Daffodil International University

Course Presentation

Only for course Teacher					
	Needs Improvement	Fair	Good	Excellent	Total Mark
Delivery					
Content/Organization					
Enthusiasm/Audience Awareness					
Comments					

Semester: Spring .... / Fall 2023

Student Name: K.M. Al Jaziz Turja
Student ID: 211-35-3164
Batch: 34 Section: A
Course Code: SE 332 Course Name: Information System Security
Course Teacher Name: Md. Maruf Hassan
Designation: Associate Professor
Submission Date: 01 / 12 / 2023

Assessment criteria or rubrics for Presentation					
Criteria	1—Needs Improvement	2—Fair	3—Good	4—Excellent	Score
Delivery	•Holds no eye contact with audience, as entire report is read from notes • Speaks in low volume and/or monotonous tone, which causes audience to disengage	• Displays minimal eye contact with audience, while reading mostly from the notes • Speaks in uneven volume with little or no inflection	Consistent use of direct eye contact with audience, but still returns to notes • Speaks with satisfactory variation of volume and inflection	Holds attention of entire audience with the use of direct eye contact, seldom looking at notes • Speaks with fluctuation in volume and inflection to maintain audience interest and emphasize key points	3
Content/Organization	• Does not have grasp of information and cannot answer questions about subject • Does not clearly define subject and purpose; provides weak or no support of subject; gives insufficient support for ideas or conclusions	•Is uncomfortable with information and is able to answer only rudimentary questions • Attempts to define purpose and subject; provides weak examples, facts, and/or statistics, which do not adequately support the subject; includes very thin data or evidence	•Is at ease with e xpected answers to all questions, without elaboration • Has somewhat clear purpose and subject; some examples, facts, and/or statistics that support the subject; includes some data or evidence that supports conclusions	Demonstrates full knowledge by answering all class questions with explanations and elaboration • Provides clear purpose and subject; pertinent examples, facts, and/or statistics; supports conclusions/ideas with evidence	3
Enthusiasm/Audience Awareness	•Shows no interest in topic presented •Fails to increase audience understanding of knowledge of topic	• Shows little or mixed feelings about the topic being presented • Raises audience understanding and knowledge of some points	• Shows some enthusiastic feelings about topic • Raises audience understanding and awareness of most points	•Demonstrates strong enthusiasm about topic during entire presentation • Significantly increases audience understanding and knowledge of topic; • convinces an audience to recognize the validity and importance of the subject	2
				Net Total out of 8	8

Youtube Link

https://www.youtube.com/watch?v=YFfXVwt2wyk

CODE VULNERABILITY & COMPROMISE SOFTWARE QUALITY

A MAJOR SECURITY THREATS

K.M. Al Jaziz Turja

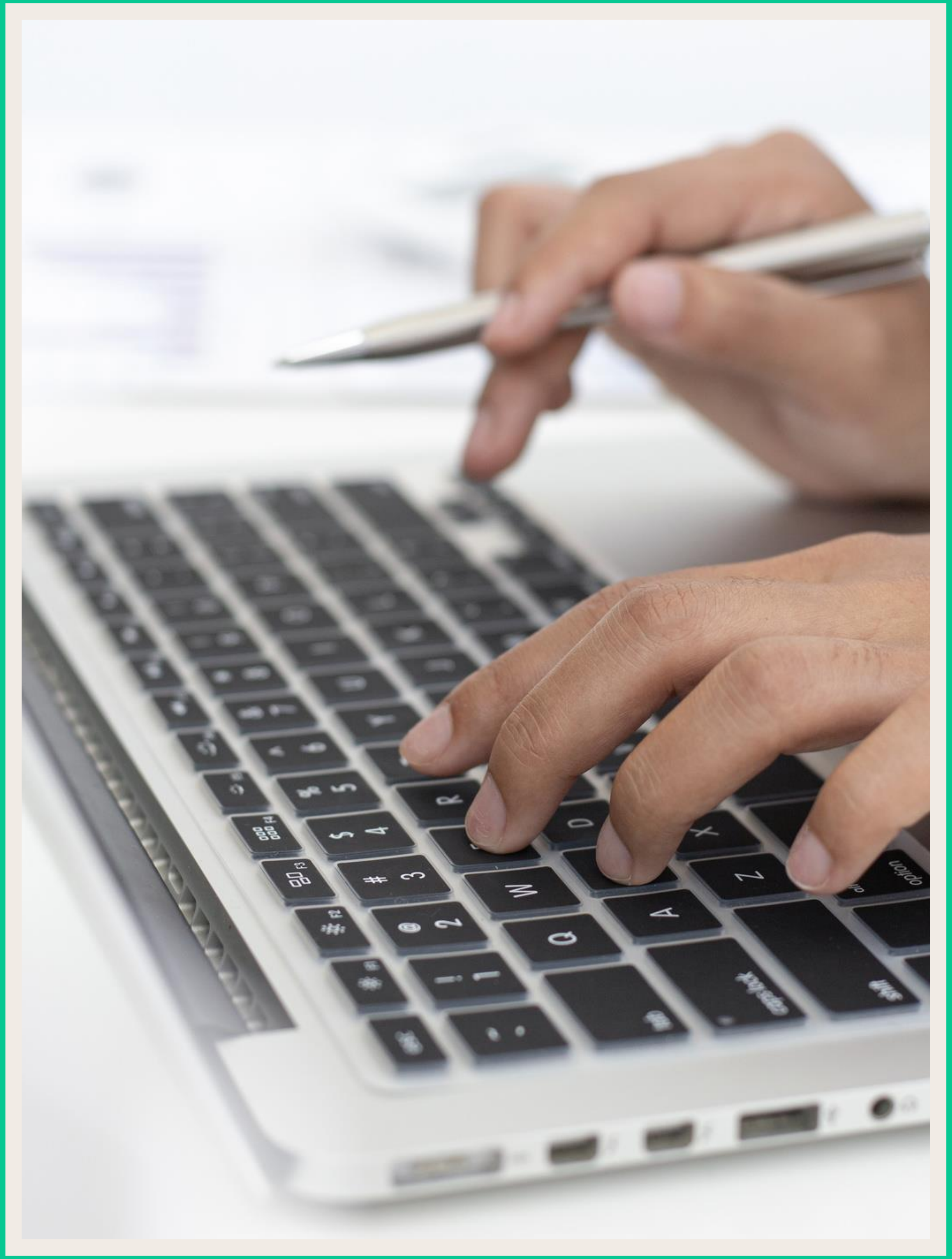
ID: 211-35-3164

Department of Software Engineering

Daffodil International University

AGENDA

- Code Quality
- Code Vulnerabilities
- Code Quality for Software Security
- Code Analysis
- Type of Code Analysis
- Static Code Analysis
- Security Issues Through Code Analysis
- Buffer Overflow



CODE QUALITY

Code quality refers to the effectiveness, readability, and maintainability of a software program, ensuring it meets functional requirements while minimizing bugs and technical debt. It encompasses best practices, standards adherence, and efficient problem-solving.

CODE VULNERABILITIES

CODE VULNERABILITY REFERS TO WEAKNESSES OR FLAWS IN A SOFTWARE APPLICATION'S CODE THAT CAN BE EXPLOITED BY ATTACKERS, POTENTIALLY LEADING TO SECURITY BREACHES OR UNAUTHORIZED ACCESS. IDENTIFYING AND ADDRESSING VULNERABILITIES IS CRUCIAL FOR MAINTAINING THE SECURITY OF A SYSTEM

TYPES OF CODE VULNEABILITIES

- Input Validation Issues
- Buffer Overflows
- Insecure Dependencies
- Inadequate Authentication and Authorization
- Security Misconfigurations
- Security Headers Missing

CODE QUALITY FOR SW SECURITY

SW Security

Software security is an idea/set of practices implemented to protect software against malicious attack and other hacker risks, so that the software continues to function correctly under such potential risks.

Poor Quality Code is a Security Threat

Developers always need to comply the code metrics standard in their code because, it's not only to deliver the functionalities of the software but also to ensure their code is bug free and the final product is not vulnerable to any security attack. Any compromise in code quality, may eventually leads towards security breach.

How to Mitigate Risk

Perform code analysis to early code quality metrices violations and take necessary preventive actions before 'Go Live'

CODE ANALYSIS

- CODE ANALYSIS IS THE PROCESS OF REVIEWING AND EVALUATING SOURCE CODE TO ENSURE COMPLIANCE WITH CODING STANDARDS, IDENTIFY POTENTIAL BUGS, AND IMPROVE OVERALL SOFTWARE QUALITY.

- Code analysis is essential to identify and rectify coding issues, enhance maintainability, and ensure adherence to best practices, promoting overall software quality and reliability.

Reasons for Code Analysis

- Identifying Security Vulnerabilities
- Maintaining Code Consistency
- Early Detection of Bugs
- Continuous Integration and Continuous Delivery (CI/CD) Integration
- Compliance with Regulations
- Quality Assurance
- Preventing Code Smells
- Optimizing Performance
- Enhancing Code Readability
- Streamlining Code Reviews

TYPE OF CODE ANALYSIS

- Static Code Analysis
- Dynamic Code Analysis
- Code Architecture Analysis

STATIC CODE ANALYSIS

Static code analysis is a method of examining source code for potential errors, security vulnerabilities, and adherence to coding standards without executing the program, providing insights into code quality and potential issues early in the development process. It aids in identifying and mitigating issues before runtime, contributing to more robust and secure software.

Error Categories of Static Code Analysis

- Blocker
- Critical
- Major
- Minor
- Information

Different Tools for Static Code Analysis

- Coverity
- SonarQube
- FindBugs
- GetaFix
- Hp Fortify
- Infer. etc.

SECURITY ISSUES THROUGH CODE ANALYSIS

TOP WEB APPLICATION SECURITY FLAWS OR SECURITY ISSUES THAT CAN BE FOUND IN A VULNERABLE CODE

-Unvalidated Input

- Broken Access Control

- Broken Authentication and Session Management

- Cross-Site Scripting (XSS) Flaws

- Buffer Overflows

- Injection Flaws

- Improper Error Handling

- Insecure Data Handling

- Insecure Storage

- Denial of Service

- Insecure Configuration Management

BUFFER OVERFLOW

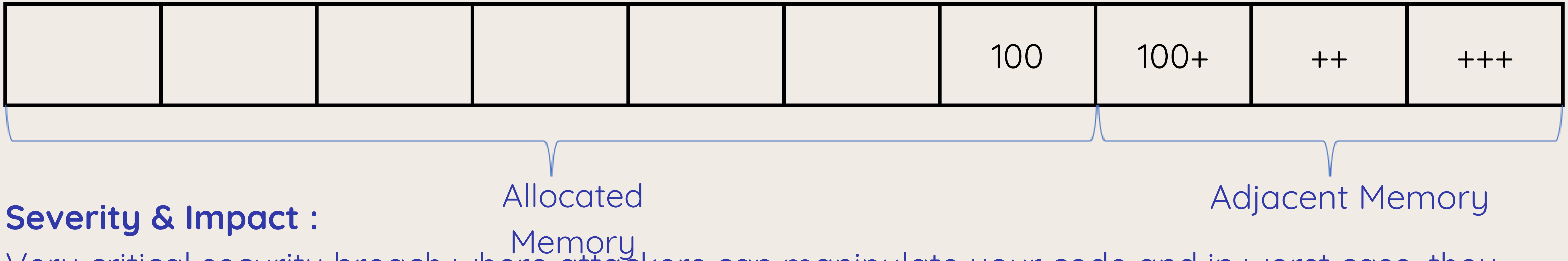
Buffer Overflow:

A buffer overflow occurs when an operation that writes data to an allocated buffer exceeds the buffer boundaries. Thus, the operation accesses adjacent memory locations that it should not.

Example:

```
#define  MAXSIZE 100
…
…
Char localBuf [MAXSIZE]
…
…
Gets (localBuf)
```

The get() function does not allow you to check if the data read to localBuf has less than MAXSIZE characters. You rely on the sender data to stay within the limit. A malicious user can easily overflow the buffer by sending data greater than MAXSIZE characters and access adjacent regions in the stack.



						100	100+	++	+++

Allocated Memory

Adjacent Memory

Severity & Impact :

Very critical security breach where attackers can manipulate your code and in worst case, they may get the full control on your system.

THANK YOU VERY MUCH!