Zolder AI Agent Project Plan



Version Table

|  |  |  |
| --- | --- | --- |
| Version | Date | Note |
| 0.1 | 18/09/2024 | First Draft |
| 1.0 | 18/11/2024 | Second Draft with more detailed information |
|  |  |  |
|  |  |  |

Contents

[Introduction 4](#_Toc182839096)

[Project Definition 4](#_Toc182839097)

[Project Goal 4](#_Toc182839098)

[In-Scope: 4](#_Toc182839099)

[Out-of-Scope: 4](#_Toc182839100)

[Dependencies: 5](#_Toc182839101)

[Research Questions 5](#_Toc182839102)

[Main Question: 5](#_Toc182839103)

[Sub-Questions: 5](#_Toc182839104)

[Management & Planning 6](#_Toc182839105)

[Project Approach 6](#_Toc182839106)

[Research methodology 6](#_Toc182839107)

[Goals & intermediate results 6](#_Toc182839108)

[Milestones: 9](#_Toc182839109)

[Risk Management: 9](#_Toc182839110)

[Potential Risks: 9](#_Toc182839111)

[Mitigation Strategies: 9](#_Toc182839112)

[Project organization & communication 9](#_Toc182839113)

[Stakeholders 9](#_Toc182839114)

[Communication 10](#_Toc182839115)

[User Stories: 10](#_Toc182839116)

[User acceptance Criteria: 11](#_Toc182839117)

[User acceptance criteria: 12](#_Toc182839118)

[User acceptance criteria: 12](#_Toc182839119)

[Documentation 12](#_Toc182839120)

[Installation Process: 13](#_Toc182839121)

[Testing 13](#_Toc182839122)

[Validation: 13](#_Toc182839123)

[Cost Analysis 13](#_Toc182839124)

[Preliminary Cost Research: 13](#_Toc182839125)

# **Introduction**

Cybersecurity tools can be difficult for small and medium-sized enterprises (SMEs) to fully utilize due to their complexity. Many SMEs struggle to interpret the data produced by these tools, especially when monitoring Office 365 environments. To address this challenge, Zolder B.V. is developing an AI agent. This agent will help users understand security incidents by providing clear explanations and actionable insights.

The AI agent will interact with AtticSecurity to process incidents and respond to user queries. It will also generate queries for Microsoft Sentinel to offer further context for alerts. The goal is to create a Proof of Concept (PoC) to demonstrate how such an agent can reduce the burden on support teams and improve the security response for SMEs

# **Project Definition**

## Project Goal

The goal of this project is to develop an AI agent that simplifies the interpretation of security incidents for users of AtticSecurity.com. By enabling users to query incidents and receive clear, understandable explanations, the AI agent will reduce the complexity of managing Office 365 security. Additionally, the agent will automate the querying of Microsoft Sentinel alerts, further aiding in threat response. This project aims to create a Proof of Concept (PoC) that demonstrates how AI agents can reduce the reliance on IT support while enhancing security operations for SMEs.

## In-Scope:

* Building an AI agent capable of processing incidents from AtticSecurity and providing understandable responses to user questions.
* Enabling the agent to generate KQL queries in Microsoft Sentinel for additional alert context.
* Ensuring that the AI agent parses and presents output data from security checks in a clear, user-friendly format.
* Delivering a working PoC to demonstrate the AI agent's capabilities.
* Providing documentation and a demonstration of the AI agent’s functionality.

## Out-of-Scope:

* Developing a UI frontend.
* Fine-tuning the AI model with custom datasets.

## Dependencies:

* Python 3.10+
* Flask Framework
* Redis or MongoDB for data storage

# **Research Questions**

## Main Question:

* What is the best approach to developing an AI agent that assists users of AtticSecurity.com in monitoring and responding to Office 365 security incidents while ensuring the solution is cost-effective?

## Sub-Questions:

* What functionality should the AI agent have to effectively process and respond to security incidents from AtticSecurity?
* How accurately can the AI agent generate KQL queries in Microsoft Sentinel to provide additional context for security alerts?
* What improvements in user understanding of security outputs can the AI agent deliver for non-technical users?

|  |  |  |
| --- | --- | --- |
| Question | Methods | Deliverables |
| What functionality should the AI agent have to effectively process and respond to security incidents from AtticSecurity? | * Problem analysis
* Brainstorm
* System test
 | - A functional AI agent that processes and responds to incidents- Demo showing functionality |
| What is the most cost-effective way to develop and deploy the AI agent while maintaining performance and scalability? ￼ | * Literature study
* Available product analysis
 | - Cost report comparing development and deployment strategies- Recommendation for most cost-efficient solution |
| How accurately can the AI agent generate KQL queries in Microsoft Sentinel to provide additional context for security alerts? ￼ | * Unit test
* Design pattern research
 | - Demo of AI-generated KQL queries and executed results- Validation of query accuracy against known data |
| What improvements in user understanding of security outputs can the AI agent deliver for non-technical users? ￼ | * Prototyping
* Stakeholder analysis
* Product review
 | - User feedback report on output clarity- Demonstration of improved output presentation |

# **Management & Planning**

## Project Approach

The project will use an Agile Scrum approach to ensure flexibility and stakeholder involvement throughout the development process. Feedback from stakeholders and coaches will guide iterative improvements.

## Research methodology

The DOT Framework will be employed:

1. **What (Domains):** General research on AI in cybersecurity and specific use cases.
2. **Why (Trade-offs):** Balance between solution feasibility and stakeholder expectations.
3. **How (Strategies & Methods):** Triangulation of methods including prototyping, stakeholder feedback, and system testing.

# **Goals & intermediate results**

|  |  |  |  |
| --- | --- | --- | --- |
| Sprint | Week | Activities | Deliverables |
| Sprint 1 | Week 1 |  | * Project plan
 |
| Week 2 | * Requirements gathering with stakeholders
* Initial planning
 |
| Week 3 | * AI agent functionality planning
* Writing project plan
 |
| Sprint 2 | Week 4 | * Review with stakeholders
* Develop basic AI agent prototype
 | * Initial version of AI agent with basic incident processing
 |
| Week 5 | * Start AI agent development for incident processing
* Begin API design
 |
|

|  |
| --- |
| Week 6 |
|  |

 | * Refinement of incident processing
* Test integration with AtticSecurity platform
 |
| Sprint 3 | Week 7 | * Continue developing user-friendly incident response capabilities
* Refine output presentation
 | * Improved AI agent prototype with user-friendly incident response
 |
| Week 8 | * Collect feedback
* Begin documentation preparation
 |
| Week 9 | * Finalize core AI functionality
 |
| Sprint 4 | Week 10 | * Develop KQL generation logic
 | * AI agent with KQL generation abilities
 |
| Week 11 | * Testing of KQL query generation and execution
 |
| Week 12 | * Final system testing (incident handling, KQL execution, output presentation)
 |
| Sprint 5 | Week 13 | * Deployment of AI agent in controlled environment
* Prepare PoC demo
 | * Deployed AI agent
* PoC demo
* Complete project documentation including user guide and cost analysis
 |
| Week 14 | * Final PoC demo
 |
| Week 15 | * Complete project documentation (installation, user guide, cost analysis)
 |

## Milestones:

* AI model implementation.
* Integration testing with Azure Sentinel and AtticSecurity.
* CI/CD pipeline setup.

## Risk Management:

### Potential Risks:

* AI hallucinations during incident interpretation.
* Prompt injection vulnerabilities.

### Mitigation Strategies:

* Remove open-ended user inputs; use predefined query templates.
* Utilize a reliable AI model to minimize hallucinations.

# **Project organization & communication**

## Stakeholders

|  |  |  |
| --- | --- | --- |
| Name | E-mail | Role |
| Rik van Duijn | rik@zolder.io | Product Owner |
| Xuemei Pu | x.pu@fontys.nl | Semester Coach |
| Ahmed Awlaqi | a.awlaqi@student.fontys.nl |  |
| Mahamed Yahye | M.yahye@ student.fontys.nl |  |
| Adnan Al-Habsi | adnan.alhabsi@student.fontys.nl |  |
| Gehad Abbas | g.abass@student.fontys.nl |  |
| Dave Derosiers | d.derosiers@student.fontys.nl |  |
| Mihai Glodici | m.glodici@student.fontys.nl |  |

## **Communication**

* Weekly meetings (Wednesday) with the Coach are conducted at TQ.
* Per sprint meeting with the product owner
* Project updates on documentation and progress will also be sent to MS Teams → Files
* During the meeting, the team will provide an update on their progress. They will explain what they managed to accomplish during the previous week and will ask further questions if needed.
* If a team member happens to be absent on meeting day, he should send a message to the others, notifying them in time, so there is no confusion between them

# **User Stories:**



## User acceptance Criteria:

* The chat bot should be able to analyze the security alert and provide the right information for that specific alert
* The chat bot should response fast and immediately (within seconds)
* When a client asks for a complex attack chain, the chat bot should divide the attack into small stages, providing specific mitigation strategy for each stage



## User acceptance criteria:

* The chat bot should analyze SIEM alerts and provide a summary for all the potential security threats related to that alert, providing risk analysis that shows the probability and impact of each alert automatically
* The chat bot should be able to connect multiple security alerts from different resources like SIEM or Microsoft Sentinel to show a general overview of a potential attack that could happen over the network



## User acceptance criteria:

* The chat bot should be trained with threat intelligence datasets to understand the nature of all the possible threats
* The chat bot will make use of the threat intelligence database to provide a human readable text for each alert and how it behaves
* The customer shouldn’t switch between tools
* When there is a vulnerability or attack vector that has been identified, the chat bot should be able to provide recommendations on patching or hardening actions for it to reduce the risk associated before the treat materialize
* The chat bot should be trained with Sentinel alerts with historical incidents that happened in the past
* The chat bot should be able to determine if the alert is part of recurring attack pattern or something new

# **Documentation**

## Installation Process:

* Full installation details available on [GitLab](https://git.fhict.nl/I485215/zolderai#requirements).

# **Testing**

## Validation:

* Unittests for AI logic and functionality.
* KQL generation tested with multiple models to ensure correctness.

# **Cost Analysis**

## Preliminary Cost Research:

* Cost estimates are under development in a separate research document.