CMDB Design Document

Mark Collins: 12696434

Word Count: 978

# Introduction and Requirements

A Configuration Management Database (CMDB) is the single source of truth for an organisation's Configuration Items (CIs). CIs include Hardware, Software, Services, Interdependencies and Documentation (Dande & Li, 2023; Farayola et al., 2023).

## Security Team Requirements

Information and Cyber security are significantly important when designing a CMDB. Configuration data contains sensitive information, and unauthorised or uncontrolled changes can have a major impact on security (Farayola et al., 2023).

A CMDB can be used as an audit tool to assure governing bodies that appropriate legislation and standards are being followed, such as GDPR and PCI DSS. Compliance with relevant standards is one of the key elements of cybersecurity (Mughal, 2021).

## Asset and Services Management

When designing a CMDB, an accurate register of Assets and Services is needed, along with how they are configured, and how critical they are to an organisation's function. Any changes to these assets will need to be accurately recorded, so the CMDB should enable robust change management processes (Farayola et al., 2023), including how rational, impact assessments, and authorisation.

Details of how to manage an incident or vulnerability should be recorded with the Asset within the CMDB, including who to contact (local owner, administrator, 3rd party support), the Recovery Time Objective (RTO) and the Recovery Point Objective (RPO).

It is essential to have a record of who is able to make changes to assets and CIs. This can be managed with Administration groups.

## Incident and Vulnerability Management

Incidents and Vulnerabilities, while similar, require slightly different management.

When a Vulnerability is identified, it will need to be assessed. For this to happen, information such as CVSS will need to be recorded, and combined with the criticality of the asset affected to determine the impact to the organisation. A low impact and the vulnerability may be tolerated, or a change requested to be managed around other dependant services. However, if vulnerability is found to have a high impact, it may trigger an incident.

Incidents will need to record the affected asset, the source of identification, and the impact on the organisation. The CMDB should be used to record any action taken to resolve the incident and trigger an update to configuration data should this be required. To manage an incident, usually the first step (after logging the issue) is to perform initial diagnostics to isolate the incident, followed by further troubleshooting by the team responsible for the CI. Next, when the incident is resolved, an assessment will need to be performed to identify the root cause (4).

# Design

## Tables and Design Scheme

The tables and interdependencies are shown in Appendix 1.

## Output

There are two main output purposes of a CMDB. Operational, and Audit/Compliance.

### Operational Output

The operational output from the dashboard provides a summary of the status of CIs. This would include (but not limited to):

* Incidents by status.
* Vulnerabilities by status.
* Open Change Requests.
* Number (or percentage of total) assets with open incidents or vulnerabilities, by type of incident.

### Compliance Output

The ability to produce compliance reports from the CMDB is important for providing assurance against various governing bodies. An example report is shown in Figure 1. For a given Asset, the incident history, and change log can be viewed, along with any known vulnerabilities.

# Demonstration

## Adding a new Asset

* Asset is identified
* Details collected from user/supplier
* New entry is created in CMDB
* Risk Assessment Performed
* Asset configured and details recorded

This manual process can be improved through the use of automation. There are tools that allow for automation with ITSM Configuration Management (Dande et al., 2024). New items added to the network can be identified and configuration data obtained automatically (IP, port, etc.).

## Updating Configuration Data

* Change request is raised and the change management process is followed (Arisenta et al., 2020).
* Relevant configuration changes are made to the Asset entry upon change completion.

This process would benefit from an automated process. There is a significant risk that the change is not accurately recorded in the correct place. Either the change details from the change request filter through the CMDB or there is an automated process as mentioned above that will detect changes.

It would be particularly beneficial to have a service running on a network, comparing CI configuration against the record in the CMDB. This way, deviations can be identified and investigated, ensuring any risk is contained.

## Conducting a vulnerability Assessment

* Vulnerability is identified through appropriate channels, e.g. Computer Emergency Response Team (CERT).
* Vulnerability is logged in CMDB.
* CVE is reviewed and CVSS combined with the local criticality of the asset.
* Impact assessment is performed, and recommendations made:
  + Act now - manage as incident
  + Act later - manage as a change request
  + Do nothing - risk is minimal, and implementing a patch may have adverse consequences on dependant systems.
* Open and tolerated vulnerabilities should be routinely assessed, using the routine Audit query shown in Appendix 1.

## Generating Compliance Reports

For reviewing against a certain standard, the following fields can be queried:

* Asset
  + Legislation/Standard (filtered for the required Standard)
  + Asset ID
  + Asset Name
  + Asset Owner
* Incidents
  + Asset ID (linked to Asset query)
  + Date Raised (filtered for period required)
  + Severity
  + Status or date closed
  + Date Closed
  + Resolution
* Vulnerabilities
  + Asset ID (linked to Asset query)
  + Date Raised (filtered for period required)
  + Severity
  + Status or date closed
  + Date Closed
* Changes
  + Asset ID (Linked to Asset query)
  + Date Raised (filtered for period required)
  + Change details
  + Implementation Date.

# Conclusion

A Configuration Management Database is a useful tool for monitoring and managing all configuration Items within an organisation. The usefulness of the CMDB is not limited to a simple registry of assets but can be used to manage incidents, vulnerabilities and change requests.

# References

Arisenta, R., Suharjito, & Sukmandhani, A. A. (2020) 'Evaluation Model of Success Change Management in Banking Institution Based on ITIL V3 (Case Study)', *2020 International Conference on Information Management and Technology (ICIMTech).* Bandung, Indonesia, August 2020. IEEE. 470–475. DOI: https://doi.org/10.1109/ICIMTech50083.2020.9211191

Dande, F., & Li, X. (2023) 'Enterprise Service Management Cybersecurity Threats: Exploring Cloud Configuration Management Database (CMDB) Implementation Within Community Colleges', *Proceedings of the International Conference on Industrial Engineering and Operations Management*. *8th North American Conference on Industrial Engineering and Operations Management*. Houston, USA, 15th June 2023. IEOM Society International. DOI: https://doi.org/10.46254/NA8.20230157

Dande, F., Li, X., Shofoluwe, M., & Mcleod, A. (2024) 'Artificial Intelligence integration in IT Service Management: An ITIL configuration management process review'. *Proceedings of the Industrial Engineering and Operations Management World Congress*, Detroit, Michigan, 9th October 2024

Farayola, O. A., Hassan, A. O., Adaramodu, O. R., Fakeyede, O. G., & Oladeinde, M. (2023) Configuration Management in the Modern Era: Best Practices, Innovations, and Challenges. *Computer Science & IT Research Journal* 4(2) DOI: https://doi.org/10.51594/csitrj.v4i2.613

Mughal, A. A. (2021) Cybersecurity Architecture for the Cloud: Protecting Network in a Virtual Environment. *International Journal of Intelligent Automation and Computing* 4(1)

# Appendix 1 – Design Scheme for CMDB

