Smart Home Security System:

A Design Document

Table of Contents

1. Introduction

This report outlines a development model of the Smart Home security system detailed in Kodali et al. (2016). The system requirements, behaviour, and structure of the package are provided along with an attack tree and CVSS calculations to provide a quantified vulnerability assessment and mitigations. The model is meant to provide a comprehensive schematic for subsequent application development.

2. Smart Home System

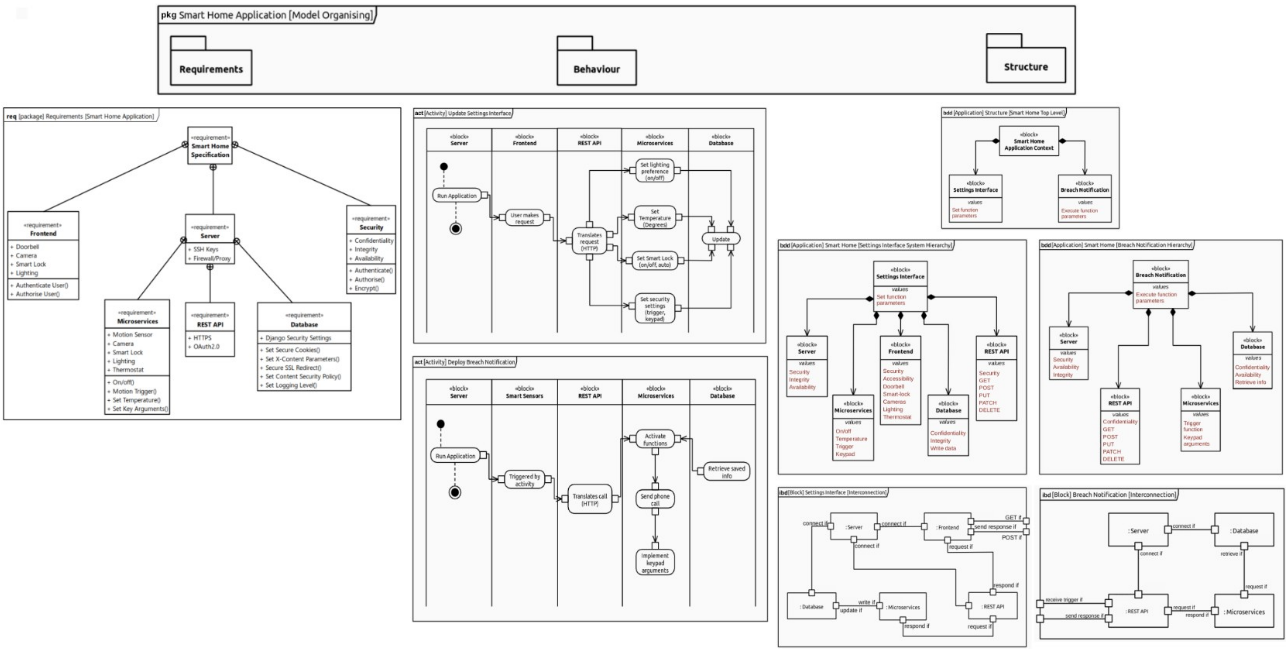

Figure 1: Package Diagram of Full Model Organisation

Kodali et al.'s (2016) Smart Home security system proposes a "Wire Home security and Home automation [...] system [which] sends alerts to the [home] owner over voice calls using the Internet" (Kodali et al., 2016: 1286). S/he can then use pre-programmed keypad arguments to control linked devices and/or alert police of a breach.

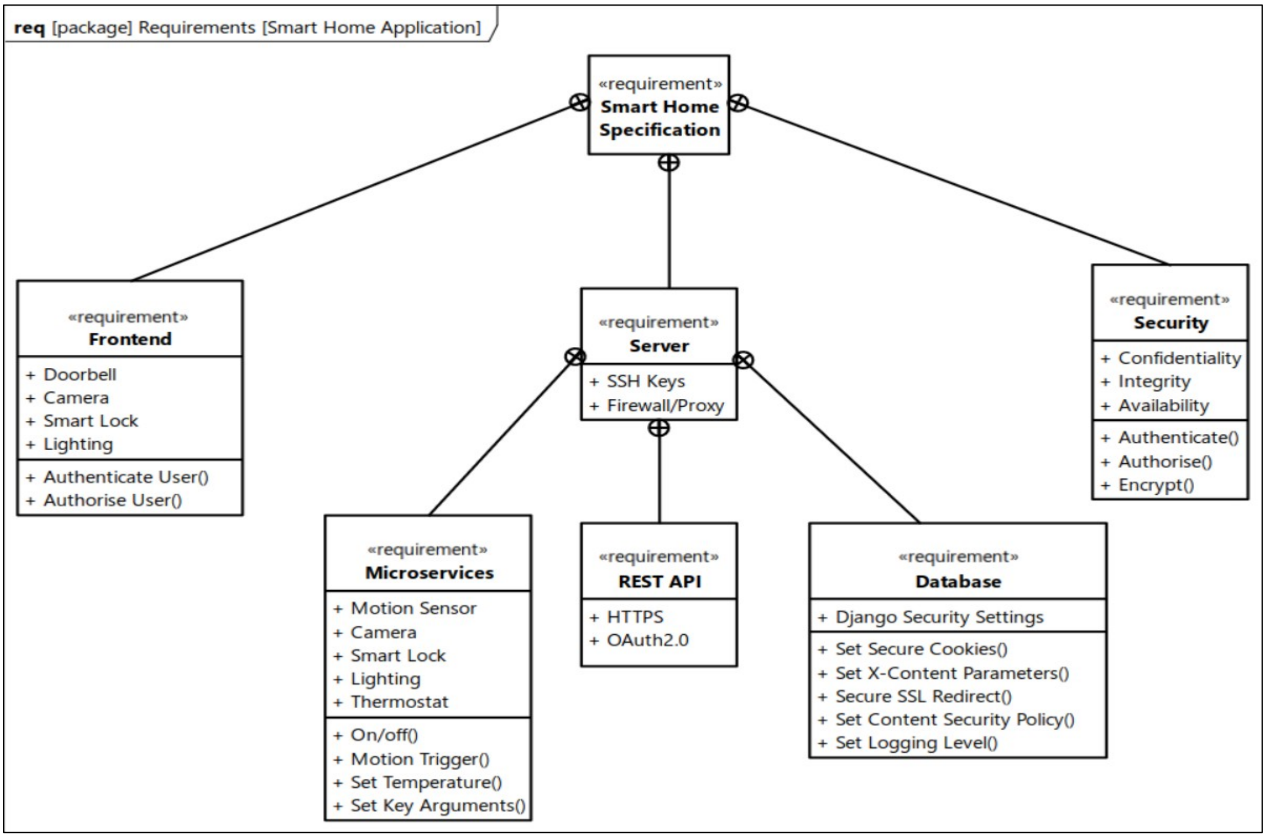

Figure 2: Package Requirements

SysML-Lite as described by Friedenthal et al. (2015) has been utilized to provide a development model of this system. This lightweight version of SysML was chosen due to the limited scope of the project, as can be seen in *Figure 1.*

The system has physical, front-end, and back-end requirements (Figure 2). Microservices are used both to connect to physical sensors and to provide behaviour arguments. The system structure of these requirements can be seen at Top-Level (Figure 3), Setting Interface (Figures 4 & 5) and Breach Notification (Figures 6 & 7) abstractions. It should be noted that *Ibd* diagrams' proxy port behaviour is marked at the beginning (closest) node.

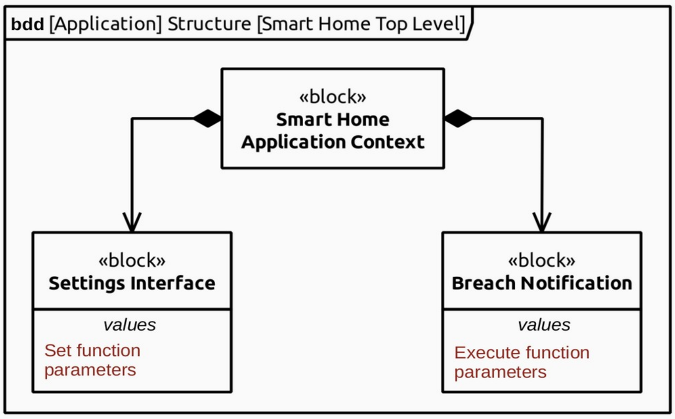

Figure 3: Application Structure: Top-Level

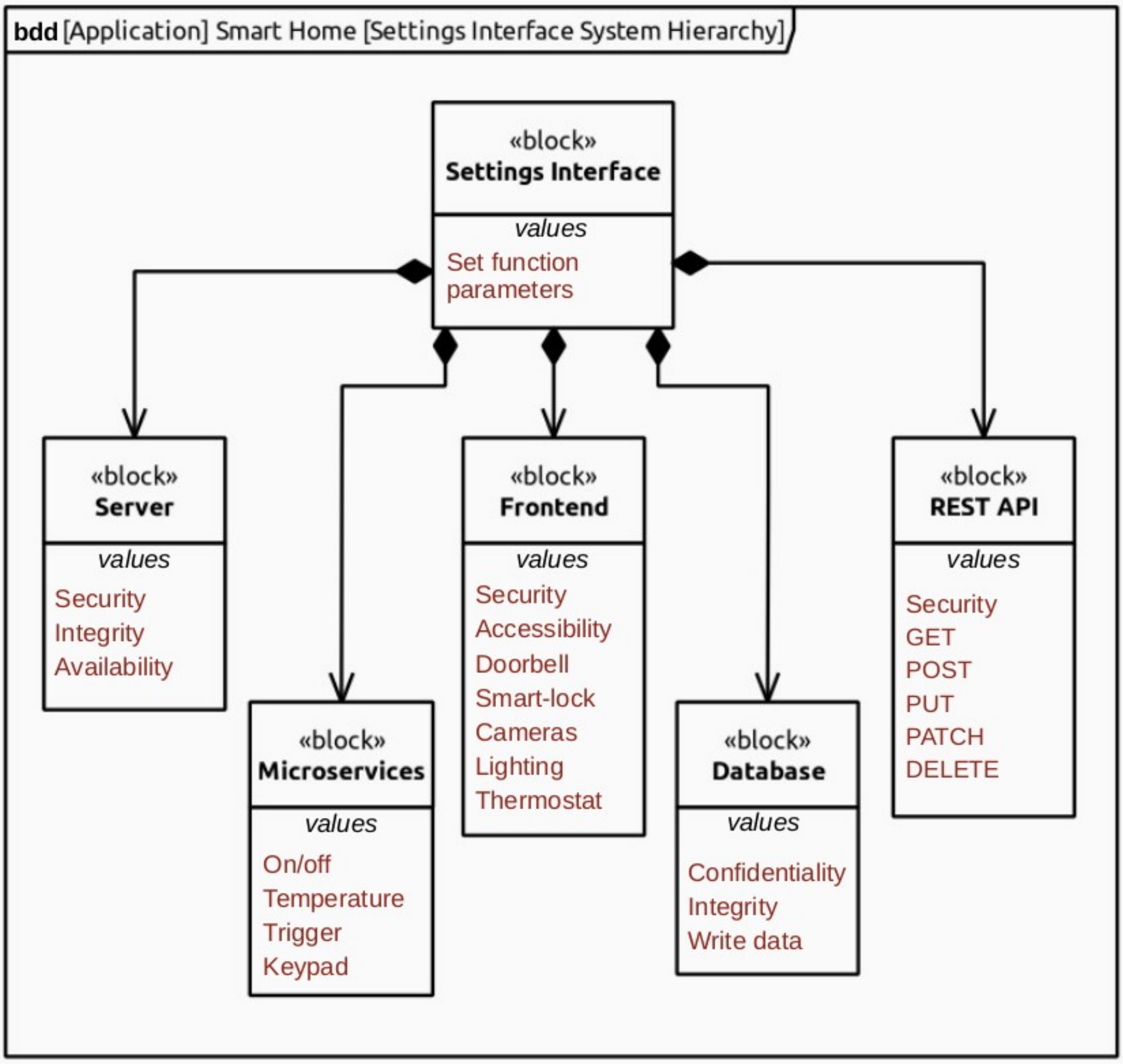

Figure 4: Settings Interface System Hierarchy

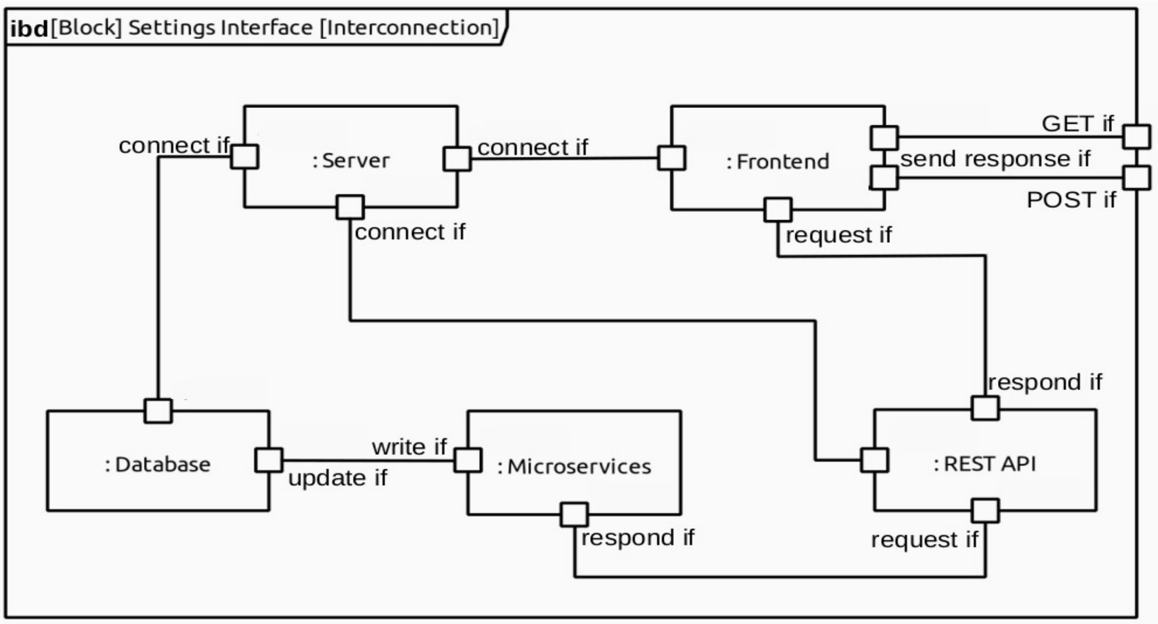

Figure 5: Settings Interface Interconnection

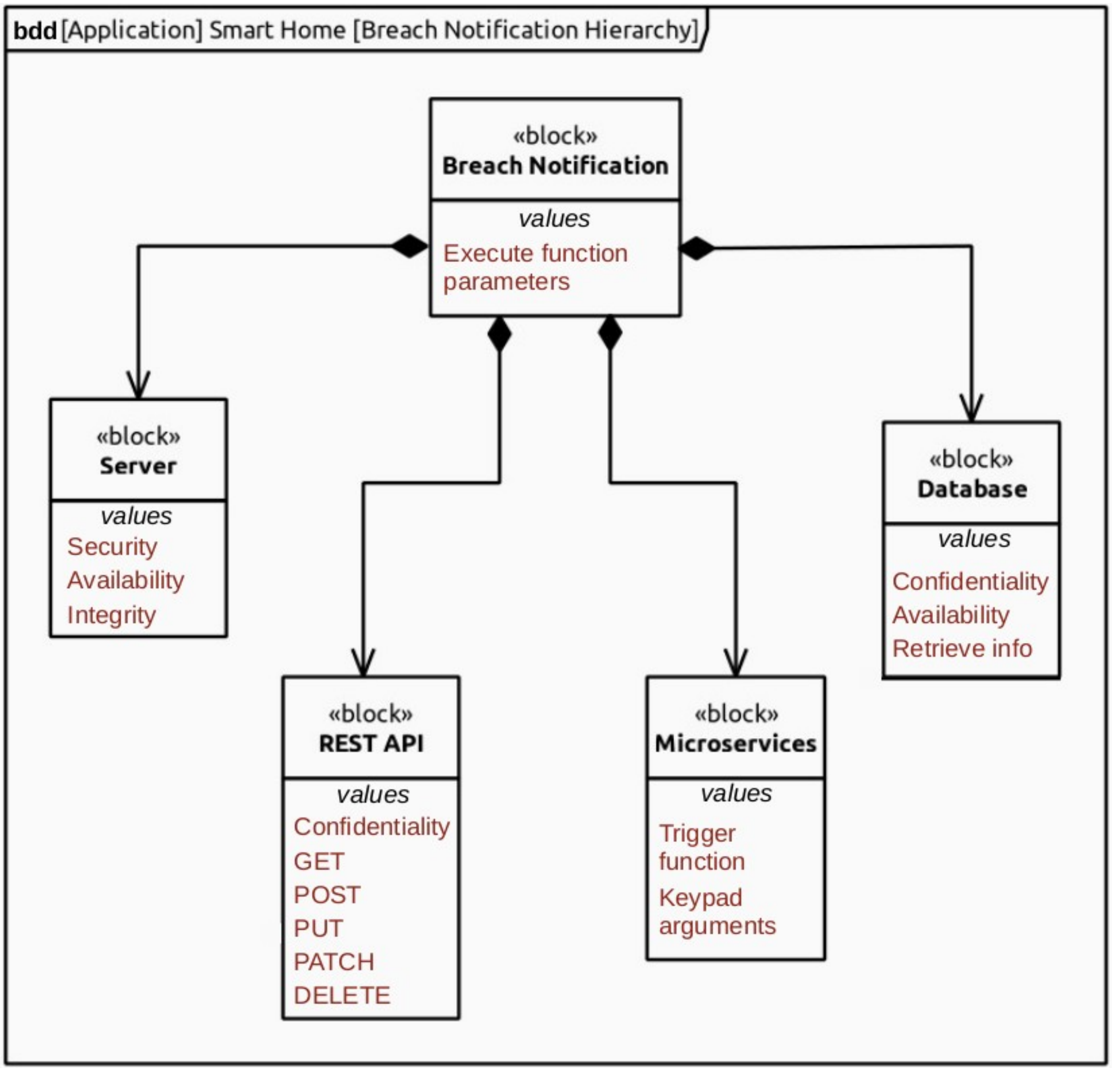

Figure 6: Breach Notification System Hierarchy

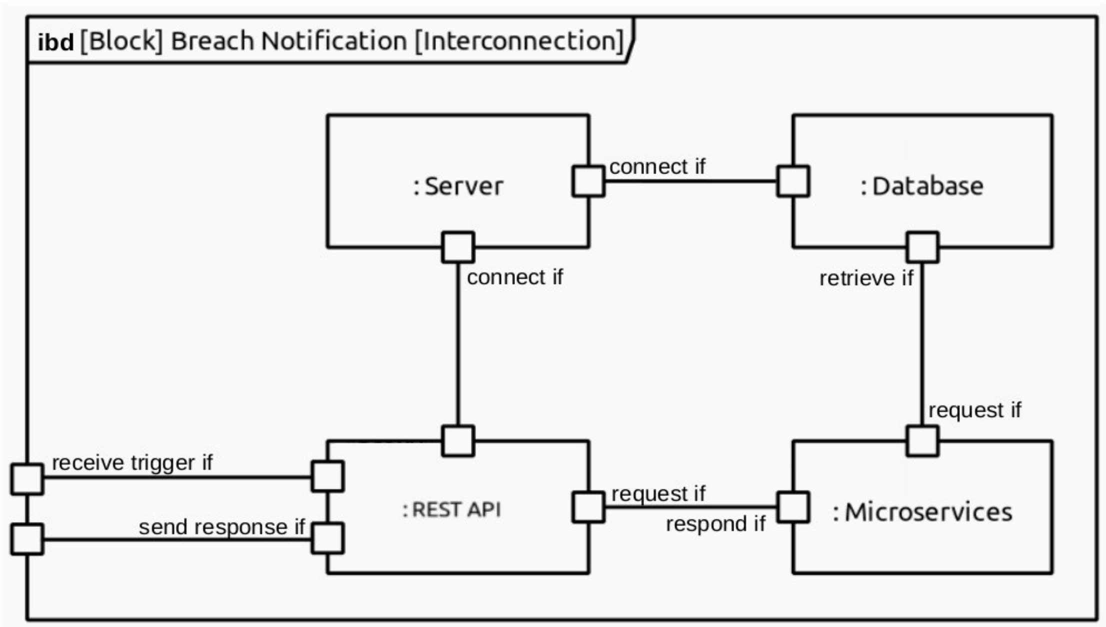

Figure 7: Breach Notification Interconnection

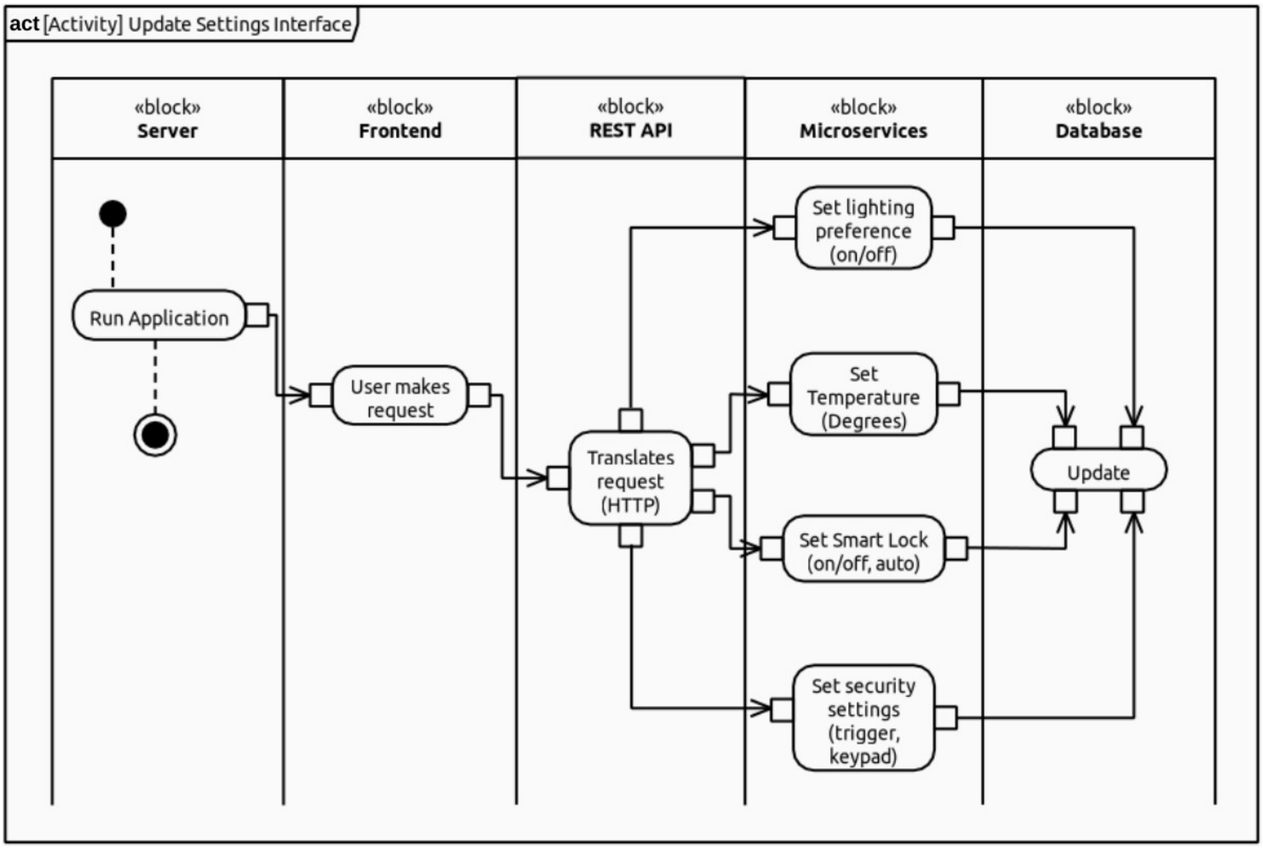

Figure 8: Update Settings Interface

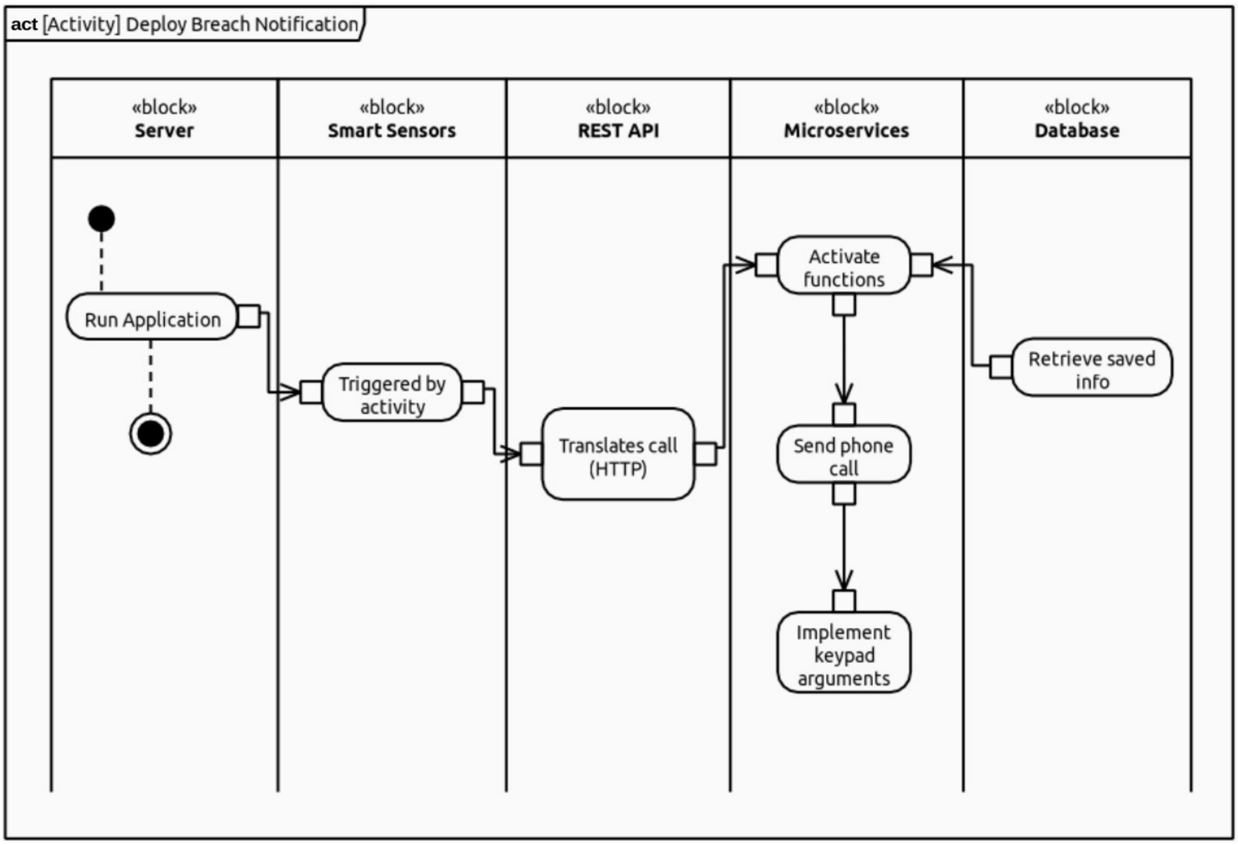

Figure 9: Deploy Breach Notification

These structures should produce two main activities: updating the Settings Interface to reflect owner preferences (Figure 8) and deploying a breach notification (Figure 9) to the owner's mobile phone if the appropriate smart sensors are triggered.

3. Vulnerabilities

Table 1: CVSS Rating Severity

Rating	CVSS Score
None	0.0
Low	0.1 – 3.9
Medium	4.0 – 6.9
High	7.0 – 8.9
Critical	9.0 – 10.0

IoT Smart Home systems are subject to various critical vulnerabilities. In an effort to quantify the severity of these vulnerabilities, FIRST(n.d.) has developed the Common Vulnerability Scoring System (CVSS) (Table 1). This system derives a "Base equation […] from two sub-equations: the Exploitability sub-score equation and the Impact sub-score equation." (FIRST, n.d: 4) which produce an overall severity rating (see Appendix I). *Table 2* outlines the most relevant IoT vulnerabilities to the Smart System, along with their severity rating and possible attacks.

Table 2: IoT Vulnerabilities

Vulnerability	**Severity**	**Possible Attacks**	**References**
Lack of MFA	8.1	• Brute-force • Flooding • Man-in-the-Middle	Hui et al., 2020; Gamundami et al., 2018; Mitre, 2018a
Lack of Role-Based Access Control	9.1	• Privilege escalation • Reflection • Cryptographic	Thilakaranthne & Wickramaaarachchi, 2018; Mitre 2018b; Mitre 2018c

Insecure Communications Protocol Usage	7.5	• Network sniffing • Injections • Forgery/tampering	Barcena & Wueest, 2015
Lack of End-to-End Encryption	9.8	• Data disclosure • Credential disclosure	Berlove, 2023; Gamundami et al., 2018
Attacker Physical Access to Devices	6.8	• Authentication defeat • Rogue Integration procedures • Cache data disclosure	Allsopp, 2009; Mitre, 2018d; Mitre, 2018e; Schneier, 2021
Exploitation of Out-Dated Firmware	9.8	• Remote code execution • Cross-site request forgery • SQL injection • Cross-site scripting	Ge et al., 2022
Exploitation of Downloaded Malware	5.4	• Replace file extension handlers • Install Rootkit • Modify existing service	Dou et al., 2020; Lakshmi & Mathane, 2021; Mitre, 2018f; Mitre, 2018g; Mitre 2018h
Data Collection without Consent	n/a	• Web scraping • Network topology mapping • Documentation disclosure	Amale,2021; Ball, 2022; Mitre 2018i

Rating Average	7.063
Standard Deviation	3.232
p-Value	0.014

A one-sample, one-tailed t-Test was performed against a hypothesized mean of 3.9, as this is the highest low-severity score possible within the CVSS framework. The t-Test type was chosen because of the small sample size and because our hypothesis contended that our mean rating average would be higher than our hypothesised mean (Berenson et al, 2015). The rating average suggests that these vulnerabilities present a high risk as a unit (see Appendix II for full statistical results). This is confirmed by the p-Value, which finds the probability that the sample mean and hypothesised mean are equal to be 1.4%.

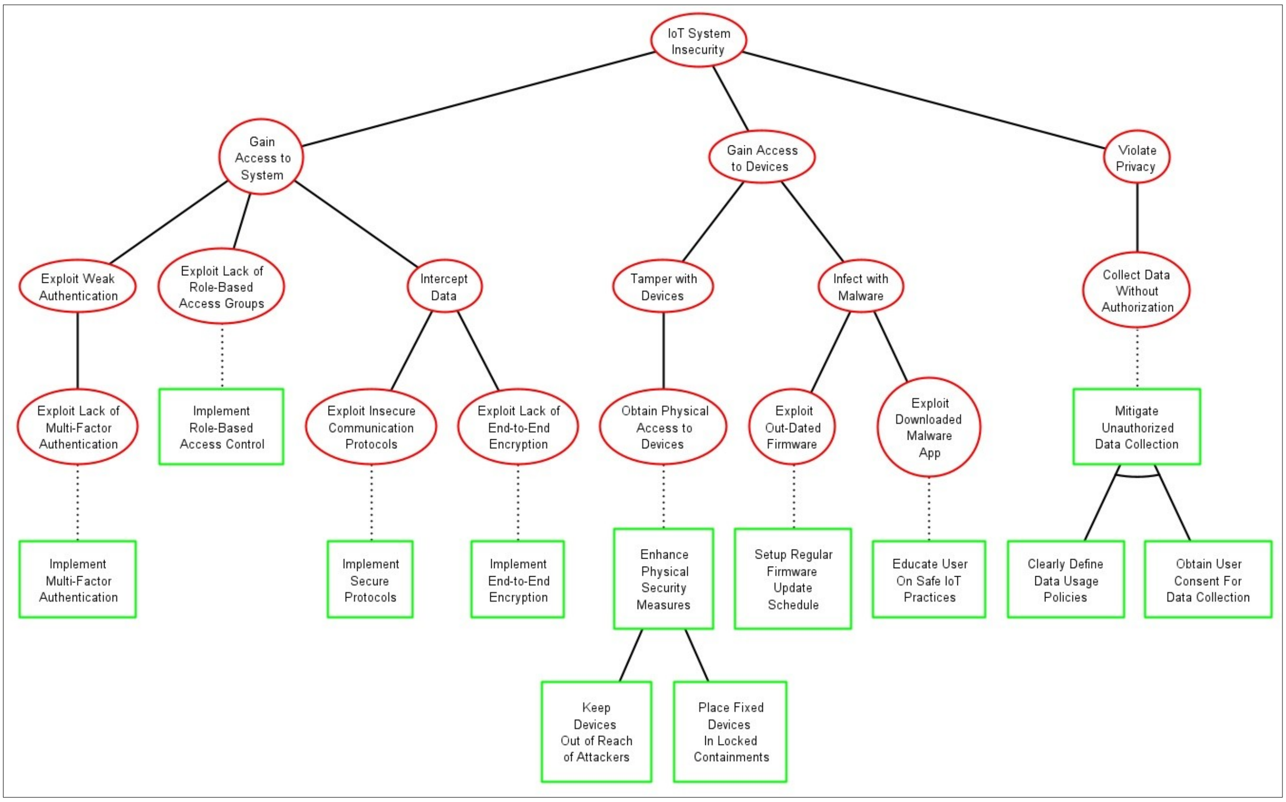

Figure 10: Attack Tree with Mitigations

The attack tree (Kordy & Schweitzer, 2015) in *Figure 10* illustrates the relationship between these vulnerabilities and attacks. Mitigation implementation is labeled in the Requirements Diagram (Figure 2) in *Section* 2, and encompasses the API (Siriwardena, 2020), server (La Lau, 2021), database (Django, 2023), and front-end (Django, 2023) level for comprehensive security.

4. End Summary

In this report, a development model detailing Kodali et al.'s (2016) Smart Home security system has outlined the requirements, behaviour, and structure of the system. Key system vulnerabilities have been discussed, along with coinciding CVSS ratings, attacks, and mitigations. The proposed model should provide a comprehensive schematic for subsequent application development.

5. References

Kordy, P. & Schweitzer, P. (2015) The ADTool Manual. ADTool. [Available Online] https://satoss.uni.lu/members/piotr/adtool/manual.pdf

Amale et al. (2021) SpyDark: Surface and Dark Web Crawler. 2021 Second International conference on Secure Cyber Computer and Communication. IEEE: 45 – 49

Ball, C. J. (2022) Hacking APIs: Breaking Web Application Programming Interfaces. San Francisco, CA, USA. No Starch Press.

Barcena, M. B. & Wueest, C. (2015) Security Response: Insecurity in the Internet of Things. Symantec: 1 -19.

Berenson, L., Levine, D, & Szabat, K. (2015) Basic Business Statistics: Concepts and Applications. 13th Ed. Harlow, UK: Pearson.

Berlove, O. (2023) *What is End-to-End Encryption and How Does it Work?* | Blog. PreVeil. [Available Online] https://www.preveil.com/blog/end-to-end-encryption/

Django (2023) *Security in Django* | Documentation. Django. [Available Online] https://docs.djangoproject.com/en/4.2/topics/security/

Dou et al. (2020) IoTMal: Towards a Hybrid IoT Honeypot for Capturing and Analyzing Malware. In: *2020 IEEE International Conference on Communications,* Dublin, Ireland. IEEE: 1 – 7

FIRST (n.d.) Common Vulnerability Scoring System version 3.1: Example, Revision 2. FIRST. [Available Online] https://www.first.org/cvss/v3-1/cvss-v31-examples_r2.pdf

Friedenthal, S., Moore, A., & Steiner, R. (2015) *A Practical Guide to SysML: The Systems Modeling Language*. 3rd ed. Waltham, MA, USA: Morgan Kaufmann

Gamundani, A. M., Philips, A., & Muyingi, H. N. (2018) An Overview of Potential Threats and Attacks on Internet of Things (IoT): A Focus on Smart Home Applications. In: *2018 IEEE Conference on Internet of Things, Green Computing and Communications, Cyber, Physical, and Social Computing, Smart, Data, Blockchain, Computer and Information Technology, Congress of Cybermatics.* IEEE: 1 – 8

Ge et al. (2022) Understanding Security Risks of Embedded Devices Through Fine-Grained Firmware Fingerprinting. *IEEE Transactions on Dependable and Secure Computing*, 19 (6): 4099 – 4111

Hui, K. L., Hong, C., Normurodov, O., & Sain, M. (2020) A Survey on the Security in Cyber Physical System with Multi-Factor Authentication. ICACT Transactions on Advanced Communications Technology, 9 (6): 1322 – 1329

Kodali, R.K., Jain, V., Bose, S. & Boppana, L. (2016) IoT Based Smart Security and Home Automation System. In: *International Conference on Computing, Communication and Automation (ICCCA)*: 1286-1289

La Lau, R. (2021) *Practical Internet Server Configuration.* New York, NY, USA. Apress.

Lakshmi, P. V. & Mathane, V. (2021) Predictive Analysis of Ransomware Attacks Using Context-AWARE AI in IoT Systems. *International Journal of Advanced Computer Science and Applications,* 12 (4): 1 – 5

MITRE (2018a) *CAPEC-125: Flooding* | Common Attack Pattern and Enumeration and Classification. capec.mitre.org. [Available Online]: https://capec.mitre.org/data/definitions/125.html

MITRE (2018b) *CAPEC-233: Privilege Escalation* | Common Attack Pattern and Enumeration and Classification. capec.mitre.org. [Available Online]: https://capec.mitre.org/data/definitions/233.html

MITRE (2018c) *CAPEC-90: Reflection Attack in Authentication Protocol* | Common Attack Pattern and Enumeration and Classification. capec.mitre.org. [Available Online]: https://capec.mitre.org/data/definitions/90.html

MITRE (2018d) *CAPEC-524: Rogue Integration Procedures* | Common Attack Pattern and Enumeration and Classification. capec.mitre.org. [Available Online]: https://capec.mitre.org/data/definitions/524.html

MITRE (2018e) *CAPEC-204: Lifting Sensitive Data Embedded in Cache* | Common Attack Pattern and Enumeration and Classification. capec.mitre.org. [Available Online]: https://capec.mitre.org/data/definitions/204.html

MITRE (2018f) *CAPEC-556: Replace File Extension Handlers* | Common Attack Pattern and Enumeration and Classification. capec.mitre.org. [Available Online]: https://capec.mitre.org/data/definitions/556.html

MITRE (2018g) *CAPEC-552: Install Rootkit* | Common Attack Pattern and Enumeration and Classification. capec.mitre.org. [Available Online]: https://capec.mitre.org/data/definitions/552.html

MITRE (2018h) *CAPEC-551: Modify Existing Service* | Common Attack Pattern and Enumeration and Classification. capec.mitre.org. [Available Online]: https://capec.mitre.org/data/definitions/551.html

MITRE (2018i) *CAPEC-309: Network Topology Mapping* | Common Attack Pattern and Enumeration and Classification. capec.mitre.org. [Available Online]: https://capec.mitre.org/data/definitions/309.html

Schneier, B. (2021) *Secrets and Lies: Digital Security in a Networked World*. Indianapolis, USA: Wiley.

Siriwardena, P. (2020) *Advanced API Security: OAuth 2.0 and Beyond.* 2nd Ed. New York, NY, USA. Apress.

Thilakarathne, N. N. & Wickramaaarachchi, D. (2018) Improving Hierarchical Role Based Access Control Model for Cloud. In: *International Research Conference on Smart Computing and Systems Engineering – 2018: 1 – 5*

6. Appendices

6.1 Appendix I

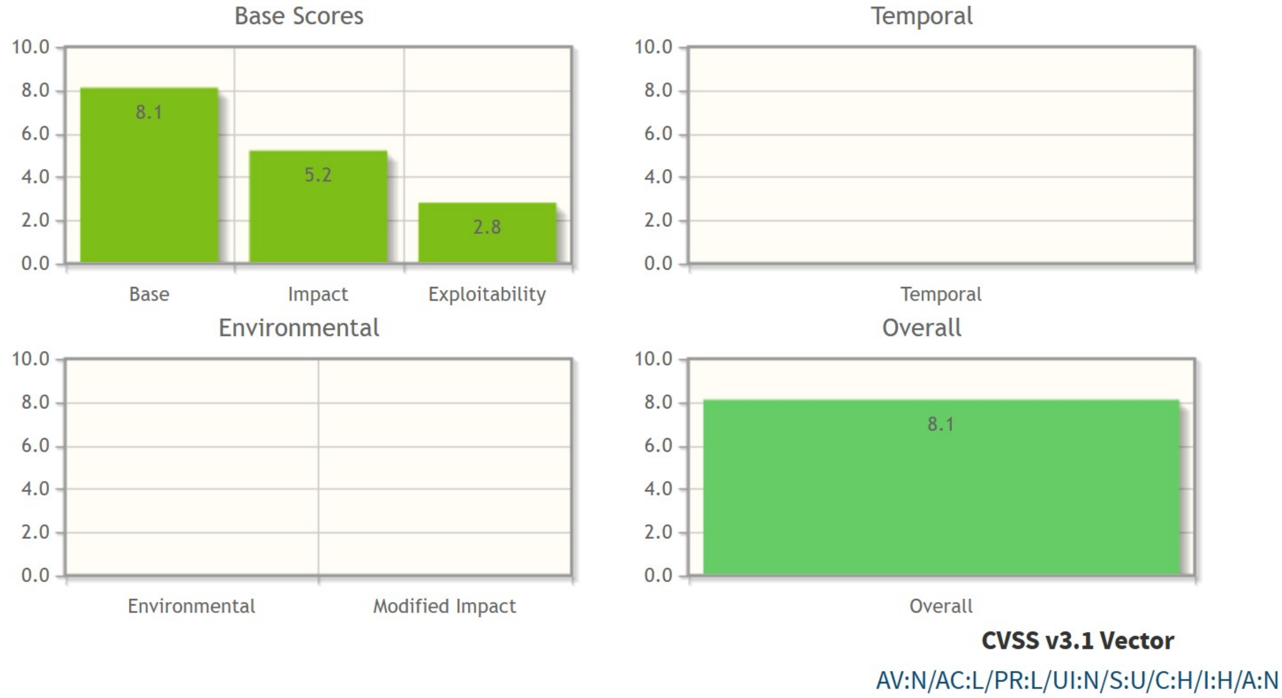

Figure 11: Lack of multi-factor authentication

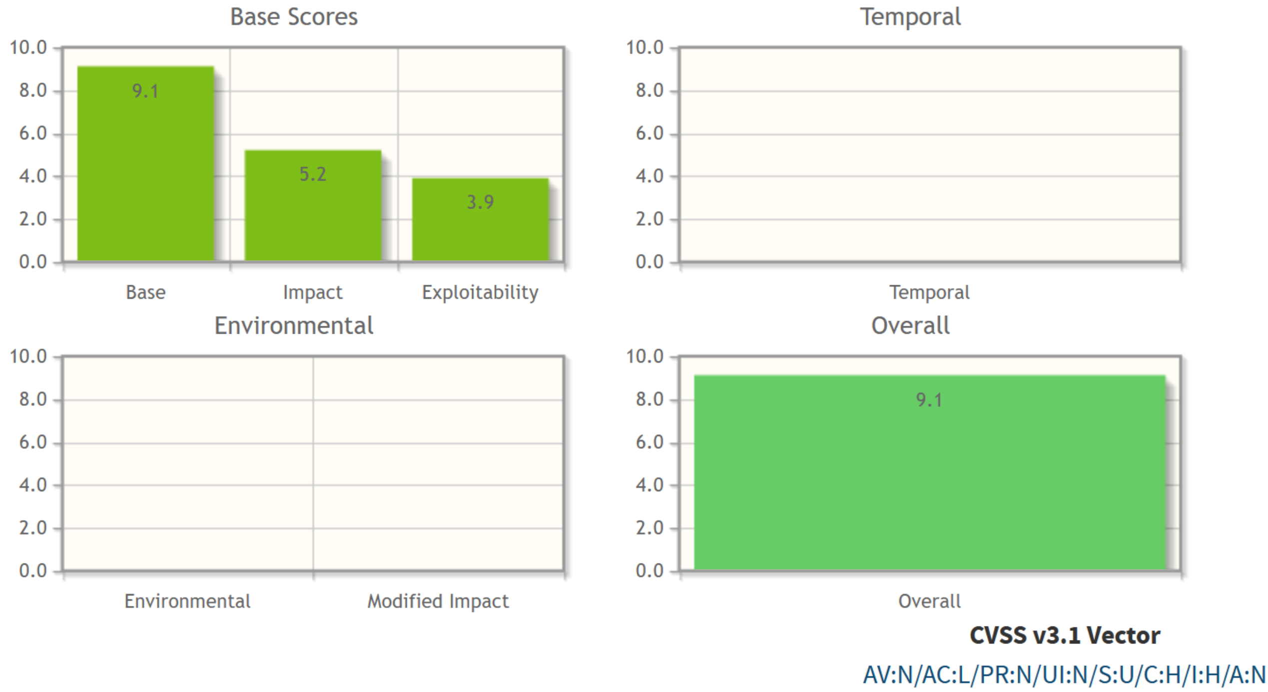

Figure 12: Lack of Role-Based Access Control

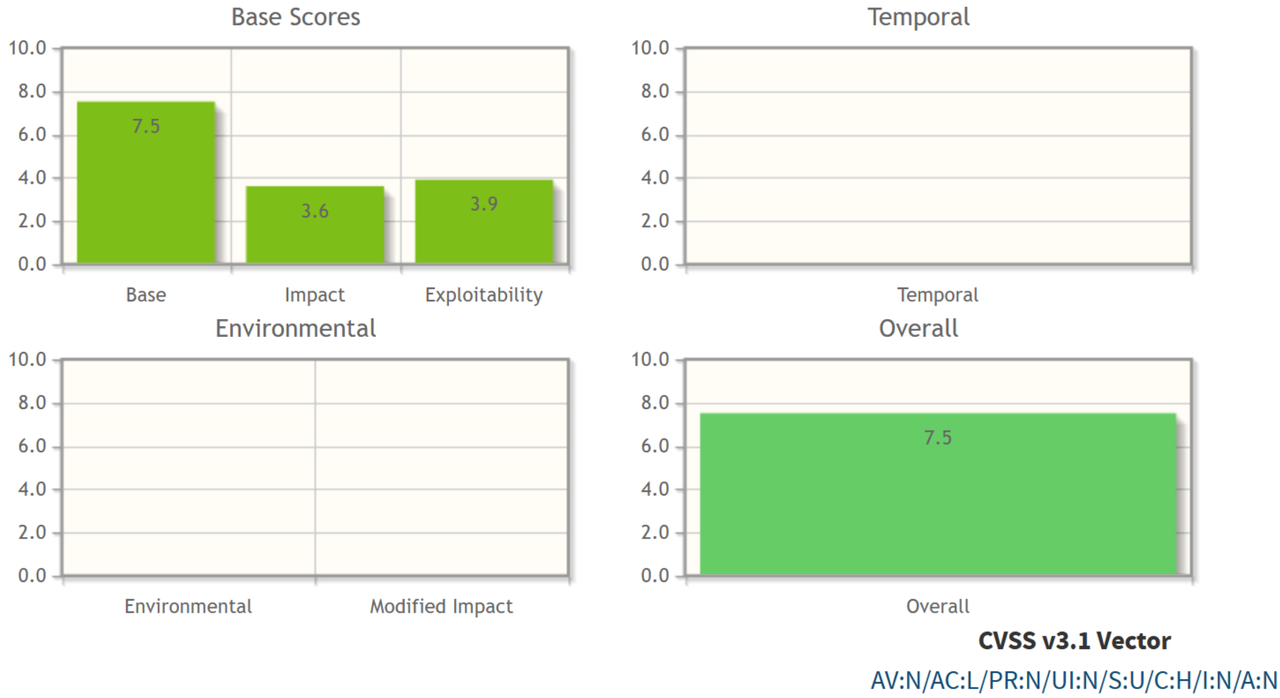

Figure 13: Insecure Communication Protocol Usage

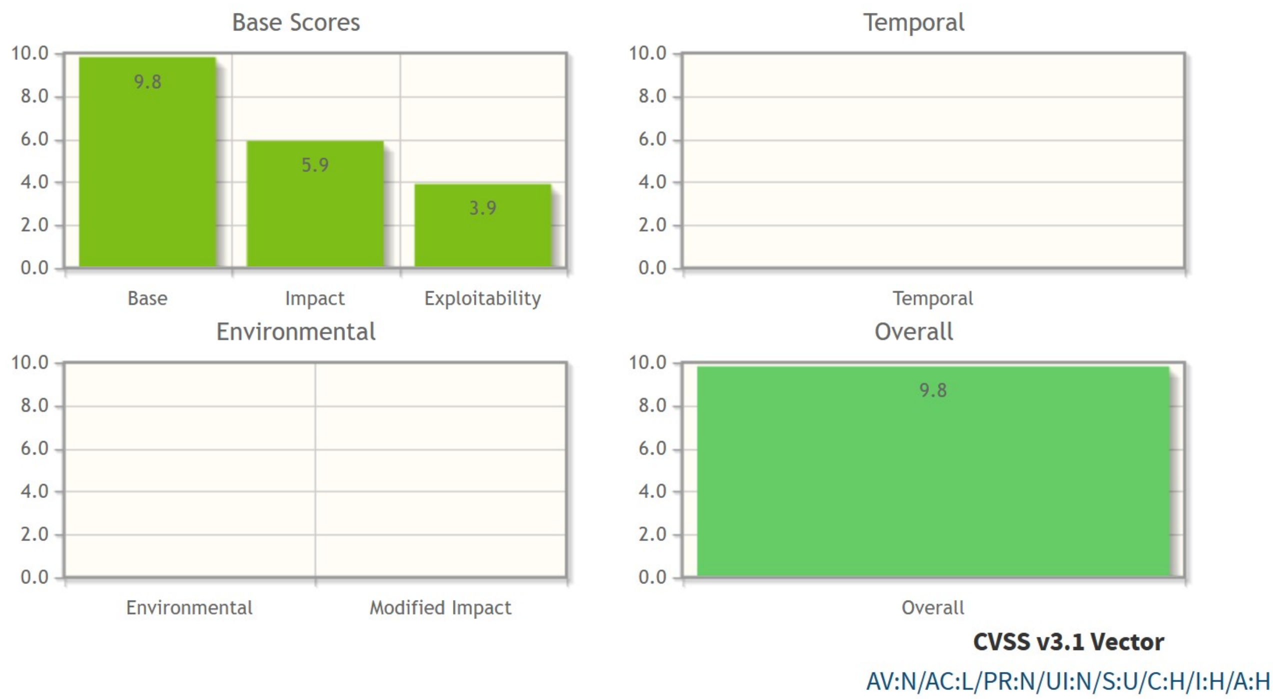

Figure 14: Lack of End-to-End Encryption

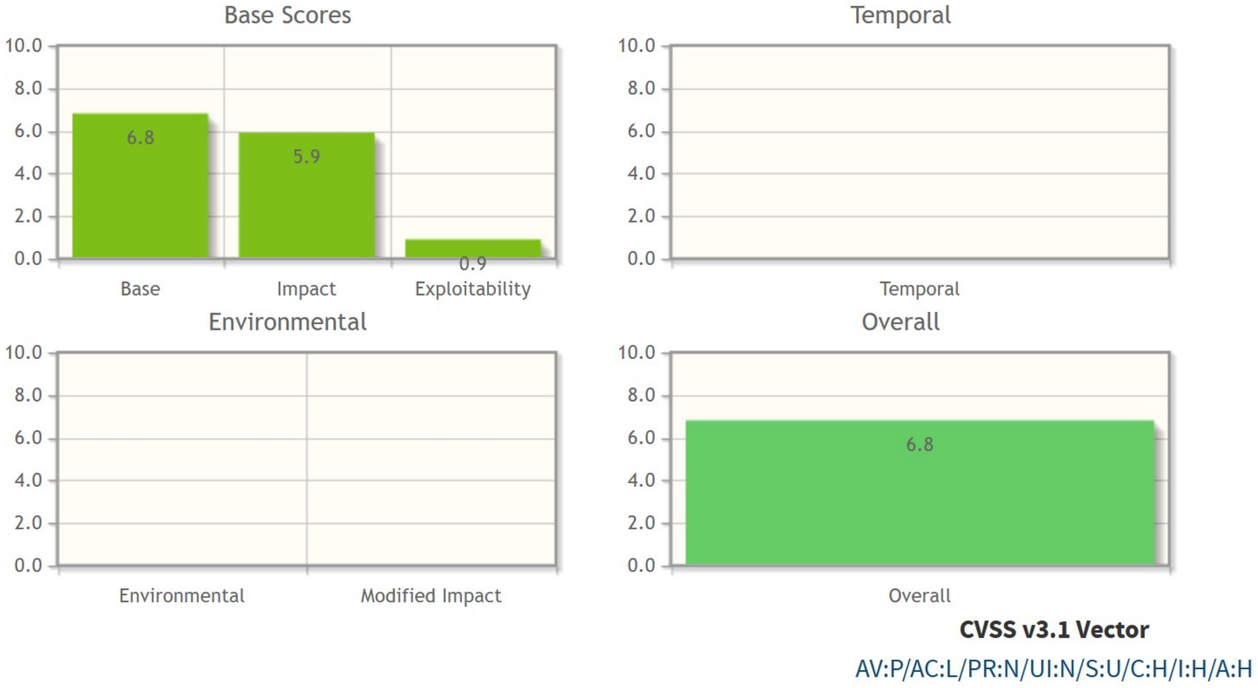

Figure 15: Attacker Physical Access to Devices

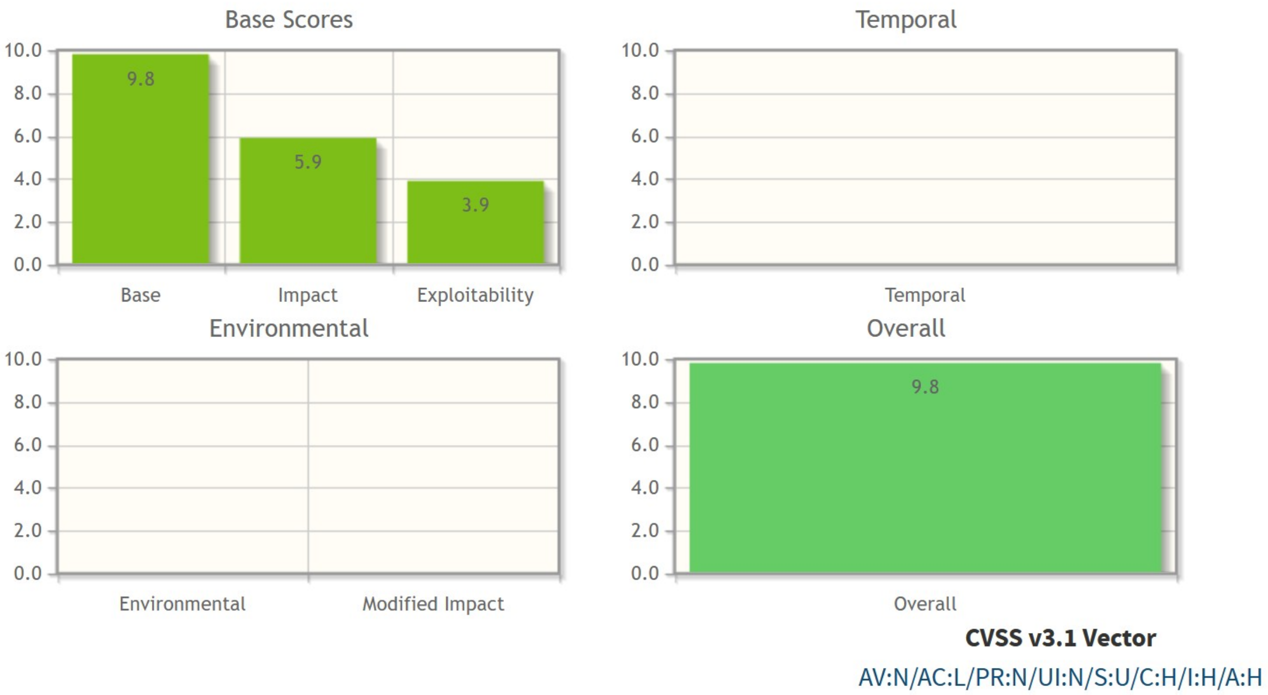

Figure 16: Exploitation of Out-Dated Firmware

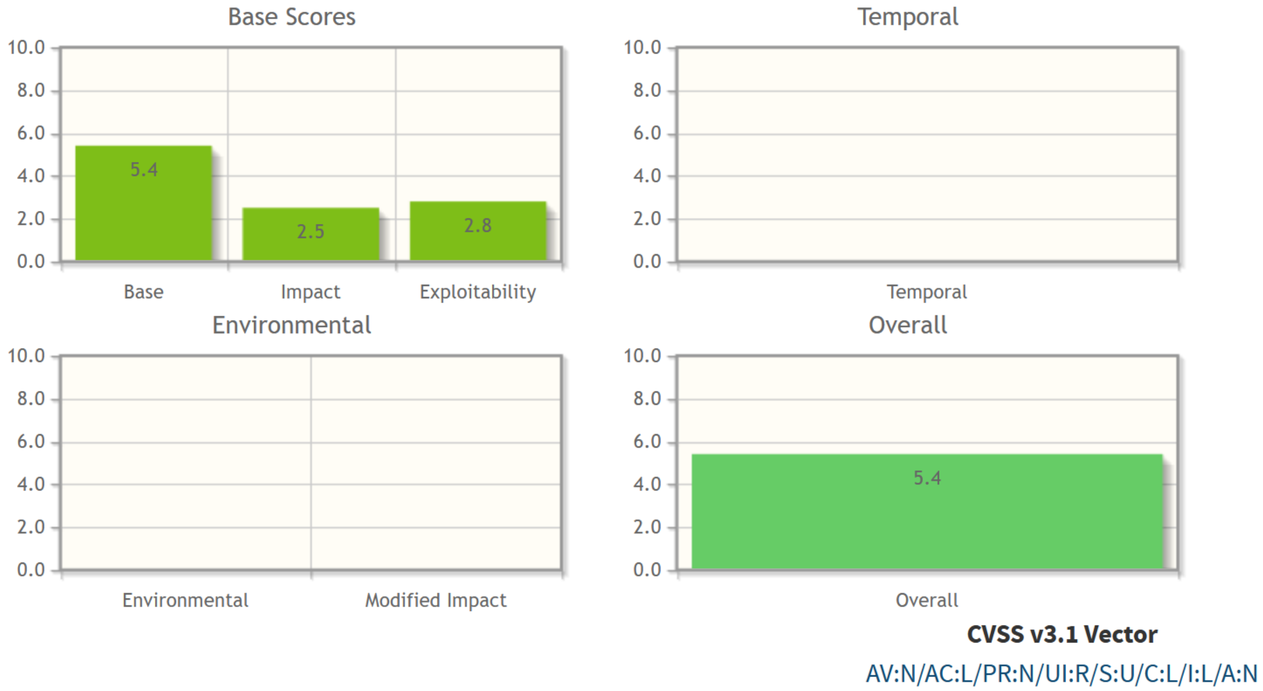

Figure 17: CVSS Scores for Exploitation of Downloaded App Malware

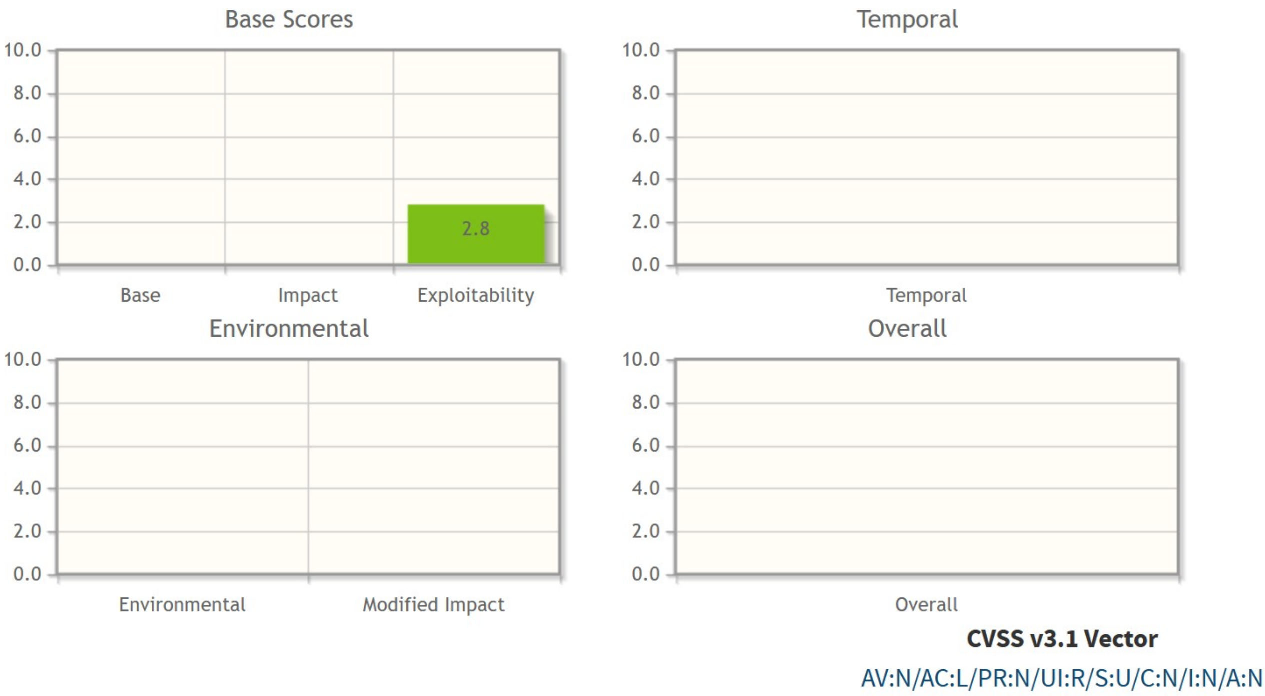

Figure 18: CVSS Scores for Data Collection Without Consent

5.2 Appendix II

Table 3: Full Statistical Analysis

Calculation	**Result**	**Equations**
Mean ($\bar{x}$)	7.063	t-Statistic: $t = \frac{\bar{x} - \mu}{\sigma_{\bar{x}}}$
Standard deviation (σ)	3.232	
Standard error of mean ($\sigma_{\bar{x}}$)	1.143	
Count	8.000	
Median	7.800	
Quartile 1	6.450	
Quartile 3	9.28	
Inter-quartile range	2.83	p-Value: $t^{*} \underset{H_1}{\overset{H_0}{\gtrless}} \alpha$
Degrees of freedom	7	
Hypothesised mean (μ)	3.9	
Alpha (α)	0.05	
t-Statistic (t^{*})	2.768	
p-Value	0.014	