29 Feb 2024 | Project Discussion

Attendees: T. Green, J.M. Becker, K.D. George, C. Ringwood, I. Sabat

Notes

- Projects will be assigned after project selection forms are completed
- Questions about the complexity and scope of the hardware projects are asked and answered, specifically what would be required on the computer vision side.
- Questions about the hardware requirements of the software tasks are asked, specifically what if any external hardware apart from the embedded platform would be used.
- Literature review due on 7 March

Action items

- ☐ Complete project selection form
- ☐ Do a literature review of the project, 4-8 papers and list 3 main tasks.

8 Mar 2024 | Project Review

Attendees: J.M. Becker, T. Green

Notes

- Look into Insecure Direct Object Referencing
- Vulnerabilities in API using JWT
- Play with OWASP BWA, DVWA
- TryHackMe has some examples
- Vulnerabilities in existing WAF
- Selenium for testing
- Focus more on the entire WAF than just the next generation features

Action items

- ☐ Get a vulnerable web application running and try to hack it.

20 Mar 2024 | Project Catchup

Attendees:

Notes

- OWASP Top 10 is not enough.
- Accuracy is the main requirement, not the functions of the WAF.
- Split the functions in existing WAF's from the next generation functions
- Use multiple vulnerable servers so that the model can generalise well, thus reducing the complexity of the WAF functions.
- Look into both client and server side attacks
- Use a man-in-the-middle reverse proxy
- Do more hacking of vulnerable websites
- Use one host for both the clients, WAF and servers
- Look into existing Open Source solutions
- First draft of Project Proposal due on 29 March
- Focus less on the testing setup and more on the WAF.

Action items

- ☐ Look into the CWE/SANS top 25 security vulnerabilities
- ☐ Get the Average, True Positive and True Negative accuracy rate needed.
- ☐ Do more research on the scalability of WAF's, specifically Load Balancing.
- ☐ Categorise the difference between the different attacks of the OWASP Top 10 and the CWE/SANS Top 25.
- ☐ Use Burp Suite as the MITM Reverse Proxy
- ☐ Do TryHackMe Tutorials on vulnerable websites.
- ☐ Look at the IronBee and F5 WAF codebase
- ☐ Finish the first draft of the Project Proposal

19 Apr 2024 | Project Proposal Revision

Attendees:
Mr. J.M. Becker
Mr. T. Green

Discussion points

- Are both session cookies and JWT's used or just JWT's
- How are the WAF functions integrated with the load balancer?
- The whole functional block diagram. How specific do I need to be with the functions of the WAF and do I need to list all the web applications?
- 95% efficacy - cutoff for null hypothesis or because its a string of attacks blocked?
- Also how accurate does it need to be in terms of True Positive/True negative efficacy?

Notes

- Don't have web applications
- Don't have users on a web application
- Not doing load balancing
- One WAF -> one Web Application
- FU4.x - Web Servers
- FU3 - WAF
    - FU 2.1 Traffic Interceptor
        - FU 2.2.1 Basic Allow List
        - FU 2.2.2 JWT Extractor
    - FU 2.3 Traffic Analyzer
    - FU 2.4 Anomaly Logger
    - FU 2.4 Traffic Proxy
    - FU 2.4 Baseline Trainer
- FU1 - UI
    - FU 3.1 - WAF Interface
    - FU 3.2 - Browser
- FU2 - Traffic Simulator
    - FU 4.1 Web App Simulator
    - FU 4.2 Action Generator
    - FU 4.3 Exploit Injector
- 3/4 Web Apps
- Conventional vs unconventional
- Time delay
- JWT's only, not session cookies

Action items

- ☐ Next Revision due Monday 22 April

25 Apr 2024 | Project Proposal Revision

Attendees:
Mr. J.M. Becker
Mr T. Green

Discussion points

- Project Proposal
- Final Submission approval
- Language editing before/after approval

Notes

- Functional Description
    - Response ‘Same scrutiny’ as request
- Functional Block Diagram
    - Remove Power supply
- System requirements and specifications
    - Use third person instead of first person
    - Requirement 1
        - WAF that will be created must be able to perform as good as current WAF’s and better at preventing unconventional attacks.
    - Requirement 2
        - The created WAF must be able to prevent conventional attacks with the same precision as existing attacks
        - The % of converntional attack vectors blocked while still allowing intended traffic.
        - 95% efficacy to block conventional attacks
        - Define conventional attacks, such as client and server side injection attacks.
    - Requirement 3
        - 75% efficacy to block conventional attacks
        - Define unconventional attacks, such as server side authentication bypass and business logic flaws attacks.
    - Requirement 4
        - The system must be able in classifing and allowing the traffic the minimum latency.
    - Requirement 5
        - The WAF must be able to serve multiple different web applications at the same time

Action items

- ☐ Final Submission approval Friday 25 April COB
- ☐ Language editing **before**/after approval

7 May 2024 | Project Catchup

Attendees:
Mr. J.M. Becker
Mr T. Green

Discussion points

- Embedded device platform choice

Notes

- Plan more for embedded device setup and debugging, especially libraries.
- Split implementation between conventional and non-conventional
- Use Gitlab integration

Action items

- ☐ Use milestones on GitLab for schedule, attach issues for smaller units
- ☐ Get libraries setup on embedded devices
- ☐ Split the implementation between conventional and unconventional
- ☐ Read up on Juggling with tokens

22 May 2024 | Project Weekly Catchup

Attendees: Koot Becker, Mr T. Green

Discussion points

- What I did last week
    - Hardware
        - Odroid M1 for server and WAF
        - Odroid M1S for interface and injection
    - Web Servers - No VM's
    - Started on WSTG
- What I am doing this week
    - Finish WSTG
    - Traffic capture on Python
    - JWT decoding
- Blockers
    - Other modules

Notes

-

Action items

- ☐ Split library model between conventional and unconventional
    - ☐ Conventional deadline 7 June
    - ☐ Unconventional deadline 14 June

3 Jun 2024 | Project Demos

Attendees: Koot Becker, Mr. T. Green

Notes

- Focus on active listening, not passive listening.
- Get a proxy server running first.

Action items

- [ ] Setup a proxy server with python.
- [ ] Dockerise web servers and run locally on PC.