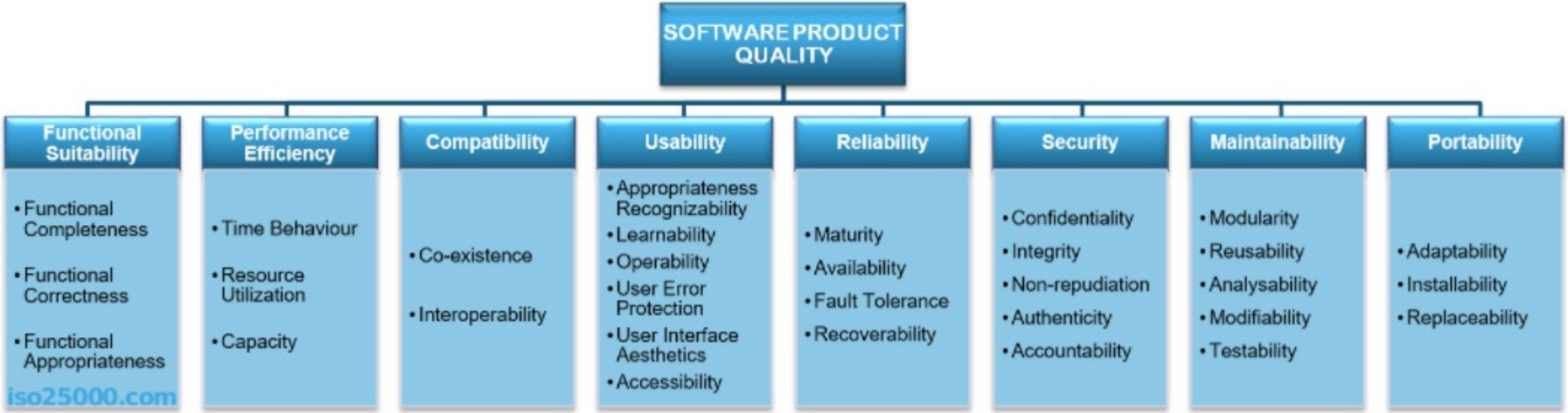
unclassified

Participants:

Institutional PI: Dr. Clemente Izurieta
ROTC Air Force: Lieutenant Colonel Zachary A. Hegedish
ROTC Army: Lieutenant Colonel Christopher L’Heureux
Graduate Research Assistant: Yvette Hastings

2021-2022 Academic year: 4 Air Force cadets
2022-2023 Academic year: 2 Air Force and 2 Army cadets
2023-2024 Academic year: 2 Air Force, 1 Army, 1 civilian

ISO 25K



SOFTWARE PRODUCT QUALITY

Functional Suitability

- Functional Completeness
- Functional Correctness
- Functional Appropriateness

Performance Efficiency

- Time Behaviour
- Resource Utilization
- Capacity

Compatibility

- Co-existence
- Interoperability

Usability

- Appropriateness Recognizability
- Learnability
- Operability
- User Error Protection
- User Interface Aesthetics
- Accessibility

Reliability

- Maturity
- Availability
- Fault Tolerance
- Recoverability

Security

- Confidentiality
- Integrity
- Non-repudiation
- Authenticity
- Accountability

Maintainability

- Modularity
- Reusability
- Analysability
- Modifiability
- Testability

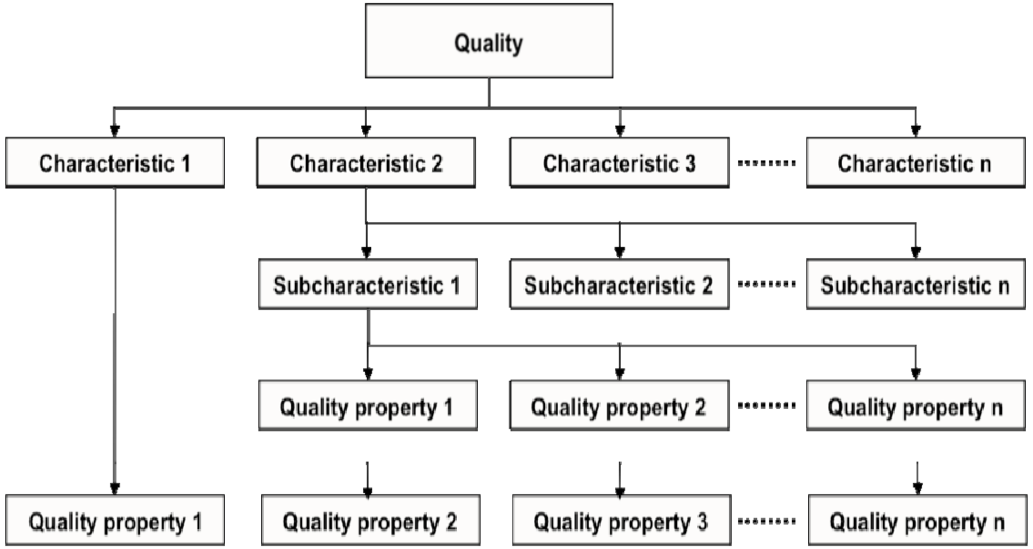
Portability

- Adaptability
- Installability
- Replaceability

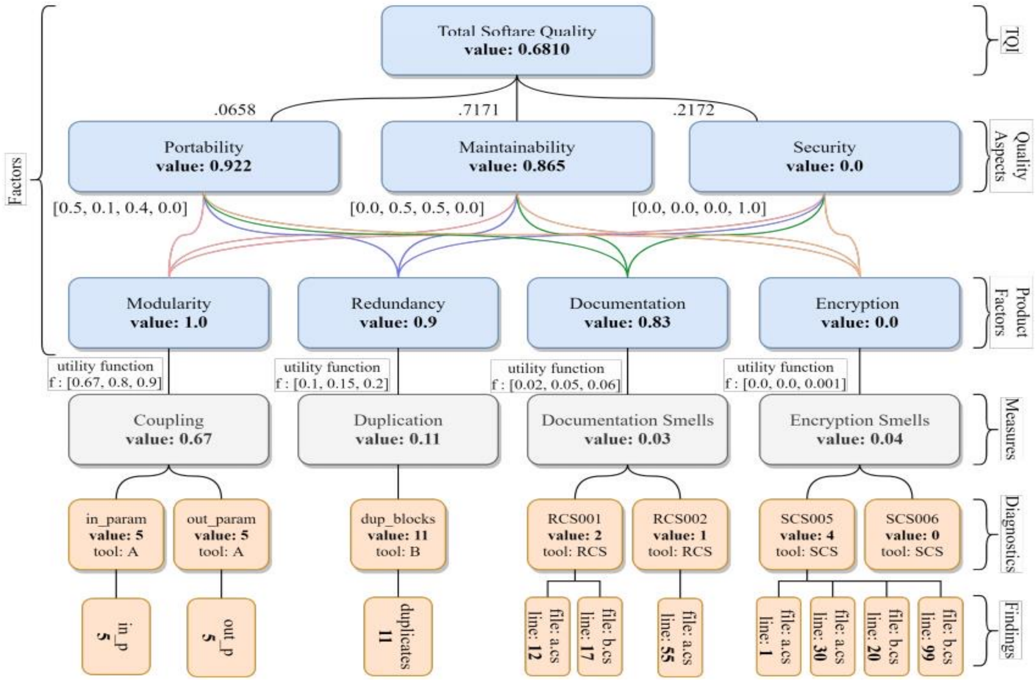
iso25000.com

Hierarchical Software QA Modeling

Theoretical



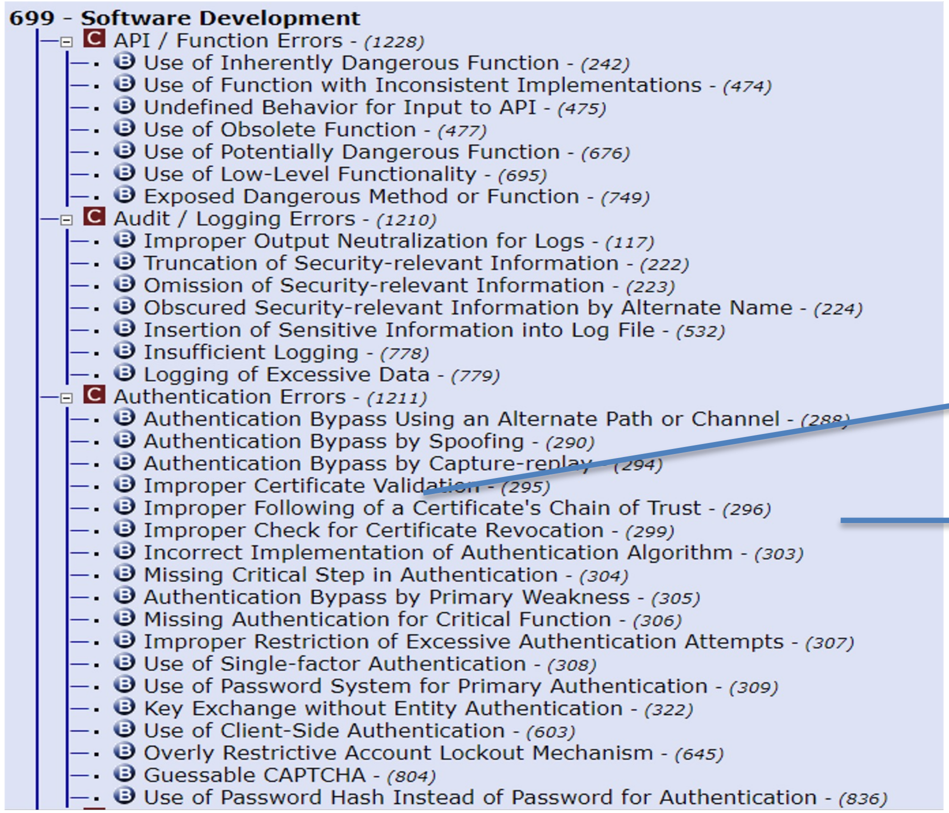
Operational



Standards
ISO/IEC 9126:2001
ISO/IEC 25010:2011
NIST 800-53/82
RMF (Risk Management Framework)

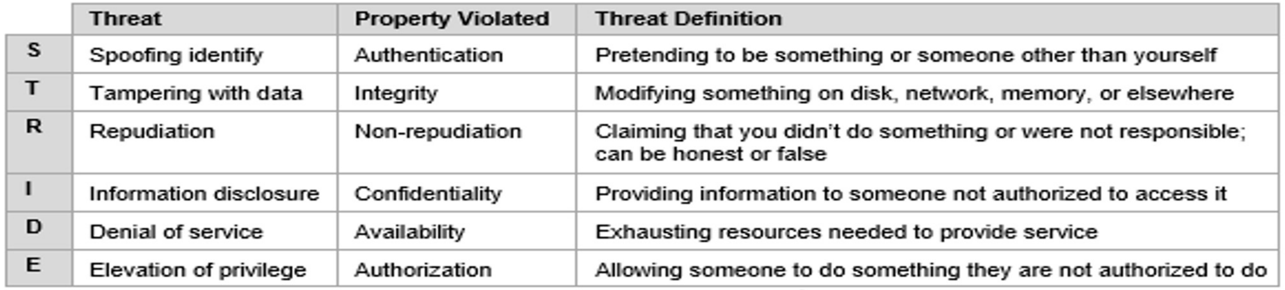
Quamoco (2012 Wagner et al.)
Qatch (2017 Miltiades et al.)
PIQUE (2020 SEL MSU)

CWE-699 View Structure

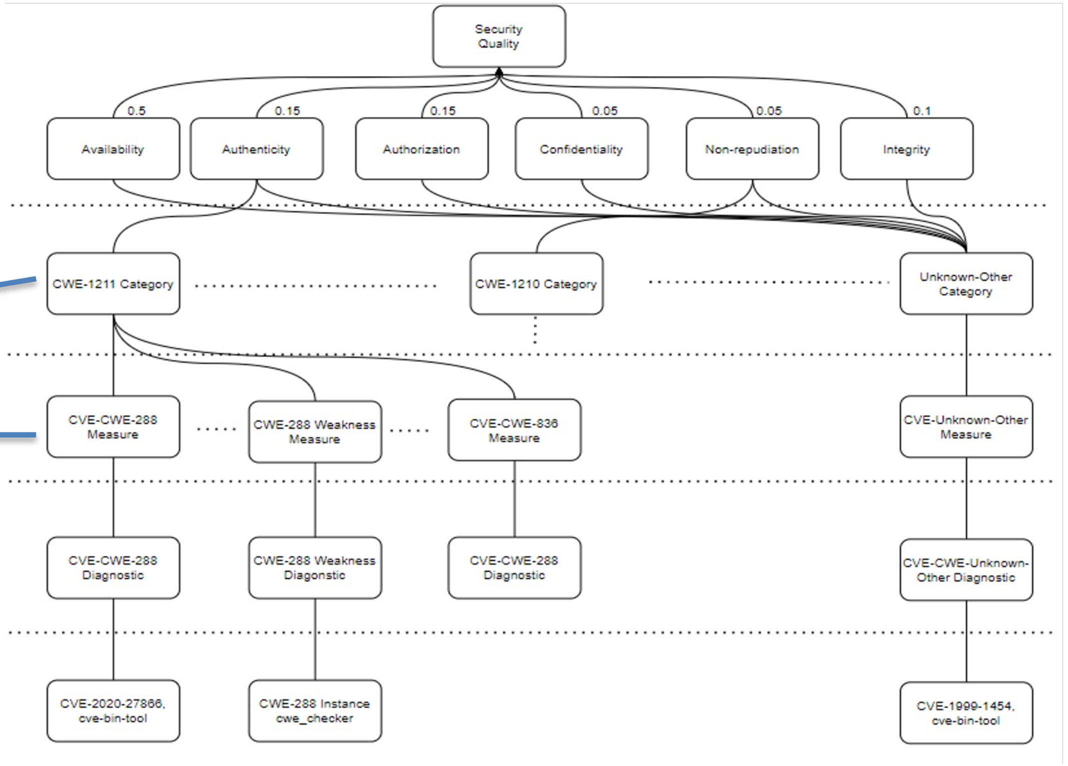


699 - Software Development
- API / Function Errors - *(1228)*
  - Use of Inherently Dangerous Function - *(242)*
  - Use of Function with Inconsistent Implementations - *(474)*
  - Undefined Behavior for Input to API - *(475)*
  - Use of Obsolete Function - *(477)*
  - Use of Potentially Dangerous Function - *(676)*
  - Use of Low-Level Functionality - *(695)*
  - Exposed Dangerous Method or Function - *(749)*
- Audit / Logging Errors - *(1210)*
  - Improper Output Neutralization for Logs - *(117)*
  - Truncation of Security-relevant Information - *(222)*
  - Omission of Security-relevant Information - *(223)*
  - Obscured Security-relevant Information by Alternate Name - *(224)*
  - Insertion of Sensitive Information into Log File - *(532)*
  - Insufficient Logging - *(778)*
  - Logging of Excessive Data - *(779)*
- Authentication Errors - *(1211)*
  - Authentication Bypass Using an Alternate Path or Channel - *(288)*
  - Authentication Bypass by Spoofing - *(290)*
  - Authentication Bypass by Capture-replay - *(294)*
  - Improper Certificate Validation - *(295)*
  - Improper Following of a Certificate's Chain of Trust - *(296)*
  - Improper Check for Certificate Revocation - *(299)*
  - Incorrect Implementation of Authentication Algorithm - *(303)*
  - Missing Critical Step in Authentication - *(304)*
  - Authentication Bypass by Primary Weakness - *(305)*
  - Missing Authentication for Critical Function - *(306)*
  - Improper Restriction of Excessive Authentication Attempts - *(307)*
  - Use of Single-factor Authentication - *(308)*
  - Use of Password System for Primary Authentication - *(309)*
  - Key Exchange without Entity Authentication - *(322)*
  - Use of Client-Side Authentication - *(603)*
  - Overly Restrictive Account Lockout Mechanism - *(645)*
  - Guessable CAPTCHA - *(804)*
  - Use of Password Hash Instead of Password for Authentication - *(836)*

Microsoft STRIDE



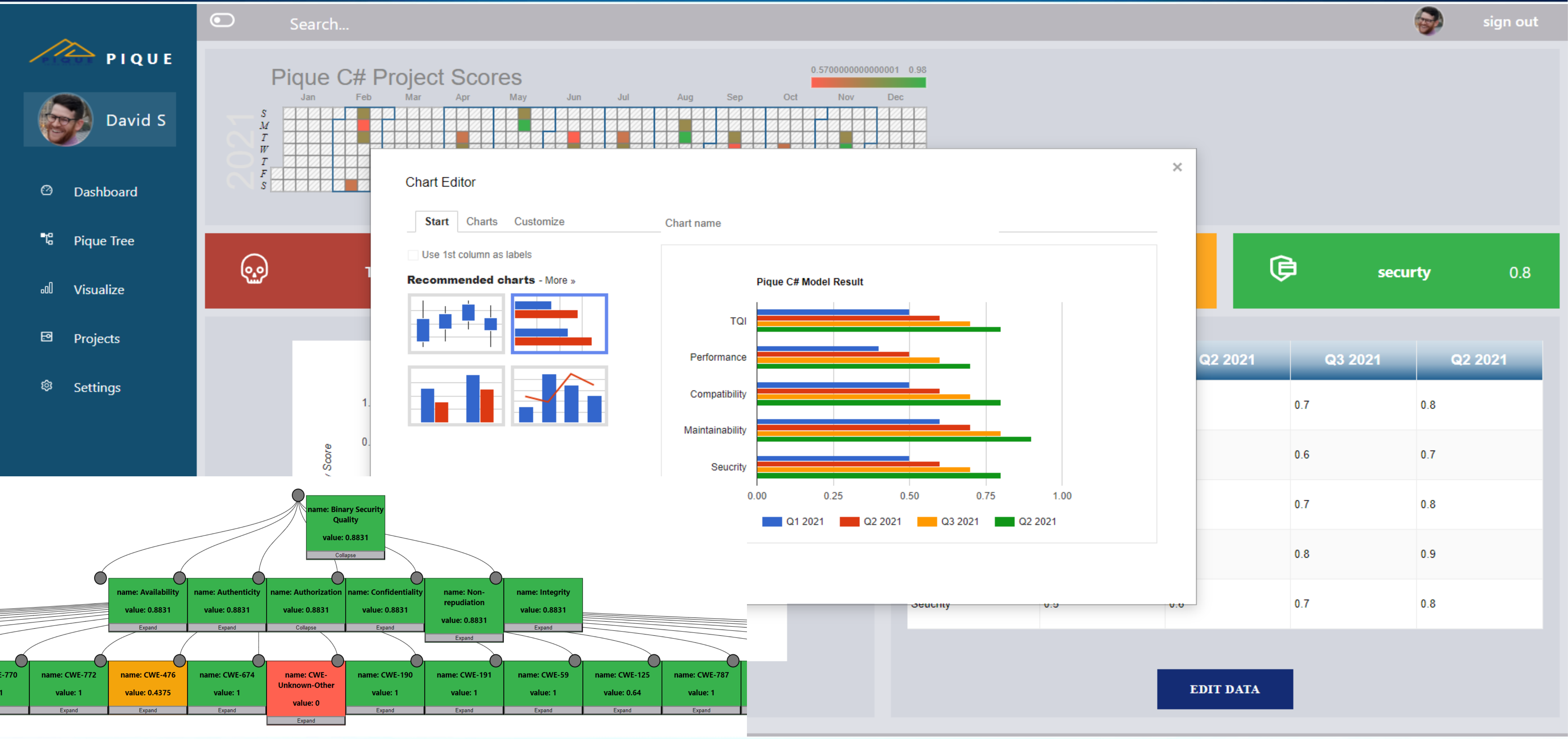
	Threat	Property Violated	Threat Definition
S	Spoofing identify	Authentication	Pretending to be something or someone other than yourself
T	Tampering with data	Integrity	Modifying something on disk, network, memory, or elsewhere
R	Repudiation	Non-repudiation	Claiming that you didn't do something or were not responsible; can be honest or false
I	Information disclosure	Confidentiality	Providing information to someone not authorized to access it
D	Denial of service	Availability	Exhausting resources needed to provide service
E	Elevation of privilege	Authorization	Allowing someone to do something they are not authorized to do



PIQUE Models

- Pique-Bin (INL, DHS)
- Pique-C# (CERL Army, Air Force)
- Pique-C#-Sec (CERL Army, Air Force, DHS)
- Pique-Azure (DHS)
- Pique-C++ (DHS)
- ***Pique-Cloud (DHS)***
- ***Pique-ICS (DHS)***

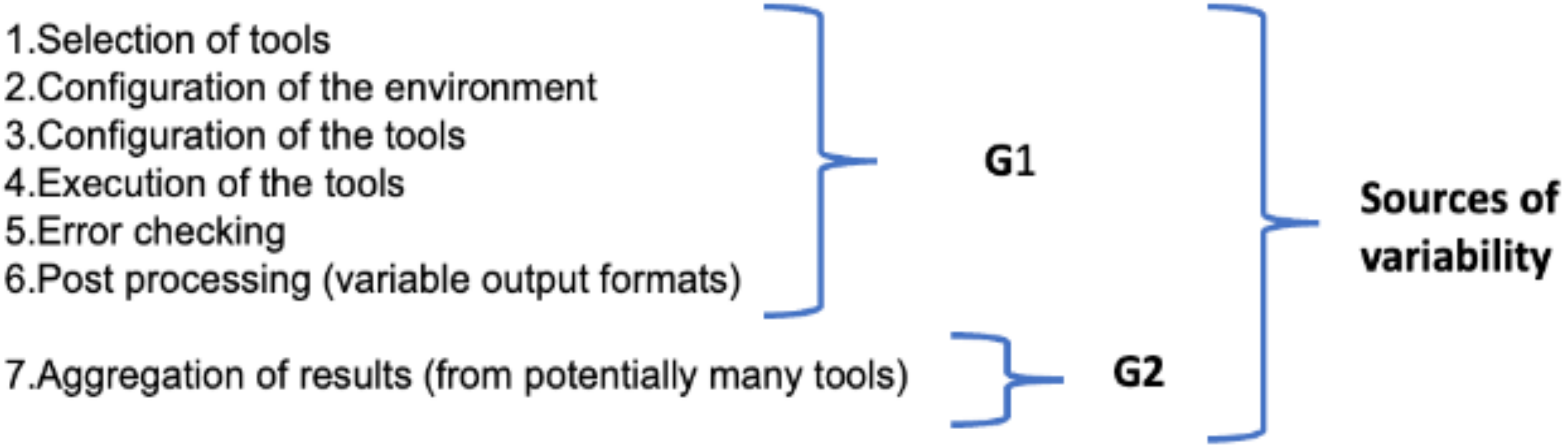
PIQUE
SOFTWARE ENGINEERING LABS


Search...
sign out
PIQUE
David S
Dashboard
Pique Tree
Visualize
Projects
Settings
Pique C# Project Scores
0.5700000000000001 0.98
2021
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
S M T W T F S
securty 0.8
Q2 2021 Q3 2021 Q2 2021
0.7 0.8
0.6 0.7
0.7 0.8
0.8 0.9
Secrity 0.5 0.6 0.7 0.8
EDIT DATA
Chart Editor
Start Charts Customize
Chart name
Use 1st column as labels
Recommended charts - More »
Pique C# Model Result
TQI
Performance
Compatibility
Maintainability
Seucrity
0.00 0.25 0.50 0.75 1.00
Q1 2021 Q2 2021 Q3 2021 Q2 2021
name: Binary Security Quality
value: 0.8831
Collapse
name: Availability value: 0.8831 Expand
name: Authenticity value: 0.8831 Expand
name: Authorization value: 0.8831 Collapse
name: Confidentiality value: 0.8831 Expand
name: Non-repudiation value: 0.8831 Expand
name: Integrity value: 0.8831 Expand
name: CWE-770 value: 1
name: CWE-772 value: 1 Expand
name: CWE-476 value: 0.4375 Expand
name: CWE-674 value: 1 Expand
name: CWE-Unknown-Other value: 0 Expand
name: CWE-190 value: 1 Expand
name: CWE-191 value: 1 Expand
name: CWE-59 value: 1 Expand
name: CWE-125 value: 0.64 Expand
name: CWE-787 value: 1 Expand

Diversity of Sources

- Variability associated with diverse sources of information is problematic:
  - data from multiple sources leads to the propagation of inconsistencies and errors
  - Accuracy and trustworthiness is hampered
  - We acknowledge that the variability inherent in vendors, tool versions, third party software, and host environments significantly influences the outcomes of security assessments
  - How do we normalize data?
  - How do we aggregate data?

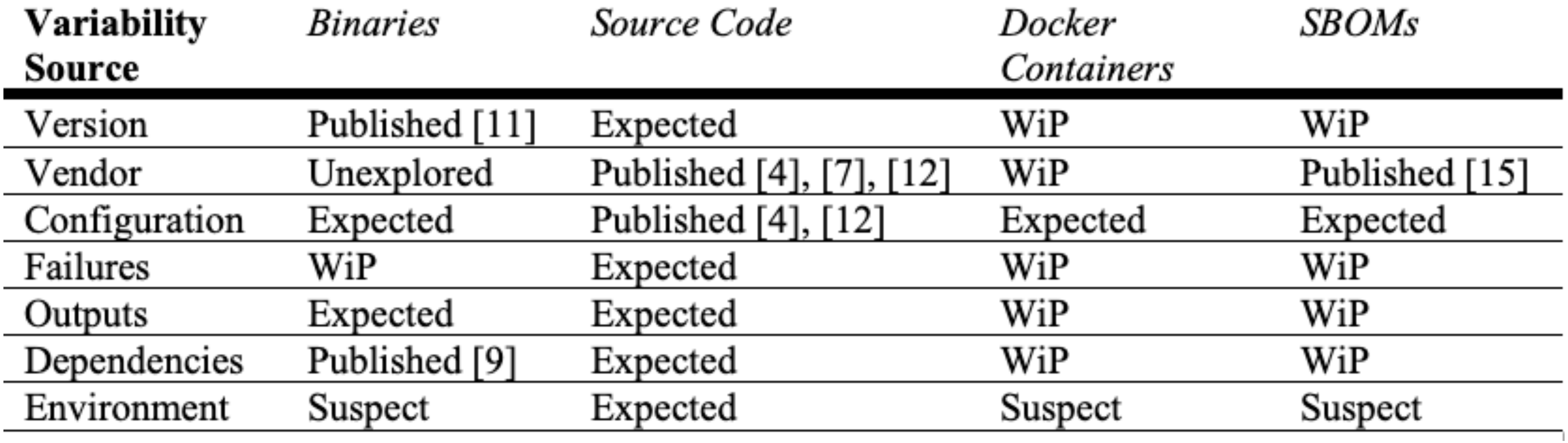
Diversity of Sources



1.Selection of tools
2.Configuration of the environment
3.Configuration of the tools
4.Execution of the tools
5.Error checking
6.Post processing (variable output formats)

7.Aggregation of results (from potentially many tools)

G1: Report on the high variability of SATs.
G2: Report on techniques used to aggregate results from multiple sources



Variability Source	*Binaries*	*Source Code*	*Docker Containers*	*SBOMs*
Version	Published [11]	Expected	WiP	WiP
Vendor	Unexplored	Published [4], [7], [12]	WiP	Published [15]
Configuration	Expected	Published [4], [12]	Expected	Expected
Failures	WiP	Expected	WiP	WiP
Outputs	Expected	Expected	WiP	WiP
Dependencies	Published [9]	Expected	WiP	WiP
Environment	Suspect	Expected	Suspect	Suspect

Diversity of Sources

G1 focuses on delineating the problem of reliance on one version of a SAT (e.g., the most recent version of the tool).

G2 offers an unbiased, tool-agnostic solution that we have developed to facilitate aggregating tool findings from multiple sources.

Experimental Methods

G1: We focus on experimentation done on:

i) binary analysis tools CVE Binary Tool and CWE Checker

We evaluated 660 publicly accessible binaries sourced from a Kali Linux distribution with multiple versions of CWE Checker and CVE Binary Tool

ii) Docker Images analysis tools Grype and Trivy.

We evaluated a single version of each of 163 Docker Official Images (i.e., containers) using the SATs Grype and Trivy. We collected these Official Images from Docker Hub

Experimental Methods

G2: We report on a procedure we have developed to aggregate results from diverse SATs.

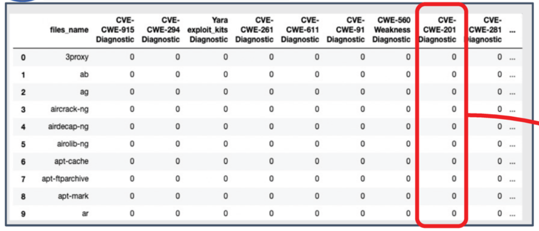
1 Software Artifacts

E.g., assemble collection of binaries

2 SATs

E.g., evaluate all binaries using cwe-checker and cve-bin-tool

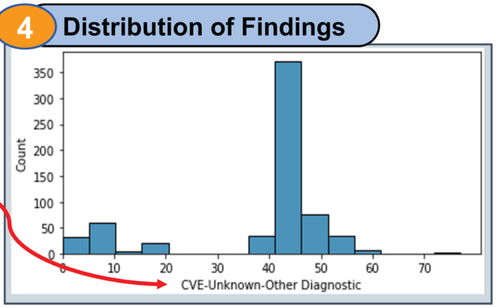
3 Aggregation File



	files_name	CVE-CWE-915 Diagnostic	CVE-CWE-294 Diagnostic	Yara exploit_kits Diagnostic	CVE-CWE-261 Diagnostic	CVE-CWE-611 Diagnostic	CVE-CWE-91 Diagnostic	CWE-560 Weakness Diagnostic	CVE-CWE-201 Diagnostic	CVE-CWE-281 Diagnostic	...
0	3proxy	0	0	0	0	0	0	0	0	0	...
1	ab	0	0	0	0	0	0	0	0	0	...
2	ag	0	0	0	0	0	0	0	0	0	...
3	aircrack-ng	0	0	0	0	0	0	0	0	0	...
4	airdecap-ng	0	0	0	0	0	0	0	0	0	...
5	airolib-ng	0	0	0	0	0	0	0	0	0	...
6	apt-cache	0	0	0	0	0	0	0	0	0	...
7	apt-ftparchive	0	0	0	0	0	0	0	0	0	...
8	apt-mark	0	0	0	0	0	0	0	0	0	...
9	ar	0	0	0	0	0	0	0	0	0	...

E.g., create file with counts of all CVEs and CWEs in each binary in collection

4 Distribution of Findings



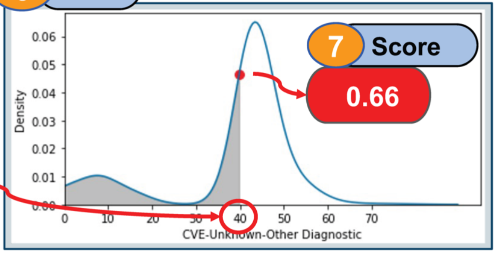
E.g., evaluate distribution of counts of each CVE in all binaries in collection (histogram for one CVE plotted above)

5 Evaluate New Artifact

Software Artifact Name	Finding Name	Count
New Artifact of Interest	CVE-Unknown-Other Diagnostic	40

E.g., the count of one finding in an end user's binary of interest

6 PDF

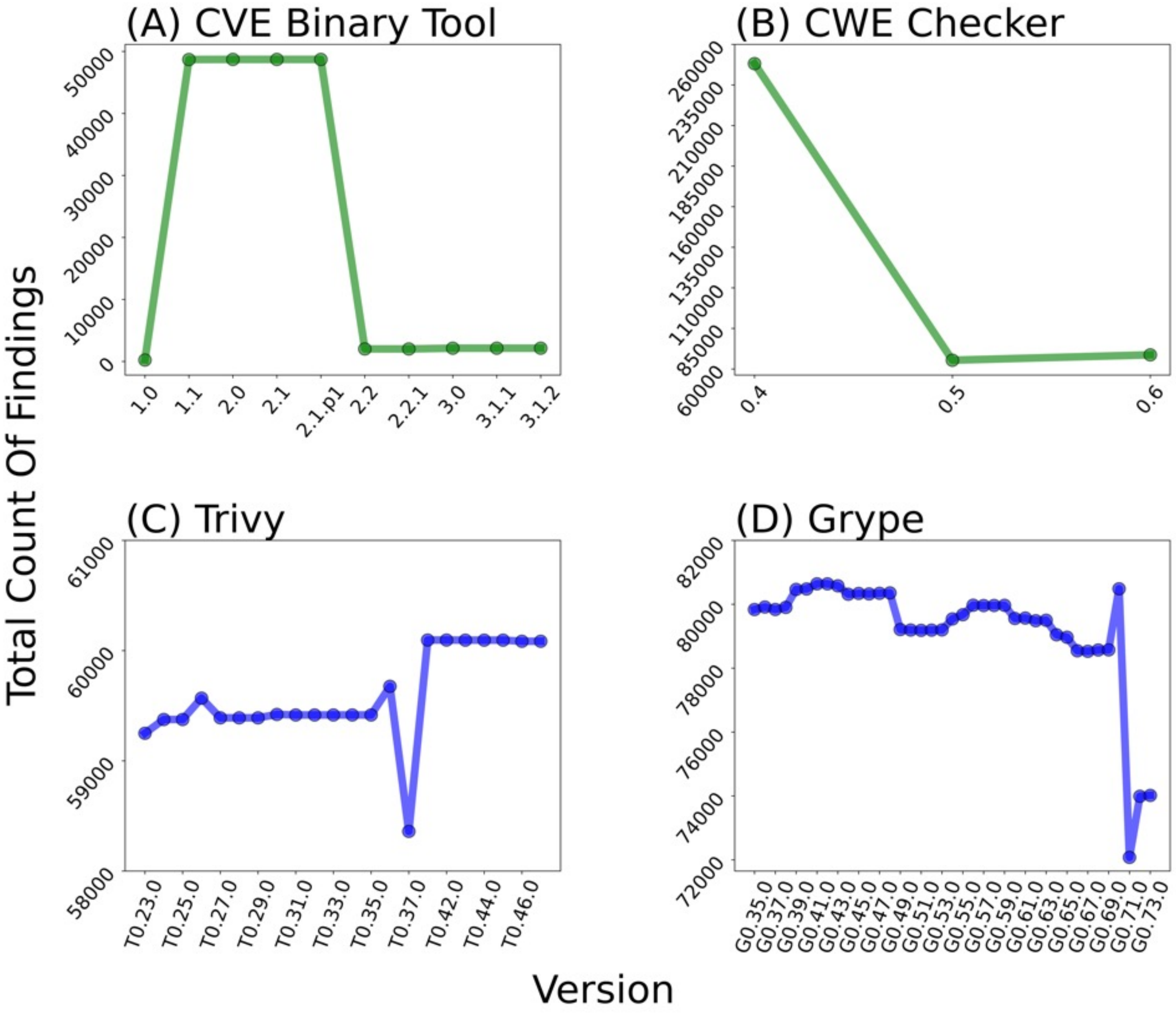


7 Score

0.66

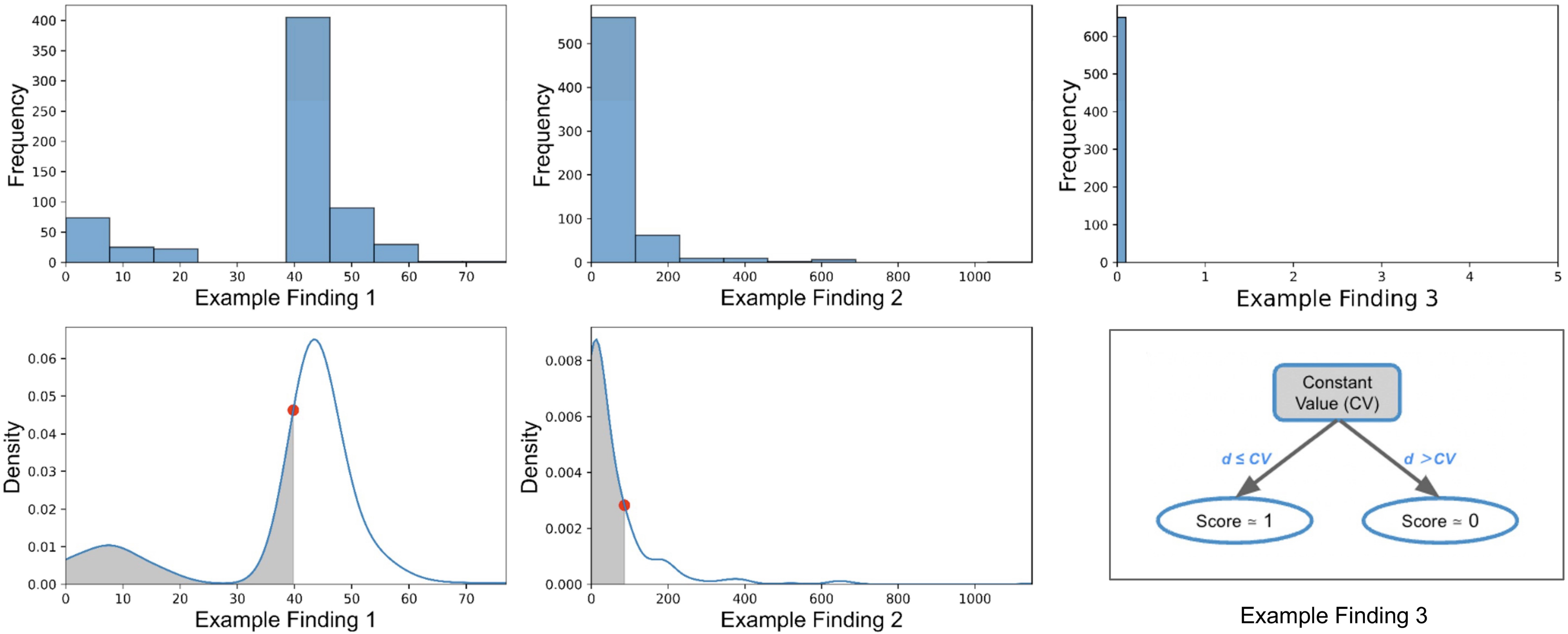
E.g., create a PDF from the counts of the CVE in all binaries in benchmark repository and score the count of a finding in an end user's binary of interest relative to all files in the collection

G1 Results



(A) CVE Binary Tool
(B) CWE Checker
(C) Trivy
(D) Grype
Total Count Of Findings
Version
0
10000
20000
30000
40000
50000
1.0
1.1
2.0
2.1
2.1.p1
2.2
2.2.1
3.0
3.1.1
3.1.2
60000
85000
110000
135000
160000
185000
210000
235000
260000
0.4
0.5
0.6
58000
59000
60000
61000
T0.23.0
T0.25.0
T0.27.0
T0.29.0
T0.31.0
T0.33.0
T0.35.0
T0.37.0
T0.42.0
T0.44.0
T0.46.0
72000
74000
76000
78000
80000
82000
G0.35.0
G0.37.0
G0.39.0
G0.41.0
G0.43.0
G0.45.0
G0.47.0
G0.49.0
G0.51.0
G0.53.0
G0.55.0
G0.57.0
G0.59.0
G0.61.0
G0.63.0
G0.65.0
G0.67.0
G0.69.0
G0.71.0
G0.73.0

G2 Results

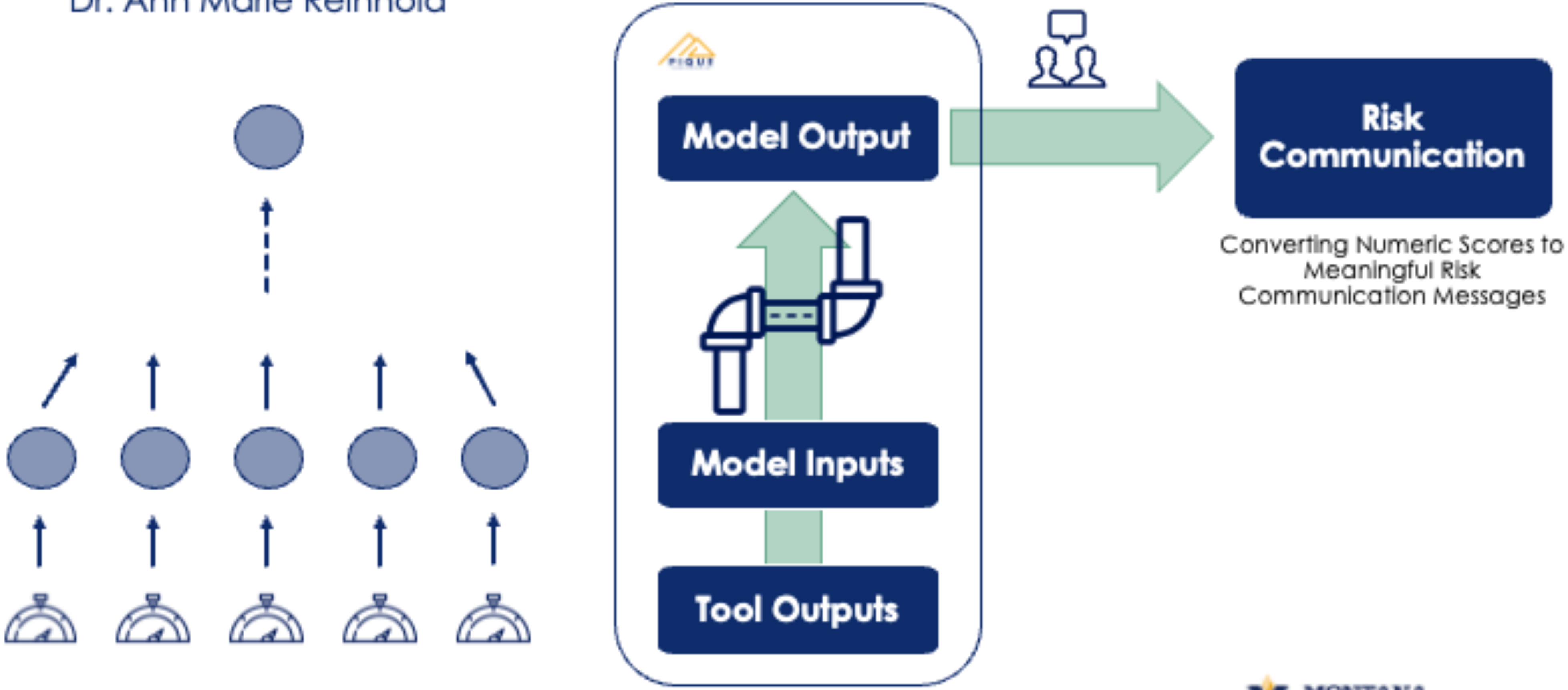

Frequency
400
350
300
250
200
150
100
50
0
0
10
20
30
40
50
60
70
Example Finding 1
Frequency
500
400
300
200
100
0
0
200
400
600
800
1000
Example Finding 2
Frequency
600
500
400
300
200
100
0
0
1
2
3
4
5
Example Finding 3
Density
0.06
0.05
0.04
0.03
0.02
0.01
0.00
0
10
20
30
40
50
60
70
Example Finding 1
Density
0.008
0.006
0.004
0.002
0.000
0
200
400
600
800
1000
Example Finding 2
Constant Value (CV)
d ≤ CV
d >CV
Score ≈ 1
Score ≈ 0
Example Finding 3

Lessons Learned and Futures

- Assessing these sources of variability simultaneously is too complex. Breaking each component down into more atomic components will facilitate understanding the nuances of each source of variability
- We offer a primary classification for sources of variability as a first step towards developing a taxonomy for classifying variability sources (e.g., vendor, version, environment)
- The solution that we present for aggregation has the benefit of being applicable for information across a range of sources where no oracles exist
- Sources of variability compound uncertainty. The compounding of uncertainty are inevitable side-effects of aggregation

Quality Assurance Pipeline

Dr. Ann Marie Reinhold



Model Output

Model Inputs

Tool Outputs

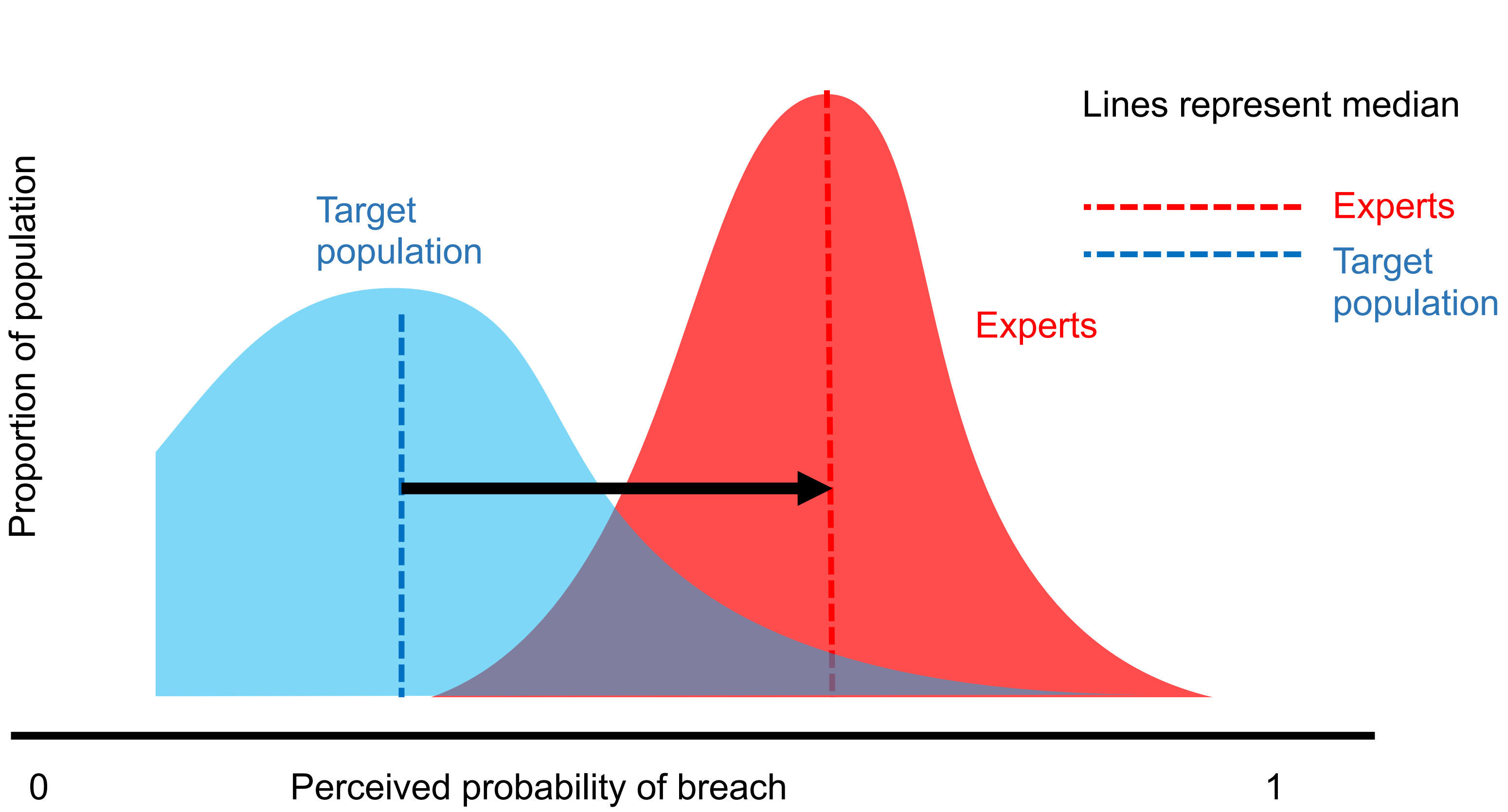
Risk Communication

Converting Numeric Scores to Meaningful Risk Communication Messages

Quality Assurance Pipeline

Risk Communication

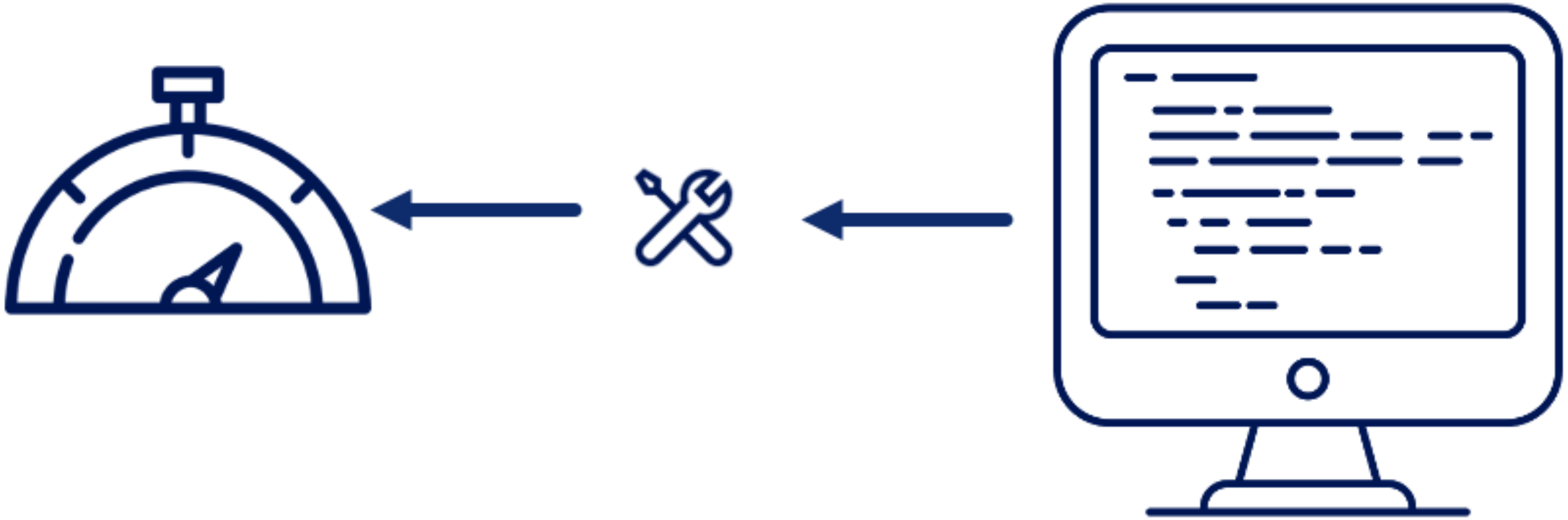
Converting Numeric Scores to Meaningful Risk Communication Messages



Closing the perception gap

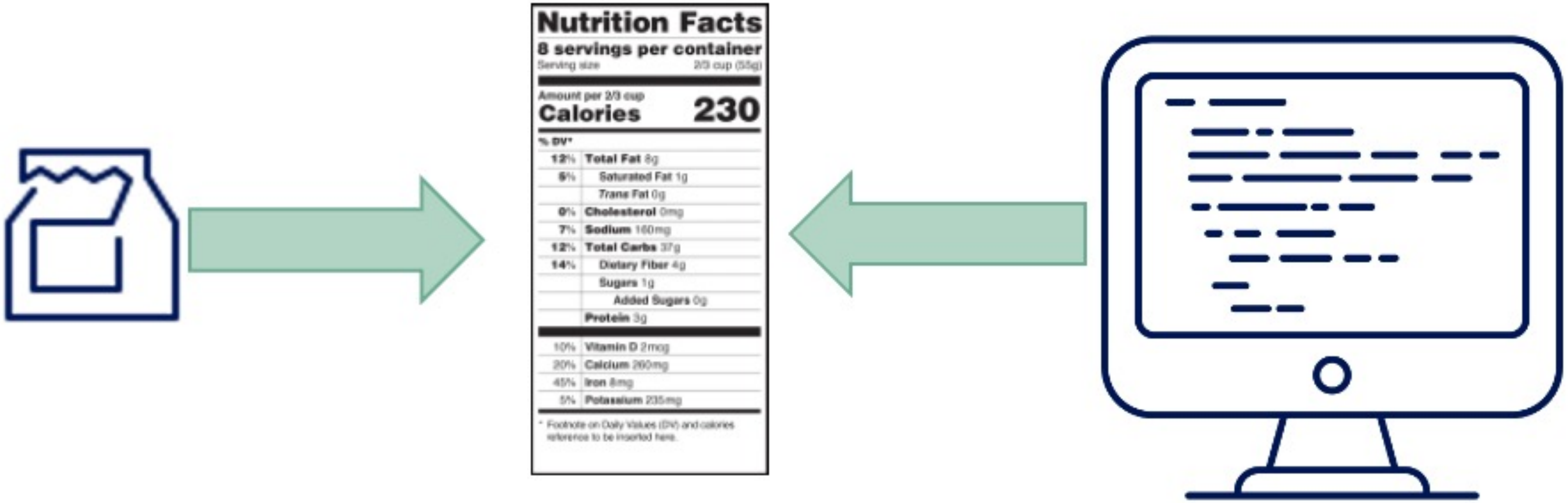
Software Bill Of Materials (SBOMs)

Eric O'Donoghue (MS Student)



Software Bill Of Materials (SBOMs)

Eric O'Donoghue (MS Student)



Nutrition Facts

8 servings per container

Serving size 2/3 cup (55g)

Amount per 2/3 cup

Calories 230

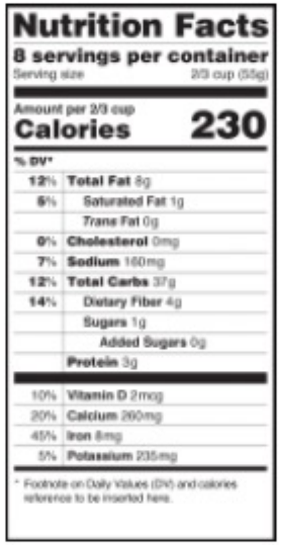
% DV*	
12%	**Total Fat** 8g
5%	Saturated Fat 1g
	Trans Fat 0g
0%	**Cholesterol** 0mg
7%	**Sodium** 160mg
12%	**Total Carbs** 37g
14%	Dietary Fiber 4g
	Sugars 1g
	Added Sugars 0g
	Protein 3g
10%	**Vitamin D** 2mcg
20%	**Calcium** 260mg
45%	**Iron** 8mg
5%	**Potassium** 235mg

* Footnote on Daily Values (DV) and calories reference to be inserted here.

A list of ingredients that make up software systems

Software Bill Of Materials (SBOMs)

Eric O'Donoghue (MS Student)



Nutrition Facts

8 servings per container

Serving size 2/3 cup (55g)

Amount per 2/3 cup

Calories 230

% DV*	
12%	**Total Fat** 8g
5%	Saturated Fat 1g
	Trans Fat 0g
0%	**Cholesterol** 0mg
7%	**Sodium** 160mg
12%	**Total Carbs** 37g
14%	Dietary Fiber 4g
	Sugars 1g
	Added Sugars 0g
	Protein 3g
10%	**Vitamin D** 2mcg
20%	**Calcium** 260mg
45%	**Iron** 8mg
5%	**Potassium** 235mg

* Footnote on Daily Values (DV) and calories reference to be inserted here.

- CycloneDX
- SWID
- SPDX

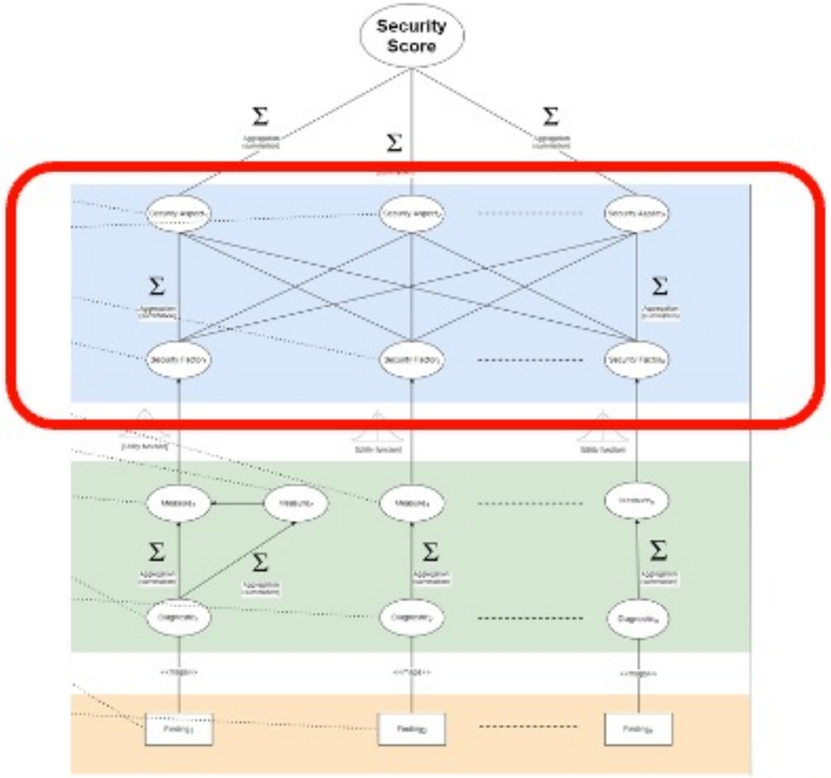
- Structural Quality of SBOM
- Security Assessment of the Contents

The Use of AI and ML to Improve Models

Dr. Bradley Whitaker

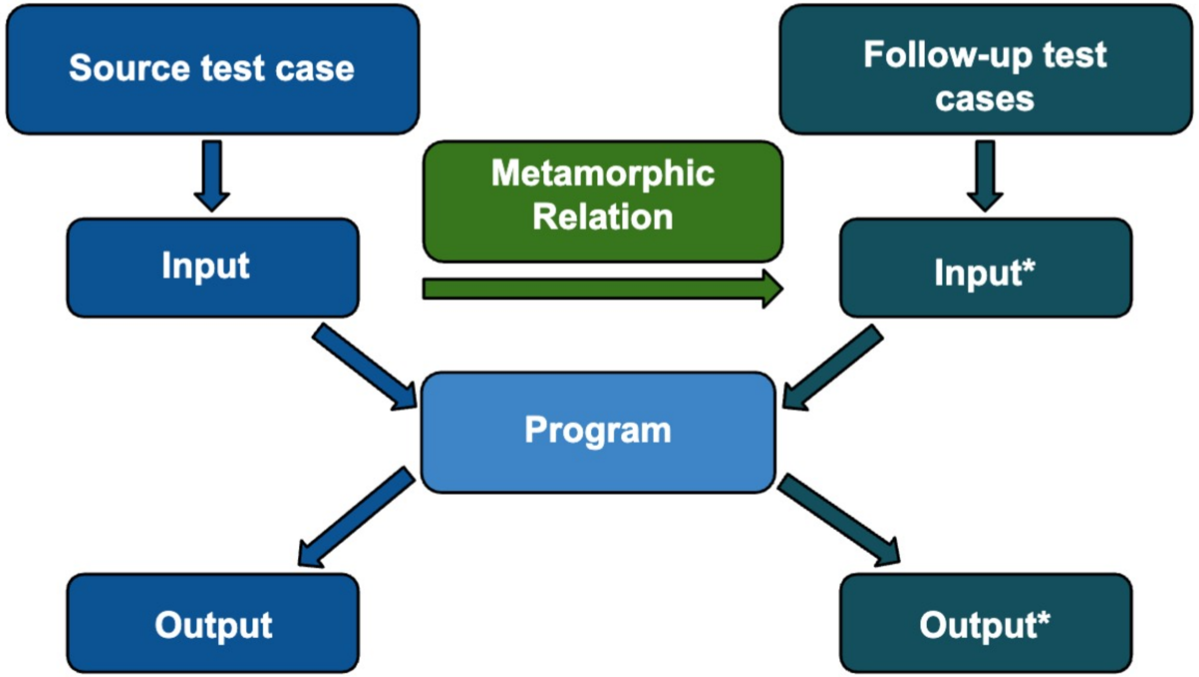
PIQUE

SOFTWARE ENGINEERING LABS



MONTANA STATE UNIVERSITY

Improving the confidence of machine learning models through improved software testing approaches

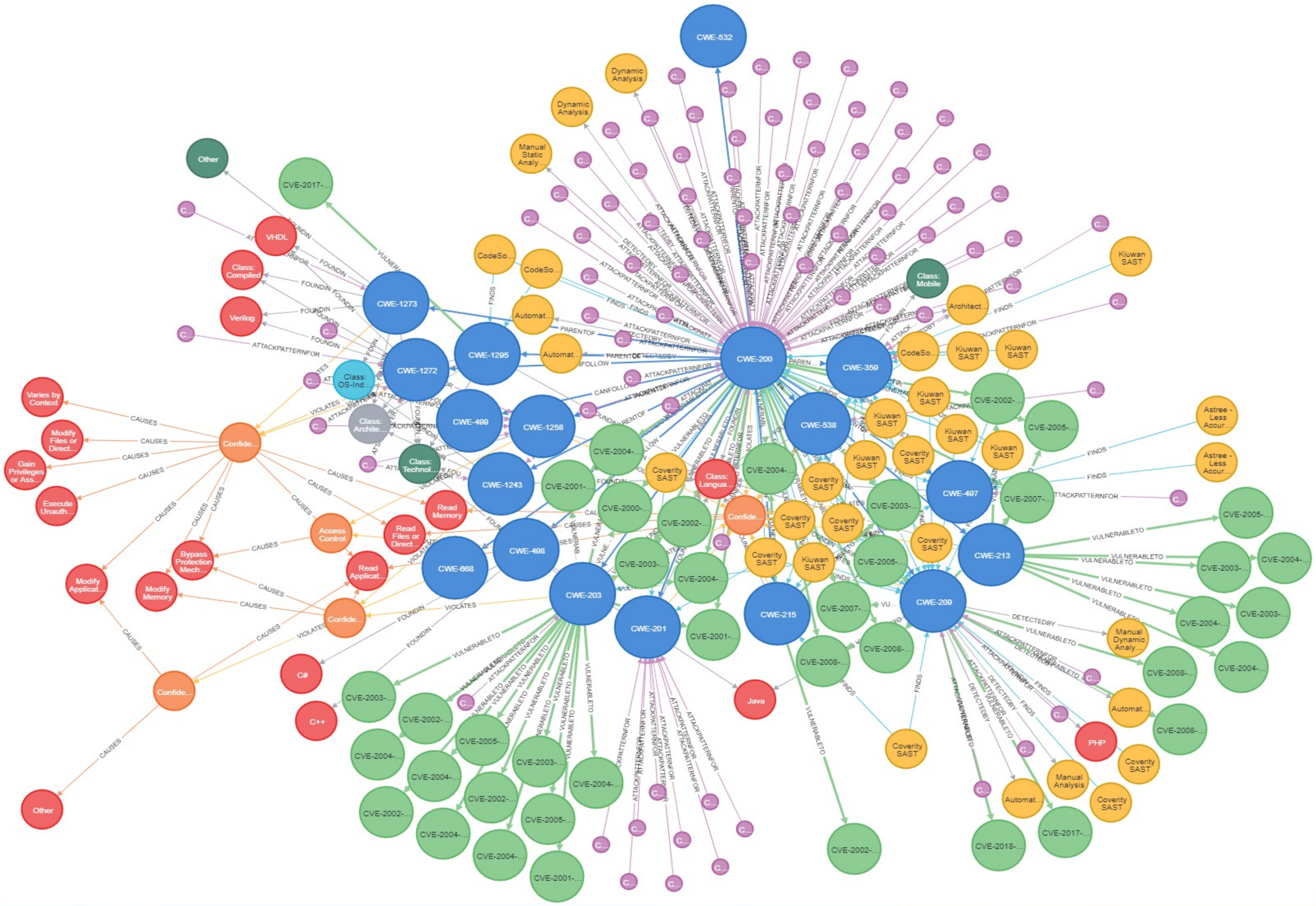


PIQUE

SOFTWARE ENGINEERING LABS

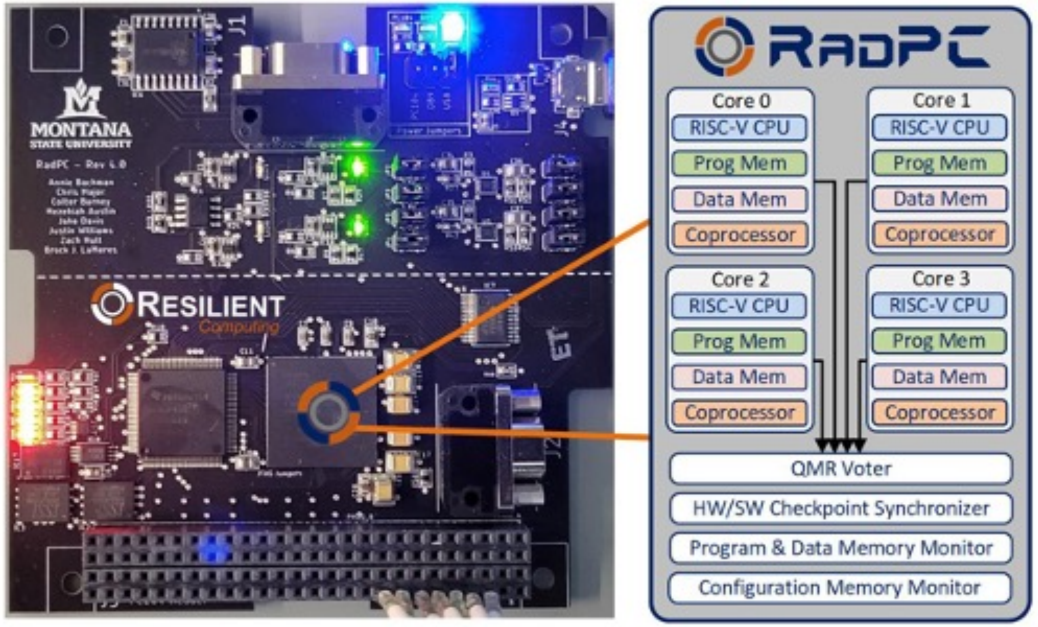
Decomposition of CWE-200

Identify security zones and sensitive sections of source code



Malware Detection Using Obfuscation of Opcodes in FPGAs

Dr. Brock LaMeres



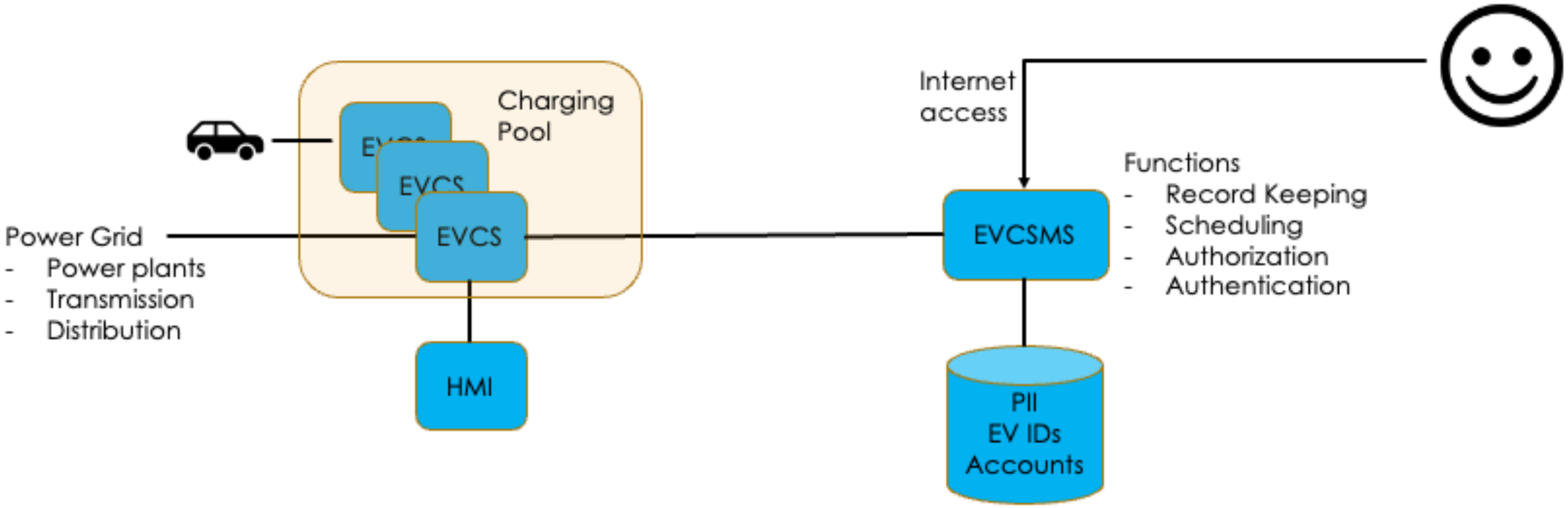
- We control the entire hardware description of the processor, thus we control the CPU implementation.
- This means we can assign random instruction codes for each core, which prevents malware from ever infecting more than one of the redundant computers

Near Future

- Goals: Detection, mitigation, guidance of Electrical Vehicle (EV) Infrastructure
  - Measurement of quality in EV infrastructure components
  - Develop repeatable and quantifiable processes for testing components
  - Identify test bed for solutions
  - Use hardware obfuscation techniques

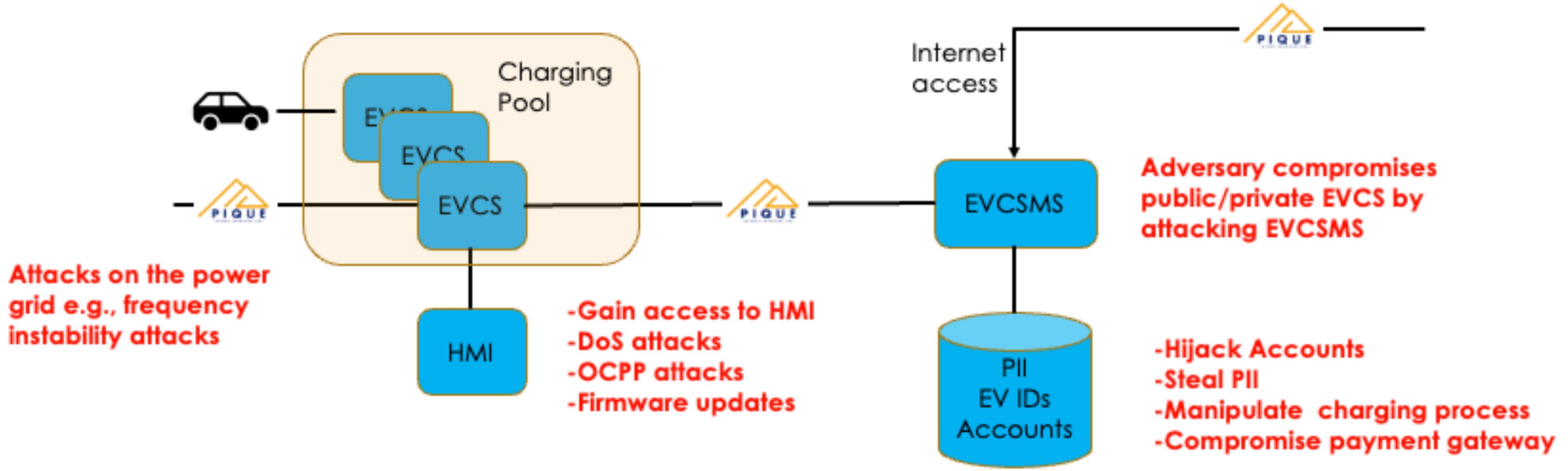
Pacific Northwest
NATIONAL LABORATORY

Electrical Vehicle Threat Models



EVCS: Electric Vehicle Charging Station
EVCSMS: EVCS Management Station (Back Office)
OCPP: Open Charge Point Protocol
HMI: Human Machine Interface

Electrical Vehicle Threat Models



Attacks on the power grid e.g., frequency instability attacks

-Gain access to HMI
-DoS attacks
-OCPP attacks
-Firmware updates

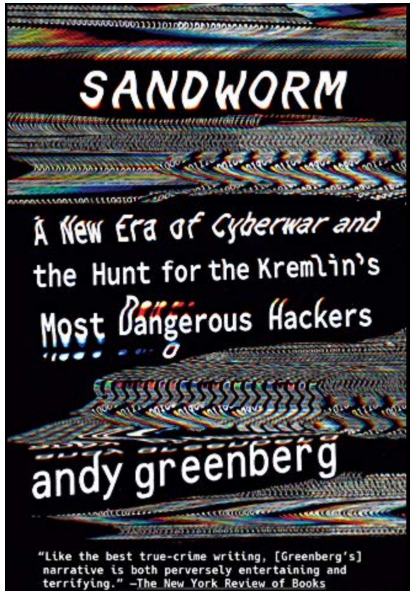
Adversary compromises public/private EVCS by attacking EVCSMS

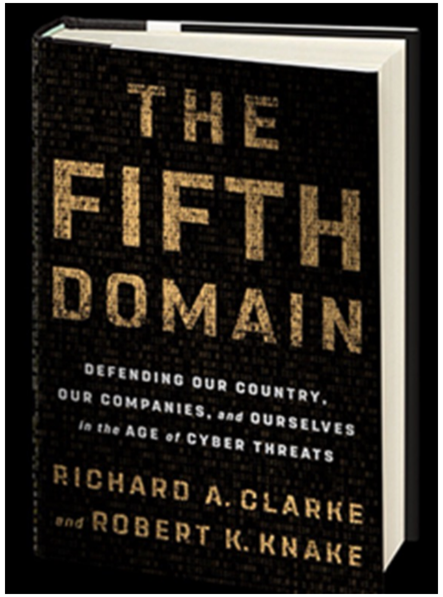
-Hijack Accounts
-Steal PII
-Manipulate charging process
-Compromise payment gateway

EVCS: Electric Vehicle Charging Station
EVCSMS: EVCS Management Station (Back Office)
OCPP: Open Charge Point Protocol
HMI: Human Machine Interface

Joyce Antler Jewish Radical Feminism
Her Face in the Mirror
Womanguides
Talking Back
Behind Every Choice Is a Story
Brenda
The Jewish Woman
Jewish Feminism and Intersectionality
Jewish and Female
Every Mother Is a Daughter
Judith Warner Perfect Madness
Beyond God the Father
Mary Daly Quintessence
Betty Friedan The Second Stage
The Quotable Jewish Woman
The Quotable Woman
Leo Rosten's Treasury of Jewish Quotations
What Is the Use of Jewish History?
New Jewish Feminism
The Other Mother
Weaving the Visions
The Coming of Lilith
Which Lilith?
A Breath of Life
Women in the Early Church
Suddenly Jewish

- *Book Club*
- *Independent study credit*
- *HackerCats club*


SANDWORM
A New Era of Cyberwar and the Hunt for the Kremlin's Most Dangerous Hackers
andy greenberg
"Like the best true-crime writing, [Greenberg's] narrative is both perversely entertaining and terrifying." —The New York Review of Books


THE FIFTH DOMAIN
DEFENDING OUR COUNTRY, OUR COMPANIES, and OURSELVES in the AGE of CYBER THREATS
RICHARD A. CLARKE and ROBERT K. KNAKE

Research Collaborations

https://www.montana.edu/cyber/

TechLink

HOPLITE INDUSTRIES

RESILIENT Computing

U.S. DEPARTMENT OF HOMELAND SECURITY

CYBERSECURITY & INFRASTRUCTURE SECURITY AGENCY CISA

CERL CONSTRUCTION ENGINEERING RESEARCH LABORATORY ERDC

MONTANA STATE UNIVERSITY APPLIED RESEARCH LABORATORY

wolfSSL

CYBERCORE integration center | INL Idaho National Laboratory

Pacific Northwest NATIONAL LABORATORY

Carnegie Mellon University Software Engineering Institute

WASHINGTON STATE UNIVERSITY

ROCHESTER INSTITUTE OF TECHNOLOGY R·I·T 1829

Education

- Associates degree in Cybersecurity (Gallatin College)
- MS in Cybersecurity
  - Board of Regents approved
  - Seeking CAE certification
- NSF REU program –Cybersecurity algorithms
- Griffiss/DoD program to train 4 ROTC cadets on a yearly basis before commissioning

Power of Collaboration