
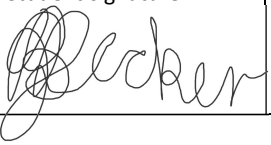


For use by the Project lecturer	Approved	Revision required
Feedback <div>For use by the Project module lecturer only</div>		

To be completed by the student						
PROJECT PROPOSAL 2024			Project no	TG4	Revision no	1
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 JSON Web Token Inspection						

Language editor details	Language editor signature
Miss L.R.S 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. <u>Approved for submission (Yes/No)</u>
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 currently increasingly susceptible to malicious attacks from threat actors. Web application firewalls (WAFs) are systems that intercept and inspect bidirectional web traffic between the web application and web clients to process and block this traffic. Current WAFs protect against vulnerabilities by inspecting session cookies or the claims of JSON web tokens (JWTs). These WAFs do not authenticate the user providing the JWT with the claims itself, merely inspecting the claim to estimate if it is a valid claim and that no harmful information is contained within. This means that current WAFs are susceptible to attacks where JWT claims could be changed to alter a different user's information. This is called JWT authentication bypass and next-generation WAFs are needed to protect against emerging threats such as these JWT authentication bypass and business logic flaws for which existing WAFs only provide limited protection.

The next-generation WAFs will have to use anomaly detection through machine learning to identify and block malicious requests by inspecting JWT claims to prevent JWT authentication bypass and business logic flaws. The system will also have to provide typical WAF features like preventing cross-site forgery **and** scripting (XSS), SQL and OS injections, and file inclusions which current WAFs have high protection rates against, but they lack advanced features like blocking JWT authentication bypass and business logic flaws.

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 first challenge is to achieve a balance between the true positive (malicious traffic) and the true negative (intended traffic) filtering accuracy.
- The second challenge is to design an accurate WAF where vulnerabilities such as cross-site scripting & forgery, file inclusion, SQL and OS injection, JWT authentication bypasses, and business logic flaws need to be prevented by efficiently and accurately implementing a next-generation WAF using JWT claims.
- Lastly, the WAF needs to build a baseline trainer per web application which utilises machine learning that can then be applied as a profile to different web applications. The baseline trainer is trained for each different web application to provide security to a range of applications.

2.2 Primary *implementation* challenges Which aspects of the implementation to you expect to be the most challenging?

- The use and integration of a range of web applications will be a major challenge to implement on limited hardware due to memory constraints.
- The use and integration of client testing software on an embedded device is expected to put strain on it.
- The integration of the WAF on an embedded device is expected to put a strain on some operating system resources.
- The integrated system is expected to drop traffic if the system is overwhelmed, and this needs to be taken into account so as not to leak memory.
- The WAF and web servers need to be connected to the same local network and the client PC needs to be connected to the same local network to access the web applications.

It is not yet necessary to indicate which communication medium is to be used between the client PC, WAF and web servers since that is a design decision. The only important aspect is that the WAF and web servers are on the same local network and that the client PC can communicate with that network.

3. Functional analysis

3.1 Functional description

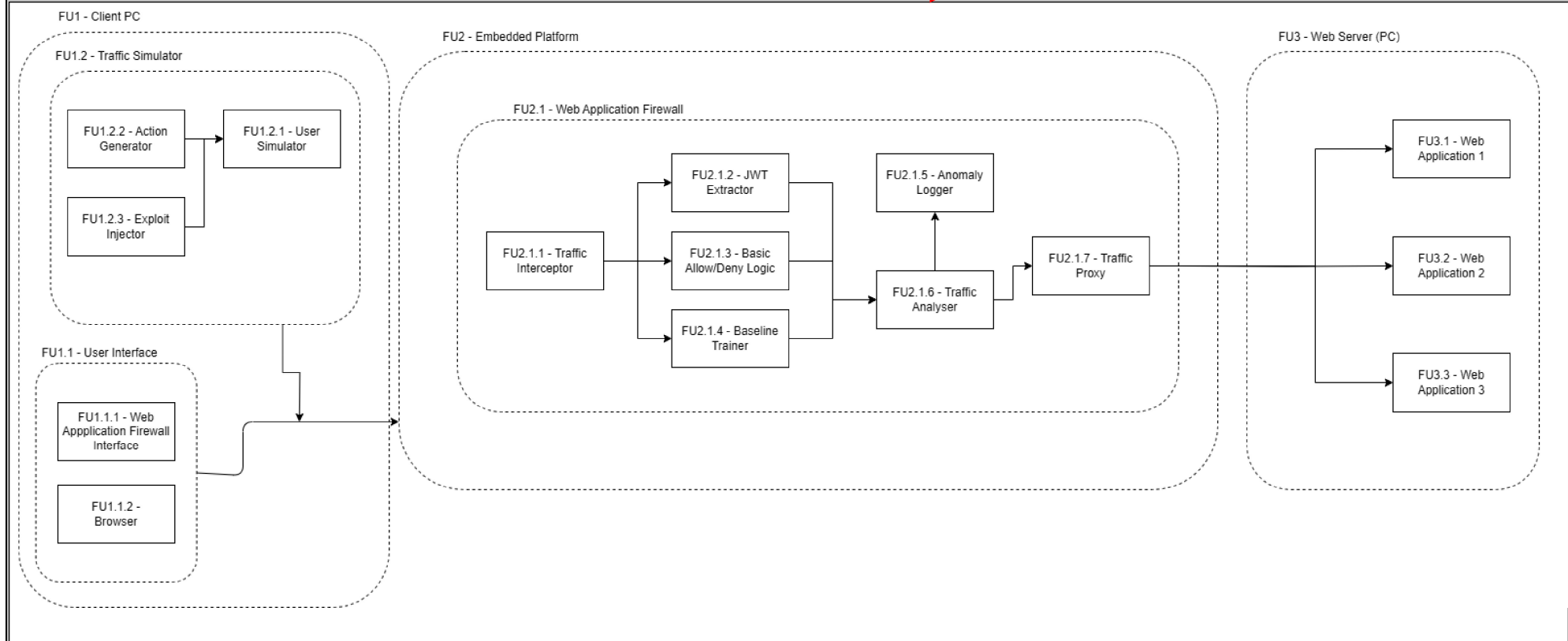
Describe the design in terms of system functions as shown on the functional block diagram in section 3.2. This description should be in *narrative format*. **DO NOT** use a bullet list.

The client interacts with the web application from the client PC (FU1) with the WAF being invisible to the user. The next-generation WAF then starts at the web client (FU1.1), where the WAF administrator interacts with the WAF management interface (FU 1.1.1) and the web browser in which the user interacts with the web applications (FU 1.1.2). The traffic simulator (FU 1.2) then monitors the traffic and uses the action generator (FU 1.2.2) to simulate a specific web request. The exploit injector (FU 1.2.3) then injects the malicious payload into the request. From here, the request is passed to a user simulator to validate it and make it appear as if it comes from the user (FU 1.2.1). The request arrives at the WAF before entering the web application, where it is intercepted (FU 2.1.1) and simultaneously passed to the JWT extractor (FU 2.1.2), which decodes the JWT and extracts the claims. The basic allow/deny logic unit (FU 2.1.3) performs some simple pre-processing on the payload, and the baseline trainer (FU 2.1.4) stores the payload/token and awaits the allow/deny result to train the machine learning model. From here, the traffic passes through the traffic analyzer (FU 2.1.5), which has all the claim information needed. The claims are processed by different functions, each of which assigns an anomaly score to the request. If the anomaly score exceeds the threat threshold, the request is denied. Otherwise, it is allowed. If the request is denied, it is logged in the threat logger (FU 2.1.6), and a **client error** is sent to the user. If it is allowed, it is passed to the traffic proxy (2.1.7), which sends the request to the correct web server (FU 3.1-3.3).

If the request is valid, the web server processes it and generates a response. This response is sent back through the WAF, where it is processed with the same scrutiny as the request.

The Traffic Simulator and the User interface was moved into one functional block called the Client PC (FU1) and the web applications was moved off of the embedded device onto the Web Server (FU3) to clearly delineate the project.

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



4. 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 <u>fundamental functional and performance requirement of your project</u>	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.	The WAF that will be created must be able to perform as well as current WAFs in preventing conventional attacks and better at preventing unconventional attacks.	The created WAF must be able to prevent conventional attacks such as client and serverside injection attacks with the same precision as existing WAFs.	The created WAF must be able to prevent unconventional attacks, such as server-side authentication bypass and business logic flaw attacks, with better precision than existing WAFs
2. What is the target specification (in <i>measurable</i> terms) to be met in order to achieve the requirement in 1. above?	The target specification is to achieve a 90% overall balanced filtering efficacy, i.e. the average of the percentage of intended traffic passed through and the percentage of malicious traffic blocked.	The percentage of conventional attacks that will be blocked is 95%.	The percentage of unconventional attacks that will be blocked is 75%.
3. Motivation: Defend the <u>specific</u> target specification, i.e. the <u>value that you selected</u> . I.e., <i>why</i> will meeting the specification given in point 2 above <i>solve the problem</i> ?	The overall balanced filtering efficacy is chosen to be 90% as this is on par with current products in the industry, and this is because if 90% of flagged requests are blocked, an attacker cannot chain together attacks.	The target specification is 95% because conventional attacks are still the most common types, representing around 75%, and therefore, the efficacy should be 5% higher than the overall balanced filtering efficacy.	The target specification is 75% as unconventional attacks are still the least common types of attacks, around 25%, and should thus be 15% lower than the overall balanced filtering efficacy.
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 <i>prove</i> these were met.	A test suite will send a malicious or valid request to the web application where it will be intercepted by the WAF. The test suite will then provide a summary of results on the efficacy of the WAF for the requests sent.	A test suite will attempt a conventional malicious attack on the web application where it will be intercepted by the WAF. The test suite then provides a summary of the results of the WAF's efficacy towards conventional attack vectors.	The target specification is 75% as unconventional attacks are still the least common types of attacks, around 25%, and should thus be 15% lower than the overall balanced filtering efficacy.
5. Your own design contribution: what are the aspects that <i>you will design and implement yourself</i> to meet the requirement in point 2? <i>If none, remove this requirement.</i>	A test suite will be designed and implemented, as well as WAF functions of which one is a baseline trainer where machine learning takes place on the embedded device.	Conventional WAF functions such as IP fencing, geo-fencing and geo-blocking, request and response inspection, security rules, anomaly scoring, will be designed and implemented on the embedded device.	Unconventional WAF functions, such as anomaly detection, will be designed and implemented on the embedded device.
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 embedded hardware platform and the operating system software will be taken off the shelf.	The aspects that will be taken off the shelf are the threat thresholds commonly used in industry.	None.

Text has been removed to explicitly mention where machine learning will take place.

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.	The system must be able to classify and allow/deny traffic with minimal latency since low latency is a critical component of web applications.	The WAF must be able to serve multiple different web applications simultaneously.	
2. What is the target specification (in measurable terms) to be met in order to achieve the requirement in 1. above?	The target specification is not to add more than a 50% delay to the original response time.	The target specification is to host 3 web applications that will be served in parallel by the WAF on one web server.	
3. Motivation: Defend the <u>specific</u> target specification, i.e. the <u>value that you selected</u> . I.e., <i>why</i> will meeting the specification given in point 2 above solve the problem?	The target specification is chosen since the web application must not be impacted when the WAF is connected in a reverse proxy mode with the web applications.	Generally, one processing core per web application is required, and since most modern microcontrollers have a quad-core processor, one core is for the WAF, and three are for the web application. This can be scaled on larger servers.	
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 <i>prove</i> these were met.	It will be demonstrated by showing the exact time that a specific requests enters the WAF versus the time the request leaves the WAF. This is then compared to the original time the request was created on the client PC.	Three different tabs on a web browser will be opened, each containing a different web application hosted on the web server with one destination address.	
5. Your own design contribution: what are the aspects that <i>you will design and implement yourself</i> to meet the requirement in point 2? <u>If none, remove this requirement.</u>	The design and implementation of the time delay measurement function. If the WAF detects that the time taken to process a request adds more delay than already exists in the system, it needs to log it in the administrator log.	The network configuration for the web applications, the WAF and the proxy server will be done from first principles including the proper routing protocols.	
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 function to retrieve the current system time will be taken off the shelf.	Existing vulnerable web applications will be taken off the shelf.	

This design contribution has been revised to indicate why this design decision is important in the context of the system and exactly what will be implemented from first principles.

This design contribution has been revised to clearly indicate what will be designed from first principles.

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 <u>specific</u> target specification, i.e. the <u>value that you selected</u> . I.e., <i>why</i> 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 <i>prove</i> these were met.			
5. Your own design contribution: what are the aspects that <i>you will design and implement yourself</i> to meet the requirement in point 2? <u>If none, remove this requirement.</u>			
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).			

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 <u>specific</u> target specification selected, i.e. the value. <i>Why</i> will meeting the specification given in point 2 above <i>solve the problem</i> ?			
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 <i>prove</i> these were met.			
5. Your own design contribution: what are the aspects that <i>you will design and implement yourself</i> to meet the requirement in point 2? If none, <i>remove this requirement</i> .			
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 must be able to operate in an enclosed and air-conditioned room at room temperature (24 degrees Celsius) to simulate data center conditions close to servers.		

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 web client, web server, and WAF need to be designed and implemented on an embedded platform.
- The system needs to be simulated on a PC where hardware constraints are not a problem and then scaled to fit on the embedded platform.
- The system needs to effectively prevent conventional attacks such as SQL-Injection, Cross-Site Scripting, File Inclusion, and unconventional attacks such as JWT authentication bypass attacks and business logic flaws to protect the vulnerable web server.

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 JWTs are and how they are used to bypass authentication measures.
- The student needs to learn what WAFs are and how security vulnerabilities in web applications are prevented.
- The student needs to learn what classifier-based algorithms are and how they can be used to model user behavior and prevent out-of-the-ordinary behavior.
- The student needs to learn how web application servers and their clients communicate to intercept the communication and how that is exploited.
- The student needs to learn how packet filtering works and how it can be utilized in machine learning to predict user behavior.
- The student needs to learn what packet inspection is and how it is utilized by WAFs to determine the intent of the packet.