A blue and orange logo

Description automatically generated

**IE2062 – Web Security**

**Year 2, Semester 2**

**Journal**

**IT21831904 – K.M. Weerasinghe**

Table of Contents

[Entry 1 3](#_Toc149758626)

[Entry 2 5](#_Toc149758627)

[Entry 3 6](#_Toc149758628)

[Entry 4 7](#_Toc149758629)

[Entry 5 8](#_Toc149758630)

[Entry 6 9](#_Toc149758631)

[Entry 7 10](#_Toc149758632)

[Entry 8 11](#_Toc149758633)

[Entry 9 14](#_Toc149758634)

[Entry 10 16](#_Toc149758635)

[Entry 11 17](#_Toc149758636)

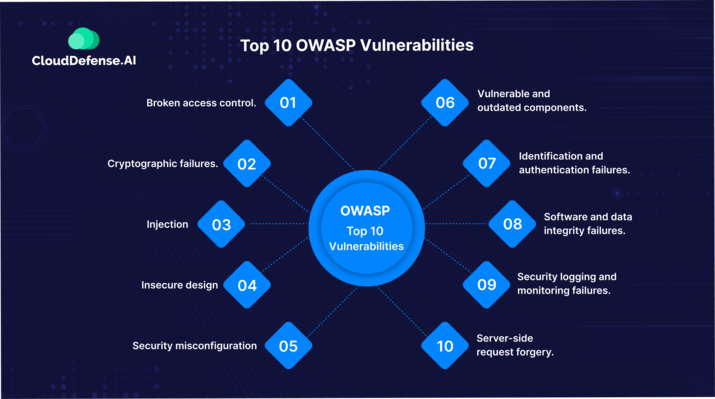
[Entry 12 18](#_Toc149758637)

[Entry 13 19](#_Toc149758638)

[Entry 14 20](#_Toc149758639)

# Entry 1

* **Date** - 2023/10/10
* **Summary of the day's activities** - getting a brief idea about the OWASP top 10 vulnerabilities
* **Reflections and takeaways**
* The list of the top 10 most significant threats to web application security is known as the OWASP Top 10. Successful bug bounty hunting requires an understanding of these risks and taking the necessary steps to prevent them. This will help us find vulnerabilities and report them in a timely manner, making web applications safer.
* As of 2023 the OWASP has come up with their list of top 10 vulnerabilities as mentioned below.
  + 1. Broken access control.
       - The attackers can view sensitive data or carry out tasks they aren't supposed to by bypassing authorization as a result of broken access control.
    2. Cryptographic failures.
       - A cryptographic failure is any vulnerability that results from using algorithms designed to protect sensitive data improperly. For web applications to guarantee user confidentiality on multiple levels, cryptography is necessary.
    3. Injection.
       - Injection attacks happen due to the application's understanding of user-controlled input as parameters or commands. Depending on the technologies being used and how they understand the input, injection attacks are possible.
    4. Insecure design.
       - These vulnerabilities happen when an inappropriate threat model is created in the application's planning stages and spread to the finished product. Developers may also introduce vulnerabilities related to insecure design at other points in time.
    5. Security misconfiguration.
       - Security misconfigurations arise from situations where proper security configuration could have been achieved but were not. Even with the most recent software download, your installation may still be vulnerable due to incorrect configurations.
    6. Vulnerable and outdated components.
       - Using out-of-date or weak components reveals an application to commonly detected security flaws, such as libraries, frameworks, or software dependencies. Attackers frequently focus on well-known flaws in these components.
    7. Identification and authentication failures.
       - Modern web applications must have both session management and authentication. Verifying a user's identity through authentication, usually done with a username and password, is the method used. Since HTTP(S) communication is stateless, the server sends a session cookie to the user's browser after verification. The server can monitor the user's activities thanks to this cookie. Sensitive information could be made public if an attacker finds weaknesses in the authentication system, which could allow them to access other users' accounts without permission.
    8. Software and data integrity failures.
       - Unauthorized modification or manipulation of software or data is a failure of software and data integrity. This may lead to inaccurate information or unauthorized software modifications, compromising data integrity.
    9. Security logging and monitoring failures.
       - Every action a user takes while configuring a web application should be logged. Because the actions of the attackers can be traced in the circumstances of an incident, logging is crucial. Their impact and risk can be determined once their actions are tracked. If an attacker gained access to a specific web application, there would be no way to determine what actions they used without logging.
    10. Server-side request forgery.
        - When an attacker has control over the contents of a request and can force a web application to send requests on their behalf to random destinations, is known as server-side request forgery. Implementations in which our web-based application requires to access external services frequently result in SSRF vulnerabilities.
* Links [Top 10 OWASP Vulnerabilities in 2023 (clouddefense.ai)](https://www.clouddefense.ai/top-10-owasp-vulnerabilities/)



# Entry 2

* Date – 2023/10/11
* Summary of the day's activities - searching, using, and exploring the reconnaissance tools
* Challenges faced and how they were overcome. –
* Recommended Tools Not Working Properly:
  + Issue: Some recommended tools were malfunctioning.
  + Solution: Changed to alternate tools with better functionality.
* Lack of Understanding About Installing Tools Using Git and Other Methods:
  + Issue: I had difficulty installing tools using git and other methods.
  + Solution: Relearned installation methods through YouTube tutorials and Google guides.
* Some Tools Not Outputting Results:
  + Issue: Certain tools were not providing the expected results.
    - Ex -sublist3r tool wouldn’t output any results
  + Solution: Switched to alternative tools that delivered the desired output.
* Tools Taking Too Much Time to Output:
  + Issue: Tools were consuming excessive time for scanning.
  + Solution: Utilized command options to expedite scans and found faster alternatives like Subfinder for improved efficiency.
* New tools, techniques, or concepts learned. -
* learned basic commands and outputting results doing basic enumeration using the Sublist3r, Amass, Subfinder, knock and domained tools.
* Learned about passive reconnaissance, brute-forcing subdomains, DNS enumeration, passive vs aggressive enumeration.
* domained - [Domained - Multi Tool Subdomain Enumeration Suite on Kali Linux - GeeksforGeeks](https://www.geeksforgeeks.org/domained-multi-tool-subdomain-enumeration-suite-on-kali-linux/)
* amass - [How to Use OWASP Amass: An Extensive Tutorial - Dionach](https://www.dionach.com/blog/how-to-use-owasp-amass-an-extensive-tutorial/)
* Reflections and takeaways
* Focusing on bug bounty hunting requires flexibility and the ability to learn how to use new tools.
* The secret to improving at tasks is constant learning, and we need to be able to learn by ourselves by using the internet and other resources. We need to be able to utilize these resources to our benefit.
* *domained* is an ***all-in-one tool*** that has all the other combined to it for subdomain enumeration, making it the best tool for recon and subdomain enumeration that I have personally used and *subfinder* is the **most efficient tool.**

# Entry 3

* Date – 2023/10/13
* Summary of the day's activities –
  + exploring domained.py for reconning
* Vulnerabilities discovered or explored.
  + By compiling subdomains, I was able to successfully find vulnerabilities pertaining to the target domain. I did this by gathering a long list of subdomains connected to the target domain using the "amass" and "subfinder" tools. Especially in the context of subdomain hijacking, this data was crucial for further testing.
* Challenges faced and how they were overcome.
  + Output of the amass subdomain files were not usable so I had to choose the sub finders output file for the further testing like subdomain hijacking
* New tools, techniques, or concepts learned.
  + This is the command used to start a subdomain scanning using the domained.py using only the amass and subfinder tools only
    - python3 domained.py -d domain.com --quick –notify
  + to do a full scan using all the tools to find subdomains using different tools we can use the below command
    - python3 domained.py -d domain.com
* Reflections and takeaways
  + My experience with "domained.py" proved helpful since it turned out to be a flexible utility that integrates the features of several reconnaissance tools into a single useful interface. This efficiency made it possible for the reconnaissance process to go more easily and extensively, which is essential for finding potential weaknesses and successfully securing the target domain.

# Entry 4

* Date - 2023/10/16
* Summary of the day's activities
  + Gathering information about the target using websites that shows the whois records and security header checking websites and other websites.
* Vulnerabilities discovered or explored.
  + Missing header like
    - Strict-Transport-Security
    - Content-Security-Policy
    - Referrer-Policy
    - Permissions-Policy
* New tools, techniques, or concepts learned.
  + Whois information using domain tool’s <https://whois.domaintools.com>
  + Securityheader.com was used to find missing headers information
* Reflections and takeaways
  + Information gathering and security analysis were the main goals of the day's activities, with a particular emphasis on finding any missing security headers in the target's web configuration. A more comprehensive understanding of web security and reconnaissance techniques was gained through this experience.

# Entry 5

* Date 2021/10/12
* Summary of the day's activities –
  + reconnaissance activities focused on subdomain takeovers. The primary goal of the day was to discover and explore subdomains and assess whether they were vulnerable to takeover.
* Vulnerabilities discovered or explored –
  + Gathering a list of subdomains associated with the target organization. To evaluate their vulnerability, I employed the tool "Subjack," a Go-based tool designed specifically for subdomain takeover detection. Subjack excels in mass-testing scenarios due to its efficiency and speed. It scans a list of subdomains to identify those that might be hijacked. After running Subjack, I manually reviewed the results to rule out false positives.
* Challenges faced and how they were overcome.
  + Some tools are taking too much time to recon
* New tools, techniques, or concepts learned.
  + Subjack – Subjack is a Go tool for subdomain takeover that finds subdomains that can be hijacked by simultaneously scanning a list of subdomains. When it comes to mass-testing, subjack truly shines thanks to its efficiency and speed. To rule out false positives, always manually double-check the results.
    - * + **Subjack -w domains.txt -v**
    - **-w domains.txt**: This flag specifies the input file containing the list of subdomains to be tested for vulnerability.
    - **-v**: This flag sets the tool to verbose mode, which provides more detailed information for each request
    - Its also possible to add number of threads -t , timeout settings and other options.
* Reflections and takeaways
  + The activities of this day gave me important insights into the identification of vulnerabilities in a target network and the process of detecting subdomain takeovers. One particularly helpful monitoring method was making use of Subjack. It reminded the importance of automated and manual verification procedures in order to guarantee vulnerability assessment correctness. Overall, this experience assisted me enhance my penetration testing and cybersecurity capabilities even further.

# Entry 6

* Date – 2023/10/17
* Summary of the day's activities
  + was focused on using Nmap and other reconnaissance tools to carry out security assessments and reconnaissance tasks. My main goal was to automate the process of scanning target systems for open ports, sitemaps, and robots.txt files.
* Vulnerabilities discovered or explored.
  + I did not encounter any vulnerabilities over the course of the day's activities. Instead, I used Nmap to look for open ports and learn important details about the targeted systems.
* Challenges faced and how they were overcome.
  + A full scan like full TCP scan or some scan would take a lot of time so it is better to tune the nmap according to our target scopes and out of scope details to get the best results
* New tools, techniques, or concepts learned.
  + Nmap is a powerful open-source tool for network discovery and security assessments It allows users to scan and map network hosts, identifying open ports, services, and potential vulnerabilities. The tool offers numerous scanning options and techniques to adapt to various scenarios.
    - Sudo Nmap Ip address – scan a single Ip address
    - Sudo Nmap domain.com – scans a domain name
    - Sudo Nmap -p- IP address – scans for all the ports
    - Sudo Nmap IP address -sV – performs a service version scan
* Reflections and takeaways
  + My knowledge of the importance of Nmap in bug bounty programs and evaluations of security has been improved by my experiences of this day. Nmap is a great tool for port scanning that has a lot of tuning possibilities to meet different goals and objectives. Every cybersecurity specialist should be able to use this versatile powerful tool while doing penetration tests and security assessments. Furthermore, one of the most important abilities in the field of cybersecurity is the capacity to optimize and fine-tune scans for efficiency.

# Entry 7

* Date – 2023/10/18
* Summary of the day's activities
  + Testing and finding new tools related to vulnerability checking
* Vulnerabilities discovered or explored.
  + Did not find any vulnerabilities today
* Challenges faced and how they were overcome.
  + There are many tools that we can find for vulnerability checking in the internet and its hard to test all of them. And some can not give the results that we expect or might not work so it is better to get recommendations of tools from trusted updated sites.
* New tools, techniques, or concepts learned.
  + Nikto - Nikto was one of the tools I looked at today. A popular web server scanner that helps in finding possible vulnerabilities.
    - **nikto -h domain.com** - This command scans a target website for potential vulnerabilities and provides a report on any issues found.
  + Rapidscan – rapidscan is a multitool web vulnerability tool used to automate the scanning process and report any vulnerabilities found. It used several other tools like Nmap Nikto Dmitry amass and other tools for the process.
    - Rapidscan.py domain.com -scans for the vulnerabilities of the website
* Reflections and takeaways
  + I am now aware how crucial it is to rely on reliable sources when seeking tool recommendations. Nikto turned out to be a useful addition to my web server scanning toolkit, and Rapidscan also demonstrated potential.

# Entry 8

* Date – 2023/10/20
* Summary of the day's activities
  + Vulnerability scanning for reports and finalizing reports with all gathered contents
* Vulnerabilities discovered or explored
  + These are the vulnerabilities I found today using the rapidscan for the today’s vulnerability scanned domains only the unique vulnerabilities are given here
  1. Webserver vulnerable to MS10-070
     + Vulnerability threat level
       - High
     + Vulnerability description
       - An attacker who successfully exploited this vulnerability could read data, such as the view state, which was encrypted by the server. This vulnerability can also be used for data tampering, which, if successfully exploited, could be used to decrypt and tamper with the data encrypted by the server
     + Impact assessment
       - The impact of this vulnerability is assessed as high, as it allows unauthorized access to encrypted data and potential data tampering.
     + Affected components
       - The webserver is the affected component in this vulnerability
     + Proposed mitigation or fix
       - Microsoft has released a set of patches on their website to mitigate this issue. The information required to fix this vulnerability can be inferred from this resource.
  2. Vulnerable to Slowloris denial of service
     + Vulnerability threat level
       - Critical
     + Vulnerability description
       - This attack works by opening multiple simultaneous connections to the web server and it keeps them alive as long as possible by continuously sending partial HTTP requests, which never get completed. They easily slip through IDS by sending partial requests.
     + Affected components
       - affects web servers, could include a wide range if web server software’s
     + Impact assessment
       - it can effectively deny legitimate users access to the targeted web server, disrupt services, and potentially lead to significant downtime.
     + How an attack could be carried out
       - Establish multiple simultaneous connections to the target web server.
       - Keep these connections open by sending partial HTTP requests that are never completed.
       - Continuously maintain these connections to exhaust server resources.
     + Proposed mitigation or fix –
       - Implementation of rate limiting to limit the number of concurrent connections from a single Ip address.
       - Implementing intrusion detection and prevention systems
       - Distribute traffic across multiple servers to reduce the impact of these type of attacks
  3. Vulnerabilities reported in SSL scans
     + vulnerability threat level
       - Medium
     + Vulnerability description
       - SSL related vulnerabilities break the confidentiality factor. An attacker may perform a MiTM attack, interpreting and eavesdrop the communication.
     + Affected components
       - SSL and TLS libraries and the communication channels using them
     + Proposed mitigation or fix
       - Proper implementation and upgraded version of SSL and TLS libraries to address and eliminate the reported vulnerabilities.
  4. Webserver is outdated
     + Vulnerability threat level
       - High
     + Vulnerability description
       - Any outdated web server may contain multiple vulnerabilities as their support would have been ended. An attacker may make use of such an opportunity to leverage attacks
     + Impact assessment
       - High, as the outdated web servers are more susceptible to attacks and compromise, potentially leading to data breaches, service disruptions, or unauthorized access.
     + Proposed mitigation or fix
       - Upgrade the web servers to latest version
  5. Webserver leaks internal IP
     + vulnerability threat level
       - low
     + vulnerability description
       - Gives attacker an idea on how the address scheming is done internally on the organizational network. Discovering the private addresses used within an organization can help attackers in carrying out network-layer attacks aiming to penetrate the organization's internal infrastructure.
     + Affected components
       - The vulnerability affects the network infrastructure and services that expose internal address information to the public.
     + impact assessment
       - The threat level for this vulnerability is low, as it primarily provides information that could be used in subsequent attacks rather than being an immediate security breach.
     + Proposed mitigation or fix
       - Restrict the banner information to the outside world from the disclosing service. More information on mitigating this vulnerability can be found here.
* New tools, techniques, or concepts learned.
  + Rapidscan use for full scan of the domain

# Entry 9

* Date 2023/10/23
* Summary of the day's activities
  + Vulnerability scanning for reports and finalizing reports with all gathered contents
* Vulnerabilities discovered or explored.

Only ***unique*** vulnerabilities found today

* 1. Secure client-initiated renegotiation is supported
     + Vulnerability threat level
       - High
     + Vulnerability description
       - Attacker may launch remote exploits to either crash the service or tools like ncrack to try brute-forcing the password of the target.
     + Impact assessment
       - The vulnerability has a high threat level, as it can lead to service disruptions or unauthorized access
     + Proposed mitigation or fix
       - It is recommended that to block the service outside world and make the service accessible only through the set of allowed IPs only necessary to prevent potential exploits.
  2. RDP server detected over UDP
     + vulnerability threat level
       - high
     + vulnerability description
       - hackers will be able to steal data from the backend and also they can authenticate themselves to the website and can impersonate as any user since they have total control over the backend. They can even wipe out the entire database. Attackers can also steal cookie information of an authenticated user and they can even redirect the target to any malicious address or totally deface the application.
     + Affected components
       - This vulnerability impacts the application's backend and data storage components
     + Impact assessment
       - The impact of this vulnerability is high, as it allows for data theft, unauthorized access, potential data loss, and various forms of user manipulation.
     + Steps to reproduce
       - Identify input fields or areas in the application that lack proper validation.
       - Inject malicious scripts or SQL queries into these input fields.
       - Observe how the application responds to these inputs, and whether it executes the injected code.
     + Proposed mitigation or fix
       - Proper input validation needs to be implemented before querying the database information
       - A developer should remember not to trust an end users input
       - Follow secure coding methodology
  3. WHOIS information publicly available (info)

# Entry 10

* Date – 2023/10/24
* Summary of the day's activities
  + Vulnerability scanning for reports and finalizing reports with all gathered contents
* Vulnerabilities discovered or explored.
  1. Uniscan detected possible XSS, SQLi, BSQLi
     + vulnerability threat level
       - high
     + vulnerability description
       - Hackers will be able to steal data from the backend and, they can authenticate themselves to the website and can impersonate as any user since they have total control over the backend. They can even wipe out the entire database. Attackers can also steal cookie information of an authenticated user and they can even redirect the target to any malicious address or totally deface the application.
     + Affected components
       - This vulnerability impacts the application's backend and data storage components
     + Impact assessment
       - The impact of this vulnerability is high, as it allows for data theft, unauthorized access, potential data loss, and various forms of user manipulation.
     + Steps to reproduce
       - Identify input fields or areas in the application that lack proper validation.
       - Inject malicious scripts or SQL queries into these input fields.
       - Observe how the application responds to these inputs, and whether it executes the injected code.
     + Proposed mitigation or fix
       - Proper input validation needs to be implemented before querying the database information
       - A developer should remember not to trust an end users’ input
       - Follow secure coding methodology

# Entry 11

* Date - 2023/10/24
* Summary of the day's activities
  + Vulnerability scanning for reports and finalizing reports with all gathered contents
* Vulnerabilities discovered or explored.
  1. FTP server detected
     + Vulnerability description
       - This protocol does not support secure communication and there are likely high chances for the attacker to eavesdrop the communication. Also, many FTP programs have exploits available on the web such that an attacker can directly crash the application or get a SHELL access to that target.
     + Affected components
       - FTP service running on the target system
     + Impact assessment
       - The impact of this vulnerability is significant, as it exposes communication to potential eavesdropping and the risk of the FTP service being exploited to compromise the target system's security.
     + Proposed mitigation or fix
       - To mitigate this vulnerability, it is recommended to replace the FTP protocol with SSH (Secure Shell) for secure communication. SSH provides a more secure alternative to FTP, ensuring confidentiality and data integrity during file transfers and remote access.
  2. MySQL DB server detected
     + Vulnerability threat level
       - low
     + Vulnerability description
       - Since the attacker has knowledge about the particular type of backend the target is running, they will be able to launch a targeted exploit for the particular version. They may also try to authenticate with default credentials to get themselves through.
     + Affected components
       - MySQL database services.
     + Proposed mitigation or fix
       - Timely security patches for the backend should be installed to address known vulnerabilities.
       - Change default credentials to more secure and unique ones to prevent unauthorized access

# Entry 12

* Date – 2023/10/26
* Summary of the day's activities
  + Vulnerability scanning for reports and finalizing reports with all gathered contents
* Vulnerabilities discovered or explored.
  1. Freak vulnerability detected
     + Vulnerability threat level
       - high
     + Vulnerability description
       - Attacker will be able to perform a MiM attack and it could compromise the confidentiality
     + Proposed mitigation or fix
       - Upgrading OpenSSL to latest version
     + Affected components
       - mitigate this vulnerability, it is recommended to upgrade OpenSSL to the latest version, which should include fixes for the FREAK vulnerability. Additionally, ensure that affected systems no longer support export-grade cryptography, as this is often the root cause of vulnerability.
  2. Open files with Uniscan
     + vulnerability threat level
       - medium
     + vulnerability description
       - attackers may find considerable amount of information from those files there is even chance attacker may get access to critical information from these files
     + Impact assessment
       - potential exposure of sensitive information.
     + Proposed mitigation or fix
       - Implement proper access controls, review, and secure sensitive files, and restrict access to unauthorized users.
       - Block or restrict access to these files unless necessary

# Entry 13

* Date 2023/10/27
* Summary of the day's activities
  + Vulnerability scanning for reports and finalizing reports with all gathered contents
* Vulnerabilities discovered or explored.
  + SNMP service detected
    - Vulnerability threat level
      * Medium
    - Vulnerability description
      * Attackers will be able to read the community strings through the service and enumerate quite a bit of information from the target
      * Also, there are multiple remote code execution and denial of service vulnerabilities related to SNMP services.
    - Affected components
      * SNMP service
    - Proposed mitigation or fix
      * Use a firewall to block the ports from the outside world

# Entry 14

* Date 2023/10/29
* Summary of the day's activities
  + Vulnerability scanning for reports and finalizing reports with all gathered contents
* Vulnerabilities discovered or explored.
  + No DNS/HTTP based load balancers found
    - vulnerability threat level
      * low
    - vulnerability definition
      * attackers may use the unavailability of load balancers as an advantage to leverage a denial-of-service attack on certain services or on the whole application itself
    - Affected components:
      * The vulnerability impacts the overall system's architecture, particularly in the context of network and traffic distribution.
    - proposed mitigation or fix
      * implementation of load balancers as that improve the performance times as well as data availability on during times of server outage