For use by the Project lecturer	Approved		Revision required	

Feedback

For use by the Project module lecturer only

To be completed by the student

PROJECT PROPOSAL 2024

Project no	TG4	Revision no	0

Title	Surname	Initials	Student no	Study leader (title, initials, surname)
MR	**BECKER**	**JM**	20426471	Mr. T. Green

Project title (the title on the project concept note)

Intelligent Web Application Firewall using Token Inspection

Language editor details	Language editor signature
Ms. L. Fernandes	
Student declaration I understand what plagiarism is and that I have to complete my project on my own.	**Study leader declaration** This is a clear and unambiguous description of what is required in this project. Approved for submission (Yes/No)
Student signature	Study leader signature and date

1. Project description

What is the problem to be solved with your project? What is your project about? What does your system have to do?

Web applications are increasingly susceptible to malicious attacks from bad actors. Web application firewalls (WAF's) are programs or devices that intercept the web traffic on their way to the web application to prevent malicious attacks from reaching the web application as opposed to traditional firewalls that only filters network traffic. Next generation WAF's, or intelligent WAF's need to be developed to combat the new innovations in malicious attacks like authentication bypass using JWT's.

The system will have to use machine learning to identify and block malicious JSON Web Tokens (JWT's) on a WAF. The system will also have to provide typical WAF functions like preventing cross-site scripting, SQL-injections and authentication bypass. According to the OWASP Top 10 2021, which is the defacto standard for what a WAF should protect against, the most important security risks to protect against are, Broken Access Control, Cryptographic Failures, Injection, Insecure Design, Security Misconfiguration, Vulnerable and Outdated Components, Identification and Authentication Failures, Software and Data Integrity Failures, Security Logging and Monitoring Failures, and Server-Side Request Forgery (SSRF) with 90% accuracy.

2. Technical challenges in this project

Describe the technical challenges that are *beyond* those encountered up to the end of third year and in other final year modules.

2.1 Primary *design* challenges

Which aspects of the design of the system do you expect to be the most challenging?

- The most challenging design component of the system is maintaining the 90% balanced filtering accuracy of existing solutions.
- To achieve the above objective, a 95%+ True Positive and a 85%+ True Negative Filtering Accuracy is needed to assure the system does not filter out regular packets with the malicious packets.
- To achieve an accurate WAF, Functions like Cross-Site Scripting & Forgery, File Inclusion, SQL Injection and JWT Authentication Bypasss need to be implemented efficiently and accurately to capture any malicious packet.
- Lastly,the WAF needs generalise well for different web applications in order to provide security to a range of applications that can be hosted on a server.

2.2 Primary *implementation* challenges

Which aspects of the implementation to you expect to be the most challenging?

- The use and integration of different Web Applications will be a major challenge to implement on limited hardware due to the processing and memory constraints.
- The use and integration of client testing software like Selenium on a embedded host device is expected to be resource heavy on an embedded device.
- The integration of the WAF on an embedded device will prove to be difficult without the use of a neural processing engine to support the CPU since it is more suitable for machine learning algorithms.
- The integrated system is expexted to drop packets if the system is overwhelmed and this needs to be mitigated by utilising FIFO queues.

3. Functional analysis

3.1 Functional description

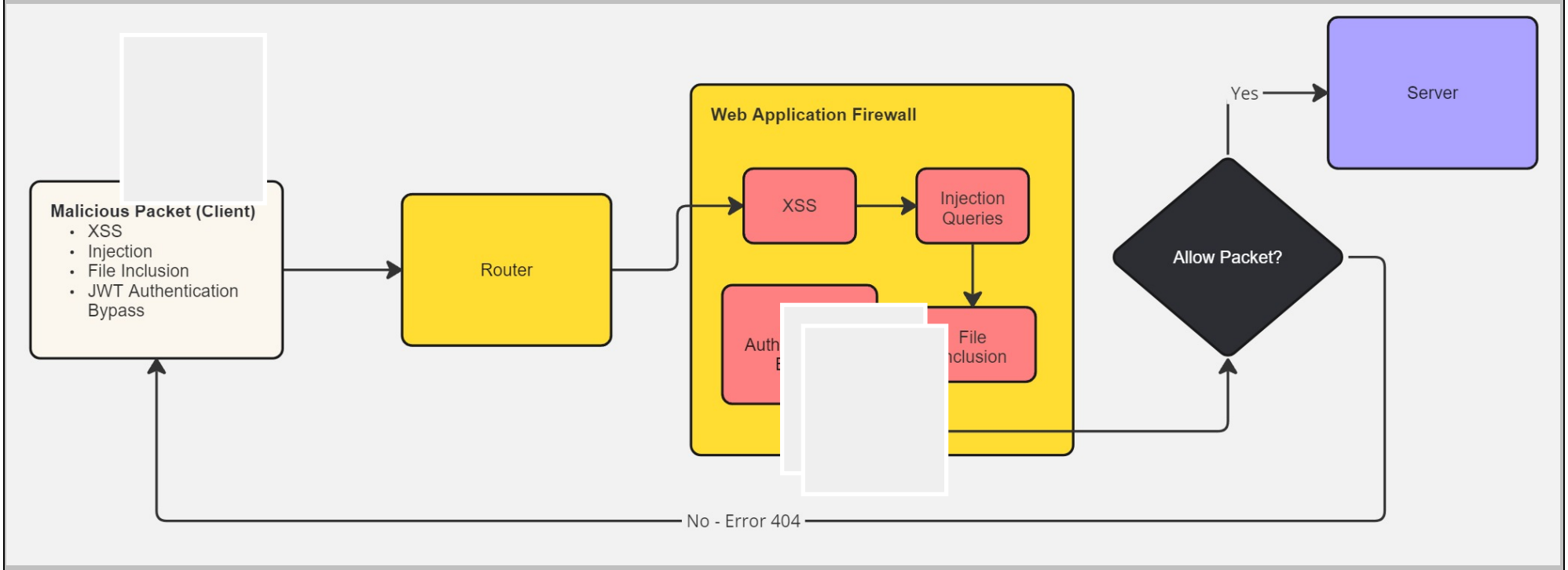
Describe the design in terms of system functions as shown on the functional block diagram in section 3.2. This description should be in *narrative format*. **DO NOT** use a bullet list.

The system has the overall goal of filtering out malicious packets from an infected host trying to cause damage to a series of vulnerable web applications. The system is designed to implement regular WAF functions like blocking Cross-site forgery & scripting, File Inclusion and SQL-injection attacks as well as next-generation functions like blocking JWT Authentication Bypass with machine learning algorithms. Its goal is to capture packets and analyse the payload in order to determine if the packet is malicious, and then either blocking or allowing it based on the rules. This keeps the vulnerable web application safe from external threats and is the best way to protect any web application today.

The system needs to generalise well on different Web Applications, thus profiling each Web Application to keep the packets of the different applications separate in order to more accurately filter the packets. The infected client, the WAF and the vulnerable Web Application are to be implemented on a single device in order to simplify the network requirements of the system since most networks already provide a level of security we want to bypass.

The system will provide the same level of security already provided by existing solutions with a 90% average, 95% true positive and 85% true negative efficacy rates. This is on par with existing solutions while providing an extra layer of security by blocking JWT Authentication Bypass, something that is not offered in existing WAF's.

3.2 Functional block diagram (this should not be a flow diagram)



System requirements and specifications

These are the core requirements of the system or product (the mission-critical requirements) in table format **IN ORDER OF IMPORTANCE**. Requirement 1 is the most fundamental requirement.

	Requirement 1: the fundamental functional and performance requirement of your project	**Requirement 2** (Number 2 in the order of importance)	**Requirement 3** (Number 3 in the order of importance)
1. Core mission requirements of the system or product. Focus on requirements that are **core** to solving the engineering problem. These will reflect the solution to the problem.	Balanced Filtering Efficacy	True Positive Filtering Efficacy	True Negative Filtering Efficacy
2. What is the target specification (in *measurable* terms) to be met in order to achieve the requirement in 1. above?	90% Efficient	95% Efficient	85% Efficient
3. Motivation: Defend the specific target specification, i.e. the value that you selected. I.e., *why* will meeting the specification given in point 2 above *solve the problem*?	Complies with industry standard	The 95th percentile is an industry standard and intercepts most malicious attacks at the application layer before travelling to the Web Application.	
4. How will you demonstrate at the examination that this requirement and specification (points 1 and 2 above) have been met? Be explicit about how you will *prove* these were met.	The client testing software will send packets to the Web Application, if the packet is returned, this is counted towards the Balanced Filtering Efficacy over the total amount of packets sent.	The client testing software will send packets to the Web Application, if the packet is returned and is malicious, this is counted towards the True Positive Filtering Efficacy over the total amount of malicious packets sent.	The client testing software will send packets to the Web Application, if the packet is returned and is not malicious, this is counted towards the True Negative Filtering Efficacy over the total amount of non-malicious packets sent.
5. Your own design contribution: what are the aspects that *you will design and implement yourself* to meet the requirement in point 2? If none, *remove this requirement.*	JWT Authentication Bypass Prevention	Cross-site forgery & scripting	File Inclusion- and SQL-injection prevention
6. What are the aspects to be taken off the shelf to meet this requirement? If none, indicate "none". Clearly specify for what tasks **library functions** will be used (if relevant to the project).	JSON Web Token Inspection	Threat Database	SQL Keyword Database

System requirements and specifications page 2

	Requirement 4	Requirement 5	Requirement 6
1. Core mission requirements of the system or product. Focus on requirements that are core to solving the engineering problem. These will reflect the solution to the problem.	Profiling	JWT Authentication Bypass	
2. What is the target specification (in measurable terms) to be met in order to achieve the requirement in 1. above?	90% Efficacy per web application	JWT Authentication Bypass	
3. Motivation: Defend the specific target specification, i.e. the value that you selected. I.e., *why* will meeting the specification given in point 2 above solve the problem?	It supports Requirement 1	To achieve Requirement 1	
4. How will you demonstrate at the examination that this requirement and specification (points 1 and 2 above) have been met? Be explicit about how you will *prove* these were met.	Measure the amount of packets returned per web application over the total amount of packets sent.	Measure the percentage of JWT Authentication Bypass attacks that are returned with a 404 error over the total amount of JWT Authentication Bypass attacks	
5. Your own design contribution: what are the aspects that *you will design and implement yourself* to meet the requirement in point 2? If none, *remove this requirement.*	Get the application to perform differently for each web application by separating information collected to train the models on.	JWT Inspection and classifier model training	
6. What are the aspects to be taken off the shelf to meet this requirement? If none, indicate "none". Clearly specify for what tasks **library functions** will be used (if relevant to the project).	The existing WAF functions like Cross-site scripting & forgery, SQL Injection and File inclusion prevention.	None	

System requirements and specifications page 3

	Requirement 7	Requirement 8	Requirement 9
1. Core mission requirements of the system or product. Focus on requirements that are core to solving the engineering problem. These will reflect the solution to the problem.			
2. What is the target specification (in measurable terms) to be met in order to achieve the requirement in 1. above?			
3. Motivation: Defend the specific target specification, i.e. the value that you selected. I.e., *why* will meeting the specification given in point 2 above solve the problem?			
4. **How will you demonstrate at the examination** that this requirement and specification (points 1 and 2 above) have been met? Be explicit about how you will *prove* these were met.			
5. Your own design contribution: what are the aspects that *you will design and implement yourself* to meet the requirement in point 2? If none, *remove this requirement.*			
6. What are the aspects to be taken off the shelf to meet this requirement? If none, indicate "none". Clearly specify for what tasks l**ibrary functions** will be used (if relevant to the project).			

System requirements and specifications page 4

	Requirement 10	Requirement 11	Requirement 12
1. Core mission requirements of the system or product. Focus on requirements that are core to solving the engineering problem. These will reflect the solution to the problem.			
2. What is the target specification (in measurable terms) to be met in order to achieve the requirement in 1. above?			
3. **Motivation:** Defend the specific target specification selected, i.e. the value. *Why* will meeting the specification given in point 2 above *solve the problem*?			
4. **How will you demonstrate at the examination** that this requirement and specification (points 1 and 2 above) have been met? Be explicit about how you will *prove* these were met.			
5. Your own design contribution: what are the aspects that *you will design and implement yourself* to meet the requirement in point 2? If none, *remove this requirement.*			
6. What are the aspects to be taken off the shelf to meet this requirement? If none, indicate "none". Explicitly indicate what tasks library functions will be used for (if relevant to the project).			

5. Field conditions

These are the REAL WORLD CONDITIONS under which your project has to work and has to be demonstrated.

	Real world field condition 1	Real world field condition 2	Real world field condition 3
Field condition requirement. In which field conditions does the system have to operate? Describe the one, two or three most important field conditions.	The system needs to process 1000 packets per second without dropping a single packet.	The system needs to operate with minimal lag and the system cannot be constrained, thus the system's resources must never be utilised more than 95%.	

6. Student tasks

6.1 Design and implementation tasks

List your primary design and implementation tasks in bullet list format (5-10 bullets). These are *not* product requirements, but *your* tasks.

- The client, vulnerable web server and WAF needs to be designed and implemented on an embedded platform.
- The system needs to be simulated on a PC using Python where hardware constraints are not a problem and then scaled to fit on the embedded platform.
- The system needs to effectively prevent SQL-Injection, Cross-Site scripting, File Inclusion and JWT authentication bypass attacks to protect the vulnerable web server.
- The system will display the result of the client tests on a webpage available on the local network.

6.2 New knowledge to be acquired

Describe what the theoretical foundation to the project is, and which new knowledge you will acquire (*beyond* that covered in any other undergraduate modules).

- The student needs to learn what JWT's are and how they are used to bypass authentication measures
- The student needs to learn what WAF's are and how security vulnerablilities in web applications are prevented.
- The student needs to learn what classifier based algorithms are and how they can be used to model user behaviour and prevent out of the ordinary behaviour.
- The students needs to learn how web application servers and their clients comminicate in order to intercept the communication and how that is exploited.
- The student needs to learn how packet filtering works and how it can be utilised in machine learning in order to predict user behaviour.
- The students needs to learn what packet inspection is and how it is utilised by WAF's to determine the intent of the package.