

Report 2

| Giáo viên | Hoàng Tuấn Anh |
| --- | --- |
| Thành viên nhóm | Nguyễn Trọng Dũng - HE172688  Nguyễn Duy Thái - HE170938  Trần Đức Thiện - HE171847  Chử Lê Minh Đăng - HE172600  Vũ Đình Sự - HE172791 |
| Môn học | SPM401 |

# **2.1. Problem Setting – Xác định vấn đề của dự án**

## **2.1.1. Tên dự án (Name of the CP):**

Nghiên cứu quy trình kiểm thử ứng dụng web, thực nghiệm.

## **2.1.2. Problem Abstraction:**

## Tóm tắt vấn đề Bảo mật trong ứng dụng web đóng vai trò then chốt đối với mọi doanh nghiệp hoạt động trên nền tảng trực tuyến. Với tính chất toàn cầu của Internet, các trang web luôn đối mặt với nguy cơ bị tấn công từ nhiều nguồn khác nhau, cả về vị trí địa lý lẫn mức độ tinh vi. Việc đảm bảo an toàn cho ứng dụng web bao gồm việc bảo vệ các trang web, ứng dụng web cũng như các dịch vụ liên quan như API khỏi các mối đe dọa tiềm ẩn.

## **2.1.3. Project Overview:**

**· Tình trạng hiện tại( Đánh giá các nghiên cứu đang có )**

Với sự phát triển mạnh mẽ của các ứng dụng web trong đời sống và hoạt động kinh doanh, các mối đe dọa về bảo mật ngày càng trở nên phổ biến và phức tạp. Bài báo "Web Application Vulnerabilities: Prevention and Detection Techniques" của M. A. Ferrag và cộng sự đã chỉ ra rằng trung bình mỗi ứng dụng web tồn tại khoảng 33 lỗ hổng bảo mật, trong đó có 6 lỗ hổng có mức độ nghiêm trọng cao, ảnh hưởng trực tiếp đến tính bảo mật, toàn vẹn và khả dụng của hệ thống. Đáng chú ý, 91% các ứng dụng web xử lý dữ liệu cá nhân và 50% các vụ rò rỉ gây lộ thông tin tài khoản, mật khẩu hoặc dữ liệu nhạy cảm. Nghiên cứu tập trung đánh giá hiệu quả của 12 công cụ quét lỗ hổng phổ biến, bao gồm cả công cụ mã nguồn mở (như OWASP ZAP, Vega) và công cụ thương mại (như Acunetix, Nessus, IBM AppScan), thông qua việc kiểm thử trên ứng dụng mô phỏng DVWA và so sánh với chuẩn OWASP Top 10. Kết quả cho thấy các công cụ có sự khác biệt đáng kể về khả năng phát hiện lỗ hổng và tỷ lệ cảnh báo sai. Một số công cụ phát hiện được nhiều lỗ hổng nhưng lại có tỷ lệ false positive cao, gây khó khăn trong việc xử lý và đánh giá đúng mức độ an toàn. Bài báo cũng nhấn mạnh tầm quan trọng của việc lựa chọn đúng công cụ và phương pháp kiểm thử (như kiểm thử hộp đen - black box testing) để phù hợp với đặc thù của từng hệ thống. Ngoài ra, tác giả kêu gọi cần có các tiêu chuẩn đánh giá hiệu quả công cụ kiểm thử bảo mật rõ ràng và thống nhất hơn nhằm hỗ trợ các tổ chức triển khai các biện pháp đảm bảo an toàn thông tin một cách hiệu quả và tiết kiệm chi phí.[[1](https://www.mdpi.com/2076-3417/12/8/4077)]

"A Survey on Web Application Security", tác giả cung cấp một cái nhìn toàn diện về lĩnh vực bảo mật ứng dụng web, hệ thống hóa các kỹ thuật hiện có thành ba nhóm chính: Security by Construction, Security by Verification và Security by Protection. Ứng dụng web hiện nay là nền tảng quan trọng cho nhiều dịch vụ trực tuyến nhưng thường xuyên gặp phải các lỗ hổng nghiêm trọng như XSS, SQL Injection, CSRF và session hijacking. Ba thuộc tính bảo mật cốt lõi của ứng dụng web được xác định gồm: xác thực dữ liệu đầu vào (Input Validity), bảo toàn trạng thái phiên làm việc (State Integrity) và tính chính xác của logic ứng dụng (Logic Correctness). Các giải pháp bảo mật hiện nay bao gồm xây dựng ứng dụng an toàn ngay từ đầu với các ngôn ngữ kiểm soát luồng dữ liệu và prepared statements (Security by Construction); kiểm tra và phát hiện lỗi qua phân tích tĩnh và động (Security by Verification); cũng như bảo vệ ứng dụng ở mức runtime bằng tường lửa ứng dụng và giám sát luồng dữ liệu (Security by Protection). Bài báo cũng chỉ ra các hạn chế hiện tại như độ chính xác của taint tracking, thiếu tin cậy của sanitization routines, và khó khăn trong việc tự động hóa bảo mật logic ứng dụng, đồng thời đề xuất hướng nghiên cứu tương lai như kết hợp nhiều kỹ thuật phân tích để nâng cao hiệu quả bảo mật. [[2](https://www.isis.vanderbilt.edu/sites/isis.vanderbilt.edu/files/bibcite_files/main_0_0.pdf)]

## *"A Survey on Web Application Penetration Testing"* cung cấp một cái nhìn toàn diện về kiểm thử xâm nhập ứng dụng web (WAPT), nhấn mạnh vai trò thiết yếu của nó trong việc bảo vệ hệ thống web khỏi các mối đe dọa bảo mật ngày càng tinh vi. Nội dung bài viết bao gồm phân tích quy trình kiểm thử gồm các bước: thu thập thông tin, phân tích lỗ hổng, khai thác và báo cáo; đánh giá chi tiết các lỗ hổng phổ biến theo OWASP Top 10 như truy cập trái phép, lỗi mã hóa, chèn mã độc, và cấu hình sai; đồng thời so sánh giữa kiểm thử thủ công và tự động, nêu rõ ưu và nhược điểm của từng phương pháp. Bài báo cũng tổng hợp nhiều công cụ phổ biến như Burp Suite, OWASP ZAP, SQLMap, và W3AF, đồng thời khuyến nghị nên kết hợp nhiều phương pháp kiểm thử cùng thực hành bảo mật tốt (session management, ghi log, cập nhật định kỳ) để nâng cao hiệu quả bảo vệ ứng dụng web. Tác giả kết luận rằng WAPT không chỉ là yêu cầu kỹ thuật mà còn là một phần cốt lõi trong chiến lược an ninh mạng toàn diện.[[3](https://www.mdpi.com/2079-9292/12/5/1229?fbclid=IwY2xjawKveGFleHRuA2FlbQIxMABicmlkETFBVjg2SVhqZGVySXVMREFmAR7sW-YkU-e_hDbC-8nkibEPQHUw1srZAPJOZtcpCJIjF4F1M7w7U8dlKGM-cw_aem_C9XTBvcXpxX-wHG-QNHTcg)]

## *"A Survey on Web Application Penetration Testing"* (Altulaihan et al., 2023) cung cấp cái nhìn tổng quan toàn diện về kiểm thử xâm nhập ứng dụng web (WAPT), nhấn mạnh vai trò của phương pháp này trong việc phát hiện và khắc phục lỗ hổng bảo mật trước khi bị khai thác. Bài viết trình bày chi tiết quy trình kiểm thử gồm các bước: thu thập thông tin, phân tích lỗ hổng, khai thác và báo cáo; đồng thời phân tích các lỗ hổng phổ biến theo OWASP Top 10 như kiểm soát truy cập kém, lỗi mã hóa, tấn công chèn mã và cấu hình sai. Tác giả cũng so sánh ưu nhược điểm giữa kiểm thử thủ công và tự động, từ đó khuyến nghị kết hợp cả hai để đạt hiệu quả tối ưu. Cuối cùng, bài báo đánh giá một số công cụ kiểm thử phổ biến như Burp Suite, OWASP ZAP, Nikto và SQLMap, đồng thời nhấn mạnh sự cần thiết của việc kiểm thử định kỳ, đào tạo an ninh mạng và tuân thủ các chuẩn bảo mật như OWASP để giảm thiểu rủi ro.

[[4](https://ieeexplore.ieee.org/abstract/document/8463920?fbclid=IwY2xjawKveN1leHRuA2FlbQIxMABicmlkETFBVjg2SVhqZGVySXVMREFmAR7-rlbM8J_Q4_dl5oeVdNjzyhHhWq7HDYduBax9UHHxy10eP1623PmA1JxRPw_aem_v5x-zbLawRXNM2DVbVLS8Q)]

"Web Application Penetration Testing" của Nagendran K và cộng sự trình bày một hướng dẫn kỹ thuật chi tiết về cách thực hiện kiểm thử xâm nhập thủ công cho các ứng dụng web, nhằm đánh giá tính toàn vẹn và bảo mật của chúng. Tác giả nhấn mạnh tầm quan trọng của việc kiểm tra các lỗ hổng bảo mật phổ biến được liệt kê trong OWASP Top 10, bao gồm các cuộc tấn công phía máy khách như XSS, CSRF, CORS và phía máy chủ như SQL Injection, Command Injection. Bài báo phân biệt giữa đánh giá lỗ hổng (chỉ phát hiện và báo cáo lỗi) và kiểm thử xâm nhập (thực hiện khai thác để đánh giá mức độ nghiêm trọng). Tác giả cũng khuyến nghị thực hiện kiểm thử định kỳ, đặc biệt sau khi triển khai hoặc cập nhật ứng dụng, để phát hiện và khắc phục các lỗ hổng mới xuất hiện. Bài viết cung cấp một phương pháp thực hành hữu ích cho các nhà phát triển và chuyên gia bảo mật trong việc bảo vệ ứng dụng web khỏi các mối đe dọa ngày càng gia tăng. [[5](https://d1wqtxts1xzle7.cloudfront.net/105013940/J91730881019-libre.pdf?1692086155=&response-content-disposition=inline%3B+filename%3DWeb_Application_Penetration_Testing.pdf&Expires=1749196534&Signature=QDsoKMDCRSUjPlk1jlm6YaDsRx5lwpBGUtfWs7XYOAmsfHDh2I~FUW~z9yTpK6lQw6DglvFUK352H1clgsjnLz2jB9R4mbmuQvL-cd-eME~nwJlQmCRMk7s~Sz2FvOjtkVvegwtdCqF-3ioScAG7p2zwamJPU2VNvMwhawYFxYmli35-RJ~k5MYdATMx8T7XOz0cXhET5ZDH8K9Slvxr7vlEwkZFEAEdKrNY3arz-B3pwU5P8xMVY~HnUw67NsVXGU-Dx7ATmwfWE2FYjGq30-K~VXWpXPTo~LKSg5nOiwW64XmLQ~k9LPfHlkJZn6ZlJqmx58aqhL6mRAPNx0rvYA__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA)]

"WAPTT – Web Application Penetration Testing Tool" của Zoran Đurić, đăng trên tạp chí Advances in Electrical and Computer Engineering năm 2014, giới thiệu một công cụ kiểm thử xâm nhập ứng dụng web mang tên WAPTT. Công cụ này được thiết kế để phát hiện các lỗ hổng bảo mật phổ biến như SQL Injection (SQLI), Cross-Site Scripting (XSS) và Buffer Overflow (BOF) thông qua phương pháp kiểm thử hộp đen (black-box), không yêu cầu truy cập vào mã nguồn ứng dụng. WAPTT hoạt động theo năm giai đoạn: thu thập dữ liệu (web crawling), phát hiện và trích xuất các điểm đầu vào của ứng dụng, kiểm thử, phân tích và tạo báo cáo. Một điểm nổi bật của WAPTT là khả năng mở rộng linh hoạt nhờ kiến trúc mô-đun, cho phép người dùng dễ dàng tùy chỉnh và bổ sung chức năng. Ngoài ra, công cụ này sử dụng thuật toán phát hiện sự tương đồng giữa các trang để nâng cao hiệu quả trong việc phát hiện lỗ hổng SQLI. Kết quả đánh giá cho thấy WAPTT đạt hiệu suất phát hiện lỗ hổng tương đương hoặc vượt trội so với sáu công cụ kiểm thử ứng dụng web phổ biến khác, chứng minh tính hiệu quả và tiềm năng ứng dụng trong thực tế. [[6](https://d1wqtxts1xzle7.cloudfront.net/46768547/aece_2014_1_15-libre.pdf?1466808275=&response-content-disposition=inline%3B+filename%3DWAPTT_Web_Application_Penetration_Testin.pdf&Expires=1749197831&Signature=R6niCUav~xRhBwCrddjUkTLjrJpCP7964nvCSUmZ7rQHph~Y4HudZP5QL0DDBXn-VOoHrcvFClZAOUuVZEzwV8BcdMqC-eUoDJYCxen-sA-12p4lb9rFxoM-R22Yam~kqz2SnPBxdJJllQrVvouJrD2mgVvmmAs4Zss3lYcqDqB1Rv7Oq~ZlGG6WELBsnXrwkp0Qz4RjE1qMO9z9F2i1fszawSscB0cRXXE-21NJ1VcgYX9GiHujtRONBLqASNXez7KYeGTOYiedDG7Jc0eGrag-jSCHncxpQxS0V8dgCTYv1JDvl36kCc7Ha6Hsb9Q6TLQqqcy2L2T5wxpwRcVxGA__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA)]

"A Review on Web Application Vulnerability Assessment and Penetration Testing" của Urshila Ravindran và Raghu Vamsi Potukuchi, đăng trên Review of Computer Engineering Studies năm 2022, cung cấp một tổng quan toàn diện về các lỗ hổng bảo mật phổ biến trong ứng dụng web và quy trình đánh giá lỗ hổng kết hợp kiểm thử xâm nhập (VAPT). Tác giả nhấn mạnh tầm quan trọng của việc thực hiện VAPT định kỳ để phát hiện và khắc phục các điểm yếu bảo mật, đặc biệt trong bối cảnh gia tăng các cuộc tấn công mạng nhắm vào dữ liệu người dùng và hệ thống doanh nghiệp. Bài viết trình bày các yêu cầu cần thiết trước khi tiến hành đánh giá bảo mật, các phương pháp kiểm thử như kiểm thử hộp đen, hộp trắng, và hộp xám, cùng với các công cụ hỗ trợ như OWASP ZAP, Burp Suite, Nmap, Acunetix và Nikto. Ngoài ra, tác giả cũng đưa ra các khuyến nghị về quy trình vá lỗi sau khi phát hiện lỗ hổng, nhằm đảm bảo tính toàn vẹn và an toàn cho ứng dụng web. Bài báo là tài liệu hữu ích cho các chuyên gia bảo mật và nhà phát triển trong việc nâng cao nhận thức và kỹ năng bảo vệ ứng dụng web khỏi các mối đe dọa ngày càng tinh vi. [[7](https://www.researchgate.net/profile/Raghu-Potukuchi/publication/360135777_A_Review_on_Web_Application_Vulnerability_Assessment_and_Penetration_Testing/links/6291a97e55273755ebbbf6df/A-Review-on-Web-Application-Vulnerability-Assessment-and-Penetration-Testing.pdf)]

**· Giải pháp đề xuất**

Để đảm bảo giảm thiểu rủi ro cao nhất trên nền tảng ứng dụng web, việc kiểm thử thâm nhập theo một quy trình chuẩn và đúng là điều không thể thiếu, OWASP là một tổ chức chuẩn mực với đội ngũ các nhà nghiên cứu giàu kinh nghiệm trong lĩnh vực công nghệ bảo mật ứng dụng web và có cộng đồng bảo mật thông tin lớn đóng góp và phát triển các kỹ thuật tấn công và phòng thủ, đó là lý do tại sao hầu hết các tổ chức an ninh mạng tin tưởng tuân thủ các quy tắc và quy trình.

Hầu hết các kiểm thử viên thâm nhập đều buộc phải kiểm thử thâm nhập thủ công với sự trợ giúp của các công cụ tự động để tránh bỏ sót và che phủ càng nhiều lỗ hổng bảo mật càng tốt. Do đó, nhóm của chúng tôi đã lên kế hoạch nghiên cứu các yêu cầu thực tiễn trong hoạt động kiểm thử xâm nhập, từ đó đề xuất các tiêu chí và giải pháp giúp kiểm thử viên tùy chỉnh quá trình gửi yêu cầu, báo cáo kết quả một cách linh hoạt và hỗ trợ mở rộng phạm vi quét lỗ hổng một cách hiệu quả.

**· Phạm vi giải pháp**

Dự án nghiên cứu này tập trung vào việc đề xuất một quy trình kiểm thử bảo mật có khả năng mở rộng, cho phép người kiểm thử tùy chỉnh các bước kiểm tra theo nhu cầu thực tế và dễ dàng chia sẻ mẫu kiểm thử với cộng đồng. Quy trình này cũng hướng đến việc đánh giá hiệu quả khi áp dụng đồng thời cho nhiều loại lỗ hổng và nhiều mục tiêu. Phạm vi nghiên cứu bao gồm bốn nội dung chính:

· Khảo sát và phân loại các mẫu kiểm thử tùy chỉnh phổ biến cho lỗ hổng ứng dụng web.

· Phân tích quy trình gửi yêu cầu có chứa payload đến mục tiêu dựa trên các mẫu đã xây dựng.

· Xây dựng tiêu chí và phương pháp phân tích phản hồi để xác định sự tồn tại của lỗ hổng.

· Đề xuất định dạng báo cáo kiểm thử theo hướng có thể tùy chỉnh và dễ tích hợp vào quy trình đánh giá bảo mật.

Những nội dung ngoài phạm vi dự án:

· Không mở rộng nghiên cứu sang kiểm thử bảo mật cho ứng dụng di động hoặc các API chuyên biệt.

· Không đi sâu vào tích hợp với hệ thống CI/CD hay quy trình DevOps.

· Không bao gồm hoạt động khai thác sâu hoặc chiếm quyền kiểm soát hệ thống sau khi phát hiện lỗ hổng.

**· Môi trường phát triển**

Công nghệ, phần mềm và công cụ bảo mật sử dụng:

* OWASP ZAP – Công cụ mã nguồn mở dùng để phân tích và phát hiện lỗ hổng ứng dụng web.
* Burp Suite (Community Edition) – Công cụ kiểm thử xâm nhập ứng dụng web, sử dụng để phân tích lưu lượng HTTP/HTTPS và kiểm tra bảo mật.
* Nikto – Trình quét máy chủ web mã nguồn mở, giúp phát hiện các cấu hình sai hoặc lỗ hổng bảo mật.
* WordPress (phiên bản lỗi có chủ đích) – CMS phổ biến được cấu hình chứa các lỗ hổng bảo mật nhằm phục vụ mục đích kiểm thử.
* DVWA (Damn Vulnerable Web Application) – Ứng dụng web cố tình chứa lỗ hổng, dùng làm môi trường thử nghiệm.

Hệ điều hành:

* Ubuntu 24.04 – Dùng làm máy chủ chạy các ứng dụng mục tiêu như DVWA hoặc WordPress.
* Kali Linux – Hệ điều hành chuyên dụng cho kiểm thử xâm nhập, dùng để cài đặt và vận hành các công cụ như Burp Suite, ZAP, Nikto.
* Windows 10/11 – Hệ điều hành cài đặt VMware Workstation Pro để triển khai môi trường ảo hóa.Môi trường mạng: Mạng nội bộ hoặc internet trong phạm vi kiểm thử hợp pháp

Môi trường mạng:

* Mạng nội bộ (LAN) sử dụng trong môi trường ảo hóa, cô lập hoàn toàn với Internet.
* Các máy ảo kết nối thông qua mạng NAT hoặc Host-only để đảm bảo an toàn và kiểm soát truy cập.

Ảo hóa và triển khai:

* VMware Workstation Pro – Dùng để tạo và quản lý các máy ảo chạy Kali Linux, Ubuntu, Windows.
* Mỗi máy ảo được cấp phát tài nguyên tối thiểu: 2 CPU, 4 GB RAM, 20–40 GB ổ đĩa cứng.
* Phần mềm hỗ trợ phân tích và báo cáo:
* Microsoft Excel / Google Sheets – Lưu trữ, xử lý và thống kê kết quả kiểm thử bảo mật.
* Microsoft Word / LaTeX – Soạn thảo tài liệu nghiên cứu, báo cáo kết quả, và trình bày quy trình đề xuất.

# **2.2. Project Organization – Tổ chức thực hiện dự án**

Trong dự án này, nhóm lựa chọn mô hình nghiên cứu khoa học làm cơ sở triển khai. Mô hình này gồm các bước nghiên cứu chuẩn, giúp nhóm tiếp cận vấn đề một cách hệ thống, khoa học và có chiều sâu.

**Team Roles – Phân công vai trò**

Dựa trên mô hình nghiên cứu khoa học, nhóm chia tiến độ thành các giai đoạn nghiên cứu chính. Mỗi thành viên đảm nhiệm vai trò cụ thể nhằm đảm bảo chất lượng trong từng bước của quy trình.

| **Giai đoạn** | **Thành viên** | **Vai trò** | **Trách nhiệm cụ thể** |
| --- | --- | --- | --- |
| **Xác định vấn đề** | Nguyễn Trọng Dũng | Research Leader | Điều phối toàn bộ dự án, xác định hướng nghiên cứu, mục tiêu cụ thể và phạm vi nghiên cứu. |
| Nguyễn Duy Thái | Assistant Planner | Hỗ trợ xây dựng mục tiêu, thu thập ý kiến nhóm, đảm bảo tính thực tiễn và khả thi của đề tài. |
| **Thu thập và phân tích tài liệu** | Vũ Đình Sự | Literature Collector | Tìm kiếm và tổng hợp các tài liệu liên quan từ OWASP, PortSwigger, các bài báo học thuật. |
| Chử Lê Minh Đăng | Tool Research Analyst | Phân tích chức năng, điểm mạnh/điểm yếu của các công cụ kiểm thử bảo mật phổ biến như ZAP, Burp, Nikto,... |
| Trần Đức Thiện | Environment Reviewer | Tổng hợp mô hình môi trường thử nghiệm từ các tài liệu, đề xuất cấu hình triển khai phù hợp để phục vụ thực nghiệm. |
| **Thiết kế nghiên cứu và thực nghiệm** | Nguyễn Duy Thái | Experiment Designer | Xây dựng quy trình thử nghiệm và tiêu chí đánh giá công cụ: độ chính xác, độ thân thiện, dễ sử dụng,... |
| Trần Đức Thiện | Lab Setup Engineer | Thiết lập môi trường kiểm thử (DVWA, WordPress, Apache) và đảm bảo tính đồng nhất giữa các công cụ thử nghiệm. |
| Nguyễn Trọng Dũng | Testing Supervisor | Giám sát việc thực nghiệm, đảm bảo việc ghi nhận kết quả nhất quán và khách quan. |
| **Phân tích dữ liệu và đánh giá** | Chử Lê Minh Đăng | Data Analyst | Phân tích kết quả thử nghiệm, so sánh hiệu quả công cụ, lập bảng tổng hợp. |
| Vũ Đình Sự | Insight Synthesizer | Trích xuất nhận xét, nhận định từ dữ liệu phân tích để hỗ trợ phần kết luận và đề xuất. |
| **Viết báo cáo & tổng hợp kết quả** | Vũ Đình Sự | Report Writer | Biên soạn nội dung báo cáo theo chuẩn học thuật, trình bày mạch lạc và rõ ràng. |
| Nguyễn Trọng Dũng | Final Reviewer | Rà soát nội dung cuối cùng, đảm bảo tính chính xác, tính logic và học thuật trong báo cáo. |
| Cả nhóm | Contributor | Thảo luận và phản biện chéo để hoàn thiện báo cáo, đề xuất hướng phát triển tiếp theo. |

**Tools and Techniques – Công cụ và kỹ thuật sử dụng**

Trong dự án nghiên cứu này, nhóm lựa chọn sử dụng tập hợp các công cụ và nền tảng phục vụ cho quá trình phân tích, thử nghiệm và đánh giá quy trình kiểm thử bảo mật ứng dụng web. Mục tiêu là hỗ trợ người học – đặc biệt là người mới tiếp cận lĩnh vực pentest – hiểu rõ cách các công cụ hiện có hoạt động, điểm mạnh/yếu của chúng, và từ đó rút ra quy trình phù hợp nhất.

| **STT** | **Công cụ / Nền tảng** | **Mô tả chi tiết và lý do sử dụng** |
| --- | --- | --- |
| **1** | **OWASP ZAP** | Công cụ kiểm thử bảo mật mã nguồn mở, dễ sử dụng, phù hợp với người mới. Dùng để thử nghiệm khả năng quét lỗ hổng tự động. |
| **2** | **Burp Suite (Community)** | Công cụ proxy phổ biến trong pentest. Dùng để kiểm thử bán tự động các lỗ hổng như XSS, SQLi, CSRF, giúp so sánh hiệu quả với ZAP. |
| **3** | **Nikto** | Công cụ quét web server đơn giản, phục vụ đánh giá mức độ bao phủ các loại lỗ hổng khác nhau khi dùng công cụ cơ bản. |
| **4** | **DVWA (Damn Vulnerable Web App)** | Ứng dụng web có chủ đích chứa nhiều lỗ hổng, phục vụ làm mục tiêu kiểm thử. Phù hợp cho các thử nghiệm có kiểm soát. |
| **5** | **WordPress (phiên bản lỗi)** | CMS phổ biến dễ cấu hình sai, được nhóm sử dụng để kiểm thử thực tế khả năng phát hiện lỗ hổng từ các công cụ. |
| **6** | **Kali Linux** | Hệ điều hành chuyên dùng cho pentest, tích hợp sẵn nhiều công cụ kiểm thử và mạng giả lập, hỗ trợ thiết lập môi trường kiểm thử. |
| **7** | **Ubuntu 24.04** | Hệ điều hành ổn định, mã nguồn mở. Được dùng làm nền tảng triển khai ứng dụng mục tiêu (DVWA, WordPress). |
| **8** | **VMWare Workstation Pro** | Công cụ tạo máy ảo giúp tạo môi trường thử nghiệm an toàn, biệt lập với hệ điều hành chính. |
| **9** | **Excel / Google Sheets** | Dùng để lưu trữ và phân tích dữ liệu kết quả thử nghiệm, lập bảng so sánh công cụ theo tiêu chí đã xây dựng. |
| **10** | **Microsoft Word / LaTeX** | Dùng để biên soạn báo cáo nghiên cứu học thuật, đảm bảo trình bày chuyên nghiệp và rõ ràng. |

# **2.3. Project Management Plan – Kế hoạch quản lý dự án**

### **Task 1: Khảo sát và thu thập tài liệu nghiên cứu**

* Kết quả cần đạt (Deliverables):  
  Danh mục tài liệu tham khảo chất lượng (OWASP, học thuật, công cụ).  
  Tổng hợp phương pháp kiểm thử và tiêu chí đánh giá công cụ.
* Tài nguyên cần thiết:  
  Truy cập Internet, tài khoản Google Scholar, GitHub, OWASP.  
  Thời gian đọc hiểu tài liệu chuyên ngành.
* Các ràng buộc, phụ thuộc:  
  Phụ thuộc vào khả năng truy cập và hiểu nội dung tài liệu.  
  Phải hoàn tất trước khi thiết kế môi trường và kịch bản kiểm thử.
* Nguy cơ/rủi ro liên quan:  
  Tài liệu không đầy đủ hoặc không phù hợp.  
  Mất thời gian lọc tài liệu dẫn đến trễ tiến độ.

### **Task 2: Xây dựng và cấu hình môi trường kiểm thử**

* Kết quả cần đạt (Deliverables):  
  Hệ thống máy ảo (VMware) chạy ổn định với Kali, Ubuntu, DVWA.  
  Công cụ kiểm thử cài đặt đầy đủ (ZAP, Burp, Nikto...).
* Tài nguyên cần thiết:  
  File ISO hệ điều hành, phần mềm VMware Workstation.  
  Tài liệu hướng dẫn cấu hình DVWA, WordPress, và công cụ.
* Các ràng buộc, phụ thuộc:  
  Phụ thuộc vào cấu hình máy tính cá nhân.  
  Cần hoàn tất trước khi bắt đầu kiểm thử.
* Nguy cơ/rủi ro liên quan:  
  Cài đặt lỗi, phần mềm không tương thích.  
  Môi trường không ổn định gây sai lệch kết quả.

### **Task 3: Thiết kế và thực hiện kiểm thử thực nghiệm**

* Kết quả cần đạt (Deliverables):  
  Bộ kết quả kiểm thử (log, báo cáo từ ZAP, Burp...).  
  Ghi chú quy trình kiểm thử và lỗi phát hiện được.
* Tài nguyên cần thiết:  
  Môi trường đã thiết lập.  
  Công cụ kiểm thử bảo mật (ZAP, Burp, Nikto...).
* Các ràng buộc, phụ thuộc:  
  Phụ thuộc vào môi trường đã hoạt động ổn định.  
  Cần đảm bảo thực hiện đúng kịch bản kiểm thử.
* Nguy cơ/rủi ro liên quan:  
  Công cụ không phát hiện ra lỗi rõ ràng.  
  Kết quả kiểm thử thiếu nhất quán do thao tác chưa chính xác.

### **Task 4: Phân tích kết quả và viết báo cáo**

* Kết quả cần đạt (Deliverables):  
  Báo cáo tổng hợp kết quả, phân tích và nhận xét.  
  Tài liệu trình bày đầy đủ quy trình, biểu đồ và đánh giá.
* Tài nguyên cần thiết:  
  File kết quả từ công cụ kiểm thử.  
  Công cụ xử lý văn bản và bảng tính (Word, Excel).
* Các ràng buộc, phụ thuộc:  
  Phụ thuộc vào độ chính xác của dữ liệu đầu vào.  
  Phải hoàn thành đúng deadline để nộp báo cáo.
* Nguy cơ/rủi ro liên quan:  
  Phân tích thiếu logic hoặc sai nội dung kỹ thuật.  
  Báo cáo trình bày chưa khoa học, khó hiểu.

**Phân công nhiệm vụ và tiến độ**

| Nhiệm vụ | Ngày bắt đầu | Ngày kết thúc |
| --- | --- | --- |
| Khởi động dự án | 21/05/2025 | 23/05/2025 |
| Phân công nhiệm vụ lần 1 | 24/05/2025 | 25/05/2025 |
| Nghiên cứu lý thuyết và thu thập tài liệu | 26/05/2025 | 01/06/2025 |
| Viết và nộp báo cáo 1 + 2 |  |  |
| Phân công nhiệm vụ lần 2 | 10/06/2025 | 10/06/2025 |
| Tìm hiểu quy trình kiểm thử OWASP | 11/06/2025 | 14/06/2025 |
| Viết và nộp báo cáo 3 |  |  |
| Họp nội bộ và rà soát mô hình quy trình | 20/06/2025 | 21/06/2025 |
| Phân công nhiệm vụ lần 3 | 22/06/2025 | 22/06/2025 |
| Thiết lập và thử nghiệm môi trường giả lập | 23/06/2025 | 24/06/2025 |
| Thực nghiệm quy trình kiểm thử | 25/06/2025 | 05/07/2025 |
| Tổng hợp và đánh giá hiệu quả quy trình | 06/07/2025 | 07/07/2025 |
| Nhóm rà soát và báo cáo | 08/07/2025 | 08/07/2025 |
| Viết và nộp báo cáo 4 + 5 + 6 |  |  |
| Nộp báo cáo cuối cùng | 15/07/2025 | 16/07/2025 |
| Chuẩn bị slide | 17/07/2025 | 17/07/2025 |
| Diễn tập thuyết trình | 18/07/2025 | 22/07/2025 |
| Thuyết trình cuối cùng | 23/07/2025 | 23/07/2025 |

**Biên bản họp**

Học phần: SPM401  
Thời gian: 21/05/2025 – 23/07/2025  
Khung giờ họp: 20:30 – 22:00, hai buổi mỗi tuần  
Hình thức: Google Meet

Thành viên tham dự:

* Trần Đức Thiện
* Nguyễn Trọng Dũng
* Vũ Đình Sự
* Nguyễn Duy Thái
* Chử Lê Minh Đăng

Nội dung họp:  
Báo cáo tiến độ, rà soát các công việc đã thực hiện và lên kế hoạch cho các bước tiếp theo.