A PERSPECTIVE ON USE OF CWE/CAPEC IN EDUCATION

Jim Whitmore

jj-whitmore@comcast.net

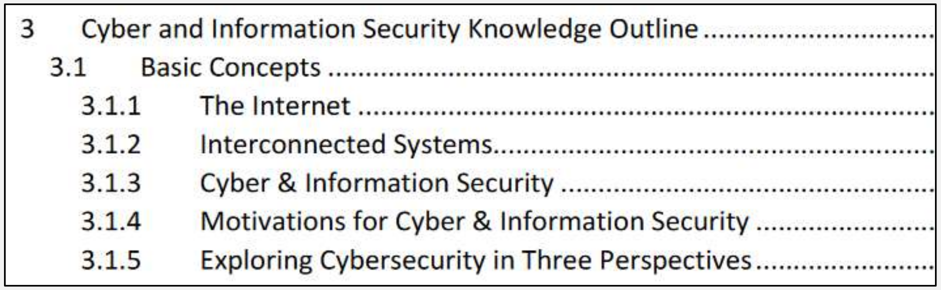
BACKGROUND

This is a collection of materials related to CWE/CAPEC that I have developed / used for cyber and information security education in academic and corporate settings.

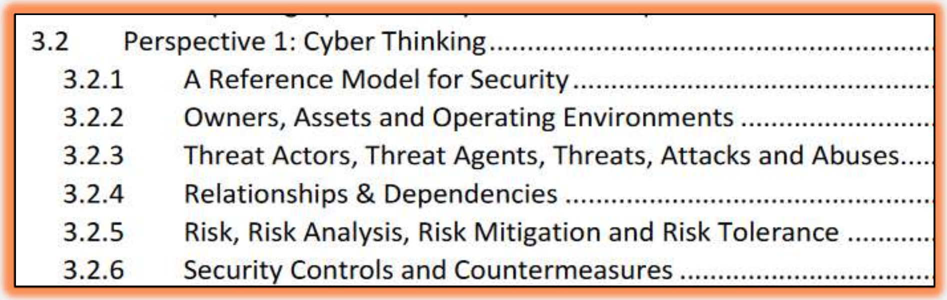
Philosophy:

1. Concepts from authoritative sources
2. Active learning is better than Passive learning (tools not rules)

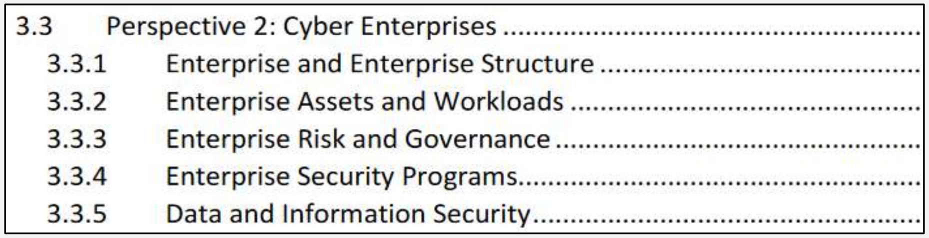
CYBER & INFORMATION SECURITY CURRICULUM OUTLINE



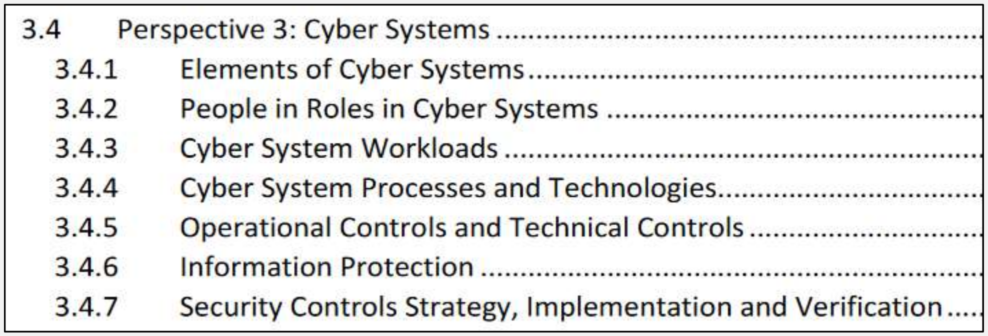
3 Cyber and Information Security Knowledge Outline
3.1 Basic Concepts
3.1.1 The Internet
3.1.2 Interconnected Systems
3.1.3 Cyber & Information Security
3.1.4 Motivations for Cyber & Information Security
3.1.5 Exploring Cybersecurity in Three Perspectives



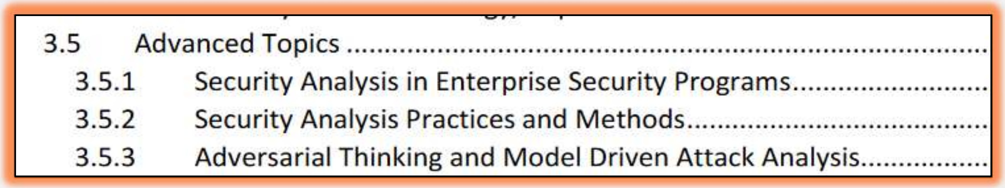
3.2 Perspective 1: Cyber Thinking
3.2.1 A Reference Model for Security
3.2.2 Owners, Assets and Operating Environments
3.2.3 Threat Actors, Threat Agents, Threats, Attacks and Abuses
3.2.4 Relationships & Dependencies
3.2.5 Risk, Risk Analysis, Risk Mitigation and Risk Tolerance
3.2.6 Security Controls and Countermeasures



3.3 Perspective 2: Cyber Enterprises
3.3.1 Enterprise and Enterprise Structure
3.3.2 Enterprise Assets and Workloads
3.3.3 Enterprise Risk and Governance
3.3.4 Enterprise Security Programs
3.3.5 Data and Information Security



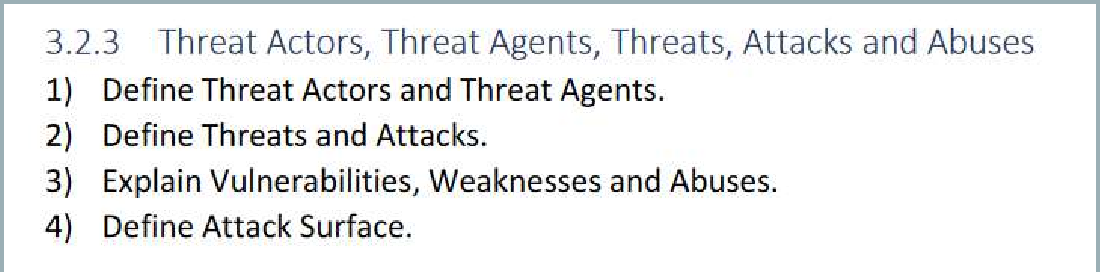
3.4 Perspective 3: Cyber Systems
3.4.1 Elements of Cyber Systems
3.4.2 People in Roles in Cyber Systems
3.4.3 Cyber System Workloads
3.4.4 Cyber System Processes and Technologies
3.4.5 Operational Controls and Technical Controls
3.4.6 Information Protection
3.4.7 Security Controls Strategy, Implementation and Verification



3.5 Advanced Topics
3.5.1 Security Analysis in Enterprise Security Programs
3.5.2 Security Analysis Practices and Methods
3.5.3 Adversarial Thinking and Model Driven Attack Analysis

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

PERSPECTIVE 1: CYBER THINKING



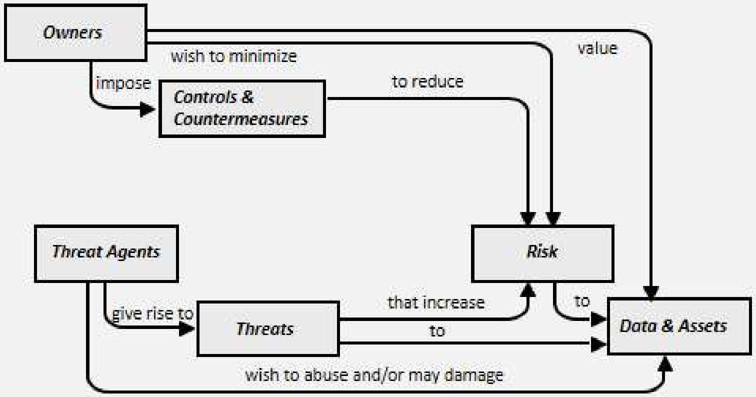
3.2.3 Threat Actors, Threat Agents, Threats, Attacks and Abuses

1) Define Threat Actors and Threat Agents.
2) Define Threats and Attacks.
3) Explain Vulnerabilities, Weaknesses and Abuses.
4) Define Attack Surface.

CYBER THINKING

… a thought model for security engineering

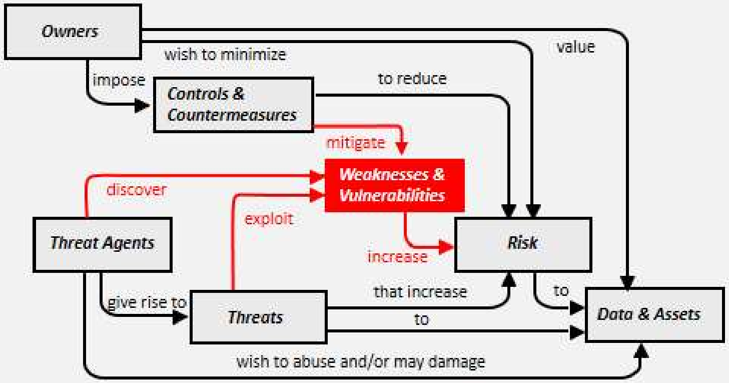
A GENERAL MODEL FOR THE CONCEPTS AND RELATIONSHIPS THAT INFLUENCE RISK.



Common Criteria for Information Technology Evaluation Part 1: Introduction & General Model
https://www.commoncriteriaportal.org/files/ccfiles/CCPART1V3.1R5.pdf

8/6/2022

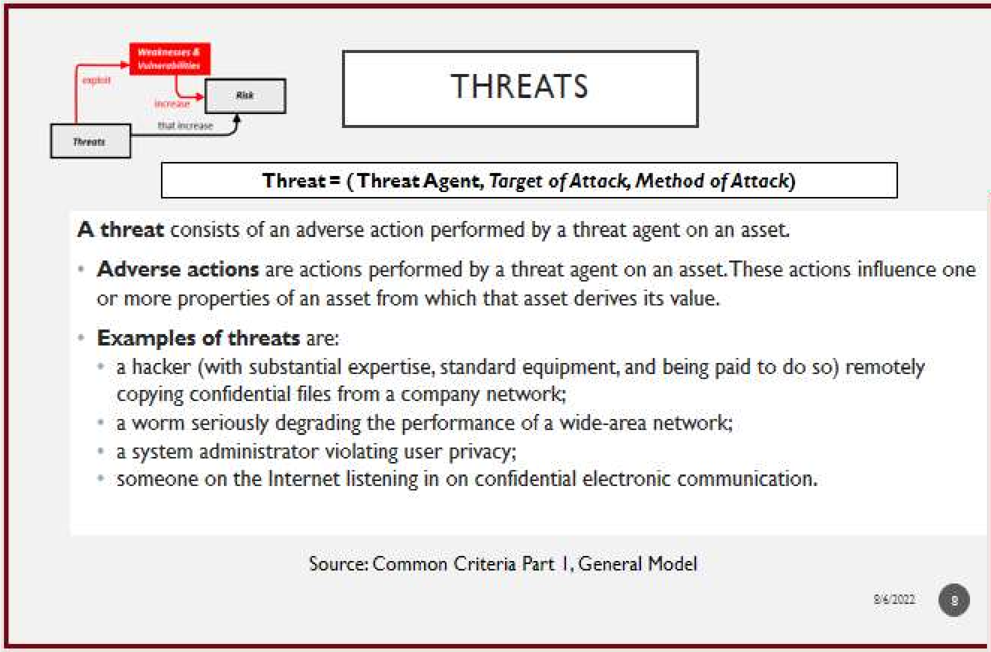
MODEL EXTENSION FOR IMPERFECT SYSTEMS



Common Criteria for Information Technology Evaluation Part 1: Introduction & General Model
https://www.commoncriteriaportal.org/files/ccfiles/CCPART1V3.1R5.pdf

8/6/2022 3

THREATS, WEAKNESSES & VULNERABILITIES



THREATS

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

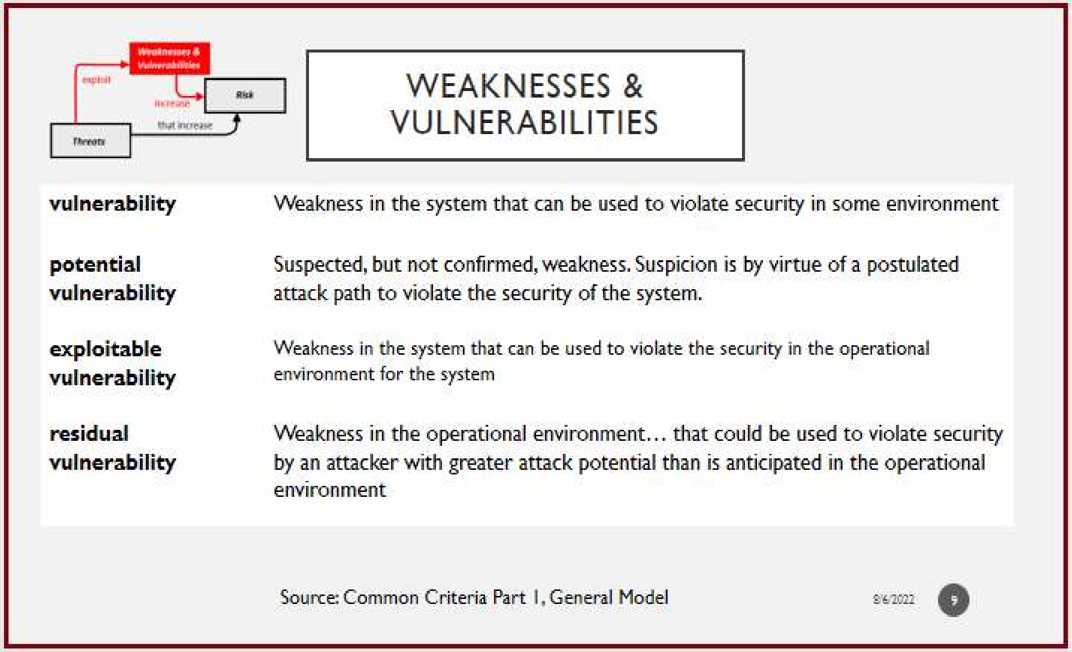
A threat consists of an adverse action performed by a threat agent on an asset.

- **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

8/6/2022 8

… the triad of security engineering (vs CIA)



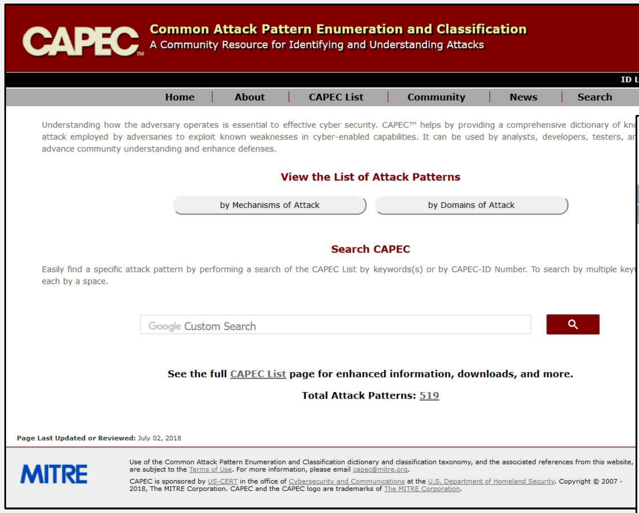
WEAKNESSES & VULNERABILITIES

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

8/6/2022 9

MITRE CORPORATION BUILDS AND MAINTAINS LIBRARIES PROVIDE A CORE BODY OF KNOWLEDGE



CAPEC™ **Common Attack Pattern Enumeration and Classification**
A Community Resource for Identifying and Understanding Attacks

Understanding how the adversary operates is essential to effective cyber security. CAPEC™ helps by providing a comprehensive dictionary of kno
attack employed by adversaries to exploit known weaknesses in cyber-enabled capabilities. It can be used by analysts, developers, testers, an
advance community understanding and enhance defenses.

View the List of Attack Patterns

Search CAPEC

Easily find a specific attack pattern by performing a search of the CAPEC List by keywords(s) or by CAPEC-ID Number. To search by multiple key
each by a space.

See the full CAPEC List page for enhanced information, downloads, and more.

Total Attack Patterns: 519

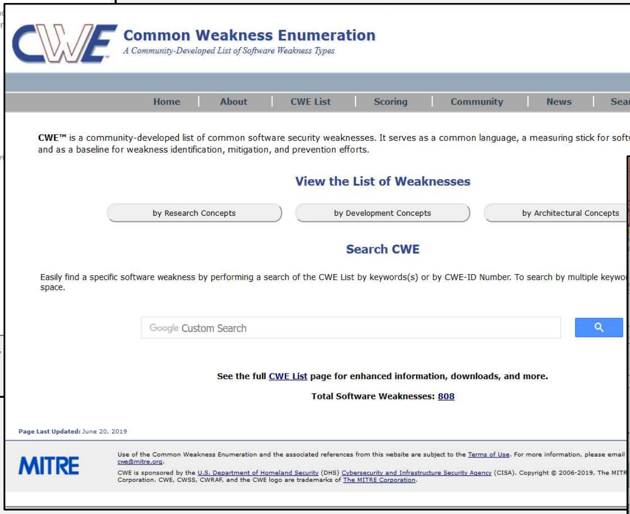
Page Last Updated or Reviewed: July 02, 2018

MITRE

Use of the Common Attack Pattern Enumeration and Classification dictionary and classification taxonomy, and the associated references from this website, are subject to the Terms of Use. For more information, please email capec@mitre.org.

CAPEC is sponsored by US-CERT in the office of Cybersecurity and Communications at the U.S. Department of Homeland Security. Copyright © 2007 - 2018, The MITRE Corporation. CAPEC and the CAPEC logo are trademarks of The MITRE Corporation.

http://capec.mitre.org/



CWE **Common Weakness Enumeration**
A Community-Developed List of Software Weakness Types

CWE and SANS Institute
TOP 25 MOST DANGEROUS SOFTWARE ERRORS
ID Lookup:
Go
Home | About | CWE List | Scoring | Community | News | Search

CWE™ is a community-developed list of common software security weaknesses. It serves as a common language, a measuring stick for software security tools, and as a baseline for weakness identification, mitigation, and prevention efforts.

View the List of Weaknesses

by Research Concepts
by Development Concepts
by Architectural Concepts

Search CWE

Easily find a specific software weakness by performing a search of the CWE List by keywords(s) or by CWE-ID Number. To search by multiple keywo
space.

See the full CWE List page for enhanced information, downloads, and more.

Total Software Weaknesses: 808

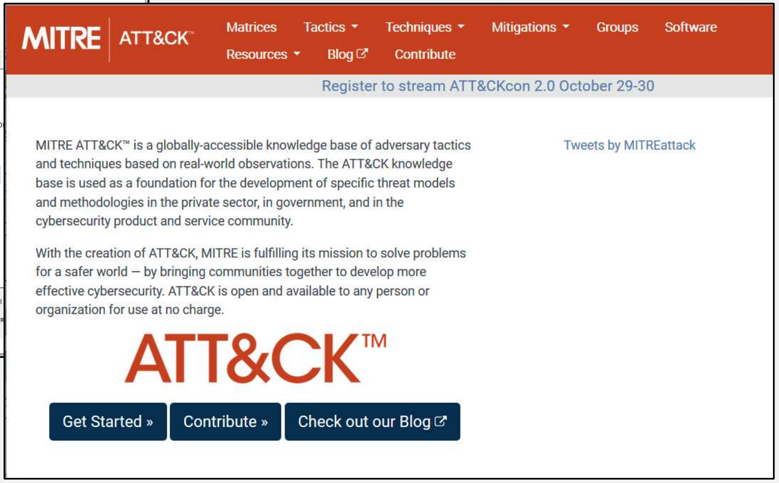
Page Last Updated: June 20, 2019

MITRE

Use of the Common Weakness Enumeration and the associated references from this website are subject to the Terms of Use. For more information, please email cwe@mitre.org.

CWE is sponsored by the U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA). Copyright © 2006-2019, The MITR
Corporation. CWE, CWSS, CWRAF, and the CWE logo are trademarks of The MITRE Corporation.

https://cwe.mitre.org/



Register to stream ATT&CKcon 2.0 October 29-30

MITRE ATT&CK™ is a globally-accessible knowledge base of adversary tactics and techniques based on real-world observations. The ATT&CK knowledge base is used as a foundation for the development of specific threat models and methodologies in the private sector, in government, and in the cybersecurity product and service community.

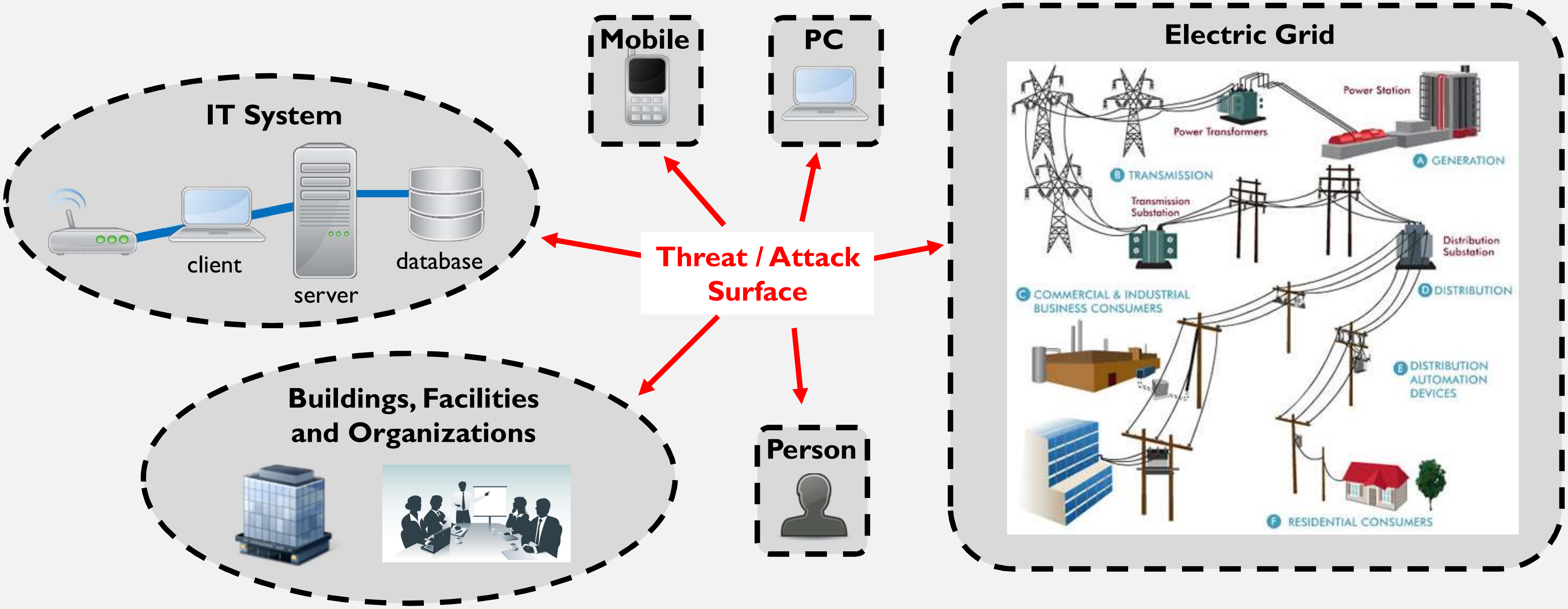
With the creation of ATT&CK, MITRE is fulfilling its mission to solve problems for a safer world — by bringing communities together to develop more effective cybersecurity. ATT&CK is open and available to any person or organization for use at no charge.

Tweets by MITREattack

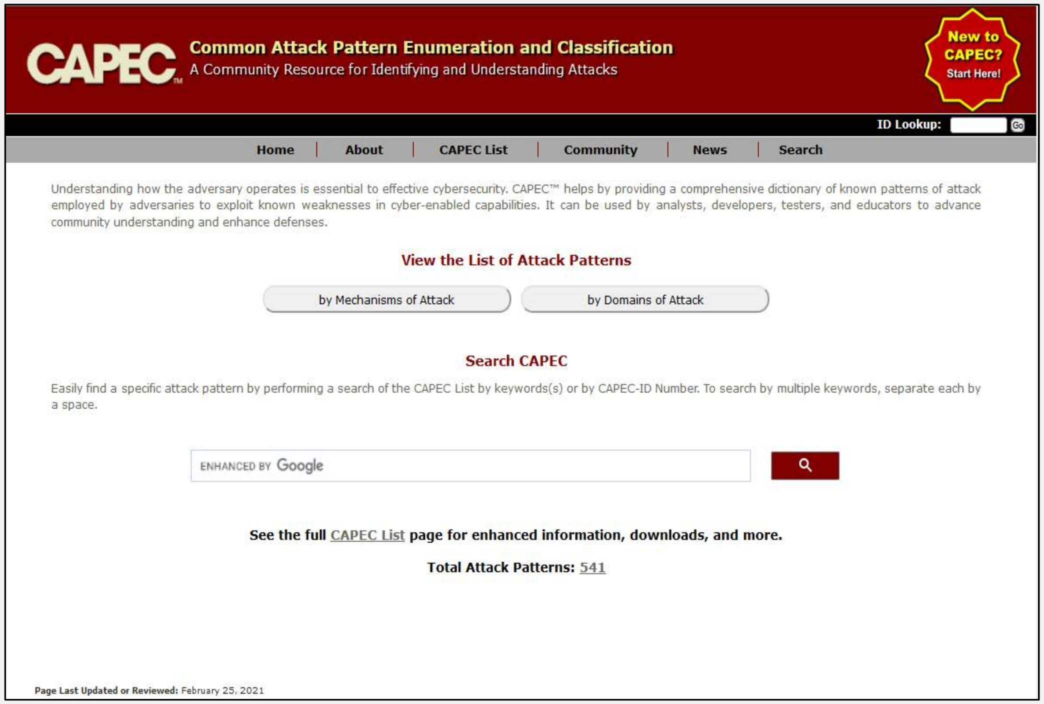
http://attack.mitre.org/

AN ATTACK SURFACE IS THE SUM OF THREAT VECTORS FOR A TARGET

Threat Surface = ∑ Threats



WE CAN ANALYZE THE ATTACK SURFACE USING CAPEC AND ITS LINKAGES TO CWE, ETC.


CAPEC™
Common Attack Pattern Enumeration and Classification
A Community Resource for Identifying and Understanding Attacks
New to CAPEC? Start Here!
ID Lookup:
Go
Home | About | CAPEC List | Community | News | Search

Understanding how the adversary operates is essential to effective cybersecurity. CAPEC™ helps by providing a comprehensive dictionary of known patterns of attack employed by adversaries to exploit known weaknesses in cyber-enabled capabilities. It can be used by analysts, developers, testers, and educators to advance community understanding and enhance defenses.

View the List of Attack Patterns

Search CAPEC

Easily find a specific attack pattern by performing a search of the CAPEC List by keywords(s) or by CAPEC-ID Number. To search by multiple keywords, separate each by a space.

See the full CAPEC List page for enhanced information, downloads, and more.

Total Attack Patterns: 541

Page Last Updated or Reviewed: February 25, 2021

https://capec.mitre.org/

Domains of Attack

Physical Security

Hardware

Supply Chain

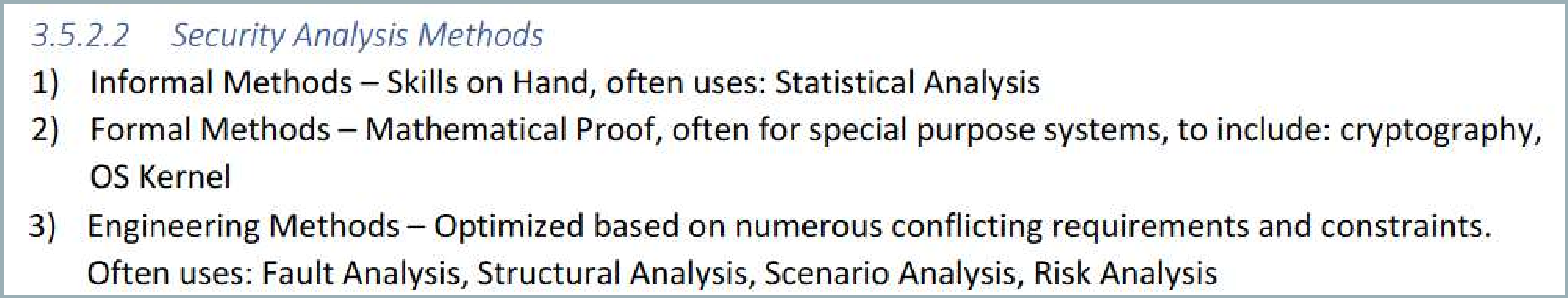
Software

Communications

Social Engineering

https://capec.mitre.org/data/definitions/3000.html

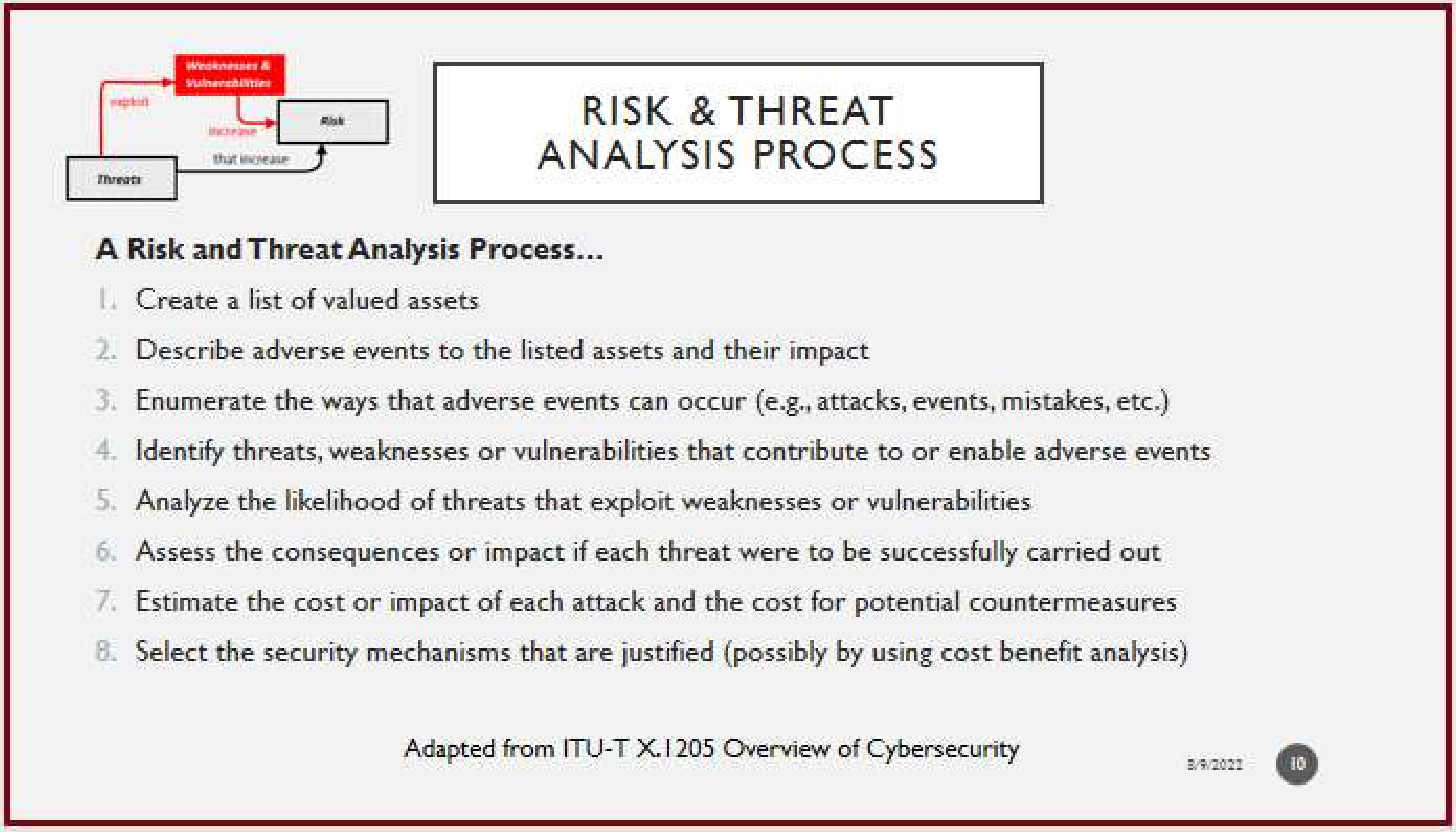
ADVANCED TOPICS: METHODS, TOOLS AND PRACTICES



3.5.2.2 Security Analysis Methods

1) Informal Methods – Skills on Hand, often uses: Statistical Analysis
2) Formal Methods – Mathematical Proof, often for special purpose systems, to include: cryptography, OS Kernel
3) Engineering Methods – Optimized based on numerous conflicting requirements and constraints. Often uses: Fault Analysis, Structural Analysis, Scenario Analysis, Risk Analysis

SECURITY ANALYSIS PROCESS


Weaknesses & Vulnerabilities
exploit
increase
Risk
that increase
Threats

RISK & THREAT ANALYSIS PROCESS

A Risk and Threat Analysis Process…

1. Create a list of valued assets
2. Describe adverse events to the listed assets and their impact
3. Enumerate the ways that adverse events can occur (e.g., attacks, events, mistakes, etc.)
4. Identify threats, weaknesses or vulnerabilities that contribute to or enable adverse events
5. Analyze the likelihood of threats that exploit weaknesses or vulnerabilities
6. Assess the consequences or impact if each threat were to be successfully carried out
7. Estimate the cost or impact of each attack and the cost for potential countermeasures
8. Select the security mechanisms that are justified (possibly by using cost benefit analysis)

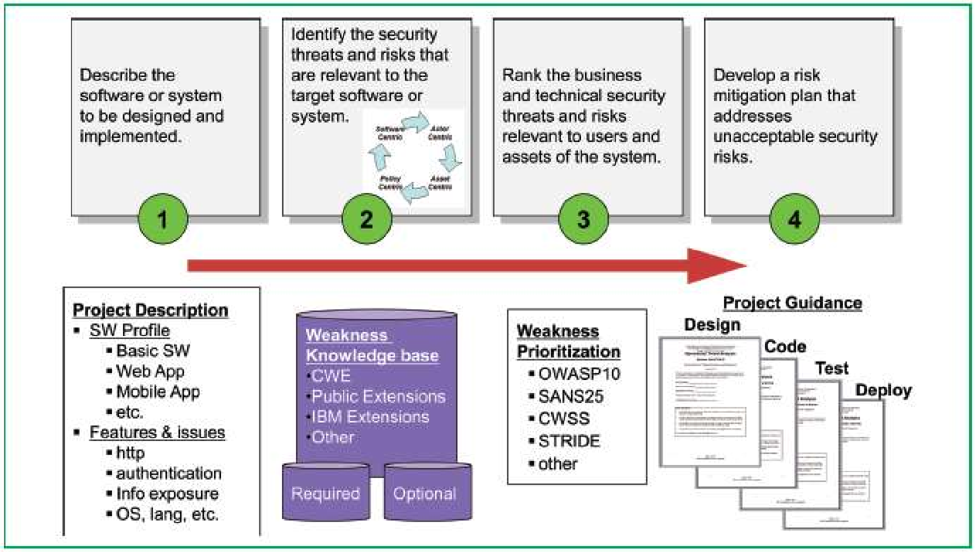
Adapted from ITU-T X.1205 Overview of Cybersecurity

Two approaches:

1. Mitigate Defects and Weaknesses
2. Mitigate Attacks

DEFECT/WEAKNESS MITIGATION APPROACH (CWE)

Threat analysis in the software development lifecycle
IBM Journal of Research and Development, 2014.
J Whitmore, S Türpe, S Triller, A Poller, C Carlson.



1. Describe the software or system to be designed and implemented.
2. Identify the security threats and risks that are relevant to the target software or system.
3. Rank the business and technical security threats and risks relevant to users and assets of the system.
4. Develop a risk mitigation plan that addresses unacceptable security risks.

Project Description
- SW Profile
  - Basic SW
  - Web App
  - Mobile App
  - etc.
- Features & issues
  - http
  - authentication
  - Info exposure
  - OS, lang, etc.

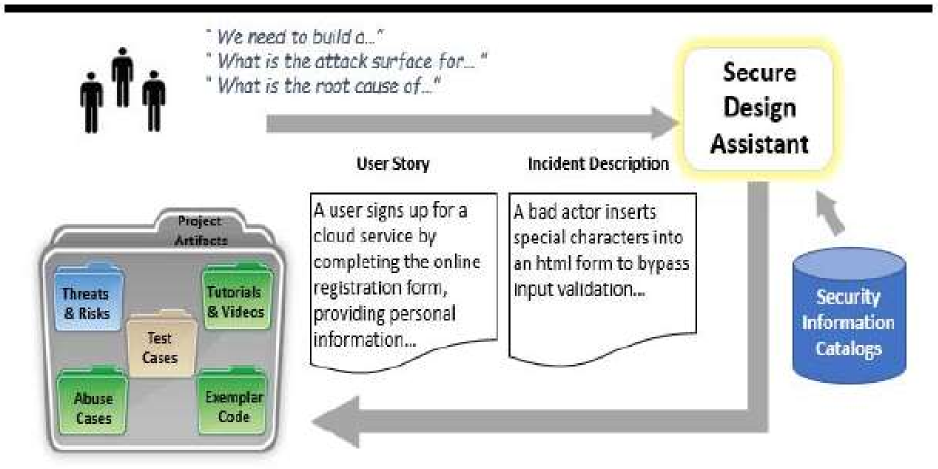
Weakness Knowledge base
- CWE
- Public Extensions
- IBM Extensions
- Other

Weakness Prioritization
- OWASP10
- SANS25
- CWSS
- STRIDE
- other

Project Guidance

ATTACK MITIGATION APPROACH (CAPEC)

Improving Attention to Security in Software Design with Analytics and Cognitive Techniques IEEE Cybersecurity Development (SecDev), 2017. J Whitmore, W Tobin



" We need to build a..."
" What is the attack surface for... "
" What is the root cause of..."

User Story: A user signs up for a cloud service by completing the online registration form, providing personal information...

Incident Description: A bad actor inserts special characters into an html form to bypass input validation...

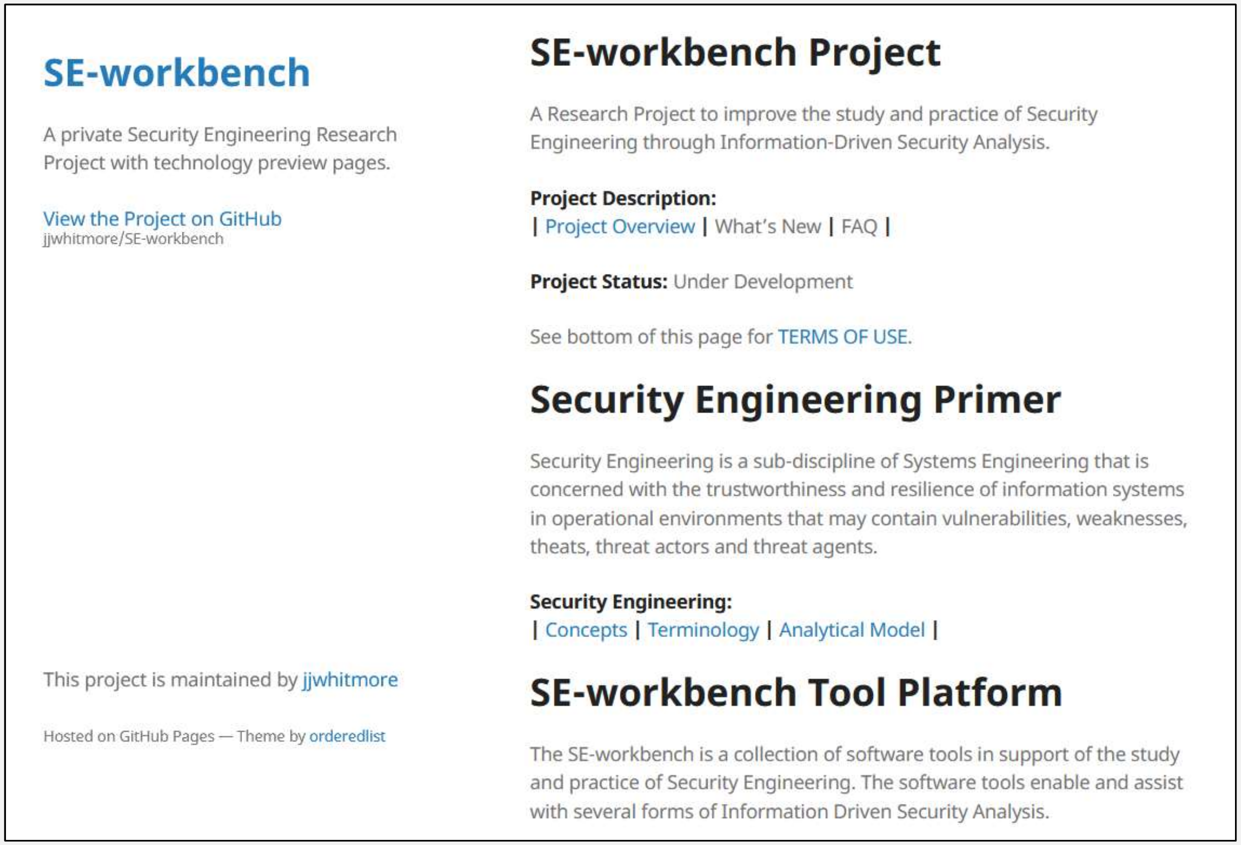
Figure 6. Security design / analysis as a transaction.

SE-WORKBENCH IS A PLATFORM FOR ACTIVE LEARNING

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.



SE-workbench

A private Security Engineering Research Project with technology preview pages.

View the Project on GitHub
jjwhitmore/SE-workbench

This project is maintained by jjwhitmore

Hosted on GitHub Pages — Theme by orderedlist

SE-workbench Project

A Research Project to improve the study and practice of Security Engineering through Information-Driven Security Analysis.

Project Description:
| Project Overview | What's New | FAQ |

Project Status: Under Development

See bottom of this page for TERMS OF USE.

Security Engineering Primer

Security Engineering is a sub-discipline of Systems Engineering that is concerned with the trustworthiness and resilience of information systems in operational environments that may contain vulnerabilities, weaknesses, theats, threat actors and threat agents.

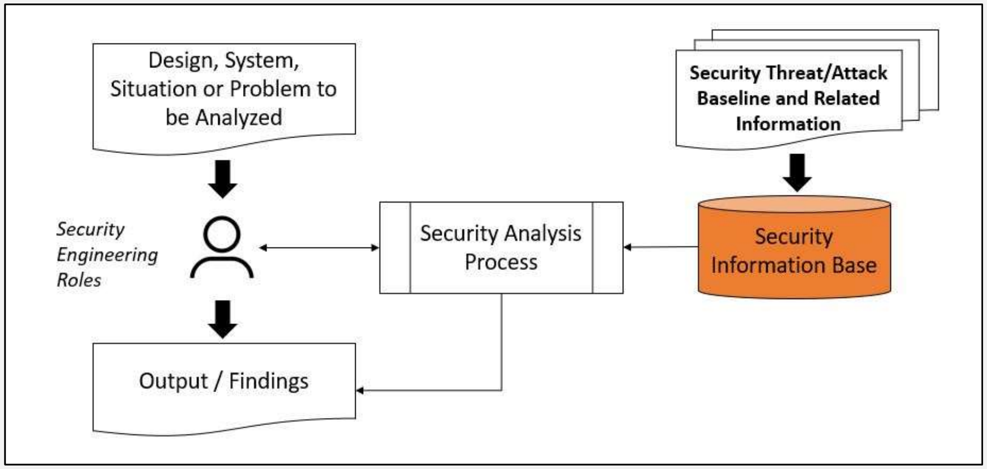
Security Engineering:
| Concepts | Terminology | Analytical Model |

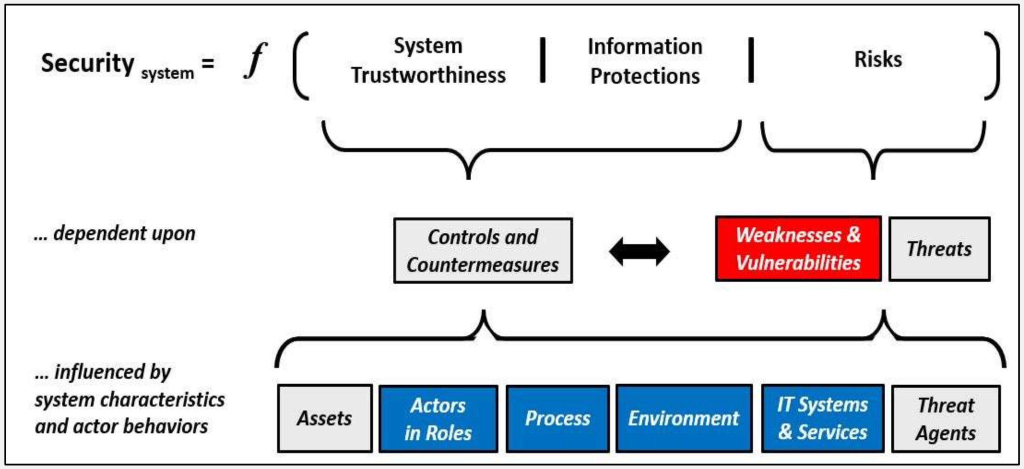
SE-workbench Tool Platform

The SE-workbench is a collection of software tools in support of the study and practice of Security Engineering. The software tools enable and assist with several forms of Information Driven Security Analysis.

https://jjwhitmore.github.io/SE-workbench/

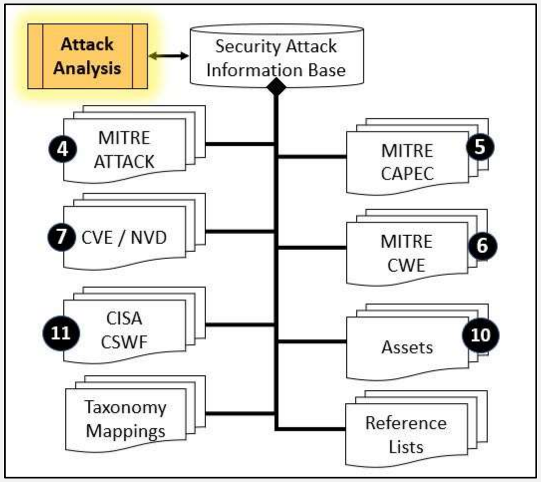
SE-WORKBENCH
INCLUDES A SECURITY ENGINEERING PRIMER


Design, System, Situation or Problem to be Analyzed
Security Threat/Attack Baseline and Related Information
Security Engineering Roles
Security Analysis Process
Security Information Base
Output / Findings

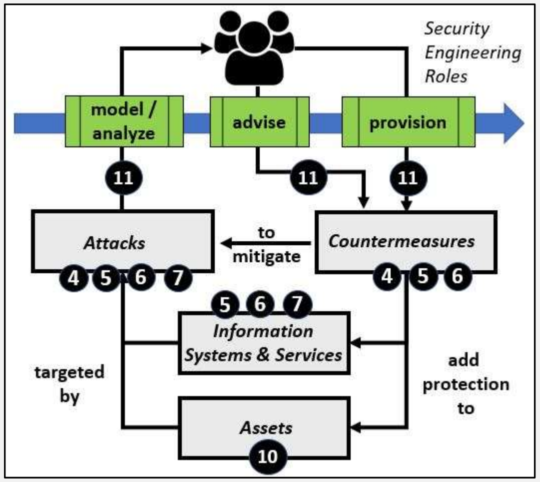


SE-WORKBENCH
INCLUDES A TUTORIAL FOR EACH TOOL

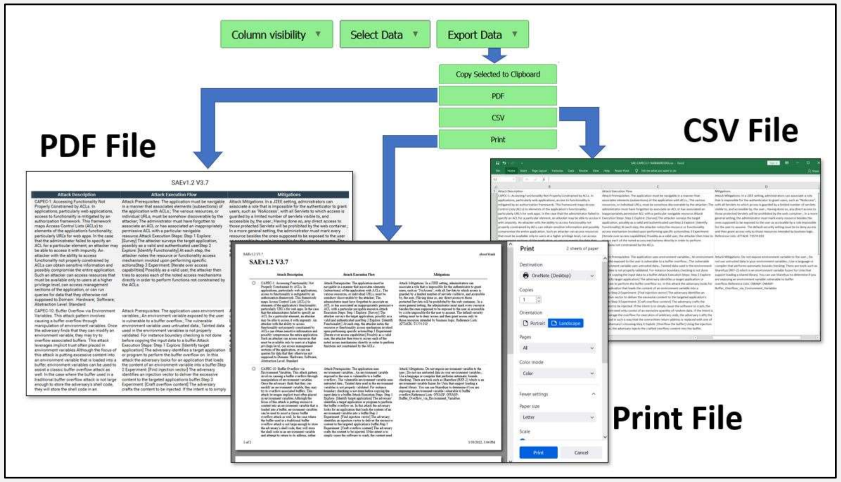
Data Model



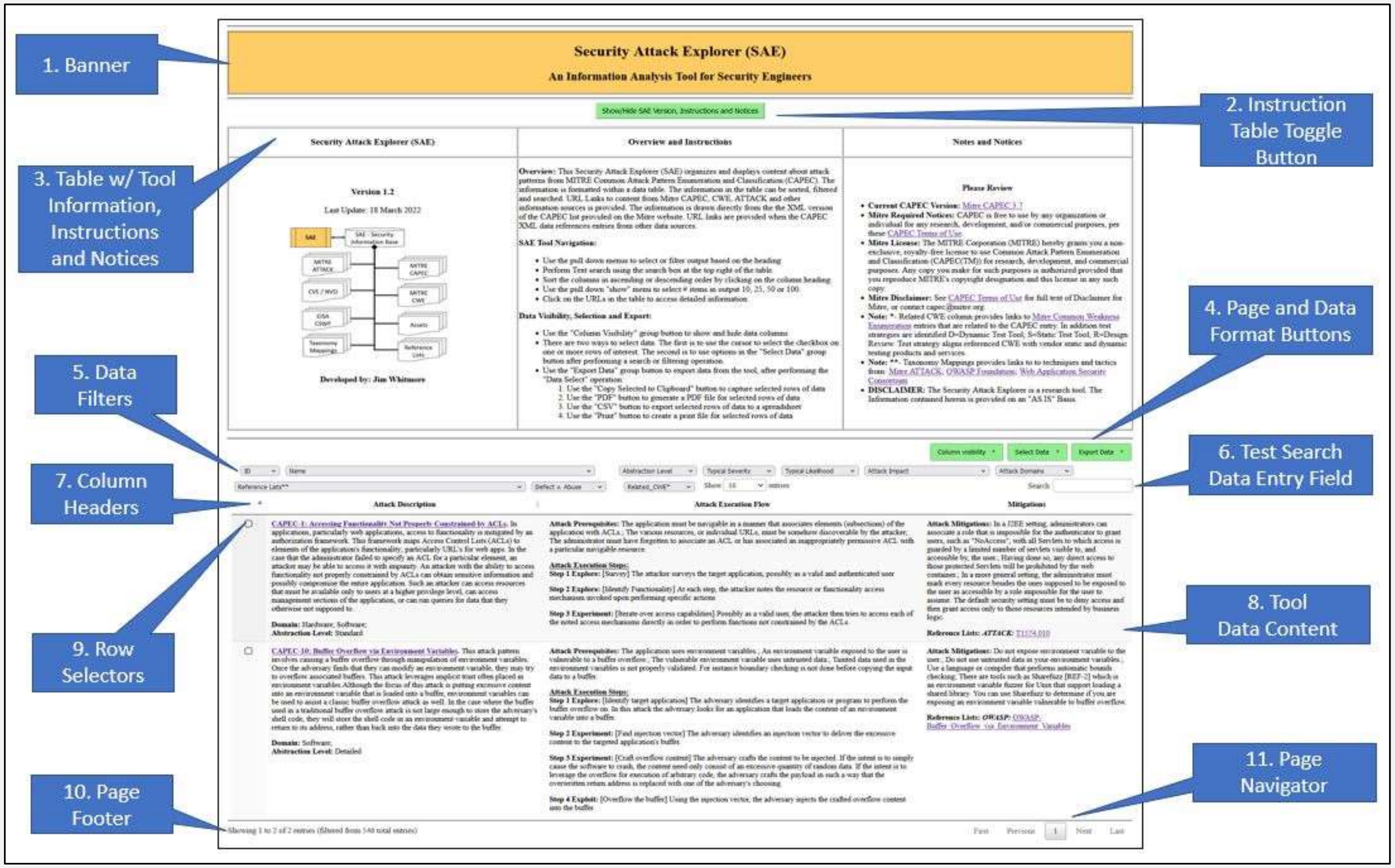
Process



Output



SE-WORKBENCH TOOLS HAVE A COMMON UI



Security Attack Explorer (SAE)

An Information Analysis Tool for Security Engineers

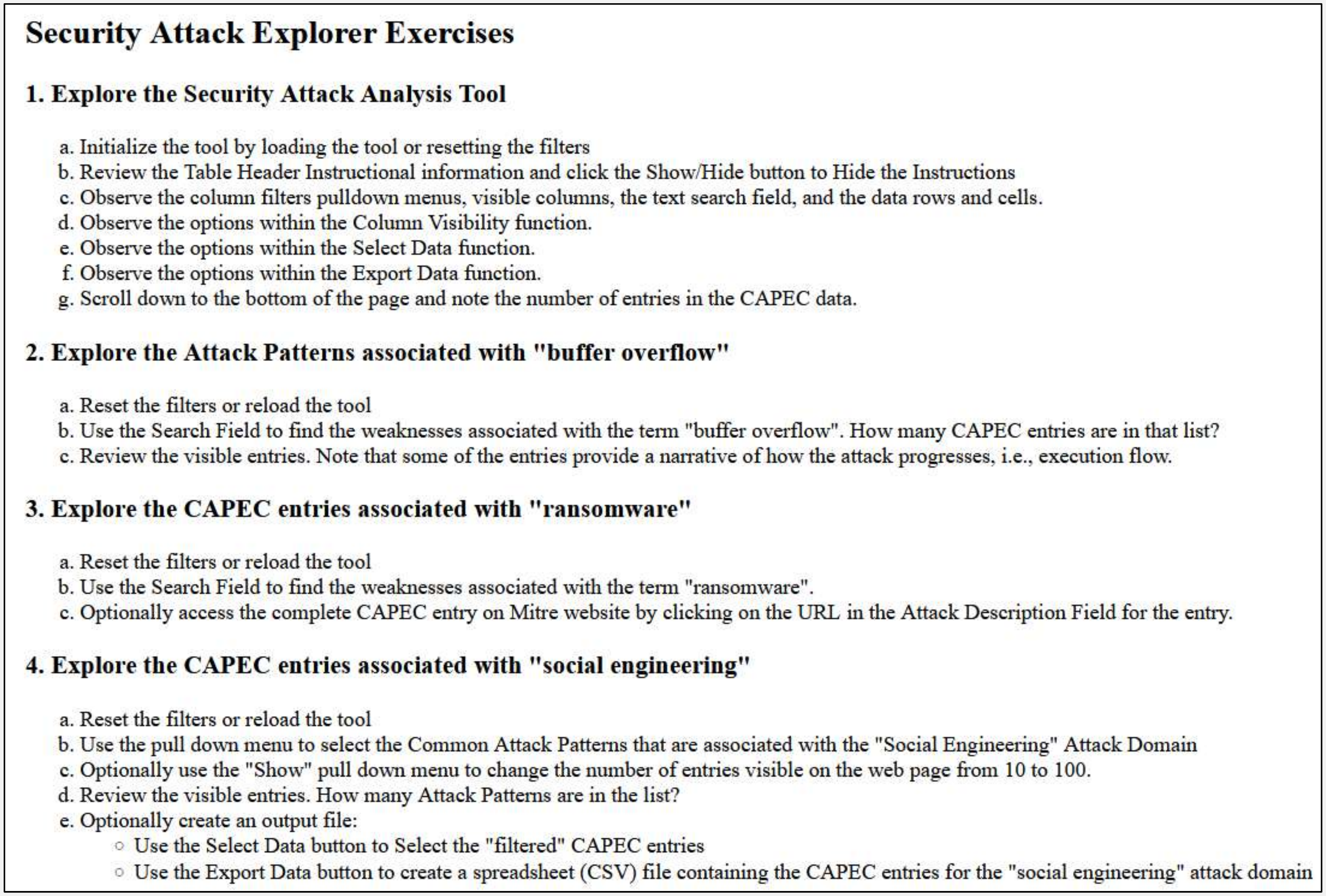
Security Attack Explorer (SAE)	Overview and Instructions	Notes and Notices
Version 1.2 Last Update: 18 March 2022 **Developed by: Jim Whitmore**	**Overview:** This Security Attack Explorer (SAE) organizes and displays content about attack patterns from MITRE Common Attack Pattern Enumeration and Classification (CAPEC). The information is formatted within a data table. The information in the table can be sorted, filtered and searched. URL Links to content from Mitre CAPEC, CWE, ATTACK and other information sources is provided. The information is drawn directly from the the XML version of the CAPEC list provided on the Mitre website. URL links are provided when the CAPEC XML data references entries from other data sources. **SAE Tool Navigation:** • Use the pull down menus to select or filter output based on the heading • Perform Text search using the search box at the top right of the table • Sort the columns in ascending or descending order by clicking on the column heading • Use the pull down "show" menu to select # items in output 10, 25, 50 or 100 • Click on the URLs in the table to access detailed information **Data Visibility, Selection and Export:** • Use the "Column Visibility" group button to show and hide data columns • There are two ways to select data. The first is to use the cursor to select the checkbox on one or more rows of interest. The second is to use options in the "Select Data" group button after performing a search or filtering operation • Use the "Export Data" group button to export data from the tool, after performing the "Data Select" operation 1. Use the "Copy Selected to Clipboard" button to capture selected rows of data 2. Use the "PDF" button to generate a PDF file for selected rows of data 3. Use the "CSV" button to export selected rows of data to a spreadsheet 4. Use the "Print" button to create a print file for selected rows of data	**Please Review** • **Current CAPEC Version:** Mitre CAPEC 3.7 • **Mitre Required Notices:** CAPEC is free to use by any organization or individual for any research, development, and/or commercial purposes, per these CAPEC Terms of Use • **Mitre License:** The MITRE Corporation (MITRE) hereby grants you a non-exclusive, royalty-free license to use Common Attack Pattern Enumeration and Classification (CAPEC(TM)) for research, development, and commercial purposes. Any copy you make for such purposes is authorized provided that you reproduce MITRE's copyright designation and this license in any such copy. • **Mitre Disclaimer:** See CAPEC Terms of Use for full text of Disclaimer for Mitre, or contact capec@mitre.org • **Note:** *- Related CWE column provides links to Mitre Common Weakness Enumeration entries that are related to the CAPEC entry. In addition test strategies are identified D=Dynamic Test Tool, S=Static Test Tool, R=Design Review. Test strategy aligns referenced CWE with vendor static and dynamic testing products and services. • **Note:** **- Taxonomy Mappings provides links to to techniques and tactics from: Mitre ATTACK, OWASP Foundation; Web Application Security Consortium • **DISCLAIMER:** The Security Attack Explorer is a research tool. The Information contained herein is provided on an "AS IS" Basis.

^	Attack Description	Attack Execution Flow	Mitigations
○	CAPEC-1: Accessing Functionality Not Properly Constrained by ACLs. In applications, particularly web applications, access to functionality is mitigated by an authorization framework. This framework maps Access Control Lists (ACLs) to elements of the application's functionality, particularly URL's for web apps. In the case that the administrator failed to specify an ACL for a particular element, an attacker may be able to access it with impunity. An attacker with the ability to access functionality not properly constrained by ACLs can obtain sensitive information and possibly compromise the entire application. Such an attacker can access resources that must be available only to users at a higher privilege level, can access management sections of the application, or can run queries for data that they otherwise not supposed to. **Domain:** Hardware; Software; **Abstraction Level:** Standard	**Attack Prerequisites:** The application must be navigable in a manner that associates elements (subsections) of the application with ACLs.; The various resources, or individual URLs, must be somehow discoverable by the attacker; The administrator must have forgotten to associate an ACL or has associated an inappropriately permissive ACL with a particular navigable resource. **Attack Execution Steps:** **Step 1 Explore:** [Survey] The attacker surveys the target application, possibly as a valid and authenticated user **Step 2 Explore:** [Identify Functionality] At each step, the attacker notes the resource or functionality access mechanism invoked upon performing specific actions **Step 3 Experiment:** [Iterate over access capabilities] Possibly as a valid user, the attacker then tries to access each of the noted access mechanisms directly in order to perform functions not constrained by the ACLs.	**Attack Mitigations:** In a J2EE setting, administrators can associate a role that is impossible for the authenticator to grant users, such as "NoAccess", with all Servlets to which access is guarded by a limited number of servlets visible to, and accessible by, the user.; Having done so, any direct access to those protected Servlets will be prohibited by the web container.; In a more general setting, the administrator must mark every resource besides the ones supposed to be exposed to the user as accessible by a role impossible for the user to assume. The default security setting must be to deny access and then grant access only to those resources intended by business logic. **Reference Lists:** ***ATTACK:*** T1574.010
□	CAPEC-10: Buffer Overflow via Environment Variables. This attack pattern involves causing a buffer overflow through manipulation of environment variables. Once the adversary finds that they can modify an environment variable, they may try to overflow associated buffers. This attack leverages implicit trust often placed in environment variables. Although the focus of this attack is putting excessive content into an environment variable that is loaded into a buffer, environment variables can be used to assist a classic buffer overflow attack as well. In the case where the buffer used in a traditional buffer overflow attack is not large enough to store the adversary's shell code, they will store the shell code in an environment variable and attempt to return to its address, rather than back into the data they wrote to the buffer. **Domain:** Software; **Abstraction Level:** Detailed	**Attack Prerequisites:** The application uses environment variables.; An environment variable exposed to the user is vulnerable to a buffer overflow.; The vulnerable environment variable uses untrusted data.; Tainted data used in the environment variables is not properly validated. For instance boundary checking is not done before copying the input data to a buffer. **Attack Execution Steps:** **Step 1 Explore:** [Identify target application] The adversary identifies a target application or program to perform the buffer overflow on. In this attack the adversary looks for an application that loads the content of an environment variable into a buffer. **Step 2 Experiment:** [Find injection vector] The adversary identifies an injection vector to deliver the excessive content to the targeted application's buffer. **Step 3 Experiment:** [Craft overflow content] The adversary crafts the content to be injected. If the intent is to simply cause the software to crash, the content need only consist of an excessive quantity of random data. If the intent is to leverage the overflow for execution of arbitrary code, the adversary crafts the payload in such a way that the overwritten return address is replaced with one of the adversary's choosing. **Step 4 Exploit:** [Overflow the buffer] Using the injection vector, the adversary injects the crafted overflow content into the buffer.	**Attack Mitigations:** Do not expose environment variable to the user.; Do not use untrusted data in your environment variables.; Use a language or compiler that performs automatic bounds checking. There are tools such as Sharefuzz [REF-2] which is an environment variable fuzzer for Unix that support loading a shared library. You can use Sharefuzz to determine if you are exposing an environment variable vulnerable to buffer overflow. **Reference Lists:** ***OWASP:*** OWASP: Buffer_Overflow_via_Environment_Variables

Showing 1 to 2 of 2 entries (filtered from 540 total entries)

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

SE-WORKBENCH INCLUDES EXERCISES FOR EACH TOOL



Security Attack Explorer Exercises

1. Explore the Security Attack Analysis Tool

a. Initialize the tool by loading the tool or resetting the filters
b. Review the Table Header Instructional information and click the Show/Hide button to Hide the Instructions
c. Observe the column filters pulldown menus, visible columns, the text search field, and the data rows and cells.
d. Observe the options within the Column Visibility function.
e. Observe the options within the Select Data function.
f. Observe the options within the Export Data function.
g. Scroll down to the bottom of the page and note the number of entries in the CAPEC data.

2. Explore the Attack Patterns associated with "buffer overflow"

a. Reset the filters or reload the tool
b. Use the Search Field to find the weaknesses associated with the term "buffer overflow". How many CAPEC entries are in that list?
c. Review the visible entries. Note that some of the entries provide a narrative of how the attack progresses, i.e., execution flow.

3. Explore the CAPEC entries associated with "ransomware"

a. Reset the filters or reload the tool
b. Use the Search Field to find the weaknesses associated with the term "ransomware".
c. Optionally access the complete CAPEC entry on Mitre website by clicking on the URL in the Attack Description Field for the entry.

4. Explore the CAPEC entries associated with "social engineering"

a. Reset the filters or reload the tool
b. Use the pull down menu to select the Common Attack Patterns that are associated with the "Social Engineering" Attack Domain
c. Optionally use the "Show" pull down menu to change the number of entries visible on the web page from 10 to 100.
d. Review the visible entries. How many Attack Patterns are in the list?
e. Optionally create an output file:
   - Use the Select Data button to Select the "filtered" CAPEC entries
   - Use the Export Data button to create a spreadsheet (CSV) file containing the CAPEC entries for the "social engineering" attack domain

THE END