CWE/CAPEC User Experience Working Group Meeting

December 14, 2022

Agenda

This meeting is being recorded :-)

- **Housekeeping**
- **Primary topics**
  - Presentation: A Different View of CWE/CAPEC – Security Workbench by Jim Whitmore
  - CAPEC Content Presentation Filters Proposal
  - A Brief Look Ahead
    - User Stories to Come (January)
- **Reminders and Adjourn**

UEWG: Reminders

- **Mission:** Identifying areas where CWE/CAPEC content, rules, guidelines, and best practices must improve to better support stakeholder community, and work collaboratively to fix them
- **Periodic reporting of activities to CWE/CAPEC Board**
  - (next quarterly Board meeting TBD Q1-2023)
- **Please solicit participations from your contacts**
  - Contact: cwe@mitre.org & capec@mitre.org

Housekeeping

- **We are working to identify further UEWG opportunities and priorities for FY23**
  - CAPEC minor release 3.9 date: **January 24**
  - CWE minor release v4.10 date: **January 31**
  - Drafting user stories to publish on CWE/CAPEC Sites

- **Q1/Q2 Efforts**
  - User Persona - > User Stories
  - Determine how to best communicate and share user stories with the community
  - (Underway) Collaborative CWE/CAPEC content development space collaboration space

Topic #1

Security Workbench

A different view of CWE/CAPEC information

Jim Whitmore, Dickinson College

MITRE CWE USER EXPERIENCE WORKGROUP

SE WORKBENCH REVIEW

Jim Whitmore

Dec 14, 2022

TOPICS

- Background
- SE Workbench
- Summary
- Worked Example

CYBER & INFORMATION SECURITY
CURRICULUM OUTLINE

My view of Cyber & Information Security

Four Units:

1. Cyber Thinking: Concepts
2. Cyber Enterprises: Governance
3. Cyber Systems: Technologies
4. Advanced Topics: Security Engineering

Published in Response to NICE Framework RFC

https://www.nist.gov/system/files/documents/2020/01/30/031_NICE%20Framework%20Request%20for%20Comments_508.pdf

CYBER THINKING
THREATS, WEAKNESSES & VULNERABILITIES

THREATS

Threat = (Threat Agent, *Target of Attack*, *Method of Attack*)

A threat consists of an adverse action performed by a threat agent on an asset.

- **Adverse actions** are actions performed by a threat agent on an asset. These actions influence one or more properties of an asset from which that asset derives its value.
- **Examples of threats** are:
  - a hacker (with substantial expertise, standard equipment, and being paid to do so) remotely copying confidential files from a company network;
  - a worm seriously degrading the performance of a wide-area network;
  - a system administrator violating user privacy;
  - someone on the Internet listening in on confidential electronic communication.

Source: Common Criteria Part 1, General Model

9/6/2022 8

… a triad of security engineering

WEAKNESSES & VULNERABILITIES

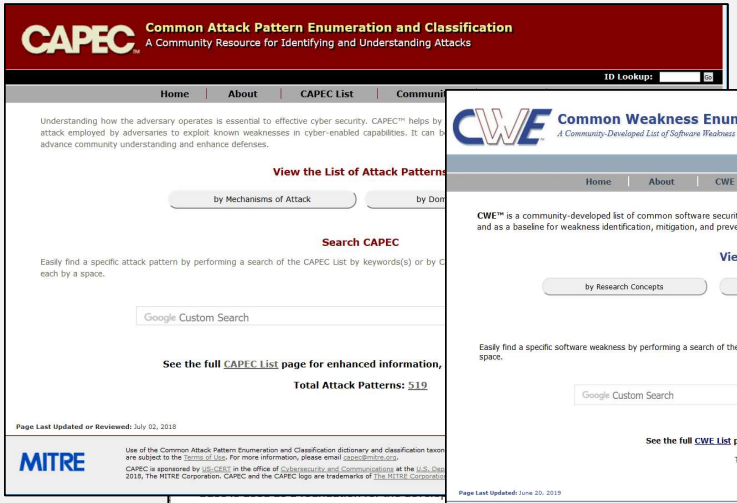
vulnerability	Weakness in the system that can be used to violate security in some environment
potential vulnerability	Suspected, but not confirmed, weakness. Suspicion is by virtue of a postulated attack path to violate the security of the system.
exploitable vulnerability	Weakness in the system that can be used to violate the security in the operational environment for the system
residual vulnerability	Weakness in the operational environment… that could be used to violate security by an attacker with greater attack potential than is anticipated in the operational environment

Source: Common Criteria Part 1, General Model

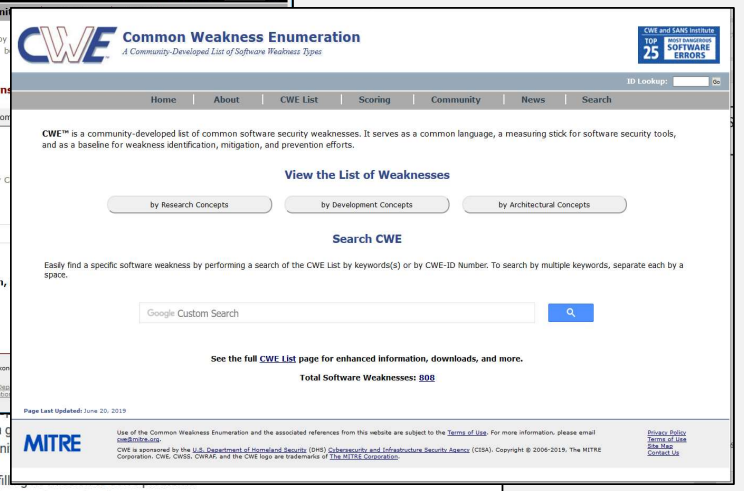
9/6/2022 9

NIST AND MITRE PROVIDE REFERENCE DOCUMENTS AND CATALOGS OF CONTROLS, ATTACKS, WEAKNESSES AND VULNERABILITIES.

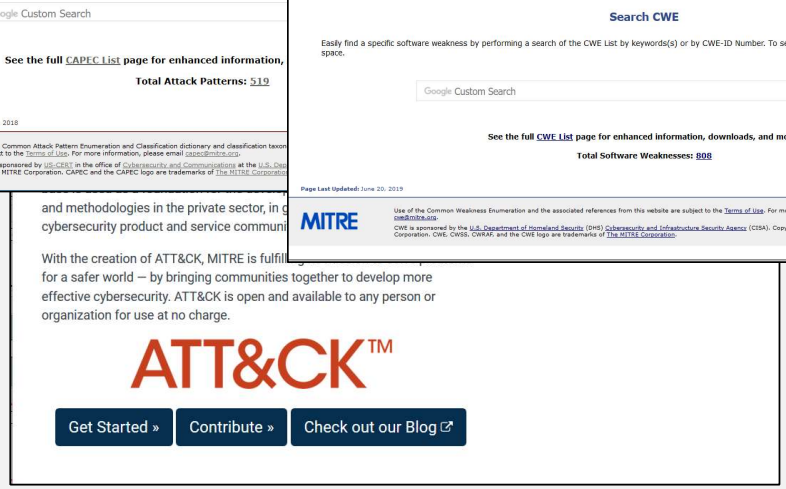
http://capec.mitre.org/



https://cwe.mitre.org/



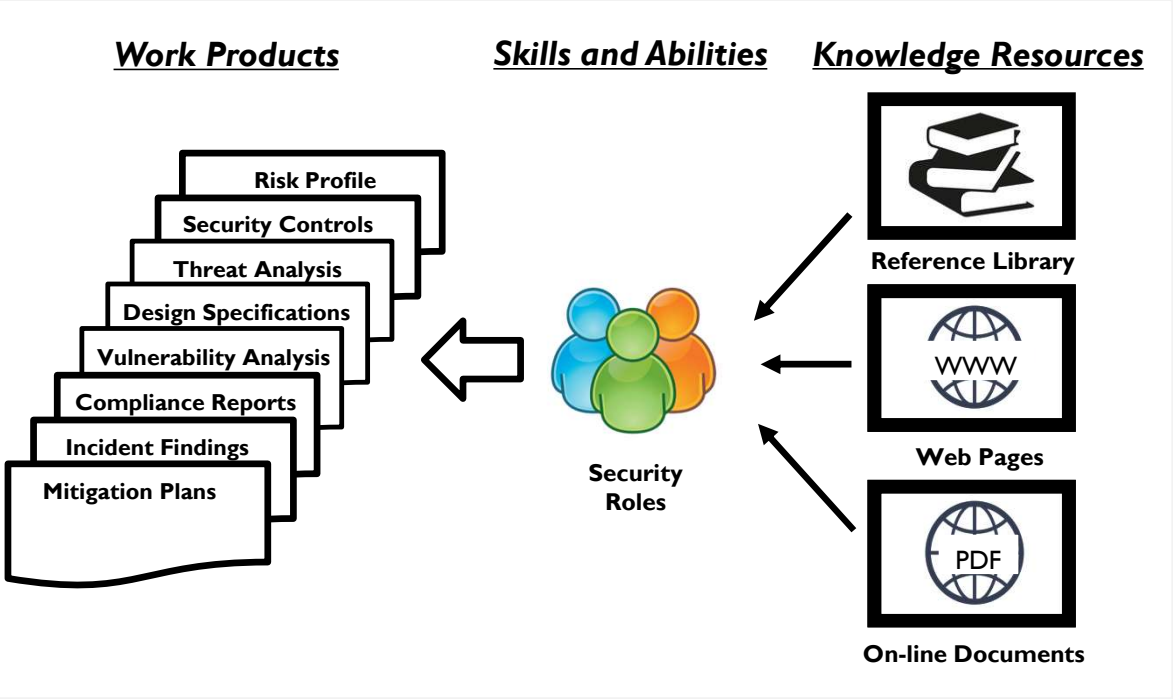
https://csrc.nist.gov/projects/cprt/catalog#/cprt/framework/version/SP_800_53_5_1_0/home



http://attack.mitre.org/

PROBLEM STATEMENT

THE VOLUME AND FORMAT OF SECURITY INFORMATION IS AN OBSTACLE FOR LEARNING AND APPLIED SECURITY ANALYSIS



Security Information Resources	Current Entry Count
ISO/IEC 15408:2022, Functional Reqmts	134
ISO/IEC 27002:2013 Security Controls	214
MITRE Attack Tactics	40
MITRE Attack Techniques	521
MITRE Common Attack Patterns (CAPEC)	555
MITRE Common Weaknesses (CWE)	933
MITRE Common Vulnerabilities (CVE)	188,461
NIST National Vulnerability Database (NVD)	199,581
NIST SP800-53R5.1 Security Controls	298

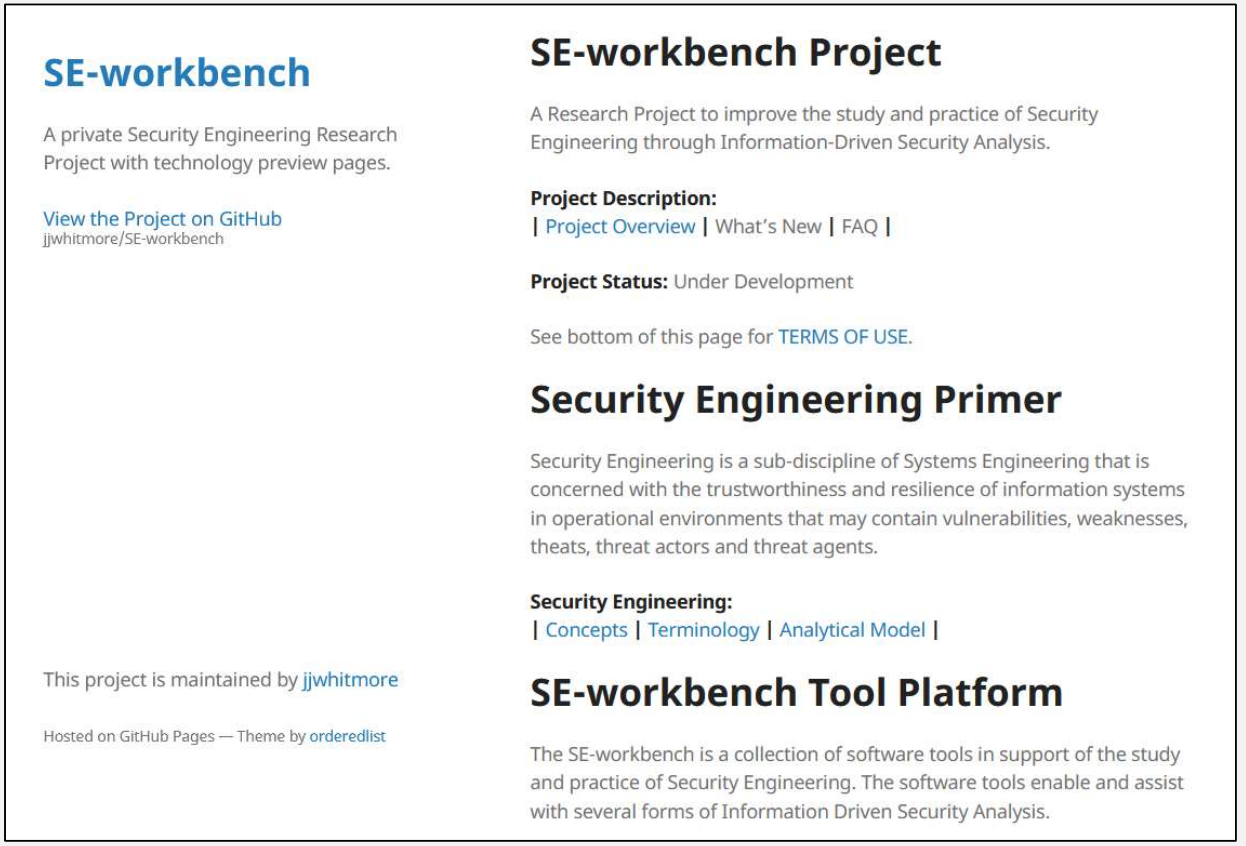
SECURITY ENGINEERING WORKBENCH

SE-WORKBENCH (2022)
EXPERIMENTAL PLATFORM

Security Control Explorer (SCE). This tool organizes and displays the security and privacy controls from NIST SP800-53, their connections with ISO27001 along with the security capabilities from the NIST Cybersecurity Framework and links to authoritative security reference documents.

Security Vulnerability Explorer (SVE). This tool organizes and displays security weaknesses and vulnerabilities from MITRE Common Weakness Enumeration (CWE), relationship to published lists such as OWASP, connections to NIST Vulnerability Database (NVD), along with analytical insights and correlation from other information sources.

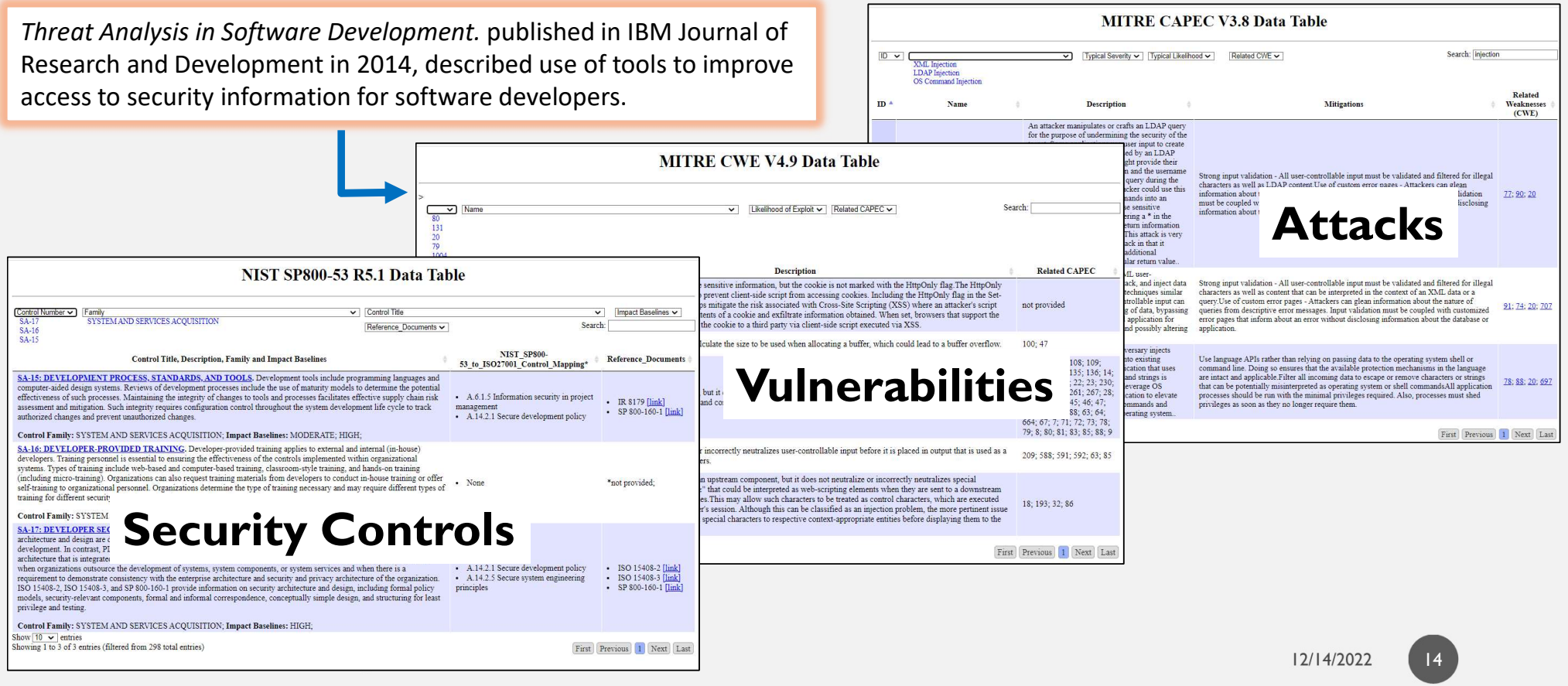
Security Attack Explorer (SAE). This tool organizes and displays security attack patterns from MITRE Common Attack Pattern Enumeration and Classification (CAPEC), along with: connections to Mitre Common Weaknesses (CWE), exploitation techniques from Mitre Attack (ATT&CK), recommended test and assurance strategies.



Project link:
https://www.researchgate.net/project/Security-Engineering-Workbench

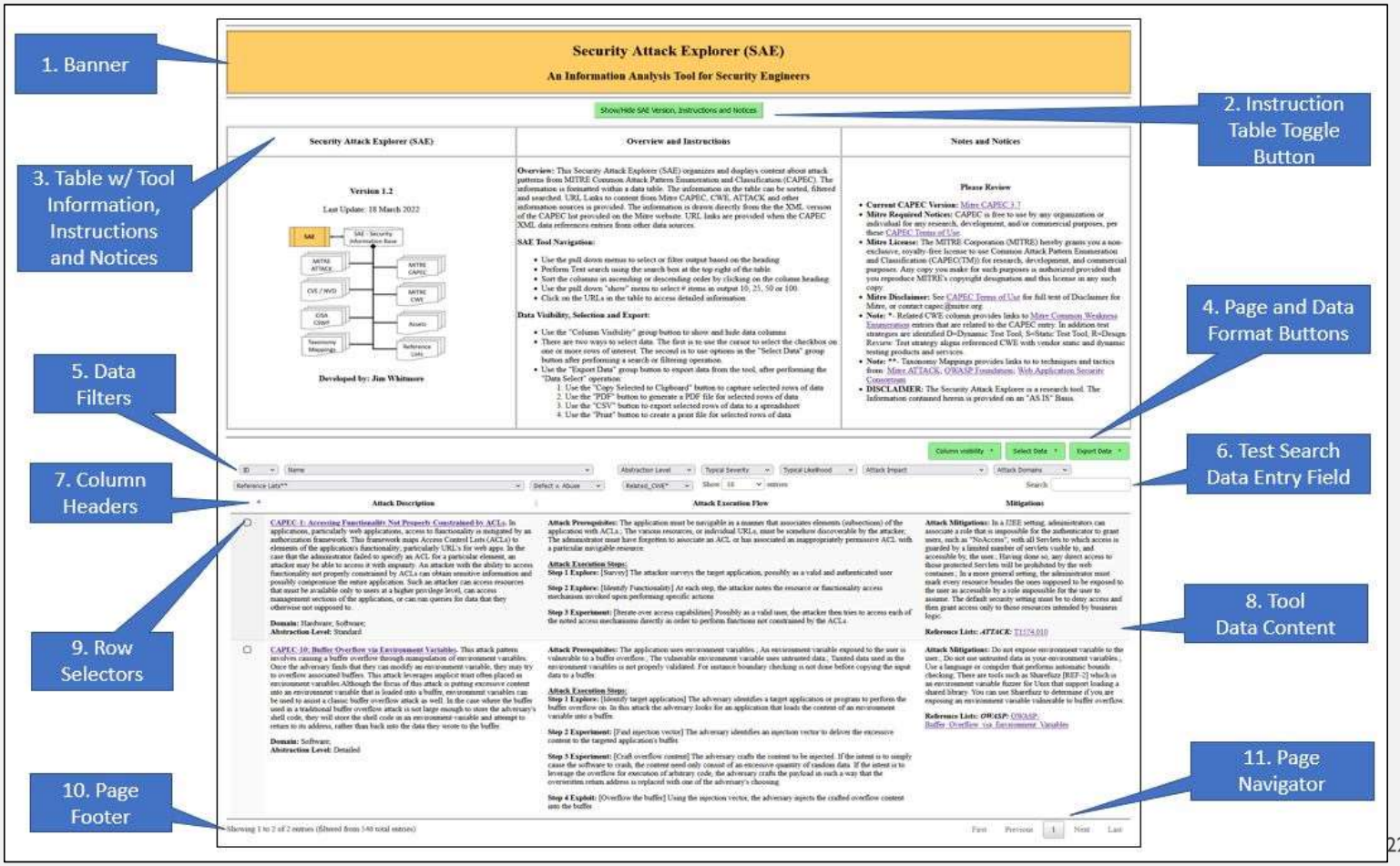
SE-WORKBENCH STARTED AS A SET OF LEARNING AIDS FOR CYBER & INFORMATION SECURITY COURSES

Threat Analysis in Software Development. published in IBM Journal of Research and Development in 2014, described use of tools to improve access to security information for software developers.

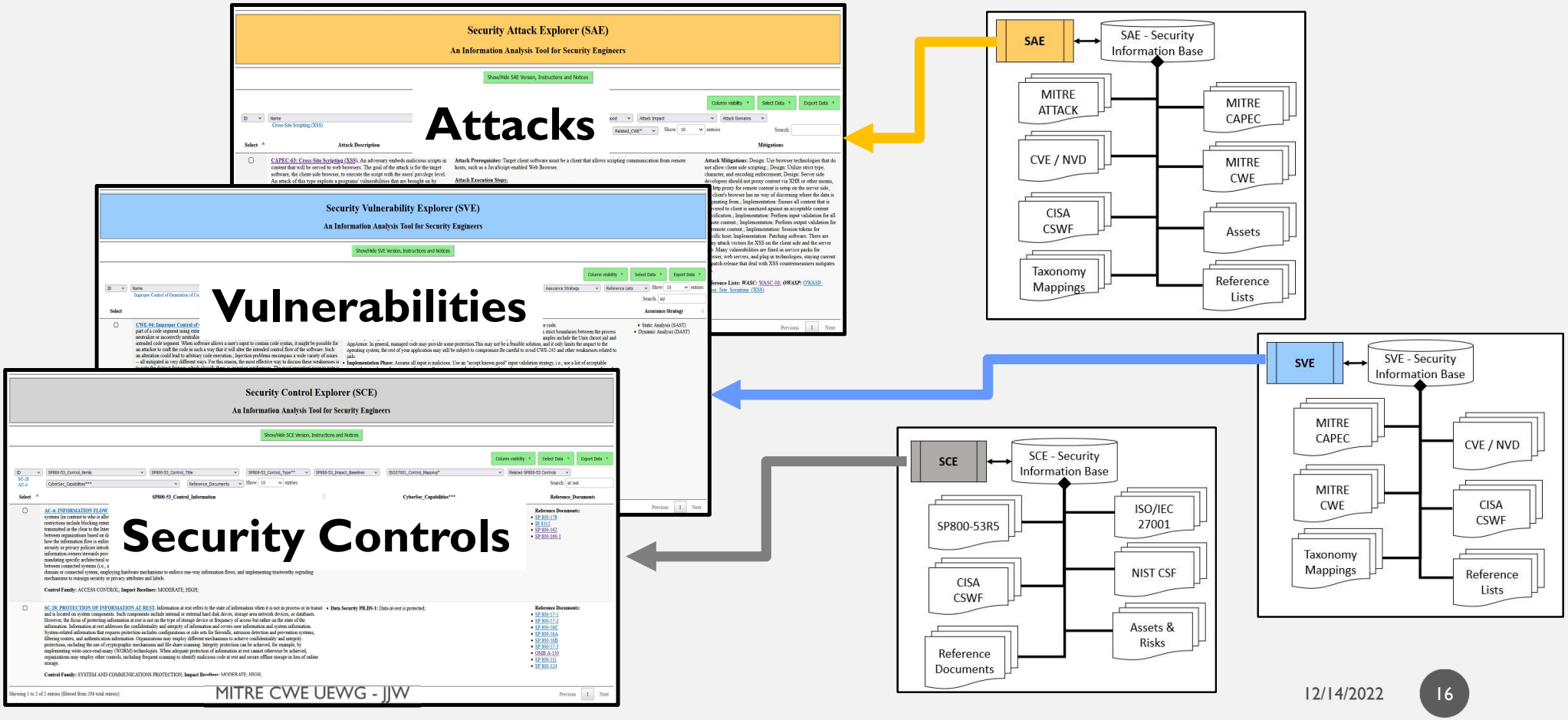


SE-WORKBENCH TOOLS HAVE A **COMMON USER INTERFACE**

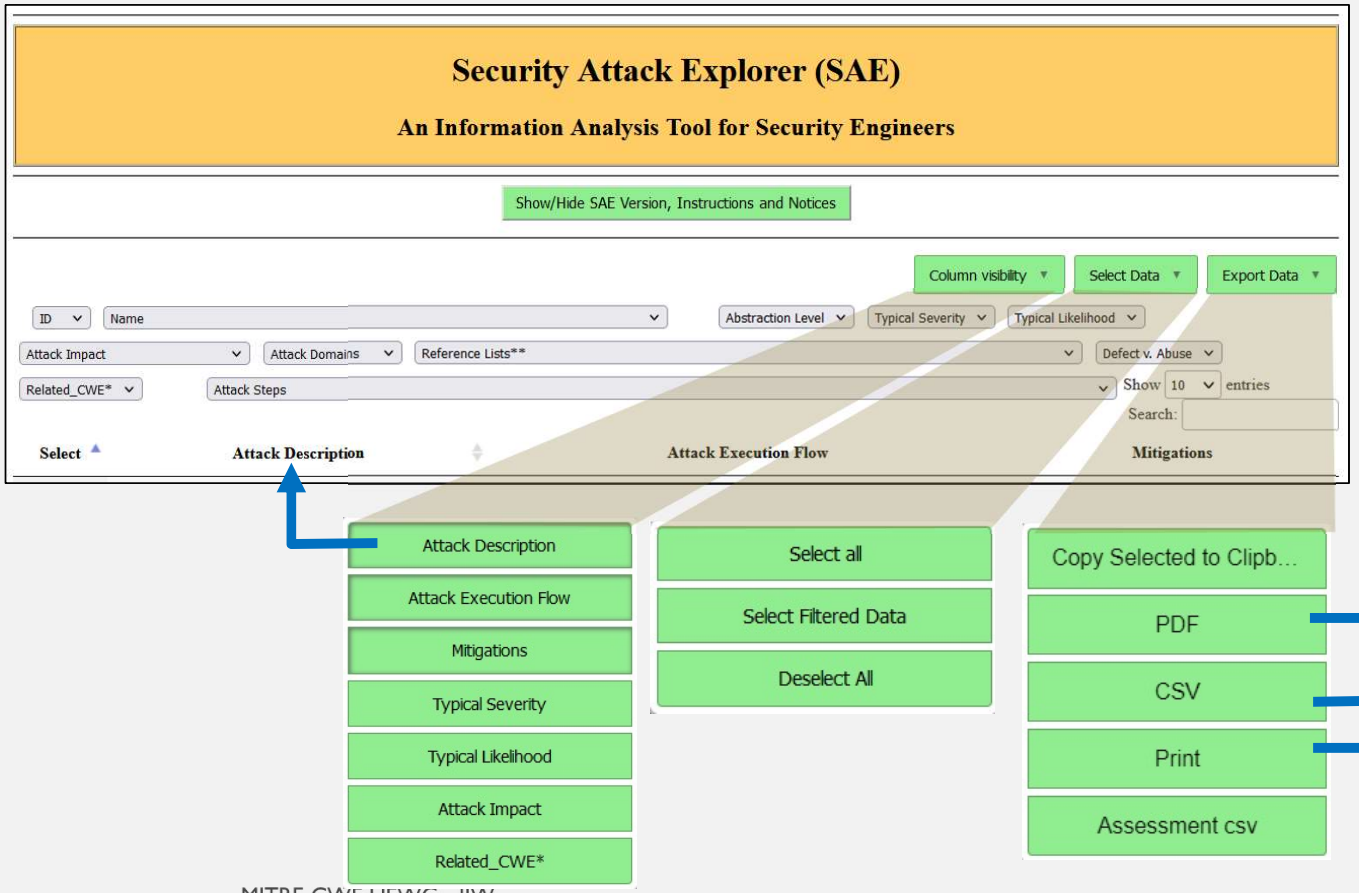
- Works w/ modern browsers
- Works on cell phones
- Requires no server-side code

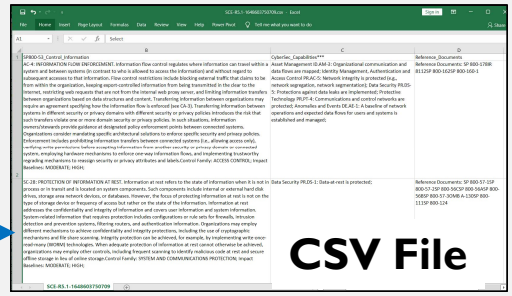


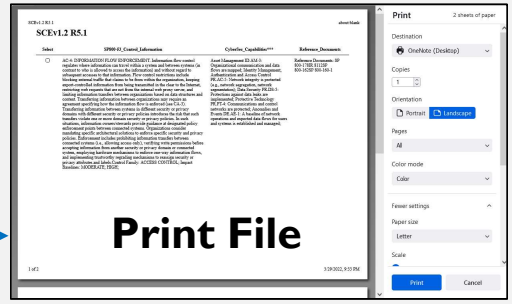
SE-WORKBENCH TOOLS HAVE EXPANDED **DATA MODELS**



USERS EXAMINE ENTRIES OF INTEREST AND CREATE PROJECT ARTIFACTS VIA **DATA FORMAT BUTTONS & MENUS**







SE WORKBENCH
TOOL DESIGN AND FUNCTIONALITY

Design

1. Built on Open Source components (DataTables, Column Filter Widgets, Javascript, CSS / XSLT)
2. Tools operate on Machine readable XML versions of MITRE CWE & CAPEC and NIST SP800-53R5.1
3. Live links to external data sources (ATTACK, OWASP, etc.)
4. Compatible with Modern browsers (including mobile phones)
5. Self-documenting with support web pages
6. No server-side application code

Functionality

1. Text Search
2. Entries Filter-able by groups and values
3. Entries sortable by column
4. Variable page length
5. Selectable data column visibility
6. Selectable data export
7. Multiple export formats: PDF, CSV, Print, Assessment
8. Extended data correlations

Project link: https://www.researchgate.net/project/Security-Engineering-Workbench

SUMMARY…

1. SE Workbench is a collection of tools and materials created as learning aids Cyber and Information Security education.
2. SE Workbench is an evolution of earlier work…
   - *Improving Attention to Security in Software Design with Analytics and Cognitive Techniques.* IEEE SecDev Conference Paper. Aug 2017.
   - *Threat analysis in the software development lifecycle.* IBM Journal of Research and Development. Dec 2013.
3. Additional Potential Uses
   - A platform for detailed security analysis.
   - A platform for security data improvement.

SUGGESTIONS FOR CURATING CWE & CAPEC XML DATA …

1. Fix data elements with missing values
   e.g., Typical Severity; Typical Likelihood; Attack Impact; Attack Steps; , etc.
2. Reformat / update elements to improve sorting and risk calculations
   e.g., add numeric values to Typical Severity; Typical Likelihood; , etc.
3. Create affinity lists for easier processing
   e.g., Domains of Attack list; Mechanisms of Attack list; , etc.
4. Normalize data elements to improve correlation
   e.g., consistent phrasing and capitalization on Attack Steps
5. Add tags to entries to categorize by attack target
   e.g., <attack target> = browser; web server; db server; network device; operating system; person; ,etc.
6. Improve correlation between CWE's and CVE's
   e.g., should be able to search CVEs for instances of CWE
7. Standardize XML coding
   e.g., <xhtml:p>, etc.

WORKED EXAMPLE

ANALYZE "EXCAVATION" ATTACKS CONSIDERING SECURITY, LIKELIHOOD AND ASSURANCE MEASURES

CAPEC-116: Excavation

Attack Pattern ID: 116
Abstraction: Meta

Presentation Filter: Complete

Description

An adversary actively probes the target in a manner that is designed to solicit information that could be leveraged for malicious purposes.

Extended Description

Likelihood Of Attack

High

Typical Severity

Medium

Relationships

Nature	Type	ID	Name
ParentOf	S	54	Query System for Information
ParentOf	S	150	Collect Data from Common Resource Locations
ParentOf	S	545	Pull Data from System Resources
ParentOf	S	569	Collect Data as Provided by Users
ParentOf	S	675	Retrieve Data from Decommissioned Devices
CanPrecede	D	163	Spear Phishing

CAPEC-54: Query System for Information

Attack Pattern ID: 54
Abstraction: Standard

Presentation Filter: Complete

Description

An adversary, aware of an application's location (and possibly authorized to use the application), probes an application's structure and evaluates its robustness by submitting requests and examining responses. Often, this is accomplished by sending variants of expected queries in the hope that these modified queries might return information beyond what the expected set of queries would provide.

CAPEC-150: Collect Data from Common Resource Locations

Attack Pattern ID: 150
Abstraction: Standard

Presentation Filter: Complete

Description

An adversary exploits well-known locations for resources for the purposes of undermining the security of the target. In many, if not most systems, files and resources are organized in a default tree structure. This can be useful for adversaries because they often know where to look for resources or files that are necessary for attacks. Even when the precise location of a targeted resource may not be known, naming conventions may indicate a small area of the target machine's file tree where the resources are typically located. For example, configuration files are normally stored in the /etc director on Unix systems. Adversaries can take advantage of this to commit other types of attacks.

CAPEC-545: Pull Data from System Resources

Attack Pattern ID: 545
Abstraction: Standard

Presentation Filter: Complete

Description

An adversary who is authorized or has the ability to search known system resources, does so with the intention of gathering useful information. System resources include files, memory, and other aspects of the target system. In this pattern of attack, the adversary does not necessarily know what they are going to find when they start pulling data. This is different than CAPEC-150 where the adversary knows what they are looking for due to the common location.

CAPEC-569: Collect Data as Provided by Users

Attack Pattern ID: 569
Abstraction: Standard

Presentation Filter: Complete

Description

An attacker leverages a tool, device, or program to obtain specific information as provided by a user of the target system. This information is often needed by the attacker to launch a follow-on attack. This attack is different than Social Engineering as the adversary is not tricking or deceiving the user. Instead the adversary is putting a mechanism in place that captures the information that a user legitimately enters into a system. Deploying a keylogger, performing a UAC prompt, or wrapping the Windows default credential provider are all examples of such interactions.

CAPEC-675: Retrieve Data from Decommissioned Devices

Attack Pattern ID: 675
Abstraction: Standard

Presentation Filter: Complete

Description

An adversary obtains decommissioned, recycled, or discarded systems and devices that can include an organization's intellectual property, employee data, and other types of controlled information. Systems and devices that have reached the end of their lifecycles may be subject to recycle or disposal where they can be exposed to adversarial attempts to retrieve information from internal memory chips and storage devices that are part of the system.

https://capec.mitre.org/data/definitions/116.html

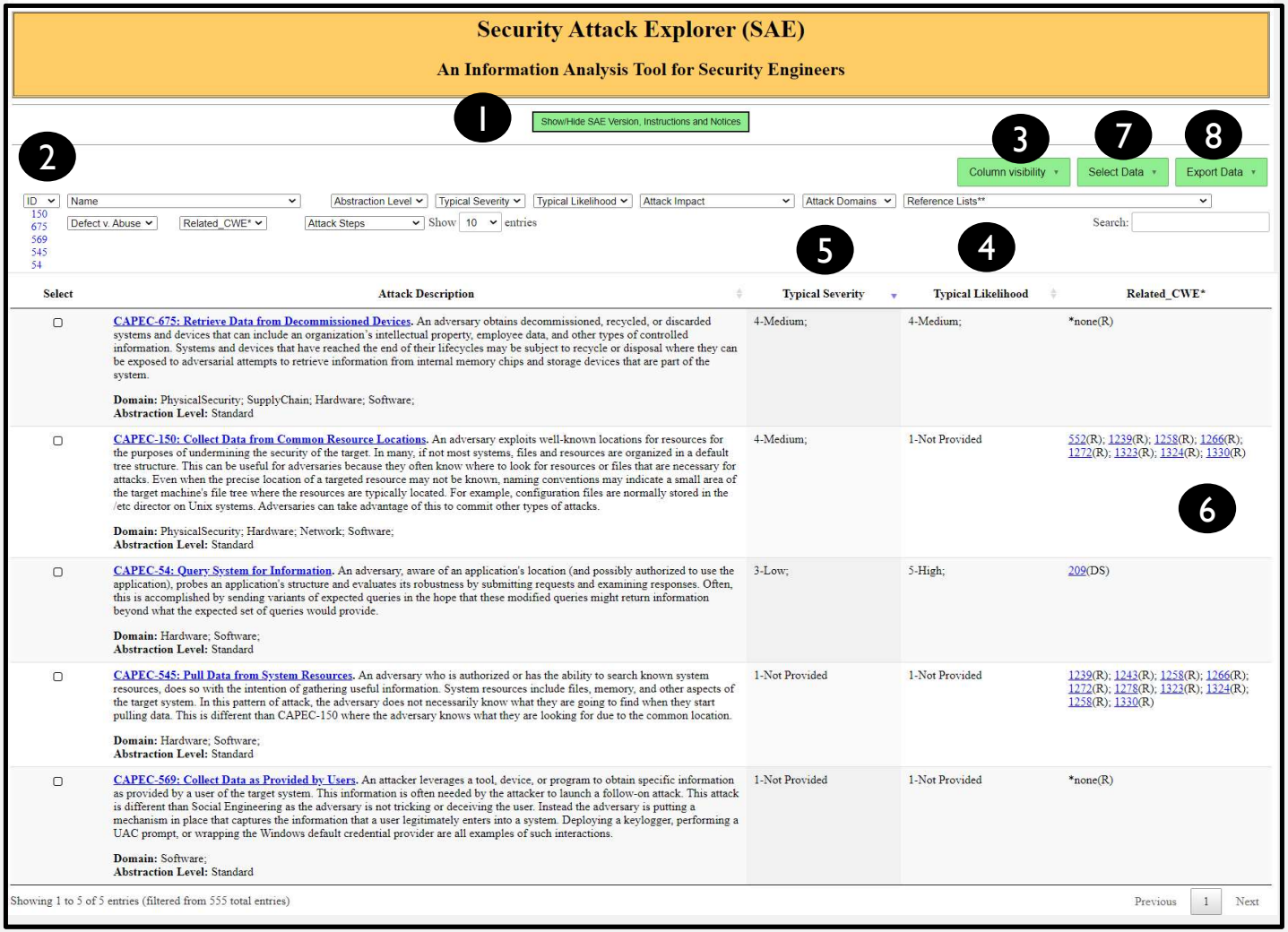
- Many Browser Tabs
- Manual data correlation
- Manual creation of project artifacts

SE WORKBENCH BROWSER DISPLAY

RANK ATTACKS BY SEVERITY & LIKELIHOOD

Basic Operation

1. View and close instructions
2. Enter attack IDs
3. Select column visibility
4. Sort Likelihood Hi to Low
5. Sort Severity Hi to Low
6. Observe and analyze entries
7. Select relevant entries
8. Create artifacts

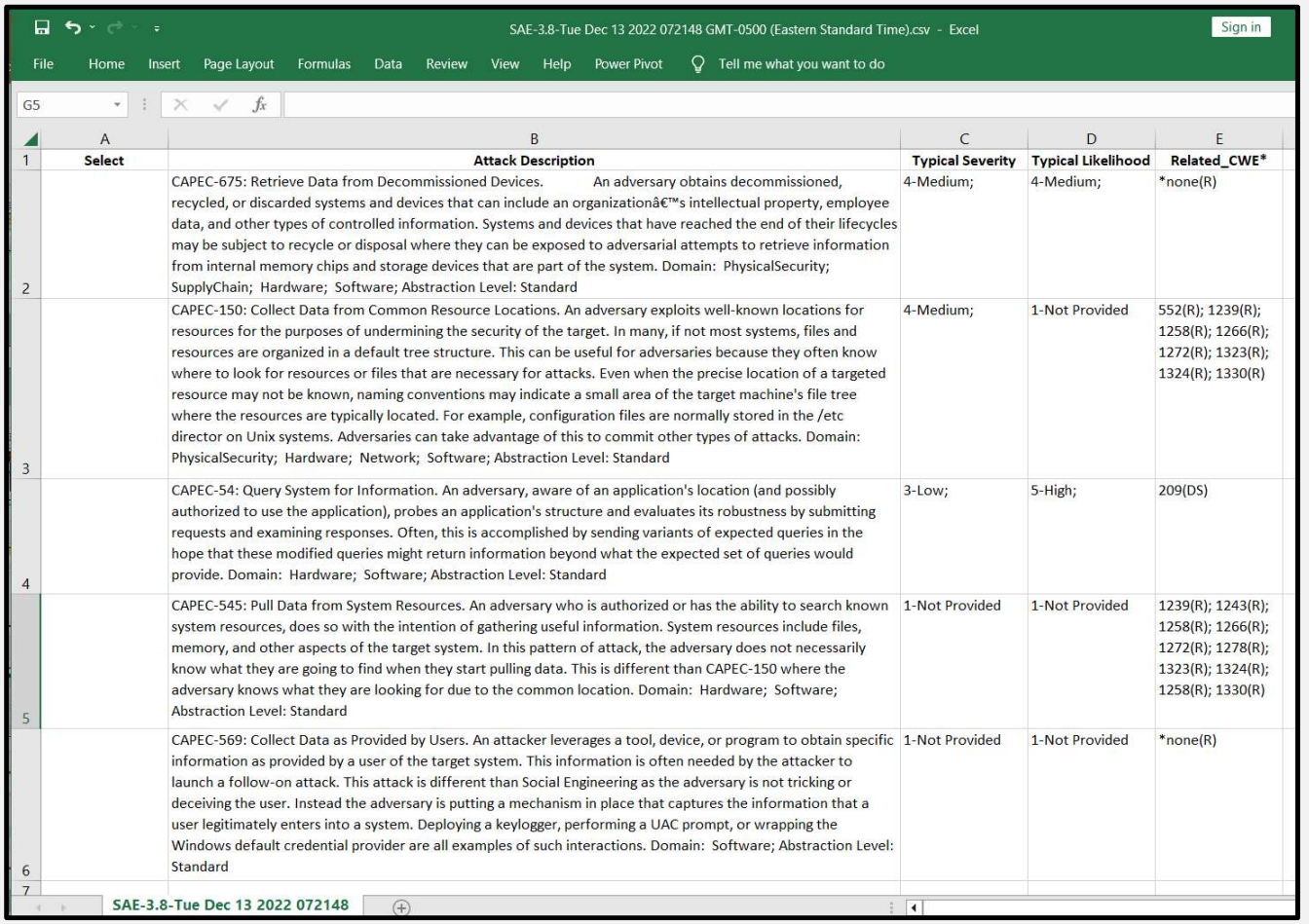


SE WORKBENCH PDF EXPORT

SAEv1.3 V3.8

Select	Attack Description	Typical Severity	Typical Likelihood	Related_CWE^A
	CAPEC-675: Retrieve Data from Decommissioned Devices. An adversary obtains decommissioned, recycled, or discarded systems and devices that can include an organization's intellectual property, employee data, and other types of controlled information. Systems and devices that have reached the end of their lifecycles may be subject to recycle or disposal where they can be exposed to adversarial attempts to retrieve information from internal memory chips and storage devices that are part of the system. Domain: PhysicalSecurity; SupplyChain; Hardware; Software; Abstraction Level: Standard	4-Medium;	4-Medium;	*none(R)
	CAPEC-150: Collect Data from Common Resource Locations. An adversary exploits well-known locations for resources for the purposes of undermining the security of the target. In many, if not most systems, files and resources are organized in a default tree structure. This can be useful for adversaries because they often know where to look for resources or files that are necessary for attacks. Even when the precise location of a targeted resource may not be known, naming conventions may indicate a small area of the target machine's file tree where the resources are typically located. For example, configuration files are normally stored in the /etc director on Unix systems. Adversaries can take advantage of this to commit other types of attacks. Domain: PhysicalSecurity; Hardware; Network; Software; Abstraction Level: Standard	4-Medium;	1-Not Provided	552(R); 1239(R); 1258(R); 1266(R); 1272(R); 1323(R); 1324(R); 1330(R)
	CAPEC-54: Query System for Information. An adversary, aware of an application's location (and possibly authorized to use the application), probes an application's structure and evaluates its robustness by submitting requests and examining responses. Often, this is accomplished by sending variants of expected queries in the hope that these modified queries might return information beyond what the expected set of queries would provide. Domain: Hardware; Software; Abstraction Level: Standard	3-Low;	5-High;	209(DS)
	CAPEC-545: Pull Data from System Resources. An adversary who is authorized or has the ability to search known system resources, does so with the intention of gathering useful information. System resources include files, memory, and other aspects of the target system. In this pattern of attack, the adversary does not necessarily know what they are going to find when they start pulling data. This is different than CAPEC-150 where the adversary knows what they are looking for due to the common location. Domain: Hardware; Software; Abstraction Level: Standard	1-Not Provided	1-Not Provided	1239(R); 1243(R); 1258(R); 1266(R); 1272(R); 1278(R); 1323(R); 1324(R); 1258(R); 1330(R)
	CAPEC-569: Collect Data as Provided by Users. An attacker leverages a tool, device, or program to obtain specific information as provided by a user of the target system. This information is often needed by the attacker to launch a follow-on attack. This attack is different than Social Engineering as the adversary is not tricking or deceiving the user. Instead the adversary is putting a mechanism in place that captures the information that a user legitimately enters into a system. Deploying a keylogger, performing a UAC prompt, or wrapping the Windows default credential provider are all examples of such interactions. Domain: Software; Abstraction Level: Standard	1-Not Provided	1-Not Provided	*none(R)

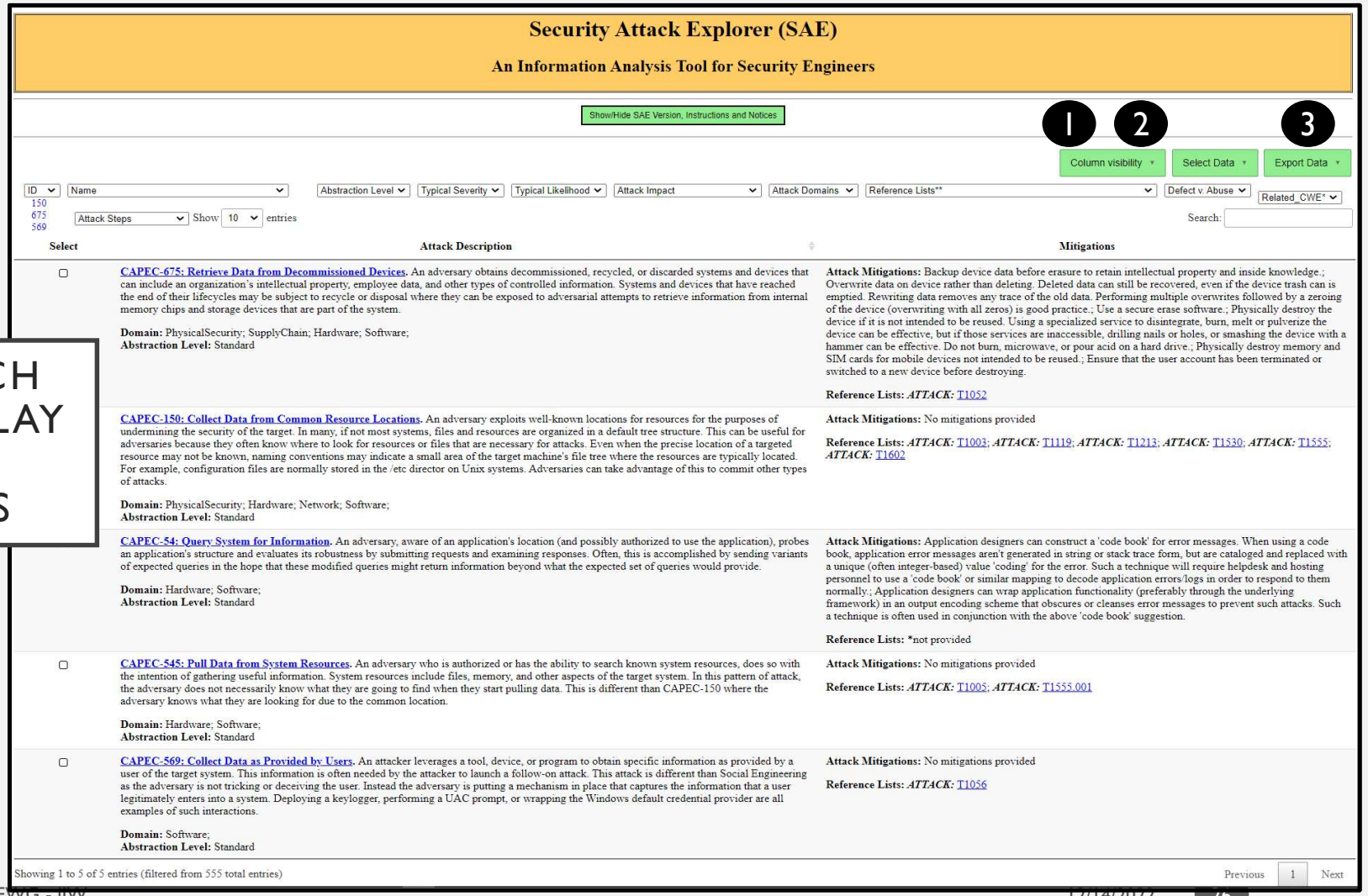
SE WORKBENCH CSV EXPORT



	A	B	C	D	E
1	**Select**	**Attack Description**	**Typical Severity**	**Typical Likelihood**	**Related_CWE***
2		CAPEC-675: Retrieve Data from Decommissioned Devices. An adversary obtains decommissioned, recycled, or discarded systems and devices that can include an organizationâ€™s intellectual property, employee data, and other types of controlled information. Systems and devices that have reached the end of their lifecycles may be subject to recycle or disposal where they can be exposed to adversarial attempts to retrieve information from internal memory chips and storage devices that are part of the system. Domain: PhysicalSecurity; SupplyChain; Hardware; Software; Abstraction Level: Standard	4-Medium;	4-Medium;	*none(R)
3		CAPEC-150: Collect Data from Common Resource Locations. An adversary exploits well-known locations for resources for the purposes of undermining the security of the target. In many, if not most systems, files and resources are organized in a default tree structure. This can be useful for adversaries because they often know where to look for resources or files that are necessary for attacks. Even when the precise location of a targeted resource may not be known, naming conventions may indicate a small area of the target machine's file tree where the resources are typically located. For example, configuration files are normally stored in the /etc director on Unix systems. Adversaries can take advantage of this to commit other types of attacks. Domain: PhysicalSecurity; Hardware; Network; Software; Abstraction Level: Standard	4-Medium;	1-Not Provided	552(R); 1239(R); 1258(R); 1266(R); 1272(R); 1323(R); 1324(R); 1330(R)
4		CAPEC-54: Query System for Information. An adversary, aware of an application's location (and possibly authorized to use the application), probes an application's structure and evaluates its robustness by submitting requests and examining responses. Often, this is accomplished by sending variants of expected queries in the hope that these modified queries might return information beyond what the expected set of queries would provide. Domain: Hardware; Software; Abstraction Level: Standard	3-Low;	5-High;	209(DS)
5		CAPEC-545: Pull Data from System Resources. An adversary who is authorized or has the ability to search known system resources, does so with the intention of gathering useful information. System resources include files, memory, and other aspects of the target system. In this pattern of attack, the adversary does not necessarily know what they are going to find when they start pulling data. This is different than CAPEC-150 where the adversary knows what they are looking for due to the common location. Domain: Hardware; Software; Abstraction Level: Standard	1-Not Provided	1-Not Provided	1239(R); 1243(R); 1258(R); 1266(R); 1272(R); 1278(R); 1323(R); 1324(R); 1258(R); 1330(R)
6		CAPEC-569: Collect Data as Provided by Users. An attacker leverages a tool, device, or program to obtain specific information as provided by a user of the target system. This information is often needed by the attacker to launch a follow-on attack. This attack is different than Social Engineering as the adversary is not tricking or deceiving the user. Instead the adversary is putting a mechanism in place that captures the information that a user legitimately enters into a system. Deploying a keylogger, performing a UAC prompt, or wrapping the Windows default credential provider are all examples of such interactions. Domain: Software; Abstraction Level: Standard	1-Not Provided	1-Not Provided	*none(R)
7					

SE WORKBENCH BROWSER DISPLAY

MITIGATIONS



Security Attack Explorer (SAE)

An Information Analysis Tool for Security Engineers

Show/Hide SAE Version, Instructions and Notices

1 2 3

Column visibility | Select Data | Export Data

ID | Name | Abstraction Level | Typical Severity | Typical Likelihood | Attack Impact | Attack Domains | Reference Lists** | Defect v. Abuse | Related_CWE*

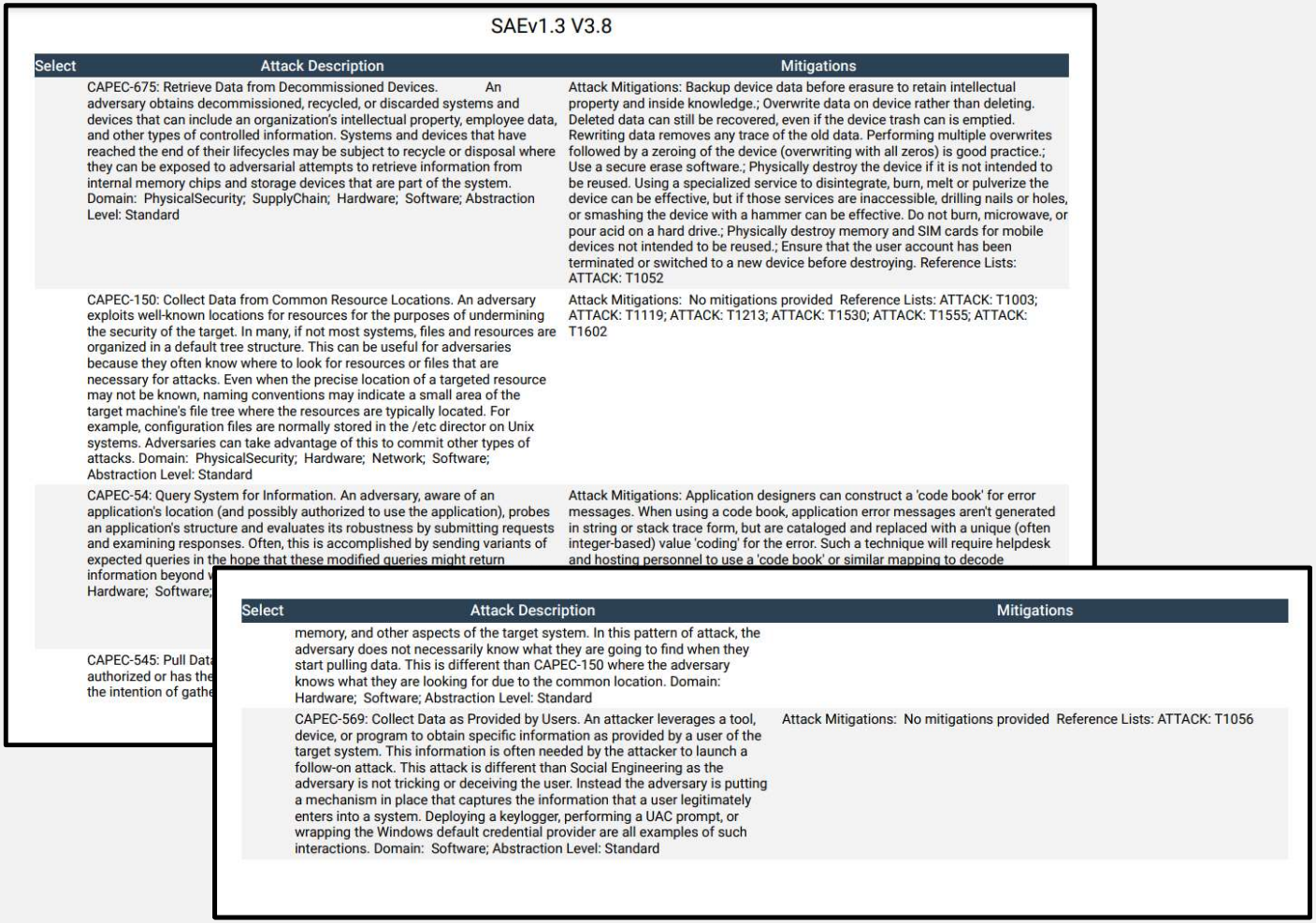
150
675
569

Attack Steps | Show 10 entries | Search:

Select	Attack Description	Mitigations
☐	**CAPEC-675: Retrieve Data from Decommissioned Devices.** An adversary obtains decommissioned, recycled, or discarded systems and devices that can include an organization's intellectual property, employee data, and other types of controlled information. Systems and devices that have reached the end of their lifecycles may be subject to recycle or disposal where they can be exposed to adversarial attempts to retrieve information from internal memory chips and storage devices that are part of the system. **Domain:** PhysicalSecurity; SupplyChain; Hardware; Software; **Abstraction Level:** Standard	**Attack Mitigations:** Backup device data before erasure to retain intellectual property and inside knowledge.; Overwrite data on device rather than deleting. Deleted data can still be recovered, even if the device trash can is emptied. Rewriting data removes any trace of the old data. Performing multiple overwrites followed by a zeroing of the device (overwriting with all zeros) is good practice.; Use a secure erase software.; Physically destroy the device if it is not intended to be reused. Using a specialized service to disintegrate, burn, melt or pulverize the device can be effective, but if those services are inaccessible, drilling nails or holes, or smashing the device with a hammer can be effective. Do not burn, microwave, or pour acid on a hard drive.; Physically destroy memory and SIM cards for mobile devices not intended to be reused.; Ensure that the user account has been terminated or switched to a new device before destroying. **Reference Lists:** ***ATTACK:*** T1052
	CAPEC-150: Collect Data from Common Resource Locations. An adversary exploits well-known locations for resources for the purposes of undermining the security of the target. In many, if not most systems, files and resources are organized in a default tree structure. This can be useful for adversaries because they often know where to look for resources or files that are necessary for attacks. Even when the precise location of a targeted resource may not be known, naming conventions may indicate a small area of the target machine's file tree where the resources are typically located. For example, configuration files are normally stored in the /etc director on Unix systems. Adversaries can take advantage of this to commit other types of attacks. **Domain:** PhysicalSecurity; Hardware; Network; Software; **Abstraction Level:** Standard	**Attack Mitigations:** No mitigations provided **Reference Lists:** ***ATTACK:*** T1003; ***ATTACK:*** T1119; ***ATTACK:*** T1213; ***ATTACK:*** T1530; ***ATTACK:*** T1555; ***ATTACK:*** T1602
	CAPEC-54: Query System for Information. An adversary, aware of an application's location (and possibly authorized to use the application), probes an application's structure and evaluates its robustness by submitting requests and examining responses. Often, this is accomplished by sending variants of expected queries in the hope that these modified queries might return information beyond what the expected set of queries would provide. **Domain:** Hardware; Software; **Abstraction Level:** Standard	**Attack Mitigations:** Application designers can construct a 'code book' for error messages. When using a code book, application error messages aren't generated in string or stack trace form, but are cataloged and replaced with a unique (often integer-based) value 'coding' for the error. Such a technique will require helpdesk and hosting personnel to use a 'code book' or similar mapping to decode application errors/logs in order to respond to them normally.; Application designers can wrap application functionality (preferably through the underlying framework) in an output encoding scheme that obscures or cleanses error messages to prevent such attacks. Such a technique is often used in conjunction with the above 'code book' suggestion. **Reference Lists:** *not provided
☐	**CAPEC-545: Pull Data from System Resources.** An adversary who is authorized or has the ability to search known system resources, does so with the intention of gathering useful information. System resources include files, memory, and other aspects of the target system. In this pattern of attack, the adversary does not necessarily know what they are going to find when they start pulling data. This is different than CAPEC-150 where the adversary knows what they are looking for due to the common location. **Domain:** Hardware; Software; **Abstraction Level:** Standard	**Attack Mitigations:** No mitigations provided **Reference Lists:** ***ATTACK:*** T1005; ***ATTACK:*** T1555.001
☐	**CAPEC-569: Collect Data as Provided by Users.** An attacker leverages a tool, device, or program to obtain specific information as provided by a user of the target system. This information is often needed by the attacker to launch a follow-on attack. This attack is different than Social Engineering as the adversary is not tricking or deceiving the user. Instead the adversary is putting a mechanism in place that captures the information that a user legitimately enters into a system. Deploying a keylogger, performing a UAC prompt, or wrapping the Windows default credential provider are all examples of such interactions. **Domain:** Software; **Abstraction Level:** Standard	**Attack Mitigations:** No mitigations provided **Reference Lists:** ***ATTACK:*** T1056

Showing 1 to 5 of 5 entries (filtered from 555 total entries) | Previous | 1 | Next

SE WORKBENCH PDF EXPORT



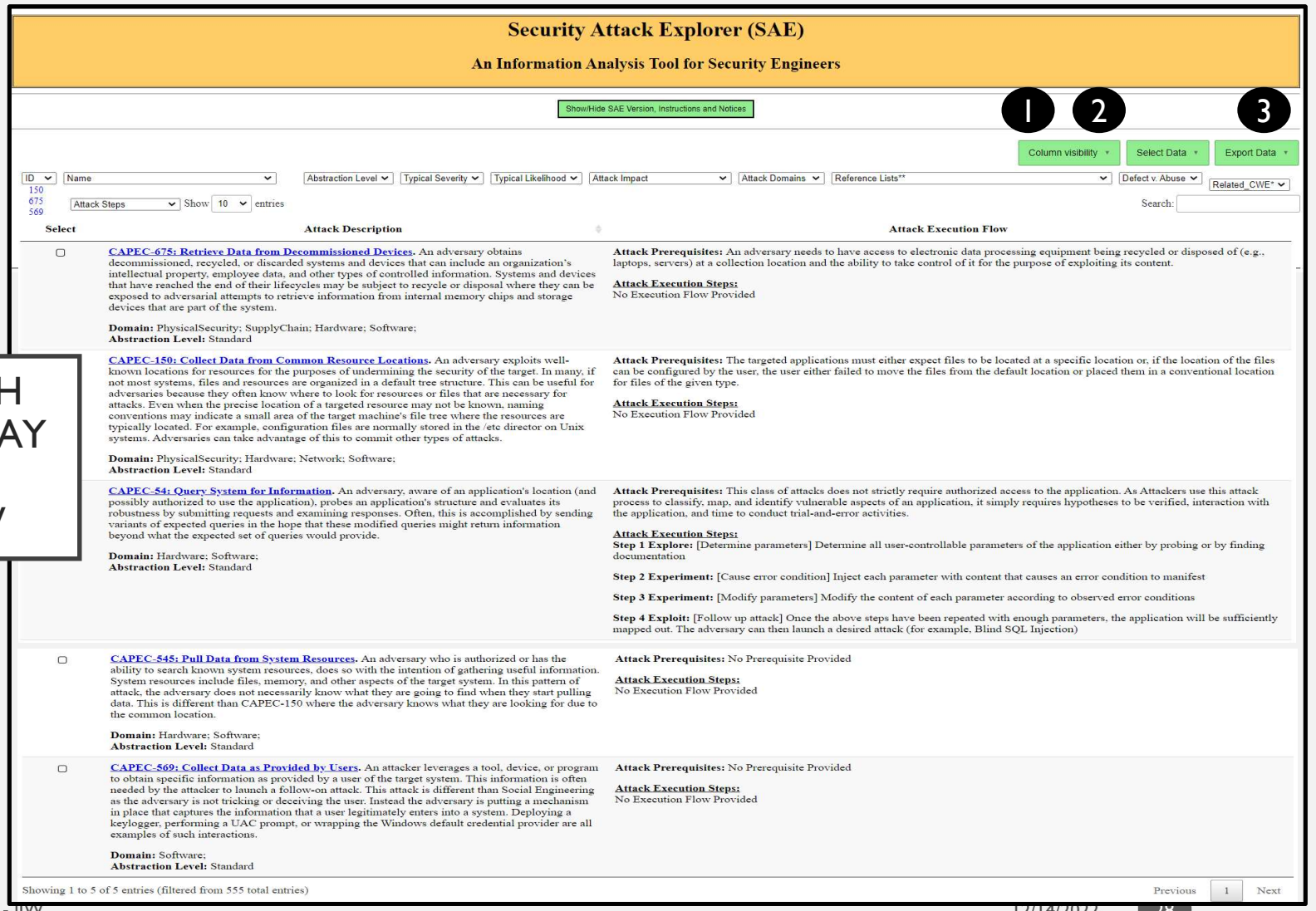
SAEv1.3 V3.8

Select	Attack Description	Mitigations
	CAPEC-675: Retrieve Data from Decommissioned Devices. An adversary obtains decommissioned, recycled, or discarded systems and devices that can include an organization's intellectual property, employee data, and other types of controlled information. Systems and devices that have reached the end of their lifecycles may be subject to recycle or disposal where they can be exposed to adversarial attempts to retrieve information from internal memory chips and storage devices that are part of the system. Domain: PhysicalSecurity; SupplyChain; Hardware; Software; Abstraction Level: Standard	Attack Mitigations: Backup device data before erasure to retain intellectual property and inside knowledge.; Overwrite data on device rather than deleting. Deleted data can still be recovered, even if the device trash can is emptied. Rewriting data removes any trace of the old data. Performing multiple overwrites followed by a zeroing of the device (overwriting with all zeros) is good practice.; Use a secure erase software.; Physically destroy the device if it is not intended to be reused. Using a specialized service to disintegrate, burn, melt or pulverize the device can be effective, but if those services are inaccessible, drilling nails or holes, or smashing the device with a hammer can be effective. Do not burn, microwave, or pour acid on a hard drive.; Physically destroy memory and SIM cards for mobile devices not intended to be reused.; Ensure that the user account has been terminated or switched to a new device before destroying. Reference Lists: ATTACK: T1052
	CAPEC-150: Collect Data from Common Resource Locations. An adversary exploits well-known locations for resources for the purposes of undermining the security of the target. In many, if not most systems, files and resources are organized in a default tree structure. This can be useful for adversaries because they often know where to look for resources or files that are necessary for attacks. Even when the precise location of a targeted resource may not be known, naming conventions may indicate a small area of the target machine's file tree where the resources are typically located. For example, configuration files are normally stored in the /etc director on Unix systems. Adversaries can take advantage of this to commit other types of attacks. Domain: PhysicalSecurity; Hardware; Network; Software; Abstraction Level: Standard	Attack Mitigations: No mitigations provided Reference Lists: ATTACK: T1003; ATTACK: T1119; ATTACK: T1213; ATTACK: T1530; ATTACK: T1555; ATTACK: T1602
	CAPEC-54: Query System for Information. An adversary, aware of an application's location (and possibly authorized to use the application), probes an application's structure and evaluates its robustness by submitting requests and examining responses. Often, this is accomplished by sending variants of expected queries in the hope that these modified queries might return information beyond v... Hardware; Software;	Attack Mitigations: Application designers can construct a 'code book' for error messages. When using a code book, application error messages aren't generated in string or stack trace form, but are cataloged and replaced with a unique (often integer-based) value 'coding' for the error. Such a technique will require helpdesk and hosting personnel to use a 'code book' or similar mapping to decode
	CAPEC-545: Pull Data... authorized or has the... the intention of gathe...	

Select	Attack Description	Mitigations
	memory, and other aspects of the target system. In this pattern of attack, the adversary does not necessarily know what they are going to find when they start pulling data. This is different than CAPEC-150 where the adversary knows what they are looking for due to the common location. Domain: Hardware; Software; Abstraction Level: Standard	
	CAPEC-569: Collect Data as Provided by Users. An attacker leverages a tool, device, or program to obtain specific information as provided by a user of the target system. This information is often needed by the attacker to launch a follow-on attack. This attack is different than Social Engineering as the adversary is not tricking or deceiving the user. Instead the adversary is putting a mechanism in place that captures the information that a user legitimately enters into a system. Deploying a keylogger, performing a UAC prompt, or wrapping the Windows default credential provider are all examples of such interactions. Domain: Software; Abstraction Level: Standard	Attack Mitigations: No mitigations provided Reference Lists: ATTACK: T1056

SE WORKBENCH BROWSER DISPLAY

ATTACK FLOW

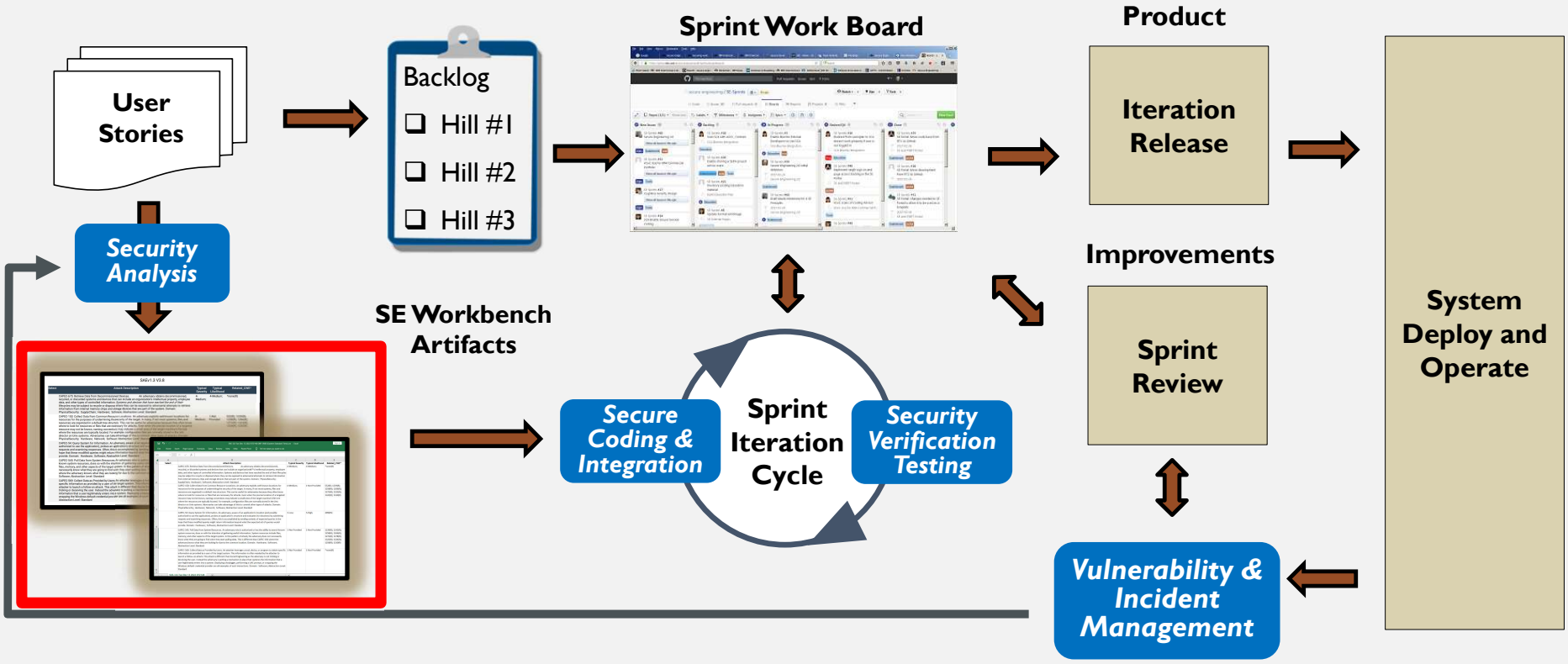


SE WORKBENCH PDF EXPORT

SAEv1.3 V3.8

Select	Attack Description	Attack Execution Flow
	CAPEC-675: Retrieve Data from Decommissioned Devices. An adversary obtains decommissioned, recycled, or discarded systems and devices that can include an organization's intellectual property, employee data, and other types of controlled information. Systems and devices that have reached the end of their lifecycles may be subject to recycle or disposal where they can be exposed to adversarial attempts to retrieve information from internal memory chips and storage devices that are part of the system. Domain: PhysicalSecurity; SupplyChain; Hardware; Software; Abstraction Level: Standard	Attack Prerequisites: An adversary needs to have access to electronic data processing equipment being recycled or disposed of (e.g., laptops, servers) at a collection location and the ability to take control of it for the purpose of exploiting its content. Attack Execution Steps: No Execution Flow Provided
	CAPEC-150: Collect Data from Common Resource Locations. An adversary exploits well-known locations for resources for the purposes of undermining the security of the target. In many, if not most systems, files and resources are organized in a default tree structure. This can be useful for adversaries because they often know where to look for resources or files that are necessary for attacks. Even when the precise location of a targeted resource may not be known, naming conventions may indicate a small area of the target machine's file tree where the resources are typically located. For example, configuration files are normally stored in the /etc director on Unix systems. Adversaries can take advantage of this to commit other types of attacks. Domain: PhysicalSecurity; Hardware; Network; Software; Abstraction Level: Standard	Attack Prerequisites: The targeted applications must either expect files to be located at a specific location or, if the location of the files can be configured by the user, the user either failed to move the files from the default location or placed them in a conventional location for files of the given type. Attack Execution Steps: No Execution Flow Provided
	CAPEC-54: Query System for Information. An adversary, aware of an application's location (and possibly authorized to use the application), probes an application's structure and evaluates its robustness by submitting requests and examining responses. Often, this is accomplished by sending variants of expected queries in the hope that these modified queries might return information beyond what the expected set of queries would provide. Domain: Hardware; Software; Abstraction Level: Standard	Attack Prerequisites: This class of attacks does not strictly require authorized access to the application. As Attackers use this attack process to classify, map, and identify vulnerable aspects of an application, it simply requires hypotheses to be verified, interaction with the application, and time to conduct trial-and-error activities. Attack Execution Steps: Step 1 Explore: [Determine parameters] Determine all user-controllable parameters of the application either by probing or by finding documentation Step 2 Experiment: [Cause error condition] Inject each parameter with content that causes an error condition to manifest Step 3 Experiment: [Modify parameters] Modify the content of each parameter according to observed error conditions Step 4 Exploit: [Follow up attack] Once the above steps have been repeated with enough parameters, the application will be sufficiently mapped out. The adversary can then launch a desired attack (for example, Blind SQL Injection)
	CAPEC-545: Pull Data from System Resources. An adversary who is authorized or has the ability to search known system resources, does so with the intention of gathering useful information. System resources include files, memory, and other aspects of the target system. In this pattern of attack, the adversary does not necessarily know what they are going to find when they start pulling data. This is different than CAPEC-150 where the adversary knows what they are looking for due to the common location. Domain: Hardware; Software; Abstraction Level: Standard	Attack Prerequisites: No Prerequisite Provided Attack Execution Steps: No Execution Flow Provided
	CAPEC-569: Collect Data as Provided by Users. An attacker leverages a tool, device, or program to obtain specific information as provided by a user of the target system. This information is often needed by the attacker to launch a follow-on attack. This attack is different than Social Engineering as the adversary is not tricking or deceiving the user. Instead the adversary is putting a mechanism in place that captures the information that a user legitimately enters into a system. Deploying a keylogger, performing a UAC prompt, or wrapping the Windows default credential provider are all examples of such interactions. Domain: Software; Abstraction Level: Standard	Attack Prerequisites: No Prerequisite Provided Attack Execution Steps: No Execution Flow Provided

SE WORKBENCH ARTIFACTS SHARED WITH AGILE / DEVOPS DEVELOPMENT PROJECTS



QUESTIONS

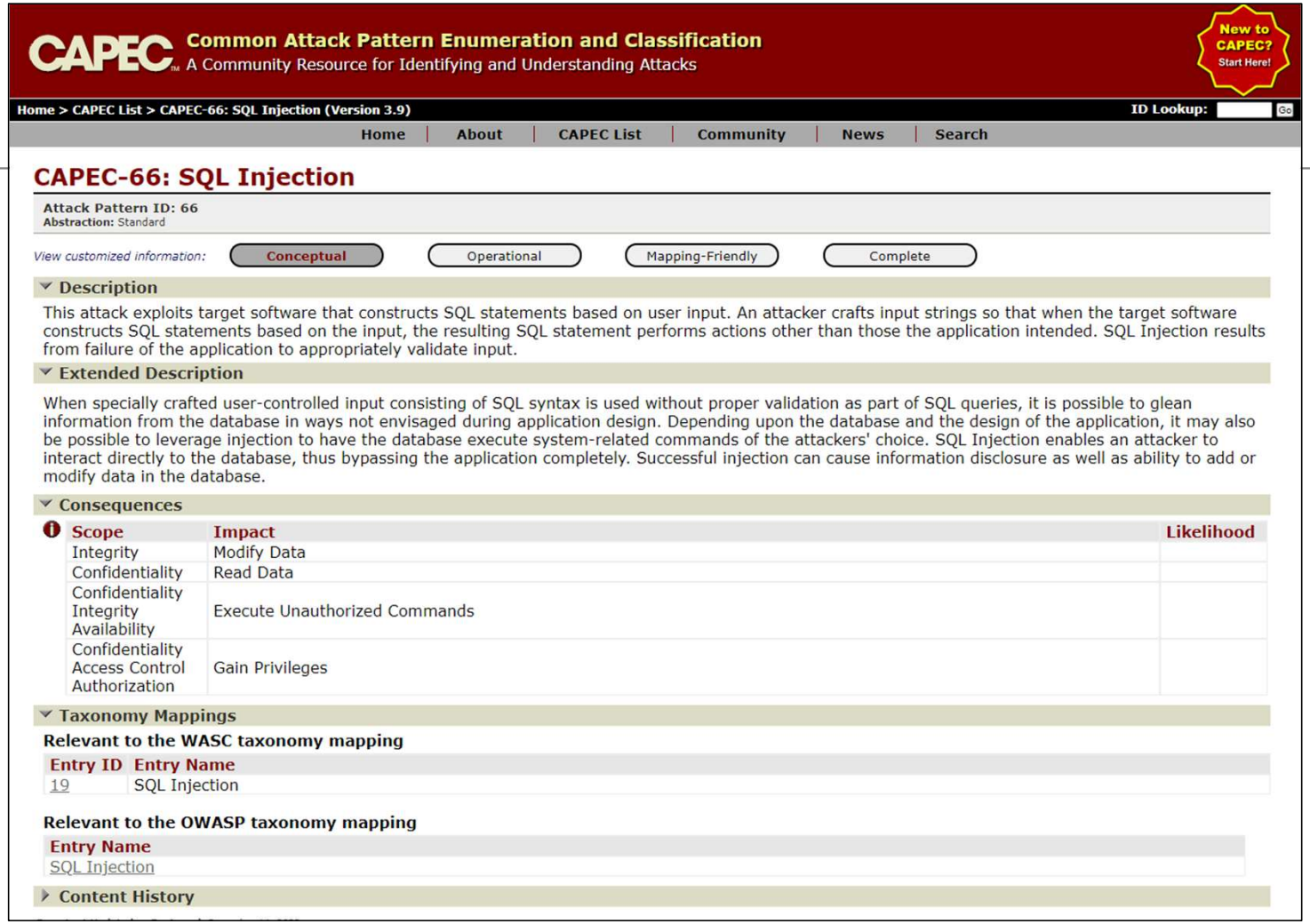
Topic #2

CAPEC Content Filters Modernization

Rich Piazza – CAPEC

Porting CWE Presentation Filters to CAPEC

CURRENT		PROPOSED			
BASIC	COMPLETE	**CONCEPTUAL**	**OPERATIONAL**	**MAPPING-FRIENDLY**	**COMPLETE**
Description	Description	Description	Description	Description	Description
Extended_Description	Extended_Description	Extended_Description	Extended_Description	Extended_Description	Extended_Description
Related_Attack_Patterns	Related_Attack_Patterns	Alternate_Terms	Alternate_Terms	Alternate_Terms	Related_Attack_Patterns
Execution_Flow	Execution_Flow	Taxonomy_Mappings	Related_Attack_Patterns	Related_Attack_Patterns	Execution_Flow
Prerequisites	Prerequisites	Consequences	Execution_Flow	Taxonomy_Mappings	Prerequisites
Mitigations	Mitigations		Prerequisites	Related_Weaknesses	Mitigations
Related_Weaknesses	Related_Weaknesses		Mitigations	Notes - Relationship	Related_Weaknesses
	Likelihood_Of_Attack		Related_Weaknesses	Notes - Terminology	Likelihood_Of_Attack
	Alternate_Terms				Alternate_Terms
	Typical_Severity				Typical_Severity
	Skills_Required				Skills_Required
	Resources_Required				Resources_Required
	Indicators				Indicators
	Consequences				Consequences
	Example_Instances				Example_Instances
	Taxonomy_Mappings				Taxonomy_Mappings
	References				References
	Notes				Notes
	Content_History				Content_History



CAPEC Common Attack Pattern Enumeration and Classification
A Community Resource for Identifying and Understanding Attacks
New to CAPEC? Start Here!
Home > CAPEC List > CAPEC-66: SQL Injection (Version 3.9)
ID Lookup: Go
Home | About | CAPEC List | Community | News | Search

CAPEC-66: SQL Injection

Attack Pattern ID: 66
Abstraction: Standard

View customized information: Conceptual | Operational | Mapping-Friendly | Complete

Description

This attack exploits target software that constructs SQL statements based on user input. An attacker crafts input strings so that when the target software constructs SQL statements based on the input, the resulting SQL statement performs actions other than those the application intended. SQL Injection results from failure of the application to appropriately validate input.

Extended Description

When specially crafted user-controlled input consisting of SQL syntax is used without proper validation as part of SQL queries, it is possible to glean information from the database in ways not envisaged during application design. Depending upon the database and the design of the application, it may also be possible to leverage injection to have the database execute system-related commands of the attackers' choice. SQL Injection enables an attacker to interact directly to the database, thus bypassing the application completely. Successful injection can cause information disclosure as well as ability to add or modify data in the database.

Consequences

Scope	Impact	Likelihood
Integrity	Modify Data	
Confidentiality	Read Data	
Confidentiality Integrity Availability	Execute Unauthorized Commands	
Confidentiality Access Control Authorization	Gain Privileges	

Taxonomy Mappings

Relevant to the WASC taxonomy mapping

Entry ID	Entry Name
19	SQL Injection

Relevant to the OWASP taxonomy mapping

Entry Name
SQL Injection

Content History

Reminders and Questions

- **Method for User Stories Collaboration/Development**
  - Google Docs?
  - Documenting User Stories

- **Get some more user stories across each user persona**
  - *Security Architect ('Rogue', Red Hat)*
  - *Hardware Verification Engineer (Jason Oberg, Cycuity)*
  - Educators
  - Technical Writers
  - Security Researchers/Analyst
  - etc.

Next Meeting – January 11 @ 12pm

NEW MEETING SERIES INVITE TO COME

PLEASE CONTACT WITH ANY QUESTIONS OR THOUGHTS

CWE@MITRE.ORG

CAPEC@MITRE.ORG