**Development Team Project ‌**

*Design Document ‌for‌ Secure Systems Architecture ‌(SSA)‌ ‌  
Smart Home Vulnerability Report*

MSc‌ ‌Cyber‌ ‌Security‌ ‌

Unit‌ ‌3 ‌Submission‌ ‌

Words 657

[Introduction 2](#_Toc89097447)

[System Design 2](#_Toc89097448)

[Figure 1 2](#_Toc89097449)

[Vulnerability Assessment 2](#_Toc89097450)

[Figure 2.1 3](#_Toc89097451)

[Figure 2.2 4](#_Toc89097452)

[Figure 2.3 4](#_Toc89097453)

[Mitigations 5](#_Toc89097454)

[Figure 3 5](#_Toc89097455)

[Conclusion 6](#_Toc89097456)

[Appendix A 7](#_Toc89097457)

[Table 1 7](#_Toc89097458)

[Appendix B 8](#_Toc89097459)

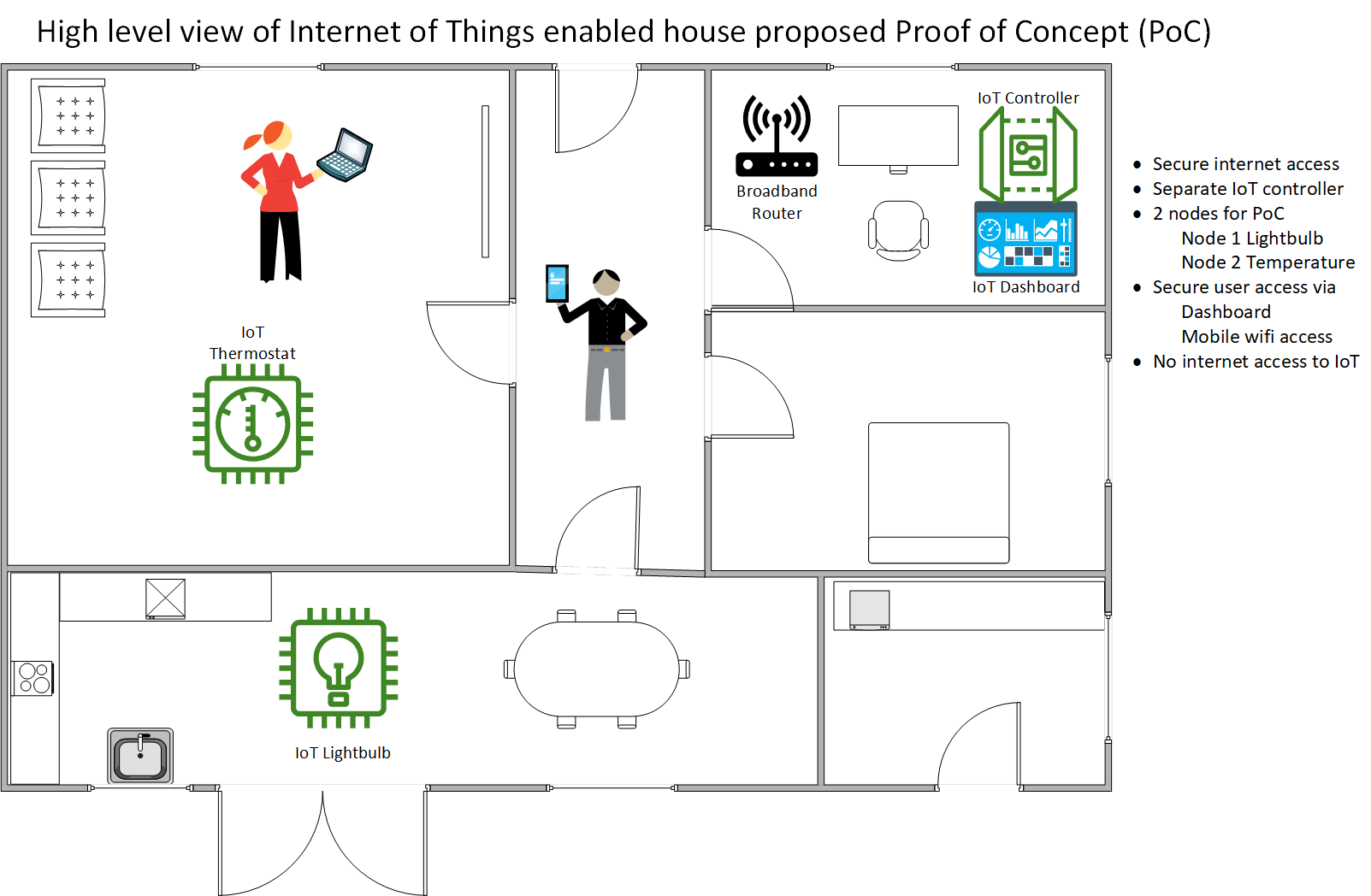
[Table 2 8](#_Toc89097460)

[References 9](#_Toc89097461)

## Introduction

A smart home system is being designed with the capability to control lighting and the temperature in a domestic setting. Internet of Things (IoT) devices, such as smart home systems, are at risk of security and privacy attacks (Davis et al., 2020). The purpose of this report is to recommend appropriate mitigations to address potential vulnerabilities, to protect users and their data.

## System Design

**

### Figure 1

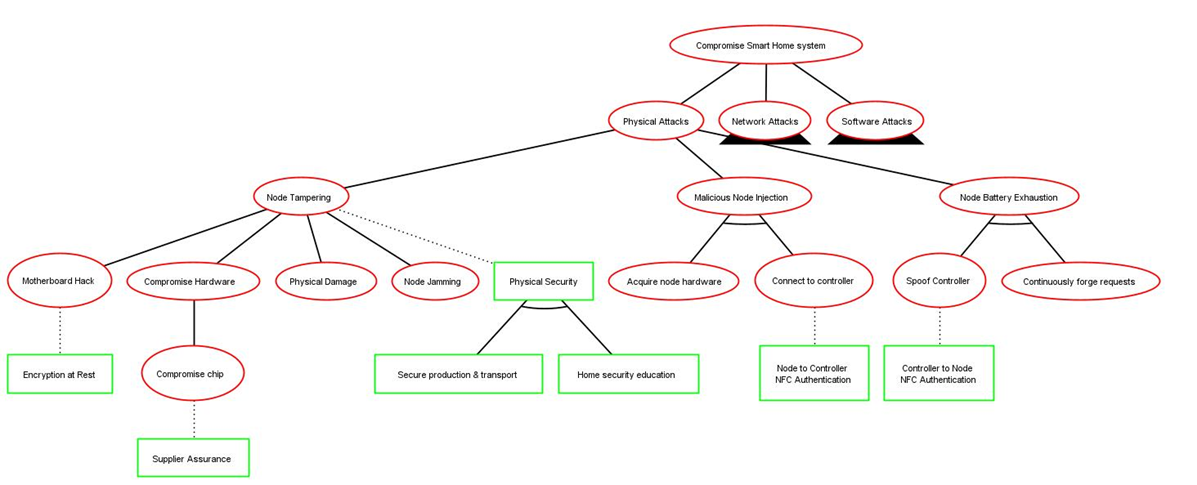
Figure 1 shows the smart home system that comprises a star network, with a controller acting as a gateway to the internet and two nodes; a smart light bulb and a thermostat.

## Vulnerability Assessment

An estimated 20 billion IoT devices were used in 2020, many holding personal data and potentially not secured (Abdullah et al., 2019). As such, these devices present a fruitful and relatively easy target for threat actors to exploit, whether the attack is targeted or random. For example, a cybercriminal may infect internet connected IoT devices with malware. Malware could be used to control devices as part of a botnet, denying access to the system in exchange for a ransom or used to collect personal data. The controller and nodes could be vulnerable to compromise, however exploitation of the controller has the capacity to impact all nodes.

Figures 2.1-2.3 show an Attack-Defense tree that enumerates potential vulnerabilities and mitigations of the smart home system (Kordy & Schweitzer, 2015; Lounis & Zulkernine, 2020). The ‘Probability of Success’ domain, based on estimations of the likelihood that a vulnerability will be successfully exploited, has been chosen to analyse the vulnerabilities in order to expose the most attractive attack paths for an attacker. There is not enough suitably detailed public attack data to accurately estimate the cost or time taken to complete attacks.

The attack objective is to compromise the smart home system and attack categories have been chosen in line with the standard IoT architecture layers; physical, network and application (Andrea et al., 2015). This ensures coverage of possible attacks, including physical, software, network and encryption attacks, as well as clarifying which layer mitigations in the diagram align to (Fraile et al., 2016). The vulnerabilities in Table 1 (Appendix A) have been deduced from Figures 2.1-2.3 and split to show which part of the system they relate to.



### Figure 2.1

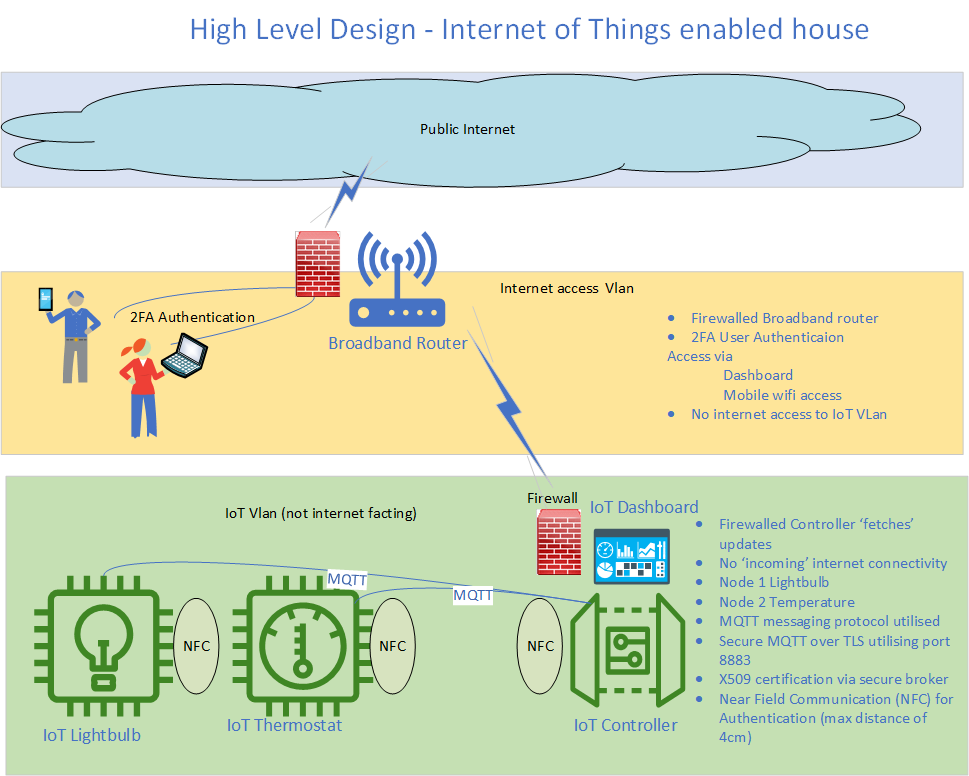
.

### Figure 2.2

### Figure 2.3

## Mitigations

The recommended mitigations (Table 2 [Appendix B]; Figure 3) have been derived from Figures 2.1-2.3 and additional best practices. These reduce security risks to the system whilst considering the impact to other system requirements, like performance, and device limitations, such as battery life. Whilst lightweight encryption is optimal for performance, Figure 2.2 demonstrates that it could be vulnerable to attack. Therefore, TLS will be used for encryption alongside a lightweight protocol, MQTT, to compensate for any strain on bandwidth and processing power. In addition, strong encryption is required to protect personal data in line with regulatory requirements (EUR-Lex, 2016). The controls are listed in priority order based on the number of potential vulnerabilities mitigated.

****The quantitative analysis (Table 1, Appendix A) shows that without controls in place, the overall probability of successful compromise is 99.8% (94.1% for network, 90.5% for software and 58.1% for physical). With the mitigations, this falls to 24.6% overall (8.7% for networks, 7.3% for software and 11% for physical). Whilst the ADTool only allows one countermeasure per vulnerability, a layered approach is recommended to improve effectiveness (Howard & LeBlanc, 2003). However, due consideration should be given to how layering may impact performance and usability.

### Figure 3

.

## Conclusion

In conclusion, this report provides an overview of a proposed smart home system. Potential security vulnerabilities and corresponding mitigations have been highlighted using an Attack-Defense Tree methodology. Cost of mitigations should be commensurate with anticipated risk based on the threats and vulnerabilities disclosed. All will be considered in the implementation of this smart home system. Also included is a high level design, showing how some security elements can be incorporated into the architecture.

## Appendix A

### Table 1

|  |  |  |
| --- | --- | --- |
| **No.** | **Description** | **Inherent Probability of Success** |
| **Both Controller & Node** | | |
| 1 | Overwhelm network capacity, e.g. Distributed Denial of Service (DDoS) | 50% |
| 2 | Malware introduction via social engineering, for example a phishing email sent to a user with a link to make urgent software updates to the smart home system. | 50% |
| 3 | Hijacking of resources, for example to create a botnet. | 50% |
| 4 | Intercept network communications | Up to 50% |
| 5 | Manipulate network communications | Up to 43.8% |
| 6 | Encryption key interception, e.g. side channel or MITM attacks. | 39.3% |
| 7 | Compromise via malicious open source software and/or libraries (Lakshmanan, 2021). | 32.5% |
| 8 | Software denial of service, for example via buffer overflow. | 25% |
| 9 | Physical damage to hardware components. | 25% |
| 10 | Malicious hardware components could be introduced throughout SDLC. | 10% |
| 11 | Cleartext data available in node hardware | 10% |
| **Controller Only** | | |
| 12 | Credential compromise via social engineering. | 50% |
| 13 | Unauthorised access to the controller via authentication attacks. | Up to 50% |
| 14 | Node spoofing | 25% |
| 15 | Remote code execution vulnerabilities. | 25% |
| **Node(s) Only** | | |
| 16 | Ability to deprive node(s) of sleep, resulting in battery exhaustion. | 12.5% |
| 17 | Controller spoofing | 25% |

## Appendix B

### Table 2

|  |  |  |
| --- | --- | --- |
| **No.** | **Recommended Mitigation** | **Connected Vulnerabilities** |
| 1 | Implement auto-update function by default (Davis et al., 2020), including security patching. | 3, 4, 5, 8, 13, 14, 16, 17 |
| 2 | End User Education - social engineering and home physical security measures. A best practice home network security guide could also be shared, including use of a VLAN to segregate the smart home system and changing default passwords (Abdullah et al., 2019). | 2, 3, 9, 10, 12, 13 |
| 3 | In-built logging and basic Intrusion Detection System on controller. | 3, 4, 5, 7, 15 |
| 4 | Firewall on controller, with use of access control lists (Andrea et al., 2015) and port restrictions to ensure internet connection is only open for system updates. The configuration should be maintained and the software should be kept up to date. | 1, 3, 5, 15 |
| 5 | Antivirus on controller with regular updates required. | 2, 3, 7, 15 |
| 6 | Bluetooth out of band pairing (OOB) via near field communication (NFC) for device authentication (Lounis & Zulkernine, 2020).  Application Layer authentication: via certificates | 6, 14, 16, 17 |
| 7 | Two-factor authentication (2FA) for the end user to authenticate with the controller to change settings and perform other actions. | 3, 12, 13 |
| 8 | Employ randomly generated, strong passwords on the controller by default and encourage users to change to a chosen strong password on setup (Coker, 2021). | 3, 13, 17 |
| 9 | All communications should be Transport Layer Security (TLS) encrypted with strong ciphers enforced. | 4, 5, 6 |
| 10 | Hardware secure key storage via dedicated chip. | 4, 5, 6 |
| 11 | Physical security controls to protect devices from tampering during manufacturing through to end user operation. | 9, 10, 11 |
| 12 | Restrict access to ports, such as Secure Shell (SSH) on port 22. | 13, 15 |
| 13 | Session management controls to support authentication on the controller, such as automatically logging the user off after one hour. | 3, 14 |
| 14 | Predetermined authorised actions based on device type, e.g. temperature alert via email but smart bulb has no authorisation to email. | 1, 3 |
| 15 | Following secure development practices, such as input validation (Howard and LeBlanc, 2003). | 8 |
| 16 | Verify software prior to download. For example, verify checksums when downloading software packages during development and verify signatures before installing firmware updates. | 7 |
| 17 | Ensure data minimisation on nodes and controller and sensitive data encrypted at rest. | 11 |
| 18 | Robust third party supplier security assurance, for chip manufacturers for example. | 10 |
| 19 | Use reputable open source software libraries. | 7 |
| 20 | Function for the operator to wipe personal data to support secure disposal (EUR-Lex, 2016). | 15 |
| 21 | Rate limit pairing attempts with the nodes to prevent malicious battery drainage. | 16 |

## References

Abdullah, T., Ali, W., Malebary, S. & Ahmed, A. (2019) A Review of Cyber Security Challenges, Attacks and Solutions for Internet of Things Based Smart Home. *International Journal of Computer Science and Network Security* 19(9): 139-146.

Andrea, I., Chrysostomou, C. & Hadjichristofi, G. (2015) ‘Internet of Things: Security vulnerabilities and challenges’, *3rd IEEE ISCC 2015 International Workshop on Smart City and Ubiquitous Computing Applications*. Larnaca, Cyprus, 6 July. IEEE. 180–187.

Coker, J. (November 24, 2021) UK Introduces New Cybersecurity Legislation for IoT Devices. *InfoSecurity Magazine*. Available from: <https://www.infosecurity-magazine.com/news/uk-cybersecurity-legislation-iot/> [Accessed 28 November 2021].

Davis, B., Mason, J. & Anwar, M. (2020) Vulnerability Studies and Security Postures of IoT Devices: A Smart Home Case Study. *IEEE Internet of Things Journal* 7(10): 10102-10110

EUR-Lex (2016) Regulation (EU) 2016/679 of the European Parliament and of the Council. Available from:<https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A02016R0679-20160504&qid=1532348683434> [Accessed 25 November 2021].

Fraile, M., Ford, M., Gadyatskaya, O., Kumar, R., Stoelinga, M. & Trujillo-Rasua, R. (2016) ‘Using Attack-Defense Trees to Analyze Threats and Countermeasures in an ATM: A Case Study’, *IFIP Working Conference on The Practice of Enterprise Modeling*. Skövde, Sweden, 8-10 November. Springer. 326-334.

Howard, M. and LeBlanc, D. (2003) *Writing secure code*. 2nd ed. Redmond, Washington: Microsoft Press.

Kordy, P. & Schweitzer, P. (2015) The ADTool Manual. Available from: <http://satoss.uni.lu/software/adtool/manual.pdf> [Accessed 20 November 2021].

Lakshmanan, R. (2021) 11 Malicious PyPI Python Libraries Caught Stealing Discord Tokens and Installing Shells. Available from: <https://thehackernews.com/2021/11/11-malicious-pypi-python-libraries.html> [Accessed 21 November 2021].

Lounis, K. & Zulkernine, M. (2020) Attacks and Defenses in Short-Range Wireless Technologies for IoT. *IEEE Access* 8: 88892-88932.