**Stage -1**

It seems there might be a misunderstanding in your question. The SANS Institute is a well-known organization in the field of cybersecurity, and the term "SANS 20" is not a widely recognized framework or standard associated with them. However, the SANS Institute is known for the SANS Critical Security Controls (CSC), which is a set of best practices designed to help organizations improve their cybersecurity posture.

The SANS Critical Security Controls (CSC) provide a prioritized approach to cybersecurity, focusing on the most effective actions that organizations can take to enhance their security. The controls are divided into three main categories:

Basic Controls:

Inventory of Authorized and Unauthorized Devices

Inventory of Authorized and Unauthorized Software

Secure Configurations for Hardware and Software on Mobile Devices, Laptops, Workstations, and Servers

Continuous Vulnerability Assessment and Remediation

Controlled Use of Administrative Privileges

Maintenance, Monitoring, and Analysis of Audit Logs

Foundational Controls:

Email and Web Browser Protections

Malware Defenses

Limitation and Control of Network Ports, Protocols, and Services

Data Protection

Secure Configurations for Network Devices, such as Firewalls, Routers, and Switches

Boundary Defense

Organizational Controls:

Data Recovery Capabilities

Security Skills Assessment and Appropriate Training to Fill Gaps

Secure Configurations for Enterprise Cloud Resources

Incident Response and Management

Penetration Tests and Red Team Exercises

Implementing a Security Awareness and Training Program

The controls are regularly updated to address emerging threats and changes in the cybersecurity landscape. Organizations can use the SANS Critical Security Controls as a framework to guide their cybersecurity efforts and improve their overall security posture.

If you were referring to a different concept or framework with "SANS 20," please provide more context so that I can offer more accurate information.

**AI-Based Threat Identification Within SANS 20**

As of my last knowledge update in January 2022, the SANS 20 (or SANS Critical Security Controls) themselves do not explicitly specify the use of artificial intelligence (AI) for threat identification. However, it's important to note that the field of cybersecurity is dynamic, and practices may evolve over time.

The use of AI in cybersecurity has been a growing trend, and organizations often leverage machine learning and other AI techniques to enhance their threat detection and response capabilities. AI can be applied across various areas within the SANS 20 framework, including:

Continuous Vulnerability Assessment and Remediation (Control 4): AI can be employed to automate vulnerability scanning, analyze results, and prioritize remediation efforts based on the severity and potential impact of vulnerabilities.

Malware Defenses (Control 8): AI-based solutions can enhance malware detection by learning and adapting to new threats in real-time. Behavioral analysis, machine learning algorithms, and anomaly detection are commonly used in this context.

Data Protection (Control 10): AI can assist in identifying and classifying sensitive data, helping organizations implement effective data protection measures and ensure compliance with privacy regulations.

Incident Response and Management (Control 16): AI technologies, such as machine learning models, can be employed to automate the early stages of incident detection and response, helping security teams detect and mitigate threats more rapidly.

Penetration Tests and Red Team Exercises (Control 17): AI may be used to simulate more realistic and adaptive attack scenarios during red teaming exercises, providing a more comprehensive assessment of an organization's security posture.

It's essential for organizations to stay informed about emerging technologies and best practices in cybersecurity, including the integration of AI. As of now, the SANS 20 framework provides a set of controls, and organizations can choose to implement AI-based solutions as part of their broader cybersecurity strategy to enhance threat identification and response capabilities. Always check for the latest updates and recommendations from the SANS Institute or other relevant authorities in the field of cybersecurity.

AI in Threat Mitigation:

Artificial Intelligence (AI) plays a significant role in threat mitigation within the field of cybersecurity. Here are several ways AI is employed to identify, analyze, and mitigate threats effectively:

Threat Detection:

Behavioral Analysis: AI algorithms can analyze patterns of behavior in network traffic, system logs, and user activities. This helps in detecting anomalies that may indicate a potential security threat.

Machine Learning Models: AI-driven machine learning models can be trained to recognize patterns associated with known and unknown threats, enabling early detection.

Malware Detection and Prevention:

Signature-based Detection: Traditional antivirus solutions use signature-based detection, but AI enhances this by learning and adapting to new malware variants.

Behavioral Analysis: AI analyzes the behavior of files and applications to identify suspicious activities indicative of malware.

Anomaly Detection:

User and Entity Behavior Analytics (UEBA): AI tools monitor and analyze user behavior, identifying deviations from normal patterns that may indicate compromised accounts or insider threats.

Network Security:

Intrusion Detection and Prevention Systems (IDPS): AI enhances IDPS by continuously learning about normal network behavior and identifying deviations or malicious activities.

Traffic Analysis: AI can analyze network traffic in real-time, identifying unusual patterns or suspicious activities that may signal an ongoing attack.

Phishing and Social Engineering Protection:

Email Filtering: AI-powered email security solutions use machine learning to identify and block phishing emails by recognizing patterns and characteristics associated with phishing attempts.

User Awareness Training: AI can support the development of personalized and targeted security awareness training programs based on user behavior and susceptibility.

Endpoint Security:

Behavioral Analysis: AI on endpoints monitors system activities and user behaviors, identifying anomalies or malicious actions in real-time.

Endpoint Protection Platforms (EPP): AI-enhanced EPP solutions provide advanced threat prevention, detection, and response capabilities.

Incident Response and Automation:

Threat Intelligence Integration: AI automates the integration of threat intelligence feeds, allowing security teams to respond quickly to emerging threats.

Automated Response: AI can automate certain response actions, such as isolating compromised systems or blocking malicious IP addresses, reducing the time to mitigate threats.

Predictive Analysis:

Predictive Threat Modeling: AI can analyze historical data to predict potential future threats, helping organizations proactively implement security measures.

Adaptive Security Measures:

Adaptive Access Controls: AI systems can dynamically adjust access controls based on real-time risk assessments, limiting the impact of compromised credentials.

Cloud Security:

Cloud Threat Intelligence: AI is used to analyze vast amounts of data in cloud environments, identifying and mitigating threats specific to cloud infrastructure.

While AI brings significant advantages to threat mitigation, it's crucial to note that it's not a silver bullet, and a holistic cybersecurity strategy involves a combination of technology, processes, and skilled personnel. Additionally, the landscape evolves, and threat actors may adapt their tactics, making it essential to continually update and improve AI-based security measures.

Examine AI Algorithms:

AI algorithms are at the core of artificial intelligence systems, driving their ability to learn, make predictions, and solve complex problems. Here's an examination of some common AI algorithms used in cybersecurity and threat mitigation:

Supervised Learning Algorithms:

Description: These algorithms learn from labeled training data, where the input data is paired with corresponding output labels. The model generalizes from this training data to make predictions on new, unseen data.

Applications in Cybersecurity:

Malware Detection: Supervised learning algorithms can be trained on labeled datasets of known malware and benign files to identify and classify new, unseen files.

Anomaly Detection: By using labeled data representing normal and anomalous behavior, supervised learning can help identify deviations that may indicate a security threat.

Unsupervised Learning Algorithms:

Description: Unsupervised learning algorithms operate on unlabeled data, seeking to find hidden patterns or structures without predefined output labels.

Applications in Cybersecurity:

Anomaly Detection: Unsupervised learning is often used to identify outliers or unusual patterns in network traffic, system logs, or user behavior.

Clustering: Grouping similar entities together, such as grouping similar malware samples or identifying patterns in attack campaigns.

Reinforcement Learning Algorithms:

Description: Reinforcement learning involves an agent that learns to make decisions by interacting with an environment. The agent receives feedback in the form of rewards or penalties based on its actions.

Applications in Cybersecurity:

Adaptive Security Measures: Reinforcement learning can be used to dynamically adjust security measures based on evolving threats and changing conditions.

Automated Response: In incident response, reinforcement learning can guide automated response actions based on the success or failure of previous actions.

Decision Trees:

Description: Decision trees are a popular method for classification tasks. They recursively split data based on features to make decisions.

Applications in Cybersecurity:

Threat Intelligence: Decision trees can be used to classify and categorize threat intelligence data, aiding in the prioritization of security alerts.

Random Forest:

Description: Random Forest is an ensemble learning technique that builds multiple decision trees and combines their outputs.

Applications in Cybersecurity:

Malware Detection: Random Forest can improve the accuracy of malware detection by combining the outputs of multiple decision trees.

Neural Networks (Deep Learning):

Description: Neural networks, especially deep learning models, consist of interconnected layers of artificial neurons that can learn complex patterns.

Applications in Cybersecurity:

Deep Packet Inspection: Deep learning models can analyze network traffic for patterns indicative of malicious activity.

Image Recognition: Deep learning is used in malware analysis, where it can identify patterns in binary files or network traffic.

Natural Language Processing (NLP):

Description: NLP focuses on the interaction between computers and human language. It includes tasks such as language translation, sentiment analysis, and text summarization.

Applications in Cybersecurity:

Phishing Detection: NLP techniques can be applied to analyze and understand the language used in emails to identify phishing attempts.

Genetic Algorithms:

Description: Inspired by the process of natural selection, genetic algorithms involve evolving a population of solutions over successive generations.

Applications in Cybersecurity:

Optimization: Genetic algorithms can be used to optimize security configurations, parameters in intrusion detection systems, or other cybersecurity-related settings.

Support Vector Machines (SVM):

Description: SVM is a supervised learning algorithm used for classification and regression tasks. It works by finding the hyperplane that best separates data into different classes.

Applications in Cybersecurity:

Intrusion Detection: SVM can be applied to identify patterns associated with malicious network activity.

Ensemble Methods:

Description: Ensemble methods combine multiple models to improve overall performance and robustness.

Applications in Cybersecurity:

Threat Detection: Combining the outputs of different algorithms, such as decision trees, neural networks, and SVMs, in an ensemble can enhance overall threat detection capabilities.

It's important to note that the effectiveness of an AI algorithm depends on factors such as the quality of training data, the algorithm's design, and its implementation. Additionally, the choice of algorithm often depends on the specific cybersecurity task or problem being addressed. Cybersecurity professionals need to carefully select and adapt these algorithms to their specific use cases to achieve the best results.

Study Integration Practices:

Integration practices in the context of cybersecurity typically refer to the seamless incorporation of various security tools, technologies, and processes within an organization's IT infrastructure. Effective integration is crucial for creating a unified and cohesive security posture. Below are key aspects and best practices related to the integration of cybersecurity solutions:

Comprehensive Security Framework:

Objective: Establish a comprehensive security framework that defines the organization's security policies, procedures, and objectives.

Integration: Align security solutions with the established framework to ensure a cohesive and standardized approach to cybersecurity.

Interoperability:

Objective: Ensure that different security tools and technologies can work together seamlessly.

Integration: Choose solutions that support common standards and protocols, allowing for easier integration with other security tools and systems.

Centralized Management and Visibility:

Objective: Achieve centralized control and visibility over security operations.

Integration: Integrate security tools into a centralized management platform or Security Information and Event Management (SIEM) system to monitor, analyze, and respond to security events in real-time.

Automation and Orchestration:

Objective: Streamline and automate routine security tasks to improve efficiency and response times.

Integration: Integrate security tools with automation and orchestration platforms to create automated workflows for incident response, threat detection, and remediation.

Threat Intelligence Integration:

Objective: Enhance threat detection and response by incorporating external threat intelligence feeds.

Integration: Integrate threat intelligence platforms with security tools to provide context and prioritize security incidents based on the latest threat information.

Cloud Security Integration:

Objective: Extend security measures to cloud environments and ensure consistent protection.

Integration: Integrate cloud security solutions with on-premises security tools to maintain visibility and control across hybrid environments.

User and Entity Behavior Analytics (UEBA):

Objective: Monitor and analyze user behavior for early detection of insider threats.

Integration: Integrate UEBA solutions with existing security infrastructure to correlate user behavior with other security events.

Incident Response Integration:

Objective: Improve incident response capabilities to minimize the impact of security incidents.

Integration: Integrate incident response tools with other security solutions for coordinated and swift response actions.

Continuous Monitoring and Auditing:

Objective: Implement continuous monitoring and auditing to identify vulnerabilities and security gaps.

Integration: Integrate monitoring tools with vulnerability assessment solutions and auditing processes for real-time insights into the security posture.

Security Awareness and Training:

Objective: Foster a security-aware culture among employees.

Integration: Integrate security awareness and training programs with other security initiatives to ensure that employees are well-informed and educated on security best practices.

Patch Management Integration:

Objective: Keep systems up to date with the latest security patches.

Integration: Integrate patch management tools with vulnerability scanners and asset management systems to automate the identification and application of security patches.

Regulatory Compliance:

Objective: Ensure compliance with relevant cybersecurity regulations and standards.

Integration: Align security practices with regulatory requirements and integrate compliance management tools to streamline audit processes.

Regular Testing and Simulation:

Objective: Assess the effectiveness of security controls through regular testing and simulations.

Integration: Integrate penetration testing, red teaming, and simulation tools with the security infrastructure to identify and remediate vulnerabilities proactively.

Vendor Collaboration:

Objective: Collaborate with security solution vendors for support and updates.

Integration: Establish relationships with vendors that encourage collaboration and ensure timely updates and support for integrated solutions.

Documentation and Communication:

Objective: Document integration processes and maintain clear communication channels.

Integration: Document integration configurations, workflows, and dependencies to facilitate ongoing maintenance and troubleshooting.

Effective integration practices help organizations build a resilient cybersecurity infrastructure, enabling them to detect, respond to, and mitigate security threats more efficiently. Regularly reviewing and updating integration strategies ensures that security measures remain robust in the face of evolving cyber threats.

**Practical Application And Simulations**

Practical application and simulations play a crucial role in cybersecurity training and readiness. They provide hands-on experience, allowing individuals and teams to apply theoretical knowledge in a controlled environment. Here are some ways practical application and simulations can be applied within the context of the SANS 20 framework:

Vulnerability Assessment Simulations:

Objective: To simulate the continuous vulnerability assessment and remediation process (Control 4).

Application: Use simulated environments to conduct vulnerability assessments, prioritize findings, and practice applying patches and fixes.

Malware Analysis and Defense Simulations:

Objective: To enhance malware defenses (Control 8).

Application: Create scenarios where participants analyze malware samples, understand their behavior, and implement defensive measures using AI-based tools or traditional antivirus solutions.

Incident Response Simulations:

Objective: To improve incident response and management (Control 16).

Application: Conduct simulated cyber-attacks or incidents, allowing participants to practice detection, containment, eradication, and recovery processes. This can include using AI-based tools for early incident detection.

Penetration Testing and Red Team Simulations:

Objective: To strengthen penetration tests and red team exercises (Control 17).

Application: Simulate real-world attack scenarios using AI-driven techniques, such as machine learning for more sophisticated and adaptive attacks, providing a realistic testing environment.

Security Awareness Training Simulations:

Objective: To implement a security awareness and training program (Control 18).

Application: Utilize simulations and phishing exercises to test and improve the awareness of employees regarding social engineering attacks and cybersecurity best practices.

Data Protection Simulations:

Objective: To enforce data protection measures (Control 10).

Application: Simulate scenarios where participants identify and classify sensitive data, implement encryption techniques, and practice incident response for data breaches.

Secure Configuration Simulations:

Objective: To implement secure configurations for hardware and software (Controls 3 and 9).

Application: Simulate the deployment of secure configurations for various devices and systems, ensuring participants understand and apply best practices.

Cloud Security Simulations:

Objective: To secure configurations for enterprise cloud resources (Control 19).

Application: Simulate cloud-based attacks and security misconfigurations, allowing participants to practice securing cloud environments.

When designing simulations, it's essential to align them with the specific controls and objectives outlined in the SANS 20 framework. This hands-on approach not only reinforces theoretical knowledge but also builds practical skills and helps individuals and teams become more adept at handling real-world cybersecurity challenges.

SANS 20 Framework Overview

SANS stands for **S**ysAdmin, **A**udit, **N**etwork, and **S**ecurity.

# SANS Category:- CWE-119: Memory Buffer Error Description:-

This buffer overflow happens when an application process tries to store more data than it can hold in the memory. The data flows to another memory location which can corrupt the data already contained in that buffer.This could be disastrous, as this can erase data, steal confidential information, and even the whole application could crash because of this buffer overflow.

# Business Impact:-

Memory buffer errors can cause applications to crash lead to system downtime. Exploitation of memory buffer errors disrupts the business operations and impact the continuity of services. If the affected system provides services to customers, the exploitation of memory buffer errors can lead to service outages, inconveniencing and frustrating users.

# SANS Category:- CWE-79: Cross-site Scripting Description:-

Cross-site Scripting (XSS) is an injection attack that usually happens when a malicious actor or an attacker injects malicious or harmful script into a web application which can be executed through the web browsers. Once the malicious script finds its way into the compromised system, it can be used to perform different malicious activities.

# Business Impact:-

Cross-Site Scripting (XSS) attacks can harm a business by stealing user data, damaging its reputation, causing financial losses, and leading to legal consequences. Fixing these vulnerabilities involves implementing secure coding practices to protect websites and applications from malicious script injections.

# SANS Category:- CWE-20: Unvalidated Input Error Description:-

The application receives input, but fails to validate the input, whether it has all necessary details needed for it to be accepted into the system for processing.When there is input sanitization, this can be used to check any potentially dangerous inputs in order to ensure that the inputs are safe to be processed with the source code or when it’s an input that is needed to communicate with other components.

# Business Impact:-

Unvalidated Input Errors (CWE-20) pose a risk of security breaches, financial losses, and reputation damage by allowing malicious data input, necessitating robust input validation measures to safeguard against unauthorized access and maintain data integrity.

# SANS Category:- CWE-200: Sensitive Information Exposure Error Description:-

This happens when the application knowingly and unknowingly exposes information that is confidential and sensitive to an attacker who does not have the authorization to access these information.Different errors lead to this information being exposed to an attacker.

# Business Impact:-

Sensitive Information Exposure (CWE-200) can result in severe business impact, including compromised data confidentiality, loss of customer trust, and potential legal consequences, necessitating robust security measures to protect sensitive information and maintain regulatory compliance.

# SANS Category:- CWE-125: Out-of-bounds Read Error Description:-

This usually occurs when the application reads data past the normal level, either to the end or before the beginning of the buffer. This gives unprivileged access to an attacker to read sensitive information from other memory locations, which can as well leads to a system or application crash.A crash will certainly happen when the code reads data and thinks there is an indicator in place that stops the read operation like a NULL that is applied to a string

# Business Impact:-

Out-of-bounds Read Error (CWE-125) can lead to security vulnerabilities, system crashes, and unauthorized access, posing a risk of data breaches and significant business disruption, emphasizing the need for thorough code reviews and preventive measures to ensure secure software development.

# SANS Category:- CWE-89: SQL Injection Description:-

[SQL injection](https://www.softwaretestinghelp.com/sql-injection-how-to-test-application-for-sql-injection-attacks/) is a form of security vulnerability whereby the attacker injects a Structured Query Language (SQL) code to the Webform input box in order to gain access to resources or change data that is not authorized to access.This vulnerability can be introduced to the application during the design, implementation, and operation stages.What this SQL query does is to make an unauthorized request to the database for some information.

# Business Impact:-

SQL Injection (CWE-89) can result in unauthorized access, data breaches, and manipulation of databases, causing severe business impact such as compromised data integrity, reputational damage, and potential legal consequences, necessitating strict input validation and parameterized queries to mitigate risks.

# SANS Category:- CWE-416: Free Memory Error Description:-

This issue is caused by the referencing of memory after it has been released, which can seriously lead to a program crash. When you use a previously freed memory, this can have adverse consequences, like corrupting of valid data, arbitrary code execution which is dependent on the flaw timing.

# Business Impact:-

Free Memory Error (CWE-416) can lead to application crashes, data corruption, and potential security vulnerabilities, posing a risk of system instability, service disruption, and exploitation by attackers, emphasizing the importance of proper memory management practices for business continuity and security.

# SANS Category:- CWE-190: Integer Overflow Error Description:-

When a calculation is processed by an application and there is a logical assumption that the resulting value will be greater than the exact value, integer overflow happens. Here, an integer value increases to a value that cannot be stored in a location.

# Business Impact:-

Integer Overflow Error (CWE-190) can result in unexpected behavior, crashes, or security vulnerabilities, posing a risk of system instability, data corruption, and potential exploitation, highlighting the need for secure coding practices to prevent business disruptions and safeguard against malicious activities.

# SANS Category:- CWE-352: Cross-Site Request Forgery Description:-

This is when a web application does not sufficiently verify the HTTP request, whether the request was actually coming from the right user or not. The webservers are designed to accept all requests and to give a response to them.

# Business Impact:-

Cross-Site Request Forgery (CWE-352) can lead to unauthorized actions on behalf of users, compromising data integrity, user accounts, and potentially causing financial losses, emphasizing the importance of anti-CSRF tokens and secure web application design to mitigate business risks.

# SANS Category:- CWE-22: Directory Traversal Description:-

Directory traversal or file path traversal is a web security vulnerability that allows an attacker to read arbitrary files on the server that is currently running an application.

# Business Impact:-

Directory Traversal (CWE-22) can result in unauthorized access to sensitive files, compromising data confidentiality, and potentially leading to data breaches, emphasizing the need for input validation and secure file access controls to prevent business-critical information exposure.

# SANS Category:- CWE-78: OS Command Injection Description:-

It is about the improper sanitization of special elements that may lead to the modification of the intended OS command that is sent to a downstream component. An attacker can execute these malicious commands on a target operating system and can access an environment to which they were not supposed to read or modify.

# Business Impact:-

OS Command Injection (CWE-78) can lead to unauthorized execution of arbitrary commands, compromising system integrity and potentially causing data breaches or service disruptions, highlighting the critical need for input validation and secure command execution practices to mitigate business risks.

# SANS Category:- CWE-787: Out-of-bounds Write Error Description:-

This happens when the application writes data past the end, or before the beginning of the designated buffer.

# Business Impact:-

Out-of-bounds Write Error (CWE-787) can result in data corruption, system crashes, and potential security vulnerabilities, posing a risk of unauthorized access and service disruption, underscoring the importance of robust bounds checking to ensure business continuity and prevent malicious exploitation.

# SANS Category:- CWE-287: Improper Authentication Error Description:-

This is when an attacker claims to have a valid identity but the software failed to verify or proves that the claim is correct.A software validates a user’s login information wrongly and as a result, an attacker could gain certain privileges within the application or disclose sensitive information that allows them to access sensitive data and execute arbitrary code.

# Business Impact:-

Improper Authentication (CWE-287) can lead to unauthorized access, compromising sensitive data and system integrity, posing a risk of data breaches, reputational damage, and potential legal consequences, emphasizing the need for robust authentication measures to safeguard business assets.

# SANS Category:- CWE-476: Dereferencing NULL Pointer Description:-

Dereferencing a null pointer is when the application dereferences a pointer that was supposed to return a valid result instead returns NULL and this leads to a crash. Dereferencing a null pointer can happen through many flaws like race conditions and some programming error.

# Business Impact:-

Dereferencing NULL Pointer (CWE-476) can result in application crashes, system instability, and potential security vulnerabilities, posing a risk of service disruption, data corruption, and unauthorized access, emphasizing the importance of rigorous error checking to ensure business continuity and prevent exploitation.

# SANS Category:- CWE-732: Incorrect Permission Assignment Description:-

This vulnerability happens when an application assigns permissions to a very important and critical resource in such a manner that exposed the resource to be accessed by a malicious user.

# Business Impact:-

Incorrect Permission Assignment (CWE-732) can lead to unauthorized access, data breaches, and potential compromise of sensitive information, posing a risk of reputational damage, legal consequences, and business disruption, highlighting the need for proper permission controls to ensure data security.

# SANS Category:- CWE-434: Unrestricted File Upload Description:-

This vulnerability occurs when the application does not validate the file types before uploading files to the application. This vulnerability is language independent but usually occurs in applications written in ASP and PHP language.

# Business Impact:-

Unrestricted File Upload (CWE-434) can result in malicious file execution, compromising system integrity and potentially leading to data breaches, reputational damage, and service disruption, emphasizing the importance of secure file upload controls to mitigate business risks.

# SANS Category:- CWE-611: Information Exposure through XML Entities Description:-

When an XML document is uploaded into an application for processing and this document contains XML entities with uniform resource identifier that resolves to another document in another location different from the intended location. This anomaly can make the application to attach incorrect documents into its output.

# Business Impact:-

Information Exposure through XML Entities (CWE-611) can lead to unauthorized access to sensitive data, posing a risk of data breaches, reputational damage, and potential legal consequences, emphasizing the need for secure XML processing and input validation to safeguard business-critical information.

# SANS Category:- CWE-94: Code Injection Description:-

The existence of code syntax in the user’s data increases the attacker’s possibility to change the planned control behavior and execute arbitrary code. This vulnerability is referred to as “injection weaknesses” and this weakness could make a data control become user-controlled.

# Business Impact:-

Code Injection (CWE-94) can result in the execution of arbitrary code, leading to unauthorized access, data breaches, and potential system compromise, posing a risk of reputational damage, financial losses, and legal consequences, highlighting the critical need for secure coding practices to mitigate business risks.

# SANS Category:- CWE-798: Hard-coded Access Key Description:-

This is when the password and access key is hard coded into the application directly for inbound authentication purpose and outbound communication to some external components and for encryption of internal data. Hard-coded login details usually cause vulnerability that paves the way for an attacker to bypass the authentication that has been configured by the software administrator.

# Business Impact:-

Hard-coded Access Key (CWE-798) can lead to unauthorized access, compromise of sensitive information, and potential security breaches, posing a risk of reputational damage, legal consequences, and business disruption, emphasizing the importance of secure key management practices to protect business assets.

# SANS Category:- CWE-400: Uncontrolled Resource Consumption Description:-

This vulnerability happens when the application does not control the allocation properly and maintenance of a limited resource, this allows an attacker to be able to influence the amount of resources consumed, which will eventually lead to the exhaustion of available resources.Part of the limited resources includes memory, file system storage, database connection pool entries, and CPU.

# Business Impact:-

Uncontrolled Resource Consumption (CWE-400) can result in system performance degradation, service disruptions, and potential denial-of-service attacks, posing a risk of business downtime, customer dissatisfaction, and financial losses, emphasizing the need for resource usage controls to maintain operational stability.