SecDevOps Gap Analysis Report

SIT374-SIT764 Team Project A (T2 2024)

*Redback Operations*

Table Of Contents

[1. Executive Summary 3](#_Toc893484994)

[2. Current State Analysis 3](#_Toc1418061763)

[2.1 DevOps Process 3](#_Toc412687225)

[2.2 MQTT Implementation 3](#_Toc259383755)

[2.3 CoAP Review T2 2024 3](#_Toc1705203639)

[3. Gap Analysis 4](#_Toc860002833)

[3.1 Malware and Vulnerability Scanning 4](#_Toc1546485126)

[3.2 Enablement for Malware and Vulnerability Scanning 4](#_Toc1347981613)

[3.3 Code Signing 4](#_Toc148110227)

[3.4 Use of SBOMs 5](#_Toc811040845)

[3.5 Policy Aligned Encryption 5](#_Toc1286123694)

[3.6 Device Birth Certificates 5](#_Toc1146761218)

[4. Recommendations 6](#_Toc1244996570)

[Strategic Recommendations 6](#_Toc1082098244)

[5. Implementation Plan 6](#_Toc1024452861)

[6. Conclusion 6](#_Toc819461019)

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Modified By | Date | Changes Made |
| V1.0 | Candice Smith | 01/08/2024 | Document Creation, draft of sections 1, 2.2, 3.1, 3.2 skeleton only, 3.3, 3.4, 3.5, 3.6 |
| V2.0 | Candice Smith | 08/09/2024 | Completed Document |
|  |  |  |  |
|  |  |  |  |

# Executive Summary

This report provides an overview of the current gaps in Redback Operations SecDevOps processes, tools, practices and controls. This report is the result of a comprehensive review by the current SecDevOps team, where current DevOps security practices including but not limited to code review and scanning, code signing, encryption and TLS, device identity, and other secure coding practices were assessed.

The intended outcome of this analysis is to create focus on areas of improvement for DevOps security practices, and opportunity for planning, testing, and implementation of tangible outcomes that will result in improved practices and security.

# Current State Analysis

## 2.1 DevOps Process

Currently the DevOps Process is to utilise GitHub as our central repository for everything. This includes all documentation from research papers, implementation plans, and code files. Any changes to GitHub repositories are made by submitting a pull request that must be reviewed by the SecDevOps Team, who then provide feedback regarding any changes that must be made to the code prior to uploading. Once the code has been completed to an acceptable level the SecDevOps reviewer approves the pull request, and the file can be finalised within the GitHub repository.

## 2.2 MQTT Implementation

In T1 2024 the Redback Cyber Security team stood up their first MQTT server. This was done within an Azure environment hosted on a VM on hardware ‘somewhere’ in the University. The borker was set up with only basic functionality in preparation for the team to deploy MQTT communications between the Redback devices and databases that will store collected consumer data. Project 1 are currently using MQTT, but they are not using the broker hosted within Azure. Project 1’s current MQTT implementation has been creating challenge with latency issues and has proven to be quite difficult to work with. The current MQTT broker being utilised by the project 1 team is the free cloud-based HiveMQ service, which is designed for testing and education purposes only. It does not offer the same level of performance that a broker such as the one that is currently sitting in our Azure environment would offer. This HiveMQ service is known to have latency issues and one of the resolutions is to host your own broker like the one set up in Azure.

## 2.3 CoAP Review T2 2024

The team for project 1 is currently undergoing a review of the use of CoAP as an alternative to MQTT for their IoT devices. They are still only in the early stages and there have been some delays in getting to the testing phase. It is not yet known which deployment of CoAP would be utilised for the implementation.

# Gap Analysis

## 3.1 Malware and Vulnerability Scanning

**Gap Analysis Finding:**

The requirement for malware and vulnerability scanning has been implemented, but there have been no supporting automated tools implemented. As a result of this, malware and vulnerability testing is estimated to be a heavy drain on resourcing within the SecDevOps team. There is additional value this team can be offering beyond this scanning. SecDevOps teams can be working proactively to eliminate barriers between security and DevOps teams, and even leading the charge on training developers in security so it becomes a more deeply entrenched practice throughout the organisation.

**Objective Summary:** To implement tools and automation that will improve efficiency within the malware and vulnerability scanning process to free up time within the SecDevOps team to focus on additional value adding activities.

**Key Subjects:**

* Review possible tools for automation of Malware and Vulnerability Checking.
* Make recommendation on a tool, or combination of tools to implement.
* Implementation of Malware and Vulnerability Checking tools.

### 3.2 Enablement for Malware and Vulnerability Scanning

**Gap Analysis Finding:**

As a core function of the SecDevOps team, it is critical that onboarding in this team includes enablement activities to ensure all team members can deliver manual code reviews as a back-up to tool failure, or in the case that intended tools cannot be used for any other reason. This must be done within the first 2-3 weeks to ensure that any pull requests coming through can be dealt with in a timely manner, ideally within 24 hours of submission.

**Objective Summary:** To deliver onboarding and enablement activities that ensure all members of the SecDevOps team understand how to conduct a manual code review, and in turn meet the KPI of review of all pull requests within 24 hours.

**Key Subjects:**

* Outline standard basic process.
* Create short video tutorials that outline individual parts of the process for conducting a manual code review.
* Present this information as an onboarding workflow.
* Include any new tools as an overview so team get a good understanding early in the unit.

### 3.3 Code Signing

**Gap Analysis Finding:**

Does not appear that Redback Operations have a current solution deployed for code signing nor any current policy. While code signing is considered best practice for development, it is also reliant on budgetary resourcing and requires tight control around who is granted the ability to sign code given the potential to integrate malware into code, which can then be signed and trusted as it moves through the software supply chain.

**Objective Summary:** To develop an easy to implement, cost effective solution for code signing as well as developing related policy.

**Key Subjects:**

* Signing throughout build process to create an audit trail.
* Integrate code signing practices with CI/CD pipeline tools such as GitHub.
* Implement FIPS 2 compliant key storage for code signing certificate to comply with baseline requirement as stipulated by CA/B Forum.
* Use timestamping to identify the point at which malware or vulnerabilities were injected, and potentially who is responsible.

### 3.4 Use of SBOMs

**Gap Analysis Finding:**

Software Bill of Materials, or SBOMs, are a useful way to itemize and track components of code throughout your organisation. In a company that has the potential to re-use sections of code across multiple applications, like various health devices for example, being able to identify problem code segments used across multiple devices is critical in tracking and mitigating vulnerabilities.

**Objective Summary:** To develop an easy to implement, cost effective solution for SBOMs and applicable policy for enforcement.

**Key Subjects:**

* Define and SBOM standards
* Research SBOM applications/programs that can be used
* Cost Effectiveness
* Ease of implementation
* Adherence to regulatory requirements

### 3.5 Policy Aligned Encryption

**Gap Analysis Finding:**

Does not appear that Redback Operations have a current solution deployed for encryption that aligns with current policy. There are current deployments of MQTT and research into a new CoAP implementation but neither seem to include a provision or consideration for encryption through the use of TLS certificates.

**Objective Summary:** To develop an easy to implement, cost effective solution for encryption that aligns with current encryption policy.

**Key Subjects:**

* Re-define what encryption standards will be used
* Re-define classification levels
* Research encryption applications/programs that can be used
* Cost Effectiveness
* Ease of implementation
* Adherence to regulatory requirements

### 3.6 Device Birth Certificates

**Gap Analysis Finding:**

Does not appear that Redback Operations include a Device Birth Certificate for their IoT devices to allow for assigning of identity, and a methodology for secure firmware updates.

**Objective Summary:** To develop an easy to implement, cost effective solution for device birth certificates as well as related policy.

**Key Subjects:**

* Research and define solution to be used for device birth certificates.
* Cost Effectiveness
* Ease of implementation
* Adherence to regulatory requirements

# Recommendations

### Strategic Recommendations

* Create a standard process for manual secure code review and ensure the SecDevOps team are trained within the first 3 weeks of the unit using enablement tools.
* Deploy integrated code review tools within GitHub to cover all repositories and minimise – potentially eliminating the need for manual review.
* Ensure encryption is implemented for all data and communications protocols using TLS or other PKI implementations.
* Ensure device identity and security around firmware updates is established for all IoT devices.

# Implementation Plan

The Redback Operations SecDevOps team will begin developing a range of material that directly addresses the gaps identified, including training programs and enablement tools, request for policy updates, and addition of TLS solution for all MQTT and CoAP implementations as per TLS Implementation Plan document. They will be researching testing and implementing automated tools that will greatly assist in the handling of pull requests as they come in from throughout the redback organisation.

# Conclusion

This gap analysis report outlines three critical areas for improvement:

* code review enablement and automated tools
* Encryption, namely TLS implementation for MQTT and CoAP protocols
* Code signing and use of SBOMs

Addressing the gaps identified will cause a drastic improvement in our security posture, aligning our practices with policies and legislation such as the Privacy Act 1988.