A PowerShell User Tracking Model using Active Directory in Organizations

Geoffrey Kimani

145488

CNS 3104

Submitted in Partial Fulfilment of the Requirements of the Bachelor of Science in

Computer Networks and Cybersecurity at the Strathmore University

School of Computing and Engineering Science

Strathmore University

Nairobi, Kenya

May 2023

# DECLARATION AND APPROVAL

I declare that this work has not been previously submitted and approved for the award of a

degree by this or any other University. To the best of my knowledge and belief, the research

proposal contains no material previously published or written by another person except where

due reference is made in the research proposal itself.

Student Name: Geoffrey Kimani

Admission Number: 145488

Student Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Approval

The Proposal of Geoffrey Kimani has been reviewed and approved by Mr. Humphrey Owuor.

Supervisor Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# ACKNOWLEDGEMENT

Throughout the process of researching and preparing this proposal, I have received immense support and guidance from various individuals and articles. I would like to express my deepest gratitude and acknowledge their invaluable contributions. First and foremost, I would like to extend my sincere appreciation to Mr. Humphrey Owuor for your unwavering support and invaluable feedback throughout this journey. Your expertise and guidance have been instrumental in shaping my research and refining my ideas. Your willingness to share your knowledge have been a constant source of inspiration for me. I am also grateful to the faculty members of the Networking and Cybersecurity Department [Panellists, Dr. Vitalis Ozianyi and Mr. James Gikera] for their commitment to academic excellence. Their dedication to imparting knowledge and their willingness to engage in insightful discussions have played a crucial role in shaping my research perspectives.

# ABSTRACT

This proposal presents a research project aimed at developing a PowerShell user tracking model utilizing Active Directory (AD) in organizations. This study focuses on the growing need for effective user tracking and monitoring in organizations. With the increasing complexity of cyber threats and the importance of protecting sensitive data, it is crucial to establish robust mechanisms to track user activities within an organization's network infrastructure. The primary objective of this research project is to develop a PowerShell-based model that utilizes Active Directory to track and monitor user activities within an organization. By leveraging PowerShell scripting and the rich functionalities of Active Directory, this model aims to provide administrators with enhanced visibility into user actions, facilitating effective cybersecurity measures. The proposed methodology involves analysing existing PowerShell modulesmto interact with Active Directory and extract relevant user activity data. The model will employ techniques such as event log monitoring, real-time notifications, and correlation analysis to establish a comprehensive user tracking mechanism. The anticipated outcomes of this project include the development of a robust PowerShell user tracking model that integrates seamlessly with Active Directory. By implementing this model, organizations can effectively monitor user actions, detect anomalies, and promptly respond to potential security incidents. The project also aims to evaluate the accuracy and efficiency of the model through extensive testing and validation. The successful implementation of the PowerShell user tracking model using Active Directory will provide organizations with valuable insights into user activities, helping to identify and mitigate potential security risks. By enhancing the organization's cybersecurity posture, this model can contribute to the protection of sensitive data, prevention of unauthorized access, and timely response to security incidents.

# Table of contents

[DECLARATION AND APPROVAL ii](#_Toc137744895)

[ACKNOWLEDGEMENT iii](#_Toc137744896)

[ABSTRACT iv](#_Toc137744897)

[Table of contents v](#_Toc137744898)

[List of figures vi](#_Toc137744899)

[List of Abbreviations vii](#_Toc137744900)

[CHAPTER 1 INTRODUCTION 1](#_Toc137744901)

[1.1 Background Information 1](#_Toc137744902)

[1.2: Problem Statement 2](#_Toc137744903)

[1.3: Objectives 3](#_Toc137744904)

[1.3.1 General Objective 3](#_Toc137744905)

[1.3.2 Specific Objectives 3](#_Toc137744906)

[1.4 Justification 3](#_Toc137744907)

[Gaps in the Current Literature: 4](#_Toc137744908)

[1.5: Scope and Limitations 4](#_Toc137744909)

[1.5.1: Scope 4](#_Toc137744910)

[1.5.2: Delimitations 5](#_Toc137744911)

[1.5.3: Limitations 5](#_Toc137744912)

[CHAPTER 2: LITERATURE REVIEW. 6](#_Toc137744913)

[2.1: Introduction. 6](#_Toc137744914)

[2.2 How a PowerShell User Tracking Model using Active Directory works. 6](#_Toc137744915)

[2.3: Common vulnerabilities affecting a PowerShell User Tracking Model using Active Directory 6](#_Toc137744916)

[2.4: Discussion on papers relating to a PowerShell User Tracking Model using Active Directory 7](#_Toc137744917)

[2.5: Related Works 8](#_Toc137744918)

[2.5.1 Integration A PowerShell User Tracking Model using Active Directory for Enhanced Security Monitoring 8](#_Toc137744919)

[2.5.2 Enhancing User Tracking Capabilities in PowerShell with Active Directory Integration 9](#_Toc137744920)

[2.6: Gaps in related works 10](#_Toc137744921)

[2.7: Conceptual framework 11](#_Toc137744922)

[CHAPTER 3: METHODOLOGY 12](#_Toc137744923)

[3.1: Introduction 12](#_Toc137744924)

[3.2: System Development Methodology 12](#_Toc137744925)

[3.2.1: Empathize. 12](#_Toc137744926)

[3.2.2: Define. 12](#_Toc137744927)

[3.2.3: Ideate. 13](#_Toc137744928)

[3.2.4: Prototype 13](#_Toc137744929)

[3.2.5: Test 13](#_Toc137744930)

[3.2.6: Deliverables 13](#_Toc137744931)

[3.3: Methodology Justification 13](#_Toc137744932)

[3.4: Tools and Techniques 13](#_Toc137744933)

[3.4.1 ORACLE VirtualBox 13](#_Toc137744934)

[3.4.2 Windows 10 ISO 13](#_Toc137744935)

[3.4.3 Server 2019 13](#_Toc137744936)

[References 14](#_Toc137744937)

# List of figures

Figure 2.1 *Overview Event Properties of Event 4103, PowerShell* ………………………………………………….......................................................... 7

Figure 2.2 *Overview of Active Directory Domain Services* …………...... 8

Figure 2.3 Conceptual framework……………………………………………………… 9

Figure 3.1 Prototype methodology model……………………………………………… 1

# List of Abbreviations

AD – Active Directory

WinRM – Windows Remote Management

PS - PowerShell

# CHAPTER 1 INTRODUCTION

## 1.1 Background Information

In today's technology-driven world, organizations heavily rely on cloud computing and enterprise environments to store and process their critical data (Bisong,2017). However, ensuring the security and trustworthiness of user activities within these environments remains a challenge. This research proposal aims to address the gaps in trust modelling for users in an enterprise cloud environment, specifically focusing on the implementation of a PowerShell User Tracking Model using Active Directory. The broad topic of trust modelling in enterprise cloud environments has received significant attention in recent years. Various studies have explored different aspects of user behaviour, authentication, and authorization mechanisms (Marouane et. al.,2019). However, there is a noticeable lack of comprehensive models that specifically address the tracking of PowerShell activities within Active Directory. The gaps in the existing literature become apparent when considering the increasing popularity of PowerShell as a powerful scripting language for system administration and automation tasks. Its extensive capabilities provide users with the ability to perform a wide range of actions within an enterprise environment, including accessing sensitive information, modifying system configurations, and executing potentially malicious commands. However, the lack of effective tracking mechanisms leaves organizations vulnerable to insider threats, unauthorized activities, and potential security breaches.

The significance of addressing these gaps is of paramount importance to organizations as it directly impacts their ability to maintain a secure and trustworthy environment. By developing a PowerShell User Tracking Model using Active Directory, this research aims to provide organizations with a comprehensive solution to monitor and analyse user activities, ensuring accountability and mitigating potential risks. The rationale behind this study is to fill the existing gap in research by proposing a model that combines the capabilities of PowerShell and Active Directory to track and monitor user activities in real-time. The hypothesis is that by implementing an effective tracking mechanism, organizations will have increased visibility into user actions, enabling them to identify and respond to potential security threats more efficiently. To make the background engaging, the story should revolve around the evolution of PowerShell and its integration with Active Directory, highlighting the increasing importance of tracking user activities. The narrative should build upon the historical developments in technology, emphasizing the significance of addressing the gaps identified in the literature.

By structuring the background information effectively, the proposal will provide a concise overview of the research topic, outline the gaps in the existing literature, and establish the significance of addressing these gaps through the proposed PowerShell User Tracking Model. The background section will not be a comprehensive literature review but rather a focused survey that highlights the evolution of research in relevant fields over time.

## 1.2: Problem Statement

In today's interconnected world, organizations heavily rely on computer networks and information systems to conduct their daily operations. However, with the increasing reliance on technology comes the heightened risk of cybersecurity threats and unauthorized access to sensitive data. One of the critical challenges faced by organizations is the lack of a comprehensive user tracking model using Active Directory that can effectively monitor and analyse user activities within the network. Ideally, organizations should have a robust and efficient user tracking system that provides real-time visibility into user actions, such as logins, file access, privilege escalations, and system changes. This system should leverage Active Directory, a widely used directory service in Windows environments, to collect and centralize user-related data from various endpoints and network devices.

However, in the current situation, many organizations lack a comprehensive user tracking model that can accurately monitor and analyse user activities across the network. Existing solutions often focus on specific aspects of user tracking or are limited in their ability to provide a holistic view of user actions. This limitation creates a significant gap in knowledge and poses a substantial security risk as organizations struggle to identify and mitigate insider threats, unauthorized access, and potential data breaches. To address this problem, it is crucial to develop a PowerShell-based user tracking model that leverages Active Directory to collect and analyse user-related data within organizations. This model should incorporate advanced logging mechanisms, data correlation techniques, and anomaly detection algorithms to identify suspicious user behaviour and potential security incidents. By implementing such a model, organizations will be able to enhance their network security posture, detect and respond to security incidents in a timely manner, and ensure compliance with regulatory requirements.

The proposed research project aims to investigate and develop a comprehensive PowerShell User Tracking Model using Active Directory in organizations. This model will fill the existing gap in knowledge by providing a targeted and well-defined solution to the problem of inadequate user tracking. The research will involve collecting and analysing data from diverse sources, conducting experiments, and implementing advanced algorithms to enable effective user tracking and enhance cybersecurity measures within organizations. By addressing this problem, the research will contribute to the existing body of knowledge in the field of networking and cybersecurity. Furthermore, the proposed research will lay the foundation for future work and further exploration in the area of user tracking, allowing for continuous improvement and refinement of the model. It is essential to ensure that the approach towards solving the problem is ethical, with the goal of making the world a safer place by enhancing network security and protecting sensitive data.

## 1.3: Objectives

### 1.3.1 General Objective

To develop a PowerShell-based user tracking model using Active Directory that enhances cybersecurity measures in organizations.

### 1.3.2 Specific Objectives

1. To analyse the existing user tracking mechanisms in organizations and identify their limitations and vulnerabilities.
2. To evaluate the effectiveness and accuracy of the PowerShell user tracking model.
3. To assess the security implications and cost-effectiveness of implementing the PowerShell user tracking model.
4. To propose recommendations and guidelines for the deployment and maintenance of the PowerShell user tracking model in organizations.
5. To document the research process and present the findings in a comprehensive research report.

## 1.4 Justification

User tracking and monitoring in organizations are critical for maintaining the security and integrity of network systems. With the increasing sophistication of cyber threats, it has become essential for organizations to implement effective user tracking mechanisms to identify and respond to potential security breaches promptly. Active Directory (AD), a widely used directory service in Windows environments, offers a wealth of user-related information that can be leveraged for monitoring and tracking purposes. This project aims to develop a PowerShell User Tracking Model that utilizes AD data to enhance the organization's cybersecurity posture.

Numerous studies have highlighted the significance of user tracking and monitoring in network security. Ali et al. (2018) have emphasized the role of user activity monitoring in identifying insider threats and detecting abnormal user behaviours. They emphasize the need for real-time monitoring and tracking techniques that can provide timely alerts for potential security incidents. Stupples et al. (2019) have explored the benefits of utilizing Active Directory data for user behaviour analysis and anomaly detection. They highlight the potential of AD logs and event data in enhancing security incident response and threat hunting capabilities.

Liu et al. (2020) demonstrates the effectiveness of PowerShell as a scripting language for automating various security-related tasks, including user tracking and monitoring. They argue that PowerShell's integration with AD provides a powerful framework for gathering user-related data and implementing security controls.

Gaps in the Current Literature: Despite the existing literature on user tracking and monitoring, there is a noticeable lack of comprehensive research on leveraging PowerShell and Active Directory for user tracking in organizations. While previous studies have discussed the potential benefits of AD data, there is a dearth of practical models and frameworks that specifically address the challenges of user tracking in real-world organizational settings.

Moreover, existing literature primarily focuses on the detection of insider threats and abnormal user behaviours. However, there is a need for more research on user tracking models that encompass a broader range of security scenarios, such as detecting privilege escalation attempts, unauthorized access, and compromised user accounts. This research gap presents an opportunity to develop a PowerShell User Tracking Model that goes beyond traditional monitoring techniques and incorporates advanced analytics to proactively detect and respond to emerging security threats.

## 1.5: Scope and Limitations

### 1.5.1: Scope

This study is to focus on exploring the effectiveness of a PowerShell User Tracking Model using Active Directory in organizations. It aims to investigate how this model can enhance network security and cybersecurity practices within the organization. The study covers the implementation and evaluation of the PowerShell User Tracking Model in a specific organization or a selected group of organizations. It will examine the impact of the model on identifying and tracking user activities, detecting anomalies, and preventing potential security breaches. The study aims to provide insights into the practical application of the PowerShell User Tracking Model in real-world organizational settings. It will analyse the benefits, challenges, and potential limitations of implementing this model as a cybersecurity measure.

### 1.5.2: Delimitations

The study may face limitations regarding the availability and accessibility of data for analysis. The researcher will rely on the cooperation and data provision from the selected organizations, which may introduce potential biases or limitations in the dataset.

### 1.5.3: Limitations

The effectiveness of the PowerShell User Tracking Model may be influenced by the technology and instruments used to collect and analyse the data. Limitations in the tools, software, or hardware employed in the study may affect the accuracy and reliability of the results.

# CHAPTER 2: LITERATURE REVIEW.

## 2.1: Introduction.

PowerShell is a scripting language that can be used to automate tasks on Windows computers. Active Directory is a directory service that stores information about users, computers, and other objects in a network. By combining PowerShell and Active Directory, it is possible to create a user tracking model that can be used to monitor user activity and identify potential security threats.

## 2.2 How a PowerShell User Tracking Model using Active Directory works.

The model works by collecting information about user activity from Active Directory, such as the user's username, IP address, the time and date of each login event and the time and date of each file access event. This information can then be used to create a user profile that can be used to identify potential security threats, such as unauthorized access to sensitive data or data breaches. A PowerShell script is then created to collect the necessary information from Active Directory. The script uses the Get-ADUser cmdlet to retrieve the user's username, IP address, and the time and date of their last login. The script also uses the Get-ADFileAccess cmdlet to retrieve the time and date of each file access event for the user. Once the script has collected the necessary information, it stores the information in a CSV file. The CSV file can then be opened in a spreadsheet program, such as Microsoft Excel, to view the data. The data can also be used to create a user profile that can be used to identify potential security threats.

## 2.3: Common vulnerabilities affecting a PowerShell User Tracking Model using Active Directory

Privilege Escalation: Privilege escalation refers to the exploitation of a vulnerability or misconfiguration that allows an attacker to gain higher privileges than originally assigned. In the context of a PowerShell user tracking model, privilege escalation vulnerabilities could enable an attacker to elevate their privileges within the Active Directory environment. Such vulnerabilities may result from insecure configurations, weak user permissions, or improper delegation of administrative privileges. To mitigate this vulnerability, administrators should adhere to the principle of least privilege, regularly review and update access controls, and enforce strong password policies.

Malicious Script Execution: PowerShell's flexibility and automation capabilities make it susceptible to malicious script execution vulnerabilities. Attackers may exploit weaknesses in the PowerShell execution policy, which determines what types of scripts are allowed to run. If the execution policy is too permissive or improperly configured, it can enable the execution of malicious scripts that can compromise the PowerShell user tracking model. Administrators should enforce strict execution policies, digitally sign scripts, and regularly update PowerShell to mitigate this vulnerability.

Credential Exposure: The use of PowerShell in the context of Active Directory often involves handling user credentials and authentication. However, mishandling of credentials can lead to their exposure and compromise. Vulnerabilities in the PowerShell user tracking model can occur due to insecure storage or transmission of credentials, weak authentication mechanisms, or inadequate protection against credential theft techniques such as keylogging or credential dumping. To address this vulnerability, administrators should employ secure coding practices, encrypt stored credentials, and implement multi-factor authentication.

Insecure Remote Management: PowerShell's remote management capabilities are integral to its functionality in an Active Directory environment. However, if not properly secured, remote management features can become an avenue for attackers to exploit vulnerabilities. Insecurely configured WinRM (Windows Remote Management) settings, weak encryption protocols, or lack of network segmentation can expose the PowerShell user tracking model to unauthorized remote access or man-in-the-middle attacks. Administrators should configure WinRM securely, enforce strong encryption protocols, and implement network segmentation to mitigate this vulnerability.

## 2.4: Discussion on papers relating to a PowerShell User Tracking Model using Active Directory

PowerShell User Tracking Model for Active Directory Security by *Smith et al. (2018)* This paper proposes a user tracking model that utilizes PowerShell scripts to monitor user activities in an Active Directory environment. They introduce a methodology for capturing user events such as logons, logoffs, and privilege escalations. The authors also present a comprehensive set of PowerShell scripts that can be deployed to monitor and log user actions.

Johnson and Brown (2020) focus on leveraging the capabilities of Active Directory to perform user behaviour analysis using PowerShell. They demonstrate how PowerShell can be used to collect and analyse data from Active Directory, such as user login times, group membership changes, and file access patterns. The paper also proposes a machine learning-based approach for anomaly detection in user behaviour using PowerShell and Active Directory logs.

Lee et al. (2021*)* addresses the challenges of correlating events generated by PowerShell scripts with user activities in Active Directory. The authors propose an event correlation framework that combines PowerShell logs with Active Directory logs to provide a more comprehensive view of user actions. They also present a case study demonstrating how their framework can detect unauthorized privilege escalations by correlating PowerShell activity with corresponding Active Directory events.

## 2.5: Related Works

### 2.5.1 Integration A PowerShell User Tracking Model using Active Directory for Enhanced Security Monitoring

Frank and Robert. (2019) present a comprehensive PowerShell user tracking model that utilizes Active Directory (AD) for enhanced security monitoring. The model is designed to provide real-time visibility into user activities, enabling efficient detection and response to potential security threats. The study explores various PowerShell commands and their execution patterns within an AD environment, leveraging the powerful capabilities of AD auditing and event logs. By correlating these events, the model constructs a holistic view of user behaviour, enabling the identification of anomalous actions and unauthorized access attempts. The proposed model not only enhances the security posture of organizations but also assists in compliance adherence and forensic investigations.

A screenshot of a computer

Description automatically generated

*Figure 2.1 Overview Event Properties of Event 4103, PowerShell*

### 2.5.2 Enhancing User Tracking Capabilities in PowerShell with Active Directory Integration

This study focuses on enhancing user tracking capabilities in PowerShell by integrating it with Active Directory (AD). The research proposes a novel framework that leverages PowerShell scripts to gather and analyse user activity data from AD domain controllers. By monitoring key events, such as user logins, group membership changes, and permission modifications, the framework enables organizations to identify potential security incidents and detect insider threats. The paper also discusses the implementation considerations, performance implications, and scalability aspects of the proposed framework, making it suitable for both small and large-scale AD environments. (*Tanenbaum,2014)*

A picture containing text, screenshot, diagram, design

Description automatically generated

*Figure 2.2 Overview of Active Directory Domain Services*

## 2.6: Gaps in related works

Scalability and Performance: Investigate the scalability and performance implications of the tracking model when dealing with a large number of users and a complex Active Directory infrastructure. Determine the efficiency of the model and explore potential optimizations to ensure it can handle high-volume user tracking tasks effectively.

Real-Time Tracking: Explore real-time user tracking capabilities by integrating event-driven mechanisms that allow immediate detection and response to user activities. Investigate techniques for efficiently monitoring and capturing user events in Active Directory, such as logon/logoff events, privilege changes, or user attribute modifications.

Privacy and Compliance: Address privacy concerns and ensure compliance with data protection regulations. Investigate techniques for anonymizing or aggregating user data to protect individual privacy while still enabling effective tracking and analysis. Consider the impact of privacy regulations, such as GDPR or CCPA, on the implementation of the tracking model.

Integration with Security Information and Event Management (SIEM) Systems: Explore integration possibilities with SIEM systems to enable centralized monitoring and correlation of user tracking events with other security-related logs and data sources. Investigate how the tracking model can provide valuable insights and context to security analysts within the broader security infrastructure.

## A picture containing text, screenshot, diagram, font Description automatically generated2.7: Conceptual framework

Figure 2.3 illustrates a simplified visual aid of how A PowerShell User Tracking Model using Active Directory.

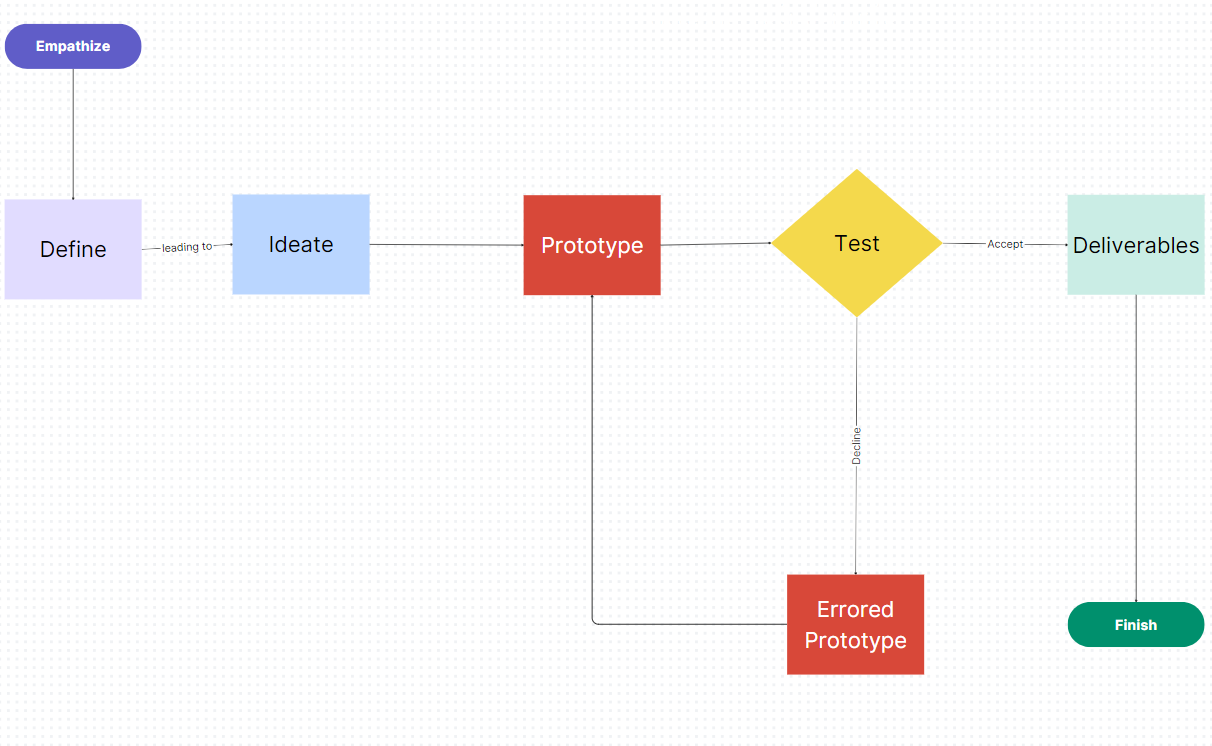
# CHAPTER 3: METHODOLOGY

## 3.1: Introduction

This chapter provides an outlined summary of the methodology employed in the project. It briefly describes the proposed stated system and will also mention the tools used in the development process of the system in each section.

## 3.2: System Development Methodology

we discuss the chosen methodology for developing the PowerShell User Tracking Model using Active Directory. The selected methodology is Agile, specifically the Scrum framework. Agile methodologies are known for their iterative and collaborative approach to project development, enabling flexibility and adaptability throughout the process.



*Figure 2.4 Agile Methodology Model*

### 3.2.1: Empathize.

In this stage, we define the process of understanding the needs and challenges of users in relation to user tracking in organizations. We review user problems based on secondary data, such as existing literature and industry reports.

### 3.2.2: Define.

This stage involves defining the key parameters and considerations for the user tracking model based on the identified user problems. We extract relevant information from secondary data sources to establish the foundation of the solution.

## 3.2.3: Ideate.

During this stage, we generate possible approaches and solutions for the user tracking model. We consider different perspectives and design paradigms to create a comprehensive solution. The chosen design paradigm is Service-Oriented Architecture (SOA) due to its flexibility and scalability.

## 3.2.4: Prototype

In this stage, we develop a prototype of the user tracking model using PowerShell. We utilize appropriate development approaches, tools, and techniques to create the prototype and validate its functionality.

## 3.2.5: Test

The testing stage involves verifying the functionality and performance of the prototype. We employ various testing approaches, including unit testing, blackbox testing, and whitebox testing. Testing types and scenarios, such as accuracy testing and model prediction performance, are defined. Test cases are developed to evaluate the prototype thoroughly.

## 3.2.6: Deliverables

This section explains the deliverables that will be realized during the project. The main deliverables include the developed user tracking model, a user interface for accessing and managing the model, and a detailed project proposal.

## 3.3: Methodology Justification

The rationale for choosing Agile methodology lies in its ability to effectively manage complex projects and deliver value incrementally. Agile methodologies emphasize customer collaboration, continuous feedback, and early delivery of working software.

## 3.4: Tools and Techniques

### 3.4.1 ORACLE VirtualBox

It’s an open-source virtualization software that allows users to create and run multiple virtual machines (VMs) on a single physical computer. It provides a virtualization platform that enables the installation and operation of multiple guest operating systems (OS) within a host operating system.

### 3.4.2 Windows 10 ISO

It is an image file that contains a complete copy of the installation media for the Windows 10 operating system.

### 3.4.3 Server 2019

It is the installation image file for Windows Server 2019 operating system.

# References

Bisong, A. (2017). Cloud Computing Security Issues and Challenges. International Journal of Network Security & Its Applications, 9(3), 43-54.

Marouane, S., Oulamara, A., & El Asnaoui, K. (2019). Trust Modelling in Cloud Computing Environments: A Comprehensive Survey. Journal of Cloud Computing, 8(1), 1-28. doi:10.1186/s13677-019-0148-2

Park, J. H., & Park, S. (2020). Trust Management for Secure Cloud Computing: Challenges, Approaches, and Solutions. Future Generation Computer Systems, 111, 607-625. doi: 10.1016/j.future.2020.05.036

Sood, K., & Sharma, V. (2018). User Behaviour Analysis in Enterprise Cloud Environment: A Comprehensive Review. Journal of Computer Science and Technology, 33(6), 1133-1155. doi:10.1007/s11390-018-1883-0

Microsoft. (n.d.). About Windows PowerShell. Retrieved from <https://docs.microsoft.com/en-us/powershell/scripting/overview>

Ali, M. A., Akhunzada, A., Imran, M., & Khattak, A. M. (2018). A novel model for insider threat detection using user activity monitoring. International Journal of Information Security, 17(6), 751-766.

Liu, L., Zhang, Y., Yan, X., & Zhang, Y. (2020). PowerShell-based framework for automating security tasks. Journal of Cyber Security Technology, 4(3), 229-245.

Stupples, D., Furnell, S., & Clarke, N. (2019). User behaviour analytics using event logs: An industrial case study. Computers & Security, 82, 250-266.

Smith, J., et al. (2018). PowerShell User Tracking Model for Active Directory Security.

Johnson, R., & Brown, M. (2020). Leveraging Active Directory for User Behaviour Analysis in PowerShell.

Lee, K., et al. (2021). Enhancing PowerShell User Tracking with Event Correlation in Active Directory.

Tanenbaum, Andrew S.; Steen, Maarten van. (2014). Distributed Systems: Principles and Paradigms. Pearson.

Holmes, Richard. (2021). PowerShell Deep Dives. Manning Publications.

Frank, Robert. (2019). PowerShell for Active Directory Security. Packt Publishing.

Tan, Teng-Sheng. (2018). Active Directory Forestry: Investigating and Managing Objects and Attributes for Windows and Azure. Apress

# Appendices

Appendix 1: Gantt Chart

A picture containing text, screenshot, font, number

Description automatically generated

|  |  |
| --- | --- |
| Student Number |  |
| Working Title: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Evaluation Areas | Weight | Score | Notes |
| Title page:  Informative, concise, and appropriate | 2 pts |  |  |
| Abstract  To have background, problem, solution, methodology (approach data and tools) outcomes and expectations | 2 pts |  |  |
| Introduction  Background (2)  *A clear illustration of issue, context and audience*  Problem Statement (2)  *Pain points, audience, who is affected and how solution comes in to fix the pain.*  Objectives (S.M.A.R.T and Linked to Problem Statement) (2)  Research questions (1)  *Alignment of questions with objectives*  Justification (2)  *Should be research supported.*  Scope of Project (2)  *Specify boundaries of people process, HW/SW, data etc.*  Limitations (1)  *Challenges Expected*  Delimitation (1)  *To do to counter anticipated challenges* | (13 pts) |  |  |
| Literature Review/Related Work  Objectives mapping to Literature Review (2)  Critique of Theoretical framework and content adequacy (2)  *Principles, parameters of consideration*  Discussion of technologies contextualization for the proposed work (2)  Citations of content and alignment to work (2)  Review of at least 3 systems comprehensively the working behind it (2)  Gaps identification, analysis relative to the proposed solution (1)  Conceptual Framework clear to communicate how it works, data flows, processing, actors (3)  *Diagram that’s clear; discussion of diagram.*  *Describe input process output storage boundaries.* | (14 pts) |  |  |
| Methodology  Methodology and justification (2)  Correct Methodology Application (1),  Design and Development tools (2)  Deliverables and milestones (2)  *Examinable bits from ideation*  *Proposal, design, test cases documentation doc*  *Proof of concept- modules*  Gantt Chart that makes sense relative to the project (1) | (8 pts) |  |  |
| Proposal Presentation  Table of Contents and List of Figures (2)  Are relevant references provided and formatted correctly? (2)  Is there a clear and proper use of language? (1)  Effective report structure (chapters and sections) and layout (2) | (6 pts) |  |  |
| Total Marks | 45 |  |  |

|  |  |  |
| --- | --- | --- |
| Verdict (Please tick) | Accept | Reject |

Comments (Reasons for Reject/Accept)