Microsoft Security Assurance

Web Application Penetration Test

Project: Business Central

Dynamics 365

December 20th, 2022

Contents

Chapter 1 | Project Summary

Between October and December of 2022, NetSPI performed remediation testing of Microsoft's Dynamics 365 Business Central application to verify that the issues identified in the web application penetration test conducted between July and September of 2022 had been fixed. The original test as well as this remediation test was performed by NetSPI on Dynamics 365 Business Central application to identify vulnerabilities, determine the level of risk they present to Microsoft, and provide actionable recommendations to reduce this risk. NetSPI compiled this report to provide Microsoft with detailed information on each vulnerability discovered within the Dynamics 365 Business Central application, including potential business impacts, specific remediation instructions, and current remediation status.

1.1 Project Objectives

NetSPI's primary goal within this project was to provide Microsoft with an understanding of the current level of security in the Dynamics 365 Business Central application and its infrastructure components.

NetSPI completed the following objectives to accomplish this goal:

- Identifying application-based threats to and vulnerabilities in the application
- Comparing Microsoft's current security measures with industry best practices
- Providing recommendations that Microsoft can implement to mitigate threats and vulnerabilities and meet industry best practices
- Conducting remediation testing on the previously identified issues to determine if they have been fixed

1.2 Scope & Timeframe

The original testing and verification was performed between July and September of 2022. Remediation testing of medium severity findings was performed between October and December of 2022.The scope of this project was limited to the Dynamics 365 Business Central application and the specific infrastructure on which the application resides. Unless otherwise noted, all tested vulnerabilities that were found to be not remediated use the original verification steps to exploit the finding.

NetSPI conducted the tests using a non-production version of Dynamics 365 Business Central. All other applications and servers were out of scope. All testing and verification was conducted from outside of Microsoft's offices.

1.3 Summary of Findings

NetSPI's assessment of the Dynamics 365 Business Central application revealed the following vulnerabilities:

- 2 medium severity vulnerabilities
- 3 low severity vulnerabilities

TABLE 1: FINDINGS SUMMARY

VULNERABILITY NAME	SEVERITY	OWASP	MICROSOFT NOTES
JWT - Excessive Token Lifetime	Medium	A7-Identification and Authentication Failures	No Change, by Design, reviewed with Microsoft identity team and determined to be a low risk, low severity finding.
Weak Configuration - SSL/TLS - Deprecated Protocol	Medium	A2-Cryptographic Failures	No Change, by Design, This finding is for graph api supporting an older version of TLS and SSL. At this time Business Central only uses TLS 1.2 across all communication. Determined not be a risk based on current implementation.
Information Disclosure - Inadequate Cache Control Security-Policy	Low	A5-Security Misconfiguration	No Change, Not Retested
Vulnerable Version - jQuery UI	Low	A6-Vulnerable and Outdated Components	No Change, Not Retested

The following table lists the OWASP Top 10 vulnerabilities and indicates which issues were identified in the Dynamics 365 Business Central application.

CATEGORY	FOUND
A1-Broken Access Control	No
A2-Cryptographic Failures	Addressed/Remediated
A3-Injection	No
A4-Insecure Design	No
A5-Security Misconfiguration	Not Retested
A6-Vulnerable and Outdated Components	Not Retested
A7-Identification and Authentication Failures	Addressed/Remediated
A8-Software and Data Integrity Failures	No
A9-Security Logging and Monitoring Failures	No
A10-Server-Side Request Forgery (SSRF)	No

TABLE 2: OWASP SUMMARY

Chapter 2 | Technical Summary

2.1 Overview

The detailed findings section contains the analysis and documentation of the vulnerabilities identified within the Dynamics 365 Business Central application. This analysis included:

- Identifying potential vulnerabilities associated with the Dynamics 365 Business Central application
- Assigning appropriate severity rankings to valid vulnerabilities and risks
- Formulating useful action-based recommendations that can improve the security posture of the IT environment

Vulnerabilities are grouped according to severity. Information for each of the vulnerabilities includes the following:

Name: The name of the vulnerability.

Severity: Each of the vulnerabilities has been assigned a severity based on its impact to the application and its associated resources. The following table summarizes the three severity levels:

SEVERITY	DESCRIPTION
High	Vulnerabilities that result in unauthorized access to application data or functionality, unauthorized access to the server file system, OS command execution, and exposure of sensitive data (e.g., personally identifiable information).
Medium	Vulnerabilities that result in the exposure of session data or security configuration information. Unencrypted transmission of sensitive data or use of weak encryption methods.
Low	Vulnerabilities that result in the exposure version information or non-critical configuration information. Implementation of weak password policies and procedures. Informational findings that may not require any remediation.

TABLE 3: SEVERITY REFERENCES

The severity ratings in this document are based upon industry standard and do not necessarily take into consideration the environment in which the vulnerabilities exist, other controls that maybe implemented within that environment, or an organization's classification of the information or functionality. As a result, the severity ratings in this document will not clearly represent the overall risk to an organization for each vulnerability instance.

OWASP Category: Reference to the OWASP Top 10 web application security risks (2021).

Affected Assets and Services: Specific assets and associated services on which the vulnerability was found.

Vulnerability Details: Comprehensive explanation of the vulnerability that was found, including a high-level summary of how the vulnerability works.

Business Impact: This describes the potential business impact of the vulnerability, should it be exploited.

Recommendation: NetSPI's solution for repairing the vulnerability or mitigating the problem if no fix is yet available.

Affected URLs and Parameters: URLs and parameters associated with the finding, if applicable.

Verification: Screenshot or sample data from one instance of the finding showing how NetSPI has verified the finding manually, when possible.

References: These are other resources that have more information on the vulnerability.

Appendix A | NetSPI Contact Information

Please contact NetSPI with any questions regarding the findings, analysis, or recommendations contained in this report.

Security Consultant
Naveen Ramesh
Naveen.Ramesh@netspi.com
+91 998.965.1250

Security Consultant
Sneha Karan
Sneha.Karan@netspi.com
+91 897.679.8131

Security Consultant
Andrew Elgard
Andrew.Elgard@netspi.com
+17633137826

Project Manager
Destiny Watson
Destiny.Watson@netspi.com
+19183731382

Account Manager
Hannah Detra
Hannah.Detra@netspi.com
+13202667580

Appendix B | Web Application Penetration Test Methodology

The following sections provide an overview of the Web Application Penetration Test.

Information Gathering

During each Web Application Penetration Test, NetSPI first works with Microsoft to define project requirements and goals, identify areas of risk and concern, and gather the information necessary to assess the application. An application walkthrough is performed with Microsoft to help NetSPI better understand the application's architecture and business logic requirements, as well as to align expectations in terms of the testing approach. This information is used by the primary consultant and supporting team members to develop a test plan. This test plan is used as a basis for assessing the application and serves as a quality assurance measure.

Testing and Evaluation

NetSPI assesses Microsoft's web application for known security vulnerabilities from the perspectives of anonymous and authenticated users. If multiple user types exist, testing is performed for each type. During the assessment, manual and automated processes are followed that leverage commercial, open source, and proprietary software. All automated test results are manually verified to reduce false positives. NetSPI also conducts manual testing to identify data flow, business logic, and access control issues. The assessment includes testing for OWASP Top 10 2021 web application vulnerabilities.

CATEGORY	DESCRIPTION
A1-Broken Access Control	Access control enforces policy such that users cannot act outside of their intended permissions. Failures typically lead to unauthorized information disclosure, modification, or destruction of all data or performing a business function outside the user's limits.
A2-Cryptographic Failures	Many web applications and APIs do not properly protect sensitive data, such as financial, healthcare, and PII. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes. Sensitive data may be compromised without extra protection, such as encryption at rest or in transit, and requires special precautions when exchanged with the browser.
A3-Injection	Injection flaws, such as SQL, NoSQL, OS, and LDAP injection, occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.
A4-Insecure Design	Insecure design is a broad category representing different weaknesses, expressed as "missing or ineffective control design." There is a difference between insecure design and insecure implementation; design flaws and implementation defects have different root causes and remediation. A secure design can still have implementation defects leading to vulnerabilities that may be exploited. An insecure design cannot be fixed by a perfect implementation as by definition, needed security controls were never created to defend against specific attacks. One of the factors that contribute to insecure design is the lack of business risk profiling inherent in the software or system being developed, and thus the failure to determine what level of security design is required.
A5-Security Misconfiguration	Security misconfiguration is the most seen issue. This is commonly a result of insecure default configurations, incomplete or ad hoc configurations, open cloud storage, misconfigured HTTP headers, and verbose error messages containing sensitive information. Not only must all operating systems, frameworks, libraries, and applications be securely configured, but they must be patched and upgraded in a timely fashion.

CATEGORY	DESCRIPTION
A6-Vulnerable and Outdated Components	Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications and APIs using components with known vulnerabilities may undermine application defenses and enable various attacks and impacts.
A7-Identification and Authentication Failures	Application functions related to authentication and session management are often implemented incorrectly, allowing attackers to compromise passwords, keys, or session tokens, or to exploit other implementation flaws to assume other users' identities temporarily or permanently.
A8-Software and Data Integrity Failures	Software and data integrity failures relate to code and infrastructure that does not protect against integrity violations. An example of this is where an application relies upon plugins, libraries, or modules from untrusted sources, repositories, and content delivery networks (CDNs). An insecure CI/CD pipeline can introduce the potential for unauthorized access, malicious code, or system compromise.
A9-Security Logging and Monitoring Failures	Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain persistence, pivot to more systems, and tamper, extract, or destroy data. Most breach studies show time to detect a breach is over 200 days, typically detected by external parties rather than internal processes or monitoring.
A10-Server-Side Request Forgery (SSRF)	In a Server-Side Request Forgery (SSRF) attack, the attacker can abuse functionality on the server to read or update internal resources. The attacker can supply or modify a URL which the code running on the server will read or submit data to, and by carefully selecting the URLs, the attacker may be able to read server configuration such as AWS metadata, connect to internal services like http enabled databases or perform post requests towards internal services which are not intended to be exposed.

Data Analysis

All of the data collected is consolidated and analyzed using the NetSPI Resolve™ platform. Additional research is conducted to identify known vulnerabilities for individual application components. After identifying, analyzing, and prioritizing vulnerabilities, NetSPI formulates recommendations for mitigating each of these security issues. During this phase, supporting team members walk through the test plan with the primary consultant to ensure the integrity of the results. A report containing findings and recommendations is then generated by the primary consultant and placed through both technical and stylistic review of supporting team members, as well as through a final review by the engagement manager.

Basis for Opinions

NetSPI, through its experience, has worked to interpret regulations and industry standards, such as National Institute of Standards and Technology (NIST) standards, the Open Web Application Security Project (OWASP) guidelines, MITRE ATT&CK® framework, and Payment Card Industry Data Security Standard (PCI DSS), recognize security best practices, and apply these within the context of Microsoft's IT environment.

Collaboration

In this phase, NetSPI presents an overview of the findings and delivers the preliminary report to the Microsoft project team. NetSPI reviews the web application's strengths and weaknesses with Microsoft and discusses the recommendations for addressing security deficiencies. Microsoft will have an opportunity to provide feedback and guidance for report revisions and the final presentation.

Presentation

After an agreed-upon timeframe, NetSPI finalizes the report, incorporating any feedback from Microsoft. This document in the final version is delivered in all required formats and to all required parties.

Appendix C | Risk Management Approach Overview

This section provides an overview of the risk management approach used by NetSPI during the project.

1. NetSPI worked with the client to identify the individuals from both sides that needed to be involved or made aware of the project. In the event of an issue, good communication helps ensure that emergency reactions to testing activities are not made; ad-hoc system changes during the test may invalidate test results and result in a service disruption.

2. NetSPI worked with the client to identify potential areas of risk that relate to the networks, systems, and applications that were tested directly or could be affected by tested.

3. NetSPI and the client created and executed on action items to address the identified areas of risk. Responsibilities were assigned to both teams.

4. NetSPI and the client created an escalation procedure that included a calling tree to address and reduce the impact of potential incidents. Calling trees typically include up to three contacts from the NetSPI and the client to ensure that the appropriate action can be taken as soon as possible.