

POLITECNICO DI TORINO

Master's Degree in Cybersecurity



Master's Degree Thesis

Enterprise Penetration Testing: A Comparative Analysis of Frameworks, Automation, and LLM-Assisted Support

Supervisors:

Prof. Cataldo Basile

Candidate:

Gioele Fontanini

Academic Year 2025/2026
Torino

Abstract

Penetration testing plays a central role in the security assessment of modern enterprise environments, where heterogeneous infrastructures, multiple exposure surfaces, and increasingly sophisticated cyber threats make vulnerability evaluation more complex than in the past. In such contexts, purely ad hoc testing practices are often insufficient, while the adoption of a single methodological framework may not fully address the operational, organizational, and analytical requirements of enterprise security assessments. At the same time, automated tools have become essential for improving scalability and efficiency, although their outputs often require careful interpretation because of incomplete coverage, false positives, and limited contextual awareness.

This thesis addresses the problem of how penetration testing methodologies can be structured and applied more effectively in enterprise environments by combining methodological guidance, automation, and analytical support. The work adopts a comparative and methodological perspective. First, it examines major penetration testing frameworks and discusses their relevance, strengths, and limitations with respect to enterprise applicability. To support this analysis, the thesis defines a structured comparison model based on explicit evaluation criteria, aimed at assessing aspects such as coverage, operational depth, formalization, enterprise suitability, automation support, reporting and validation support, and adaptability to hybrid workflows.

Building on this comparative analysis, the thesis then discusses the role of automated tools in practical penetration testing and examines the potential contribution of LLM-assisted support in activities such as result interpretation, preliminary validation, and attack-path reasoning. Rather than treating artificial intelligence as a replacement for the analyst, the work considers it as a complementary aid within a broader hybrid methodological approach in which human validation remains essential.

The proposed perspective is finally discussed through a real-world enterprise penetration testing experience conducted during an internship at KPMG, which involved assessing external, internal, and physical security dimensions. The thesis therefore aims to provide a structured discussion of penetration testing methodologies in enterprise settings, with particular attention to the interplay among reference frameworks, automation, LLM-assisted analytical support, and expert-driven validation.

Ringraziamenti

*“Non est ad astra mollis e terris via.”
Non esiste una strada semplice dalla terra alle stelle.
Seneca*

Arrivato alla fine di questo percorso, sento il bisogno di fermarmi e dedicare qualche parola a tutte le persone che, in modi diversi, hanno fatto parte di questi anni. Questo traguardo porta il mio nome, ma dentro ci sono il sostegno, la presenza, i consigli e l'affetto di tante persone che mi hanno accompagnato lungo il cammino.

Desidero ringraziare il mio relatore, il **Prof. Cataldo Basile**, per la disponibilità e per il supporto fornito durante lo sviluppo di questo lavoro. Lo ringrazio per le osservazioni, per i consigli e per avermi aiutato a dare una forma più chiara e solida alla tesi, guidandomi nel collegare l'esperienza svolta in azienda con una riflessione metodologica più ampia. Desidero ringraziarlo anche per la simpatia e per la capacità di rendere più leggeri, con naturalezza, anche alcuni momenti più strettamente accademici. Il suo confronto è stato importante per rendere il lavoro più consapevole e più equilibrato.

Sul piano personale, il primo ringraziamento va a **mia madre** e a **mio padre**. Grazie per avermi insegnato, ciascuno a modo proprio, ciò che la vita vi ha dato e ciò che avete imparato lungo il vostro percorso. Grazie per avermi dato la forza e la possibilità di scegliere sempre liberamente cosa fare, aiutandomi però a capire, anche nei momenti più complessi, quale fosse la strada giusta da seguire. Se oggi arrivo a questo risultato con consapevolezza e determinazione, è anche perché mi avete trasmesso il valore dell'impegno, della responsabilità e della libertà.

A **mia madre** va un grazie speciale per essere sempre stata la mia fan n.1, con la sua forza, la sua sensibilità e la sua capacità di sostenermi anche quando le parole non erano necessarie. Grazie per avermi insegnato a ponderare ogni scelta, a mettere in fila ogni tassello e a costruire passo dopo passo la strada migliore per arrivare all'obiettivo. Grazie per aver seguito ogni mio passo da vicino, con attenzione, presenza e fiducia, aiutandomi a non perdere mai di vista ciò che contava davvero. Insieme a lei, desidero ringraziare anche **Michele**, che da anni rappresenta un'ottima spalla nella sua vita e nella mia. Con garbo, eleganza e simpatia, ha sempre trovato il modo giusto per starmi accanto, offrirmi consigli scolastici, di vita e persino musicali, senza mai imporsi ma riuscendo sempre a lasciare qualcosa di importante.

A **mio padre** va il mio grazie per la forza, il carattere e l'esempio che mi ha sempre trasmesso. Da lui ho imparato cosa significhi davvero il sacrificio, cosa voglia dire non

fermarsi davanti a nulla e continuare a cercare una strada anche quando io non riuscivo ancora a vederla. Grazie per avermi insegnato a non arrendermi, a difendere le mie idee e ad affrontare le difficoltà con determinazione. Grazie anche per essere stato, tante volte, uno scudo contro ogni freccia che la vita ha provato a scagliare, proteggendomi e sostenendomi anche nei momenti più complessi. Accanto a lui, ringrazio anche **Sabrina**, per essere capace di gestire e ponderare la sua forza caratteriale e, sia nei momenti di contrasto sia in quelli di piena concordanza di pensiero, per avermi offerto la sua esperienza, la sua visione della vita e consigli preziosi.

Un ringraziamento va anche alla mia famiglia: **Zia Maria, Zia Minusk, Zio Gaetano, Zio Antonio, Zio Massimo, Zia Roberta, Matteo, Nonna Maria, Nonno Giuseppe e Nonno Salvo**. Grazie per avermi sempre supportato, incoraggiato e accompagnato nelle mie scelte, ognuno con la propria presenza e con il proprio modo di volermi bene.

Un pensiero speciale va anche a chi non c'è più, in particolare a **Nonna Angela**. Nella sua semplicità e nel suo modo unico di vivere la vita come un'artista, ha sempre saputo riconoscere l'importanza di questo traguardo. Il suo ricordo resta per me una presenza silenziosa ma profonda, capace ancora oggi di accompagnarmi.

Ringrazio **Federico**, anche noto come **Pino**, che è stato molto più di un cugino: un fratello. Lo ringrazio per avermi consigliato di venire a Torino e per avermi aiutato, forse più di quanto immagini, rafforzando un rapporto che, fin da bambini, nonostante la distanza, abbiamo sempre coltivato. Il suo consiglio ha avuto un peso importante in una delle scelte più significative del mio percorso, e sapere di poter contare su una presenza familiare in una città nuova ha reso questo cambiamento più semplice e meno solitario. In questi anni è stato un punto di riferimento, una persona con cui confrontarmi, scherzare, condividere pensieri, difficoltà e momenti di leggerezza. Oggi rappresenta un punto fermo della mia vita, una presenza sincera su cui so di poter contare e uno di quei legami che il tempo e la distanza non hanno fatto altro che rafforzare. Non ho dubbi che questo rapporto continuerà ad accompagnarci per tutta la vita.

Grazie anche a **Giuseppe**, mio coinquilino, che con calma e pazienza è riuscito a diventare molto più di un collega o di una persona con cui condividere casa. Oggi lo considero un amico fondamentale e un ottimo consigliere, capace di esserci con discrezione, equilibrio e sincerità.

Un ringraziamento speciale va a **Sofia**, una delle persone più importanti che Torino mi abbia regalato. Nel bene e nel male, è sempre stata presente durante questo percorso, sapendo scuotere la mia razionalità e portandomi a vivere alcune delle esperienze più stravaganti e indimenticabili della mia vita. La ringrazio per avermi mostrato un modo diverso di guardare le cose, per avermi spinto spesso fuori dalla mia zona di comfort.

Ringrazio gli amici di una vita: **Roberto, Gabriele, Christian e Francesca**. Grazie per essermi stati accanto anno dopo anno, per aver attraversato con me fasi diverse della vita e per essere rimasti un punto fermo, anche quando tutto il resto cambiava. Insieme abbiamo vissuto momenti indimenticabili, siamo cresciuti, abbiamo sbagliato, imparato e migliorato passo dopo passo, costruendo un legame che il tempo non ha fatto altro che rafforzare. La vostra amicizia è una di quelle certezze che non ha bisogno di troppe spiegazioni, ma che si riconosce nella presenza, nella sincerità e nella naturalezza con cui siete sempre stati parte della mia vita. Non vedo l'ora di vedere cosa vi riserverà il futuro,

perché siete tra le persone più belle che io conosca e meritate tutto il bene che la vita saprà donarvi.

Un grazie va anche ai miei colleghi di università: **Luca C, Erik, Pietro, Carolina, Diego, Flavio, Francesco, Fabrizio, Gabriele M, Alessandro, Marco T, Christian, Claudia, Debora, Luca F, Giulia, Marco S, Michele, Gaetano, Carmelo, Gabriele S e Matteo**. Grazie a chi è stato presente fin dalla triennale, a chi è arrivato durante la magistrale, a chi considero un fratello, a chi mi ha regalato un bel rapporto e belle esperienze, e anche a chi, semplicemente, ha saputo offrirmi momenti di spensieratezza durante il percorso universitario. Ognuno di voi ha contribuito, in modo diverso, a rendere questi anni più pieni e più leggeri.

Desidero ringraziare anche i colleghi conosciuti durante il tirocinio: **Valeria, Alessandro, Enrico, Fabiana, Fabiola, Chiara, Morgan, Marianna, Roberta, Pietro, Alessio, Federico, Elsa, Nicholas, Luca, Lorenzo e Antonino**. Grazie a chi è entrato nella mia quotidianità e oggi posso considerare una persona importante nella mia vita, ma anche a chi, con una semplice battuta in ufficio o con un momento condiviso, ha contribuito a creare un clima piacevole e umano. Avete reso il lavoro meno pesante.

Ringrazio **Alessandro, Paolo, Pierpaolo e Daniele**, per aver reso il pranzo domenicale un momento di svago, leggerezza e normalità durante tutto questo percorso. In mezzo agli impegni, alle scadenze e alle responsabilità, quei momenti hanno rappresentato una pausa semplice ma preziosa.

Un ringraziamento speciale va anche a **Marco e Cala**, i miei **lini**, per essermi stati accanto fin dai tempi del liceo. Anche a distanza, siete riusciti a regalarmi momenti di leggerezza e spensieratezza, spesso anche solo con una semplice chiamata. In un percorso fatto di cambiamenti, responsabilità e nuove strade, sapere di poter ritrovare in voi quella naturalezza di sempre è stato qualcosa di prezioso. Grazie per avermi ricordato, anche nei periodi più pieni, l'importanza delle amicizie semplici, sincere e capaci di resistere al tempo e alla distanza.

Infine, ringrazio gli amici che continuo a coltivare da Palermo: **Stefano, Claudio, Claudia, Rosangela, Alessandro D, Dario D, Alessandro A, Dario A, Rosalba, Roberto, Irene, Giuseppe A, Giuseppe G, Giovanna, Giorgia, Alessia, Loredana, Manfredi, Greta, Sara, Antonella, Walter, Roberta D, Sergio, Salvo, Alessandro, Giulio, Sveva** e molti altri. Anche se la distanza cambia le abitudini e rende più difficile condividere la quotidianità, certi legami restano e continuano ad accompagnarmi. A tutti voi va il mio grazie per essere parte della mia storia, anche da lontano.

Un pensiero va anche a chi, per motivi diversi, non è riuscito ad arrivare a questo traguardo o ha dovuto interrompere il proprio cammino lungo la strada. A loro auguro ogni bene, con la speranza che possano continuare a credere nei propri sogni e trovare sempre la forza di lottare per ciò che desiderano davvero. Ogni percorso ha i suoi tempi, le sue difficoltà e le sue deviazioni, ma nessun sogno perde valore solo perché richiede più tempo, più fatica o una strada diversa da quella immaginata.

A tutte queste persone, e a chi in modi diversi ha incrociato il mio cammino, va il mio grazie più sincero. Questo traguardo è anche il risultato delle relazioni, degli incontri e degli affetti che mi hanno sostenuto fino a qui.

Table of Contents

List of Figures	XIII
Acronyms	XV
1 Introduction	1
1.1 Context and Motivation	1
1.1.1 Increasing Complexity of Enterprise Infrastructures	1
1.1.2 Growing Sophistication of Cyber Threats	2
1.1.3 Need for Structured Penetration Testing Methodologies	2
1.1.4 Expanding Role of Automation in Security Assessment	2
1.1.5 Emergence of AI and LLM-Assisted Support in Cybersecurity Analysis	3
1.2 Aim and Scope of the Thesis	3
1.2.1 Focus on Penetration Testing in Enterprise Environments	3
1.2.2 Comparative and Methodological Perspective	3
1.2.3 Attention to Automation and LLM-Assisted Analytical Support . .	4
1.2.4 Application to a Real Internship Experience	4
1.3 Main Contributions	4
1.3.1 Comparative Evaluation of Major Penetration Testing Frameworks	4
1.3.2 Definition of Comparison Criteria for Enterprise Applicability . . .	5
1.3.3 Discussion of the Role of Automation in Practical Penetration Testing	5
1.3.4 Analysis of LLM-Assisted Support for Result Interpretation and Attack-Path Reasoning	5
1.3.5 Discussion Through a Real Enterprise Penetration Testing Project .	5
1.4 Thesis Structure	6
1.4.1 Organization of the Chapters	6
2 Background and Foundations of Penetration Testing	8
2.1 Definition and Scope of Penetration Testing	8
2.1.1 Vulnerability Assessment vs Penetration Testing	9
2.1.2 Black-Box, White-Box, and Grey-Box Approaches	9

2.2	Main Phases of a Penetration Test	10
2.2.1	Pre-Engagement	10
2.2.2	Reconnaissance	11
2.2.3	Threat Modeling	12
2.2.4	Vulnerability Analysis	12
2.2.5	Exploitation	12
2.2.6	Post-Exploitation	13
2.2.7	Reporting	13
2.3	Role of Automated Tools in the Penetration Testing Lifecycle	13
2.3.1	Network Discovery and Mapping	14
2.3.2	OSINT and Information Gathering	14
2.3.3	Vulnerability Scanning	14
2.3.4	Exploitation Support	15
2.3.5	Reporting and Result Organization	15
2.4	Limits of Traditional Automation	15
2.4.1	Coverage Limitations	16
2.4.2	False Positives and False Negatives	17
2.4.3	Lack of Contextual Interpretation	17
2.4.4	Trade-Off Between Scalability and Analytical Depth	17
2.5	Penetration Testing in the Enterprise Security Context	18
2.5.1	Relationship with Enterprise Risk Management	18
2.5.2	Relevance in Complex Organizational Environments	18
2.5.3	Interaction with Governance and Security Processes	19
3	Research Problem, Objectives and Questions	20
3.1	Research Motivation	20
3.1.1	Need for Structured Enterprise-Oriented Penetration Testing	21
3.1.2	Limits of Isolated Methodological Frameworks	21
3.1.3	Need to Balance Automation Efficiency and Analytical Reliability	22
3.1.4	Potential Role of LLM-Assisted Support in Analyst Decision-Making	22
3.2	Problem Statement	22
3.2.1	No Single Framework Fully Addresses the Needs of Enterprise Penetration Testing	23
3.2.2	Automated Tools Improve Efficiency but May Produce Incomplete or Noisy Results	23
3.2.3	Analytical Support Is Needed to Interpret Findings and Reason on Attack Paths	24
3.2.4	LLM-Based Assistance May Be Useful, but Requires Human Supervision and Validation	24
3.3	Research Objectives	24

3.3.1	Compare Major Penetration Testing Frameworks Using Explicit Evaluation Criteria	25
3.3.2	Assess Their Suitability for Enterprise Environments	25
3.3.3	Analyze the Contribution and Limitations of Automated Tools in Practical Testing	25
3.3.4	Examine the Role of LLM-Assisted Support in Result Validation and Attack-Path Reasoning	26
3.3.5	Define and Discuss a Hybrid Methodological Approach	26
3.3.6	Discuss Its Practical Relevance Through a Real Enterprise Application	26
3.4	Research Questions	27
3.4.1	Can a Single Penetration Testing Framework Adequately Support Enterprise Testing Activities?	27
3.4.2	Which Criteria Are Most Relevant for Comparing Penetration Testing Frameworks in Enterprise Contexts?	27
3.4.3	How Do Automated Tools Affect the Quality and Efficiency of Penetration Testing?	28
3.4.4	In Which Ways Can LLM-Assisted Support Help Analysts in Interpreting Findings and Exploring Attack Paths?	28
3.4.5	What Benefits Can Be Derived from a Hybrid Approach Combining Frameworks, Automation, LLM-Assisted Support, and Human Validation?	28
4	Comparison Model for Penetration Testing Frameworks	29
4.1	Need for a Structured Comparison Model	29
4.1.1	Limits of Purely Descriptive Comparisons	30
4.1.2	Need for Reusable and Explicit Evaluation Criteria	30
4.2	Definition of Comparison Criteria	31
4.2.1	Coverage Scope	31
4.2.2	Operational Depth	32
4.2.3	Level of Formalization	32
4.2.4	Enterprise Suitability	33
4.2.5	Support for Automation	33
4.2.6	Reporting and Validation Support	34
4.2.7	Adaptability to Hybrid Workflows	34
4.3	Use of the Comparison Model in the Thesis	35
4.3.1	How the Criteria Are Applied	36
4.3.2	How the Model Supports the Research Objectives	37
4.3.3	Methodological Limits of the Comparison	38
5	Comparative Analysis of Major Penetration Testing Frameworks	39
5.1	PTES – Operational Standardization	39

5.1.1	General Structure	40
5.1.2	Main Strengths	40
5.1.3	Main Limitations	41
5.1.4	Evaluation According to the Comparison Criteria	41
5.2	OSSTMM – Quantitative Security Testing	42
5.2.1	General Structure	43
5.2.2	Main Strengths	43
5.2.3	Main Limitations	44
5.2.4	Evaluation According to the Comparison Criteria	44
5.3	NIST SP 800-115 – Institutional and Procedural Perspective	45
5.3.1	General Structure	46
5.3.2	Main Strengths	46
5.3.3	Main Limitations	47
5.3.4	Evaluation According to the Comparison Criteria	47
5.4	OWASP Testing Guide – Application Security Perspective	48
5.4.1	General Structure	49
5.4.2	Main Strengths	49
5.4.3	Main Limitations	50
5.4.4	Evaluation According to the Comparison Criteria	50
5.5	MITRE ATT&CK – Adversarial Knowledge and Attack Modeling	51
5.5.1	General Structure	52
5.5.2	Main Strengths	52
5.5.3	Main Limitations	53
5.5.4	Evaluation According to the Comparison Criteria	53
5.6	Comparative Discussion	55
5.6.1	Key Differences Among Frameworks	55
5.6.2	Complementary Aspects	56
5.6.3	Relevance for Enterprise Penetration Testing	58
6	Automation, LLM-Assisted Support and Hybrid Methodological Approaches	59
6.1	Role of Automation in Enterprise Penetration Testing	59
6.1.1	Operational Advantages	60
6.1.2	Scalability and Efficiency	60
6.1.3	Common Limitations in Practice	61
6.2	Automated Tools in Practice	61
6.2.1	Network Scanning and Mapping Tools	62
6.2.2	OSINT Tools	62
6.2.3	Vulnerability Scanners	63

6.2.4	Automated Validation and Attack Simulation Platforms	63
6.3	LLM-Assisted Support in Penetration Testing Activities	64
6.3.1	Support for Result Interpretation	64
6.3.2	Assistance in Attack-Path Reasoning	65
6.3.3	Help in Correlating Findings	65
6.3.4	Use in Preliminary Validation and Analytical Triage	66
6.4	Limits of LLM-Assisted Support	66
6.4.1	Hallucinations and Inaccurate Inferences	67
6.4.2	Lack of Direct Environmental Awareness	67
6.4.3	Dependence on Prompt Quality and Analyst Guidance	68
6.4.4	Need for Expert Validation	68
6.5	Human Validation as a Critical Component	69
6.5.1	Contextual Interpretation	69
6.5.2	Technical Verification	70
6.5.3	Risk Prioritization	70
6.5.4	Final Assessment Responsibility	71
6.6	Toward a Hybrid Methodological Approach	71
6.6.1	Integration of Frameworks, Automation, and Analyst Reasoning . .	72
6.6.2	Possible Role of LLM Support in the Workflow	73
6.6.3	Expected Benefits in Enterprise Environments	74
7	Enterprise Penetration Testing Application	75
7.1	Context and Scope of the Project	76
7.1.1	Consulting Context of the Internship	76
7.1.2	Enterprise Environment Under Assessment	77
7.1.3	External, Internal, and Physical Security Perimeter	77
7.1.4	Objectives and Boundaries of the Activity	78
7.2	Methodological Organization of the Activity	79
7.2.1	Phases Followed During the Project	79
7.2.2	Retrospective Interpretation of the Workflow	80
7.2.3	Relation with the Frameworks Discussed in the Thesis	81
7.3	Tools Used in the Activity	82
7.3.1	Nmap	83
7.3.2	Maltego	84
7.3.3	Nessus	85
7.3.4	Pentera	85
7.3.5	Other Supporting Tools	87
7.4	Use of LLM-Assisted Support	88
7.4.1	Support in Reviewing Tool-Generated Findings	89

7.4.2	Support in Reasoning About Possible Attack Paths	89
7.4.3	Support in Correlating Technical Evidence	90
7.4.4	Observed Strengths and Limitations in Practice	90
7.5	Main Findings	91
7.5.1	Relevant Vulnerabilities Identified	91
7.5.2	Issues Emerging from External, Internal, and Physical Assessment	92
7.5.3	False Positives and Validation Issues	93
7.5.4	Role of Automated Tools in the Findings Generation Process	94
7.6	Observed Benefits and Practical Limits	94
7.6.1	Observed Benefits of a Structured Methodological Perspective	95
7.6.2	Observed Benefits and Limits of Automation	95
7.6.3	Observed Benefits and Limits of LLM-Assisted Analytical Support	96
7.6.4	Role of Human Oversight in Final Validation	97
7.7	Limits of the Application	98
7.7.1	Constraints Related to the Internship Project	98
7.7.2	Activities Not Fully Completed or Not Generalizable	99
7.7.3	Limits in Evaluating Broader Applicability	99
8	Discussion and Conclusions	101
8.1	Discussion of Results	101
8.1.1	Interpretation of the Framework Comparison	102
8.1.2	Interpretation of the Practical Application	103
8.1.3	Coherence with the Research Objectives	104
8.2	Answers to the Research Questions	105
8.2.1	Synthesis of the Answers	105
8.3	Main Contributions of the Thesis	107
8.3.1	Methodological Contributions	107
8.3.2	Practical Contributions	107
8.4	Limitations of the Study	108
8.4.1	Scope Limitations	108
8.4.2	Methodological Limitations	109
8.5	Future Developments	110
8.5.1	Extension of the Comparison Model	110
8.5.2	Further Validation in Additional Enterprise Contexts	111
8.5.3	More Systematic Study of LLM-Assisted Support	111
	Bibliography	113

List of Figures

2.1	Main phases of the penetration testing process.	11
6.1	Hybrid methodological approach integrating frameworks, automated tools, LLM-assisted support, and human validation in enterprise penetration testing.	72
7.1	Reconstructed example of service enumeration output based on Nmap-style results. The figure is included for explanatory purposes only and does not reproduce customer data.	83
7.2	Illustrative example of OSINT-based relationship mapping used to support external exposure analysis. The figure does not reproduce original customer data.	84
7.3	Illustrative example of a vulnerability scanning dashboard used to support exposure identification. Sensitive and customer-specific information is omitted.	86
7.4	Illustrative example of automated exposure validation and attack-path analysis. The figure does not reproduce original customer data.	87

Acronyms

AI

Artificial Intelligence

API

Application Programming Interface

ATT&CK

Adversarial Tactics, Techniques, and Common Knowledge

LLM

Large Language Model

NIST

National Institute of Standards and Technology

OSINT

Open-Source Intelligence

OSSTMM

Open Source Security Testing Methodology Manual

OWASP

Open Worldwide Application Security Project

PTES

Penetration Testing Execution Standard

TTP

Tactics, Techniques, and Procedures

WSTG

Web Security Testing Guide

ASN

Autonomous System Number

DNS

Domain Name System

DoS

Denial of Service

FTP

File Transfer Protocol

FTPS

File Transfer Protocol Secure

IP

Internet Protocol

MITM

Man-in-the-Middle

SIEM

Security Information and Event Management

SMB

Server Message Block

SOC

Security Operations Center

AD

Active Directory

Chapter 1

Introduction

Penetration testing has become an established component of modern cybersecurity assessments. In enterprise environments, where systems are heterogeneous, interconnected, and operationally complex, security evaluation needs methods that go beyond simple technical checks. In this context, penetration testing is important not only for identifying weaknesses, but also for providing a clearer understanding of how such weaknesses may be exploited in practice.

At the same time, penetration testing is influenced by changes in methods and technology. Automated tools are now integrated into assessment processes, and large language models are beginning to be considered as possible tools for analytical support. These developments raise questions about how frameworks, automation, and human judgment should work together in enterprise-oriented security assessments. This thesis explores these questions through a comparative and practical discussion based on a real internship experience.

1.1 Context and Motivation

The role of penetration testing in enterprise cybersecurity is linked to the changes in today's digital environments. The complexity of infrastructure, evolving threat scenarios, and the increased use of automation have altered the needs and expectations for security assessments.

1.1.1 Increasing Complexity of Enterprise Infrastructures

Contemporary enterprise infrastructures include on-premise systems, cloud services, distributed access models, third-party integrations, and heterogeneous software and hardware components. In enterprise environments, these elements cannot be considered in isolation. They form an interconnected ecosystem in which security exposure may emerge not only from individual vulnerabilities but also from interactions among multiple systems.

Such complexity makes penetration testing significantly more demanding. In simpler and well-defined settings, vulnerability assessments can be conducted with a focus on

a limited number of technical elements or attack vectors. In enterprise environments, however, the broader attack surface and the dependencies among components require a more structured approach to planning and conducting the assessment. As a result, besides identifying weaknesses, the understanding of their place in the overall context becomes equally important.

1.1.2 Growing Sophistication of Cyber Threats

The evolution of cyber threats has increased the importance of realistic security assessment. Contemporary attackers use multi-phase attack strategies that include reconnaissance, credential harvesting, exploiting vulnerabilities, privilege escalation, and lateral movement. As a result, security risks cannot be analyzed solely by conducting vulnerability assessments since actual risk usually depends on interactions between multiple weaknesses within a broader attack path.

In this respect, penetration testing remains relevant because it is oriented toward exploitability rather than the mere identification of vulnerabilities. It allows analysts to move beyond simple scanning activities and to reason in terms of attack feasibility and potential impact on the target environment.

1.1.3 Need for Structured Penetration Testing Methodologies

As penetration testing becomes increasingly relevant in modern environments, the need for well-established methodologies to guide the process becomes more important. Testing operations depend on scope restrictions, terms of engagement, reporting requirements, and other factors. In the absence of a clear methodological reference, the success of the assessment will strongly depend on personal preferences and actions of the testers throughout the engagement.

A methodology addresses this problem by defining phases, principles, and expected results for penetration testing. It promotes uniformity, improves traceability, and offers a common point of view for discussion regarding scope and outcomes. Moreover, each framework differs in its underlying assumptions and focus area. Methodologies can be either operational or formalized; some focus on certain areas, while others emphasize particular views on testing. Thus, comparing different methodologies is particularly important in enterprise environments.

1.1.4 Expanding Role of Automation in Security Assessment

Automation has become an essential component of modern penetration testing. In enterprise environments, automation supports tasks such as network reconnaissance, service scanning, vulnerability identification, and information gathering. Analysts can perform these tasks more efficiently when relying on automation rather than purely manual methods.

Nevertheless, the value of automation should be considered critically. Automation results may be affected by gaps in testing, false positives, and a lack of context. Although automation makes the process faster, it does not replace the need to validate and interpret

the results. For this reason, automation should be regarded not as a substitute for analysis, but as one component of a broader assessment process.

1.1.5 Emergence of AI and LLM-Assisted Support in Cybersecurity Analysis

Recent advances in artificial intelligence have introduced new possibilities for supporting analytical tasks in cybersecurity. In penetration testing, the potential value of these models lies mainly in tasks such as interpreting findings, correlating technical evidence, and supporting reasoning on possible attack paths.

However, such a perspective requires some caution. A key limitation of large language models is that they lack direct awareness of the target environment; as a result, they cannot independently verify technical conditions. While their conclusions might be considered analytical insights that can be used by penetration testers, they cannot replace technical verification.

1.2 Aim and Scope of the Thesis

The thesis examines penetration testing from both methodological and practical perspectives, with specific reference to enterprise environments. The intention is not to establish a completely new approach to conducting penetration tests. Instead, the goal is to examine how existing frameworks, tools, and LLM-assisted analytical support can be discussed and related within a structured perspective on enterprise penetration testing.

1.2.1 Focus on Penetration Testing in Enterprise Environments

Enterprise penetration testing represents the primary focus of this study. This choice is motivated by the specific nature of enterprise environments, where technical complexity, operational constraints, and risk governance intersect. Consequently, this work does not aim to provide an exhaustive treatment of all possible offensive techniques, nor is it restricted to a single specialized domain such as application security. Instead, it focuses on the methodological perspective required to organize and interpret large-scale penetration testing engagements.

1.2.2 Comparative and Methodological Perspective

A central feature of the thesis is its comparative nature. The penetration testing methodologies discussed in this thesis are not evaluated as isolated frameworks, but through a comparative analysis based on explicit criteria. This enables assessment of their relevance, limitations, and potential complementarity in enterprise environments.

The thesis adopts a broader methodological perspective that treats penetration testing as more than a purely technical toolset. It involves decisions concerning organization, evidence management, validation, and reasoning during the assessment process. This

perspective provides the basis for connecting the discussion of frameworks with tool usage and analyst reasoning.

1.2.3 Attention to Automation and LLM-Assisted Analytical Support

Particular attention is also given to the role of automation in practical penetration testing. Automation tools are considered essential elements in modern testing procedures, particularly under conditions where effectiveness and scalability are paramount. The contribution of automated tools will be examined in terms of both their operational value and limitations, with a focus on how their findings can be used in real-world settings.

Moreover, the thesis also examines the possible role of LLM-assisted analytical support in penetration testing. In doing so, however, this issue will be treated cautiously and limited to interpreting results, initial validation, and inferring likely attack vectors. The objective here is not to consider LLMs as autonomous testing tools but to assess whether they might provide useful support within a larger process.

1.2.4 Application to a Real Internship Experience

The practical component of the thesis is based on an internship carried out at KPMG, during which penetration testing activities were performed in a complex enterprise environment. The activity involved the assessment of external, internal, and physical security dimensions through the use of tools commonly adopted in professional practice.

The main function of this practical component is to ground the methodological and comparative discussion of the thesis in a real operational context.

1.3 Main Contributions

The main contribution of the thesis to the field of enterprise penetration testing lies in its synthesis of these aspects into a coherent analytical model rather than examining them individually. Within this broader perspective, the thesis develops several more specific contributions.

1.3.1 Comparative Evaluation of Major Penetration Testing Frameworks

A first contribution lies in the comparative evaluation of major penetration testing frameworks. Instead of presenting them as isolated methodological references, the thesis discusses them in relation to one another and in light of their practical relevance for enterprise-oriented security assessments. This approach supports a more reasoned understanding of how different frameworks emphasize different phases, priorities, and assumptions.

The comparative analysis is particularly useful because enterprise penetration testing rarely fits entirely within a single rigid methodological model. By examining frameworks

comparatively, the thesis highlights both their individual value and the limits that emerge when they are applied to complex organizational settings.

1.3.2 Definition of Comparison Criteria for Enterprise Applicability

A second contribution is the definition of a set of explicit comparison criteria to assess the applicability of penetration testing frameworks in enterprise settings. These criteria are not presented as universal or definitive metrics, but as a structured analytical model developed for the purposes of the thesis.

Their role is to provide a clearer methodological basis for the comparison. By making the evaluation criteria explicit, the thesis avoids a purely intuitive discussion and adopts a more transparent and consistent approach to framework analysis.

1.3.3 Discussion of the Role of Automation in Practical Penetration Testing

A further contribution lies in the contextualization of automation within contemporary penetration testing practice. Automated tools are particularly relevant for supporting activities such as host discovery, network mapping, preliminary vulnerability detection, and triage. However, the thesis also addresses their limitations, showing that automated outputs must be interpreted, validated, and integrated into a broader analytical workflow rather than treated as conclusive results.

1.3.4 Analysis of LLM-Assisted Support for Result Interpretation and Attack-Path Reasoning

The thesis also provides a critical discussion of LLM-assisted support in penetration testing workflows. More specifically, it examines whether and how large language models can assist analysts with tasks such as interpreting technical findings, correlating information across multiple tools, organizing preliminary hypotheses, and reasoning about possible attack paths.

This contribution is relevant because the role of large language models in penetration testing remains uncertain and is often described in overly optimistic or overly skeptical terms. The thesis adopts a more cautious position, recognizing their possible value in analytical reasoning while also stressing their lack of direct environmental awareness and the need for technical verification.

1.3.5 Discussion Through a Real Enterprise Penetration Testing Project

A final contribution lies in the connection between the methodological discussion developed in the thesis and a real enterprise penetration testing project conducted during an internship

at KPMG. This practical component gives the work an applied dimension that helps assess the relevance of the broader discussion against an actual professional context.

Although the practical application does not claim broad experimental generalization, it strengthens the thesis by showing how the issues discussed at the conceptual and methodological level emerge in an actual engagement. In this sense, the practical component is valuable not only as an illustrative example, but also as a context in which the interaction between frameworks, automation, LLM-assisted support, and human validation can be examined more concretely.

1.4 Thesis Structure

The thesis is organized into eight chapters that follow a logical progression from the general setting and theoretical background of the issue to the comparative analysis, practical implementation, and conclusions.

Chapter 2: Background and Foundations of Penetration Testing Covers the foundations of penetration testing, analyzing its lifecycle, the role of traditional automation, and its significance within enterprise security and risk management.

Chapter 3: Research Problem, Objectives and Questions Formalizes the central problem statement, research goals, and the core questions guiding this study.

Chapter 4: Comparison Model for Penetration Testing Frameworks Defines the qualitative evaluation model and the seven structural criteria used to assess the methodologies.

Chapter 5: Comparative Analysis of Major Penetration Testing Frameworks Applies the model to major testing frameworks, highlighting their specific areas of focus and complementarity.

Chapter 6: Automation, LLM-Assisted Support and Hybrid Methodological Approaches Discusses the operational interplay between automation, large language models, and human-led validation.

Chapter 7: Enterprise Penetration Testing Application Presents the empirical case study conducted during the internship at KPMG, spanning external, internal, and physical perimeters.

Chapter 8: Discussion and Conclusions Reconsiders the research questions, outlines the contributions and limitations of the study, and proposes future research avenues.

1.4.1 Organization of the Chapters

The structure of the thesis is intended to preserve coherence between its theoretical, methodological, and applied dimensions. The first chapters establish the conceptual and analytical basis of the work; the central chapters develop the comparative and

methodological discussion; and the final chapters connect these elements to practical experience and to the concluding interpretation of the results.

Chapter 2

Background and Foundations of Penetration Testing

Before addressing the comparative and methodological questions that guide this thesis, it is necessary to establish the conceptual and operational foundations of penetration testing. In enterprise environments, penetration testing cannot be understood merely as a technical activity aimed at identifying vulnerabilities. Its broader significance also depends on how the assessment is structured, which phases it includes, how evidence is interpreted, and how technical findings are related to organizational risk considerations.

To this end, this chapter introduces the essential components required to frame the remainder of the work. The discussion begins by delineating the boundaries of penetration testing, distinguishing it from related security assessment activities, and outlining the main testing approaches. It then examines the typical phases of an engagement, the role of automated tools throughout the assessment lifecycle, and the main limitations that arise when automation is used without sufficient contextual interpretation. Finally, the chapter places penetration testing within the broader enterprise security context, highlighting its relationship with risk management, governance, and remediation processes. Taken together, these elements provide the theoretical and practical basis needed for the research problem and the framework comparison developed in the following chapters.

2.1 Definition and Scope of Penetration Testing

Penetration testing is a security assessment activity used to evaluate whether vulnerabilities in a target environment can be exploited in practice. Unlike broader security reviews, which may focus on compliance, configuration issues, or the identification of known weaknesses, penetration testing is concerned with the validation of exploitability under controlled conditions [1, 2]. Its purpose is therefore not limited to listing technical flaws, but extends to assessing their actual security relevance in terms of feasibility, possible attack paths, and potential impact [1, 2].

Penetration testing, therefore, has a specific role within cybersecurity practice. It

combines technical analysis, adversarial reasoning, and validation to produce results closer to realistic attack scenarios [1, 2]. Although the scope of an engagement may vary according to its objectives, rules of engagement, available information, and target type, the defining characteristic of penetration testing remains the controlled simulation of offensive actions for defensive and evaluative purposes.

2.1.1 Vulnerability Assessment vs Penetration Testing

Vulnerability assessment and penetration testing, though closely related, are not equivalent [1, 2]. A vulnerability assessment aims to identify weaknesses, misconfigurations, outdated software, and other exposure conditions that may affect the target environment. It therefore focuses primarily on detection and cataloguing, often relying on scanners, checklists, and structured review procedures [1]. In this sense, it provides visibility over potential exposure, but does not necessarily determine whether the identified issues are exploitable in practice [1, 2].

Penetration testing goes further by attempting to validate whether selected weaknesses can be exploited under controlled conditions. Its objective is to determine whether an attacker could compromise confidentiality, integrity, or availability by leveraging the identified exposure [1, 2]. This distinction is important because an identified weakness may not provide a realistic path to compromise in the specific environment, while a finding that appears limited in isolation may become significant when combined with other vulnerabilities within a broader attack path.

In summary, the key difference between the two activities lies in their depth and purpose. Vulnerability assessment usually provides broader coverage, while penetration testing provides deeper validation of selected weaknesses. The two activities can therefore be complementary: vulnerability assessment can help identify potential areas of exposure, while penetration testing can validate their practical relevance and support more accurate prioritization [1, 2].

2.1.2 Black-Box, White-Box, and Grey-Box Approaches

Penetration testing can be performed under varying assumptions concerning the information available to the tester. These assumptions are typically categorized into three main models: black-box, white-box, and grey-box testing [3, 1]. The distinction is relevant because the amount of prior knowledge available during the assessment significantly affects both the methodology and the interpretation of the results.

In black-box penetration testing, the tester has little to no knowledge of the internal environment. An individual performing a black-box test is in a similar position to an outside attacker. This test is often useful for assessing exposure to the outside world and analyzing the systems from an external perspective which can be extremely valuable during reconnaissance and external attack simulation [3]. Black-box testing provides a good insight into discoverability and external attack feasibility, but can also take longer and cover a narrower scope due to the constraints of an engagement.

In a white-box approach, the tester has extensive knowledge of the target, including

the environment in which testing will take place. The tester may have access to network diagrams, architecture information, source code, usernames/passwords, and other documentation. A white-box test should provide full coverage of an environment, and it is also useful for revealing weaknesses that may not be visible from an external perspective alone [3, 1]. White-box testing is less focused on discoverability and more focused on analytical depth, making it particularly suitable for internal assessments.

Grey-box penetration testing can be seen as an intermediate condition between white-box and black-box testing. It assumes that the tester has partial knowledge of the target environment. In many enterprise engagements, this model provides a practical balance between realism and operational efficiency [3]. By reducing some of the uncertainty typical of black-box testing while preserving an adversarial perspective, grey-box testing can support more focused and effective assessment activities.

These approaches describe different testing conditions, but they are not always mutually exclusive in practice. Their main value lies in clarifying the assumptions under which the penetration test is conducted and in helping define the scope, depth, and interpretation of the results.

2.2 Main Phases of a Penetration Test

Penetration test methodologies vary by terminology, formality, and objective, but for the most part penetration testing can be characterised in a series of repeating phases [2, 1]. These stages do not always follow each other linearly, since practical assessments often involve iteration, feedback, and partial revisiting of earlier steps. However, familiarity with these phases is essential for understanding how the assessment is structured, how evidence is gathered and verified, and how technical results are translated into a report [2, 1].

The value of this phased perspective lies not only in operational clarity, but also in methodological consistency. Dividing the penetration test into phases helps structure the assessment as a process in which planning, exploration, validation, and interpretation are all essential. This is especially important in enterprise settings, where the quality of the assessment depends not only on technical execution, but also on the ability to maintain coherence between objectives, evidence, and final reporting.

The main phases commonly involved in a penetration testing process are summarized in Figure 2.1.

2.2.1 Pre-Engagement

The pre-engagement phase specifies the environment in which the penetration test will be executed. This includes defining the assessment scope, identifying the assets or environments to be tested, clarifying objectives, and specifying constraints, assumptions, and rules of engagement [2, 1]. This phase is particularly important in the context of a professional assessment as it determines the legal, operational, and methodological boundaries of the activity.

A clearly defined pre-engagement phase helps minimize ambiguity and supports the

Main Phases of a Penetration Testing Process

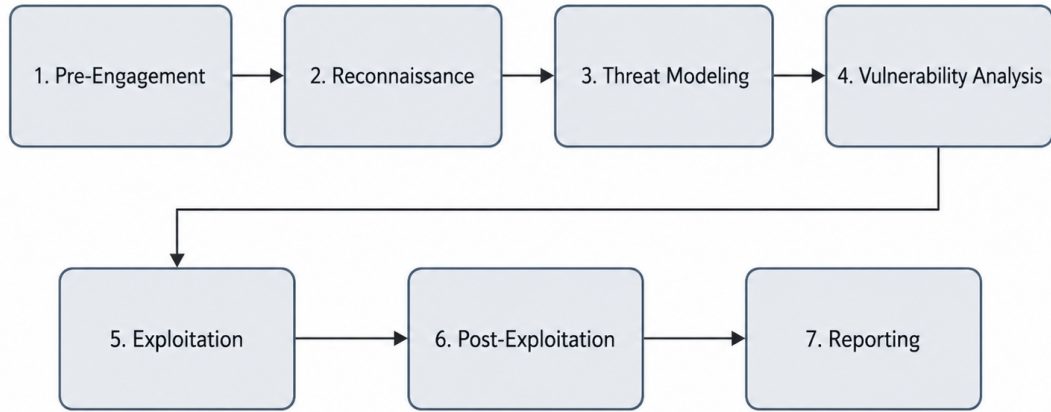


Figure 2.1: Main phases of the penetration testing process.

management of stakeholders' expectations. It clarifies what is permitted and what is excluded, how far exploitation may proceed, and under which circumstances testing should be stopped, or limited [2]. In an enterprise context, where business-critical systems may be involved, and several stakeholders may participate in defining and controlling the assessment, it is essential that the test provides value without disrupting operations.

2.2.2 Reconnaissance

Reconnaissance consists of collecting information about the target environment to support later analysis and exploitation [2, 1]. Depending on the type of engagement, this phase may include external observation, service enumeration, network discovery, public information gathering, and the identification of potentially relevant assets, technologies, and entry points.

This phase is particularly important because the quality of later testing activities often depends on the quality of the information collected here. Reconnaissance allows the tester to reduce uncertainty, identify possible exposure points, and form an initial view of the environment under assessment [2]. In black-box or grey-box contexts, it also plays a central role in approximating the perspective of an attacker who begins with limited knowledge of

the target.

2.2.3 Threat Modeling

Threat modeling supports the identification of realistic attack scenarios by relating the observed characteristics of the target environment to the possible behavior of an adversary [2]. Rather than focusing only on isolated technical weaknesses, this phase helps frame the assessment in terms of feasible attack paths, relevant attacker goals, and plausible exploitation sequences.

In penetration testing, threat modeling does not necessarily take the form of a formal standalone procedure, but it is often present as an analytical activity that guides prioritization and decision-making. It helps determine which assets are most relevant, which vulnerabilities deserve closer validation, and which combinations of conditions may lead to meaningful compromise scenarios [2]. This phase is especially relevant in enterprise environments, where the number of potential findings may be large and not all of them have the same strategic importance.

2.2.4 Vulnerability Analysis

Vulnerability analysis aims to identify technical weaknesses, insecure configurations, exposed services, and other conditions that may create security risk [2, 1]. This phase may involve manual inspection, automated scanning, configuration review, and the interpretation of evidence collected during reconnaissance. Its objective is not only to detect weaknesses, but also to assess their plausibility and relevance within the specific context of the engagement.

In practice, vulnerability analysis often marks the transition from observation to active validation. It is the stage in which the tester moves from collecting information to formulating concrete hypotheses about possible exploitation opportunities [2]. In complex environments, this phase also requires a degree of judgment, since not every detected issue deserves the same attention and not every technically plausible weakness will lead to a relevant attack path.

2.2.5 Exploitation

The exploitation phase consists of the controlled attempt to determine whether identified weaknesses can be effectively leveraged to gain unauthorized access, elevate privileges, or otherwise compromise the target environment [2, 1]. Exploitation is a defining characteristic of penetration testing, moving beyond the detection layer. It provides empirical confirmation of a vulnerability's severity, distinguishing the engagement from assessment models that restrict their scope to mere weakness identification.

Crucially, exploitation within a professional context is never executed for its own sake; rather, it serves as a strict diagnostic mechanism calibrated to measure systemic resilience. It is conducted under explicit constraints and should remain proportionate to the objectives of the engagement. The purpose is to validate risk, not to maximize disruption [1]. For

this reason, exploitation must be interpreted as a controlled validation mechanism through which the tester determines whether a vulnerability has real security significance in the environment under assessment.

2.2.6 Post-Exploitation

Post-exploitation investigates the consequences of a successful compromise [2]. After a successful compromise, the tester can analyze which further actions may be possible, such as privilege escalation, lateral movement, persistence, access to sensitive data, or interaction with critical services. This phase is important because the impact of a compromise often depends less on the initial entry point itself and more on what the attacker can achieve after gaining access.

In an enterprise network, this phase is particularly significant because architectural complexity, internal trust relationships, and system connectivity can amplify the consequences of what appear to be minor weaknesses. Consequently, the post-exploitation phase contributes to a more realistic understanding of risk by moving beyond initial access and considering the downstream implications of compromise across the corporate infrastructure.

2.2.7 Reporting

Reporting formalizes the results of the penetration test by translating results, supporting evidence, impact, and remediation advice into a structured form [2, 1]. Reporting should not be viewed as an administrative task. This is the phase in which technical actions are converted into meaningful knowledge that will be used by executives, security, and other stakeholders.

The quality of a penetration test depends not only on what was technically discovered, but also on how clearly the results are communicated. Effective reporting should distinguish validated findings from preliminary observations, explain their relevance to organizational risk, and connect conclusions to supporting evidence [1]. In an enterprise setting, reporting is also essential for remediation planning and risk management, and therefore represents a central phase of the engagement.

2.3 Role of Automated Tools in the Penetration Testing Lifecycle

Automated tools have become a critical part of contemporary penetration testing engagements [1, 3]. Their relevance derives largely from the scale, speed, and heterogeneity of contemporary systems, which make entirely manual review unfeasible in many corporate settings. Automation does not replace the need for analyst judgment, nor can a security assessment be fully automated. Rather, specialized tools augment human capabilities by supporting processes that would otherwise be too labor-intensive. Within the penetration testing lifecycle, automation primarily serves an enabling role: it improves efficiency, enhances visibility, and streamlines the organization of technical findings.

Automation is not a standalone solution. An automated tool is effective only when it is properly integrated into the assessment, when its results are reviewed critically, and when its limitations are addressed through human validation.

2.3.1 Network Discovery and Mapping

One of the most common uses of automation in penetration testing concerns the discovery and mapping of the target environment. Automation tools can discover hosts, enumerate services, find open ports, and provide a first depiction of the surface under assessment [1]. This role is particularly useful in the initial phase of the engagement, when the tester has to decrease the uncertainty and gain a technical overview of the system.

In enterprise contexts, where infrastructures may be extensive and distributed, manual discovery alone would often be too slow or incomplete. Automated mapping allows the analyst to identify areas of interest more efficiently and to build an initial inventory of relevant systems and services. This does not eliminate the need for interpretation, but it provides an operational baseline from which more focused testing activities can proceed.

2.3.2 OSINT and Information Gathering

Automation also helps in open-source intelligence and wider reconnaissance. Publicly available data may reveal domains, subdomains, exposed services, technologies, employee information, infrastructure relationships, or other details that contribute to the attack surface [3]. Automated OSINT tools allow analysts to gather, correlate, and organize open-source intelligence much faster than relying exclusively on manual searches.

This capability is particularly useful in black-box or external assessment scenarios, where initial knowledge of the target may be limited. By supporting structured information gathering, automation helps the tester identify potentially relevant entry points and contextual clues. At the same time, these tools do not interpret the significance of the collected information on their own. Their value depends on the analyst's ability to distinguish relevant evidence from noise and to connect publicly available information to realistic attack opportunities.

2.3.3 Vulnerability Scanning

One of the most well-established uses of automation in the security assessment phase is in vulnerability scanning [1, 3]. Automated vulnerability scanners can report known vulnerabilities, insecure configurations, outdated software versions, and other potential exposure conditions. These tools allow testers to obtain broad technical coverage in a relatively short time, which is particularly valuable when large numbers of assets are involved.

Vulnerability scanning, however, should not be viewed as straightforward. Automated scanners provide indications rather than conclusions, and their outputs often require confirmation. Some findings may be false positives, others may be technically correct but not exploitable in the specific environment, and others may fail to reflect the actual

business relevance of the affected asset. For this reason, vulnerability scanning is best understood as a support mechanism for further analysis rather than as a final assessment in itself.

2.3.4 Exploitation Support

Some automated or semi-automated platforms can also support exploitation-related activities [2]. These tools may assist in validating findings, simulating attack steps, or guiding the execution of exploit sequences under controlled conditions. This does not necessarily mean that the tool performs exploitation autonomously. Rather, it can support the tester in determining whether a reported weakness is practically exploitable under the engagement's conditions.

This support is particularly valuable when the assessment involves systematically verifying several findings and when the simulation of attack paths is aligned with the engagement goals. However, exploitation support tools are still limited by the validity of their assumptions and the reliability of their interpretation. Their results should therefore be viewed as elements of the validation process and not as conclusive evidence.

2.3.5 Reporting and Result Organization

Automation can also assist with the organization of findings and the production of intermediate data to be used during reporting [1]. Many tools, including scanners, attack simulators, and assessment utilities, generate structured technical data that can support documentation, evidence tracking, and traceable results. This can be particularly useful in large engagements with many findings, where consistency and organization are important for producing technical deliverables.

Even in this area, however, automation remains supportive rather than sufficient. A penetration test report is not merely a collection of tool outputs. It requires selectivity, appropriate context, and translating raw technical data into insights relevant to the intended audience. Therefore, while the organization of results can benefit from automation, the overall reporting stage still relies heavily on the judgment and communication skills of the analyst.

The role of automation across the main phases of penetration testing can be summarized as shown in Table 2.1.

2.4 Limits of Traditional Automation

Although automation is indispensable in modern penetration testing, it represents only one component of the assessment's overall value [1, 3]. The use of automated tools can improve speed, coverage, and repeatability. However, these tools do not provide a complete or standalone view of the target's security posture. Their outputs are shaped by predefined logic, available signatures, technical assumptions, and the specific data they can collect.

Table 2.1: Role of automated tools across the penetration testing lifecycle.

Phase	Automated support	Main benefit	Main limitation
Reconnaissance	Network discovery, service enumeration, information gathering	Faster collection of technical information and broader initial visibility	Limited contextual understanding of the collected data
Vulnerability analysis	Vulnerability scanning, configuration checks, identification of known weaknesses	Broad technical coverage in a relatively short time	False positives, false negatives, and limited relevance assessment
Exploitation support	Guided validation, attack-path simulation, semi-automated exploitation workflows	More efficient validation of selected findings	Dependence on pre-defined assumptions and need for analyst verification
Reporting and result organization	Structured output generation, documentation support, evidence aggregation	Improved traceability and consistency of intermediate technical results	Inability to replace contextual explanation and prioritization

For this reason, traditional automation is highly valuable as a support mechanism, but insufficient as a standalone basis for penetration testing conclusions.

This distinction is particularly relevant in enterprise settings, where network segmentation, business continuity, operational constraints, and architectural interdependencies influence the actual severity of a technical exposure. A tool may report a theoretically critical flaw while overlooking more subtle weaknesses that emerge only through the interaction of multiple conditions. Recognizing these boundaries is therefore essential to prevent overreliance on tool-generated findings and to preserve the central role of expert human interpretation.

2.4.1 Coverage Limitations

Automated tools can provide useful high-level technical visibility, but their coverage is never complete. Their ability to identify weaknesses depends on what they are designed to inspect, the depth of access they have, and the assumptions built into their detection logic [1]. Some risks may go undetected because they fall outside the tool’s expected behavior, depend on particular environmental states, or are only observable as a result of interactions between several systems.

This limitation is particularly relevant in environments with complex architectures, custom logic, fragmented infrastructure, or non-standard configurations. In these situations,

automation can provide valuable insight, but it cannot be assumed to cover every aspect of the architecture. Analyst-driven exploration remains necessary whenever the environment includes conditions that are difficult to capture through predefined technical checks.

2.4.2 False Positives and False Negatives

One of the most common limitations of automated processes is the presence of both false positives and false negatives [1]. A false positive occurs when an automated system reports a technology or configuration as vulnerable even though it is not actually exploitable in the specific target environment. A false negative occurs when an automated process misses a potentially significant vulnerability. Both cases reduce the assessment's accuracy, though in different ways.

False positives may lead the analyst toward unproductive validation efforts, while false negatives may create a false sense of reassurance. In penetration testing, this distinction is particularly important because the value of the assessment depends on prioritization and validation. Automated results therefore need to be reviewed not only to determine their technical accuracy, but also to evaluate their practical relevance within the specific engagement context.

2.4.3 Lack of Contextual Interpretation

Traditional automation generally lacks contextual understanding [3]. It may detect technical conditions, signatures, or deviations from expected configurations, but it cannot evaluate these elements in relation to the organization's priorities, business consequences, attacker objectives, or overall target architecture. As a result, an automated system can identify potential problems, but not how important those problems are.

This limitation becomes especially relevant when the importance of a finding depends on its position in the environment, its combination with other conditions, or its potential operational consequences. The real security relevance of a weakness often depends on context, and this kind of interpretation remains largely outside the reach of traditional automated tools.

2.4.4 Trade-Off Between Scalability and Analytical Depth

A further limitation of automation lies in the trade-off between scalability and analytical depth. An automated tool is effective because it is able to work on large numbers of assets, configurations, and/or observations in a reasonably short period. Such a tool relies on standardized logic and general technical heuristics, which can sacrifice the thoroughness of an analysis on each individual occurrence [1, 3].

In practice, automation performs best in breadth-oriented tasks such as discovery, enumeration, and initial detection, while it is less effective in assessments that require deep reasoning, environmental awareness, or multi-step validation. For this reason, automation is most valuable when combined with human expertise. Scalability can improve the operational reach of the assessment, but analytical depth still depends on the ability of

the tester to interpret, correlate, and validate the technical evidence produced during the engagement.

2.5 Penetration Testing in the Enterprise Security Context

Within an enterprise environment, penetration testing is not a purely technical activity. The value of penetration testing depends on how effectively its outputs can guide security decision-making, support security awareness, and contribute to a stronger organizational security posture [1]. It therefore bridges the technical and managerial dimensions of security: it produces evidence through offensive action, but this evidence must be contextualized within the enterprise environment to become useful.

In large and complex environments, technical findings may have implications that extend beyond the directly affected asset. A weakness may influence multiple systems, affect a business process, or reveal broader issues related to governance, segmentation, access control, or mitigation processes. Penetration testing is therefore relevant not only because it identifies exploitable conditions, but also because it helps translate technical evidence into a more actionable understanding of organizational risk.

2.5.1 Relationship with Enterprise Risk Management

Penetration testing can support enterprise risk management by providing evidence-based insights into exploitable weaknesses and their potential consequences [1]. While risk assessments may rely on policy reviews, asset inventories, or qualitative evaluations to identify potential risks, penetration testing can provide concrete evidence that a particular vulnerability is, in fact, exploitable under certain controlled conditions. This kind of testing is particularly useful when the organization needs evidence of practical exploitability rather than only policy-based or inventory-based indications of exposure.

It supports risk management by helping confirm the presence, severity, and exploitability of technical problems. By simulating exploitation scenarios, penetration testing can support more accurate prioritization of known security issues and help security teams focus on risks that can materialize under realistic conditions.

2.5.2 Relevance in Complex Organizational Environments

In enterprise organizations, penetration testing is particularly relevant because security exposure often emerges from scale, heterogeneity, and interdependent operations. These characteristics indicate that risk may arise from the interaction of several components rather than from a single isolated vulnerability. As a result, the significance of technical findings often depends on a broader view of the environment rather than a purely local analysis. Trust boundaries, internal communication mechanisms, and dependencies between infrastructure layers therefore become particularly important areas of concern.

Penetration testing is valuable in these contexts because it supports a more integrated understanding of exposure. By combining technical analysis with controlled validation, it helps reveal not only what is vulnerable, but also how compromise may develop within the organization. This is particularly important in enterprise settings where systems, users, services, and governance structures are closely connected and where the practical significance of a weakness may depend on its relation to a much broader environment.

2.5.3 Interaction with Governance and Security Processes

The results of a penetration test may also support governance and security-related processes that go beyond the strictly technical aspects of the engagement [1]. They may inform remediation planning, security control assessment, decision-making processes, risk owner discussions, technical team reviews, and management-level evaluations. In this respect, a penetration test is not only a source of technical findings but also an input to wider organizational processes aimed at improving security.

Its interaction with governance is particularly relevant when reporting is clear, findings are well contextualized, and recommendations are aligned with operational realities. A penetration test is most valuable when its outputs can be translated into decisions and actions that fit the organization's security lifecycle. For this reason, the enterprise relevance of penetration testing depends not only on the technical quality of the assessment, but also on the ability to connect its results to governance, remediation, and strategic security management.

Chapter 3

Research Problem, Objectives and Questions

After establishing the conceptual and operational foundations of penetration testing, the thesis now formalizes its central research perspective. Developing a coherent comparative and methodological analysis requires a clear definition of the problem addressed, the objectives guiding the work, and the research questions structuring the following chapters. This step is necessary because the thesis does not aim only to describe penetration testing frameworks and practices, but to examine how they can be interpreted and applied in enterprise contexts shaped by methodological complexity, operational constraints, automation, and the emerging use of LLM-assisted analytical support.

This chapter therefore defines the analytical framework of the thesis. It first explains the research motivation, highlighting the need for structured enterprise-oriented penetration testing and the limits of considering frameworks, automation, or analytical support in isolation. It then formalizes the problem statement, identifies the main research objectives, and presents the research questions that guide the following chapters. Taken together, these elements provide the logical bridge between the background developed in the previous chapter and the structured comparison model introduced in the next one.

3.1 Research Motivation

The motivation of this thesis arises from the observation that penetration testing in enterprise environments cannot be treated as a purely technical activity. In complex organizational settings, the effectiveness of a security assessment depends not only on the identification of weaknesses, but also on the ability to structure the assessment process, interpret findings correctly, and relate technical evidence to realistic attack scenarios. For this reason, the present work is motivated by the need to examine penetration testing from a broader methodological perspective that accounts for frameworks, automation, analytical support, and practical applicability.

This motivation is further strengthened by the gap that often exists between methodological references and operational practice. While several penetration testing frameworks provide useful guidance, enterprise assessments are frequently conducted under conditions that require adaptation, prioritization, and the integration of different forms of support. In this context, the thesis is motivated by the need to understand how methodological structure can be preserved while also responding to the practical demands of real penetration testing activities.

3.1.1 Need for Structured Enterprise-Oriented Penetration Testing

Enterprise penetration testing cannot rely on ad hoc execution or isolated technical checks. Because modern corporate infrastructures are characterized by technical heterogeneity and complex organizational constraints, assessments require a clear methodological foundation. A structured approach helps ensure consistency across testing phases, maintain coherence between initial objectives and final results, and contextualize technical findings within the broader enterprise environment.

This need becomes even more relevant when the assessment concerns infrastructures that include external exposure, internal systems, and interconnected organizational processes. In these cases, the value of penetration testing depends not only on the identification of exploitable conditions, but also on the ability to frame them within a process that is systematic, traceable, and aligned with enterprise requirements. The motivation of this thesis, therefore, lies in the need to discuss penetration testing as a structured enterprise-oriented activity rather than as a purely technical exercise.

3.1.2 Limits of Isolated Methodological Frameworks

A further motivation of the thesis lies in the limitations that emerge when methodological frameworks are considered in isolation. Existing frameworks offer valuable guidance, but they differ significantly in focus, level of formalization, scope, and intended use. Some emphasize process structure, others provide stronger support for documentation or compliance, while others are more useful for application security or attack modeling. As a result, no single framework appears to cover all the dimensions that may become relevant in enterprise penetration testing.

This does not reduce the usefulness of existing methodological references, but it raises the need for a comparative perspective. Understanding the strengths and weaknesses of different frameworks is important for assessing how well they align with the requirements of enterprise engagements. The thesis is therefore motivated by the need to move beyond isolated descriptions of frameworks and toward a more structured discussion of their applicability, complementarity, and practical relevance.

3.1.3 Need to Balance Automation Efficiency and Analytical Reliability

The growing role of automation in penetration testing represents another central motivation for this work. Automated tools have become essential in practical assessments because they improve speed, scalability, and initial visibility across large or complex environments. In enterprise settings, these benefits are often indispensable, since manual assessment alone may be insufficient to handle the breadth of the target surface within realistic operational constraints.

At the same time, automation introduces important limitations. Tool-generated outputs may be incomplete, contain false positives, or fail to reflect the true security significance of a finding in its specific context. This creates a tension between efficiency and analytical reliability. The thesis is motivated by the need to examine this balance more carefully, discussing how automation can be integrated into penetration testing workflows without assuming that speed and coverage are sufficient on their own.

3.1.4 Potential Role of LLM-Assisted Support in Analyst Decision-Making

A final motivation of the thesis concerns the emerging possibility of using LLM-assisted support in penetration testing activities. Large language models are increasingly considered as potential aids for interpretation, correlation of findings, and reasoning about plausible attack paths. Their possible contribution is particularly interesting in contexts where analysts must connect fragmented technical evidence and formulate hypotheses about exploitation scenarios.

While LLM-based assistance shows promise as an analytical aid, its adoption is complicated by concerns related to accuracy, lack of direct environmental awareness, and the need for human validation. For this reason, the thesis avoids the unrealistic premise that AI can replace the analyst. Instead, it critically evaluates the extent to which LLMs can support human reasoning and decision-making when integrated into a structured penetration testing methodology.

3.2 Problem Statement

The central problem addressed by this thesis concerns the difficulty of structuring penetration testing activities in a way that is methodologically sound and practically effective for enterprise environments. Modern enterprise assessments are characterized by technical complexity, multiple exposure surfaces, operational constraints, and the widespread use of automated tools. In such settings, penetration testing cannot be reduced to either the mechanical application of a single framework or the uncritical use of technical tools. Instead, it requires a structured perspective that combines methodological guidance, practical execution, and analytical interpretation.

This problem is particularly relevant because enterprise penetration testing often takes

place in contexts where breadth of coverage, depth of analysis, and quality of interpretation must be balanced at the same time. Existing frameworks provide useful methodological references, but none of them appears sufficient on its own to address all the demands that emerge in complex enterprise engagements. At the same time, automation improves operational efficiency, yet frequently produces outputs that require contextual validation and prioritization. Ultimately, the issue is not just how to conduct a penetration test, but how to structure it to maintain methodological discipline without sacrificing practical or analytical responsiveness.

3.2.1 No Single Framework Fully Addresses the Needs of Enterprise Penetration Testing

The first aspect of the problem concerns the limitations that individual penetration testing frameworks present when they are applied in enterprise settings. Frameworks vary in scope, degree of formalization, operational depth, and intended use. Some provide strong process guidance, others are more useful for compliance-oriented contexts, while others focus on specific areas such as application security or adversarial modeling. For this reason, no single framework appears capable of fully addressing the methodological and operational requirements of an enterprise penetration test.

This does not imply that existing frameworks are inadequate. Rather, it highlights that enterprise penetration testing often requires dimensions that cannot be fully covered by one framework alone, such as broad technical applicability, adaptability to heterogeneous infrastructures, support for automation, and compatibility with practical analyst workflows. This provides the basis for a more structured and comparative discussion.

3.2.2 Automated Tools Improve Efficiency but May Produce Incomplete or Noisy Results

A second element of the problem concerns the role of automated tools in practical penetration testing. Automation has become indispensable in modern assessments because it improves speed, scalability, and initial technical visibility. In enterprise contexts, these advantages are often essential for handling broad and heterogeneous attack surfaces within realistic time and resource constraints.

However, tool-generated findings cannot serve as standalone conclusions. They may be incomplete, affected by false positives, or unable to reflect the actual exploitability and relevance of a finding in the target environment. This creates a problem of interpretation and reliability. The issue is therefore not whether automation is useful, but how its contribution can be integrated without overestimating the value of tool-generated outputs.

3.2.3 Analytical Support Is Needed to Interpret Findings and Reason on Attack Paths

A further problem arises from the fact that penetration testing does not end with the detection of technical issues. Findings must be interpreted, related to one another, and assessed in terms of realistic attack paths and possible consequences. This kind of analytical work often requires reasoning that goes beyond direct technical output, especially in enterprise environments where the significance of a weakness depends on context, architecture, and interaction with other conditions.

For this reason, enterprise penetration testing involves a persistent need for analytical support. The problem is not limited to identifying weaknesses, but extends to understanding which findings matter most, how they may be combined, and whether they can produce meaningful compromise scenarios. This interpretative dimension is one of the reasons why penetration testing cannot be treated as a purely mechanical process.

3.2.4 LLM-Based Assistance May Be Useful, but Requires Human Supervision and Validation

A final element of the problem concerns the possible role of LLM-based assistance in penetration testing workflows. Large language models may provide useful support for activities such as result interpretation, correlation of findings, and preliminary reasoning about attack paths. Their potential relevance stems from their ability to help analysts organize complex technical information and formulate plausible hypotheses more efficiently.

At the same time, such assistance raises important limitations. LLMs do not possess direct awareness of the target environment and cannot independently verify technical conditions. Thus, while LLM-generated outputs may provide useful analytical support, they cannot be treated as definitive conclusions. This creates a dual challenge: LLMs may enhance the assessment workflow, but they require expert oversight to prevent erroneous or misleading interpretations.

3.3 Research Objectives

The thesis pursues a set of research objectives aimed at addressing the methodological and practical issues discussed in the previous section. These objectives are not limited to a descriptive overview of penetration testing practices, but are intended to support a more structured understanding of how frameworks, automation, analytical support, and human validation can be related in enterprise environments. Taken together, they define the analytical direction of the work and clarify the contribution that the thesis aims to provide.

More specifically, the objectives of the thesis concern the comparative analysis of methodological references, the discussion of automation in practical penetration testing, the examination of LLM-assisted analytical support, and the validation of these elements through a real enterprise application. In this sense, the research objectives reflect both the

methodological orientation of the work and its applied dimension.

3.3.1 Compare Major Penetration Testing Frameworks Using Explicit Evaluation Criteria

The first objective is to conduct a structured comparison of leading penetration testing frameworks using explicit evaluation criteria. By moving beyond isolated descriptions, this work establishes a common analytical baseline to clearly delineate the distinct advantages, constraints, and operational focus of each methodology.

This objective is important because the comparative discussion constitutes the basis for a more rigorous methodological assessment. By relying on explicit criteria rather than purely intuitive judgment, the thesis seeks to provide a clearer analytical structure for understanding how different frameworks may respond to the needs of enterprise penetration testing.

3.3.2 Assess Their Suitability for Enterprise Environments

A second objective is to assess the suitability of the selected frameworks for enterprise-oriented penetration testing activities. This means examining not only their general characteristics, but also the extent to which they can support assessments carried out in complex organizational contexts shaped by architectural fragmentation, rigid execution limits, and multifaceted risk profiles.

The purpose of this objective is to move the analysis from general methodological description to contextual applicability. Enterprise environments introduce requirements that are not always fully addressed by methodological references taken in isolation, and for this reason the thesis aims to evaluate how well the selected frameworks align with those requirements.

3.3.3 Analyze the Contribution and Limitations of Automated Tools in Practical Testing

Another objective of the thesis is to analyze the practical contribution and limitations of automated tools in penetration testing workflows. This involves examining the role of automation in activities such as discovery, scanning, validation support, and result organization, while also discussing the limitations that emerge in terms of false positives, incomplete coverage, and lack of contextual interpretation.

This objective is particularly relevant because automation has become a structural component of contemporary penetration testing practice. The thesis therefore aims not only to acknowledge its operational value, but also to assess more critically the extent to which automated outputs can support or distort the assessment process when they are not properly interpreted.

3.3.4 Examine the Role of LLM-Assisted Support in Result Validation and Attack-Path Reasoning

A further objective is to examine the possible role of LLM-assisted support in result validation and in the exploration of attack paths. The focus here is not on treating large language models as autonomous testing instruments, but on assessing whether they may provide useful support in interpreting findings, correlating information from multiple sources, and assisting analysts in reasoning about plausible exploitation scenarios.

This objective is driven by the increasing attention given to LLM-based systems in cybersecurity workflows. The thesis aims to consider this possibility critically, identifying both the areas in which such support may be useful and the limits that prevent it from replacing technical verification and expert judgment. Particular attention is given to risks such as inaccurate inferences, lack of direct environmental awareness, and the need for human validation.

3.3.5 Define and Discuss a Hybrid Methodological Approach

The thesis also aims to define and discuss a hybrid methodological approach capable of relating frameworks, automation, and analyst reasoning within a broader enterprise-oriented perspective. The purpose is not to propose a wholly new formal standard, but to examine how different methodological and operational elements may be combined in a more coherent and practically relevant way.

This objective follows directly from the previous ones. If individual frameworks are only partially sufficient, and if automation and analytical support both provide benefits and limitations, then it becomes necessary to discuss how these elements may interact within a structured methodology. The hybrid perspective is therefore one of the central outcomes toward which the thesis is directed.

3.3.6 Discuss Its Practical Relevance Through a Real Enterprise Application

The final objective is to discuss the practical relevance of the proposed perspective through a real enterprise penetration testing application. The practical case developed in the thesis, based on the internship experience with KPMG, provides a concrete setting in which the relevance of frameworks, automation, LLM-assisted support, and human validation can be discussed in relation to actual assessment activity.

This objective does not aim at broad statistical generalization or formal validation. The practical application is intended to show whether the methodological discussion developed in the earlier chapters remains meaningful when considered in the context of a real engagement involving external, internal, and physical security dimensions.

3.4 Research Questions

The research questions of this thesis are derived from the methodological and practical issues discussed in the previous sections. Their role is to translate the general problem of the thesis into a set of more precise questions capable of guiding the comparative analysis, the discussion of automation and LLM-assisted support, and the interpretation of the practical application. In this sense, the research questions provide the logical link between the conceptual framing of the problem and the analytical development of the following chapters.

Overall, these questions reflect the thesis’s central direction. They emphasize that complex enterprise assessments cannot be fully addressed through a standalone methodology or an isolated set of automated tools. Instead, such environments require an integrated perspective that connects established frameworks, automation, LLM-assisted support, and human validation.

3.4.1 Can a Single Penetration Testing Framework Adequately Support Enterprise Testing Activities?

The first research question investigates whether a single penetration testing framework can be considered sufficient for the methodological and operational needs of enterprise environments. This question emerges from the observation that available frameworks differ in scope, depth, intended use, and degree of formalization, and that enterprise penetration testing often involves requirements that extend across several of these dimensions.

The purpose of this question is not to identify the “best” framework in absolute terms, but to assess whether any individual methodological reference can adequately support enterprise testing activities on its own. In doing so, the thesis aims to clarify whether a more integrated perspective is needed.

3.4.2 Which Criteria Are Most Relevant for Comparing Penetration Testing Frameworks in Enterprise Contexts?

The second research question concerns the criteria that are most useful for comparing penetration testing frameworks in enterprise-oriented scenarios. Since the thesis adopts a structured comparison model, it becomes necessary to determine which dimensions of evaluation are most meaningful in relation to the broader objectives of the work.

This question is important because the value of a comparative analysis depends strongly on the transparency and relevance of the criteria on which it is based. The thesis therefore seeks to identify the dimensions that best capture enterprise applicability, methodological usefulness, and compatibility with practical testing needs.

3.4.3 How Do Automated Tools Affect the Quality and Efficiency of Penetration Testing?

The third research question examines the impact of automated tools on both the efficiency and the quality of penetration testing activities. Automation clearly improves speed, scalability, and visibility, especially in large and heterogeneous environments, but these benefits must be considered together with the possibility of incomplete coverage, false positives, and limited contextual understanding.

The purpose of this question is to assess automation in a balanced way. Rather than treating it as either unquestionably beneficial or fundamentally insufficient, the thesis aims to understand how automation affects the assessment process and under which conditions its contribution remains reliable and useful.

3.4.4 In Which Ways Can LLM-Assisted Support Help Analysts in Interpreting Findings and Exploring Attack Paths?

The fourth research question addresses the possible contribution of LLM-assisted support to analyst reasoning in penetration testing workflows. More specifically, it examines whether large language models can help in interpreting findings, relating technical evidence, and exploring plausible attack paths in a way that is practically meaningful for the assessment process.

This question is motivated by the growing relevance of LLM-based systems in analytical tasks, but it is approached critically. The thesis does not assume that such systems can replace the analyst. Instead, it seeks to understand in which ways they may offer useful support and where their contribution remains constrained by the need for supervision and verification.

3.4.5 What Benefits Can Be Derived from a Hybrid Approach Combining Frameworks, Automation, LLM-Assisted Support, and Human Validation?

The final research question concerns the possible benefits of adopting a hybrid methodological approach that combines frameworks, automation, LLM-assisted support, and human validation. This question reflects the broader direction of the thesis, which is oriented toward understanding not only the value of each element in isolation, but also the value that may emerge from their combination.

Through this question, the thesis aims to assess whether a hybrid perspective can offer a more effective response to the needs of enterprise penetration testing than isolated approaches. In particular, it seeks to understand whether such an approach can improve methodological coherence, analytical quality, and practical relevance in complex assessment contexts.

Chapter 4

Comparison Model for Penetration Testing Frameworks

To compare penetration testing frameworks in a way that is methodologically consistent and relevant to the objectives of this thesis, it is necessary to adopt a structured comparison model. The frameworks considered in the following chapters differ in scope, operational orientation, level of formalization, and intended use [2, 3, 4, 1, 5]. For this reason, a simple descriptive overview would not be sufficient to support a rigorous assessment of their relevance for enterprise penetration testing. A more explicit analytical structure is needed to make the comparison transparent, coherent, and aligned with the broader research problem defined in the previous chapter.

This chapter introduces the comparison model adopted in the thesis and explains the rationale behind its construction. It first discusses why purely descriptive comparisons are insufficient and why reusable and explicit evaluation criteria are needed. It then defines the criteria used to assess the selected frameworks and clarifies how these criteria are applied throughout the thesis. Finally, the chapter discusses the role of the comparison model in supporting the research objectives, while also acknowledging its methodological limits. In this way, the chapter provides the analytical basis for the framework evaluation developed in Chapter 5.

4.1 Need for a Structured Comparison Model

A structured comparison model is necessary because the frameworks considered in this thesis differ significantly in scope, operational orientation, level of formalization, and intended use [2, 3, 4, 1, 5]. When methodological references are discussed only in descriptive terms, the analysis risks remaining fragmented and overly impressionistic. In such a case, it becomes difficult to determine whether the observed differences are truly relevant for enterprise

penetration testing or are simply the result of different presentation styles, terminologies, or historical contexts.

For the purpose of this thesis, comparing penetration testing frameworks is not intended as a broad descriptive exercise, but as a methodological process for evaluating their relevance to enterprise environments. This requires an analytical structure capable of supporting consistent comparison across multiple dimensions. A structured comparison model therefore becomes necessary in order to connect the framework analysis to the broader research problem of the thesis, namely the difficulty of organizing penetration testing activities in a way that is both methodologically sound and practically effective in complex enterprise settings.

4.1.1 Limits of Purely Descriptive Comparisons

A purely descriptive comparison is inherently constrained, as it lacks the stable analytical dimensions necessary for systematic framework evaluation. Although this approach may successfully outline individual contents, highlight salient strengths, and identify specific limitations, it precludes rigorous comparison across different methodologies. Ultimately, this results in an analysis that, despite being informative at a surface level, provides minimal methodological utility.

This shortcoming is particularly relevant for this research, as its goal extends beyond outlining familiar frameworks and aims to evaluate their practical relevance in enterprise security assessments. A purely descriptive approach obscures key differentiators: it prevents a clear understanding of which frameworks provide the most robust operational guidance, and which possess the inherent adaptability required for automation and hybrid analytical workflows. For this reason, description alone is not enough to support the analytical aims of the work.

4.1.2 Need for Reusable and Explicit Evaluation Criteria

To address the limitations of a purely descriptive approach, this thesis employs a comparative model grounded in explicit evaluation criteria. This choice improves transparency because it defines in advance the analytical dimensions used to assess each framework. It also supports consistency, since the same evaluative perspective is applied to all selected methodological references, rather than allowing the comparison to change implicitly according to the framework being discussed.

A further reason for adopting explicit criteria is that they make the comparison more structured and reusable. The model developed in this thesis is aligned with the specific research objectives of the work and does not claim universal applicability. Nevertheless, it provides a stronger methodological foundation than ad hoc qualitative observations. By defining these criteria, the thesis supports a reasoned evaluation of how different frameworks address enterprise requirements, operational execution, automated tool integration, reporting, validation, and hybrid assessment workflows.

Finally, explicit evaluation criteria help reduce the inevitable subjectivity in qualitative analysis. Without clearly defined analytical boundaries, a comparative discussion may be

influenced by implicit preferences, uneven observations, or inconsistent emphasis across frameworks. The use of explicit criteria provides a rational basis for explaining why certain framework characteristics are relevant and how they relate to the broader objectives of the thesis. In this sense, the criteria do not replace human judgment; rather, they guide it and make the interpretation more transparent, structured, and methodologically defensible.

4.2 Definition of Comparison Criteria

To compare penetration testing frameworks in a way that is analytically consistent with the objectives of this thesis, it is necessary to define a set of explicit evaluation criteria. These criteria do not aim to establish a universal standard for assessing all methodological frameworks. Rather, they are introduced as a thesis-specific analytical tool designed to support the comparison of selected penetration testing frameworks across enterprise environments, practical execution, automation, and hybrid methodological integration.

The choice of these criteria reflects the broader research problem discussed in the previous chapter. Since enterprise penetration testing involves technical complexity, operational constraints, tool-assisted workflows, and a strong need for contextual interpretation [1, 2, 5], the comparison cannot be limited to generic methodological descriptions. Instead, it must focus on the dimensions most relevant to understanding how different frameworks can support real penetration testing in complex organizational settings. For this reason, the comparison model adopted in this thesis is based on seven criteria: coverage scope, operational depth, level of formalization, enterprise suitability, support for automation, reporting and validation support, and adaptability to hybrid workflows.

These criteria were selected to evaluate the frameworks not only as theoretical references but also as practical instruments for organizing penetration testing activities. They cover two interconnected areas of analysis. On the one hand, they assess methodological characteristics such as coverage scope, operational depth, and level of formalization. On the other hand, they address execution-oriented requirements such as enterprise suitability, support for automation, reporting and validation support, and adaptability to hybrid workflows. The result is a balanced comparison model that helps differentiate the selected frameworks while remaining aligned with the analytical goals of the thesis.

It is also important to clarify that the model follows a qualitative rather than quantitative logic. The purpose is not to produce numerical rankings or mechanical scores, but to create a structured basis for interpretation. This choice is consistent with the nature of the thesis, since many of the relevant dimensions cannot be reduced convincingly to fixed measurements without oversimplifying the methodological differences among frameworks. The role of the criteria is therefore to guide analysis, not to replace it.

4.2.1 Coverage Scope

The coverage scope criterion evaluates the extent of activities, target environments, and operational boundaries addressed by a specific framework. Methodological references vary significantly: some provide holistic support throughout the penetration testing lifecycle,

whereas others deliberately focus on specific analytical phases or specialized technical ecosystems [2, 3, 4, 5]. This metric is therefore essential for determining whether a framework provides broad methodological support or whether its value is concentrated in a more limited part of the assessment process.

In enterprise penetration testing, coverage scope is particularly significant because corporate infrastructures involve overlapping exposure vectors and diverse assessment perspectives. Frameworks with extensive coverage naturally facilitate the coordination of technical execution and structured reporting. Nevertheless, a wide-ranging scope should not be conflated with absolute methodological strength, since breadth may occasionally occur at the expense of analytical depth. For this reason, coverage scope should not be evaluated in isolation, but in relation to the other dimensions of the comparison model.

4.2.2 Operational Depth

Operational depth concerns the extent to which a framework provides practical guidance for executing penetration testing activities. While some methodological references remain relatively high-level and focus on principles, structure, or recommended phases, others provide more detailed support in terms of procedures, techniques, execution logic, and task-oriented recommendations [2, 3, 5]. This criterion is therefore intended to capture how far a framework moves from a general methodological orientation toward concrete operational usefulness.

This dimension is important because enterprise penetration testing often requires more than conceptual guidance. Within complex infrastructures, analysts often need frameworks that go beyond high-level goal definition and provide guidance on the practical organization of testing workflows. A methodology with strong operational depth is therefore valuable in real engagements, especially when the assessment involves complex execution steps, coordination among multiple stakeholders, and the need to translate methodological structure into concrete operational activities.

4.2.3 Level of Formalization

The criterion of formalization evaluates the structural rigor of a framework, assessing how explicitly it defines analytical phases, internal logic, and expected deliverables. Highly formalized methodologies present their components systematically, resulting in a process that is highly communicable and inherently reproducible. In contrast, methodologies with lower formalization may provide excellent tactical advice while relying on a more fluid organizational structure [4, 2, 1]. Evaluating this dimension is critical because formalization influences both clarity and reproducibility.

A well-defined formal structure supports traceability during enterprise assessments that involve multiple stakeholders and reporting requirements. However, excessive rigidity can limit operational responsiveness when complex infrastructures require adaptation or deviation from predefined procedures. Thus, a high level of formalization does not automatically indicate a superior framework. It is instead a methodological property that influences reproducibility, flexibility, and practical implementation within enterprise

contexts.

4.2.4 Enterprise Suitability

The concept of enterprise suitability captures a framework’s capacity to support penetration testing operations in complex organizational environments. Specifically, it evaluates how well a methodology can adapt to contexts characterized by diverse technological infrastructures, overlapping internal and external perimeters, operational constraints, and the need to translate localized technical evidence into broader risk-related understanding.

This criterion is central to the comparison model because the thesis is specifically focused on enterprise penetration testing. A framework may be methodologically sound in general terms but still show limitations when applied to business environments that require coordination, scalability, documentation, and contextual interpretation of findings. Enterprise suitability is therefore intended to capture not only the technical relevance of a framework, but also its ability to remain meaningful when testing is conducted in organizational settings rather than in isolated or narrowly defined technical scenarios.

Modern enterprise environments justify the need for this criterion because they are shaped by structural scale, segmented architectures, operational dependencies, and continuity requirements. A framework suited for enterprise use should support more than isolated technical discovery; it should help connect localized findings to broader operational, managerial, and risk-related considerations. Consequently, this dimension evaluates whether a methodology remains practically useful when applied within environments governed by complex organizational and business dependencies.

Evaluating this dimension also helps distinguish frameworks that are strong in principle from those that remain effective in context. A theoretical model may appear complete when considered in isolation, yet become less practical in settings requiring extensive reporting, cross-team coordination, and nuanced risk interpretation. For this reason, enterprise suitability represents a primary criterion for assessing the practical utility of penetration testing frameworks within the scope of this thesis.

4.2.5 Support for Automation

Support for automation assesses the extent to which a penetration testing framework can be integrated with automated tools throughout the assessment lifecycle. This does not mean that the methodology must explicitly prescribe the use of automation or depend on a specific toolset. Rather, it indicates whether the structure of the framework can remain compatible with software utilities used for activities such as reconnaissance, vulnerability scanning, information gathering, exploitation support, evidence organization, and reporting.

This criterion is particularly relevant because automation has become a structural component of modern penetration testing practice [1, 2, 5]. In complex enterprise environments, the scale and heterogeneity of the target infrastructure often make a fully manual approach impractical. As a result, methodological references should ideally be compatible with workflows supported by automated utilities. A framework that can accommodate automation provides greater practical value during contemporary assessments, provided

that technical efficiency remains balanced with contextual validation and expert human oversight.

The relevance of this criterion also lies in its capacity to align methodological structure with operational reality. A framework may be conceptually robust, but its practical utility decreases if it cannot accommodate the software utilities commonly used to accelerate host discovery, support vulnerability scanning, correlate findings, and organize large volumes of technical data. For this reason, the comparison model assesses whether a methodology can support tool-assisted workflows without forcing the analyst into an exclusively manual or overly rigid execution structure.

At the same time, this criterion should not be interpreted in an overly simplistic manner. Strong compatibility with automated tools does not automatically make a framework superior, and limited automation support does not necessarily make a methodology unsuitable. Instead, this dimension is designed to capture how easily a framework can preserve its usefulness in practical conditions where automated tools are widely deployed. For the objectives of this thesis, this aspect is significant because one of the research problems concerns the balance between methodological discipline and the operational efficiency required in enterprise environments.

4.2.6 Reporting and Validation Support

Reporting and validation support evaluates the degree to which a framework helps organize findings, confirm results, and communicate outcomes in a structured way. Certain methodologies place significant emphasis on evidence collection, vulnerability verification, and documentation standards. Conversely, other frameworks concentrate predominantly on operational execution, providing less detailed guidance on how verified findings should be presented, prioritized, and translated into actionable recommendations [2, 1, 3].

This criterion is important because a penetration test generates value not only through technical discovery, but also through the interpretation, validation, and communication of the results. In complex enterprise environments, systematic reporting is necessary to support remediation planning, establish risk priorities, and facilitate communication with both technical and managerial stakeholders. A framework that provides stronger support for validation and reporting can therefore help connect specialized testing activities with broader security management processes.

Furthermore, this dimension captures the need to translate complex security evidence into deliverables suitable for audiences beyond the immediate assessment team. It does not merely assess administrative documentation capabilities. Instead, it evaluates whether a methodology can support the transformation of raw technical data into coherent operational knowledge that remains useful for remediation, governance, and decision-making within the organization.

4.2.7 Adaptability to Hybrid Workflows

The criterion of adaptability to hybrid workflows evaluates the capacity of a framework to operate within broader methodological combinations involving automated tools, analytical

support, and continuous human validation. This dimension goes beyond the narrower question of automation compatibility. It considers whether the framework can remain useful in assessment processes where formal procedures are combined with software-generated outputs, contextual interpretation, iterative verification, and analyst-driven reasoning.

This dimension is relevant to the present research because enterprise penetration testing increasingly relies on hybrid forms of execution. Methodological references are rarely applied in isolation from technological tools or supporting analytical processes. Instead, they often function as structured guidelines within complex workflows that combine formal methodology, automated discovery, evidence correlation, contextual reasoning, and expert judgment. A framework that is adaptable in this respect is better suited to contemporary practice, especially in enterprise contexts where rigid methodological isolation is often unrealistic.

The value of this criterion lies in its ability to capture flexibility at the methodological level. A framework may present broad coverage, strong operational depth, or a high level of formalization, but still be difficult to integrate into modern assessment routines. This difficulty may emerge when automated tools generate large volumes of technical data, when analysts need support to interpret findings, or when attack paths must be validated through multiple iterations. Adaptability to hybrid workflows therefore concerns the ability of a framework to remain useful when testing activities combine structured procedures, tool-assisted evidence, and human interpretation.

Finally, this criterion is important for the overall argument of the thesis because it connects the framework comparison with the later discussion of automation, LLM-assisted support, and human supervision. It helps determine whether a framework can support traditional penetration testing logic while also accommodating more integrated and iterative assessment practices. In this sense, adaptability to hybrid workflows reflects one of the main methodological concerns of the thesis: understanding how established frameworks can remain relevant in enterprise assessments shaped by automation, analytical support, and expert validation.

The comparison criteria adopted in this thesis are summarized in Table 4.1.

4.3 Use of the Comparison Model in the Thesis

The comparison model established in this chapter functions as an analytical tool for supporting the main objectives of the research. Its primary purpose is to provide a consistent methodological baseline for evaluating the selected penetration testing frameworks. This ensures that the comparative analysis developed in the following chapter does not remain limited to description or subjective impressions. By using explicit criteria, the model supports a structured examination of different methodologies and connects the evaluation to the central challenges of enterprise penetration testing discussed in the previous chapters.

The model also fulfills an important interpretative function. The objective of the thesis is not simply to categorize methodological references, but to understand how their structural characteristics influence enterprise applicability, operational utility, automation compatibility, reporting support, and integration within hybrid workflows. For this reason,

Table 4.1: Comparison criteria adopted for the evaluation of penetration testing frameworks.

Criterion	Main focus	Relevance for the thesis
Coverage scope	Breadth of activities, targets, and testing dimensions addressed by the framework	Helps determine whether a framework can support broad enterprise assessments or only narrower testing contexts
Operational depth	Degree of practical guidance provided for execution	Clarifies how useful a framework is in real engagements requiring actionable methodological support
Level of formalization	Clarity of phases, internal logic, and expected outputs	Supports the evaluation of methodological structure, reproducibility, and traceability
Enterprise suitability	Capacity to remain effective in complex organizational environments	Connects the comparison directly to the enterprise focus of the thesis
Support for automation	Compatibility with tool-assisted workflows across the assessment lifecycle	Reflects the practical relevance of frameworks in contemporary penetration testing
Reporting and validation support	Ability to organize findings, confirm results, and communicate outcomes	Highlights the role of frameworks in turning technical evidence into actionable results
Adaptability to hybrid workflows	Capacity to integrate automation, analytical support, and human validation	Supports the discussion of hybrid methodological approaches developed in the thesis

the comparison model acts as a link between the methodological discussion and the applied orientation of the thesis. This dual analytical and functional role allows the model to organize the comparative results coherently while showing their relevance to the broader argument developed in the following chapters.

4.3.1 How the Criteria Are Applied

The evaluative dimensions defined in the preceding section are applied to each selected framework in a consistent way, so that the comparative analysis remains coherent across the complete set of references examined in this thesis. This approach ensures that the methodologies are not merely described through general summaries, but evaluated through the same analytical lenses. As a result, the comparison can identify differences in scope, practical guidance, structural organization, enterprise applicability, automation

compatibility, reporting support, and adaptability to hybrid assessment workflows.

The application of these criteria is qualitative rather than numerical. The thesis does not attempt to assign rigid scores or generate a ranking based on abstract quantitative metrics. Instead, the criteria function as structured interpretative dimensions that make it possible to identify specific strengths, limitations, and points of emphasis across the selected frameworks. This approach is consistent with the objectives of the work, since the goal is not to designate a single superior framework, but to evaluate how different methodological references address enterprise requirements and the broader analytical themes of the research.

Each criterion operates as a recurring analytical lens through which the selected methodologies are discussed. This does not mean that every criterion has the same relevance for every framework, nor that every dimension is represented with the same level of detail. Some frameworks may distinguish themselves through strong process structure, while others may be more relevant for operational depth, application security, attack modeling, or integration within hybrid workflows. The comparison model is therefore intended to highlight methodological distinctions and make them analytically comparable without reducing their specific characteristics.

This method of applying the criteria preserves both structural consistency and interpretative flexibility. Using the same dimensions throughout the comparative analysis maintains coherence, while the qualitative focus keeps the discussion sensitive to the specific logic, emphasis, and limitations of each framework. For the purposes of this thesis, this approach is preferable to a rigid scoring system, which might suggest a level of precision that would not fully reflect the methodological nuances relevant to professional penetration testing.

Consequently, the comparison model should be understood as more than a preparatory element for the following chapter. It represents the methodological mechanism through which the thesis connects the research problem to the comparative analysis of the frameworks. By structuring the evaluation around explicit criteria, the model enables a transition from general methodological concerns to a more detailed examination of how different frameworks respond to the practical and analytical demands of enterprise penetration testing.

4.3.2 How the Model Supports the Research Objectives

The comparison model directly supports the research objectives defined in the previous chapter. First, it provides the methodological basis for comparing major penetration testing frameworks through explicit evaluation criteria rather than through isolated descriptive accounts. Second, it enables assessing their suitability for enterprise environments by focusing the comparison on dimensions that are especially relevant in organizational contexts marked by complexity, operational constraints, and heterogeneous testing needs.

In addition, the model supports the broader analytical orientation of the thesis by incorporating criteria related to automation, reporting, validation, and hybrid workflows. In this sense, it does not serve only to compare frameworks in abstract methodological terms, but also to connect the comparison to the practical questions that motivate the work.

Through this model, the thesis relates the framework analysis to the role of automated tools, the emerging use of LLM-assisted support, and the continuing importance of human judgment in enterprise penetration testing.

4.3.3 Methodological Limits of the Comparison

While the comparison model provides a systematic analytical foundation, it also has methodological limits that must be acknowledged. First, the model is context-dependent, since it has been developed in relation to the specific objectives of this thesis. It does not claim universal applicability and does not aim to establish an absolute taxonomy for evaluating all security frameworks. Instead, the selected criteria reflect the priorities of this work, with particular attention to enterprise suitability, practical execution, automation, reporting, validation, and hybrid methodological integration.

A second limitation concerns the interpretative nature of the comparative process. Even when explicit criteria are used, assessing methodological frameworks still requires analytical judgment, especially for dimensions such as enterprise suitability or adaptability to hybrid workflows. As a result, the model improves transparency and consistency, but it cannot eliminate the need for reasoned interpretation. For this reason, the final analysis should be understood as a structured qualitative discussion rather than as an objective or quantitative calculation.

Finally, the model does not include every possible criterion through which penetration testing frameworks could be evaluated. Alternative dimensions could be defined depending on different research goals, technical specializations, or organizational requirements. The purpose of this model is therefore relevance rather than exhaustiveness. Its value lies in supporting a transparent and coherent comparison that remains aligned with the specific research problem addressed in the thesis.

Chapter 5

Comparative Analysis of Major Penetration Testing Frameworks

This chapter applies the previously defined comparison model to a set of selected penetration testing frameworks. The purpose is not to identify a single superior methodology, but to examine how different frameworks contribute to enterprise penetration testing from different perspectives. Each framework is analyzed with respect to its structure, strengths, limitations, and suitability, as defined by the evaluation criteria introduced in Chapter 4.

The frameworks considered in this chapter were selected because they represent different but relevant perspectives on penetration testing and security assessment. PTES provides an operationally oriented penetration testing structure; OSSTMM offers a more formal, measurable view of security testing; NIST SP 800-115 provides an institutional and procedural perspective; OWASP WSTG focuses on application security testing; and MITRE ATTCK supports adversarial modeling and attack-path reasoning. Taken together, these references provide a sufficiently broad basis for discussing the methodological needs of enterprise penetration testing.

5.1 PTES – Operational Standardization

The Penetration Testing Execution Standard represents a relevant methodological reference for penetration testing and security assessment activities. Its core utility lies in structuring the testing process into distinct phases, providing a common operational framework for planning, executing, and documenting engagements consistently. For this reason, PTES is particularly useful when penetration testing is considered not only as a set of technical actions, but as a structured process that requires preparation, execution, validation, and reporting [2].

Within the scope of this thesis, PTES is considered an operationally oriented framework.

Its relevance does not derive from formal measurement or institutional regulation, but from its ability to describe the practical flow of a penetration testing engagement. This makes it a useful reference for enterprise penetration testing, especially when the objective is to maintain coherence across the assessment's phases and avoid treating testing activities as isolated technical tasks.

5.1.1 General Structure

The Penetration Testing Execution Standard is based on a lifecycle-oriented perspective of security assessments. The framework defines several main phases, including pre-engagement interactions, intelligence gathering, threat modeling, vulnerability analysis, exploitation, post-exploitation, and reporting [2]. This organization reflects the idea that professional security assessments should not begin directly with technical exploitation [2]. Instead, PTES proposes a gradual process in which planning, reconnaissance, analysis, validation, and reporting all contribute to the overall quality of the assessment.

Establishing a pre-engagement phase is critically important because it underscores the need to define the scope, objectives, and boundaries of the assessment before any active technical testing begins. This aspect is particularly important in professional and corporate environments, where assessment activities must remain aligned with authorization boundaries and operational limits. Similarly, the presence of reporting as the final phase confirms that the ultimate outcome of a penetration test transcends mere technical vulnerability discovery. It focuses instead on the production of structured evidence and remediation-oriented information that provides actionable knowledge to the organization.

Furthermore, the methodology highlights the importance of managing the transition from initial reconnaissance to exploitation and post-exploitation in a disciplined way. In this context, intelligence gathering and threat modeling support the identification of potential attack paths and plausible adversarial routes. These stages are followed by vulnerability analysis and exploitation, which provide the basis for validating whether identified weaknesses can be leveraged within the target environment. Finally, the post-exploitation phase examines the broader consequences of a successful compromise. Consequently, the overall structure of PTES is consistent with the practical logic of penetration testing as an adversarial but controlled diagnostic activity.

5.1.2 Main Strengths

The main strength of PTES is its practical orientation. Unlike frameworks centered on theoretical principles or institutional orientations, this standard establishes an explicit operational cycle that reflects realistic testing procedures. Consequently, the framework serves as an effective instrument for structuring the engagement lifecycle, ensuring that the evaluation integrates preliminary planning, analytical validation, and structured documentation with technical activities.

A second strength is the clarity of its phases. By segmenting the engagement into distinct yet interrelated stages, the framework enables practitioners to preserve a coherent perspective throughout the assessment. This structured methodology is particularly

advantageous in enterprise environments where operations often involve an array of assets, diverse testing viewpoints, and numerous stakeholders. A further advantage of this structured lifecycle is that technical evaluations are not conducted in isolation from the broader project activities.

PTES is also valuable because it explicitly includes phases that support adversarial reasoning. Intelligence gathering, threat modeling, exploitation, and post-exploitation contribute to an assessment perspective that goes beyond vulnerability identification. They encourage the tester to reason about how weaknesses may be discovered, combined, validated, and translated into actual impact. This makes PTES particularly relevant for penetration testing activities where the objective is to understand realistic compromise scenarios rather than merely produce a list of vulnerabilities.

5.1.3 Main Limitations

Despite its practical usefulness, PTES also presents some limitations. One drawback is the lack of a strong quantitative or measurement-based model. While it defines the main phases of a penetration test, it does not offer detailed metrics for evaluating the quality, completeness, or effectiveness of the assessment. This can make it less suitable when the objective is to compare results across engagements or to produce measurable security indicators.

Another limitation concerns enterprise governance integration. PTES is useful for structuring the technical and operational flow of a penetration test, but it provides less explicit guidance on integrating results into broader governance, risk management, and organizational decision-making processes. In enterprise environments, this is relevant because the value of a penetration test depends not only on technical validation but also on how findings are prioritized, assigned, remediated, and linked to organizational risk.

A further limitation is that PTES may require adaptation when applied to complex enterprise engagements. The lifecycle it proposes is clear and useful, but enterprise assessments often include heterogeneous infrastructures, multiple business units, internal and external scopes, and constraints related to continuity of operations. In such situations, PTES can provide a strong operational reference, but it may need to be complemented by other frameworks or internal procedures in order to fully address the organizational dimension of the assessment.

5.1.4 Evaluation According to the Comparison Criteria

According to the comparison criteria defined in Chapter 4, PTES provides a broad and practical view of the penetration testing lifecycle. Its phases cover the main activities that characterize a professional engagement, from pre-engagement preparation to reporting. This makes it particularly effective as a practical reference for organizing the execution of a penetration test.

In terms of coverage scope, PTES provides a broad view of the engagement lifecycle, but its coverage is mainly oriented toward penetration testing as an operational process. It does not focus specifically on a single technical domain, such as web application testing,

nor does it provide a complete enterprise governance model. For this reason, its coverage can be considered broad from the perspective of the penetration testing process, but more limited when the assessment is considered within the full enterprise security management lifecycle.

With respect to operational depth, PTES performs strongly. Its structure is close to practical testing activity and provides useful guidance for organizing the sequence of work. This makes it suitable for engagements where the tester needs a clear operational reference. However, in terms of level of formalization, PTES is less rigorous than more measurement-oriented frameworks. It defines phases and activities, but does not provide a highly formal model for measuring the consistency or completeness of the assessment.

Regarding enterprise suitability, PTES is useful but not sufficient on its own. It can support enterprise penetration testing by providing a clear process structure, but it does not fully address all organizational requirements, such as governance integration, risk ownership, or long-term remediation tracking. Its suitability therefore increases when it is combined with other references or internal enterprise processes.

In relation to automation, PTES is compatible with tool-assisted workflows, especially in phases such as intelligence gathering, vulnerability analysis, exploitation support, and reporting. However, the framework does not define automation as a central methodological element. Automated tools can be integrated naturally into the PTES lifecycle, but their use still depends on the analyst's judgment and on the practical organization of the engagement.

With respect to reporting and validation support, PTES provides a useful contribution because it explicitly includes exploitation, post-exploitation, and reporting as part of the engagement lifecycle. This helps ensure that findings are not treated only as isolated technical observations, but are connected to evidence, practical validation, and final communication. However, PTES does not provide a highly formalized reporting or validation model, and therefore its effectiveness in this area still depends on the analyst's ability to document evidence clearly, distinguish validated findings from preliminary observations, and translate technical results into actionable recommendations.

Finally, while PTES can serve as a strong operational backbone for integrating manual techniques, automated tools, and human expertise, its adaptability to modern hybrid workflows remains mostly indirect. The framework does not explicitly address emerging analytical practices such as LLM-assisted interpretation, automated evidence correlation, or structured reasoning about complex attack paths. Consequently, its lifecycle is best used as a primary execution layer that can be complemented by contemporary tools and specialized analytical frameworks.

5.2 OSSTMM – Quantitative Security Testing

The Open Source Security Testing Methodology Manual is a methodological reference that approaches security testing from a more formal and structured perspective. Unlike frameworks that mainly describe the operational flow of a penetration testing engagement, OSSTMM places strong emphasis on the idea of testing as a measurable and repeatable

activity. Its objective is not limited to guiding the tester through phases of execution, but also extends to defining a more systematic way of assessing operational security [4].

Within the context of this thesis, OSSTMM is considered relevant because it introduces a perspective that differs from more execution-oriented frameworks such as PTES. The significance of this methodology lies in its emphasis on methodological precision, empirical measurement, and the view of security testing as a structured process. These attributes are particularly valuable within corporate settings where repeatability, traceability, and operational consistency are established as fundamental requirements. Nevertheless, the inherent formality of the framework presents specific obstacles when applied to the dynamic nature of practical penetration testing engagements.

5.2.1 General Structure

OSSTMM is structured as a comprehensive methodology for security testing and analysis. It is not limited to a single technical domain, but is designed to address different areas of operational security. The methodology organizes testing activities through a structured approach that aims to define how security can be evaluated across multiple channels and interaction points [4]. This broad orientation makes OSSTMM different from frameworks that focus only on a specific type of asset, such as web applications, or only on the practical sequence of a penetration testing engagement.

A central characteristic of OSSTMM is its attention to measurement. The methodology introduces the idea that security testing should not only identify weaknesses, but also support a more objective evaluation of the tested environment. This approach is relevant because it attempts to reduce ambiguity in assessment activities and to provide a more consistent basis for interpreting test results. In this sense, OSSTMM is closer to a security testing methodology than to a simple operational checklist.

The general structure of OSSTMM is therefore broader and more formal than many traditional penetration testing references. By prioritizing a model based on consistent results and detailed records, the methodology ensures that testing outcomes can be effectively compared over time. This orientation is highly relevant for organizations that use penetration testing to satisfy audit requirements or to measure security progress over an extended period. Despite these advantages, the extensive requirements for formalization can make the framework less suitable for engagements that prioritize speed and direct technical exploitation.

5.2.2 Main Strengths

One of the main strengths of OSSTMM is its methodological rigor. The framework attempts to provide a structured and systematic basis for security testing, reducing the risk that the assessment depends only on the individual preferences of the tester. This is important in professional contexts where consistency and repeatability are required. By defining testing as a structured process, OSSTMM supports a more disciplined approach to security assessment [4].

A second strength is its emphasis on measurement. While many penetration testing

methodologies focus mainly on phases, techniques, and findings, OSSTMM introduces a stronger concern for how security testing results can be assessed and interpreted in a more formal way. This makes it valuable when the objective is not only to discover vulnerabilities, but also to understand the tested environment through a more systematic analytical lens.

Furthermore, the broad scope of the standard supports a multifaceted evaluation of security that is not restricted to a single technical field. This characteristic is useful in large-scale organizations where risk may be distributed across systems, processes, and communication channels. In this sense, the framework encourages security evaluations to be considered as part of a broader understanding of organizational posture, rather than as a narrow exercise limited to vulnerability discovery.

5.2.3 Main Limitations

Despite its strengths, OSSTMM also presents some limitations when considered in relation to practical penetration testing. The first limitation concerns accessibility. Because the methodology is formal and conceptually dense, it may be less immediately usable for testers who need direct operational guidance during an engagement. Compared to more practical frameworks, OSSTMM may require more interpretation before it can be translated into concrete testing activities.

A second limitation is its practical complexity. The emphasis on structure and measurement can be valuable, but it may also make the methodology difficult to apply fully in engagements constrained by time, scope, or operational priorities. In many enterprise security assessments, the evaluation process must balance methodological rigor with efficient technical execution. In such cases, OSSTMM may provide useful principles and analytical structure, but may be too demanding to apply as a complete operational model.

An additional limitation arises from the fact that the framework is not explicitly organized around the conventional penetration testing lifecycle. While the methodology provides a robust structure for systematic security testing, it offers less granular guidance for specific operational activities such as exploitation, post-exploitation, and the development of sophisticated attack paths. For this reason, its contribution is stronger in terms of measurement and methodological discipline than in terms of practical offensive workflow.

5.2.4 Evaluation According to the Comparison Criteria

According to the comparison criteria defined in Chapter 4, OSSTMM performs strongly in terms of level of formalization and methodological rigor. The systematic organization and focus on quantifiable security evaluations establish this framework as a highly formalized methodology within the present analysis. This characteristic is especially beneficial when the goal is to ensure that assessment activities remain repeatable, consistent, and traceable over time. By emphasizing empirical analysis and quantifiable metrics, the standard establishes a scientifically rigorous foundation for organizations that require objective and verifiable evidence of their actual security posture.

In terms of coverage scope, OSSTMM has a broad orientation. The framework is not

restricted to a specific technological field and remains applicable across diverse dimensions of security evaluation. This extensive reach is particularly significant within corporate environments where assessments must encompass a wide variety of asset categories and exposure surfaces. Nevertheless, such conceptual breadth does not inherently provide specific operational instructions for the execution of each individual stage of a penetration testing mission.

With respect to operational depth, OSSTMM offers a structured methodology, but its operational usability is different from that of PTES. It provides depth in terms of testing principles, structure, and measurement, but it is less directly aligned with the practical sequence of a penetration testing engagement. Therefore, its operational depth can be considered strong from a methodological perspective, but more limited from the point of view of hands-on exploitation workflows.

Regarding enterprise suitability, OSSTMM is relevant because of its emphasis on repeatable and measurable testing. These characteristics can be useful in organizations that require structured assessment processes and documented security evaluation. However, its complexity may reduce its immediate applicability in enterprise engagements that prioritize speed, practical validation, and direct remediation support. Consequently, the framework appears more appropriate as a secondary architectural reference rather than a solitary methodology for governing a comprehensive enterprise security evaluation.

In relation to automation, OSSTMM does not primarily focus on automated tools as a central element of its methodology. Nevertheless, its structured approach can be compatible with tool-assisted testing, especially when automated outputs are used as evidence within a broader assessment process. Automation can support data collection and technical verification, but the interpretation of results remains dependent on the methodology and on analyst judgment.

With respect to reporting and validation support, OSSTMM provides a strong contribution because of its emphasis on structured testing and measurable results. This can help organize findings and support more consistent evaluation. However, its reporting value depends on the ability to translate its formal concepts into deliverables that are understandable and useful for enterprise stakeholders.

Ultimately, OSSTMM shows moderate adaptability to contemporary hybrid security operations. Its formalized structure provides a valuable baseline for organizing evidence, supporting measurement consistency, and maintaining procedural traceability. However, its rigidity and methodological complexity make it less directly suited to fluid automation pipelines, iterative exploit validation, or LLM-assisted analytical interpretation. Therefore, the framework is best understood as a disciplined reference architecture that can reinforce a broader testing ecosystem, rather than as an agile standalone operational framework.

5.3 NIST SP 800-115 – Institutional and Procedural Perspective

NIST SP 800-115, *Technical Guide to Information Security Testing and Assessment*, represents an institutional and procedural reference for technical security testing activities.

Unlike frameworks that are primarily developed around the operational flow of penetration testing, NIST SP 800-115 provides broader guidance on how organizations can plan, conduct, and interpret technical security assessments. Its perspective is therefore particularly relevant when penetration testing is considered as part of a wider information security program rather than as an isolated technical engagement [1].

Within the context of this thesis, NIST SP 800-115 is relevant because it presents a structured, institutionally recognized view of security testing. Its value lies not only in the description of testing techniques, but also in its attention to planning, assessment methods, documentation, and the relationship between testing activities and organizational security management. This makes it a useful reference for enterprise environments, especially where penetration testing must be aligned with governance, risk management, and internal security processes.

5.3.1 General Structure

NIST SP 800-115 is organized as a technical guide for information security testing and assessment. It describes different assessment techniques, including review techniques, target identification and analysis techniques, target vulnerability validation techniques, and security assessment planning and reporting activities [1]. This structure shows that the document does not focus exclusively on penetration testing, but places it within a broader family of technical assessment activities.

A relevant aspect of its structure is the distinction between planning, execution, and post-assessment activities. The guide emphasizes that security testing should be prepared carefully, conducted within defined boundaries, and followed by structured analysis and reporting. This procedural perspective is important because it connects technical activity with organizational requirements, including authorization, documentation, and the communication of results.

NIST SP 800-115 also provides a high-level classification of testing and assessment techniques. This makes it useful as a general reference for understanding where penetration testing fits within the broader context of information security assessment. Rather than presenting penetration testing only as an offensive sequence, it frames it as one possible method among several techniques used to evaluate the security posture of systems and organizations.

5.3.2 Main Strengths

A primary strength of NIST SP 800-115 is its institutional authority. As a NIST publication, it provides a highly recognized and structured reference for organizations requiring authoritative guidance for security testing activities. This characteristic is especially significant in enterprise and regulated sectors, where assessment procedures must be thoroughly justified, documented, and aligned with established security management practices.

A second strength is its procedural clarity. NIST SP 800-115 does not only describe technical techniques, but also emphasizes planning, execution control, evidence handling, and reporting. This is important because enterprise penetration testing requires more than

technical capability alone. The assessment must be authorized, scoped, documented, and translated into outputs that can support remediation and decision-making. In this sense, the guide provides a useful bridge between technical testing and organizational security management.

The framework is also useful because it places penetration testing within the broader information security assessment landscape. It provides a clear taxonomy that explains the interplay between various technical evaluation strategies, including penetration testing and vulnerability scanning. For enterprise environments, such clarity is essential because it allows security assessments to be understood as constituent parts of a global assurance strategy rather than as disconnected technical tasks.

5.3.3 Main Limitations

The main limitation of NIST SP 800-115 is its relatively high-level nature when compared with frameworks that are more directly oriented toward operational penetration testing. Although it provides valuable procedural guidance, it does not describe the offensive workflow with the same level of practical detail offered by PTES. In particular, activities such as exploitation, post-exploitation, and attack-path development are not treated as central phases of a hands-on penetration testing lifecycle.

A second limitation derives from the broad scope of the publication. NIST SP 800-115 is not exclusively focused on enterprise penetration testing, but addresses information security testing and assessment more generally. This broader perspective is useful from an institutional standpoint, since it places penetration testing within a wider assessment context. However, it can also reduce the direct applicability of the guide when the objective is to support a complex operational engagement from reconnaissance to exploitation and final reporting.

A further limitation concerns the age of the document. Although many of its principles remain relevant, NIST SP 800-115 was published in 2008 and therefore does not explicitly address several developments that have become central in contemporary security assessment, including cloud-native infrastructures, modern attack simulation platforms, highly automated workflows, and LLM-assisted analytical support. For this reason, it should be interpreted as a foundational procedural reference rather than as a complete and up-to-date model for contemporary enterprise penetration testing.

5.3.4 Evaluation According to the Comparison Criteria

According to the comparison criteria defined in Chapter 4, NIST SP 800-115 performs strongly in terms of level of formalization and procedural structure. It provides clear guidance on how technical security assessments should be planned, conducted, and reported, making it particularly useful for organizations that require traceable and well-documented assessment processes.

In terms of coverage scope, NIST SP 800-115 has broad coverage across information security testing and assessment activities. It is not limited to penetration testing alone, but includes several techniques that can support the evaluation of systems and networks. This

broad scope is valuable in enterprise environments because it places penetration testing within a wider assessment ecosystem. However, the same broadness also means that its guidance is less specialized than frameworks focused specifically on offensive testing.

Regarding its operational depth, NIST SP 800-115 provides moderate support. It describes technical assessment techniques and procedural phases, but it does not provide highly detailed guidance for exploitation, post-exploitation, or attack-path development. Its operational depth is therefore stronger in planning, assessment organization, and reporting than in the practical execution of adversarial testing activities.

With respect to enterprise suitability, NIST SP 800-115 is highly relevant. Its institutional nature, procedural clarity, and emphasis on planning and reporting make it suitable for organizations that need to align penetration testing with governance, risk management, and internal security processes. It can help ensure that testing activities are not treated as isolated technical exercises, but as structured assessments integrated into a broader security program.

In relation to automation, NIST SP 800-115 recognizes the role of technical testing tools and automated assessment techniques, especially in vulnerability scanning and target identification activities [1]. However, automation is not developed as a central methodological theme. The guide supports the use of tools, but still requires human interpretation, validation, and contextual analysis in order to transform tool outputs into meaningful assessment results.

With respect to reporting and validation support, NIST SP 800-115 is particularly strong. It emphasizes the importance of documenting findings, communicating results, and using assessment outputs to support security improvement. This makes it highly valuable in enterprise contexts, where the effectiveness of a penetration test depends not only on technical discovery, but also on the ability to produce clear evidence, actionable recommendations, and useful information for decision-making.

Finally, NIST SP 800-115 has moderate adaptability to hybrid workflows. Its procedural structure can support the integration of different testing techniques, including manual analysis and tool-assisted assessment. However, it does not explicitly address newer forms of hybrid practice involving attack simulation platforms, LLM-assisted interpretation, or advanced attack-path reasoning. Therefore, its main contribution to a hybrid approach lies in its ability to provide institutional structure and procedural discipline, while other frameworks or tools may be needed to support more modern and adversarial aspects of enterprise penetration testing.

5.4 OWASP Testing Guide – Application Security Perspective

The OWASP Web Security Testing Guide represents a reference methodology specifically focused on web application security testing. Unlike broader frameworks that address penetration testing or security assessment as a whole, the OWASP Testing Guide concentrates on the application layer and provides structured guidance for identifying, analyzing, and validating vulnerabilities in web applications and related components [3].

Within the context of this thesis, OWASP WSTG is relevant because web applications are a central component of many enterprise environments. Corporate systems often expose web portals, authentication interfaces, APIs, administrative panels, and customer-facing services, making application security a frequent and important area of penetration testing. For this reason, although OWASP WSTG does not provide a complete methodology for enterprise penetration testing as a whole, it offers strong and detailed support for one of its most common and critical domains.

5.4.1 General Structure

The OWASP Web Security Testing Guide is organized as a structured reference for testing the security of web applications. It provides guidance across several testing areas, including information gathering, configuration and deployment management, identity management, authentication, authorization, session management, input validation, error handling, cryptography, business logic, client-side testing, and API testing [3]. This organization reflects the application-centered nature of the framework and its focus on the main categories of weaknesses that commonly affect web systems.

A relevant characteristic of OWASP WSTG is its task-oriented structure. Rather than presenting only general principles, the guide describes specific areas that should be considered during a web application assessment. This makes it particularly useful for testers who need to conduct systematic checks across different parts of an application. The framework therefore supports both coverage and consistency in application-layer testing.

At the same time, the structure of OWASP WSTG is different from a full penetration testing lifecycle. It does not primarily focus on enterprise-wide scoping, post-exploitation, internal movement, or governance integration. Instead, its value lies in providing depth within a clearly defined technical domain. For this reason, OWASP WSTG should be interpreted as a specialized methodological reference for application security rather than as a complete end-to-end framework for enterprise penetration testing.

5.4.2 Main Strengths

The primary strength of OWASP WSTG is its detailed focus on the modern web ecosystem. Compared with broader frameworks, it provides more detailed guidance on application-layer vulnerabilities and testing activities. This makes it particularly valuable in engagements that involve web applications, APIs, authentication flows, session-handling mechanisms, input-processing logic, or business-logic components.

A second strength is its practical usability. The guide is structured around recognizable testing categories and can therefore be used as a checklist-like reference during application assessments. This does not mean that its use should be mechanical, but it makes the methodology useful for ensuring that important testing areas are not overlooked. In enterprise contexts, where web applications may be numerous and heterogeneous, this structured approach can support consistency across different assessments.

OWASP WSTG is also valuable because it is widely recognized in the application security community. Its terminology and testing categories provide a common reference for

testers, developers, and security teams. This shared language can improve communication between technical and non-technical stakeholders, especially when findings need to be explained, prioritized, and remediated within software development or application security processes.

5.4.3 Main Limitations

The main limitation of OWASP WSTG is its narrower scope compared with broader penetration testing frameworks. The guide is specifically designed for web application security testing and does not attempt to cover all dimensions of enterprise penetration testing. As a result, it provides limited support for internal infrastructure assessment, physical security, wireless testing, social engineering, lateral movement, and enterprise-wide post-exploitation activities.

A second limitation is that OWASP WSTG is not primarily designed as a governance or risk management framework. Although its findings clearly support remediation and security improvements, the guide does not provide a comprehensive model for integrating application security results into broader enterprise decision-making processes. In this sense, it is highly useful at the technical testing level, but it may need to be combined with other frameworks or internal processes in order to support enterprise-level prioritization and governance.

A further limitation concerns the need for contextual interpretation. The guide can help identify relevant testing areas and potential weaknesses, but the practical impact of an application vulnerability still depends on the specific environment, the affected functionality, the application's business role, and the possibility of chaining the issue to other weaknesses. Therefore, OWASP WSTG provides strong technical coverage, but the evaluation of real-world risk remains dependent on analyst judgment and contextual validation.

5.4.4 Evaluation According to the Comparison Criteria

According to the comparison criteria defined in Chapter 4, OWASP WSTG provides a strong contribution within the specific domain of web application security testing. Its value is not that of a complete enterprise penetration testing methodology, but of a specialized reference that can strengthen the application security component of a broader assessment process.

In terms of coverage scope, OWASP WSTG has a specialized rather than comprehensive scope. It covers web application security in considerable detail, but it does not address the full range of activities that may be involved in enterprise penetration testing. For this reason, its coverage is deep but domain-specific. It is highly relevant for application security assessments, but insufficient as the only methodological reference for enterprise-wide penetration testing.

With respect to operational depth, OWASP WSTG is one of the strongest frameworks considered in this comparison for web-oriented testing. Its testing categories provide concrete guidance for examining different aspects of application security, from authentication

and authorization to input validation, business logic, and API testing. However, this operational depth is concentrated in the application layer and does not extend equally to infrastructure, internal network, physical, or broader adversarial testing.

Regarding the level of formalization, OWASP WSTG provides a clear and organized structure based on testing categories and objectives. It is formalized enough to support systematic assessment and repeatability within web application testing. Nevertheless, it does not define a full engagement lifecycle in the same way as PTES, nor does it provide a measurement-oriented model comparable to OSSTMM. Its formalization is therefore strong within its domain, but limited when considered as a general penetration testing methodology.

From the perspective of enterprise suitability, OWASP WSTG is highly relevant but partial. It is suitable for enterprise environments because web applications and APIs are frequently central to organizational exposure. However, its enterprise applicability depends on integration with broader methodologies that can cover scoping, infrastructure testing, reporting, governance, and cross-domain attack paths. In this sense, OWASP WSTG is best understood as a specialized component of an enterprise penetration testing methodology rather than as a complete framework on its own.

In relation to automation, OWASP WSTG is compatible with tool-assisted testing, especially in areas such as information gathering, vulnerability discovery, configuration review, and input validation. Automated tools can support many application testing activities, but the guide itself does not reduce testing to automated scanning. Many relevant weaknesses, especially those involving business logic, authorization flaws, or chained vulnerabilities, require manual analysis and contextual reasoning.

With respect to reporting and validation support, OWASP WSTG can support structured documentation by organizing findings according to recognizable testing categories. This can help testers present application security issues in a systematic way and relate them to specific areas of the application. However, the transformation of findings into enterprise-level remediation priorities still requires additional interpretation, especially when business impact and organizational risk must be assessed.

Finally, OWASP WSTG has good adaptability to hybrid workflows when the focus is application security. It can be combined with automated scanners, manual testing, code review outputs, and analyst reasoning. It can also support LLM-assisted interpretation in limited contexts, for example when organizing findings, mapping them to testing categories, or reasoning about possible exploitation chains. However, because its scope is domain-specific, its contribution to hybrid enterprise penetration testing is strongest when it is integrated with broader frameworks such as PTES, NIST SP 800-115, or adversarial models such as MITRE ATT&CK.

5.5 MITRE ATT&CK – Adversarial Knowledge and Attack Modeling

MITRE ATT&CK is different from the other references considered in this chapter because it is not a penetration testing methodology in the strict sense. The framework does

not provide a complete procedural lifecycle for planning, executing, and reporting a professional penetration testing engagement. Instead, it provides a structured knowledge base of adversary tactics, techniques, and procedures that can be used to reason about attacker behavior and to support threat-informed security assessment activities [5].

Within the context of this thesis, MITRE ATT&CK is relevant because enterprise penetration testing often requires more than the identification of isolated vulnerabilities. In complex environments, analysts must also understand how an attacker could move from an initial entry point to broader compromise objectives. MITRE ATT&CK can support this kind of reasoning by providing a common language for describing adversarial behavior and by helping analysts relate technical findings to plausible attack paths. For this reason, it is considered here as a complementary framework that can strengthen the analytical dimension of penetration testing.

5.5.1 General Structure

MITRE ATT&CK organizes adversary behavior through a structured model based on tactics and techniques. Tactics represent the higher-level objectives that an attacker may pursue during an intrusion, such as initial access, execution, persistence, privilege escalation, defense evasion, credential access, discovery, lateral movement, collection, command and control, exfiltration, and impact. Techniques represent “how” an adversary achieves a tactical goal by performing an action [5].

This structure makes MITRE ATT&CK particularly useful for representing attacks as sequences of actions rather than as isolated technical events. Instead of focusing only on the presence of a vulnerability, the framework helps describe how a weakness, misconfiguration, or exposed service may fit within a broader adversarial scenario. This is especially relevant in enterprise environments, where the practical impact of a finding often depends on how it can be combined with other conditions and used to progress through the target environment.

Another important characteristic of MITRE ATT&CK is its role as a shared reference language. By organizing adversary behavior into commonly recognized tactics and techniques, it allows analysts, defenders, and decision-makers to discuss attack scenarios in a more structured way. In penetration testing, this can support the interpretation of findings, the construction of attack hypotheses, and the communication of possible compromise paths to technical and managerial stakeholders.

5.5.2 Main Strengths

The main strength of the MITRE ATT&CK framework lies in its capacity to support adversarial reasoning. While many frameworks focus on testing phases, procedures, or specific technical checks, MITRE ATT&CK helps analysts think in terms of attacker objectives and behaviors. This is particularly valuable in penetration testing because a realistic assessment should not only identify weaknesses, but also consider how those weaknesses could be used as part of a broader attack sequence.

A second strength is its usefulness for attack-path understanding. Enterprise environments are often characterized by multiple systems, trust relationships, credentials, internal services, and segmented networks. In such contexts, the severity of a finding may depend on whether it enables further actions, such as privilege escalation, lateral movement, or access to sensitive resources. MITRE ATT&CK provides a structured way to reason about these possible developments and to connect individual findings to more realistic compromise scenarios.

MITRE ATT&CK is also valuable because it provides a common taxonomy for adversarial behavior. This can improve communication between penetration testers, blue teams, threat intelligence analysts, and security managers. Findings can be described not only in terms of vulnerability names or technical symptoms, but also in relation to recognized adversarial techniques. This makes the framework useful for linking penetration testing results with detection engineering, defensive validation, and threat-informed security improvement.

5.5.3 Main Limitations

The primary constraint of the MITRE ATT&CK framework resides in its specialized nature as a behavioral taxonomy rather than an exhaustive, end-to-end penetration testing methodology. It does not define how to scope an engagement, how to conduct pre-engagement activities, how to perform vulnerability analysis, or how to produce a complete penetration testing report. For this reason, it cannot replace frameworks such as PTES or institutional references such as NIST SP 800-115 when the objective is to structure the full lifecycle of a penetration test.

A second limitation is that MITRE ATT&CK describes adversarial behavior, but does not automatically determine which techniques are relevant in a specific target environment. The presence of an adversarial technique in the repository provides a catalog of possibilities but does not guarantee its practical relevance to a specific engagement. Analysts must still evaluate the technical conditions of the environment, the available access level, the constraints of the test, and the evidence collected during the assessment.

A further limitation concerns the risk of using the framework in a checklist-like manner. If MITRE ATT&CK is treated only as a list of techniques to map against findings, its analytical value may be reduced. Its usefulness depends on the ability of the tester to interpret techniques in context and to relate them to plausible attack paths. Therefore, MITRE ATT&CK functions more as a structured knowledge repository that supports adversarial reasoning than as a mechanical substitute for expert judgment.

5.5.4 Evaluation According to the Comparison Criteria

According to the comparison criteria defined in Chapter 4, MITRE ATT&CK should be evaluated as a complementary adversarial knowledge framework rather than as a complete penetration testing methodology. Its strongest contribution lies in supporting attack modeling, adversarial reasoning, and the interpretation of findings within realistic compromise scenarios.

In terms of coverage scope, MITRE ATT&CK has broad coverage of adversary tactics and techniques, but this coverage is different from the lifecycle coverage provided by penetration testing methodologies. It does not cover pre-engagement, reporting, or governance activities in a complete way. Instead, it covers the behavioral dimension of attacks, making it particularly valuable for understanding how adversaries operate across different phases of an intrusion.

With respect to operational depth, MITRE ATT&CK provides useful technical and behavioral detail about adversarial techniques, but it does not provide a complete execution methodology for penetration testers. It can help analysts identify possible actions, hypotheses, and attack paths, but it does not define the full operational process through which an engagement should be conducted. Its operational depth is therefore strong in relation to adversarial behavior, but limited in relation to end-to-end penetration testing execution.

Regarding the level of formalization, MITRE ATT&CK is highly structured. Its organization into tactics, techniques, sub-techniques, platforms, and related information provides a clear and consistent model for describing adversary behavior. However, this formalization concerns the classification of attacker actions rather than the organization of a penetration testing engagement. For this reason, it is formalized as a knowledge base, not as a complete assessment methodology.

From the perspective of enterprise suitability, MITRE ATT&CK is highly relevant. Enterprise penetration testing often requires reasoning about lateral movement, privilege escalation, persistence, credential access, and other adversarial behaviors that may determine the real impact of an initial compromise. ATT&CK can help analysts connect technical findings to these broader scenarios and can support communication with defensive teams, especially when penetration testing results are used to improve detection and response capabilities.

In relation to automation, MITRE ATT&CK is compatible with tool-assisted and threat-informed workflows. Many security tools, attack simulation platforms, and detection engineering activities use ATT&CK as a reference model for organizing adversarial techniques. In penetration testing, this compatibility can support the mapping of findings, the organization of attack hypotheses, and the alignment between offensive testing and defensive validation. However, the framework itself does not replace the need for manual verification and contextual interpretation.

With respect to reporting and validation support, MITRE ATT&CK can improve the way findings are contextualized. Mapping a finding or attack step to a recognized tactic or technique can help explain why it matters and how it may contribute to a broader compromise path. However, ATT&CK does not provide a complete reporting methodology, and its use in reporting must be integrated with other frameworks or organizational reporting standards.

Finally, MITRE ATT&CK has strong adaptability to hybrid workflows. It can be combined with traditional penetration testing methodologies, automated tools, threat intelligence, detection engineering, and analyst reasoning. It is also particularly suitable for supporting LLM-assisted analysis, since its structured representation of adversarial behavior can help organize findings and reason about possible attack paths. However, this

support remains analytical and complementary: ATT&CK can guide interpretation, but it cannot independently validate technical exploitability or replace human judgment.

5.6 Comparative Discussion

The analysis developed in the previous sections shows that the selected frameworks do not address penetration testing from the same perspective. Each of them emphasizes different aspects of security assessment, ranging from operational execution to formal measurement, institutional procedure, application security, and adversarial behavior modeling. For this reason, the comparison does not support the idea of a single framework being sufficient in all circumstances. Instead, it highlights the need to understand the specific contribution that each framework can provide within an enterprise penetration testing context.

This comparative discussion therefore focuses on the main differences and complementarities that emerge from the analysis. The objective is not to produce a rigid ranking of the frameworks, but to clarify how their respective strengths and limitations can inform a more integrated methodological perspective. This is particularly relevant for enterprise environments, where penetration testing often requires operational structure, technical depth, governance alignment, automation support, and contextual interpretation at the same time.

The main results of the framework comparison are summarized in Tables 5.1 and 5.2.

5.6.1 Key Differences Among Frameworks

The first major difference among the selected frameworks concerns their scope. PTES is specifically oriented toward the penetration testing lifecycle and provides a practical sequence of phases that reflects the way many engagements are organized. OSSTMM, by contrast, adopts a broader and more formal view of security testing, with stronger attention to measurement and repeatability. NIST SP 800-115 also has a broad scope, but its contribution is mainly procedural and institutional, since it places penetration testing within the wider context of information security testing and assessment.

OWASP WSTG differs from these frameworks because of its domain-specific nature. Its scope is narrower, but deeper within the area of web application security. This makes it particularly useful when the target includes web applications, APIs, authentication mechanisms, or application-layer business logic. MITRE ATT&CK differs even more clearly, because it is not a penetration testing methodology in the strict sense. Its value lies in the organization of adversarial behavior and in its ability to support attack modeling and threat-informed reasoning.

A second relevant difference concerns operational depth. PTES and OWASP WSTG provide the strongest practical support for hands-on testing, although in different ways. PTES supports the overall organization of a penetration testing engagement, while OWASP WSTG provides detailed technical guidance for application security testing. OSSTMM offers depth in terms of methodological structure and measurement, but is less directly aligned with offensive execution. NIST SP 800-115 provides procedural depth, especially

Table 5.1: Qualitative comparison of the selected frameworks: methodological and enterprise dimensions.

Framework	Coverage scope	Operational depth	Formalization	Enterprise suitability
PTES	Broad lifecycle coverage	High	Medium	High, if integrated with governance processes
OSSTMM	Broad security testing scope	Medium	High	Medium to high, depending on practical applicability
NIST SP 800-115	Broad assessment scope	Medium	High	High, especially in procedural and regulated contexts
OWASP WSTG	Application-specific scope	High within web application testing	Medium to high	High for application-layer assessments
MITRE ATT&CK	Broad adversarial behavior coverage	High for attack modeling	High as a knowledge base	High as a complementary analytical reference

in planning and reporting, but remains less detailed in exploitation-oriented activities. MITRE ATT&CK provides technical and behavioral depth regarding adversarial techniques, but does not define the complete workflow of a penetration test.

A further difference concerns formalization. OSSTMM is the framework with the strongest formal and measurement-oriented perspective. NIST SP 800-115 is also highly structured, but its formalization is more procedural than quantitative. PTES is less formal from a measurement point of view, but it provides a clear lifecycle structure. OWASP WSTG is formalized within the specific domain of web application testing, while MITRE ATT&CK is highly structured as a knowledge base of tactics and techniques, rather than as a testing methodology. These differences show that formalization can take different forms depending on the purpose of the framework.

5.6.2 Complementary Aspects

The comparison also shows that the selected frameworks can be interpreted as complementary rather than mutually exclusive. PTES can provide the operational backbone of a penetration testing engagement, since it defines a clear sequence from pre-engagement activities to reporting. However, it may benefit from the support of other references when

Table 5.2: Qualitative comparison of the selected frameworks: automation, validation, and hybrid adaptability.

Framework	Automation support	sup-	Reporting and validation support	Hybrid adaptability
PTES	Compatible with tool-assisted workflows, but automation is not central		Good, especially through exploitation and reporting phases	Good as an operational backbone for integrated workflows
OSSTMM	Compatible, but mostly indirect		Strong through structured and measurable testing	Moderate; useful as a formal methodological component
NIST SP 800-115	Supported, but not developed as a central theme		Strong through planning, documentation, and reporting	Moderate; strong procedural support, but limited on modern hybrid practices
OWASP WSTG	Strong compatibility with web application testing tools		Good for organizing and validating application-layer findings	Good within application-security workflows
MITRE ATT&CK	Strong compatibility with threat-informed tools and attack simulation platforms		Good for contextualizing findings through adversarial techniques	Strong for adversarial reasoning and analytical workflows

stronger formalization, measurement, application-specific guidance, or adversarial modeling is required.

OSSTMM can complement PTES by introducing a more rigorous and measurable view of security testing. Its value is particularly evident when the assessment must support consistency, repeatability, and structured interpretation of results. At the same time, because OSSTMM may be complex to apply as a complete operational model, it can be more effective when used selectively, as a methodological reference that strengthens the discipline of the assessment rather than replacing the practical flow of the engagement.

NIST SP 800-115 can provide institutional and procedural support, especially in contexts where penetration testing must be connected to governance, risk management, authorization, documentation, and remediation processes. Its contribution is therefore particularly useful at the organizational level. While it may not provide the same offensive depth as PTES, it helps frame penetration testing as part of a broader security assessment program.

OWASP WSTG adds a specialized technical dimension. In enterprise environments, web applications and APIs often represent critical exposure points, and a general penetration testing methodology may not provide enough detail to assess them properly. In this respect, OWASP WSTG can complement broader frameworks by providing structured and detailed

guidance for application-layer testing.

MITRE ATT&CK complements the other frameworks by supporting adversarial reasoning. It helps analysts interpret findings in relation to tactics, techniques, and possible attack paths. This is especially useful when the objective is not only to identify vulnerabilities, but to understand how they may be combined in a realistic compromise scenario. For this reason, MITRE ATT&CK is particularly relevant for connecting penetration testing with threat-informed defense, detection engineering, and attack-path analysis.

5.6.3 Relevance for Enterprise Penetration Testing

From an enterprise penetration testing perspective, the comparison suggests that no single framework fully satisfies all the methodological and operational needs identified in this thesis. Enterprise assessments require a combination of lifecycle structure, practical execution guidance, formalization, reporting support, application-layer detail, and adversarial reasoning. Each selected framework contributes to some of these needs, but none of them covers all dimensions completely.

PTES appears particularly relevant as an operational foundation because it structures the main phases of a penetration testing engagement. However, it does not fully address measurement, governance integration, or newer hybrid workflows. OSSTMM contributes methodological rigor and measurability, but may be less accessible as a practical guide for fast-moving engagements. NIST SP 800-115 provides institutional structure and procedural discipline, but is less detailed in offensive execution and does not directly address several modern testing practices. OWASP WSTG is highly valuable for application security, but its scope is too narrow to represent the whole enterprise testing process. MITRE ATT&CK is highly useful for adversarial modeling, but it remains complementary rather than procedural.

For these reasons, the most effective approach in enterprise contexts is not the isolated adoption of a single framework, but the construction of an integrated and threat-informed methodological perspective. Such an approach can use PTES as the operational lifecycle backbone, NIST SP 800-115 as a procedural and governance-oriented reference, OWASP WSTG as a specialized guide for application-layer testing, OSSTMM as a source of formal and measurable testing principles, and MITRE ATT&CK as support for adversarial reasoning and attack-path interpretation.

This conclusion is consistent with the broader direction of the thesis. Enterprise penetration testing is not only a matter of executing technical checks, but of organizing complex assessments in a way that remains methodologically coherent, operationally realistic, and analytically meaningful. The comparative analysis therefore supports the need for an integrated approach in which frameworks, automated tools, analytical support, and human validation are combined rather than treated as separate or competing elements.

Chapter 6

Automation, LLM-Assisted Support and Hybrid Methodological Approaches

The previous chapter showed that no single penetration testing framework is sufficient to address all the methodological and operational needs of enterprise environments. While structured frameworks provide useful guidance, practical assessments also depend on the ability to manage large volumes of technical information, interpret tool-generated outputs, and reason about realistic attack paths. For this reason, the discussion must now move from the comparison of methodological references to the operational elements that can support penetration testing activities in practice.

This chapter examines the role of automation, LLM-assisted support, and human validation within enterprise penetration testing workflows. Automation is first analyzed as a means to improve efficiency, scalability, and initial technical coverage, while also considering the limitations arising from noisy outputs, limited visibility, and a lack of contextual understanding. The chapter then discusses the possible contribution of large language models as analytical support tools, especially in relation to result interpretation, finding correlation, and attack-path reasoning. Finally, the chapter argues that these elements should not be treated as substitutes for expert judgment, but as components of a hybrid methodological approach in which frameworks, automated tools, LLM-assisted support, and human validation operate together.

6.1 Role of Automation in Enterprise Penetration Testing

Automation plays an important role in enterprise penetration testing because modern organizational environments are often too large, heterogeneous, and dynamic to be assessed only through manual techniques. Within these complex environments, automated tools

help analysts collect information, identify exposed services, detect known vulnerabilities, and organize technical evidence in a more efficient way. Their value is therefore primarily operational: they allow the testing team to increase visibility over the target environment and to perform repetitive tasks at a scale that would be difficult to sustain manually [1, 3].

At the same time, automation should not be interpreted as a replacement for the penetration tester. Automated tools can support several phases of the assessment, but their outputs require interpretation, verification, and contextualization. This is especially important in enterprise environments, where the actual significance of a finding often depends on the affected asset, the surrounding architecture, the business role of the system, and the possibility of combining the issue with other weaknesses. For this reason, automation is best understood as an enabling component of the penetration testing workflow rather than as an autonomous assessment capability.

6.1.1 Operational Advantages

The primary advantage of automation is the ability to accelerate repetitive and time-consuming tasks. Activities such as host discovery, port scanning, service enumeration, technology fingerprinting, and basic vulnerability detection can be partially automated, allowing analysts to obtain an initial view of the target surface more quickly. This is particularly useful during the early phases of an engagement, when the testing team must reduce uncertainty and identify areas that deserve closer manual analysis [1].

Automation also supports consistency in the execution of technical checks. When properly configured, tools can apply the same scanning logic across multiple assets, reducing the risk that similar systems are assessed in inconsistent ways. This does not eliminate the need for human judgment, but it provides a more stable operational baseline from which the analyst can proceed. In enterprise engagements, where the number of assets may be significant, this consistency can help maintain control over the assessment process.

Another operational advantage concerns the organization of evidence. Many tools generate structured outputs that can be used to support documentation, reporting, and later validation. This can be useful when several findings must be reviewed, grouped, prioritized, and translated into a coherent report. However, the usefulness of these outputs depends on the analyst's ability to distinguish relevant evidence from noise and to verify whether the reported issues are actually meaningful in the specific environment under assessment.

6.1.2 Scalability and Efficiency

The contribution of automation becomes especially relevant when penetration testing is conducted in large and complex enterprise environments. Corporate infrastructures often include external services, internal networks, cloud resources, web applications, authentication systems, remote access services, and multiple business units. In such scenarios, a manual-only approach would often be too slow and incomplete, especially when the assessment must be conducted within limited time and operational constraints.

Automation improves scalability by allowing the testing team to process larger volumes of technical information in a shorter time. This does not necessarily mean that the assessment becomes deeper, but it helps broaden the initial coverage of the target environment. For example, automated scanning can support the identification of exposed hosts, outdated services, insecure configurations, or known vulnerabilities that may later be investigated manually. In this sense, automation contributes to breadth, while human analysis remains necessary for depth.

Efficiency is also important from a methodological perspective. By reducing the time spent on repetitive discovery and detection activities, automation allows analysts to dedicate more attention to interpretation, validation, and attack-path reasoning. This is particularly relevant in enterprise penetration testing, where the most valuable results often emerge not from the isolated detection of a vulnerability, but from understanding whether and how that vulnerability can contribute to a realistic compromise scenario.

6.1.3 Common Limitations in Practice

Despite its operational advantages, automation presents several limitations in practical penetration testing. One of the most common issues is incomplete coverage. Automated tools can only detect what they are designed to identify and what they are able to observe from their available perspective. As a result, they may miss vulnerabilities that depend on business logic, complex authorization rules, chained conditions, or environment-specific configurations.

A second limitation concerns noisy outputs. Automated tools may produce false positives, duplicate findings, low-relevance alerts, or results that are technically correct but not practically exploitable in the tested environment. This can create additional work for the analyst, who must review and filter the results before they can be used as reliable evidence. In this sense, automation can increase efficiency, but it can also introduce analytical overhead if its outputs are not carefully managed.

The most important limitation is the absence of contextual understanding. Automated tools can detect technical conditions, but they generally cannot evaluate their real meaning within the organization. They do not understand the business role of an asset, the strategic importance of a system, the operational constraints of the engagement, or the way different weaknesses may interact across an enterprise architecture. For this reason, automated results should be treated as inputs to the assessment process, not as final conclusions. Their real value depends on human interpretation, technical verification, and integration within a structured methodological workflow.

6.2 Automated Tools in Practice

In practical penetration testing engagements, automation is commonly implemented through different categories of tools, each supporting a specific part of the assessment workflow. These tools do not have the same purpose: some are mainly used to discover and map the target environment, others support information gathering, vulnerability detection,

or the validation of possible attack paths. For this reason, automated tools should not be treated as a single homogeneous category, but as a set of operational instruments that contribute differently to the penetration testing lifecycle.

In enterprise environments, the practical relevance of these tools is particularly evident. Large infrastructures often require the analyst to collect and process information from multiple technical sources before meaningful conclusions can be drawn. Automated tools can therefore support the initial structuring of the assessment, helping the testing team identify exposed assets, recognize relevant technologies, detect known weaknesses, and organize evidence. However, their contribution remains dependent on correct configuration, appropriate scope definition, and human interpretation of the results.

6.2.1 Network Scanning and Mapping Tools

Network scanning and mapping tools support the identification of hosts, services, ports, and communication surfaces within the target environment. These tools are commonly used during the early phases of an assessment, when the analyst needs to understand which systems are reachable, which services are exposed, and how the target surface is structured. In this sense, they provide an initial technical map that can guide later analysis and testing activities [1].

Their main value lies in reducing uncertainty. In large enterprise environments, manual discovery alone may be insufficient to identify the full set of relevant assets within the available time. Automated scanning can help reveal exposed services, unexpected open ports, outdated protocols, or systems that require closer investigation. This does not mean that the output of the tool is complete or final, but it provides a baseline from which the analyst can decide where to focus additional effort.

At the same time, network scanning requires careful interpretation. A discovered service does not automatically represent a vulnerability, and an open port may have different meanings depending on the role of the system, the network segment, and the exposure context. Moreover, scanning activities must remain aligned with the rules of engagement, since aggressive scans may generate operational impact in sensitive environments. For this reason, network scanning is useful as a discovery and mapping mechanism, but its results must be validated and contextualized before they can support security conclusions.

6.2.2 OSINT Tools

OSINT tools support the collection and organization of information from public and semi-public sources. In penetration testing, open-source intelligence can help identify domains, subdomains, exposed services, technologies, leaked information, employee-related data, third-party dependencies, or other elements that may contribute to the external attack surface. This is especially relevant in black-box and external assessments, where the tester initially operates with limited knowledge of the target environment [3].

The main contribution of OSINT tools is their ability to accelerate information gathering and correlation. Instead of relying exclusively on manual searches, analysts can use automated support to collect data from multiple sources and obtain a broader initial picture

of the organization's public exposure. This can reveal assets that are not immediately visible, highlight relationships between systems, or provide clues about technologies and services used by the target.

However, OSINT results require careful filtering. Publicly available information may be outdated, incomplete, duplicated, or unrelated to the actual scope of the engagement. In addition, not all information discovered through OSINT is operationally relevant. The analyst must therefore distinguish between useful evidence and background noise, ensuring that collected data is interpreted in relation to the objectives and boundaries of the assessment. OSINT tools are valuable because they expand visibility, but they do not independently determine the security significance of the information collected.

6.2.3 Vulnerability Scanners

Vulnerability scanners are among the most widely used automated tools in technical security assessments. Their purpose is to identify known weaknesses, insecure configurations, missing patches, outdated software versions, and other conditions that may expose systems to risk. In enterprise penetration testing, scanners are particularly useful because they allow analysts to assess large numbers of assets in a relatively short amount of time [1, 3].

The main advantage of vulnerability scanners is breadth. They can quickly provide an overview of potential weaknesses across multiple systems, helping the analyst identify findings that may deserve further validation. This is especially useful when the target environment includes many hosts, services, or applications. Scanner outputs can also support prioritization by highlighting issues associated with known vulnerabilities, severity ratings, or affected services.

Nevertheless, scanner outputs should never be treated as final conclusions. Automated vulnerability detection is affected by false positives, false negatives, and limited contextual awareness. A scanner may report a vulnerability that is not exploitable in the specific environment, or it may fail to detect weaknesses that depend on business logic, chained conditions, or authenticated access. For this reason, vulnerability scanning should be understood as an initial detection mechanism that supports further analysis, not as a substitute for penetration testing validation.

6.2.4 Automated Validation and Attack Simulation Platforms

Automated validation and attack simulation platforms represent a more advanced use of automation in penetration testing and security assessment workflows, especially when validation activities are organized around adversarial behaviors, tactics, and techniques [5]. In contrast to traditional scanners, which mainly identify potential weaknesses, these platforms aim to verify whether certain attack steps or exploitation paths can be executed under controlled conditions. Their purpose is therefore closer to validation than to simple detection.

In enterprise environments, this type of automation can be useful when the assessment requires a more realistic understanding of exposure. Automated validation may help determine whether a reported weakness can actually be exploited, whether a specific

attack path is feasible, or whether existing controls are able to prevent or detect certain adversarial actions [5]. In this sense, these platforms can support the transition from vulnerability identification to attack-path reasoning.

However, automated validation also has important limitations. The fact that a platform can simulate or verify a specific attack step does not mean that it fully understands the business context, the operational constraints, or the broader implications of compromise. Moreover, automated simulations are still based on predefined assumptions, available techniques, and tool-specific logic. Their results must therefore be reviewed by analysts and integrated with manual verification, contextual reasoning, and the objectives of the engagement.

For this reason, automated validation and attack simulation platforms should be considered useful support tools rather than autonomous penetration testing solutions. They can improve the efficiency and realism of certain validation activities, but they do not replace the need for expert judgment. Their greatest value emerges when they are integrated into a structured methodology that combines automation, analyst reasoning, and human validation.

6.3 LLM-Assisted Support in Penetration Testing Activities

LLM-assisted support represents an emerging area of interest in penetration testing workflows. Unlike traditional automated tools, which usually operate through predefined checks, signatures, or scanning logic, large language models can support analysts in tasks that require interpretation, summarization, correlation, and reasoning over technical information. Their potential contribution is therefore not primarily related to the direct execution of attacks, but to the analytical processing of information produced during the assessment.

In enterprise penetration testing, this type of support may be relevant because analysts often have to manage large volumes of heterogeneous data. Tool outputs, scan results, configuration details, vulnerability descriptions, network information, and manual notes must be interpreted together in order to understand which findings are actually relevant. In this context, LLM-assisted support can help organize information, formulate preliminary hypotheses, and support reasoning about possible attack paths. However, this support must remain subordinate to human supervision and technical validation, since LLM-generated outputs cannot be considered reliable evidence on their own [6, 7].

6.3.1 Support for Result Interpretation

One possible use of LLM-assisted support concerns the interpretation of results generated by automated tools. Vulnerability scanners, network mapping tools, and other assessment platforms often produce outputs that are technically detailed but difficult to interpret immediately in relation to the overall objectives of the engagement. An LLM can help summarize these outputs, explain recurring patterns, and reorganize findings in a more

understandable form for the analyst.

This can be useful in enterprise environments, where the volume of findings may be high and where multiple tools may produce partially overlapping results. For example, LLM-assisted support may help distinguish between different categories of findings, group related issues, or provide an initial explanation of why a specific weakness may be relevant. This does not mean that the model validates the finding. Rather, it can support the analyst in transforming raw technical output into a more structured basis for further examination.

The value of this support lies mainly in reducing the cognitive effort required to process large amounts of textual and technical information. By helping the analyst navigate tool outputs more efficiently, LLMs may contribute to a faster preliminary understanding of the assessment results. Nevertheless, the interpretation remains provisional. Any conclusion regarding exploitability, impact, or remediation priority must still be verified through technical analysis and contextual evaluation.

6.3.2 Assistance in Attack-Path Reasoning

LLM-assisted support may also contribute to attack-path reasoning. In penetration testing, findings are rarely meaningful only in isolation. A vulnerability, misconfiguration, weak credential policy, exposed service, or insufficient segmentation control may become more relevant when combined with other conditions. Understanding these relationships requires reasoning about possible sequences of actions an attacker could follow to move from initial access to a broader compromise.

Large language models can assist this process by organizing known information into potential exploitation chains or by suggesting relationships between findings that warrant further investigation. For instance, they may help the analyst reason about how an exposed service, a credential-related issue, and an internal trust relationship could form a plausible attack path. This type of support is especially relevant when combined with adversarial knowledge references such as MITRE ATT&CK, which provide a structured vocabulary for describing attacker behavior [5].

However, LLM-assisted attack-path reasoning must be treated with caution. A model can suggest plausible connections, but it cannot independently verify whether those connections are technically feasible in the target environment. For this reason, its role should be understood as hypothesis generation rather than proof. The analyst remains responsible for validating whether the proposed attack path is realistic, authorized within the engagement scope, and supported by actual technical evidence.

6.3.3 Help in Correlating Findings

Another relevant contribution of LLM-assisted support concerns the correlation of findings from multiple sources. Enterprise penetration testing often involves data collected through different tools and activities, such as network scanning, OSINT, vulnerability scanning, manual testing, and documentation review. These sources may reveal related information, but the relationships between them are not always immediately visible.

An LLM can support the analyst by helping compare and organize information across different outputs. It may assist in identifying repeated patterns, grouping findings by affected asset, mapping technical weaknesses to broader categories, or highlighting possible dependencies between issues. This can be useful when the analyst needs to move from a collection of individual findings toward a more coherent understanding of the target environment.

Correlation support is particularly useful when dealing with noisy or redundant data. Automated tools may report similar findings in different ways, or they may produce results that need to be deduplicated and interpreted together. LLM-assisted support can help accelerate this preliminary organization, but it should not determine the final relevance of the findings. The analyst must still assess whether the correlations are technically valid and whether they reflect meaningful security relationships in the tested environment.

6.3.4 Use in Preliminary Validation and Analytical Triage

LLM-assisted support can also be useful during preliminary validation and analytical triage. In this context, the model may help the analyst review findings, identify which issues require closer attention, and formulate initial questions for further verification. This can be useful when the assessment produces a large number of potential weaknesses and the testing team must decide where to focus manual effort first.

For example, LLM-assisted triage may support the classification of findings according to technical category, affected asset, possible impact, or relation to known attack techniques. It may also help the analyst identify which findings appear incomplete, which require additional evidence, and which may be more relevant because of their position in the environment. This can improve the efficiency of the analysis phase, especially when tool outputs are extensive or heterogeneous.

Nevertheless, this role must remain preliminary. LLMs may assist in prioritizing attention, but they cannot replace the technical validation required to confirm exploitability or business impact. Their output should therefore be considered as an analytical aid that supports the tester's reasoning, not as a final decision mechanism. In a professional penetration testing context, validation must always rely on evidence, controlled verification, and expert judgment.

6.4 Limits of LLM-Assisted Support

Although LLM-assisted support can provide useful help in penetration testing workflows, its limitations must be considered carefully. Large language models can assist analysts in interpreting results, organizing technical information, and formulating preliminary hypotheses, but they do not provide independent technical verification. Their outputs are generated from linguistic and contextual patterns, rather than from direct interaction with the target environment. For this reason, LLM-assisted analysis should be treated as a support mechanism and not as a source of final evidence.

This distinction is particularly important in cybersecurity activities, where incorrect

assumptions may lead to fallacious conclusions, wasted effort, or inaccurate risk evaluation. In penetration testing, the value of a finding depends on evidence, exploitability, scope, and environmental context. LLM-generated outputs may help the analyst reason about these elements, but they cannot replace controlled validation or expert judgment. Therefore, the use of LLMs in penetration testing must be governed by caution, supervision, and explicit verification procedures [7, 8].

6.4.1 Hallucinations and Inaccurate Inferences

A primary limitation of LLM-assisted support is the risk that the model may generate hallucinated or inaccurate outputs. In this context, hallucination refers to the production of information that appears coherent and plausible, but is not grounded in the available evidence. This issue is particularly relevant in penetration testing because the model may describe a vulnerability, suggest an attack path, or infer a technical relationship that is not actually present in the tested environment [8].

For this reason, these deceptive inferences undermine the efficiency of the assessment workflow whenever the initial data set lacks high-fidelity technical metrics. If the model is asked to interpret scan results, correlate findings, or suggest exploitation paths, it may produce explanations that appear coherent while relying on assumptions that have not been technically verified. This can create a false sense of confidence and may lead the tester to overestimate the importance or feasibility of a finding.

Therefore, LLM-generated analytical insights should never be interpreted as definitive evidence. They may provide significant analytical support by facilitating hypothesis generation, structuring complex reasoning, and highlighting specific technical aspects for further validation; nevertheless, their model-derived inferences lack the evidentiary status required to stand as technical proof. Consequently, any statement concerning exploitability, systemic impact, or attack-path feasibility cannot rely on theoretical inference alone; it must be rigorously confirmed through manual analysis, environment-specific validation, and empirical evidence collected during the engagement.

6.4.2 Lack of Direct Environmental Awareness

Another important limitation is the lack of direct environmental awareness. Unless explicitly connected to verified data sources, an LLM does not know the real configuration of the target environment, the actual state of systems, the existence of compensating controls, or the operational constraints of the engagement. It can reason only on the basis of the information provided by the analyst and on the patterns learned during training.

This limitation is particularly relevant in enterprise penetration testing, where the practical significance of a vulnerability often depends on context. The same technical weakness may have very different implications depending on network segmentation, authentication mechanisms, asset criticality, business function, exposure level, and the possibility of combining it with other findings. An LLM may help organize these factors when they are provided, but it cannot independently observe or verify them.

As a result, LLM-assisted support should not be used as a substitute for environmental

analysis. The analyst must ensure that the model receives accurate and relevant information, and must still verify whether the resulting interpretation is compatible with the real target environment. Without this validation, the model may produce reasoning that is logically plausible but technically incorrect.

6.4.3 Dependence on Prompt Quality and Analyst Guidance

The quality of LLM-assisted outputs strongly depends on the quality of the prompt and on the analyst's ability to guide the interaction. The utility of large language models remains strictly dependent on the quality of the input data; consequently, a poorly structured or ambiguous prompt can trigger generic and misleading inferences, failing to provide the level of environment-specific grounding required for professional analysis. This makes prompt formulation an important practical factor in the effective use of LLM-assisted support.

In penetration testing, this dependence is particularly relevant because technical findings often require precise context. The analyst may need to specify the target type, affected service, observed evidence, tool output, engagement constraints, and the specific question being investigated. Without these details, the model may respond with general cybersecurity knowledge rather than with analysis that is useful for the specific engagement.

This limitation also means that LLM-assisted support depends heavily on the competence of the human user. An experienced analyst is more likely to formulate appropriate prompts, recognize weak or unsupported outputs, and ask follow-up questions that refine the analysis. Conversely, an inexperienced user may accept plausible but incorrect responses without sufficient verification. For this reason, the usefulness of LLMs in penetration testing is closely tied to the expertise of the analyst guiding the interaction.

6.4.4 Need for Expert Validation

Given these limitations, expert validation remains indispensable. LLMs may support interpretation, correlation, and preliminary reasoning, but their outputs must be checked against technical evidence and the actual conditions of the engagement. In professional penetration testing, the final assessment cannot rely on unverified model-generated conclusions, especially when findings may influence remediation priorities, risk acceptance, or organizational decisions.

Expert validation is necessary at several levels. First, the analyst must verify the technical accuracy of the information produced by the model. Second, the analyst must assess whether the suggested reasoning is compatible with the target environment and with the rules of engagement. Third, the analyst must determine whether a proposed finding or attack path has real security relevance and whether it can be supported by evidence.

For this reason, LLM-assisted support should be integrated into a human-supervised workflow. While the deployment of model-driven support can significantly accelerate preliminary analysis, alleviate cognitive load, and catalyze hypothesis generation, the final responsibility for technical validation and risk prioritization remains strictly within the purview of the expert testing team. This reinforces the broader argument of the thesis: LLMs can be useful in penetration testing only when they are combined with structured

methodology, technical verification, and expert human judgment.

6.5 Human Validation as a Critical Component

Human validation remains an indispensable pillar of modern penetration testing workflows; even when the engagement is heavily supported by automated utilities and LLM-assisted analysis, the final determination of exploitability and systemic risk strictly necessitates expert human oversight. Automation can accelerate discovery, scanning, and preliminary validation, while LLMs can support interpretation, correlation, and hypothesis generation. However, neither of these components can independently determine the actual security relevance of a finding. In professional penetration testing, conclusions must be grounded in evidence, technical verification, and contextual understanding.

This is especially important in enterprise environments, where the impact of a vulnerability cannot be evaluated only from a technical perspective. The same finding may have different levels of severity depending on the affected asset, its business function, the surrounding architecture, existing compensating controls, and the possibility of combining it with other weaknesses. For this reason, human analysts remain responsible for transforming tool outputs and model-assisted suggestions into validated security conclusions.

6.5.1 Contextual Interpretation

Contextual interpretation is one of the main reasons why human validation remains indispensable. Automated tools and LLM-assisted systems can identify or describe technical conditions, but they cannot fully understand the organizational context in which those conditions exist. Within complex enterprise environments, this specific context serves as the primary metric to determine whether a given finding possesses genuine operational relevance, how it should be prioritized within the risk hierarchy, and what kind of remediation-oriented strategy is appropriate.

For example, an exposed service, a weak configuration, or a known vulnerability may have different implications depending on whether it affects a production system, a test environment, a critical business application, or an isolated internal host. The technical condition may appear similar, but the associated risk can vary significantly. Human analysts are therefore required to evaluate findings in relation to asset criticality, network position, exposure level, authentication requirements, and business impact.

This interpretative role is also important when multiple findings interact. A single weakness may appear minor when considered in isolation, yet it can achieve significant operational relevance when combined with other security deficiencies, such as loose access controls, insufficient network segmentation, or accidental credential exposure. Identifying these relationships requires contextual reasoning that goes beyond the direct output of automated tools or LLM-assisted suggestions. Ultimately, an expert-led evaluation is indispensable to determine whether these theoretical attack paths are achievable in practice, ensuring that every validated vector remains completely aligned with the strategic objectives and constraints of the engagement.

6.5.2 Technical Verification

Technical verification is another essential element of human validation. In penetration testing, a finding should not be considered confirmed only because it is reported by a tool or suggested by an analytical support system. It must be checked against the actual behavior of the target environment and, when authorized by the rules of engagement, validated through controlled testing.

This verification process may involve reproducing a vulnerability, confirming the affected version of a service, checking whether a configuration is actually exploitable, or determining whether compensating controls reduce the practical feasibility of an attack. The objective is not only to determine whether a weakness exists, but also to understand whether it can be exploited under the conditions of the engagement. This distinction is central to penetration testing, since the value of the assessment depends on validating practical risk rather than merely listing potential exposures.

Technical verification also helps reduce the impact of false positives and misleading outputs. Automated scanners may report vulnerabilities that are not applicable to the specific environment, while LLM-assisted systems may infer relationships that appear plausible but are not technically supported. Human verification allows the testing team to distinguish between preliminary indications and validated findings. Consequently, this approach significantly enhances the reliability of the final assessment and strengthens the evidentiary foundation of the penetration testing report.

6.5.3 Risk Prioritization

Human expertise is also necessary for risk prioritization. Within complex enterprise environments, technical findings that appear identical from a purely systemic perspective often carry vastly different operational significance. Consequently, effective risk prioritization cannot rely solely on baseline vulnerability scores, but must account for the specific architectural placement and business role of the affected asset. Prioritization requires the analyst to consider exploitability, potential impact, asset criticality, exposure, likelihood of abuse, and remediation urgency. This process cannot be reduced to a mechanical reading of severity scores or automated classifications.

Automated tools may assign severity levels based on known vulnerability databases, technical characteristics, or predefined scoring systems. While these standardized metrics deliver a valuable baseline for preliminary reference, they frequently fail to encapsulate the complete, context-driven reality of the target organization. A vulnerability with a high generic severity may be less urgent if it affects an isolated system with strong compensating controls, while a medium-severity issue may become critical if it enables lateral movement, privilege escalation, or access to sensitive business processes.

Risk prioritization therefore requires human judgment. The analyst must evaluate which findings are most relevant to the organization and which remediation actions should be addressed first. This is particularly important in enterprise settings, where security teams often face resource constraints and cannot remediate every issue immediately. A well-prioritized penetration testing report helps the organization focus on the weaknesses

that create the most realistic and significant risk.

6.5.4 Final Assessment Responsibility

The final responsibility for the assessment necessarily remains with the analyst or the testing team. Automated tools and LLM-assisted systems can support the workflow, but they do not carry professional responsibility for the conclusions presented in the final report. In a penetration testing engagement, findings may influence remediation priorities, business decisions, risk acceptance, and security investments. For this reason, the final conclusions must be produced and validated by qualified human professionals.

This dimension carries profound significance within complex corporate networks, where business continuity must be rigorously preserved and specific operational constraints meticulously respected throughout the lifecycle of the security engagement. Human analysts must decide which activities are appropriate, which tests should be limited or avoided, and how evidence should be collected without causing unnecessary disruption. Consequently, managing these parameters is uniquely critical in enterprise settings. The execution of offensive testing activities must be constantly calibrated to mitigate systemic risk, ensuring that engagement-specific boundaries protect business continuity and operational stability.

Ultimately, human validation provides the accountability that automated and LLM-assisted components cannot offer. It ensures that conclusions are evidence-based, technically verified, contextually meaningful, and communicated in a form that supports organizational decision-making. This reinforces the central argument of the chapter: automation and LLM-assisted support can increase the efficiency and analytical capacity of penetration testing, but their value depends on integration within a workflow governed by human expertise and professional responsibility.

6.6 Toward a Hybrid Methodological Approach

The previous sections have shown that automation, LLM-assisted support, and human validation should not be interpreted as separate or competing elements. Each of them contributes to a different part of the penetration testing workflow. Automation improves scale, speed, and initial technical visibility; LLM-assisted support may help analysts interpret information, correlate findings, and reason about possible attack paths; human validation remains necessary to verify technical feasibility, contextual relevance, and final assessment conclusions. For this reason, the most coherent perspective is not based on replacing one component with another, but on integrating them within a hybrid methodological approach.

A hybrid approach is particularly relevant in enterprise penetration testing because organizational environments require both methodological discipline and operational flexibility. Structured frameworks provide the baseline for organizing the assessment, defining phases, and maintaining coherence across the engagement. Automated tools support the collection and processing of technical information. LLM-assisted support may help manage complexity by assisting with interpretation and hypothesis generation. However, the final

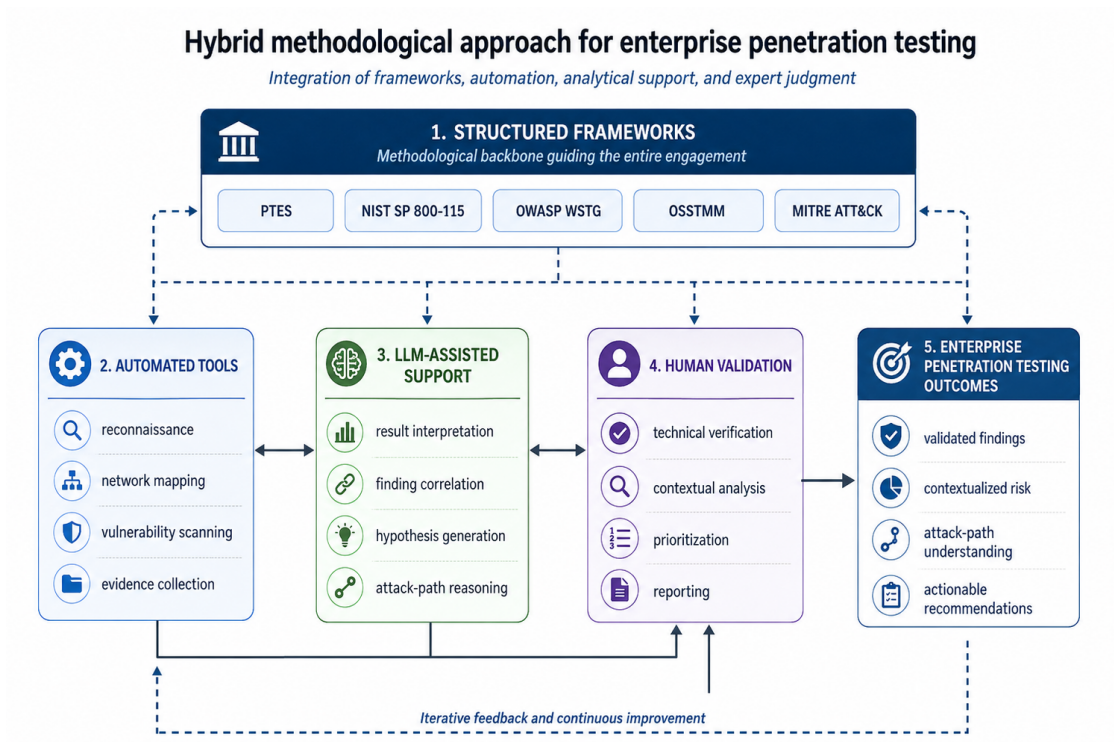


Figure 6.1: Hybrid methodological approach integrating frameworks, automated tools, LLM-assisted support, and human validation in enterprise penetration testing.

value of the workflow depends on the analyst’s ability to validate outputs, connect findings to the target environment, and translate technical evidence into meaningful security conclusions.

Figure 6.1 summarizes the hybrid methodological approach proposed in this thesis by showing how structured frameworks, automated tools, LLM-assisted support, and human validation interact within an enterprise penetration testing workflow.

6.6.1 Integration of Frameworks, Automation, and Analyst Reasoning

The integration of frameworks, automation, and analyst reasoning represents the core of the hybrid methodological approach proposed in this thesis. Frameworks provide structure and ensure that the assessment does not become a collection of disconnected technical activities. They help define the sequence of work, the expected outputs, and the relationship between planning, execution, validation, and reporting. In this sense, frameworks remain necessary even when the practical execution of the assessment is strongly supported by automated tools.

Automation contributes by improving the operational efficiency of the workflow. Tools can support reconnaissance, network mapping, vulnerability scanning, evidence collection, and preliminary validation. These activities are especially useful in enterprise environments, where the scale of the target surface may make purely manual analysis insufficient. However, the outputs produced by automated tools must be interpreted within the methodological structure of the engagement. Without this structure, automation may generate large amounts of data without producing clear security knowledge.

Analyst reasoning connects these components. The analyst determines which tool outputs are relevant, which findings require validation, which attack paths are realistic, and how the results should be prioritized. This reasoning activity is essential because penetration testing is not limited to detecting technical weaknesses. It also requires understanding how weaknesses can be exploited, combined, and translated into actual impact. A hybrid methodology therefore depends on the interaction between structured frameworks, automated evidence collection, and human analytical judgment.

6.6.2 Possible Role of LLM Support in the Workflow

Within a hybrid methodological approach, LLM-assisted support may act as a complementary analytical layer. Its role is not to replace the tester or to perform autonomous penetration testing, but to support specific reasoning-oriented activities. These may include summarizing tool outputs, comparing findings from different sources, organizing preliminary hypotheses, mapping findings to possible adversarial behaviors, and assisting in the explanation of technical results.

This role is particularly useful when the assessment produces heterogeneous information. Enterprise penetration testing may involve network scan results, vulnerability scanner outputs, OSINT data, manual notes, configuration details, and evidence from exploitation attempts. An LLM can help the analyst organize this information into a more coherent form, making it easier to identify relationships between findings and to decide which elements deserve further investigation.

However, this support must remain explicitly supervised. LLM-generated outputs should be treated as hypotheses or analytical suggestions, not as validated conclusions. The analyst must verify whether the proposed interpretation is technically correct, whether the suggested attack path is feasible, and whether the reasoning is supported by evidence collected during the engagement. In this sense, LLM-assisted support is useful only when it is embedded in a workflow that preserves human control, technical verification, and methodological accountability.

Recent developments in cyber-oriented frontier AI systems further reinforce the need for this cautious perspective. For example, evaluations of Claude Mythos Preview have shown increasing capabilities in controlled cybersecurity tasks, including capture-the-flag exercises and multi-step cyber-attack simulations [9]. At the same time, initiatives such as Project Glasswing present these capabilities within a defensive and controlled context, aimed at supporting vulnerability discovery and the protection of critical software ecosystems [10]. These developments suggest that AI-based support may become increasingly relevant in cybersecurity workflows, but they also confirm the need for clear operational boundaries,

human supervision, and responsible validation.

6.6.3 Expected Benefits in Enterprise Environments

The expected benefit of a hybrid methodological approach is the possibility of combining the strengths of different components while reducing their individual limitations. Frameworks contribute structure, but may be too general or incomplete when applied in isolation. Automated tools contribute speed and coverage, but may produce noisy or incomplete outputs. LLM-assisted support may improve interpretation and correlation, but can generate inaccurate or unsupported reasoning. Human validation provides contextual judgment and accountability, but can be constrained by time, scale, and cognitive load.

By integrating these elements, the assessment workflow can become both more efficient and more analytically reliable. Automation can support broad technical visibility, LLM-assisted support can help manage and interpret complex information, and human analysts can focus on validation, prioritization, and final conclusions. This balance is especially important in enterprise environments, where penetration testing must address heterogeneous infrastructures, multiple exposure surfaces, operational constraints, and organizational risk considerations.

A hybrid approach also improves methodological coherence. Instead of treating frameworks, tools, and analyst reasoning as separate layers, it connects them into a single workflow. The framework defines the structure of the assessment, automated tools provide technical evidence, LLM-assisted support helps organize and interpret that evidence, and human validation ensures that conclusions remain accurate, contextualized, and defensible. This makes the approach more suitable for enterprise penetration testing than isolated reliance on a single framework, a single tool category, or unsupervised AI-based assistance.

Ultimately, the hybrid methodological approach proposed in this thesis reflects the practical nature of modern penetration testing. Effective enterprise assessments require structured methodology, operational efficiency, analytical support, and expert validation at the same time. The goal is therefore not to replace the penetration tester with automation, but to enhance the analyst's ability to manage complexity. By reducing repetitive workload, supporting the organization of technical evidence, and facilitating preliminary reasoning on attack paths, automated and LLM-assisted components can help the testing team deliver results that are more structured, better validated, and more relevant to the organization's security posture.

Chapter 7

Enterprise Penetration Testing Application

After discussing penetration testing frameworks, automation, LLM-assisted support, and hybrid methodological integration, the thesis now turns to a practical enterprise penetration testing experience. The purpose of this chapter is not to disclose sensitive details about the client environment or to present the internship project as a formal validation of the methodological perspective developed in the previous chapters. Rather, the chapter uses the practical case as a professional context in which some of the issues discussed earlier can be observed and interpreted. For this reason, all identifying information about the client, the specific brands involved, and the technical infrastructure has been omitted or generalized.

The practical case is based on an enterprise penetration testing project carried out during the internship experience at KPMG. The assessment concerned an Italian organization operating in the fashion sector and managing multiple companies and brands within the same corporate group. The activity lasted approximately three months and included three main dimensions: external perimeter assessment, internal network assessment, and physical security assessment. This multi-dimensional scope made the project relevant for the thesis because it involved several elements discussed in the previous chapters, including methodological structure, automated tools, analyst reasoning, LLM-assisted support, and expert validation.

The chapter first introduces the context, scope, objectives, and boundaries of the engagement. It then describes the methodological organization of the project, explaining how the practical workflow can be related retrospectively to the frameworks and hybrid perspective discussed in the thesis. The chapter also presents the main categories of tools used during the activity, the role of LLM-assisted support, the principal findings identified across the different assessment areas, and the benefits and limitations observed in practice. In this way, the chapter connects the theoretical and methodological discussion of the thesis to a concrete professional experience, while preserving the distinction between practical observation and formal validation.

7.1 Context and Scope of the Project

The practical application discussed in this chapter is based on enterprise penetration testing activities in which the author was directly involved during the internship experience with KPMG. The project concerned an Italian organization operating in the fashion sector and managing multiple brands and companies within the same corporate group. For confidentiality reasons, the name of the client, the specific brands involved, the internal infrastructure details, and any directly identifiable technical information are not disclosed in this thesis.

The engagement lasted approximately three months and involved the assessment of different security dimensions of the organization. More specifically, the activity included the external perimeter, the internal network environment, and the physical security perimeter. This made the project particularly relevant for the purposes of the thesis, since it provided a concrete example of enterprise penetration testing in a complex setting characterized by multiple exposure surfaces, operational constraints, and the need to combine methodological structure, automated tools, analyst reasoning, and human validation.

The assessment was conducted according to a black-box approach. This means that, for each testing dimension, the assessment team operated with limited prior knowledge of the target environment, in a way that approximated the perspective of an external or initially uninformed attacker. At the same time, the activity was fully authorized and performed within the boundaries defined by the engagement rules. These constraints were essential to ensure that testing activities remained controlled, proportionate, and compatible with the client's operational continuity requirements.

7.1.1 Consulting Context of the Internship

The penetration testing activity was performed within a professional consulting context. The work was carried out as part of a security assessment project delivered to an external client organization. The consulting nature of the engagement influenced both the methodological organization of the work and the way in which results had to be validated, documented, and communicated.

Unlike laboratory exercises or isolated technical tests, a consulting engagement requires continuous attention to scope, evidence, operational impact, and client communication. The goal is not only to identify vulnerabilities, but also to produce findings that are reliable, reproducible, and useful for remediation. For this reason, the practical activity was not limited to the execution of technical tools. It also required the organization of collected evidence, the validation of tool-generated outputs, and the interpretation of findings in relation to the client's environment.

The activity was carried out within a team-based professional setting. This aspect is relevant because enterprise penetration testing is rarely the result of isolated individual work. It normally involves coordination among testers, managers, client representatives, and defensive teams. In the specific project discussed here, the assessment activities were performed under the supervision and coordination of the consulting team, ensuring that the work remained aligned with the objectives and boundaries agreed with the client.

7.1.2 Enterprise Environment Under Assessment

The target environment belonged to a large corporate group in the fashion sector, composed of several companies and brands. This organizational structure made the assessment more complex than a test performed against a single isolated entity. The external attack surface included domains, subdomains, exposed services, web applications, and public information associated with the different companies of the group. The internal environment also included multiple reachable systems, network segments, and services that required careful mapping and validation.

This type of environment is representative of many enterprise contexts. Large organizations often evolve through acquisitions, brand diversification, technological stratification, and the coexistence of legacy and modern systems. As a result, their security posture cannot be evaluated only by looking at a single application or a single network range. The assessment must instead consider the interaction between different assets, exposure points, operational processes, and security controls.

The fashion sector context also introduced additional relevance from a business perspective. In such environments, digital systems may support e-commerce platforms, catalogue management, internal logistics, production-related processes, brand communication, and corporate services. Consequently, a technical weakness may have implications that go beyond the affected system alone. It may influence confidentiality, operational continuity, reputation, or the protection of commercially sensitive information.

7.1.3 External, Internal, and Physical Security Perimeter

The project covered three main assessment areas: the external perimeter, the internal network, and the physical security perimeter. This multi-dimensional scope was particularly useful for evaluating the practical relevance of the hybrid methodological perspective discussed in the previous chapters.

The external perimeter assessment focused on publicly reachable assets associated with the corporate group. The objective was to identify domains, subdomains, exposed hosts, services, web interfaces, and other elements that could define the organization's external attack surface. This phase involved open-source intelligence activities, domain and subdomain enumeration, DNS analysis, correlation of public information, service discovery, and vulnerability identification.

The internal network assessment was conducted after obtaining authorized access to the client's environment. The purpose was to evaluate the exposure of internal systems, reachable services, network segmentation, and possible attack paths within the organization. Unlike the external phase, the internal assessment provided visibility from a position closer to that of an attacker who had already obtained some level of internal access. This perspective was useful for understanding whether internal weaknesses could support lateral movement, privilege escalation, or access to more sensitive network areas.

The physical security assessment focused on the possibility of accessing restricted areas through weaknesses in physical access control, employee routines, and perimeter supervision. The objective was not to cause disruption, but to evaluate whether the organization's

physical controls were sufficient to prevent unauthorized access to sensitive areas. This part of the engagement was also conducted under predefined authorization conditions, with at least one internal reference aware of the activity for safety and escalation purposes.

Taken together, these three assessment areas provided a broad view of the client's exposure. The external perimeter helped identify publicly accessible weaknesses, the internal assessment highlighted risks associated with the corporate network, and the physical security activity showed how non-digital controls could also influence the overall security posture. This multi-dimensional scope is useful for the discussion developed in the thesis because it shows that enterprise penetration testing is rarely limited to a single technical layer. In practice, digital exposure, internal reachability, operational constraints, and physical access conditions may all contribute to the assessment. The case therefore provides a concrete setting in which the need for multiple perspectives can be discussed.

7.1.4 Objectives and Boundaries of the Activity

The main objective of the activity was to assess the security posture of the client's enterprise environment across external, internal, and physical dimensions. The assessment aimed to identify exploitable weaknesses, validate their practical relevance, and provide evidence-based findings that could support remediation and risk reduction. In line with the methodological perspective developed in the thesis, the purpose was not simply to collect tool outputs, but to interpret and validate findings within the specific context of the engagement.

Before the beginning of the active testing phases, the client was informed about the expected testing windows and the source IP addresses from which the assessment traffic would originate. This communication was necessary to avoid unnecessary escalation by monitoring systems, such as the Security Operations Center or SIEM-based detection mechanisms, and to ensure that the activity remained distinguishable from unauthorized malicious activity.

The rules of engagement explicitly excluded activities that could compromise business continuity. In particular, denial-of-service techniques and brute-force attacks were not permitted. These restrictions were important because the client environment included production systems and business-relevant services. The assessment therefore had to balance realism with operational safety, ensuring that validation activities remained controlled and proportionate.

These boundaries influenced the way the penetration test was conducted. The assessment team adopted a progressive approach, starting from low-impact information gathering and gradually increasing the depth of testing where appropriate and authorized. Tool configuration, scanning intensity, and validation attempts were selected carefully in order to reduce the risk of disruption. This operational discipline was essential in preserving the professional nature of the engagement and in ensuring that the results could be used reliably by the client.

Overall, the context and scope of the project make it an appropriate practical case for this thesis. The engagement involved a complex enterprise environment, multiple testing dimensions, clear operational constraints, extensive use of automated tools, LLM-assisted

analytical support, and continuous human validation. For this reason, it provides a concrete basis for discussing how the methodological perspective developed in the previous chapters can be used to interpret a real penetration testing activity.

7.2 Methodological Organization of the Activity

The practical activity was organized according to the operational needs of the consulting engagement, rather than through the mechanical application of a single framework or of a new methodology proposed by the thesis. The assessment had to respect the agreed scope, the rules of engagement, the available time, the client's operational constraints, and the confidentiality requirements of the project. Within these boundaries, the workflow combined structured planning, tool-assisted analysis, analyst reasoning, and human validation.

For the purposes of this thesis, the project can be interpreted through the methodological perspective developed in the previous chapters. The activity showed several elements that are consistent with a hybrid view of enterprise penetration testing: frameworks helped describe the general logic of the process, automated tools supported reconnaissance and validation, LLM-assisted support contributed to interpretation and organization of information, and human analysts remained responsible for final technical judgment. This interpretation should be understood retrospectively, since the professional engagement itself was governed by the procedures and constraints defined for the client project.

7.2.1 Phases Followed During the Project

The project followed a sequence of activities broadly aligned with the phases normally found in professional penetration testing engagements. Before active testing began, the assessment boundaries were agreed with the client. This included the definition of the scope, the testing periods, the source IP addresses used by the assessment team, and the activities explicitly excluded from the engagement, as discussed in Section 7.1.4.

This initial phase was necessary because, in an enterprise environment, penetration testing cannot begin directly with technical exploitation. The team first had to clarify what was authorized, which systems and perimeters were included, which activities were prohibited, and how testing traffic should be distinguished from unauthorized activity. In this project, the client was informed in advance about the testing windows and source IP addresses, so that monitoring structures such as the SOC or SIEM-based systems could correctly contextualize the activity.

After this preliminary definition, the work moved to reconnaissance and information gathering. The external perimeter assessment started with the collection of public information related to the organization, including domain intelligence, DNS records, subdomains, public exposure, historical endpoints, and infrastructure-related information. This phase helped build an initial view of the external attack surface and identify assets that required further analysis. The activity was supported by both manual investigation and automated tools.

The next phase concerned vulnerability analysis and technical validation. Once relevant

assets, services, and exposure points had been identified, the assessment team used scanning and validation tools to detect possible weaknesses and compare results obtained from different sources. At this stage, the main task was not simply to collect tool-generated findings, but to understand which results were actually relevant, which required additional verification, and which could support realistic attack scenarios.

Where authorized and appropriate, selected findings were then tested through controlled validation attempts. The objective was to determine whether specific weaknesses could have practical security relevance in the assessed environment. This phase remained strictly constrained by the rules of engagement. The team avoided techniques that could affect service availability or create excessive operational risk, focusing instead on proportionate validation activities compatible with the agreed scope.

The internal network assessment followed a similar logic, but from a different position. After authorized access to the internal environment was provided, the activity focused on mapping reachable systems, identifying exposed services, analyzing internal weaknesses, and evaluating possible attack paths across more sensitive network segments. This phase was useful for understanding how an attacker with some level of internal access could potentially move beyond the initial point of connection.

The physical security assessment introduced an additional dimension. The activity included observation of the external perimeter, analysis of employee routines, assessment of access control procedures, and controlled attempts to evaluate whether unauthorized access to sensitive areas could occur. This phase showed that the security posture of an enterprise cannot be assessed only through digital systems, since physical access conditions may also influence the overall exposure of the organization.

7.2.2 Retrospective Interpretation of the Workflow

The engagement can be read as a practical example of the methodological reasoning developed in the thesis, although it was not designed as a formal application of the proposed perspective. The activity was not based solely on a framework, a tool, or an individual technique. In practice, it required the combination of planning, automated data collection, manual analysis, LLM-assisted reasoning, and expert validation.

Automated tools were especially important during reconnaissance, mapping, scanning, and preliminary validation. In the external perimeter assessment, they supported domain intelligence, subdomain enumeration, DNS analysis, public exposure mapping, service discovery, and vulnerability identification. In the internal assessment, they supported host discovery, service enumeration, vulnerability analysis, and the exploration of possible internal attack paths. Without automation, obtaining an initial view of a broad and distributed enterprise environment would have been significantly slower and less complete.

However, automation alone was not sufficient. The outputs generated by different tools had to be compared, filtered, and interpreted. Some results overlapped, some required technical confirmation, and others were not relevant in the specific context of the engagement. This required human analysis to distinguish between potential exposure and validated risk. In this sense, the practical workflow reflected one of the recurring points discussed in the previous chapters: automated tools improve efficiency and scalability, but

they do not replace contextual interpretation.

LLM-assisted support was used as an analytical aid in different phases of the activity. It helped in reading and summarizing command outputs, comparing results produced by different tools, reasoning about possible vulnerability relevance, and organizing preliminary attack-path hypotheses. It was also useful for supporting the preparation of scripts, command variations, and technical checks, always under human review. Its contribution was therefore mainly analytical and organizational, rather than autonomous or decision-making.

The role of the analyst remained central. Every relevant output produced by automated tools or supported by LLM-based reasoning had to be checked against the actual conditions of the target environment. This was particularly important when dealing with findings such as exposed secrets, weak administrative credentials, insecure legacy services, internal protocol weaknesses, or possible attack paths involving multiple systems. In each case, the assessment team had to determine whether the issue was real, exploitable, relevant to the client, and supported by evidence.

The workflow can therefore be described as hybrid in practice, but only in an interpretative sense. Structured planning provided order, automation provided scale, LLM-assisted support helped organize and reason about information, and human validation ensured reliability. This balance was important in the context of an enterprise assessment, where the complexity of the environment, the operational constraints, and the need for responsible validation made purely manual or purely automated approaches insufficient.

7.2.3 Relation with the Frameworks Discussed in the Thesis

The workflow observed during the project can be related to the frameworks discussed in the previous chapters, although none of them was applied as a complete and exclusive methodology. PTES is the closest reference from an operational perspective, because its lifecycle-oriented structure resembles many phases of the engagement. The project included pre-engagement definition, intelligence gathering, vulnerability analysis, controlled validation, post-exploitation reasoning, and reporting. These phases were not followed mechanically, but they provide a useful way to describe the organization of the work [2].

The pre-engagement phase of PTES can be recognized in the definition of testing periods, source IP addresses, excluded techniques, and authorization boundaries. Intelligence gathering was reflected in the external reconnaissance and OSINT activities. Vulnerability analysis was carried out through scanning, manual review, and comparison of tool outputs. Exploitation and post-exploitation were approached mainly as controlled validation and impact assessment, always within the limits of the rules of engagement. Reporting then transformed the technical evidence into findings useful for remediation and decision-making.

MITRE ATT&CK was relevant in a different way. It was not used as an end-to-end penetration testing methodology, but as a conceptual reference for adversarial reasoning and attack-path interpretation [5]. This was useful when evaluating whether individual weaknesses could be connected to broader compromise scenarios. For example, findings related to exposed services, weak credentials, insecure internal protocols, or insufficient segmentation could be considered not only as isolated issues, but also as possible elements

of a larger attack path.

NIST SP 800-115 was useful from a procedural and governance-oriented perspective. Its emphasis on planning, controlled execution, evidence collection, and reporting is consistent with the requirements of a professional engagement [1]. This was particularly relevant because the assessment was conducted in an active enterprise environment, where technical testing had to remain compatible with operational continuity and organizational expectations.

OWASP WSTG was relevant mainly for the analysis of web-facing assets and application-layer exposure [3]. During the external perimeter assessment, web applications and management interfaces represented important components of the attack surface. In this context, OWASP provided useful conceptual support for reasoning about application security weaknesses, exposed functionality, insecure configurations, and the validation of web-related findings.

OSSTMM was less directly connected to the operational flow of the engagement, but its emphasis on structure, repeatability, and systematic testing remained conceptually relevant [4]. In particular, the need to document evidence, compare results, and maintain consistency across the assessment reflected some of the methodological concerns associated with OSSTMM. However, its more formal and measurement-oriented approach was less directly applicable to the practical flow of the engagement than PTES.

Overall, the project is consistent with the comparative conclusion developed in Chapter 5: different frameworks provide different forms of value, and no single reference is sufficient on its own. PTES helps describe the operational lifecycle, MITRE ATT&CK supports adversarial reasoning, NIST SP 800-115 reinforces procedural discipline, OWASP WSTG helps with application-layer analysis, and OSSTMM contributes a broader concern for structured and repeatable testing. The practical engagement therefore supports, rather than formally validates, the usefulness of a hybrid methodological perspective adapted to the specific characteristics of the enterprise environment.

7.3 Tools Used in the Activity

The practical activity relied on different categories of tools, each used for a specific purpose within the assessment. The objective was not to depend on a single automated solution, but to collect technical information from multiple sources and then interpret it in relation to the agreed scope. This was particularly important because the project covered external, internal, and physical assessment dimensions, each requiring different forms of information gathering, scanning, validation, and analysis.

The tools used during the engagement supported activities such as reconnaissance, network mapping, vulnerability identification, automated validation, exploitation support, and evidence organization. Their outputs were treated as working material rather than as final conclusions. In practice, each relevant result had to be compared with other evidence, reviewed by the assessment team, and interpreted in relation to the client environment.

```
nmap -sT localhost.example.com

Starting Nmap 6.40 ( http://nmap.org ) at 2019-07-21 17:08 EDT
Nmap scan report for localhost.example.com (127.0.0.1)
Host is up (0.0010s latency).
Not shown: 991 closed ports
PORT      STATE      SERVICE
22/tcp    filtered  ssh
25/tcp    open       smtp
80/tcp    open       http
111/tcp   open       rpcbind
631/tcp   open       ipp
3000/tcp  open       ppp
4000/tcp  open       remoteanything
8080/tcp  open       http-proxy
9090/tcp  open       zeus-admin

Nmap done: 1 IP address (1 host up) scanned in 1.25 seconds
```

Figure 7.1: Reconstructed example of service enumeration output based on Nmap-style results. The figure is included for explanatory purposes only and does not reproduce customer data.

7.3.1 Nmap

Nmap was used to support network discovery, port scanning, and service enumeration. Its role became particularly relevant after the identification of potentially interesting hosts and services, because it allowed the team to obtain a more precise view of exposed network surfaces. In both the external and internal phases, Nmap helped identify reachable systems, open ports, running services, and, where possible, service versions.

Figure 7.1 provides an illustrative example of service enumeration output based on Nmap-style results.

During the external perimeter assessment, Nmap was used after the initial OSINT and asset discovery activities. Once domains, subdomains, and potentially reachable hosts had been identified, the tool helped determine which services were actually exposed and which of them required deeper analysis. This made it possible to move from a broad list of candidate assets to a more concrete view of the external attack surface.

During the internal assessment, Nmap was useful for mapping reachable systems and services from an authorized internal access point. This phase helped the team understand which systems were visible from that position, which services were exposed internally, and which areas required further validation.

The use of Nmap was adapted to the operational constraints of the engagement. Since the client environment included production systems, scan intensity and timing had to be selected carefully. The objective was to collect useful technical information without creating unnecessary disruption. For this reason, Nmap was used as part of a controlled workflow, where scan configuration and follow-up analysis remained guided by the assessment team.

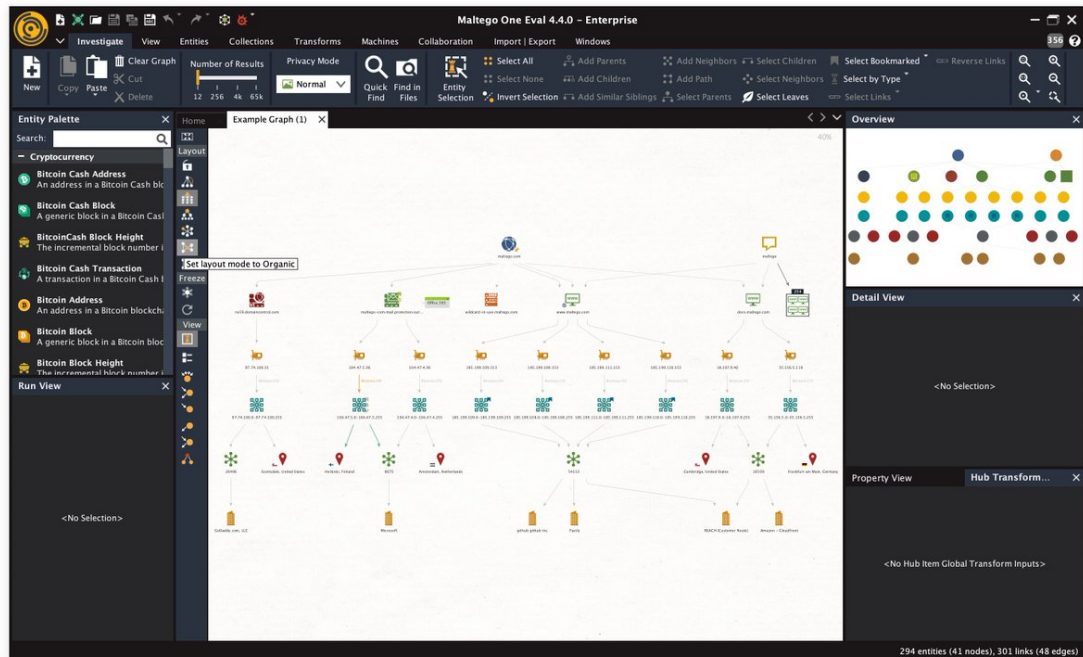


Figure 7.2: Illustrative example of OSINT-based relationship mapping used to support external exposure analysis. The figure does not reproduce original customer data.

7.3.2 Maltego

Maltego was used to support information gathering and relationship mapping across relevant entities and publicly available data. In the external perimeter phase, it helped organize information related to domains, subdomains, infrastructure elements, public records, and possible relationships between different entities connected to the corporate group.

Its value was not limited to collecting individual data points. Maltego was useful because it allowed the team to visualize relationships between entities and reason more clearly about the structure of the external exposure. In a corporate group composed of multiple companies and brands, this was relevant because assets were not always isolated or immediately attributable to a single organizational unit.

Figure 7.2 shows an illustrative example of OSINT-based relationship mapping used to support external exposure analysis.

Maltego also helped correlate information collected from different public sources. Domain relationships, technical indicators, infrastructure references, and public records could be organized in a more readable form. This supported the identification of possible areas of interest and helped guide subsequent activities, such as subdomain enumeration, service discovery, and vulnerability scanning.

At the same time, the information collected through Maltego required careful validation.

Public sources may contain outdated, duplicated, or irrelevant data. For this reason, the results produced during the OSINT phase were treated as preliminary indications. The analyst still had to verify whether each identified entity was relevant to the agreed scope and whether it could support further assessment activity.

7.3.3 Nessus

Nessus was used to identify known vulnerabilities, insecure configurations, outdated services, and other exposure conditions affecting the assessed assets. Its role was particularly relevant during the vulnerability analysis phase, where automated scanning helped produce an initial list of possible weaknesses requiring further review.

The main advantage of Nessus was its ability to provide broad coverage across a heterogeneous target surface. In an enterprise infrastructure containing both legacy and modern assets, purely manual vulnerability identification would have been inefficient and potentially incomplete. The scanner therefore provided a preliminary baseline of possible exposure conditions, helping the assessment team identify areas that deserved closer manual validation.

Figure 7.3 provides an illustrative example of a vulnerability scanning dashboard used to support exposure identification.

In the external assessment, Nessus helped identify vulnerabilities and misconfigurations affecting exposed systems and services. In the internal assessment, it supported the analysis of reachable hosts and services from within the client environment, contributing to the discovery of weaknesses that were not visible from the public Internet.

Nessus outputs, however, required careful interpretation. Some findings could be false positives, while others could be technically correct but not exploitable in the specific environment. In addition, generic severity ratings did not always reflect the actual business or operational relevance of the affected system. For this reason, the results produced by Nessus were compared with other sources of evidence and manually reviewed before being considered reliable findings.

7.3.4 Pentera

Pentera was used as an automated validation and attack simulation platform to support both the external and internal phases of the assessment. Its role was different from that of a traditional vulnerability scanner, because it helped evaluate whether some exposure conditions could be connected to more practical attack scenarios, within the limits defined by the engagement.

During the external perimeter assessment, Pentera complemented the information obtained through OSINT, network scanning, and vulnerability scanning activities. In this context, the platform was useful for examining whether externally reachable assets and exposed services could be associated with relevant attack paths. This was important because the external perimeter included several domains, subdomains, web interfaces, and public-facing services connected to the corporate group. Pentera therefore supported the

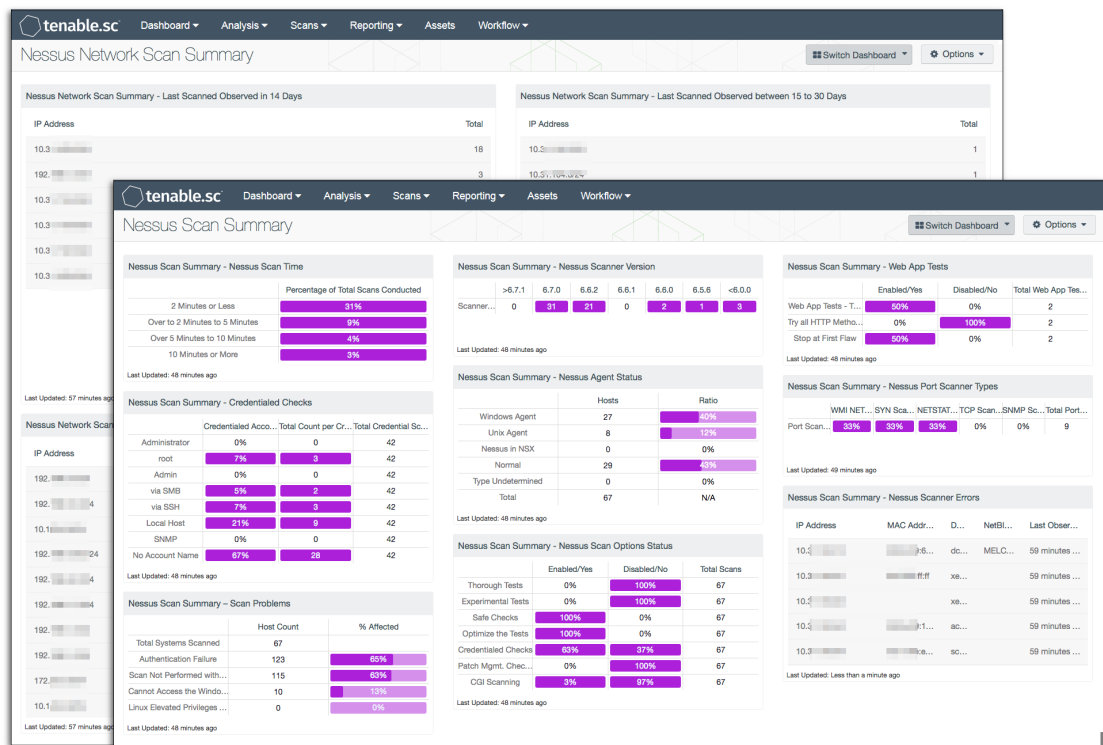


Figure 7.3: Illustrative example of a vulnerability scanning dashboard used to support exposure identification. Sensitive and customer-specific information is omitted.

transition from asset discovery and vulnerability identification toward a more validation-oriented analysis of the external exposure.

Figure 7.4 shows an illustrative example of automated exposure validation and attack-path analysis.

During the internal network assessment, Pentera supported the analysis of reachable systems, exposed services, and possible internal attack paths. This was relevant because the internal environment included more sensitive network segments, and the assessment required an understanding of how weaknesses could potentially interact across systems. In this phase, the platform helped examine whether internal exposures could support lateral movement, access to sensitive resources, or chained compromise scenarios.

Pentera also helped compare potential vulnerabilities with more practically relevant exposure conditions. In some cases, automated scanners may report weaknesses without showing whether they can be meaningfully used in the assessed environment. Automated validation platforms can reduce part of this gap by testing certain attack steps under controlled conditions. This made the tool useful for prioritizing findings and for identifying issues that deserved closer manual analysis.

Nevertheless, Pentera outputs were not treated as autonomous conclusions. The results still required review by the assessment team, especially when they involved possible attack

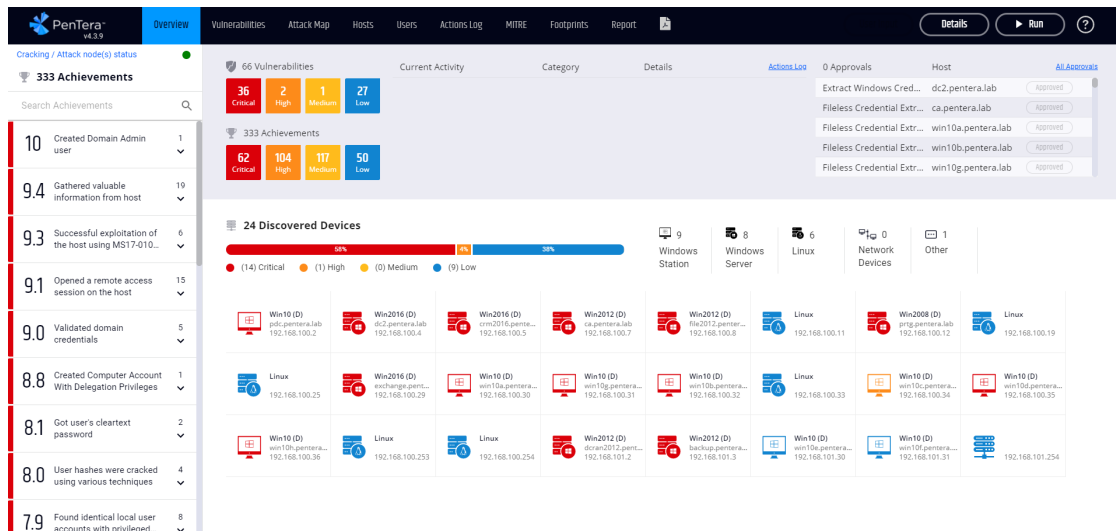


Figure 7.4: Illustrative example of automated exposure validation and attack-path analysis. The figure does not reproduce original customer data.

paths, exploitation chains, external exposure, or internal movement scenarios. The platform provided useful technical indications, but the final interpretation depended on human validation, the rules of engagement, and the specific context of the client environment.

7.3.5 Other Supporting Tools

In addition to the main tools described above, several supporting tools were used during reconnaissance, OSINT, and validation activities. These tools contributed to domain intelligence, subdomain enumeration, DNS analysis, certificate transparency review, autonomous system and netblock discovery, public exposure correlation, historical endpoint collection, and targeted verification of specific findings. Their purpose was to expand the initial visibility of the attack surface and to support more focused testing activities.

During the external reconnaissance phase, supporting tools helped identify possible assets associated with the client's domains. Subdomain enumeration tools collected candidate hostnames, while certificate transparency sources supported the discovery of names and endpoints that could reveal additional parts of the external perimeter. DNS-related utilities were used to analyze records, resolution behavior, mail infrastructure, and other technical information relevant to the initial mapping of the environment. ASN and netblock discovery tools helped identify infrastructure potentially associated with the organization, while public exposure sources supported the correlation of visible services.

Historical endpoint collection also played a useful role. Previously exposed URLs, archived paths, and historical references may provide indications about legacy applications, deprecated endpoints, or forgotten assets. Not every historical result is relevant or still active, but this type of information can help the analyst build a broader understanding of the target's public exposure. In this project, such information was treated as an input for

further verification rather than as evidence of vulnerability by itself.

Other supporting tools were relevant during the validation phase. Once potential weaknesses had been identified, the team needed to determine whether they had practical security relevance within the boundaries of the engagement. In this context, exploitation-oriented frameworks such as Metasploit and manual validation utilities could support controlled testing activities, especially when a finding required further technical confirmation beyond the output of automated scanners.

The role of exploitation-oriented tools was not to maximize compromise, but to support evidence-based validation. They can be useful in professional penetration testing when used to verify known vulnerabilities in a controlled and authorized manner. Their use must remain aligned with the rules of engagement, the technical scope, and the operational constraints of the client environment. In this project, this meant avoiding disruptive techniques and focusing only on validation activities compatible with the agreed boundaries of the assessment.

Manual utilities were also useful for protocol analysis, service interaction, web request inspection, and targeted verification of specific findings. These tools supported the transition from automated detection to validated evidence. They were especially important when scanner outputs required confirmation, when a service had to be inspected more closely, or when the analyst needed to understand whether a reported weakness had practical exploitability or business relevance.

As with the other tools used during the engagement, supporting and exploitation-oriented utilities were not treated as autonomous decision-making instruments. Their outputs had to be interpreted by the assessment team and considered in relation to the agreed scope, the technical evidence collected, and the potential impact on the client environment. Consistently with the engagement boundaries defined in Section 7.1.4, these utilities were used only to support controlled evidence collection, targeted verification, and contextual analysis.

7.4 Use of LLM-Assisted Support

LLM-assisted support was used during the project as an analytical aid, not as an autonomous penetration testing capability. Its role was limited to supporting the interpretation of technical information, the organization of tool outputs, the formulation of preliminary hypotheses, and the preparation of some validation activities. In line with the methodological discussion developed in Chapter 6, all outputs produced with LLM support remained subject to human review and technical verification.

This distinction is important because the engagement generated a large amount of heterogeneous information. External reconnaissance outputs, DNS records, subdomain enumeration results, network scan results, vulnerability scanner findings, automated validation outputs, manual observations, and physical security notes had to be interpreted together. In this context, LLM-assisted support was useful for reducing part of the effort required to summarize, compare, and organize information. However, it was never treated as a source of evidence by itself.

7.4.1 Support in Reviewing Tool-Generated Findings

One of the main uses of LLM-assisted support concerned the review of tool-generated findings. Automated tools such as scanners, reconnaissance utilities, and attack simulation platforms produced outputs that were often extensive, technical, and partially overlapping. LLM-assisted support helped reorganize these outputs, summarize relevant elements, and highlight which findings required closer attention.

This was useful when comparing information produced by different tools. For example, a vulnerability scanner could identify a possible weakness, while a network-mapping utility could provide additional context about service exposure, software version, or asset reachability. LLM-assisted support helped place these pieces of information into a more readable structure, making it easier to identify duplicated results, unclear findings, or issues that required further verification.

However, this support remained preliminary. A model could help explain why a finding might be relevant or suggest which aspects deserved further analysis, but it could not independently confirm exploitability. For this reason, LLM-assisted outputs were used as working notes or analytical support, not as final validation. The actual confirmation of findings remained based on technical evidence, controlled testing, and manual review.

This distinction was especially important when dealing with scanner outputs that could include false positives or findings with limited practical relevance. In such cases, LLM-assisted support helped frame the questions to be investigated, but the final decision depended on whether the finding could be reproduced, whether it affected an in-scope asset, and whether it had meaningful security implications in the client environment.

7.4.2 Support in Reasoning About Possible Attack Paths

LLM-assisted support was also useful when reasoning about possible attack paths emerging from the collected evidence. In enterprise penetration testing, individual findings do not always provide a clear view of the overall risk. A weak administrative interface, an exposed service, an insecure protocol, or a hardcoded secret may become more significant when considered together with other weaknesses.

In this project, LLM-assisted support helped organize preliminary hypotheses about how different findings could interact. For example, it could support reasoning about whether exposed credentials, weak authentication, vulnerable services, or insufficient segmentation might contribute to a broader compromise scenario. This was useful when moving from isolated technical observations to a more structured analysis of possible exploitation chains.

The support was also relevant when using MITRE ATT&CK as a conceptual reference. Since ATT&CK provides a structured vocabulary for describing adversarial behavior, it helped frame some observations in relation to possible attacker tactics and techniques [5]. LLM-assisted reasoning could support this organization, but it did not determine the attack path autonomously.

Every proposed attack path still required human validation. A sequence of steps may appear plausible in theory, but this does not mean that it is technically feasible in the target environment. The assessment team had to verify whether the required conditions

were present, whether the scenario respected the rules of engagement, and whether the reasoning was supported by evidence. In this sense, LLM-assisted support contributed mainly to hypothesis generation, while the analyst remained responsible for validation.

7.4.3 Support in Correlating Technical Evidence

Another use of LLM-assisted support concerned the correlation of technical evidence from different sources. The assessment involved OSINT results, domain and subdomain information, DNS records, network scan outputs, vulnerability scanner findings, Pentera results, manual observations, and physical security notes. These sources were not always directly comparable, and their relationships were not always immediately clear.

LLM-assisted support helped organize this information into more readable groups. It could support the grouping of findings by asset, technical category, exposure type, or possible relevance to attack paths. This was useful when several tools identified related conditions using different terminology or different levels of detail.

This type of support was particularly useful in the external perimeter assessment. Information discovered through OSINT, certificate transparency sources, DNS analysis, and service enumeration had to be connected in order to understand which assets were actually relevant to the client's exposure. LLM-assisted support helped structure these relationships and identify elements that required further verification.

A similar use was observed during the internal assessment. In that phase, evidence related to reachable hosts, exposed services, internal protocols, authentication-related weaknesses, and segmentation issues had to be considered together. LLM-assisted support helped organize possible relationships among these observations, but the final interpretation remained dependent on manual analysis and technical confirmation.

7.4.4 Observed Strengths and Limitations in Practice

The practical use of LLM-assisted support showed both useful contributions and clear limitations. Its main strength was the ability to help the analyst manage complex and heterogeneous information. By supporting summarization, comparison, and preliminary reasoning, it reduced part of the effort required to move from raw outputs to a more organized view of the assessment.

A second strength was its usefulness in supporting reasoning. LLM-assisted support could help formulate better questions about a finding, identify possible relationships between technical observations, and organize validation priorities. This made it useful as an analytical companion, especially in phases where the team had to move from tool outputs to more meaningful security interpretation.

At the same time, several limitations were evident. LLM-assisted outputs could be generic, incomplete, or based on assumptions not fully supported by the available evidence. In some cases, the model could suggest technically plausible explanations that still required careful verification. This is consistent with the limitation already discussed in Chapter 6: LLMs do not have direct awareness of the target environment and cannot independently verify its real technical conditions [7, 8].

The usefulness of LLM-assisted support therefore depended strongly on analyst guidance. The quality of the output improved when the model received precise context, structured tool outputs, clear constraints, and specific questions. Vague prompts, instead, tended to produce generic responses with limited operational value. This means that LLM-assisted support did not reduce the need for expertise; it became useful only when guided by someone able to ask appropriate questions and recognize unsupported conclusions.

Overall, the project suggested that LLM-assisted support can have practical value in penetration testing workflows, but only as part of a controlled and supervised process. Its contribution was strongest in interpretation, correlation, summarization, and preliminary attack-path reasoning. Its limitations concerned reliability, contextual awareness, and the risk of unsupported inferences. For this reason, the experience supports a cautious interpretation: LLMs may help analysts manage complexity, but they cannot replace expert validation, technical evidence, or professional responsibility.

7.5 Main Findings

The penetration testing activity led to the identification of several relevant exposure conditions across the external perimeter, the internal network, and the physical security dimension. The objective of this section is not to reproduce the full technical report or disclose sensitive details about the client environment, but to summarize the main categories of findings that emerged during the engagement. For confidentiality reasons, specific hostnames, IP addresses, application names, screenshots containing identifiable information, and internal architectural details are omitted or generalized.

The findings were not considered only as isolated technical observations. Each relevant issue was reviewed in relation to exploitability, affected asset, possible impact, and position within a broader attack scenario. This was important because some weaknesses had limited relevance when considered individually, but became more significant when connected to other exposure conditions. In this sense, automated tools supported detection and preliminary validation, LLM-assisted support helped organize and interpret evidence, and human analysts remained responsible for confirming the practical relevance of the findings.

7.5.1 Relevant Vulnerabilities Identified

One relevant category of findings concerned exposed secrets and sensitive information within externally accessible resources. During the external assessment, the analysis of web-facing assets and client-side resources led to the identification of hardcoded API keys exposed in frontend code. This type of weakness is significant because secrets embedded in client-side components may be accessible to unauthenticated users and can potentially be reused to interact with backend services, third-party APIs, or internal application functions, depending on their permissions and configuration.

Another relevant finding concerned weak administrative access controls on a web management interface. In one case, an administrative panel was reachable from the external perimeter and was protected by weak default-like credentials. The presence of

credentials such as `admin:admin` represented a serious weakness, since it allowed access to administrative functionality without requiring advanced exploitation techniques. This finding also showed the importance of verifying insecure operational practices and poor credential hygiene, not only software vulnerabilities.

The assessment also identified insecure legacy service exposure. In particular, an FTP service was found to rely on weak authentication and on an insecure file transfer protocol rather than a secure alternative such as FTPS or SFTP. This issue was relevant because legacy services may expose sensitive data, credentials, or operational workflows if they are not properly protected. In an enterprise environment, such services may also become useful entry points or supporting elements in broader attack paths.

Another important issue concerned unauthorized access to catalogue-management functionality. The assessment showed that certain management functions related to product and catalogue handling were insufficiently protected, creating an exposure vector closely connected to the business model of the fashion sector. In this context, pre-release collections, product information, commercial planning data, and logistics-related information may have significant market sensitivity. Consequently, an application-layer weakness of this kind can introduce confidentiality, integrity, and reputational risks, even when it does not immediately lead to full infrastructure compromise.

During the internal assessment, additional weaknesses emerged in relation to network segmentation, internal service exposure, and authentication-related attack paths. From an authorized internal position, the team identified reachable services and internal systems that suggested insufficient separation between different areas of the network. This increased the relevance of some findings because internal exposure may support lateral movement or access to more sensitive resources if combined with weak authentication, insecure protocols, or misconfigured services.

7.5.2 Issues Emerging from External, Internal, and Physical Assessment

The external perimeter assessment highlighted the relevance of public exposure management. The initial reconnaissance phase revealed a broad and distributed attack surface, consistent with the structure of a corporate group managing multiple brands and companies. Domains, subdomains, public-facing services, historical endpoints, and exposed web interfaces required careful filtering and validation in order to distinguish relevant assets from outdated or unrelated information.

From the external perspective, the most significant issues were associated with exposed services, weak access control, insecure web-facing functionality, and sensitive information leakage. These findings showed that the external perimeter cannot be assessed only through vulnerability scanning. OSINT, service enumeration, application review, and manual validation were all necessary to determine whether the discovered assets had practical security relevance.

The internal network assessment provided a different perspective. Once the team obtained authorized access to the internal environment, the focus shifted from public exposure

to internal reachability, segmentation, service configuration, and possible attack paths. In this phase, weaknesses related to internal protocols, name resolution, authentication exposure, and insufficient segmentation became particularly relevant. The assessment showed that internal security controls should not rely on the assumption that internal access is always trusted or harmless.

The internal phase also made it possible to examine how different weaknesses could interact. For example, an exposed internal service may not represent a critical issue in isolation, but it may become more significant if combined with weak credentials, insufficient segmentation, or inadequate monitoring. This made attack-path reasoning useful, since it allowed the team to consider the relationship between findings rather than treating each issue separately.

The physical security assessment introduced a further dimension of exposure. The activity showed that physical access control weaknesses may allow unauthorized individuals to enter restricted areas by exploiting employee routines, insufficient supervision, or weaknesses in access procedures. In the specific engagement, controlled physical assessment activities demonstrated the possibility of reaching sensitive areas without immediate detection. This included access to areas related to production activities and proximity to sensitive technical spaces.

The physical findings were relevant because they showed that enterprise exposure is not limited to digital systems. In some circumstances, physical access can affect information confidentiality, infrastructure protection, and operational security. A weakness in physical controls may also support further technical compromise if an attacker can access network ports, unattended workstations, server rooms, or sensitive documents. For this reason, physical security should be considered as part of the broader enterprise security posture.

7.5.3 False Positives and Validation Issues

The project also highlighted the importance of distinguishing between preliminary findings and validated vulnerabilities. Automated tools generated several indications that required further analysis before being considered reliable. Some findings were duplicated across tools, some were technically correct but not practically exploitable, and others required additional context in order to determine their actual relevance.

False positives were particularly relevant in vulnerability scanning activities. A scanner may report a vulnerability based on a detected version, banner, or configuration pattern, but the finding may not be exploitable if the service is patched, protected by compensating controls, not reachable under realistic conditions, or outside the agreed scope. For this reason, scanner results had to be reviewed manually and compared with additional evidence before being included among the final findings.

Validation issues also emerged when dealing with externally discovered assets. OSINT and historical endpoint collection may identify resources that are no longer active, no longer owned by the organization, or not relevant to the current assessment scope. This required careful filtering in order to avoid attributing irrelevant exposure to the client. The same issue applied to subdomains, certificate transparency results, and public infrastructure references, which often required additional checks before being considered meaningful.

LLM-assisted support helped organize and interpret some of these outputs, but it did not remove the need for human validation. In some cases, model-assisted reasoning was useful for formulating hypotheses or identifying which findings deserved further review. However, the final decision depended on technical verification, scope confirmation, and analyst judgment. This point is particularly important because both automation and LLM-assisted support are useful only when integrated into a workflow that includes expert validation.

7.5.4 Role of Automated Tools in the Findings Generation Process

Automated tools played an important role in the generation and organization of findings. During the external assessment, reconnaissance tools supported the identification of domains, subdomains, DNS records, public exposure, and historical endpoints. Network scanning tools then helped determine which hosts and services were reachable, while vulnerability scanners provided an initial view of known weaknesses and misconfigurations.

During the internal assessment, automated tools were useful for mapping reachable systems, identifying exposed services, and supporting the analysis of internal attack paths. Platforms such as Nessus and Pentera contributed in different ways: Nessus supported broad vulnerability identification, while Pentera helped examine whether some weaknesses could be associated with more practical validation or attack-path scenarios. The combination of these tools helped the assessment team move from detection to prioritization.

However, the role of automated tools remained supportive rather than conclusive. Their outputs provided technical indicators, but the findings became meaningful only after interpretation and validation. This was particularly important when the same issue appeared in different tool outputs or when a vulnerability had to be assessed in relation to business impact, asset criticality, or exploitability under the engagement constraints.

The findings generation process therefore reflected the practical balance discussed throughout the thesis. Automation improved visibility and efficiency, LLM-assisted support helped organize and reason about technical evidence, and human validation transformed preliminary outputs into reliable findings. This process was especially important in an enterprise context, where the complexity of the environment made isolated manual analysis insufficient, while the limitations of automation prevented tool outputs from being accepted without verification.

7.6 Observed Benefits and Practical Limits

The practical activity provided a useful opportunity to observe some benefits and limits of the methodological perspective discussed in the thesis. The engagement did not constitute a formal validation of a new model, since the project was conducted within the rules, procedures, and constraints of a professional consulting activity. However, it made it possible to examine how structured planning, automated tools, LLM-assisted support, and human validation can interact in a real enterprise penetration testing workflow.

The benefits observed during the project were not limited to the identification of vulnerabilities. They also concerned the way in which the assessment was organized, how information was collected and interpreted, how findings were prioritized, and how technical evidence was transformed into conclusions useful for remediation. At the same time, the experience also showed the limits of each component when considered in isolation. Frameworks provided structure but did not solve operational complexity by themselves; automated tools improved visibility but required validation; LLM-assisted support helped organize reasoning but could not provide evidence; and human oversight remained necessary throughout the process.

7.6.1 Observed Benefits of a Structured Methodological Perspective

A structured methodological perspective helped organize the activity across its different phases and assessment dimensions. Since the project included external perimeter testing, internal network assessment, and physical security evaluation, a purely informal approach would have made it more difficult to maintain coherence between objectives, evidence, findings, and reporting. The use of a structured approach helped keep the engagement aligned with a logical progression from scope definition to reconnaissance, vulnerability analysis, validation, and reporting.

This benefit was visible during the transition between phases. The external assessment required reconnaissance before meaningful vulnerability analysis could be performed. The internal assessment required mapping and service enumeration before reasoning about internal attack paths. The physical assessment required observation and planning before controlled access attempts could be evaluated. In each case, structure helped avoid treating the activity as a sequence of disconnected technical actions.

The structured approach also helped preserve alignment with the rules of engagement. Since potentially disruptive techniques were excluded from the assessment, the team had to select validation activities carefully. Methodological organization made it easier to distinguish between acceptable testing actions and activities that would have exceeded the agreed boundaries. This was especially relevant in an enterprise environment where business continuity and operational safety had to remain central considerations.

Another observed benefit concerned reporting and communication. Findings collected across different assessment areas had to be organized in a way that could support remediation and decision-making. A structured approach helped connect each finding to evidence, affected assets, possible impact, and recommended remediation priorities. In this sense, the project showed the importance of treating penetration testing as an organized assessment process, rather than as a collection of independent technical checks.

7.6.2 Observed Benefits and Limits of Automation

Automation provided clear benefits in terms of efficiency, scalability, and initial technical visibility. The external perimeter involved multiple domains, subdomains, services, and publicly exposed assets associated with the corporate group. Without automated support,

collecting and organizing this information would have required considerably more time and would likely have produced a less complete initial view of the attack surface.

Automated tools were useful during domain intelligence, subdomain enumeration, DNS analysis, service discovery, vulnerability scanning, and attack-path validation. They allowed the assessment team to process large amounts of technical information and identify areas that required deeper manual analysis. This was particularly important in the early phases of the engagement, where the objective was to reduce uncertainty and obtain a more accurate understanding of the target environment.

In the internal assessment, automation also supported network mapping and vulnerability identification. The ability to identify reachable systems, exposed services, and possible weaknesses helped the team focus manual effort on the most relevant areas. In this sense, automation was useful mainly because it increased breadth and supported prioritization, while human analysis remained responsible for depth and final validation.

The project also showed that automation can help compare different sources of evidence. Outputs produced by tools such as Nmap, Nessus, Pentera, Maltego, and supporting OSINT utilities could be compared to identify overlapping indications, inconsistencies, or findings requiring confirmation. This improved the organization of the assessment, provided that the outputs were not accepted mechanically but interpreted within the specific context of the engagement.

At the same time, automation had clear limits. Tool outputs could include false positives, duplicated results, incomplete information, or findings that were technically correct but not relevant in the specific environment. Automated severity ratings also did not always reflect the real business or operational impact of an issue. For this reason, automation was useful as a support layer, but it could not replace manual verification, contextual interpretation, or professional judgment.

7.6.3 Observed Benefits and Limits of LLM-Assisted Analytical Support

LLM-assisted analytical support provided benefits mainly during the interpretation and organization of technical information. The engagement produced heterogeneous outputs from reconnaissance tools, network scanners, vulnerability scanners, automated validation platforms, manual observations, and physical assessment notes. In this context, LLM-assisted support helped summarize information, compare outputs, structure preliminary hypotheses, and reason about possible relationships between findings.

One observed benefit was the reduction of analytical effort during the preliminary review of tool-generated outputs. LLM-assisted support could help reorganize technical data, explain possible relevance, and suggest which elements deserved further investigation. This was useful when dealing with large volumes of information or when different tools described related issues using different terminology.

Another useful contribution concerned attack-path reasoning. LLM-assisted support helped organize possible relationships between weaknesses, exposed services, weak credentials, insecure protocols, and segmentation issues. This did not replace technical

verification, but it supported the generation of hypotheses that could then be evaluated by the assessment team. Its value was therefore mainly analytical: it helped structure reasoning, but did not determine conclusions.

The project also made clear the limits of this type of support. LLM-generated outputs could be generic, incomplete, or based on assumptions not fully supported by the available evidence. The usefulness of the output depended strongly on the quality of the prompt and on the amount of context provided. More importantly, the model could not independently verify whether a vulnerability was exploitable, whether an attack path was feasible, or whether a finding had real business impact.

For this reason, LLM-assisted support remained subordinate to human validation and evidence-based reasoning. It was useful when treated as a support for analysis, summarization, and hypothesis generation, but it could not be used as a source of evidence by itself. The practical experience therefore suggests a cautious interpretation: LLMs can help analysts manage complexity, but their contribution depends on expert guidance and technical verification.

7.6.4 Role of Human Oversight in Final Validation

Human oversight was essential throughout the project and represented the final control layer of the assessment. Automated tools and LLM-assisted support produced useful inputs, but the final validation of findings required technical expertise, contextual understanding, and professional judgment. This was especially important because the engagement involved production environments, operational constraints, and sensitive enterprise assets.

Human validation was necessary to distinguish between potential exposure and confirmed risk. Some tool-generated findings required additional verification before being considered reliable. Others had to be deprioritized because they were not exploitable in the specific environment or because they affected assets with limited relevance to the agreed scope. This showed that the value of the assessment depended not on the quantity of findings produced, but on the analyst's ability to interpret and validate evidence.

Human oversight was also essential for assessing impact. The severity of a finding could not be determined only through generic scores or automated classifications. It had to be evaluated in relation to asset criticality, business function, exposure level, exploitability, and the possibility of chaining the weakness with other conditions. This was particularly relevant for findings involving administrative access, exposed secrets, internal network weaknesses, and physical access control issues.

Finally, human supervision ensured that the engagement remained aligned with professional and ethical constraints. The assessment team had to decide which validation activities were appropriate, which actions could create operational risk, and which findings were sufficiently supported by evidence to be included in the final reporting process. In this sense, human oversight was not only a safeguard, but a necessary condition for producing reliable and professionally defensible penetration testing results.

Overall, the practical activity suggests that the combination of structured methodology, automation, LLM-assisted support, and human validation can be useful in enterprise penetration testing. However, the case should not be read as a formal validation of the

hybrid perspective. It is better understood as an applied example showing how these elements can coexist in a real engagement, while also highlighting the limits that require analyst control, technical evidence, and contextual judgment.

7.7 Limits of the Application

Although the practical activity provided a useful opportunity to discuss the methodological perspective developed in the thesis, its limitations must be clearly acknowledged. The case discussed in this chapter was based on a specific engagement, conducted within a defined organizational environment, under specific rules of engagement, and within the boundaries of an internship-based professional project. For this reason, the observations derived from the case cannot be interpreted as universally generalizable to all enterprise penetration testing scenarios.

The purpose of the case study is therefore not to provide statistical validation or to establish a definitive model for all organizations. Rather, it offers an applied example of how structured methodology, automated tools, LLM-assisted support, and human validation may interact in a practical penetration testing workflow. Its value lies in making the discussion more concrete, while also showing that additional applications would be necessary to evaluate the broader applicability of the perspective across different sectors, infrastructures, and organizational contexts.

7.7.1 Constraints Related to the Internship Project

The first limitation concerns the nature of the project itself. The activity was conducted during an internship experience and within a professional consulting engagement. As a result, the scope, depth, timing, and disclosure of the assessment were influenced by operational, contractual, and confidentiality constraints. These constraints are normal in professional penetration testing, but they affect the way in which the case can be described and analyzed in an academic thesis.

Confidentiality requirements represented one of the most important constraints. The client name, specific brands, internal architecture, hostnames, IP addresses, screenshots, detailed vulnerability evidence, and other identifiable information could not be disclosed. For this reason, the discussion necessarily relies on anonymized descriptions and generalized references to findings, tools, and assessment phases. This allows the thesis to discuss the methodological relevance of the engagement without exposing sensitive information.

A second constraint concerns the agreed rules of engagement. As discussed in Section 7.1.4, activities capable of producing operational disruption were excluded from the assessment. This influenced the validation process, since some scenarios could not be tested aggressively even when they were theoretically relevant. The assessment therefore prioritized controlled validation, proportionate testing, and evidence-based reasoning within the authorized boundaries.

The black-box nature of the assessment also introduced specific limits. Since the team operated with limited prior knowledge of the target environment, some aspects

of the infrastructure could only be inferred from observable evidence, tool outputs, and authorized internal access. This approach was useful for approximating an attacker-oriented perspective, but it also meant that some architectural details, internal dependencies, and compensating controls were not always fully visible from the beginning of the engagement.

7.7.2 Activities Not Fully Completed or Not Generalizable

A further limitation concerns the fact that not all potentially relevant activities could be explored with the same level of depth. Enterprise environments are complex, and the duration of the project, although significant, necessarily imposed practical limits on the amount of testing that could be performed. The assessment lasted approximately three months, but the target environment included multiple companies, brands, assets, services, and security dimensions. As a result, the activity had to prioritize the areas considered most relevant within the agreed scope.

Some findings required additional validation or remediation follow-up that fell outside the direct scope of the thesis. For example, the assessment could identify and validate certain weaknesses, but the complete remediation process, long-term monitoring of corrective actions, and reassessment after fixes were not the central focus of this work. This limits the possibility of evaluating the full lifecycle of improvement resulting from the penetration test.

The physical security assessment also had practical boundaries. Although it provided relevant observations about access control and supervision weaknesses, it was conducted under specific authorization and safety conditions. Therefore, its results should be interpreted in relation to the particular facilities, procedures, and timing of the engagement. They cannot automatically be generalized to all sites, offices, or operational contexts of the organization.

Similarly, the internal network assessment was influenced by the access position provided to the assessment team and by the scope agreed with the client. The observed internal attack paths and segmentation issues were meaningful for the tested environment, but they do not necessarily represent all possible internal security conditions across the entire corporate group. Additional testing from different access points, network segments, or user privilege levels could have produced further evidence.

7.7.3 Limits in Evaluating Broader Applicability

The case study provides useful material for discussing the practical value of the hybrid methodological perspective, but it does not allow broad generalizations without further validation. The engagement concerned a single organization operating in the fashion sector. Although the environment was complex and enterprise-oriented, other sectors may present different constraints, architectures, regulatory requirements, threat models, and operational priorities.

For example, financial institutions, healthcare organizations, industrial environments, public administrations, and technology companies may require different testing priorities and different balances between automation, manual validation, and governance integration.

The usefulness of a hybrid perspective may remain relevant across these contexts, but its practical implementation would likely change depending on the organization's structure, maturity, exposure, and risk tolerance.

Another limitation concerns the evaluation of LLM-assisted support. The project showed that LLMs can help with interpretation, correlation, summarization, and preliminary attack-path reasoning. However, this observation remains qualitative and experience-based. The thesis does not provide a controlled quantitative comparison between workflows with and without LLM assistance. Therefore, the practical benefits observed during the engagement should be interpreted as indicative rather than definitive.

The same consideration applies to automation. Automated tools improved efficiency and visibility during the assessment, but the thesis does not attempt to measure their contribution through formal quantitative metrics such as time saved, coverage increase, false-positive reduction, or validation accuracy. The discussion is based on methodological and practical observation rather than statistical measurement.

Despite these limitations, the case remains useful for the objectives of the thesis. It provides a concrete professional context in which the concepts discussed in the previous chapters can be reconsidered in practice. It also suggests that structured frameworks, automated tools, LLM-assisted support, and human validation are more meaningful when considered together rather than in isolation. However, this conclusion should be interpreted within the limits of the case study. The practical application supports the final discussion of the thesis, but it does not replace the need for further analysis across additional enterprise contexts.

Chapter 8

Discussion and Conclusions

This final chapter discusses the results of the thesis and draws the main conclusions from the comparative, methodological, and practical analysis developed in the previous chapters. The purpose is not to introduce new technical material, but to provide a final interpretation of the framework comparison, the discussion on automation and LLM-assisted support, and the enterprise penetration testing experience considered in the thesis.

The thesis started from the observation that enterprise penetration testing is increasingly shaped by complexity, automation, and the need for contextual interpretation. In such environments, the isolated adoption of a single methodological framework may be insufficient, while automated tools and LLM-assisted support can contribute to the assessment process only when their outputs are interpreted and validated by human analysts. The previous chapters have therefore examined penetration testing as an activity in which methodological references, tools, analyst reasoning, and validation interact.

This chapter first discusses the main results emerging from the comparative analysis and from the practical application. It then answers the research questions defined in Chapter 3, clarifying how the thesis addresses the initial problem statement. The chapter also summarizes the main methodological and practical contributions of the work, acknowledges its limitations, and outlines possible future developments. In this way, the chapter provides the final synthesis of the thesis while also defining the limits within which its conclusions should be interpreted.

8.1 Discussion of Results

The results of the thesis can be interpreted across two main dimensions. The first dimension concerns the comparative analysis of penetration testing frameworks. This analysis highlighted that each framework provides a specific contribution, but also that no single framework fully addresses all the requirements of enterprise penetration testing. The second dimension concerns the practical application developed through the internship experience, which provided a concrete professional context in which the methodological perspective discussed in the thesis could be observed and interpreted.

Taken together, these results support a cautious interpretation of the thesis argument: enterprise penetration testing benefits from an integrated methodological perspective. Structured frameworks remain useful because they provide order, consistency, and traceability. Automated tools improve efficiency and visibility across large attack surfaces. LLM-assisted support may help analysts organize and interpret technical information. However, these components do not remove the need for human validation, since expert judgment remains necessary to confirm exploitability, assess contextual relevance, and translate technical evidence into meaningful conclusions.

8.1.1 Interpretation of the Framework Comparison

The comparative analysis developed in Chapter 5 showed that the selected frameworks differ significantly across the main evaluation dimensions, including scope, operational depth, formalization, enterprise suitability, automation support, reporting support, and adaptability to hybrid workflows. These differences confirm that penetration testing frameworks should not be interpreted as interchangeable references. Each one reflects a specific methodological orientation and provides value in relation to particular assessment needs.

PTES emerged as a strong operational reference because it provides a clear lifecycle for structuring penetration testing engagements. Its main value lies in its practical structure, which supports the progression from pre-engagement activities to reconnaissance, vulnerability analysis, exploitation, post-exploitation, and reporting. However, the comparison also showed that PTES does not fully address enterprise governance, formal measurement, or modern hybrid workflows involving automation and LLM-assisted support.

OSSTMM contributed a different perspective, centered on structured and measurable security testing. Its strength lies in methodological rigor, repeatability, and formalization. This makes it relevant when the objective is to promote consistency and systematic evaluation. At the same time, its practical complexity and less direct alignment with hands-on offensive workflows make it less immediately usable as the sole reference for operational enterprise penetration testing.

NIST SP 800-115 provided an institutional and procedural perspective. Its relevance is particularly strong when penetration testing must be aligned with planning, authorization, documentation, reporting, and broader security management processes. This makes it valuable in enterprise and regulated contexts. Nevertheless, its relatively high-level nature and the age of the publication limit its ability to address newer operational realities, such as cloud-native environments, automated attack simulation platforms, and LLM-assisted analytical workflows.

OWASP WSTG showed strong value as a specialized reference for application security testing. Its operational depth is significant within the web application domain, especially for assessing authentication, authorization, input validation, session management, business logic, and APIs. However, its scope remains domain-specific. For this reason, it cannot represent a complete enterprise penetration testing methodology, although it is highly useful as part of a broader integrated approach.

MITRE ATT&CK was interpreted as a complementary adversarial knowledge framework

rather than as a full penetration testing methodology. Its value lies in supporting attack-path reasoning, adversarial modeling, and the interpretation of findings in relation to attacker behavior. It is particularly useful for connecting isolated findings to plausible compromise scenarios. However, it does not provide the procedural structure needed to plan, execute, and report a complete penetration testing engagement.

The overall interpretation of the framework comparison is therefore that each framework contributes to a different layer of enterprise penetration testing. PTES supports operational organization, OSSTMM contributes formal rigor, NIST SP 800-115 reinforces procedural discipline, OWASP WSTG provides application-security depth, and MITRE ATT&CK supports adversarial reasoning. The most relevant result is not the identification of a single superior framework, but the recognition that enterprise assessments benefit from a selective and integrated use of multiple methodological references.

8.1.2 Interpretation of the Practical Application

The practical application discussed in Chapter 7 provided an applied perspective on the methodological discussion developed throughout the thesis. The enterprise penetration testing project involved a complex organization operating in the fashion sector and included external perimeter assessment, internal network assessment, and physical security assessment. For this reason, the case was useful for discussing how methodological structure, automation, analyst reasoning, and validation appear in a real professional engagement.

It is important to clarify that the practical activity was not designed as a formal validation of the methodological perspective proposed in the thesis. The assessment was conducted within the rules, procedures, and operational constraints of the consulting engagement. Therefore, the role of the case study is mainly interpretative: it provides a concrete context in which the concepts discussed in the previous chapters can be examined retrospectively.

The experience showed that enterprise penetration testing cannot be reduced to the execution of isolated technical checks. The assessment required scope definition, rules of engagement, reconnaissance, tool-assisted discovery, vulnerability analysis, controlled validation, attack-path reasoning, reporting, and continuous attention to operational constraints. The black-box nature of the engagement and the presence of explicit restrictions, including the exclusion of denial-of-service and brute-force activities, also highlighted the need to balance assessment realism with operational safety.

The external perimeter assessment illustrated the value of combining OSINT, subdomain enumeration, DNS analysis, service discovery, vulnerability scanning, and manual verification. The internal assessment made it possible to observe the importance of mapping reachable systems, evaluating internal services, considering segmentation, and reasoning about possible internal attack paths. The physical security component further broadened the perspective of the assessment, showing that enterprise security exposure is not limited to digital assets alone.

The project also provided a practical example of the role of automation. Tools such as Nmap, Maltego, Nessus, Pentera, and additional supporting utilities improved visibility, accelerated discovery, and supported preliminary validation. At the same time, the

experience showed that tool outputs could not be used as conclusions without further review. They had to be interpreted, compared, filtered, and validated by the assessment team before being considered reliable findings.

LLM-assisted support was useful mainly in result interpretation, technical summarization, finding correlation, and preliminary attack-path reasoning. Its contribution was relevant because the assessment generated heterogeneous information from multiple sources. Nevertheless, the practical case also made clear that LLM outputs could not be treated as final evidence. Their value depended on the quality of the analyst's guidance and on subsequent human validation.

The main interpretation of the practical application is therefore more limited than a formal validation of the hybrid approach. The case suggests that structured methodology, automated tools, LLM-assisted support, and human validation can coexist within a real enterprise assessment. However, the effectiveness of this integration remains dependent on professional control, engagement boundaries, technical verification, and contextual judgment.

8.1.3 Coherence with the Research Objectives

The results of the thesis can be related to the research objectives defined in Chapter 3, although with different degrees of strength. Some objectives were addressed mainly through the theoretical and comparative parts of the work, while others were discussed through the practical experience described in Chapter 7. For this reason, the coherence between objectives and results should be understood in methodological and interpretative terms, rather than as a formal experimental validation.

The first objective was to compare major penetration testing frameworks through explicit evaluation criteria. This objective was addressed through the comparison model defined in Chapter 4 and applied in Chapter 5. The analysis made it possible to discuss how the selected frameworks differ and how their strengths and limitations can be interpreted in relation to enterprise penetration testing.

The second objective was to assess the suitability of these frameworks for enterprise environments. This objective was addressed by considering not only the general structure of each framework, but also dimensions such as enterprise suitability, reporting and validation support, support for automation, and adaptability to hybrid workflows. The comparison suggested that the frameworks are useful in different ways, but that none of them is sufficient on its own to cover the full complexity of enterprise assessments.

The third objective was to analyze the contribution and limitations of automated tools in practical penetration testing. This was developed in Chapter 6 and later discussed in relation to the internship experience. The thesis highlighted that automation can improve efficiency, scalability, and technical visibility, but also that automated outputs require interpretation, validation, and contextualization before they can support reliable conclusions.

The fourth objective was to examine the role of LLM-assisted support in result interpretation and attack-path reasoning. This objective was addressed by discussing possible uses and limitations of LLMs in Chapter 6 and by considering their role in the practical

case. The resulting interpretation is deliberately cautious: LLM-assisted support may help analysts manage complex information, but it cannot replace technical verification or expert judgment.

The fifth objective was to define and discuss a hybrid methodological approach. This objective represents the connecting element of the thesis. The hybrid perspective emerged from the framework comparison, the analysis of automation, the discussion of LLM-assisted support, and the need for human validation. It is not presented as a new formal standard, but as an interpretative perspective for relating existing methodologies, tools, analytical support, and expert reasoning.

The sixth objective concerned the use of a real enterprise application to discuss the practical relevance of this perspective. The case described in Chapter 7 contributed to this objective by providing a concrete professional context in which some of the issues discussed in the thesis could be observed. However, the practical application should not be read as a formal validation of the model. It is better understood as an applied discussion, limited by the rules, procedures, and confidentiality constraints of the consulting engagement.

Overall, the thesis remains aligned with its initial objectives, but its conclusions should be interpreted with the appropriate level of caution. The work does not claim to produce a universal methodology or a quantitatively validated model. Rather, it provides a structured comparative and applied discussion of how penetration testing can be interpreted in enterprise environments through the interaction of frameworks, automation, LLM-assisted support, and expert human validation.

8.2 Answers to the Research Questions

This section provides a final synthesis of the research questions presented in Section 3.4. The answers are based on the comparative analysis, the discussion of automation and LLM-assisted support, and the practical experience described in the previous chapter.

The purpose of this section is not to repeat the full analysis already developed in the thesis, but to clarify how the different parts of the work contribute to the initial questions. The answers should therefore be read as a reasoned synthesis of the thesis, rather than as definitive or universally generalizable conclusions.

8.2.1 Synthesis of the Answers

The first research question asked whether a single penetration testing framework can adequately support enterprise testing activities. The analysis suggests that a single framework can provide useful guidance, but is unlikely to address all the methodological and operational needs of complex enterprise assessments on its own. PTES offers a strong operational lifecycle, OSSTMM introduces formal rigor, NIST SP 800-115 provides institutional and procedural structure, OWASP WSTG supports application security testing in depth, and MITRE ATT&CK helps with adversarial reasoning. However, each of these references also leaves some aspects only partially covered. For this reason, enterprise penetration testing appears better supported by a selective and integrated use of multiple

methodological references.

The second research question concerned the criteria that are most relevant for comparing penetration testing frameworks in enterprise contexts. The comparison model proposed in the thesis used seven criteria: coverage scope, operational depth, level of formalization, enterprise suitability, support for automation, reporting and validation support, and adaptability to hybrid workflows. These criteria helped structure the analysis and made the comparison more transparent. They also made it possible to distinguish between frameworks that are useful for operational execution, frameworks that provide stronger procedural or formal structure, and frameworks that are better understood as specialized or complementary references.

The third research question asked how automated tools affect the quality and efficiency of penetration testing. The thesis suggests that automation is valuable mainly because it improves speed, scalability, and initial visibility over the target environment. This is particularly important in enterprise contexts, where the attack surface may be too broad to manage through manual activity alone. At the same time, automation does not automatically improve the quality of the assessment. Tool outputs may contain false positives, incomplete results, duplicate findings, or indications that are technically correct but not relevant in the specific environment. Their usefulness therefore depends on how they are reviewed, contextualized, and validated by the analyst.

The fourth research question concerned the possible role of LLM-assisted support in interpreting findings and exploring attack paths. The thesis discusses LLMs as analytical aids rather than as autonomous testing tools. Their possible value lies in helping analysts summarize tool outputs, compare information from different sources, organize technical evidence, formulate preliminary hypotheses, and reason about possible attack paths. However, this support remains limited by the lack of direct environmental awareness and by the possibility of inaccurate or unsupported inferences. For this reason, LLM-assisted outputs must be treated as suggestions to be verified, not as evidence in themselves.

The fifth research question asked what benefits may derive from a hybrid approach combining frameworks, automation, LLM-assisted support, and human validation. The main benefit is the possibility of combining complementary elements. Frameworks provide methodological structure, automated tools support efficiency and coverage, LLM-assisted support may help manage analytical complexity, and human validation remains necessary to establish accuracy, relevance, and accountability. The practical case offers an example of how these elements can coexist in a professional engagement, although it does not provide a formal validation of the approach. It suggests that a hybrid perspective can be useful for interpreting enterprise penetration testing activities, especially when the assessment involves multiple technical and organizational dimensions.

Overall, the research questions converge toward a cautious conclusion. Enterprise penetration testing can be better understood as a structured and adaptive process in which methodological references, automated tools, analytical support, and human expertise interact. The thesis does not propose a new universal standard, but it argues that a hybrid methodological perspective can help discuss the complexity, scale, and interpretative requirements of modern enterprise assessments.

8.3 Main Contributions of the Thesis

This section summarizes the main contributions of the thesis, taking into account the methodological nature of the work and the limits of the practical application. The contribution of the thesis should not be understood as the definition of a new penetration testing standard or as the experimental validation of a new methodology. Rather, its value lies in the structured comparison of existing references, in the discussion of their possible integration with automation and LLM-assisted support, and in the interpretation of a real enterprise penetration testing experience through this perspective.

The contributions can therefore be grouped into two main areas. The first concerns the methodological organization of the analysis, especially the comparison of penetration testing frameworks and the discussion of hybrid workflows. The second concerns the practical perspective provided by the internship experience, which allowed some of the issues discussed in the thesis to be considered in relation to a real professional engagement.

8.3.1 Methodological Contributions

The first methodological contribution of the thesis is the definition of a structured comparison model for penetration testing frameworks. Instead of presenting the selected frameworks only through descriptive summaries, the thesis introduced explicit evaluation criteria aimed at assessing their relevance for enterprise penetration testing. These criteria made it possible to compare the frameworks in a more transparent and consistent way, considering dimensions such as coverage scope, operational depth, formalization, enterprise suitability, automation support, reporting and validation support, and adaptability to hybrid workflows.

A second contribution is the interpretation of the selected frameworks as complementary references rather than as mutually exclusive alternatives. The analysis suggested that PTES, OSSTMM, NIST SP 800-115, OWASP WSTG, and MITRE ATT&CK contribute to different aspects of penetration testing. This perspective supports a more flexible understanding of methodological adoption, where different frameworks can be selected or combined according to the objectives, scope, and constraints of the engagement.

A further methodological contribution is the discussion of hybrid methodological integration. The thesis considered how structured frameworks, automated tools, LLM-assisted analytical support, and human validation may interact within enterprise penetration testing workflows. This contribution does not consist in proposing a new formal standard, but in offering a reasoned perspective on how existing methodologies and operational supports can be related in complex enterprise assessments.

8.3.2 Practical Contributions

The main practical contribution of the thesis is the discussion of a real enterprise penetration testing experience conducted in a consulting context during the internship at KPMG. This practical case made it possible to connect the methodological analysis to an actual assessment involving external perimeter testing, internal network assessment, and physical

security evaluation. In this sense, the thesis does not remain limited to a theoretical comparison of frameworks, but also considers how some of the discussed concepts appear in a concrete professional setting.

The practical application also contributes to the discussion of automated tools in enterprise penetration testing. Tools such as Nmap, Maltego, Nessus, Pentera, and other supporting utilities were considered as components of a broader workflow, rather than as isolated technical instruments. This helped clarify how automation can improve visibility, scalability, and preliminary validation, while still requiring careful interpretation and expert control.

Another practical contribution concerns the discussion of LLM-assisted support within an assessment workflow. The thesis considered how LLMs may assist analysts in interpreting outputs, correlating findings, organizing technical evidence, and reasoning about possible attack paths. At the same time, the practical case highlighted that such support must remain subordinate to human supervision, since LLM-generated outputs cannot independently validate technical conditions or replace professional judgment.

Finally, the thesis emphasizes the importance of human validation as the final control layer of the assessment. The practical experience showed that the quality of enterprise penetration testing depends not only on the tools used or the frameworks adopted, but also on the ability of analysts to verify findings, assess impact, prioritize risks, and communicate results in a way that supports remediation and decision-making.

8.4 Limitations of the Study

Although the thesis provides a structured comparative and practical discussion of enterprise penetration testing, several limitations must be acknowledged. These limitations concern the scope of the frameworks analyzed, the qualitative nature of the comparison model, the characteristics of the practical case, and the way LLM-assisted support was evaluated.

The thesis does not aim to define a new penetration testing standard or a complete operational methodology applicable to every organization. Its objective is more limited: to discuss how existing frameworks, automated tools, LLM-assisted support, and human validation can be related in enterprise-oriented assessments. For this reason, the conclusions should be interpreted as analytical observations rather than as prescriptive rules.

8.4.1 Scope Limitations

A first limitation concerns the selection of the frameworks considered in the analysis. The thesis focused on PTES, OSSTMM, NIST SP 800-115, OWASP WSTG, and MITRE ATT&CK because they represent different and relevant perspectives on penetration testing, security assessment, application security, and adversarial modeling. However, this selection is not exhaustive. Other standards, methodologies, internal company practices, and sector-specific approaches could also be relevant in professional penetration testing.

This limitation is partly a consequence of the need to keep the work focused and manageable. Including a larger number of references would have broadened the comparison,

but it would also have made the analysis less coherent and more difficult to connect to the practical case. Therefore, the selected frameworks should be understood as representative references, not as the complete set of methodologies available for enterprise security assessment.

A second scope limitation concerns the enterprise context itself. Enterprise environments can differ significantly depending on sector, size, infrastructure maturity, regulatory requirements, cloud adoption, security organization, and threat model. The thesis discusses enterprise penetration testing in general terms, but the practical application is based on a single professional engagement. For this reason, the observations derived from the case cannot automatically be generalized to all enterprise environments.

The practical case concerned an organization operating in the fashion sector and involved external, internal, and physical assessment dimensions. This made the case useful for discussing a broad enterprise security perspective, but it also means that the results remain tied to the characteristics of that specific engagement. Other sectors, such as finance, healthcare, public administration, energy, or industrial production, may involve different constraints, priorities, and risk profiles.

8.4.2 Methodological Limitations

A second group of limitations concerns the methodological nature of the thesis. The comparison model developed in Chapter 4 is qualitative rather than quantitative. It is based on explicit evaluation criteria, but it does not assign numerical scores or produce a formal ranking of the selected frameworks. This choice is consistent with the purpose of the work, but it also means that the comparison remains interpretative.

The use of qualitative criteria improves transparency and helps organize the analysis, but it cannot remove the role of judgment. Criteria such as enterprise suitability, operational depth, reporting support, or adaptability to hybrid workflows require interpretation. Another analyst could give different weight to the same dimensions, depending on professional experience, engagement type, or organizational context.

Another limitation concerns the practical application. The enterprise case provides a useful professional context, but it was not designed as an experimental validation of the comparison model or of the hybrid methodological perspective. The activity followed the rules, procedures, and constraints of the consulting engagement. The model developed in the thesis was therefore used mainly to interpret the experience retrospectively, rather than to govern the project as a prescribed methodology.

For this reason, the practical case should be understood as an applied discussion, not as a formal validation. It helps show how some of the issues discussed in the thesis appear in a real enterprise assessment, but it does not prove that the proposed perspective would work in the same way across different organizations, sectors, or engagement types. Further case studies would be needed to evaluate this more systematically.

The evaluation of LLM-assisted support is also limited. The thesis discusses its role on the basis of methodological reasoning and practical observation, but it does not include a controlled comparison between workflows with and without LLM assistance. As a result, the observed benefits should be interpreted cautiously. They suggest possible usefulness in

interpretation, correlation, and preliminary reasoning, but they do not quantify the impact of LLM support on time efficiency, analytical accuracy, false-positive reduction, or analyst performance.

Finally, the thesis is limited by confidentiality constraints related to the practical case. Since the project was conducted in a professional consulting context, several technical details, screenshots, hostnames, IP addresses, infrastructure information, and specific evidence could not be disclosed. This reduces the level of technical granularity that can be presented, but it is necessary to preserve confidentiality and professional responsibility.

Taken together, these limitations define the proper scope of the thesis. The work provides a structured discussion of enterprise penetration testing and offers an interpretative perspective on the interaction between frameworks, automation, LLM-assisted support, and human validation. Its conclusions should therefore be read as a basis for further analysis and refinement, rather than as a definitive or universally validated model.

8.5 Future Developments

The limitations discussed in the previous section also indicate possible directions for future work. Since the thesis is based on a qualitative comparison and on a single practical case, future developments should mainly focus on testing the proposed perspective in broader and more differentiated contexts. This would make it possible to understand which aspects of the analysis remain valid across different environments and which ones depend more strongly on the specific conditions of the engagement.

Future work should not necessarily aim to transform the perspective discussed in this thesis into a completely new methodology. A more realistic objective would be to refine the comparison model, apply it to additional contexts, and study more systematically the role of automation and LLM-assisted support in professional penetration testing workflows. In this sense, the thesis can be considered a starting point for further analysis rather than a closed methodological proposal.

8.5.1 Extension of the Comparison Model

A first direction for future work concerns the refinement and possible extension of the comparison model developed in this thesis. The model was intentionally designed as a qualitative and thesis-specific analytical tool, focused on the needs of enterprise penetration testing. Future research could examine whether the selected criteria remain adequate when applied to more specialized assessment contexts, such as cloud infrastructures, Active Directory environments, industrial systems, or organizations subject to strict compliance requirements.

The model could also be extended by including additional frameworks and standards beyond those considered in this thesis. This would make the comparison more complete and would help clarify whether the conclusions reached in this work remain valid when a wider set of methodological references is analyzed. At the same time, such an extension should preserve the coherence of the model, avoiding an excessive expansion that would

make the comparison less focused.

Another possible development concerns the way the criteria are applied. This thesis adopted a qualitative approach, without assigning numerical scores or producing a ranking of the frameworks. Future studies could explore whether some criteria can be translated into structured assessment scales or expert-based evaluation matrices. This could make comparisons more systematic, but it should be done carefully, since excessive quantification may create a misleading impression of precision in an area that still requires professional interpretation.

8.5.2 Further Validation in Additional Enterprise Contexts

A second direction for future work concerns the application of the hybrid methodological perspective to additional enterprise contexts. The practical case discussed in this thesis was based on one penetration testing project conducted in the fashion sector. Although the engagement included external, internal, and physical assessment dimensions, it remains a single case and cannot be used to draw general conclusions about all enterprise environments.

Future studies could consider organizations operating in different sectors, such as finance, healthcare, public administration, manufacturing, energy, or technology. These environments may present different infrastructures, regulatory constraints, business priorities, and threat models. For example, a financial organization may place stronger emphasis on transaction integrity, auditability, and data confidentiality, while an industrial environment may prioritize continuity, safety, and resilience of operational processes. Comparing these contexts would help clarify how the proposed perspective should be adapted according to sector-specific requirements.

Further work could also consider different types of security engagements. The case discussed in this thesis was framed as an enterprise penetration testing activity, but similar questions may emerge in red team operations, purple team exercises, cloud assessments, internal compromise assessments, or recurring security validation programs. Studying these cases would help determine whether the interaction between frameworks, automation, LLM-assisted support, and human validation changes depending on the type of assessment.

A multi-case analysis would also be useful. Comparing several practical engagements would make it possible to identify recurring patterns, differences, and context-specific adaptations. This would provide stronger empirical support than a single case study and could help distinguish between elements that are generally useful and elements that depend mainly on the specific scope, maturity, infrastructure, or constraints of the organization under assessment.

8.5.3 More Systematic Study of LLM-Assisted Support

A third direction for future work concerns the more systematic study of LLM-assisted support in penetration testing. This thesis discussed LLMs as analytical aids for result interpretation, finding correlation, preliminary validation, and attack-path reasoning. The practical case suggested that this type of support can be useful, but also showed that its

value depends on careful supervision and on the availability of reliable technical evidence.

Future work could compare penetration testing workflows with and without LLM-assisted support. Such a comparison could help evaluate whether LLMs reduce analysis time, improve the organization of findings, support correlation across different sources, or help analysts reason about possible attack paths more efficiently. This would make it possible to move beyond qualitative observation and obtain more structured evidence about the actual impact of LLM-assisted support.

Another useful direction would be to study the risks associated with LLM use in professional assessment workflows. Particular attention should be given to hallucinations, unsupported inferences, overreliance, prompt sensitivity, and the possibility of accepting plausible but incorrect reasoning. These risks are relevant because penetration testing conclusions must be based on evidence, not only on coherent explanations.

Future developments could also focus on defining controlled procedures for LLM-assisted analysis. Such procedures could clarify when LLMs may be used, which information can be provided to them, how their outputs should be reviewed, and which validation steps are necessary before model-assisted reasoning can influence the assessment. This would help move LLM-assisted support from an informal analytical aid toward a more disciplined and evidence-controlled component of the workflow.

Finally, future work could examine the interaction between LLM-assisted support and structured adversarial references such as MITRE ATT&CK. Since ATT&CK provides a formal vocabulary for describing adversary behavior, it may help guide the organization of attack-path hypotheses. However, any model-generated mapping or interpretation should remain grounded in technical evidence and reviewed by an analyst before being used in a professional assessment.

Overall, future work can extend the thesis in three main directions: refining the comparison model, applying the hybrid perspective to additional enterprise contexts, and studying LLM-assisted support more systematically. These developments would help clarify the practical value and the limits of the perspective discussed in this thesis, while preserving the central role of technical evidence, operational constraints, and human judgment in enterprise penetration testing.

Bibliography

- [1] Karen Scarfone, Murugiah Souppaya, Amanda Cody, and Angela Orebaugh. *Technical Guide to Information Security Testing and Assessment*. Tech. rep. NIST Special Publication 800-115. National Institute of Standards and Technology, 2008. DOI: 10.6028/NIST.SP.800-115.
- [2] Penetration Testing Execution Standard. *The Penetration Testing Execution Standard*. 2014. URL: <http://www.pentest-standard.org/> (visited on 04/27/2026).
- [3] OWASP Foundation. *OWASP Web Security Testing Guide*. 2024. URL: <https://owasp.org/www-project-web-security-testing-guide/> (visited on 04/27/2026).
- [4] ISECOM. *OSSTMM 3 – Open Source Security Testing Methodology Manual*. 2010. URL: <https://www.isecom.org/OSSTMM.3.pdf> (visited on 04/27/2026).
- [5] MITRE Corporation. *MITRE ATT&CK*. 2025. URL: <https://attack.mitre.org/> (visited on 04/27/2026).
- [6] Gelei Deng et al. «PentestGPT: Evaluating and Harnessing Large Language Models for Automated Penetration Testing». In: *33rd USENIX Security Symposium*. USENIX Association, 2024. URL: <https://www.usenix.org/conference/usenixsecurity24/presentation/deng>.
- [7] National Institute of Standards and Technology. *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*. Tech. rep. NIST AI 100-1. National Institute of Standards and Technology, 2023. DOI: 10.6028/NIST.AI.100-1.
- [8] OWASP Foundation. *OWASP Top 10 for Large Language Model Applications*. 2025. URL: <https://owasp.org/www-project-top-10-for-large-language-model-applications/> (visited on 05/18/2026).
- [9] UK AI Security Institute. *Our evaluation of Claude Mythos Preview’s cyber capabilities*. 2026. URL: <https://www.aisi.gov.uk/blog/our-evaluation-of-claude-mythos-previews-cyber-capabilities> (visited on 05/19/2026).
- [10] Anthropic. *Project Glasswing: Securing critical software for the AI era*. 2026. URL: <https://www.anthropic.com/glasswing> (visited on 05/19/2026).