Secure Software Development

SE4030

[2022/JUL]

Assignment 2

Submitted to

Sri Lanka Institute of Information Technology

Development Group Assignment Report

2022_REG_25

17.10.2022

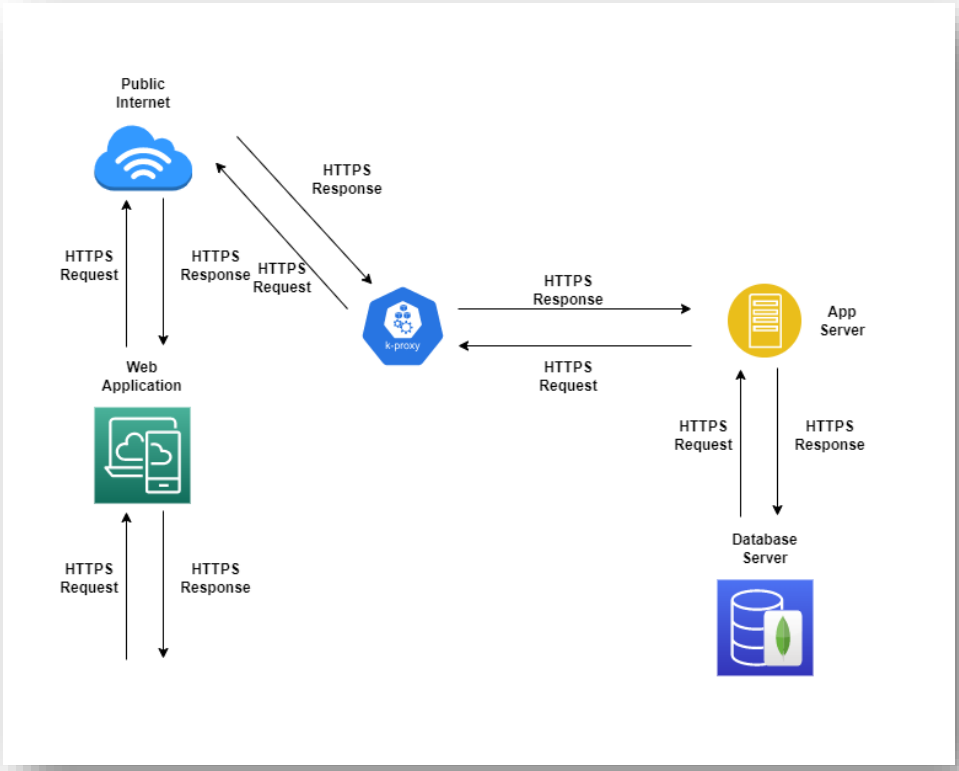
Name	Registration Number
Soujanyaah. K	IT19123400
Sivarajah. K	IT19119786
Viknatharshan. N	IT19107110
Sivashankar. S	IT19047102

Declaration

I certify that this report does not incorporate without acknowledgement, any material previously submitted for a degree or diploma in any university, and to the best of my knowledge and belief it does not contain any material previously published or written by another person, except where due reference is made in text.

Source Code Git Repository Link - https://github.com/soujanyaah/Group2022_Reg_25

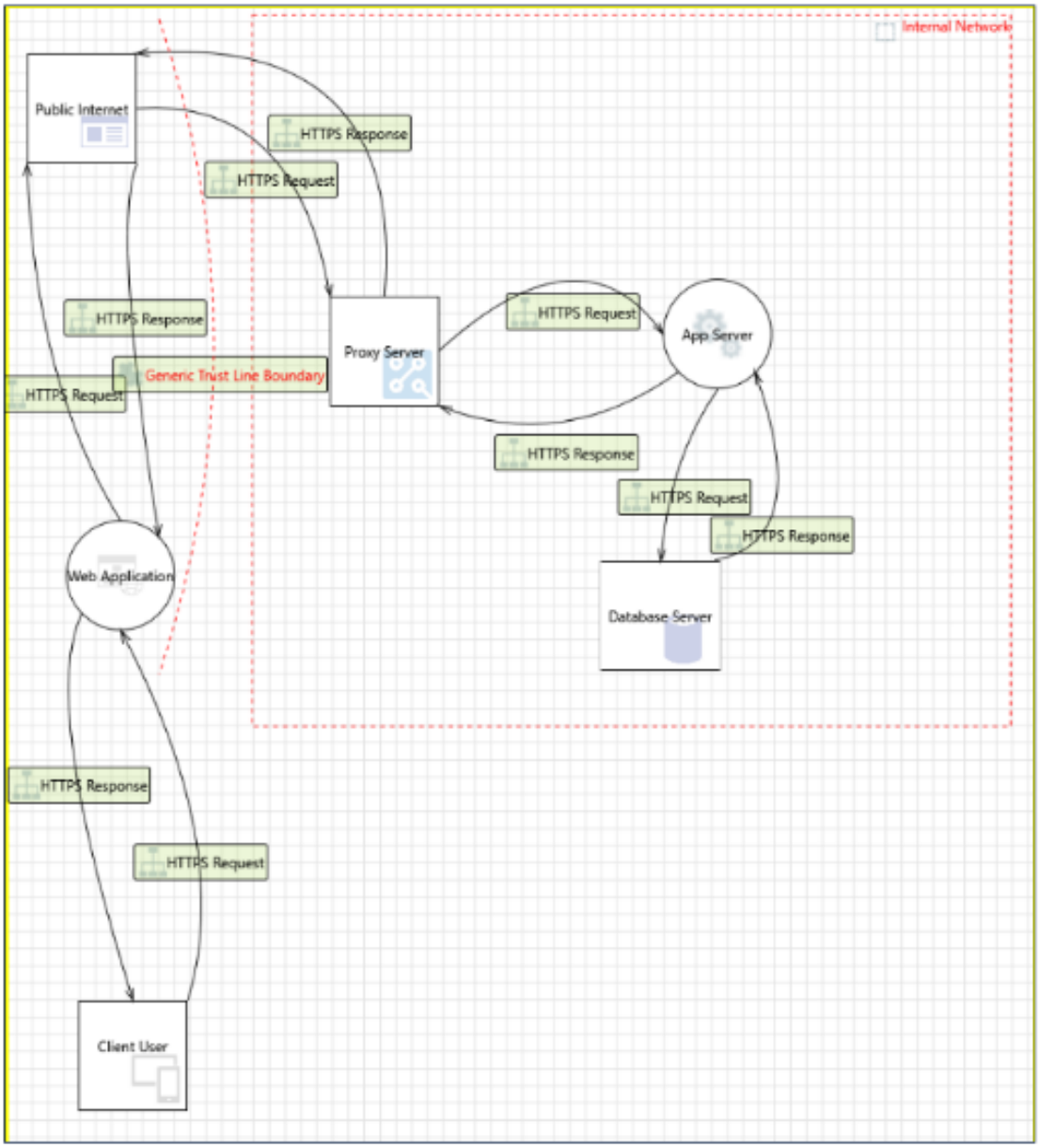
Software System Diagram



Thread Modeling

- We used **Microsoft Thread Modeling Tool** to design the main process of the software developing and identify the threats that the software faces and the mitigations that can be used to secure it.

Diagram: System Diagram



System Diagram Summary

Threat Modeling Report

Created on 16/11/2022 17:46:43

Threat Model Name: ABC Company Threat Model

Owner:

Reviewer:

Contributors:

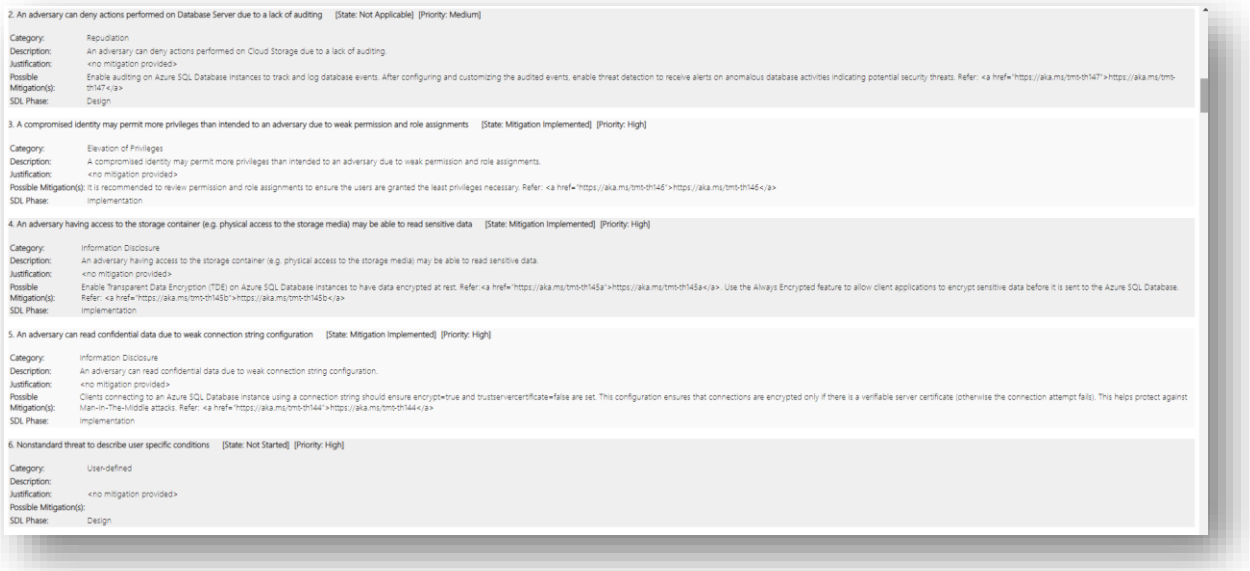
Description:

Assumptions:

External Dependencies:

Threat Model Summary:

Not Started	35
Not Applicable	5
Needs Investigation	1
Mitigation Implemented	25
Total	66
Total Migrated	0



2. An adversary can deny actions performed on Database Server due to a lack of auditing [State: Not Applicable] [Priority: Medium]

Category: Repudiation
Description: An adversary can deny actions performed on Cloud Storage due to a lack of auditing.
Justification: <no mitigation provided>
Possible Mitigation(s): Enable auditing on Azure SQL Database instances to track and log database events. After configuring and customizing the audited events, enable threat detection to receive alerts on anomalous database activities indicating potential security threats. Refer: https://aka.ms/tmt-th147
SDL Phase: Design

3. A compromised identity may permit more privileges than intended to an adversary due to weak permission and role assignments [State: Mitigation Implemented] [Priority: High]

Category: Elevation of Privileges
Description: A compromised identity may permit more privileges than intended to an adversary due to weak permission and role assignments.
Justification: <no mitigation provided>
Possible Mitigation(s): It is recommended to review permission and role assignments to ensure the users are granted the least privileges necessary. Refer: https://aka.ms/tmt-th146
SDL Phase: Implementation

4. An adversary having access to the storage container (e.g. physical access to the storage media) may be able to read sensitive data [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure
Description: An adversary having access to the storage container (e.g. physical access to the storage media) may be able to read sensitive data.
Justification: <no mitigation provided>
Possible Mitigation(s): Enable Transparent Data Encryption (TDE) on Azure SQL Database instances to have data encrypted at rest. Refer:https://aka.ms/tmt-th145a. Use the Always Encrypted feature to allow client applications to encrypt sensitive data before it is sent to the Azure SQL Database. Refer: https://aka.ms/tmt-th145b
SDL Phase: Implementation

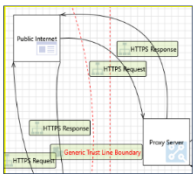
5. An adversary can read confidential data due to weak connection string configuration [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure
Description: An adversary can read confidential data due to weak connection string configuration.
Justification: <no mitigation provided>
Possible Mitigation(s): Clients connecting to an Azure SQL Database instance using a connection string should ensure encrypt=true and trustservercertificate=false are set. This configuration ensures that connections are encrypted only if there is a verifiable server certificate (otherwise the connection attempt fails). This helps protect against Man-in-The-Middle attacks. Refer: https://aka.ms/tmt-th144
SDL Phase: Implementation

6. Nonstandard threat to describe user specific conditions [State: Not Started] [Priority: High]

Category: User-defined
Description:
Justification: <no mitigation provided>
Possible Mitigation(s):
SDL Phase: Design

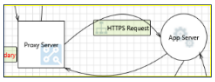
Interaction: HTTPS Request



7. An adversary may execute unknown code on Proxy Server [State: Not Started] [Priority: High]

Category: Tampering
Description: An adversary may launch malicious code into Proxy Server and execute it
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that unknown code cannot execute on devices. Refer: https://aka.ms/tmtconfigmgmt#unknown-exe
SDL Phase: Design

Interaction: HTTPS Request



8. An adversary may gain unauthorized access to privileged features on Proxy Server [State: Mitigation Implemented] [Priority: High]

Category: Elevation of Privileges
Description: An adversary may get access to admin interface or privileged services like WiFi, SSH, File shares, FTP etc., on a device
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that all admin interfaces are secured with strong credentials. Refer: https://aka.ms/tmtconfigmgmt#admin-strong
SDL Phase: Implementation

10. An adversary may exploit known vulnerabilities in unpatched devices [State: Not Started] [Priority: High]

Category: Tampering
Description: An adversary may leverage known vulnerabilities and exploit a device if the firmware of the device is not updated
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that the Cloud Gateway implements a process to keep the connected devices firmware up to date. Refer: https://aka.ms/tmtconfigmgmt#cloud-firmware
SDL Phase: Design

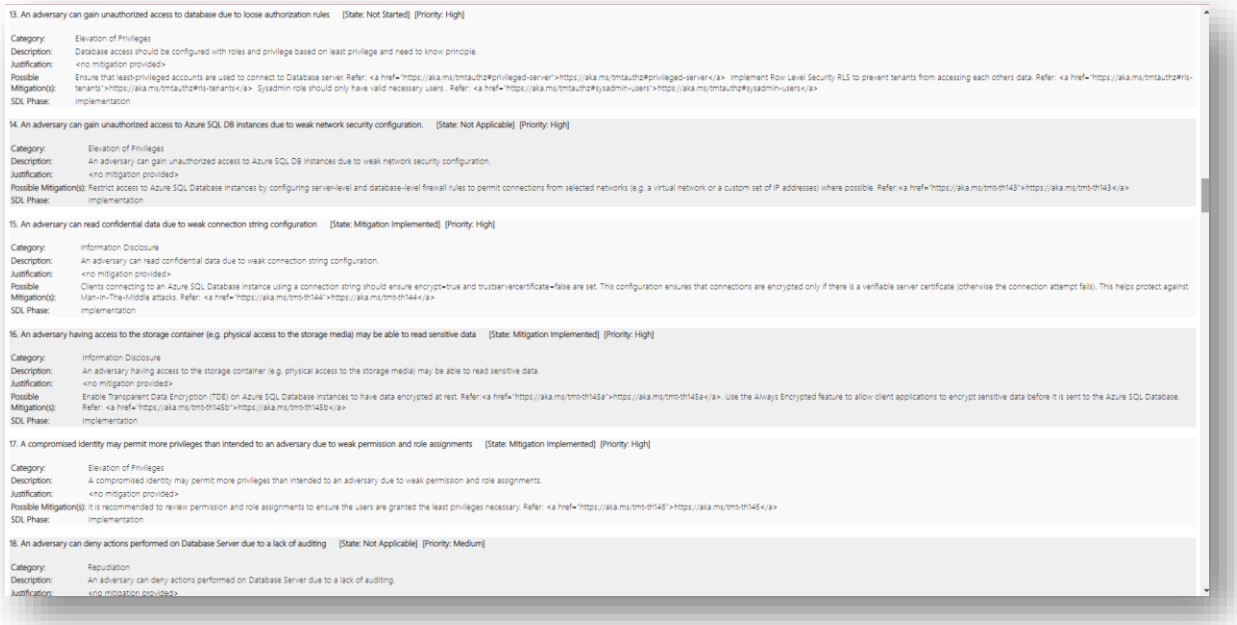
11. An adversary may tamper Proxy Server and extract cryptographic key material from it [State: Not Started] [Priority: High]

Category: Tampering
Description: An adversary may partially or wholly replace the software running on App Server, potentially allowing the replaced software to leverage the genuine identity of the device if the key material or the cryptographic facilities holding key materials were available to the illicit program. For example an attacker may leverage extracted key material to intercept and suppress data from the device on the communication path and replace it with false data that is authenticated with the stolen key material.
Justification: <no mitigation provided>
Possible Mitigation(s): Store Cryptographic Keys securely on IoT Device. Refer: https://aka.ms/tmtcrypto#keys-iot
SDL Phase: Design

12. An adversary may tamper the OS of a device and launch offline attacks [State: Not Started] [Priority: High]

Category: Tampering
Description: An adversary may launch offline attacks made by disabling or circumventing the installed operating system, or made by physically separating the storage media from the device in order to attack the data separately.
Justification: <no mitigation provided>
Possible Mitigation(s): Encrypt OS and additional partitions of IoT Device with Bitlocker. Refer: https://aka.ms/tmtconfigmgmt#partition-iot
SDL Phase: Design

Interaction: HTTPS Request



13. An adversary can gain unauthorized access to database due to loose authorization rules [State: Not Started] [Priority: High]

Category: Elevation of Privileges
Description: Database access should be configured with roles and privilege based on least privilege and need to know principle.
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that least-privileged accounts are used to connect to Database server. Refer: https://aka.ms/tmtauthz#privileged-server Implement Row Level Security RLS to prevent tenants from accessing each others data. Refer: https://aka.ms/tmtauthz#rls-tenants Sysadmin role should only have valid necessary users . Refer: https://aka.ms/tmtauthz#sysadmin-users
SDL Phase: Implementation

14. An adversary can gain unauthorized access to Azure SQL DB instances due to weak network security configuration. [State: Not Applicable] [Priority: High]

Category: Elevation of Privileges
Description: An adversary can gain unauthorized access to Azure SQL DB instances due to weak network security configuration.
Justification: <no mitigation provided>
Possible Mitigation(s): Restrict access to Azure SQL Database instances by configuring server-level and database-level firewall rules to permit connections from selected networks (e.g. a virtual network or a custom set of IP addresses) where possible. Refer:https://aka.ms/tmt-th143
SDL Phase: Implementation

15. An adversary can read confidential data due to weak connection string configuration [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure
Description: An adversary can read confidential data due to weak connection string configuration.
Justification: <no mitigation provided>
Possible Mitigation(s): Clients connecting to an Azure SQL Database instance using a connection string should ensure encrypt=true and trustservercertificate=false are set. This configuration ensures that connections are encrypted only if there is a verifiable server certificate (otherwise the connection attempt fails). This helps protect against Man-in-The-Middle attacks. Refer: https://aka.ms/tmt-th144
SDL Phase: Implementation

16. An adversary having access to the storage container (e.g. physical access to the storage media) may be able to read sensitive data [State: Mitigation Implemented] [Priority: High]

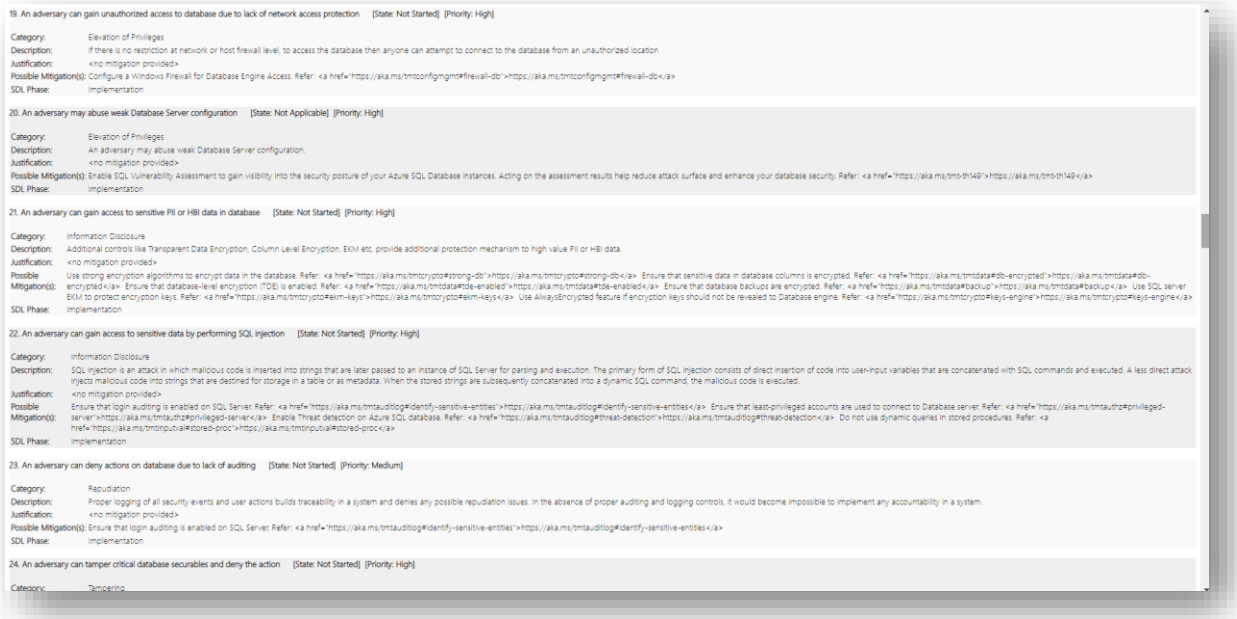
Category: Information Disclosure
Description: An adversary having access to the storage container (e.g. physical access to the storage media) may be able to read sensitive data.
Justification: <no mitigation provided>
Possible Mitigation(s): Enable Transparent Data Encryption (TDE) on Azure SQL Database instances to have data encrypted at rest. Refer:https://aka.ms/tmt-th145a. Use the Always Encrypted feature to allow client applications to encrypt sensitive data before it is sent to the Azure SQL Database. Refer: https://aka.ms/tmt-th145b
SDL Phase: Implementation

17. A compromised identity may permit more privileges than intended to an adversary due to weak permission and role assignments [State: Mitigation Implemented] [Priority: High]

Category: Elevation of Privileges
Description: A compromised identity may permit more privileges than intended to an adversary due to weak permission and role assignments.
Justification: <no mitigation provided>
Possible Mitigation(s): It is recommended to review permission and role assignments to ensure the users are granted the least privileges necessary. Refer: https://aka.ms/tmt-th146
SDL Phase: Implementation

18. An adversary can deny actions performed on Database Server due to a lack of auditing [State: Not Applicable] [Priority: Medium]

Category: Repudiation
Description: An adversary can deny actions performed on Database Server due to a lack of auditing.
Justification: <no mitigation provided>



19. An adversary can gain unauthorized access to database due to lack of network access protection [State: Not Started] [Priority: High]

Category: Elevation of Privileges
Description: If there is no restriction at network or host firewall level, to access the database then anyone can attempt to connect to the database from an unauthorized location
Justification: <no mitigation provided>
Possible Mitigation(s): Configure a Windows Firewall for Database Engine Access. Refer: https://aka.ms/tmtconfigmgmt#firewall-db
SDL Phase: Implementation

20. An adversary may abuse weak Database Server configuration [State: Not Applicable] [Priority: High]

Category: Elevation of Privileges
Description: An adversary may abuse weak Database Server configuration.
Justification: <no mitigation provided>
Possible Mitigation(s): Enable SQL Vulnerability Assessment to gain visibility into the security posture of your Azure SQL Database instances. Acting on the assessment results help reduce attack surface and enhance your database security. Refer: https://aka.ms/tmt-th149
SDL Phase: Implementation

21. An adversary can gain access to sensitive PII or HBI data in database [State: Not Started] [Priority: High]

Category: Information Disclosure
Description: Additional controls like Transparent Data Encryption, Column Level Encryption, EKM etc. provide additional protection mechanism to high value PII or HBI data.
Justification: <no mitigation provided>
Possible Mitigation(s): Use strong encryption algorithms to encrypt data in the database. Refer: https://aka.ms/tmtcrypto#strong-db Ensure that sensitive data in database columns is encrypted. Refer: https://aka.ms/tmtdata#db-encrypted Ensure that database-level encryption (TDE) is enabled. Refer: https://aka.ms/tmtdata#tde-enabled Ensure that database backups are encrypted. Refer: https://aka.ms/tmtdata#backup Use SQL server EKM to protect encryption keys. Refer: https://aka.ms/tmtcrypto#ekm-keys Use AlwaysEncrypted feature if encryption keys should not be revealed to Database engine. Refer: https://aka.ms/tmtcrypto#keys-engine
SDL Phase: Implementation

22. An adversary can gain access to sensitive data by performing SQL injection [State: Not Started] [Priority: High]

Category: Information Disclosure
Description: SQL injection is an attack in which malicious code is inserted into strings that are later passed to an instance of SQL Server for parsing and execution. The primary form of SQL injection consists of direct insertion of code into user-input variables that are concatenated with SQL commands and executed. A less direct attack injects malicious code into strings that are destined for storage in a table or as metadata. When the stored strings are subsequently concatenated into a dynamic SQL command, the malicious code is executed.
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that login auditing is enabled on SQL Server. Refer: https://aka.ms/tmtauditlog#identify-sensitive-entities Ensure that least-privileged accounts are used to connect to Database server. Refer: https://aka.ms/tmtauthz#privileged-server Enable Threat detection on Azure SQL database. Refer: https://aka.ms/tmtauditlog#threat-detection Do not use dynamic queries in stored procedures. Refer: https://aka.ms/tmtinputval#stored-proc
SDL Phase: Implementation

23. An adversary can deny actions on database due to lack of auditing [State: Not Started] [Priority: Medium]

Category: Repudiation
Description: Proper logging of all security events and user actions builds traceability in a system and denies any possible repudiation issues. In the absence of proper auditing and logging controls, it would become impossible to implement any accountability in a system.
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that login auditing is enabled on SQL Server. Refer: https://aka.ms/tmtauditlog#identify-sensitive-entities
SDL Phase: Implementation

24. An adversary can tamper critical database securables and deny the action [State: Not Started] [Priority: High]

Category: Tampering

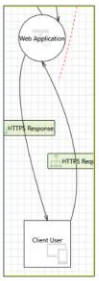
24. An adversary can tamper critical database securables and deny the action [State: Not Started] [Priority: High]

Category: Tampering
Description: An adversary can tamper critical database securables and deny the action
Justification: <no mitigation provided>
Possible Mitigation(s): Add digital signature to critical database securables. Refer: https://aka.ms/tmtcrypto#securables-db
SDL Phase: Design

25. An adversary may leverage the lack of monitoring systems and trigger anomalous traffic to database [State: Not Started] [Priority: High]

Category: Tampering
Description: An adversary may leverage the lack of intrusion detection and prevention of anomalous database activities and trigger anomalous traffic to database
Justification: <no mitigation provided>
Possible Mitigation(s): Enable Threat detection on Azure SQL database. Refer: https://aka.ms/tmtauditlog#threat-detection
SDL Phase: Design

Interaction: HTTPS Request



26. An adversary may jail break into a mobile device and gain elevated privileges [State: Mitigation Implemented] [Priority: High]

Category: Elevation of Privileges
Description: An adversary may jail break into a mobile device and gain elevated privileges
Justification: <no mitigation provided>
Possible Mitigation(s): Implement implicit jailbreak or rooting detection. Refer: https://aka.ms/tmtauthz#rooting-detection

26. An adversary may jail break into a mobile device and gain elevated privileges [State: Mitigation Implemented] [Priority: High]

Category: Elevation of Privileges
Description: An adversary may jail break into a mobile device and gain elevated privileges
Justification: <no mitigation provided>
Possible Mitigation(s): Implement implicit jailbreak or rooting detection. Refer: https://aka.ms/tmtauthz#rooting-detection
SDL Phase: Design

27. An adversary can reverse weakly encrypted or hashed content [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure
Description: An adversary can reverse weakly encrypted or hashed content
Justification: <no mitigation provided>
Possible Mitigation(s): Do not expose security details in error messages. Refer: https://aka.ms/tmtimgmt#messages Implement Default error handling page. Refer: https://aka.ms/tmtimgmt#default Set Deployment Method to Retail in IIS. Refer: https://aka.ms/tmtimgmt#deployment. Use only approved symmetric block ciphers and key lengths. Refer: https://aka.ms/tmtcrypto#cipher-length Use approved block cipher modes and initialization vectors for symmetric ciphers. Refer: https://aka.ms/tmtcrypto#vector-ciphers Use approved asymmetric algorithms, key lengths, and padding. Refer: https://aka.ms/tmtcrypto#padding Use approved random number generators. Refer: https://aka.ms/tmtcrypto#number Do not use symmetric stream ciphers. Refer: https://aka.ms/tmtcrypto#stream-ciphers Use approved MAC/HMAC/keyed hash algorithms. Refer: https://aka.ms/tmtcrypto#mac-hash Use only approved cryptographic hash functions. Refer: https://aka.ms/tmtcrypto#hash-functions Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections. Refer: https://aka.ms/tmtcommsec#x509-ssltls
SDL Phase: Implementation

28. An adversary may gain access to sensitive data from log files [State: Not Started] [Priority: High]

Category: Information Disclosure
Description: An adversary may gain access to sensitive data from log files
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that the application does not log sensitive user data. Refer: https://aka.ms/tmtauditlog#log-sensitive-data Ensure that Audit and Log Files have Restricted Access. Refer: https://aka.ms/tmtauditlog#log-restricted-access
SDL Phase: Implementation

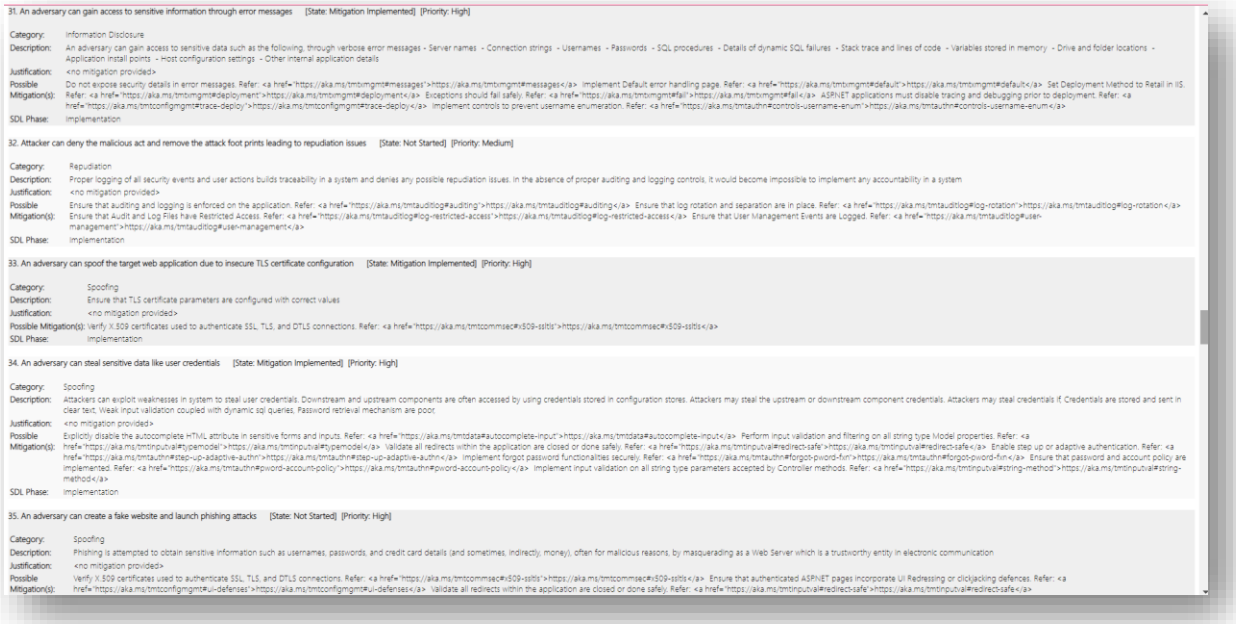
29. An adversary can gain access to sensitive data by sniffing traffic from Mobile client [State: Not Started] [Priority: High]

Category: Information Disclosure
Description: An adversary can gain access to sensitive data by sniffing traffic from Mobile client
Justification: <no mitigation provided>
Possible Mitigation(s): Implement Certificate Pinning. Refer: https://aka.ms/tmtcommsec#cert-pinning
SDL Phase: Implementation

30. An adversary can gain sensitive data from mobile device [State: Not Started] [Priority: High]

Category: Information Disclosure
Description: If application saves sensitive PII or HBI data on phone SD card or local storage, then it ay get stolen.
Justification: <no mitigation provided>
Possible Mitigation(s): Encrypt sensitive or PII data written to phones local storage. Refer: https://aka.ms/tmtdata#pii-phones
SDL Phase: Implementation

31. An adversary can gain access to sensitive information through error messages [State: Mitigation Implemented] [Priority: High]



31. An adversary can gain access to sensitive information through error messages [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure
Description: An adversary can gain access to sensitive data such as the following, through verbose error messages - Server names - Connection strings - Usernames - Passwords - SQL procedures - Details of dynamic SQL failures - Stack trace and lines of code - Variables stored in memory - Drive and folder locations - Application install points - Host configuration settings - Other internal application details
Justification: <no mitigation provided>
Possible Mitigation(s): Do not expose security details in error messages. Refer: https://aka.ms/tmtimgmt#messages Implement Default error handling page. Refer: https://aka.ms/tmtimgmt#default Set Deployment Method to Retail in IIS. Refer: https://aka.ms/tmtimgmt#deployment Exceptions should fail safely. Refer: https://aka.ms/tmtimgmt#fail ASP.NET applications must disable tracing and debugging prior to deployment. Refer: https://aka.ms/tmtconfigmgmt#trace-deploy Implement controls to prevent username enumeration. Refer: https://aka.ms/tmtauthn#controls-username-enum
SDL Phase: Implementation

32. Attacker can deny the malicious act and remove the attack foot prints leading to repudiation issues [State: Not Started] [Priority: Medium]

Category: Repudiation
Description: Proper logging of all security events and user actions builds traceability in a system and denies any possible repudiation issues. In the absence of proper auditing and logging controls, it would become impossible to implement any accountability in a system
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that auditing and logging is enforced on the application. Refer: https://aka.ms/tmtauditlog#auditing Ensure that log rotation and separation are in place. Refer: https://aka.ms/tmtauditlog#log-rotation Ensure that Audit and Log Files have Restricted Access. Refer: https://aka.ms/tmtauditlog#log-restricted-access Ensure that User Management Events are Logged. Refer: https://aka.ms/tmtauditlog#user-management
SDL Phase: Implementation

33. An adversary can spoof the target web application due to insecure TLS certificate configuration [State: Mitigation Implemented] [Priority: High]

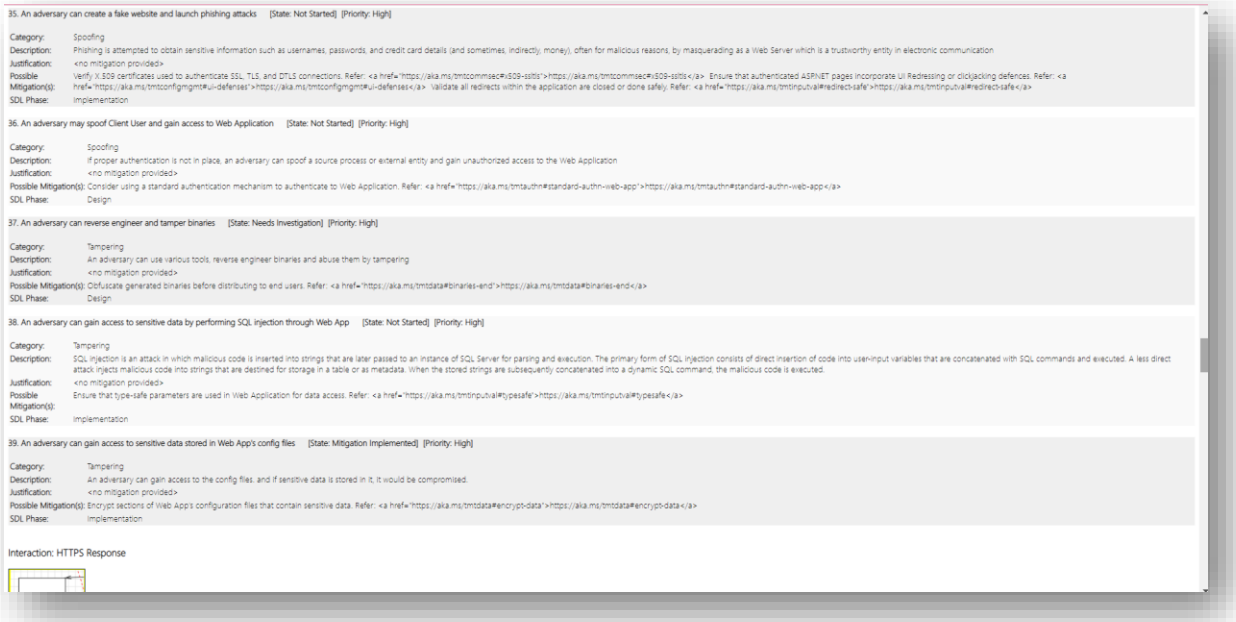
Category: Spoofing
Description: Ensure that TLS certificate parameters are configured with correct values
Justification: <no mitigation provided>
Possible Mitigation(s): Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections. Refer: https://aka.ms/tmtcommsec#x509-ssltls
SDL Phase: Implementation

34. An adversary can steal sensitive data like user credentials [State: Mitigation Implemented] [Priority: High]

Category: Spoofing
Description: Attackers can exploit weaknesses in system to steal user credentials. Downstream and upstream components are often accessed by using credentials stored in configuration stores. Attackers may steal the upstream or downstream component credentials. Attackers may steal credentials if, Credentials are stored and sent in clear text, Weak input validation coupled with dynamic sql queries, Password retrieval mechanism are poor
Justification: <no mitigation provided>
Possible Mitigation(s): Explicitly disable the autocomplete HTML attribute in sensitive forms and inputs. Refer: https://aka.ms/tmtdata#autocomplete-input Perform input validation and filtering on all string type Model properties. Refer: https://aka.ms/tmtinputval#typemodel Validate all redirects within the application are closed or done safely. Refer: https://aka.ms/tmtinputval#redirect-safe Enable step up or adaptive authentication. Refer: https://aka.ms/tmtauthn#step-up-adaptive-authn Implement forgot password functionalities securely. Refer: https://aka.ms/tmtauthn#forgot-pword-fun Ensure that password and account policy are implemented. Refer: https://aka.ms/tmtauthn#pword-account-policy Implement input validation on all string type parameters accepted by Controller methods. Refer: https://aka.ms/tmtinputval#string-method
SDL Phase: Implementation

35. An adversary can create a fake website and launch phishing attacks [State: Not Started] [Priority: High]

Category: Spoofing
Description: Phishing is attempted to obtain sensitive information such as usernames, passwords, and credit card details (and sometimes, indirectly, money), often for malicious reasons, by masquerading as a Web Server which is a trustworthy entity in electronic communication
Justification: <no mitigation provided>
Possible Mitigation(s): Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections. Refer: https://aka.ms/tmtcommsec#x509-ssltls Ensure that authenticated ASP.NET pages incorporate UI Redressing or clickjacking defences. Refer: https://aka.ms/tmtconfigmgmt#ui-defenses Validate all redirects within the application are closed or done safely. Refer: https://aka.ms/tmtinputval#redirect-safe



35. An adversary can create a fake website and launch phishing attacks [State: Not Started] [Priority: High]

Category: Spoofing
Description: Phishing is attempted to obtain sensitive information such as usernames, passwords, and credit card details (and sometimes, indirectly, money), often for malicious reasons, by masquerading as a Web Server which is a trustworthy entity in electronic communication
Justification: <no mitigation provided>
Possible Mitigation(s): Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections. Refer: https://aka.ms/tmtcommsec#x509-ssltls Ensure that authenticated ASP.NET pages incorporate UI Redressing or clickjacking defences. Refer: https://aka.ms/tmtconfigmgmt#ui-defenses Validate all redirects within the application are closed or done safely. Refer: https://aka.ms/tmtinputval#redirect-safe
SDL Phase: Implementation

36. An adversary may spoof Client User and gain access to Web Application [State: Not Started] [Priority: High]

Category: Spoofing
Description: If proper authentication is not in place, an adversary can spoof a source process or external entity and gain unauthorized access to the Web Application
Justification: <no mitigation provided>
Possible Mitigation(s): Consider using a standard authentication mechanism to authenticate to Web Application. Refer: https://aka.ms/tmtauthn#standard-authn-web-app
SDL Phase: Design

37. An adversary can reverse engineer and tamper binaries [State: Needs Investigation] [Priority: High]

Category: Tampering
Description: An adversary can use various tools, reverse engineer binaries and abuse them by tampering
Justification: <no mitigation provided>
Possible Mitigation(s): Obfuscate generated binaries before distributing to end users. Refer: https://aka.ms/tmtdata#binaries-end
SDL Phase: Design

38. An adversary can gain access to sensitive data by performing SQL injection through Web App [State: Not Started] [Priority: High]

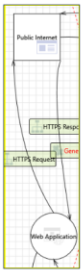
Category: Tampering
Description: SQL injection is an attack in which malicious code is inserted into strings that are later passed to an instance of SQL Server for parsing and execution. The primary form of SQL injection consists of direct insertion of code into user-input variables that are concatenated with SQL commands and executed. A less direct attack injects malicious code into strings that are destined for storage in a table or as metadata. When the stored strings are subsequently concatenated into a dynamic SQL command, the malicious code is executed.
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that type-safe parameters are used in Web Application for data access. Refer: https://aka.ms/tmtinputval#typesafe
SDL Phase: Implementation

39. An adversary can gain access to sensitive data stored in Web App's config files [State: Mitigation Implemented] [Priority: High]

Category: Tampering
Description: An adversary can gain access to the config files, and if sensitive data is stored in it, it would be compromised.
Justification: <no mitigation provided>
Possible Mitigation(s): Encrypt sections of Web App's configuration files that contain sensitive data. Refer: https://aka.ms/tmtdata#encrypt-data
SDL Phase: Implementation

Interaction: HTTPS Response

Interaction: HTTPS Response



40. An adversary can gain access to sensitive data stored in Web App's config files [State: Not Started] [Priority: High]

Category: Tampering
Description: An adversary can gain access to the config files, and if sensitive data is stored in it, it would be compromised.
Justification: <no mitigation provided>
Possible Mitigation(s): Encrypt sections of Web App's configuration files that contain sensitive data. Refer: https://aka.ms/tmtdata#encrypt-data
SDL Phase: Implementation

41. An adversary can gain access to sensitive data by performing SQL injection through Web App [State: Mitigation Implemented] [Priority: High]

Category: Tampering
Description: SQL injection is an attack in which malicious code is inserted into strings that are later passed to an instance of SQL Server for parsing and execution. The primary form of SQL injection consists of direct insertion of code into user-input variables that are concatenated with SQL commands and executed. A less direct attack injects malicious code into strings that are destined for storage in a table or as metadata. When the stored strings are subsequently concatenated into a dynamic SQL command, the malicious code is executed.
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that type-safe parameters are used in Web Application for data access. Refer: https://aka.ms/tmtinputval#typesafe
SDL Phase: Implementation

42. An attacker steals messages off the network and replays them in order to steal a user's session [State: Not Started] [Priority: High]

Category: Tampering
Description: An attacker steals messages off the network and replays them in order to steal a user's session
Justification: <no mitigation provided>

42. An attacker steals messages off the network and replays them in order to steal a user's session [State: Not Started] [Priority: High]

Category: Tampering
Description: An attacker steals messages off the network and replays them in order to steal a user's session
Justification: <no mitigation provided>
Possible Mitigation(s):
SDL Phase: Implementation

43. An adversary can deface the target web application by injecting malicious code or uploading dangerous files [State: Not Started] [Priority: High]

Category: Tampering
Description: Website defacement is an attack on a website where the attacker changes the visual appearance of the site or a webpage.
Justification: <no mitigation provided>
Possible Mitigation(s): Implement Content Security Policy (CSP), and disable inline javascript. Refer: https://aka.ms/tmtconfigmgmt#csp-js Enable browser's XSS filter Refer: https://aka.ms/tmtconfigmgmt#xss-filter Access third party javascripts from trusted sources only. Refer: https://aka.ms/tmtconfigmgmt#js-trusted Enable ValidateRequest attribute on ASP.NET Pages. Refer: https://aka.ms/tmtconfigmgmt#validate-aspnet Ensure that each page that could contain user controllable content opts out of automatic MIME sniffing . Refer: https://aka.ms/tmtinputval#sub-sniffing Use locally-hosted latest versions of JavaScript libraries . Refer: https://aka.ms/tmtconfigmgmt#local-js Ensure appropriate controls are in place when accepting files from users. Refer: https://aka.ms/tmtinputval#controls-users Disable automatic MIME sniffing. Refer: https://aka.ms/tmtconfigmgmt#mime-sniff Encode untrusted web output prior to rendering. Refer: https://aka.ms/tmtinputval#rendering Perform input validation and filtering on all string type Model properties. Refer: https://aka.ms/tmtinputval#typemodel Ensure that the system has inbuilt defences against misuse. Refer: https://aka.ms/tmtauditlog#inbuilt-defenses Enable HTTP Strict Transport Security (HSTS). Refer: https://aka.ms/tmtcommsec#http-hsts Implement input validation on all string type parameters accepted by Controller methods. Refer: https://aka.ms/tmtinputval#string-method Avoid using Html.Raw in Razor views. Refer: https://aka.ms/tmtinputval#html-razor Sanitization should be applied on form fields that accept all characters e.g. rich text editor . Refer: https://aka.ms/tmtinputval#richtext Do not assign DOM elements to sinks that do not have inbuilt encoding . Refer: https://aka.ms/tmtinputval#inbuilt-encode
SDL Phase: Implementation

44. An adversary may spoof Public Internet and gain access to Web Application [State: Not Started] [Priority: High]

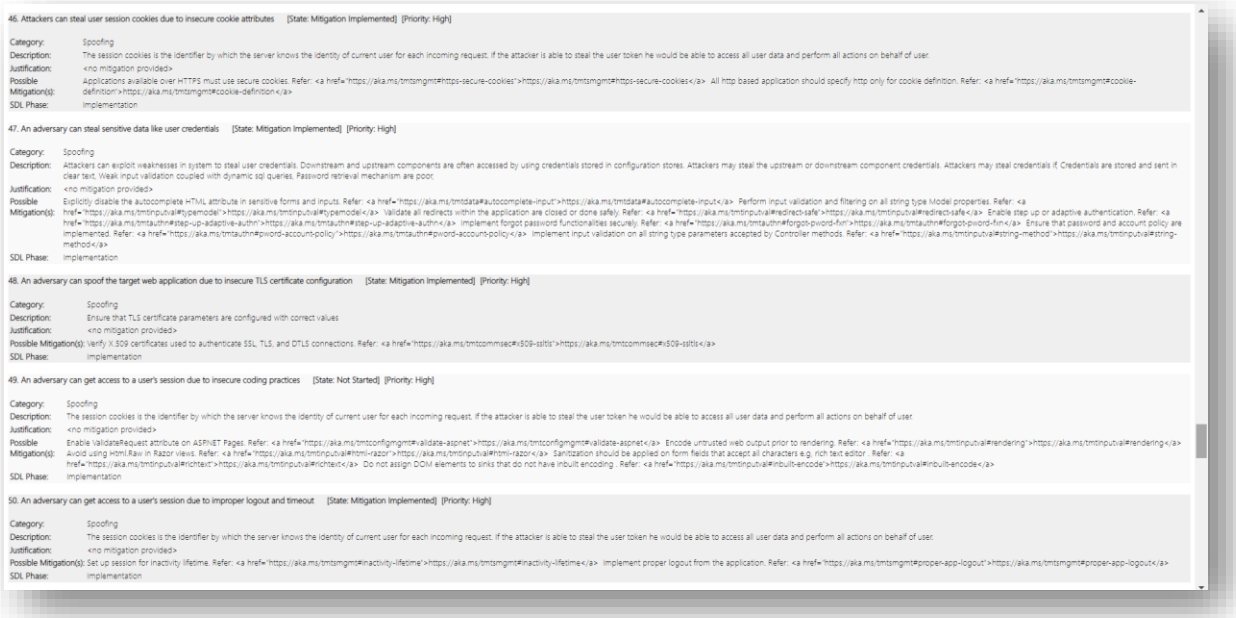
Category: Spoofing
Description: If proper authentication is not in place, an adversary can spoof a source process or external entity and gain unauthorized access to the Web Application
Justification: <no mitigation provided>
Possible Mitigation(s): Consider using a standard authentication mechanism to authenticate to Web Application. Refer: https://aka.ms/tmtauthn#standard-authn-web-app
SDL Phase: Design

45. An adversary can create a fake website and launch phishing attacks [State: Mitigation Implemented] [Priority: High]

Category: Spoofing
Description: Phishing is attempted to obtain sensitive information such as usernames, passwords, and credit card details (and sometimes, indirectly, money), often for malicious reasons, by masquerading as a Web Server which is a trustworthy entity in electronic communication
Justification: <no mitigation provided>
Possible Mitigation(s): Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections. Refer: https://aka.ms/tmtcommsec#x509-ssltls Ensure that authenticated ASP.NET pages incorporate UI Redressing or clickjacking defences. Refer: https://aka.ms/tmtconfigmgmt#ui-defenses Validate all redirects within the application are closed or done safely. Refer: https://aka.ms/tmtinputval#redirect-safe
SDL Phase: Implementation

46. Attackers can steal user session cookies due to insecure cookie attributes [State: Mitigation Implemented] [Priority: High]

Category: Spoofing
Description: The session cookies is the identifier by which the server knows the identity of current user for each incoming request. If the attacker is able to steal the user token he would be able to access all user data and perform all actions on behalf of user.
Justification: <no mitigation provided>
Possible Mitigation(s): Applications available over HTTPS must use secure cookies. Refer: https://aka.ms/tmtsmgmt#https-secure-cookies All http based application should specify http only for cookie definition. Refer: https://aka.ms/tmtsmgmt#cookie-definition



46. Attackers can steal user session cookies due to insecure cookie attributes [State: Mitigation Implemented] [Priority: High]

Category: Spoofing
Description: The session cookies is the identifier by which the server knows the identity of current user for each incoming request. If the attacker is able to steal the user token he would be able to access all user data and perform all actions on behalf of user.
Justification: <no mitigation provided>
Possible Mitigation(s): Applications available over HTTPS must use secure cookies. Refer: https://aka.ms/tmtsmgmt#https-secure-cookies All http based application should specify http only for cookie definition. Refer: https://aka.ms/tmtsmgmt#cookie-definition
SDL Phase: Implementation

47. An adversary can steal sensitive data like user credentials [State: Mitigation Implemented] [Priority: High]

Category: Spoofing
Description: Attackers can exploit weaknesses in system to steal user credentials. Downstream and upstream components are often accessed by using credentials stored in configuration stores. Attackers may steal the upstream or downstream component credentials. Attackers may steal credentials if: Credentials are stored and sent in clear text. Weak input validation coupled with dynamic sql queries. Password retrieval mechanism are poor.
Justification: <no mitigation provided>
Possible Mitigation(s): Explicitly disable the autocomplete HTML attribute in sensitive forms and inputs. Refer: https://aka.ms/tmtdata#autocomplete-input Perform input validation and filtering on all string type Model properties. Refer: https://aka.ms/tmtinputval#typemodel Validate all redirects within the application are closed or done safely. Refer: https://aka.ms/tmtinputval#redirect-safe Enable step up or adaptive authentication. Refer: https://aka.ms/tmtauthn#step-up-adaptive-authn Implement forgot password functionalities securely. Refer: https://aka.ms/tmtauthn#forgot-pword-fxn Ensure that password and account policy are implemented. Refer: https://aka.ms/tmtauthn#pword-account-policy Implement input validation on all string type parameters accepted by Controller methods. Refer: https://aka.ms/tmtinputval#string-method
SDL Phase: Implementation

48. An adversary can spoof the target web application due to insecure TLS certificate configuration [State: Mitigation Implemented] [Priority: High]

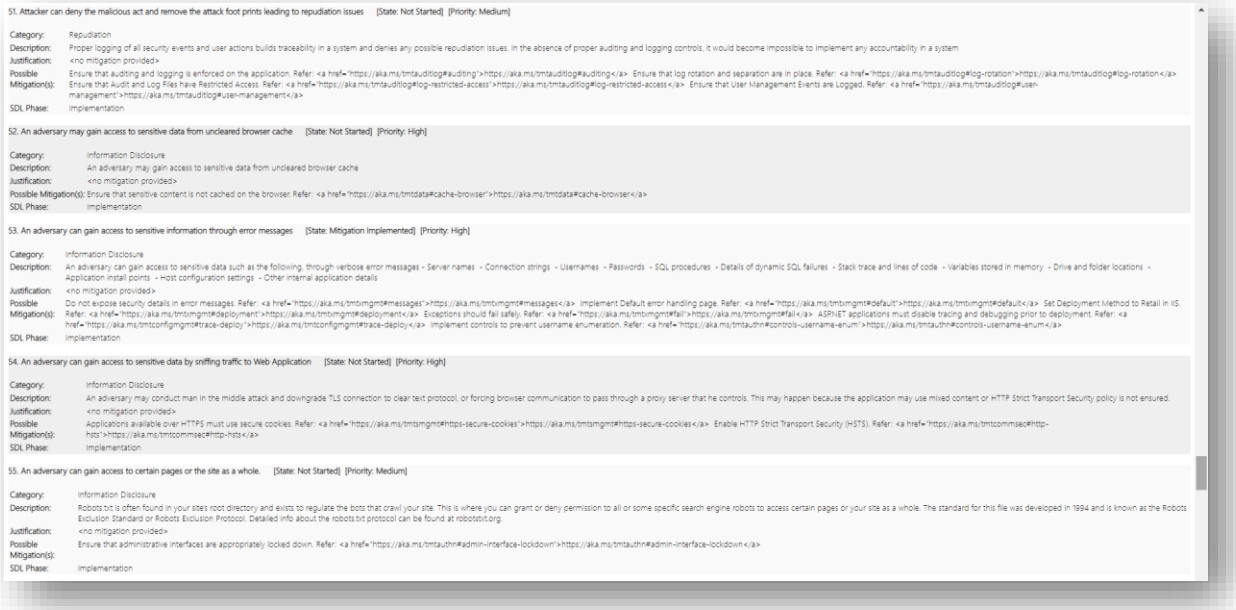
Category: Spoofing
Description: Ensure that TLS certificate parameters are configured with correct values
Justification: <no mitigation provided>
Possible Mitigation(s): Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections. Refer: https://aka.ms/tmtcommsec#x509-ssltls
SDL Phase: Implementation

49. An adversary can get access to a user's session due to insecure coding practices [State: Not Started] [Priority: High]

Category: Spoofing
Description: The session cookies is the identifier by which the server knows the identity of current user for each incoming request. If the attacker is able to steal the user token he would be able to access all user data and perform all actions on behalf of user.
Justification: <no mitigation provided>
Possible Mitigation(s): Enable ValidateRequest attribute on ASPNET Pages. Refer: https://aka.ms/tmtconfigmgmt#validate-aspnet Encode untrusted web output prior to rendering. Refer: https://aka.ms/tmtinputval#rendering Avoid using Html.Raw in Razor views. Refer: https://aka.ms/tmtinputval#html-razor Sanitization should be applied on form fields that accept all characters e.g. rich text editor . Refer: https://aka.ms/tmtinputval#richtext Do not assign DOM elements to sinks that do not have inbuilt encoding . Refer: https://aka.ms/tmtinputval#inbuilt-encode
SDL Phase: Implementation

50. An adversary can get access to a user's session due to improper logout and timeout [State: Mitigation Implemented] [Priority: High]

Category: Spoofing
Description: The session cookies is the identifier by which the server knows the identity of current user for each incoming request. If the attacker is able to steal the user token he would be able to access all user data and perform all actions on behalf of user.
Justification: <no mitigation provided>
Possible Mitigation(s): Set up session for inactivity lifetime. Refer: https://aka.ms/tmtsmgmt#inactivity-lifetime Implement proper logout from the application. Refer: https://aka.ms/tmtsmgmt#proper-app-logout
SDL Phase: Implementation



51. Attacker can deny the malicious act and remove the attack foot prints leading to repudiation issues [State: Not Started] [Priority: Medium]

Category: Repudiation
Description: Proper logging of all security events and user actions builds traceability in a system and denies any possible repudiation issues. In the absence of proper auditing and logging controls, it would become impossible to implement any accountability in a system
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that auditing and logging is enforced on the application. Refer: https://aka.ms/tmtauditlog#auditing Ensure that log rotation and separation are in place. Refer: https://aka.ms/tmtauditlog#log-rotation Ensure that Audit and Log Files have Restricted Access. Refer: https://aka.ms/tmtauditlog#log-restricted-access Ensure that User Management Events are Logged. Refer: https://aka.ms/tmtauditlog#user-management
SDL Phase: Implementation

52. An adversary may gain access to sensitive data from uncleared browser cache [State: Not Started] [Priority: High]

Category: Information Disclosure
Description: An adversary may gain access to sensitive data from uncleared browser cache
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that sensitive content is not cached on the browser. Refer: https://aka.ms/tmtdata#cache-browser
SDL Phase: Implementation

53. An adversary can gain access to sensitive information through error messages [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure
Description: An adversary can gain access to sensitive data such as the following, through verbose error messages - Server names - Connection strings - Usernames - Passwords - SQL procedures - Details of dynamic SQL failures - Stack trace and lines of code - Variables stored in memory - Drive and folder locations - Application install points - Host configuration settings - Other internal application details
Justification: <no mitigation provided>
Possible Mitigation(s): Do not expose security details in error messages. Refer: https://aka.ms/tmtvmgmt#messages Implement Default error handling page. Refer: https://aka.ms/tmtvmgmt#default Set Deployment Method to Retail in IIS. Refer: https://aka.ms/tmtvmgmt#deployment Exceptions should fail safely. Refer: https://aka.ms/tmtvmgmt#fail ASPNET applications must disable tracing and debugging prior to deployment. Refer: https://aka.ms/tmtconfigmgmt#trace-deploy Implement controls to prevent username enumeration. Refer: https://aka.ms/tmtauthn#controls-username-enum
SDL Phase: Implementation

54. An adversary can gain access to sensitive data by sniffing traffic to Web Application [State: Not Started] [Priority: High]

Category: Information Disclosure
Description: An adversary may conduct man in the middle attack and downgrade TLS connection to clear text protocol, or forcing browser communication to pass through a proxy server that he controls. This may happen because the application may use mixed content or HTTP Strict Transport Security policy is not ensured.
Justification: <no mitigation provided>
Possible Mitigation(s): Applications available over HTTPS must use secure cookies. Refer: https://aka.ms/tmtsmgmt#https-secure-cookies Enable HTTP Strict Transport Security (HSTS). Refer: https://aka.ms/tmtcommsec#http-hsts
SDL Phase: Implementation

55. An adversary can gain access to certain pages or the site as a whole. [State: Not Started] [Priority: Medium]

Category: Information Disclosure
Description: Robots.txt is often found in your sites root directory and exists to regulate the bots that crawl your site. This is where you can grant or deny permission to all or some specific search engine robots to access certain pages or your site as a whole. The standard for this file was developed in 1994 and is known as the Robots Exclusion Standard or Robots Exclusion Protocol. Detailed info about the robots.txt protocol can be found at robotstxt.org.
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that administrative interfaces are appropriately locked down. Refer: https://aka.ms/tmtauthn#admin-interface-lockdown
SDL Phase: Implementation

56. An adversary may gain access to unmasked sensitive data such as credit card numbers [State: Not Started] [Priority: High]

Category: Information Disclosure
Description: An adversary may gain access to unmasked sensitive data such as credit card numbers
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that sensitive data displayed on the user screen is masked. Refer: https://aka.ms/tmtdata#data-mask
SDL Phase: Implementation

57. An adversary may gain access to sensitive data from log files [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure
Description: An adversary may gain access to sensitive data from log files
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that the application does not log sensitive user data. Refer: https://aka.ms/tmtauditlog#log-sensitive-data Ensure that Audit and Log Files have Restricted Access. Refer: https://aka.ms/tmtauditlog#log-restricted-access
SDL Phase: Implementation

58. An adversary can reverse weakly encrypted or hashed content [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure
Description: An adversary can reverse weakly encrypted or hashed content
Justification: <no mitigation provided>
Possible Mitigation(s): Do not expose security details in error messages. Refer: https://aka.ms/tmtsmgmt#messages Implement Default error handling page. Refer: https://aka.ms/tmtsmgmt#default Set Deployment Method to Retail in IIS. Refer: https://aka.ms/tmtsmgmt#deployment Use only approved symmetric block ciphers and key lengths. Refer: https://aka.ms/tmtcrypto#cipher-length Use approved block cipher modes and initialization vectors for symmetric ciphers. Refer: https://aka.ms/tmtcrypto#vector-ciphers Use approved asymmetric algorithms, key lengths, and padding. Refer: https://aka.ms/tmtcrypto#padding Use approved random number generators. Refer: https://aka.ms/tmtcrypto#numgen Do not use symmetric stream ciphers. Refer: https://aka.ms/tmtcrypto#stream-ciphers Use approved MAC/HMAC/keyed hash algorithms. Refer: https://aka.ms/tmtcrypto#mac-hash Use only approved cryptographic hash functions. Refer: https://aka.ms/tmtcrypto#hash-functions Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections. Refer: https://aka.ms/tmtcommsec#x509-ssltls
SDL Phase: Implementation

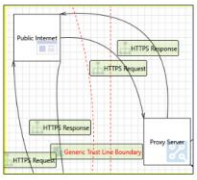
59. An adversary may bypass critical steps or perform actions on behalf of other users (victims) due to improper validation logic [State: Mitigation Implemented] [Priority: High]

Category: Elevation of Privileges
Description: Failure to restrict the privileges and access rights to the application to individuals who require the privileges or access rights may result into unauthorized use of data due to inappropriate rights settings and validation.
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that administrative interfaces are appropriately locked down. Refer: https://aka.ms/tmtauthn#admin-interface-lockdown Enforce sequential step order when processing business logic flows. Refer: https://aka.ms/tmtauthz#sequential-logic Ensure that proper authorization is in place and principle of least privileges is followed. Refer: https://aka.ms/tmtauthz#principle-least-privilege Business logic and resource access authorization decisions should not be based on incoming request parameters. Refer: https://aka.ms/tmtauthz#logic-request-parameters Ensure that content and resources are not enumerable or accessible via forceful browsing. Refer: https://aka.ms/tmtauthz#enumerable-browsing
SDL Phase: Implementation

60. An adversary can perform action on behalf of other user due to lack of controls against cross domain requests [State: Not Started] [Priority: High]

Category: Denial of Service
Description: Failure to restrict requests originating from third party domains may result in unauthorized actions or access of data
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that authenticated ASP.NET pages incorporate UI Redressing or clickjacking defences. Refer: https://aka.ms/tmtconfigmgmt#ui-defenses Ensure that only trusted origins are allowed if CORS is enabled on ASP.NET Web Applications. Refer: https://aka.ms/tmtconfigmgmt#cors-aspnet Mitigate against Cross-Site Request Forgery (CSRF) attacks on ASP.NET web pages. Refer: https://aka.ms/tmtsmgmt#csrf-asp

Interaction: HTTPS Response



61. An adversary may gain unauthorized access to privileged features on Proxy Server [State: Mitigation Implemented] [Priority: High]

Category: Elevation of Privileges
Description: An adversary may get access to admin interface or privileged services like WiFi, SSH, File shares, FTP etc., on a device
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that all admin interfaces are secured with strong credentials. Refer: https://aka.ms/tmtconfigmgmt#admin-strong
SDL Phase: Implementation

62. An adversary may exploit unused services or features in Public Internet [State: Not Started] [Priority: High]

Category: Elevation of Privileges
Description: An adversary may use unused features or services on Public Internet such as UI, USB port etc. Unused features increase the attack surface and serve as additional entry points for the adversary
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that only the minimum services/features are enabled on devices. Refer: https://aka.ms/tmtconfigmgmt#min-enable
SDL Phase: Implementation

63. An adversary may exploit known vulnerabilities in unpatched devices [State: Not Started] [Priority: High]

Category: Tampering
Description: An adversary may leverage known vulnerabilities and exploit a device if the firmware of the device is not updated
Justification: <no mitigation provided>
Possible Mitigation(s): Ensure that the Cloud Gateway implements a process to keep the connected devices firmware up to date. Refer: https://aka.ms/tmtconfigmgmt#cloud-firmware
SDL Phase: Design

64. An adversary may tamper Proxy Server and extract cryptographic key material from it [State: Not Started] [Priority: High]

Category: Tampering
Description: An adversary may partially or wholly replace the software running on Public Internet, potentially allowing the replaced software to leverage the genuine identity of the device if the key material or the cryptographic facilities holding key materials were available to the illicit program. For example an attacker may leverage extracted key material to intercept and suppress data from the device on the communication path and replace it with false data that is authenticated with the stolen key material.
Justification: <no mitigation provided>
Possible Store Cryptographic Keys securely on IoT Device. Refer: https://aka.ms/tmtcrypto#keys-iot

Threats, Mitigations Suggested and Mitigations Implemented

01. User authentication

- Auth Token expire

- Tokens expire

02. Access control -

- User privileges

03. Integrity -

- Validating the user inputs to prevent xss

- Using middleware validating user inputs

04. Confidentiality -

- Message encrypted

- Password hashing

05. Message authentication -

- Handling the error messages

- Handling try catch method

Testing Plan and Results

Test Plan

- Manual Testing

Manual Testing is a type of software testing in which test cases are executed manually by a tester without using any automated tools. The purpose of Manual Testing is to identify the bugs, issues, and defects in the software application. Manual software testing is the most primitive technique of all testing types and it helps to find critical bugs in the software application.

  - Functional Testing – **Sivarajah. K**
  - API Testing – **Sivashankar. S**

- Burp Suite Security Testing – **Soujanyaah. K**

Burp Suite is an integrated platform/graphical tool for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.

- Password Hashing Testing – **Viknatharshan. N**

Password Hashing makes password storage and management more secure. Hashing uses a formula to transform a password into a predictable, yet encrypted form that obscures the actual password and makes it much harder for bad actors to decipher it.

Test Results

Manual Testing

Functional Testing

Test Case No	Test Case 01
Description	Login as an admin with valid credentials.
Test Steps	1. Go to the login page. 2. Enter the valid email address and password. 3. Click the “Login” button.
Expected Result	Should display the “Add New Staff” form.
Actual Result	Displayed the “Add New Staff” form.
Test Status	**Pass**
User Role	Admin

Test Case No	Test Case 02
Description	Login as an admin with invalid credentials.
Test Steps	1. Go to the login page. 2. Enter the invalid email address and password. 3. Click the “Login” button.
Expected Result	*Should display the error message “Something went wrong Server error please reload”.*

Actual Result	Displayed the error message "Something went wrong Server error please reload"**.**
Test Status	**Pass**
User Role	Admin

Test Case No	Test Case 03
Description	Check the required field validation of login form
Test Steps	1. Go to the login page. 2. Enter the email address and put the password field as empty. 3. Click the "Login" button.
Expected Result	*Should display the error message "Please fill out this field".*
Actual Result	Displayed the error message "Please fill out this field".
Test Status	**Pass**
User Role	Admin

Test Case No	Test Case 04
Description	Check the required field validation of add new staff form
Test Steps	1. Go to the login page. 2. Enter the email address and password.

	3. Click the “Login” button. (Navigate to add new staff form) 4. Click the “Create New Account” button without filling any field in “Add New Staff” form.
Expected Result	*Should display the error message “Please fill out this field”.*
Actual Result	Displayed the error message “Please fill out this field”.
Test Status	**Pass**
User Role	Admin

Test Case No	Test Case 05
Description	Create an account for worker and manager.
Test Steps	1. Go to the login page. 2. Enter the email address and password. 3. Click the “Login” button. (Navigate to add new staff form) 4. Enter the email, first name, last name, password and select the user role. 5. Click the “Create New Account” button.

Expected Result	*Should display the success message "User Added Successfully!".*
Actual Result	Displayed the success message "User Added Successfully!".
Test Status	**Pass**
User Role	Admin

Test Case No	Test Case 06
Description	Login as a manager with valid credentials.
Test Steps	1. Go to the login page. 2. Enter the valid email address and password. 3. Click the "Login" button.
Expected Result	Should display the message and upload file fields.
Actual Result	Displayed the message and upload file fields.
Test Status	**Pass**
User Role	Manager

Test Case No	Test Case 07

Description	Login as a manager with invalid credentials.
Test Steps	1. Go to the login page. 2. Enter the invalid email address and password. 3. Click the “Login” button.
Expected Result	*Should display the error message “Something went wrong Server error please reload”.*
Actual Result	Displayed the error message “Something went wrong Server error please reload”**.**
Test Status	**Pass**
User Role	Manager

Test Case No	Test Case 08
Description	Click the “Save message” button after enter the message.
Test Steps	1. Login as a manager. 2. Enter the message in message field. 3. Click the “Save message” button.
Expected Result	*Should display the success message “Message has been saved”.*
Actual Result	Displayed the success message “Message has been saved”.
Test Status	**Pass**

User Role	Manager

Test Case No	Test Case 09
Description	Click the “Upload File” button without selecting any file.
Test Steps	1. Login as a manager. 2. Click the “Upload file” button.
Expected Result	*Should display the error message “Please select a file to upload”.*
Actual Result	Displayed the error message “Please select a file to upload”.
Test Status	**Pass**
User Role	Manager

Test Case No	Test Case 10
Description	Click the “Upload File” button after select any file.
Test Steps	1. Login as a manager. 2. Choose any file. 3. Click the “Upload file” button.
Expected Result	*Should display the success message “File has been uploaded, Successfully”.*
Actual Result	Displayed the success message “File has been uploaded, Successfully”.

Test Status	**Pass**
User Role	Manager

Test Case No	Test Case 11
Description	Login as a worker with valid credentials.
Test Steps	1. Go to the login page. 2. Enter the valid email address and password. 3. Click the “Login” button.
Expected Result	Should display the message field.
Actual Result	Displayed the message field.
Test Status	**Pass**
User Role	Worker

Test Case No	Test Case 12
Description	Login as a worker with invalid credentials.
Test Steps	1. Go to the login page. 2. Enter the invalid email address and password. 3. Click the “Login” button.

Expected Result	*Should display the error message "Something went wrong Server error please reload".*
Actual Result	Displayed the error message "Something went wrong Server error please reload"**.**
Test Status	**Pass**
User Role	Worker

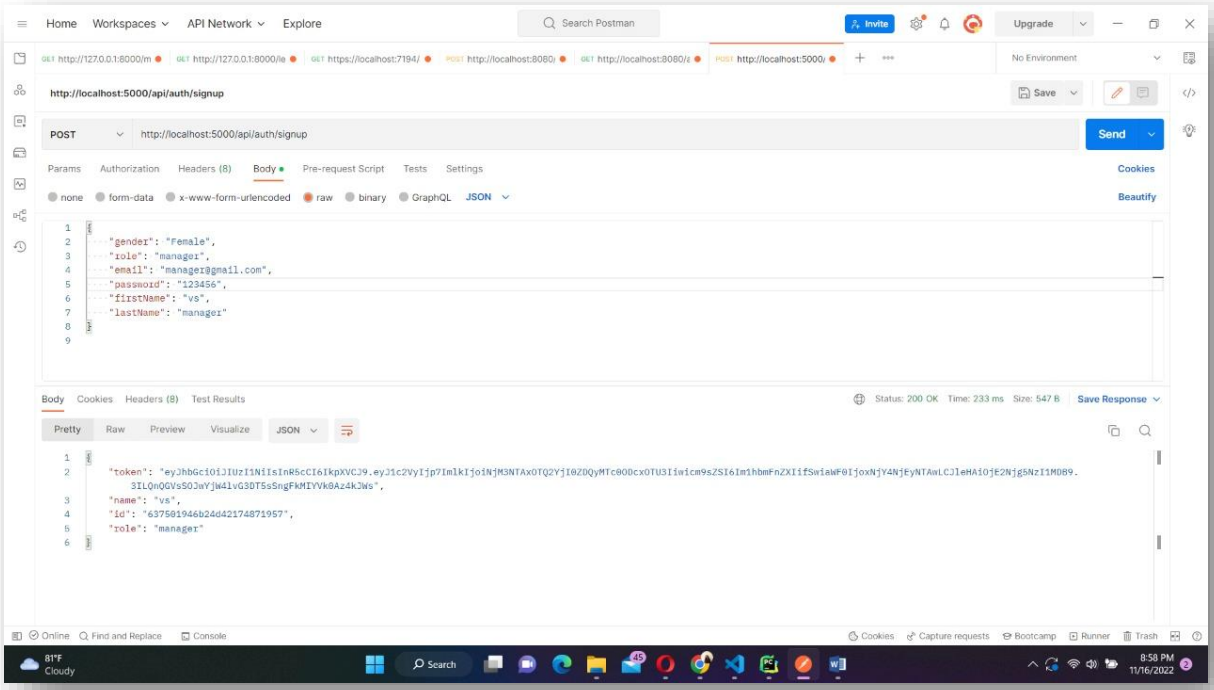
Test Case No	Test Case 13
Description	Click "Save message" button after enter the message.
Test Steps	1. Login as a worker. 2. Enter the message in message field. 3. Click the "Save message" button.
Expected Result	*Should display the success message "Message has been saved".*
Actual Result	Displayed the success message "Message has been saved".
Test Status	**Pass**
User Role	Worker

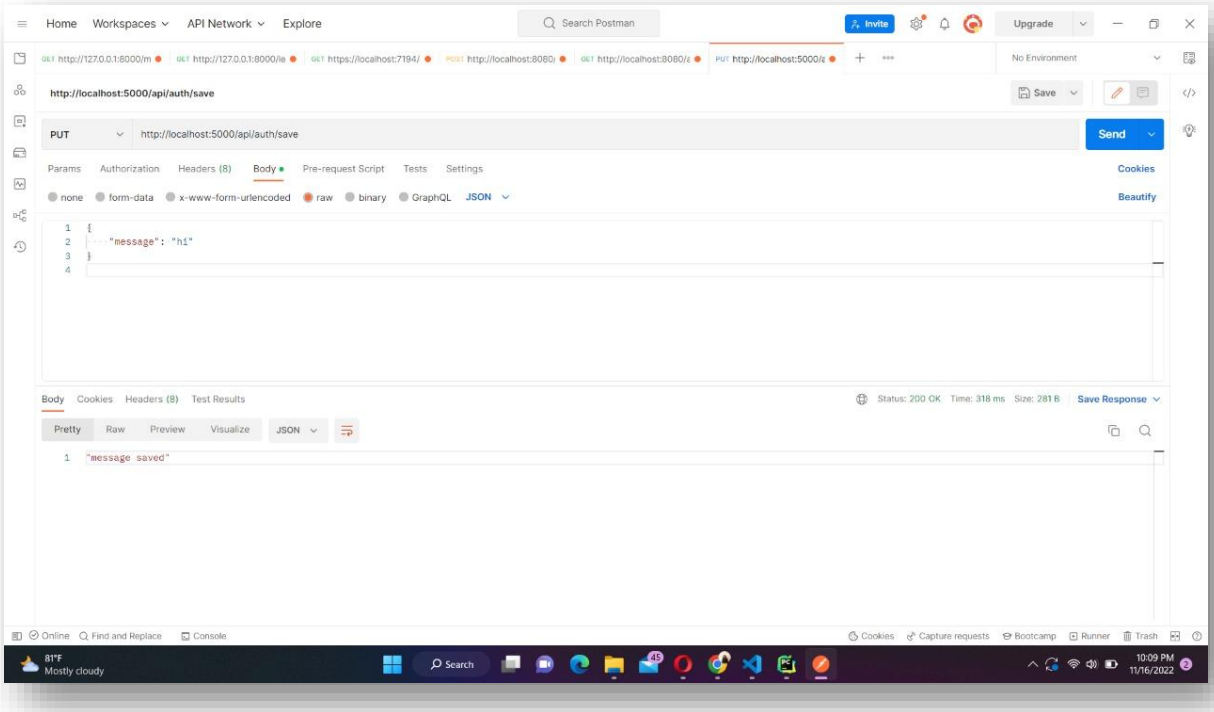
Test Case No	Test Case 14
Description	Click "Save message" button without entering the message.

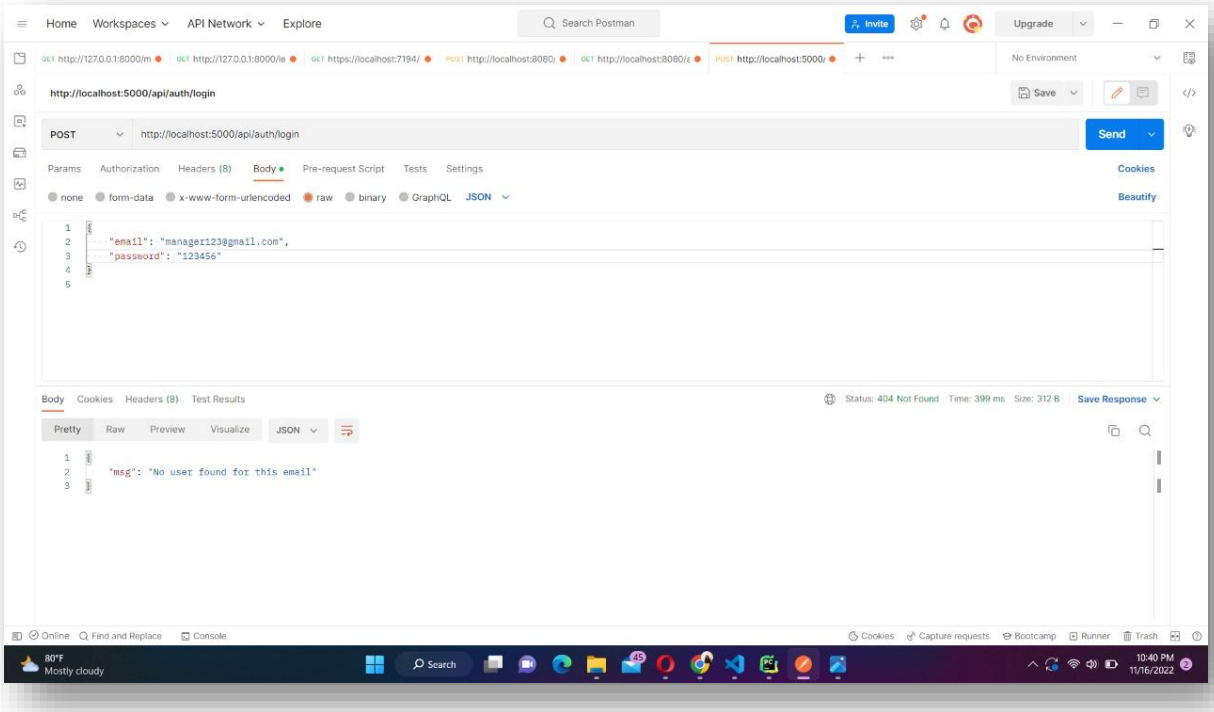
Test Steps	1. Login as a worker. 2. Click the “Save message” button.
Expected Result	*Should display the error message “Message Field is Empty”.*
Actual Result	Displayed the error message “Message Field is Empty”.
Test Status	**Pass**
User Role	Worker

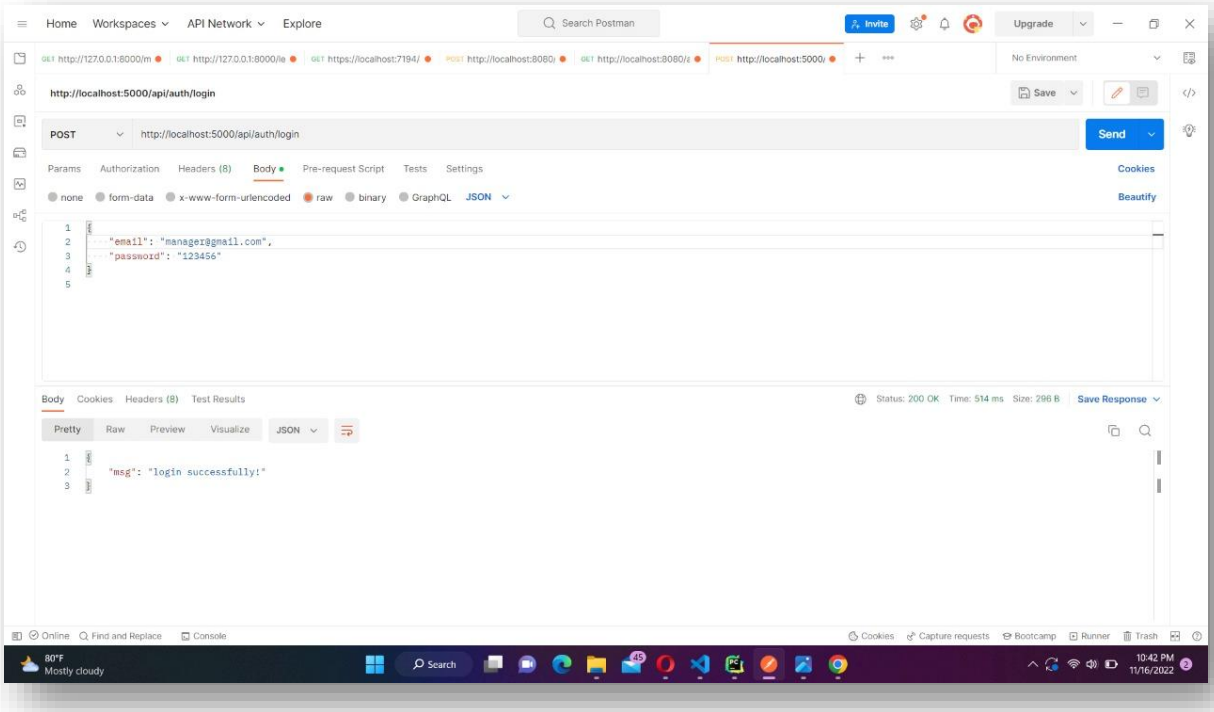
Test Case No	Test Case 15
Description	Click “Save message” button without entering the message.
Test Steps	1. Login as a manager. 2. Click the “Save message” button.
Expected Result	*Should display the error message “Message Field is Empty”.*
Actual Result	Displayed the error message “Message Field is Empty”.
Test Status	**Pass**
User Role	Manager

API Testing

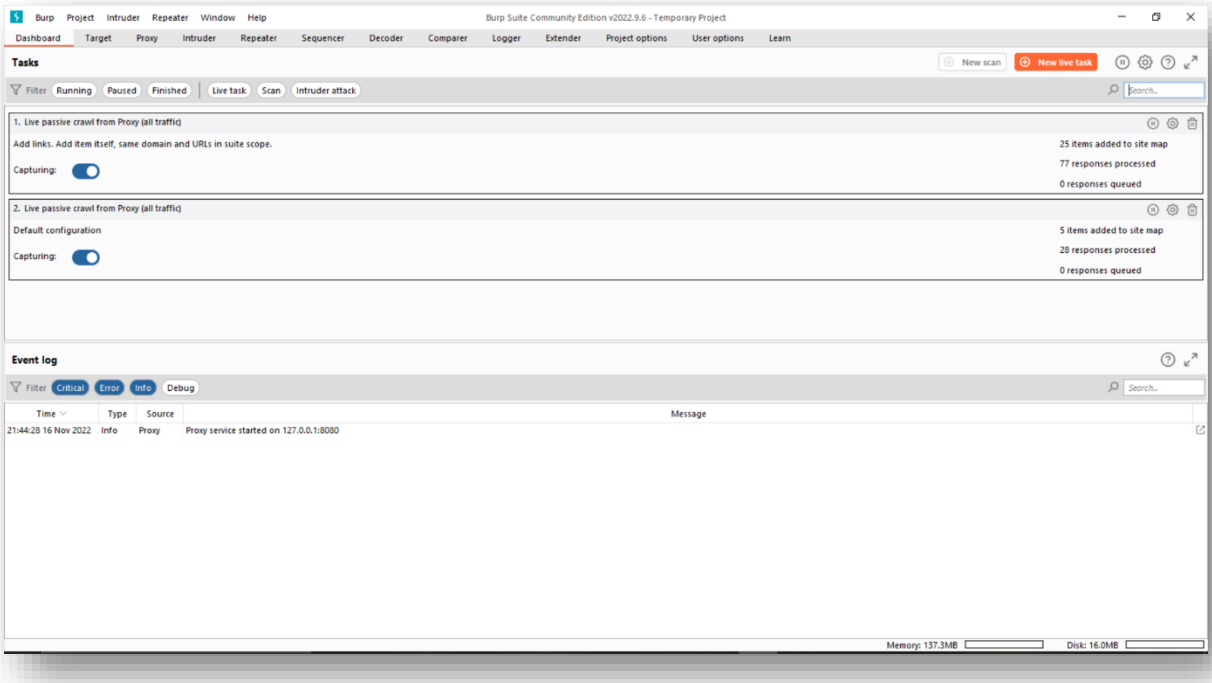

Home Workspaces API Network Explore
Search Postman
Invite
Upgrade
http://localhost:5000/api/auth/signup
POST
http://localhost:5000/api/auth/signup
Send
Params Authorization Headers (8) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"gender": "Female",
"role": "manager",
"email": "manager@gmail.com",
"password": "123456",
"firstName": "vs",
"lastName": "manager"
}
Body Cookies Headers (8) Test Results
Status: 200 OK Time: 233 ms Size: 547 B
Save Response
Pretty Raw Preview Visualize JSON
{
"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjM3NTAxOTQ2YjI0ZDQyMTc0ODcxOTU3IiwicmFsZSI6Im1hbmFnZXIifSwiaWF0IjoxNjY4NjEyNTAwLCJleHAiOjE2Njg5NzI1MDB9.
3ILQnQGVsSOJwYjW41vG3DT5sSngFkMIYVk0Az4kJWs",
"name": "vs",
"id": "637501946b24d42174871957",
"role": "manager"
}
Online Find and Replace Console
Cookies Capture requests Bootcamp Runner Trash
81°F Cloudy
Search
8:58 PM 11/16/2022

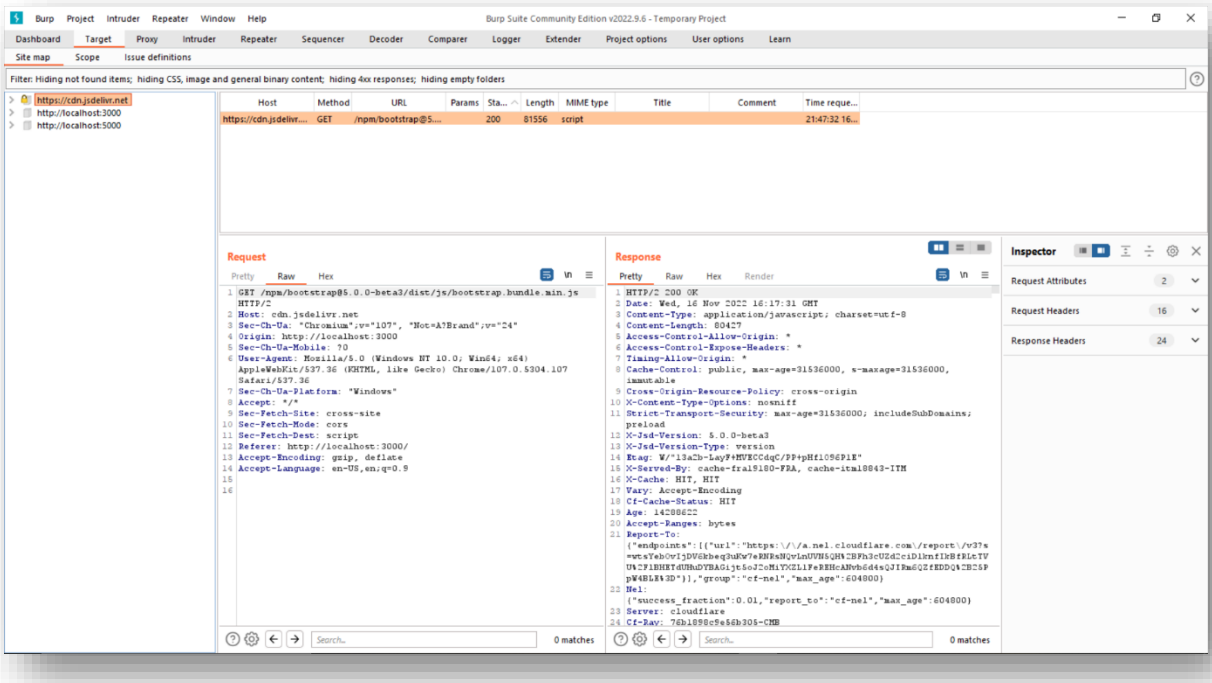

Home Workspaces API Network Explore
Search Postman
Invite
Upgrade
http://localhost:5000/api/auth/save
PUT
http://localhost:5000/api/auth/save
Send
Params Authorization Headers (8) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"message": "hi"
}
Body Cookies Headers (8) Test Results
Status: 200 OK Time: 318 ms Size: 281 B
Save Response
Pretty Raw Preview Visualize JSON
"message saved"
Online Find and Replace Console
Cookies Capture requests Bootcamp Runner Trash
81°F Mostly cloudy
Search
10:09 PM 11/16/2022

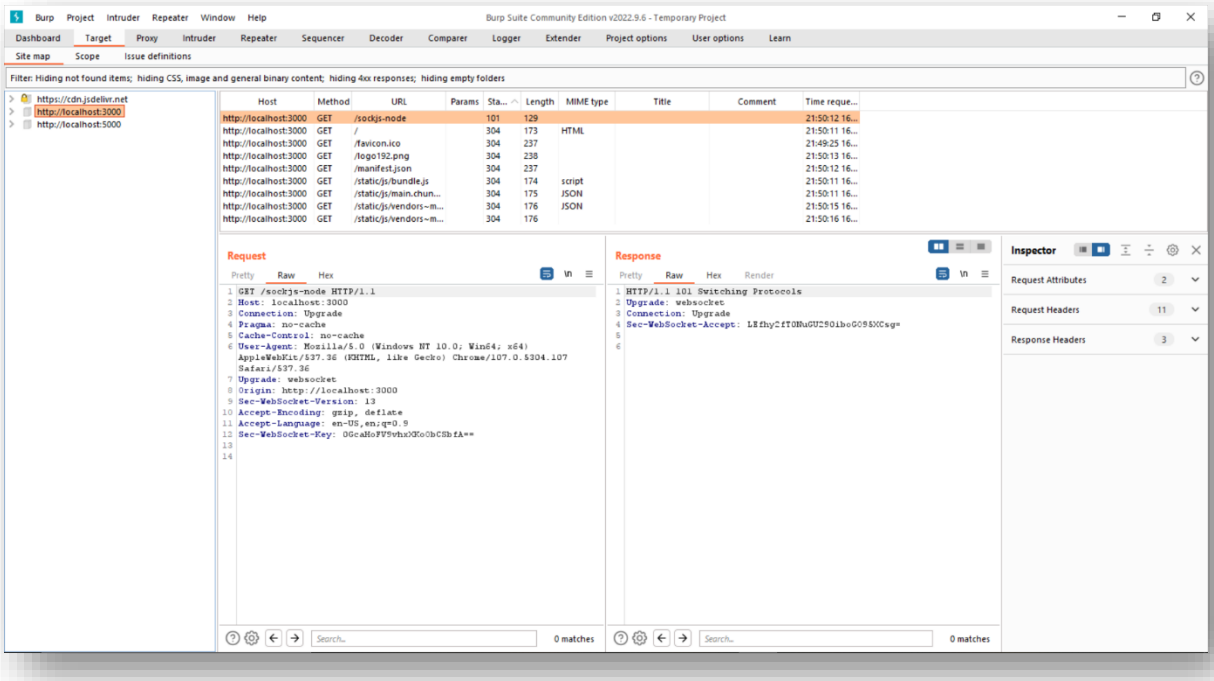

Home Workspaces API Network Explore
Search Postman
Invite
Upgrade
http://localhost:5000/api/auth/login
POST
http://localhost:5000/api/auth/login
Send
Params Authorization Headers (8) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"email": "manager123@gmail.com",
"password": "123456"
}
Body Cookies Headers (8) Test Results
Status: 404 Not Found Time: 399 ms Size: 312 B
Save Response
Pretty Raw Preview Visualize JSON
{
"msg": "No user found for this email"
}
Online Find and Replace Console
Cookies Capture requests Bootcamp Runner Trash
80°F
Mostly cloudy
Search
10:40 PM
11/16/2022

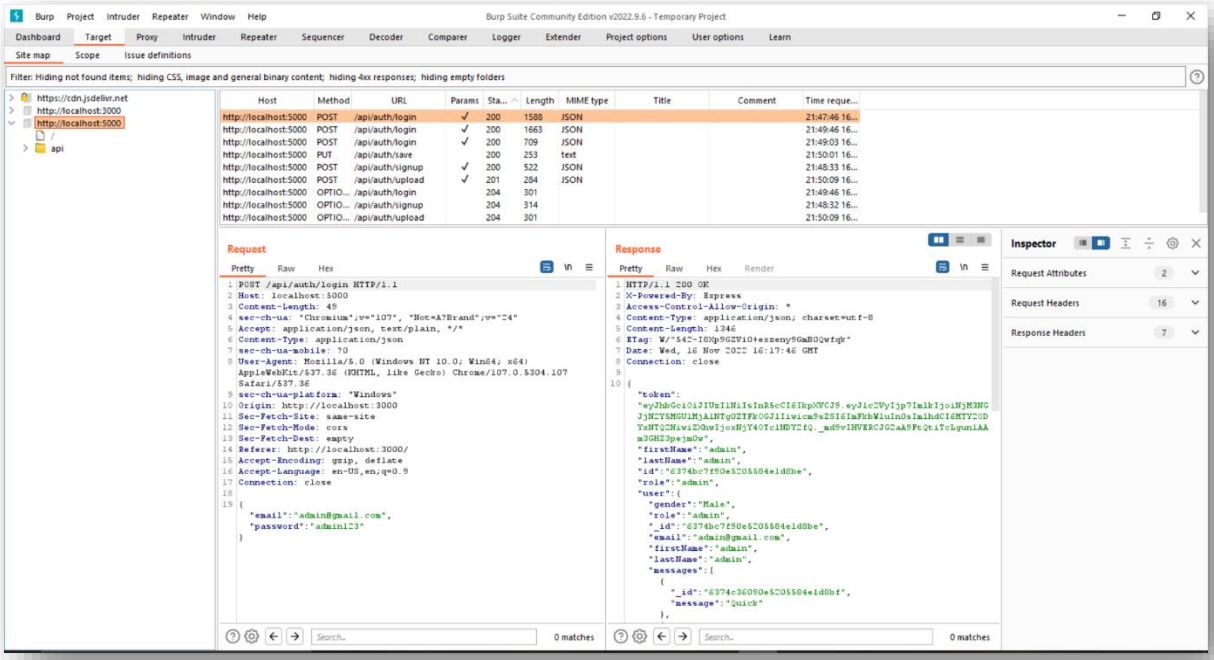

Home Workspaces API Network Explore
Search Postman
Invite
Upgrade
http://localhost:5000/api/auth/login
POST
http://localhost:5000/api/auth/login
Send
Params Authorization Headers (8) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"email": "manager@gmail.com",
"password": "123456"
}
Body Cookies Headers (8) Test Results
Status: 200 OK Time: 514 ms Size: 296 B
Save Response
Pretty Raw Preview Visualize JSON
{
"msg": "login successfully!"
}
Online Find and Replace Console
Cookies Capture requests Bootcamp Runner Trash
80°F
Mostly cloudy
Search
10:42 PM
11/16/2022

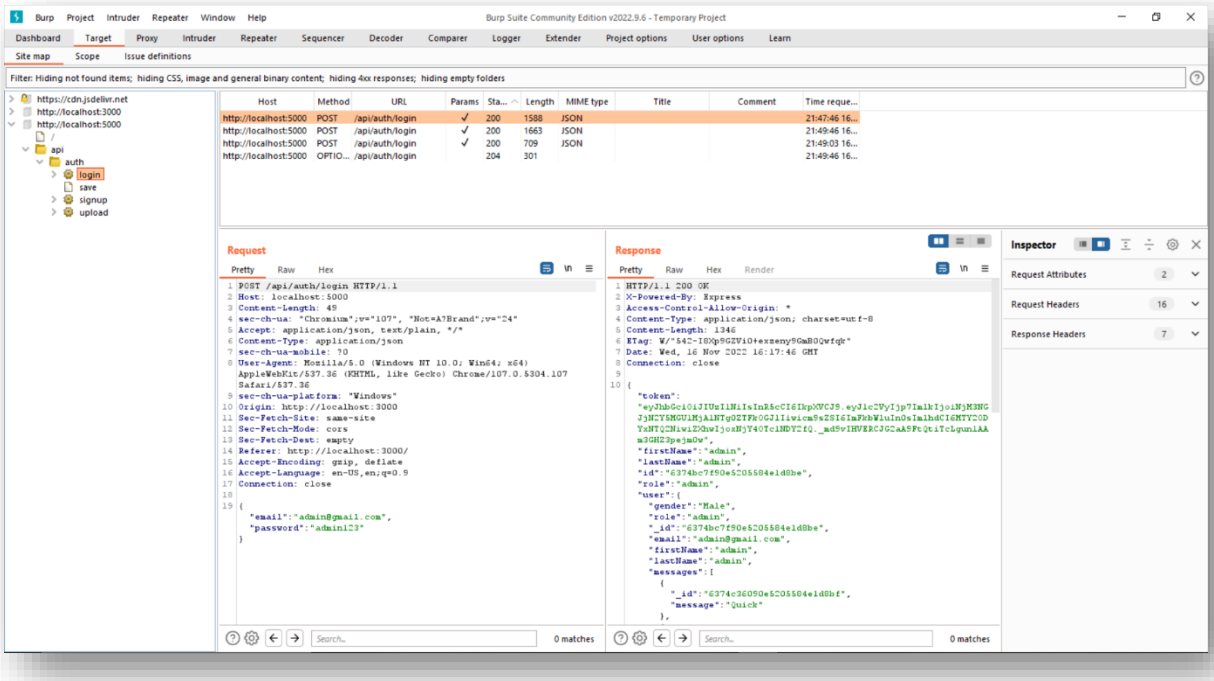
Burp Suite Security Testing

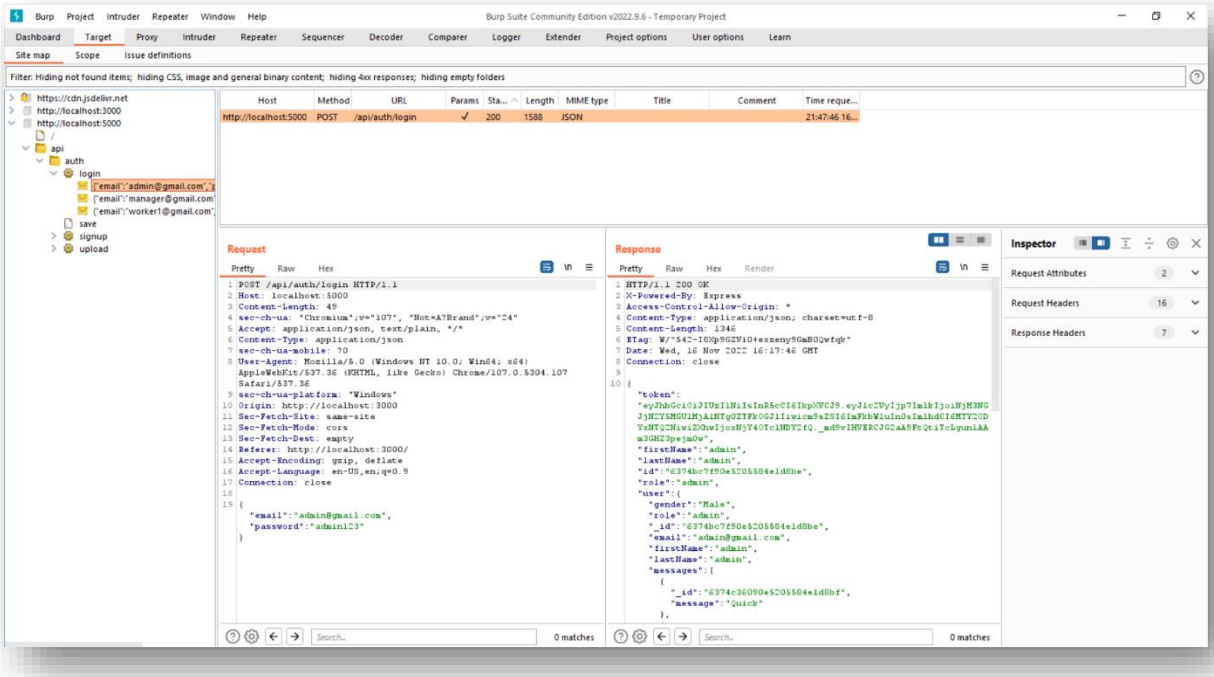


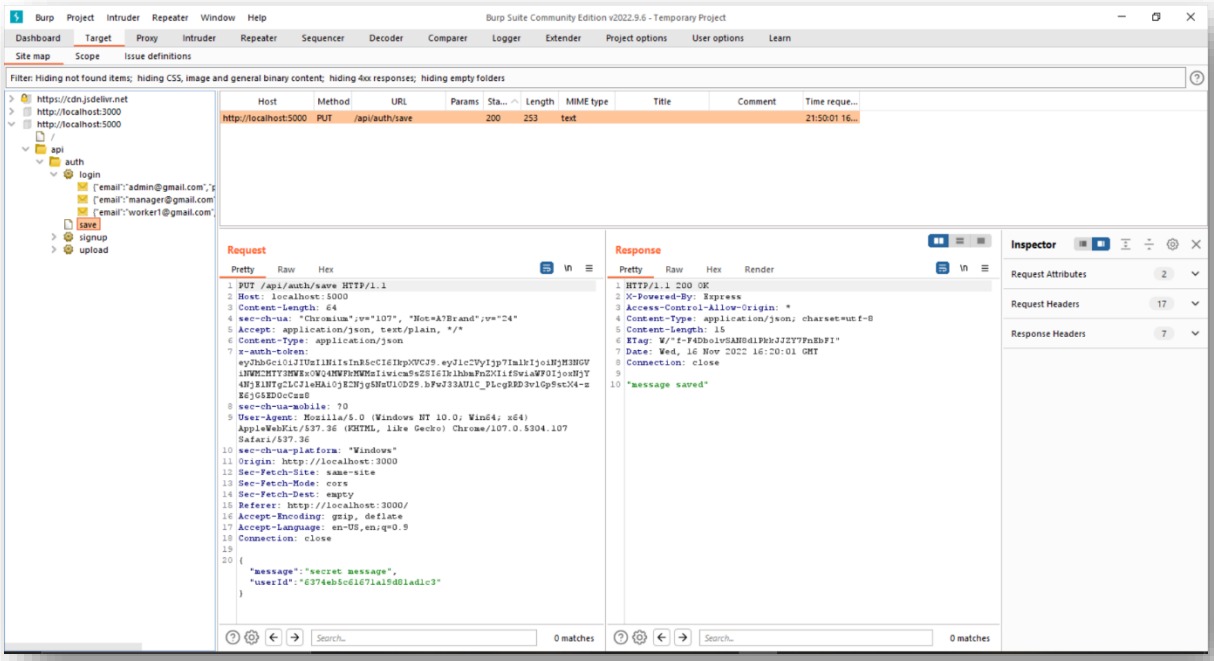


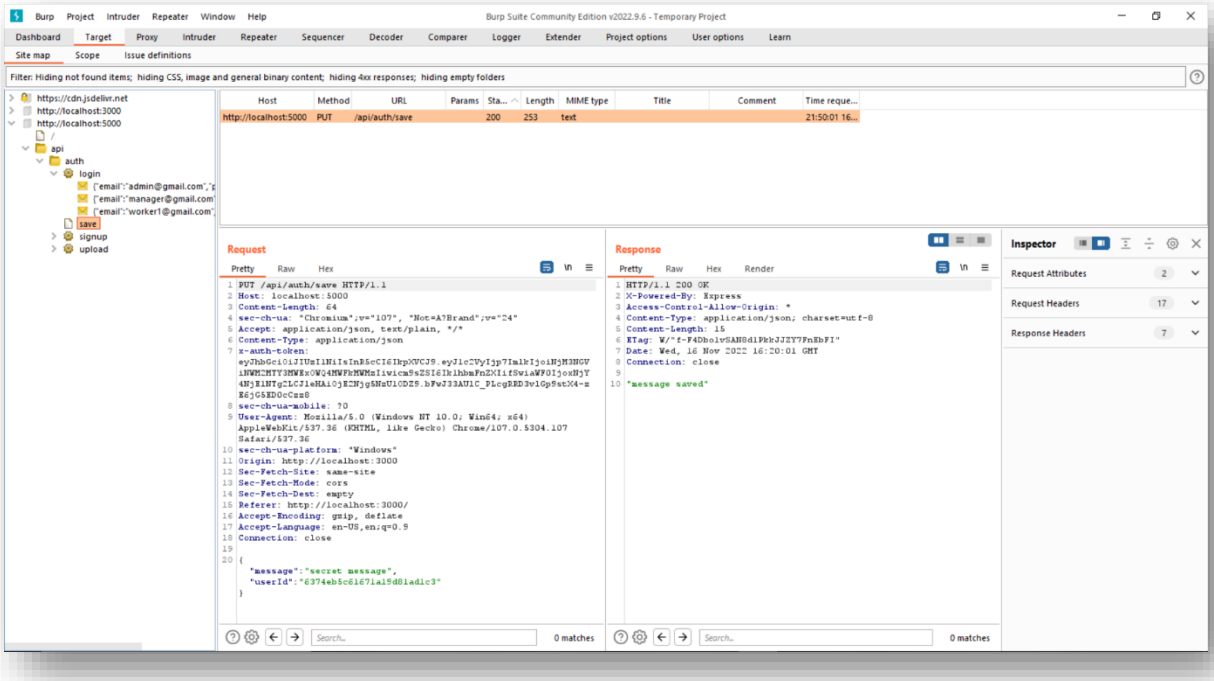

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Site map Scope Issue definitions
Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders
https://cdn.jsdelivr.net
http://localhost:3000
http://localhost:5000
Host Method URL Params Sta... Length MIME type Title Comment Time reque...
http://localhost:3000 GET /sockjs-node 101 129 21:50:12 16...
http://localhost:3000 GET / 304 173 HTML 21:50:11 16...
http://localhost:3000 GET /favicon.ico 304 237 21:49:25 16...
http://localhost:3000 GET /logo192.png 304 238 21:50:13 16...
http://localhost:3000 GET /manifest.json 304 237 21:50:12 16...
http://localhost:3000 GET /static/js/bundle.js 304 174 script 21:50:11 16...
http://localhost:3000 GET /static/js/main.chun... 304 175 JSON 21:50:11 16...
http://localhost:3000 GET /static/js/vendors~m... 304 176 JSON 21:50:15 16...
http://localhost:3000 GET /static/js/vendors~m... 304 176 21:50:16 16...
Request
Pretty Raw Hex
1 GET /sockjs-node HTTP/1.1
2 Host: localhost:3000
3 Connection: Upgrade
4 Pragma: no-cache
5 Cache-Control: no-cache
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.5304.107 Safari/537.36
7 Upgrade: websocket
8 Origin: http://localhost:3000
9 Sec-WebSocket-Version: 13
10 Accept-Encoding: gzip, deflate
11 Accept-Language: en-US,en;q=0.9
12 Sec-WebSocket-Key: GGcaHoFV5vhxXKo0bCSbfA==
13
14
Response
Pretty Raw Hex Render
1 HTTP/1.1 101 Switching Protocols
2 Upgrade: websocket
3 Connection: Upgrade
4 Sec-WebSocket-Accept: LEfhyCfTONuGUZ90iboGO9SXCsg=
5
6
Inspector
Request Attributes 2
Request Headers 11
Response Headers 3
Search... 0 matches
Search... 0 matches

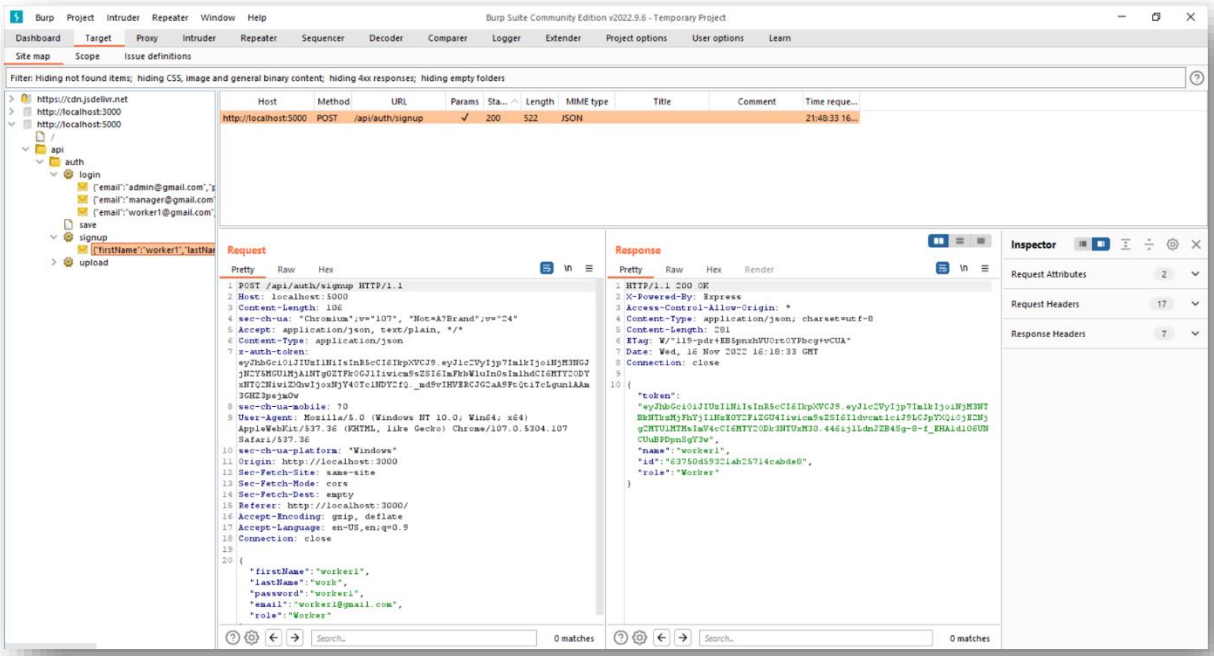

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Site map Scope Issue definitions
Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders
https://cdn.jsdelivr.net
http://localhost:3000
http://localhost:5000
/
api
Host Method URL Params Sta... Length MIME type Title Comment Time reque...
http://localhost:5000 POST /api/auth/login ✓ 200 1588 JSON 21:47:46 16...
http://localhost:5000 POST /api/auth/login ✓ 200 1663 JSON 21:49:46 16...
http://localhost:5000 POST /api/auth/login ✓ 200 709 JSON 21:49:03 16...
http://localhost:5000 PUT /api/auth/save 200 253 text 21:50:01 16...
http://localhost:5000 POST /api/auth/signup ✓ 200 522 JSON 21:48:33 16...
http://localhost:5000 POST /api/auth/upload ✓ 201 284 JSON 21:50:09 16...
http://localhost:5000 OPTIO... /api/auth/login 204 301 21:49:46 16...
http://localhost:5000 OPTIO... /api/auth/signup 204 314 21:48:32 16...
http://localhost:5000 OPTIO... /api/auth/upload 204 301 21:50:09 16...
Request
Pretty Raw Hex
1 POST /api/auth/login HTTP/1.1
2 Host: localhost:5000
3 Content-Length: 45
4 sec-ch-ua: "Chromium";v="107", "Not=A?Brand";v="24"
5 Accept: application/json, text/plain, */*
6 Content-Type: application/json
7 sec-ch-ua-mobile: ?0
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.5304.107 Safari/537.36
9 sec-ch-ua-platform: "Windows"
10 Origin: http://localhost:3000
11 Sec-Fetch-Site: same-site
12 Sec-Fetch-Mode: cors
13 Sec-Fetch-Dest: empty
14 Referer: http://localhost:3000/
15 Accept-Encoding: gzip, deflate
16 Accept-Language: en-US,en;q=0.9
17 Connection: close
18
19 {
"email":"admin@gmail.com",
"password":"admin123"
}
Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Access-Control-Allow-Origin: *
4 Content-Type: application/json; charset=utf-8
5 Content-Length: 1346
6 ETag: W/"542-IEXp9GZViO+exzeny5GmBQwfqk"
7 Date: Wed, 16 Nov 2022 16:17:46 GMT
8 Connection: close
9
10 {
"token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjM3NGJyNTY4NGU1MjAlHTg2ZTFkOGI1liwicm9sZSI6ImFkbWluIn0sImlhdCI6MTY2ODYxNTQ2NiwiZXhwIjoxNjY4OTc1MDY2fQ._md9v1HVEBCJG2aA9PtQtiTcLqunlAAm3GH23pejmUw",
"firstName":"admin",
"lastName":"admin",
"id":"6374bc7f50e5205504e1d8be",
"role":"admin",
"user":{
"gender":"Male",
"role":"admin",
"_id":"6374bc7f50e5205504e1d8be",
"email":"admin@gmail.com",
"firstName":"admin",
"lastName":"admin",
"messages":[
{
"_id":"6374c36090e5205504e1d8bf",
"message":"Quick"
},
Inspector
Request Attributes 2
Request Headers 16
Response Headers 7
Search... 0 matches
Search... 0 matches

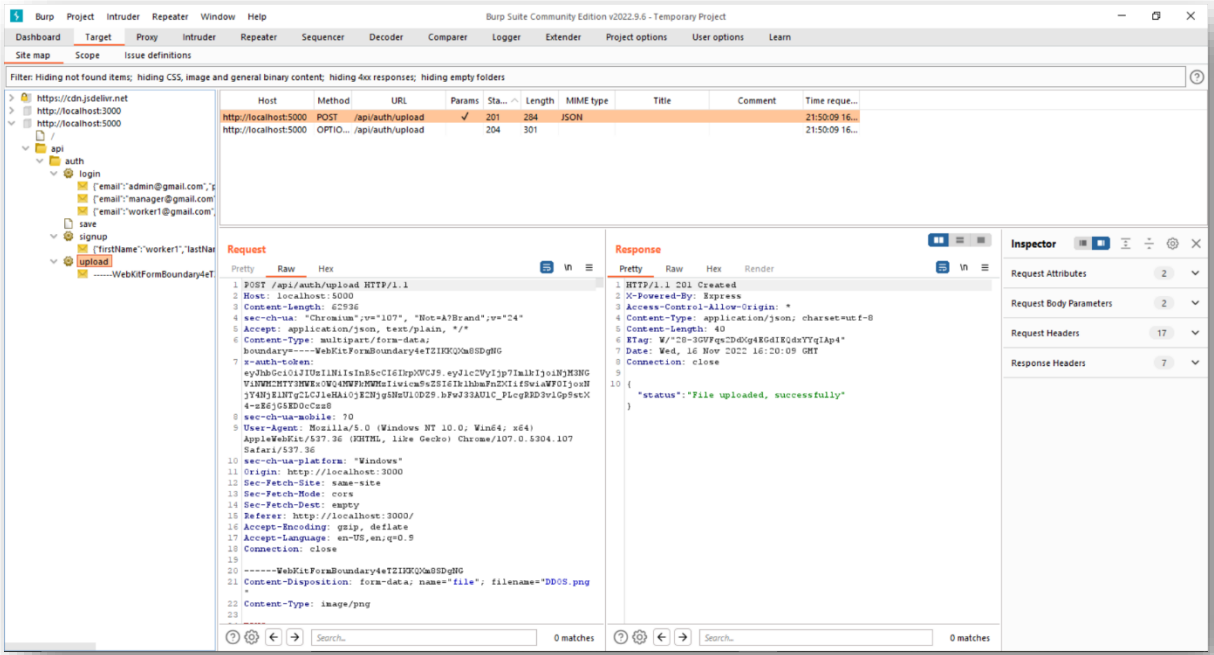

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Site map Scope Issue definitions
Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders
https://cdn.jsdelivr.net
http://localhost:3000
http://localhost:5000
api
auth
login
save
signup
upload
Host Method URL Params Sta... Length MIME type Title Comment Time reque...
http://localhost:5000 POST /api/auth/login 200 1588 JSON 21:47:46 16...
http://localhost:5000 POST /api/auth/login 200 1663 JSON 21:49:46 16...
http://localhost:5000 POST /api/auth/login 200 709 JSON 21:49:03 16...
http://localhost:5000 OPTIO... /api/auth/login 204 301 21:49:46 16...
Request
Pretty Raw Hex
POST /api/auth/login HTTP/1.1
Host: localhost:5000
Content-Length: 49
sec-ch-ua: "Chromium";v="107", "Not=A?Brand";v="24"
Accept: application/json, text/plain, */*
Content-Type: application/json
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.5304.107 Safari/537.36
sec-ch-ua-platform: "Windows"
Origin: http://localhost:3000
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://localhost:3000/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close
{
"email":"admin@gmail.com",
"password":"admin123"
}
Response
Pretty Raw Hex Render
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 1346
ETag: W/"542-IE0p9GZVlO+exzenySGmBQwfqk"
Date: Wed, 16 Nov 2022 16:17:46 GMT
Connection: close
{
"token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjM3NG
JjNzY5MGUlMjAiNTg0ZTFkOGJlIiwicm9sZSI6ImFkbWluIn0sImlhdCI6MTY2OD
YxNTQ2NiwiZXhwIjoxNjY4OTc1NDY2fQ._md5vIHVERCJGCaA9PtQtiTcLqunlAA
m3GH2BpejmOw",
"firstName":"admin",
"lastName":"admin",
"id":"6374bc7f90e5205584e1d8be",
"role":"admin",
"user":{
"gender":"Male",
"role":"admin",
"_id":"6374bc7f90e5205584e1d8be",
"email":"admin@gmail.com",
"firstName":"admin",
"lastName":"admin",
"messages":[
{
"_id":"6374c36090e5205584e1d8bf",
"message":"Quick
},
Search... 0 matches
Search... 0 matches
Inspector
Request Attributes 2
Request Headers 16
Response Headers 7

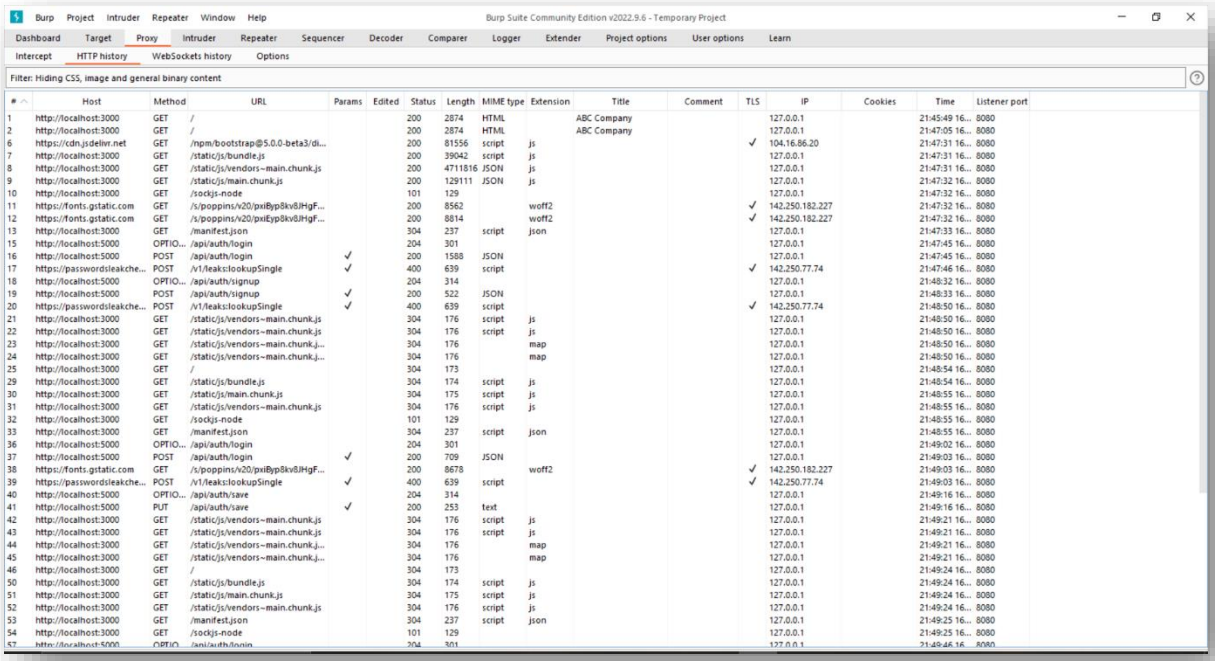

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Site map Scope Issue definitions
Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders
https://cdn.jsdelivr.net
http://localhost:3000
http://localhost:5000
api
auth
login
{"email":"admin@gmail.com",
{"email":"manager@gmail.com"
{"email":"worker1@gmail.com"
save
signup
upload
Host Method URL Params Sta... Length MIME type Title Comment Time reque...
http://localhost:5000 POST /api/auth/login 200 1588 JSON 21:47:46 16...
Request
Pretty Raw Hex
POST /api/auth/login HTTP/1.1
Host: localhost:5000
Content-Length: 49
sec-ch-ua: "Chromium";v="107", "Not=A?Brand";v="24"
Accept: application/json, text/plain, */*
Content-Type: application/json
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.5304.107 Safari/537.36
sec-ch-ua-platform: "Windows"
Origin: http://localhost:3000
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://localhost:3000/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close
{
"email":"admin@gmail.com",
"password":"admin123"
}
Response
Pretty Raw Hex Render
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 1346
ETag: W/"542-IE0p9GZVlO+exzenySGmBQwfqk"
Date: Wed, 16 Nov 2022 16:17:46 GMT
Connection: close
{
"token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjM3NG
JjNzY5MGUlMjAiNTg0ZTFkOGJlIiwicm9sZSI6ImFkbWluIn0sImlhdCI6MTY2OD
YxNTQ2NiwiZXhwIjoxNjY4OTc1NDY2fQ._md5vIHVERCJGCaA9PtQtiTcLqunlAA
m3GH2BpejmOw",
"firstName":"admin",
"lastName":"admin",
"id":"6374bc7f90e5205584e1d8be",
"role":"admin",
"user":{
"gender":"Male",
"role":"admin",
"_id":"6374bc7f90e5205584e1d8be",
"email":"admin@gmail.com",
"firstName":"admin",
"lastName":"admin",
"messages":[
{
"_id":"6374c36090e5205584e1d8bf",
"message":"Quick
},
Search... 0 matches
Search... 0 matches
Inspector
Request Attributes 2
Request Headers 16
Response Headers 7

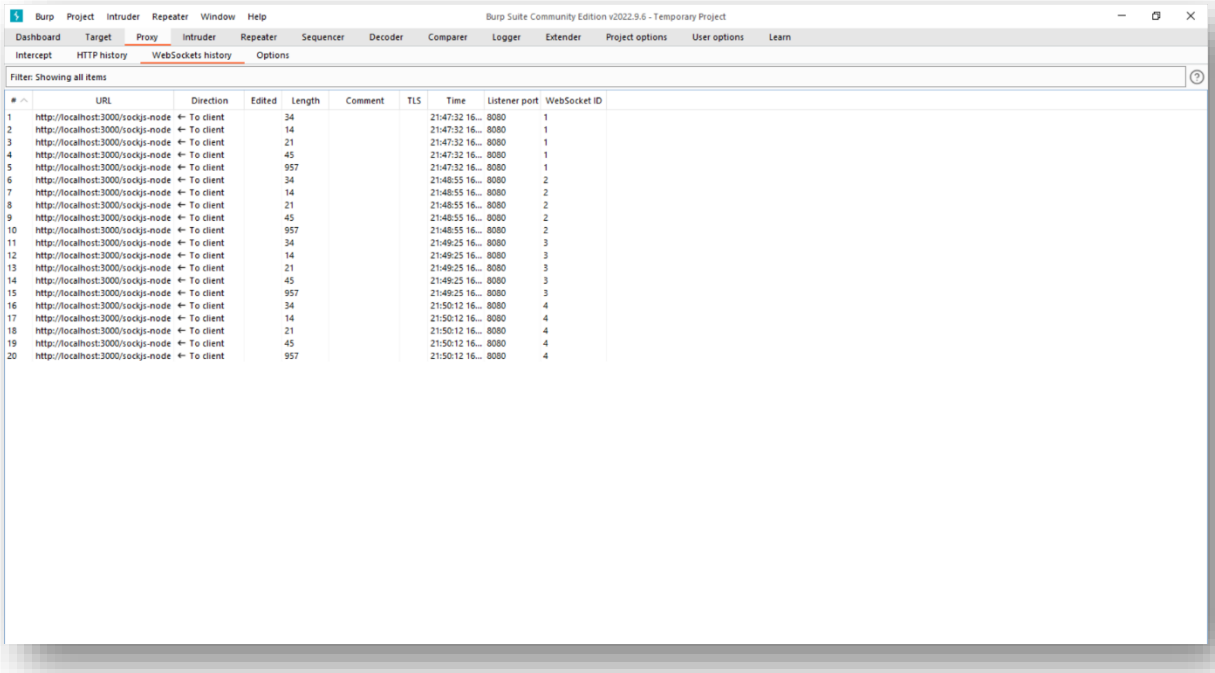

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Site map Scope Issue definitions
Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders
Host Method URL Params Sta... Length MIME type Title Comment Time reque...
http://localhost:5000 PUT /api/auth/save 200 253 text 21:50:01 16...
Request
Pretty Raw Hex
PUT /api/auth/save HTTP/1.1
Host: localhost:5000
Content-Length: 64
sec-ch-ua: "Chromium";v="107", "Not=A?Brand";v="24"
Accept: application/json, text/plain, */*
Content-Type: application/json
x-auth-token:
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7Imlk IjoiNjM3NGV
iNWM2MTY3MWExOWQ4MWFkMWMxIiwicmFsZSI6Ikhbm FnZXIifSwiaWF0IjoxNjY
4NjE1NTgZLCJleHAiOjE2NjgNzU1ODZ9.bFwJ33AU1C_PLcgRRD3vlGp5stX4-z
E6jG5EDoCzsE
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.5304.107
Safari/537.36
sec-ch-ua-platform: "Windows"
Origin: http://localhost:3000
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://localhost:3000/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close
{
"message":"secret message",
"userId":"6374eb5c6171a15d81adlc3"
}
Response
Pretty Raw Hex Render
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 15
ETag: W/"f-F4DbolvSAH0d1PhkJJZY7FnEhFI"
Date: Wed, 16 Nov 2022 16:20:01 GMT
Connection: close
"message saved"
Inspector
Request Attributes 2
Request Headers 17
Response Headers 7
0 matches
0 matches

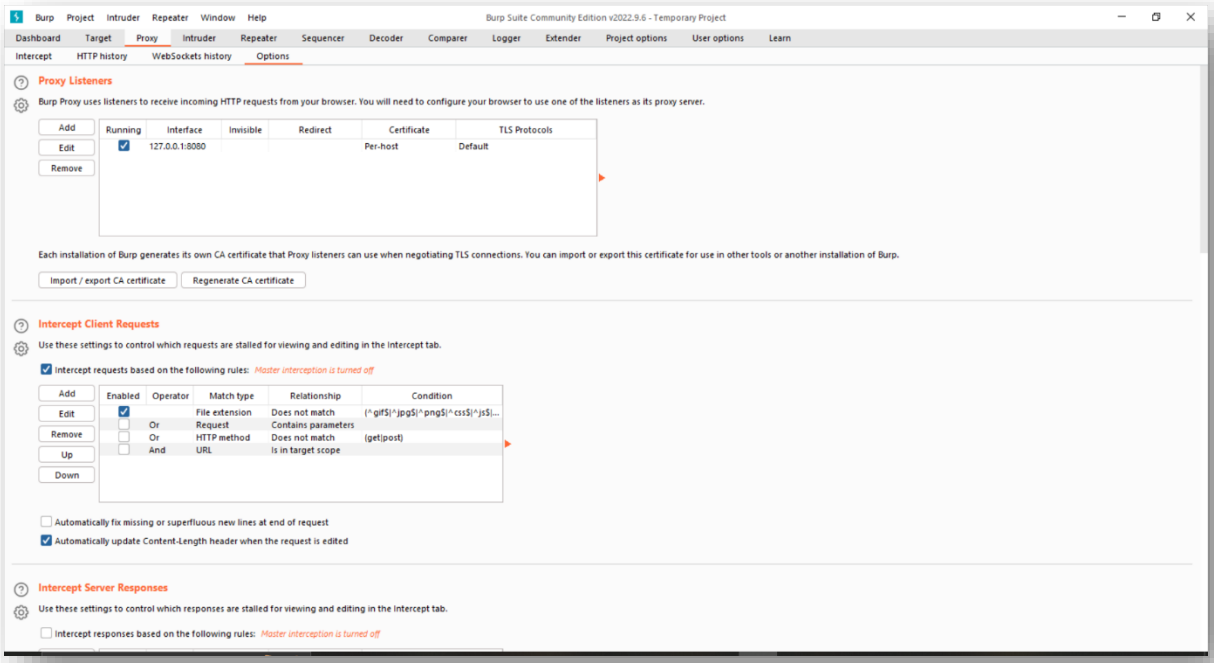

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Site map Scope Issue definitions
Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders
Host Method URL Params Sta... Length MIME type Title Comment Time reque...
http://localhost:5000 PUT /api/auth/save 200 253 text 21:50:01 16...
Request
Pretty Raw Hex
PUT /api/auth/save HTTP/1.1
Host: localhost:5000
Content-Length: 64
sec-ch-ua: "Chromium";v="107", "Not=A?Brand";v="24"
Accept: application/json, text/plain, */*
Content-Type: application/json
x-auth-token:
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7Imlk IjoiNjM3NGV
iNWM2MTY3MWExOWQ4MWFkMWMxIiwicmFsZSI6Ikhbm FnZXIifSwiaWF0IjoxNjY
4NjE1NTgZLCJleHAiOjE2NjgNzU1ODZ9.bFwJ33AU1C_PLcgRRD3vlGp5stX4-z
E6jG5EDoCzsE
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.5304.107
Safari/537.36
sec-ch-ua-platform: "Windows"
Origin: http://localhost:3000
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://localhost:3000/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close
{
"message":"secret message",
"userId":"6374eb5c6171a15d81adlc3"
}
Response
Pretty Raw Hex Render
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 15
ETag: W/"f-F4DbolvSAH0d1PhkJJZY7FnEhFI"
Date: Wed, 16 Nov 2022 16:20:01 GMT
Connection: close
"message saved"
Inspector
Request Attributes 2
Request Headers 17
Response Headers 7
0 matches
0 matches

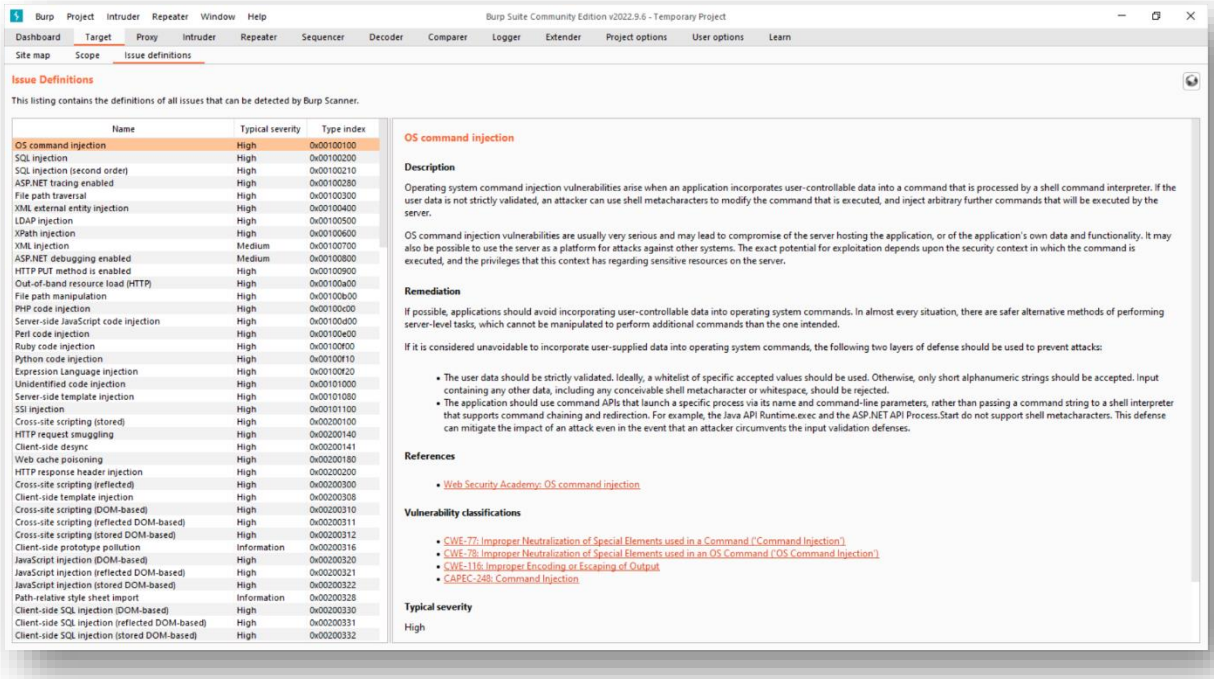

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Site map Scope Issue definitions
Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders
Host Method URL Params Sta... Length MIME type Title Comment Time reque...
http://localhost:5000 POST /api/auth/signup ✓ 200 522 JSON 21:48:33 16...
Request
Response
Inspector
Request Attributes 2
Request Headers 17
Response Headers 7

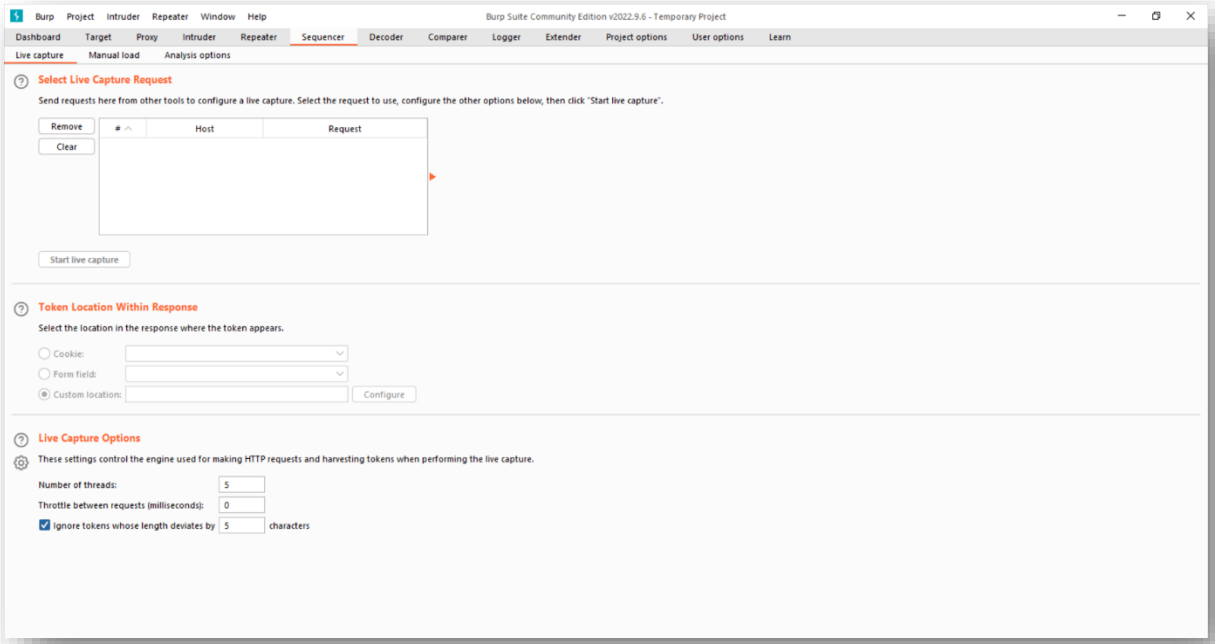

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Site map Scope Issue definitions
Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders
Host Method URL Params Sta... Length MIME type Title Comment Time reque...
http://localhost:5000 POST /api/auth/upload ✓ 201 284 JSON 21:50:09 16...
http://localhost:5000 OPTIO... /api/auth/upload 204 301 21:50:09 16...
Request
Response
Inspector
Request Attributes 2
Request Body Parameters 2
Request Headers 17
Response Headers 7

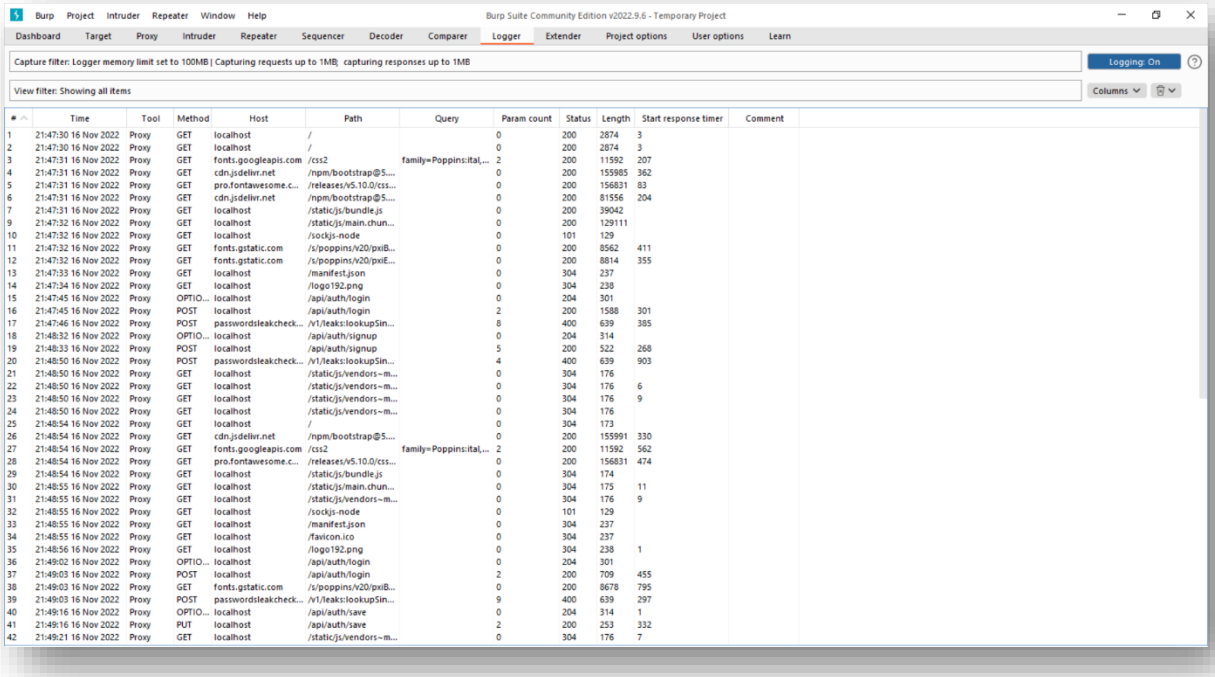

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Intercept HTTP history WebSockets history Options
Filter: Hiding CSS, image and general binary content
Host Method URL Params Edited Status Length MIME type Extension Title Comment TLS IP Cookies Time Listener port
1 http://localhost:3000 GET / 200 2874 HTML ABC Company 127.0.0.1 21:45:49 16... 8080
2 http://localhost:3000 GET / 200 2874 HTML ABC Company 127.0.0.1 21:47:05 16... 8080
6 https://cdn.jsdelivr.net GET /npm/bootstrap@5.0.0-beta3/di... 200 81556 script js ✓ 104.16.86.20 21:47:31 16... 8080
7 http://localhost:3000 GET /static/js/bundle.js 200 39042 script js 127.0.0.1 21:47:31 16... 8080
8 http://localhost:3000 GET /static/js/vendors~main.chunk.js 200 4711816 JSON js 127.0.0.1 21:47:31 16... 8080
9 http://localhost:3000 GET /static/js/main.chunk.js 200 129111 JSON js 127.0.0.1 21:47:32 16... 8080
10 http://localhost:3000 GET /sockjs-node 101 129 127.0.0.1 21:47:32 16... 8080
11 https://fonts.gstatic.com GET /s/poppins/v20/pxiEyp8kv8JHgF... 200 8562 woff2 ✓ 142.250.182.227 21:47:32 16... 8080
12 https://fonts.gstatic.com GET /s/poppins/v20/pxiEyp8kv8JHgF... 200 8814 woff2 ✓ 142.250.182.227 21:47:32 16... 8080
13 http://localhost:3000 GET /manifest.json 304 237 script json 127.0.0.1 21:47:33 16... 8080
15 http://localhost:5000 OPTIO... /api/auth/login 204 301 127.0.0.1 21:47:45 16... 8080
16 http://localhost:5000 POST /api/auth/login ✓ 200 1588 JSON 127.0.0.1 21:47:45 16... 8080
17 https://passwordsleakche... POST /v1/leaks:lookupSingle ✓ 400 639 script ✓ 142.250.77.74 21:47:46 16... 8080
18 http://localhost:5000 OPTIO... /api/auth/signup 204 314 127.0.0.1 21:48:32 16... 8080
19 http://localhost:5000 POST /api/auth/signup ✓ 200 522 JSON 127.0.0.1 21:48:33 16... 8080
20 https://passwordsleakche... POST /v1/leaks:lookupSingle ✓ 400 639 script ✓ 142.250.77.74 21:48:50 16... 8080
21 http://localhost:3000 GET /static/js/vendors~main.chunk.js 304 176 script js 127.0.0.1 21:48:50 16... 8080
22 http://localhost:3000 GET /static/js/vendors~main.chunk.js 304 176 script js 127.0.0.1 21:48:50 16... 8080
23 http://localhost:3000 GET /static/js/vendors~main.chunk.j... 304 176 map 127.0.0.1 21:48:50 16... 8080
24 http://localhost:3000 GET /static/js/vendors~main.chunk.j... 304 176 map 127.0.0.1 21:48:50 16... 8080
25 http://localhost:3000 GET / 304 173 127.0.0.1 21:48:54 16... 8080
29 http://localhost:3000 GET /static/js/bundle.js 304 174 script js 127.0.0.1 21:48:54 16... 8080
30 http://localhost:3000 GET /static/js/main.chunk.js 304 175 script js 127.0.0.1 21:48:55 16... 8080
31 http://localhost:3000 GET /static/js/vendors~main.chunk.js 304 176 script js 127.0.0.1 21:48:55 16... 8080
32 http://localhost:3000 GET /sockjs-node 101 129 127.0.0.1 21:48:55 16... 8080
33 http://localhost:3000 GET /manifest.json 304 237 script json 127.0.0.1 21:48:55 16... 8080
36 http://localhost:5000 OPTIO... /api/auth/login 204 301 127.0.0.1 21:49:02 16... 8080
37 http://localhost:5000 POST /api/auth/login ✓ 200 709 JSON 127.0.0.1 21:49:03 16... 8080
38 https://fonts.gstatic.com GET /s/poppins/v20/pxiEyp8kv8JHgF... 200 8678 woff2 ✓ 142.250.182.227 21:49:03 16... 8080
39 https://passwordsleakche... POST /v1/leaks:lookupSingle ✓ 400 639 script ✓ 142.250.77.74 21:49:03 16... 8080
40 http://localhost:5000 OPTIO... /api/auth/save 204 314 127.0.0.1 21:49:16 16... 8080
41 http://localhost:5000 PUT /api/auth/save ✓ 200 253 text 127.0.0.1 21:49:16 16... 8080
42 http://localhost:3000 GET /static/js/vendors~main.chunk.js 304 176 script js 127.0.0.1 21:49:21 16... 8080
43 http://localhost:3000 GET /static/js/vendors~main.chunk.js 304 176 script js 127.0.0.1 21:49:21 16... 8080
44 http://localhost:3000 GET /static/js/vendors~main.chunk.j... 304 176 map 127.0.0.1 21:49:21 16... 8080
45 http://localhost:3000 GET /static/js/vendors~main.chunk.j... 304 176 map 127.0.0.1 21:49:21 16... 8080
46 http://localhost:3000 GET / 304 173 127.0.0.1 21:49:24 16... 8080
50 http://localhost:3000 GET /static/js/bundle.js 304 174 script js 127.0.0.1 21:49:24 16... 8080
51 http://localhost:3000 GET /static/js/main.chunk.js 304 175 script js 127.0.0.1 21:49:24 16... 8080
52 http://localhost:3000 GET /static/js/vendors~main.chunk.js 304 176 script js 127.0.0.1 21:49:24 16... 8080
53 http://localhost:3000 GET /manifest.json 304 237 script json 127.0.0.1 21:49:25 16... 8080
54 http://localhost:3000 GET /sockjs-node 101 129 127.0.0.1 21:49:25 16... 8080

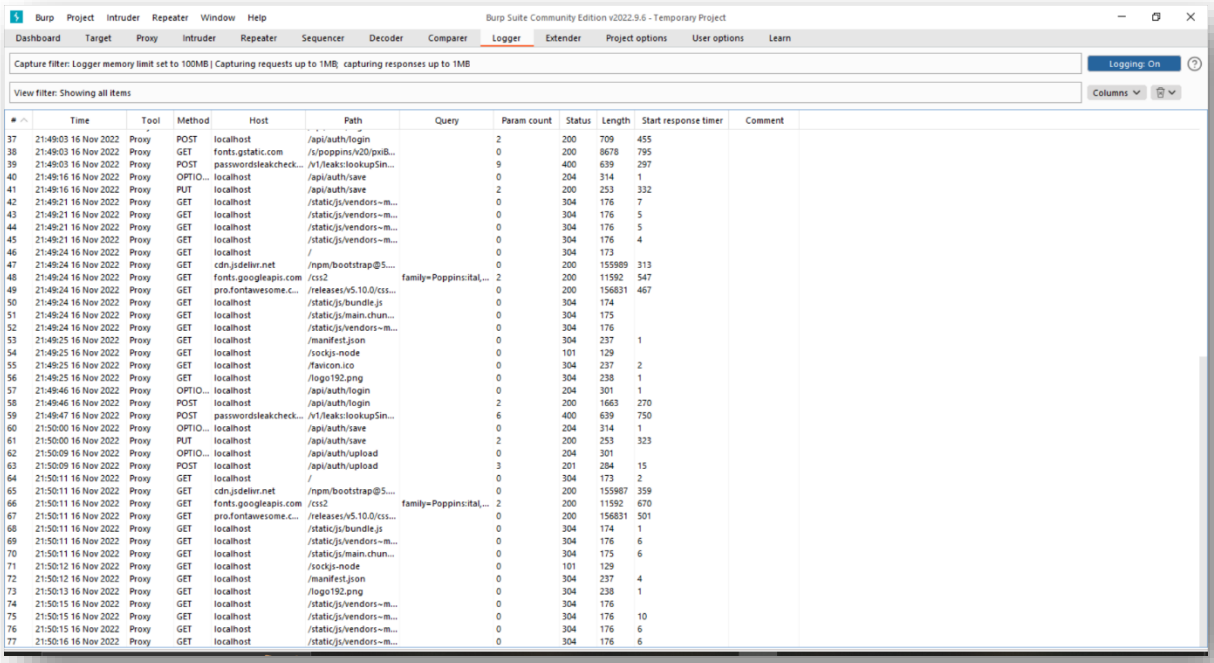

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Intercept HTTP history WebSockets history Options
Filter: Showing all items
URL Direction Edited Length Comment TLS Time Listener port WebSocket ID
1 http://localhost:3000/sockjs-node ← To client 34 21:47:32 16... 8080 1
2 http://localhost:3000/sockjs-node ← To client 14 21:47:32 16... 8080 1
3 http://localhost:3000/sockjs-node ← To client 21 21:47:32 16... 8080 1
4 http://localhost:3000/sockjs-node ← To client 45 21:47:32 16... 8080 1
5 http://localhost:3000/sockjs-node ← To client 957 21:47:32 16... 8080 1
6 http://localhost:3000/sockjs-node ← To client 34 21:48:55 16... 8080 2
7 http://localhost:3000/sockjs-node ← To client 14 21:48:55 16... 8080 2
8 http://localhost:3000/sockjs-node ← To client 21 21:48:55 16... 8080 2
9 http://localhost:3000/sockjs-node ← To client 45 21:48:55 16... 8080 2
10 http://localhost:3000/sockjs-node ← To client 957 21:48:55 16... 8080 2
11 http://localhost:3000/sockjs-node ← To client 34 21:49:25 16... 8080 3
12 http://localhost:3000/sockjs-node ← To client 14 21:49:25 16... 8080 3
13 http://localhost:3000/sockjs-node ← To client 21 21:49:25 16... 8080 3
14 http://localhost:3000/sockjs-node ← To client 45 21:49:25 16... 8080 3
15 http://localhost:3000/sockjs-node ← To client 957 21:49:25 16... 8080 3
16 http://localhost:3000/sockjs-node ← To client 34 21:50:12 16... 8080 4
17 http://localhost:3000/sockjs-node ← To client 14 21:50:12 16... 8080 4
18 http://localhost:3000/sockjs-node ← To client 21 21:50:12 16... 8080 4
19 http://localhost:3000/sockjs-node ← To client 45 21:50:12 16... 8080 4
20 http://localhost:3000/sockjs-node ← To client 957 21:50:12 16... 8080 4

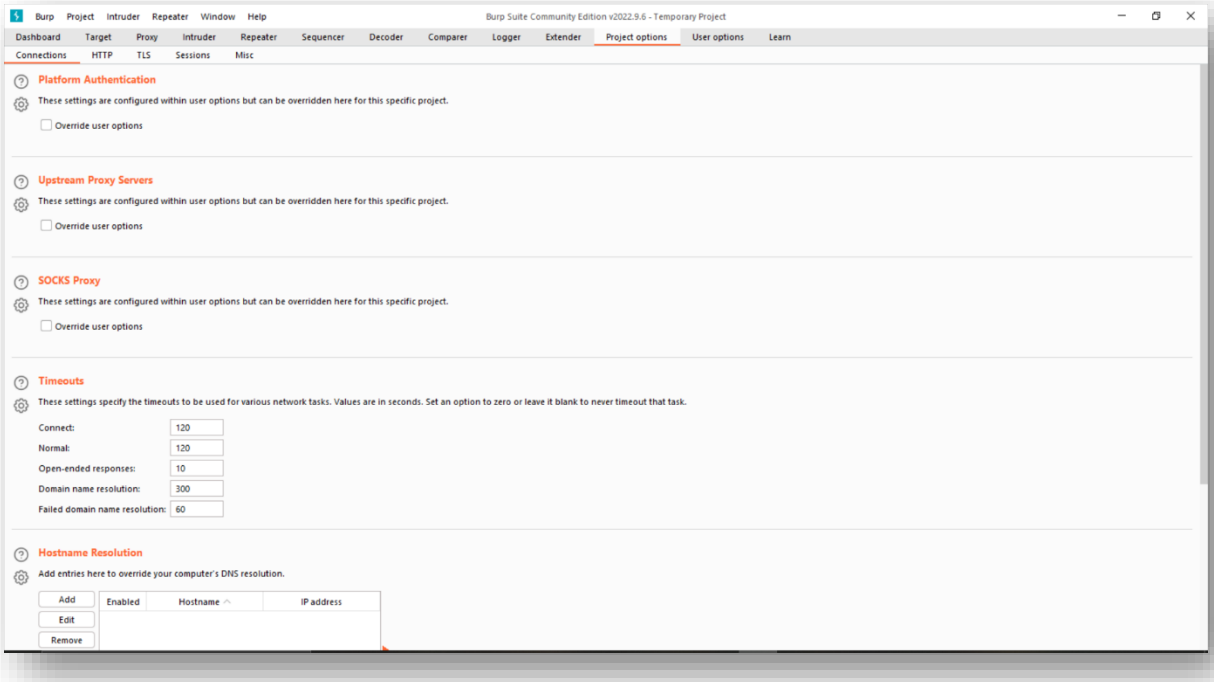

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Intercept HTTP history WebSockets history Options
Proxy Listeners
Burp Proxy uses listeners to receive incoming HTTP requests from your browser. You will need to configure your browser to use one of the listeners as its proxy server.
Add
Edit
Remove
Running Interface Invisible Redirect Certificate TLS Protocols
127.0.0.1:8080 Per-host Default
Each installation of Burp generates its own CA certificate that Proxy listeners can use when negotiating TLS connections. You can import or export this certificate for use in other tools or another installation of Burp.
Import / export CA certificate
Regenerate CA certificate
Intercept Client Requests
Use these settings to control which requests are stalled for viewing and editing in the Intercept tab.
Intercept requests based on the following rules: Master interception is turned off
Add
Edit
Remove
Up
Down
Enabled Operator Match type Relationship Condition
File extension Does not match (^gif$|^jpg$|^png$|^css$|^js$|...
Or Request Contains parameters
Or HTTP method Does not match (get|post)
And URL Is in target scope
Automatically fix missing or superfluous new lines at end of request
Automatically update Content-Length header when the request is edited
Intercept Server Responses
Use these settings to control which responses are stalled for viewing and editing in the Intercept tab.
Intercept responses based on the following rules: Master interception is turned off

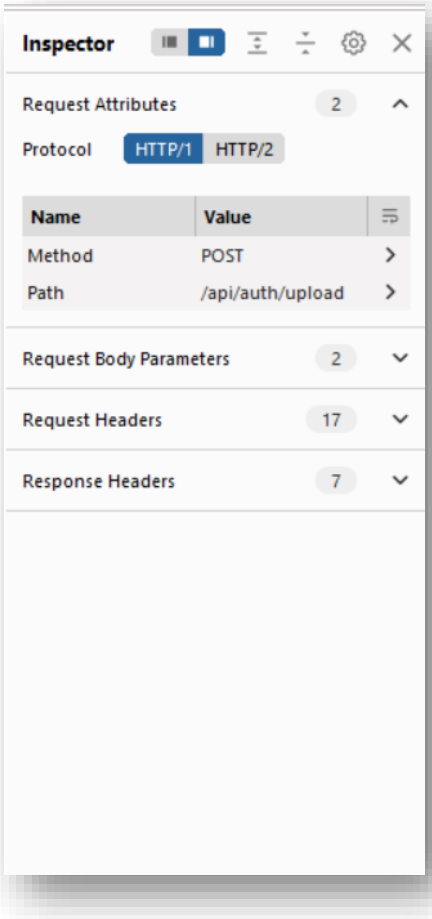

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Site map Scope Issue definitions
Issue Definitions
This listing contains the definitions of all issues that can be detected by Burp Scanner.
Name Typical severity Type index
OS command injection High 0x00100100
SQL injection High 0x00100200
SQL injection (second order) High 0x00100210
ASP.NET tracing enabled High 0x00100280
File path traversal High 0x00100300
XML external entity injection High 0x00100400
LDAP injection High 0x00100500
XPath injection High 0x00100600
XML injection Medium 0x00100700
ASP.NET debugging enabled Medium 0x00100800
HTTP PUT method is enabled High 0x00100900
Out-of-band resource load (HTTP) High 0x00100a00
File path manipulation High 0x00100b00
PHP code injection High 0x00100c00
Server-side JavaScript code injection High 0x00100d00
Perl code injection High 0x00100e00
Ruby code injection High 0x00100f00
Python code injection High 0x00100f10
Expression Language injection High 0x00100f20
Unidentified code injection High 0x00101000
Server-side template injection High 0x00101080
SSI injection High 0x00101100
Cross-site scripting (stored) High 0x00200100
HTTP request smuggling High 0x00200140
Client-side desync High 0x00200141
Web cache poisoning High 0x00200180
HTTP response header injection High 0x00200200
Cross-site scripting (reflected) High 0x00200300
Client-side template injection High 0x00200308
Cross-site scripting (DOM-based) High 0x00200310
Cross-site scripting (reflected DOM-based) High 0x00200311
Cross-site scripting (stored DOM-based) High 0x00200312
Client-side prototype pollution Information 0x00200316
JavaScript injection (DOM-based) High 0x00200320
JavaScript injection (reflected DOM-based) High 0x00200321
JavaScript injection (stored DOM-based) High 0x00200322
Path-relative style sheet import Information 0x00200328
Client-side SQL injection (DOM-based) High 0x00200330
Client-side SQL injection (reflected DOM-based) High 0x00200331
Client-side SQL injection (stored DOM-based) High 0x00200332
OS command injection
Description
Operating system command injection vulnerabilities arise when an application incorporates user-controllable data into a command that is processed by a shell command interpreter. If the user data is not strictly validated, an attacker can use shell metacharacters to modify the command that is executed, and inject arbitrary further commands that will be executed by the server.
OS command injection vulnerabilities are usually very serious and may lead to compromise of the server hosting the application, or of the application's own data and functionality. It may also be possible to use the server as a platform for attacks against other systems. The exact potential for exploitation depends upon the security context in which the command is executed, and the privileges that this context has regarding sensitive resources on the server.
Remediation
If possible, applications should avoid incorporating user-controllable data into operating system commands. In almost every situation, there are safer alternative methods of performing server-level tasks, which cannot be manipulated to perform additional commands than the one intended.
If it is considered unavoidable to incorporate user-supplied data into operating system commands, the following two layers of defense should be used to prevent attacks:
The user data should be strictly validated. Ideally, a whitelist of specific accepted values should be used. Otherwise, only short alphanumeric strings should be accepted. Input containing any other data, including any conceivable shell metacharacter or whitespace, should be rejected.
The application should use command APIs that launch a specific process via its name and command-line parameters, rather than passing a command string to a shell interpreter that supports command chaining and redirection. For example, the Java API Runtime.exec and the ASP.NET API Process.Start do not support shell metacharacters. This defense can mitigate the impact of an attack even in the event that an attacker circumvents the input validation defenses.
References
Web Security Academy: OS command injection
Vulnerability classifications
CWE-77: Improper Neutralization of Special Elements used in a Command ('Command Injection')
CWE-78: Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
CWE-116: Improper Encoding or Escaping of Output
CAPEC-248: Command Injection
Typical severity
High

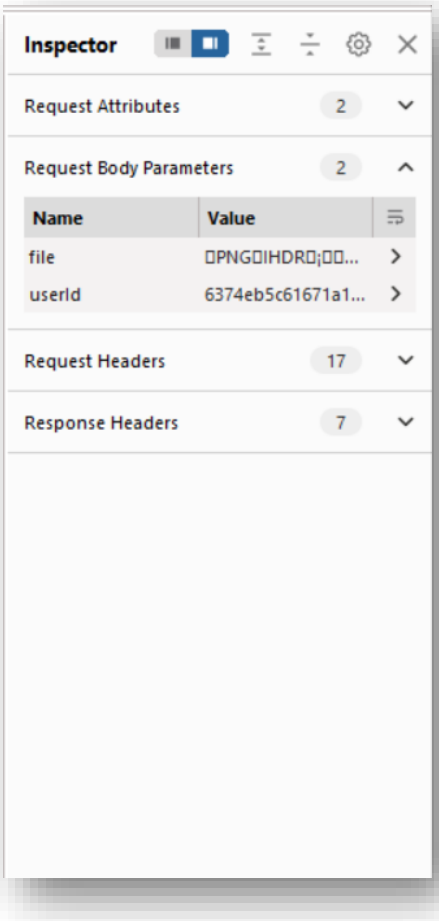

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Live capture Manual load Analysis options
Select Live Capture Request
Send requests here from other tools to configure a live capture. Select the request to use, configure the other options below, then click "Start live capture".
Remove
Clear
Host Request
Start live capture
Token Location Within Response
Select the location in the response where the token appears.
Cookie:
Form field:
Custom location:
Configure
Live Capture Options
These settings control the engine used for making HTTP requests and harvesting tokens when performing the live capture.
Number of threads: 5
Throttle between requests (milliseconds): 0
Ignore tokens whose length deviates by 5 characters

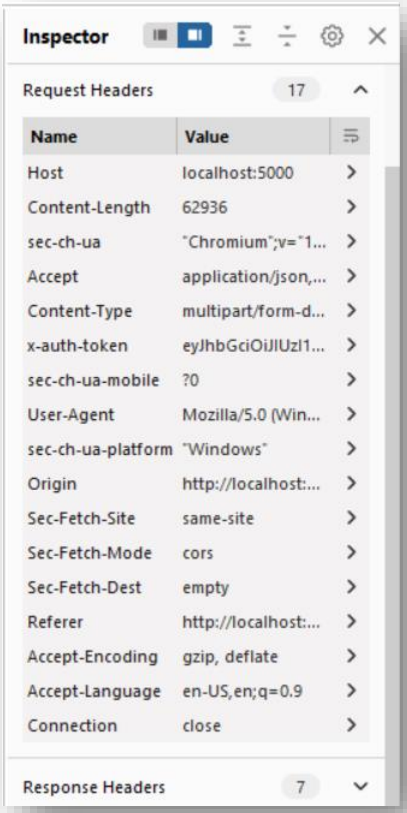

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Capture filter: Logger memory limit set to 100MB | Capturing requests up to 1MB, capturing responses up to 1MB
Logging: On
View filter: Showing all items
Columns
Time Tool Method Host Path Query Param count Status Length Start response timer Comment
1 21:47:30 16 Nov 2022 Proxy GET localhost / 0 200 2874 3
2 21:47:30 16 Nov 2022 Proxy GET localhost / 0 200 2874 3
3 21:47:31 16 Nov 2022 Proxy GET fonts.googleapis.com /css2 family=Poppins:ital,... 2 200 11592 207
4 21:47:31 16 Nov 2022 Proxy GET cdn.jsdelivr.net /npm/bootstrap@5.... 0 200 155985 362
5 21:47:31 16 Nov 2022 Proxy GET pro.fontawesome.c... /releases/v5.10.0/css... 0 200 156831 83
6 21:47:31 16 Nov 2022 Proxy GET cdn.jsdelivr.net /npm/bootstrap@5.... 0 200 81556 204
7 21:47:31 16 Nov 2022 Proxy GET localhost /static/js/bundle.js 0 200 39042
9 21:47:32 16 Nov 2022 Proxy GET localhost /static/js/main.chun... 0 200 129111
10 21:47:32 16 Nov 2022 Proxy GET localhost /sockjs-node 0 101 129
11 21:47:32 16 Nov 2022 Proxy GET fonts.gstatic.com /s/poppins/v20/pxiB... 0 200 8562 411
12 21:47:32 16 Nov 2022 Proxy GET fonts.gstatic.com /s/poppins/v20/pxiE... 0 200 8814 355
13 21:47:33 16 Nov 2022 Proxy GET localhost /manifest.json 0 304 237
14 21:47:34 16 Nov 2022 Proxy GET localhost /logo192.png 0 304 238
15 21:47:45 16 Nov 2022 Proxy OPTIO... localhost /api/auth/login 0 204 301
16 21:47:45 16 Nov 2022 Proxy POST localhost /api/auth/login 2 200 1588 301
17 21:47:46 16 Nov 2022 Proxy POST passwordsleakcheck... /v1/leaks:lookupSin... 8 400 639 385
18 21:48:32 16 Nov 2022 Proxy OPTIO... localhost /api/auth/signup 0 204 314
19 21:48:33 16 Nov 2022 Proxy POST localhost /api/auth/signup 5 200 522 268
20 21:48:50 16 Nov 2022 Proxy POST passwordsleakcheck... /v1/leaks:lookupSin... 4 400 639 903
21 21:48:50 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176
22 21:48:50 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 6
23 21:48:50 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 9
24 21:48:50 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176
25 21:48:54 16 Nov 2022 Proxy GET localhost / 0 304 173
26 21:48:54 16 Nov 2022 Proxy GET cdn.jsdelivr.net /npm/bootstrap@5.... 0 200 155991 330
27 21:48:54 16 Nov 2022 Proxy GET fonts.googleapis.com /css2 family=Poppins:ital,... 2 200 11592 562
28 21:48:54 16 Nov 2022 Proxy GET pro.fontawesome.c... /releases/v5.10.0/css... 0 200 156831 474
29 21:48:54 16 Nov 2022 Proxy GET localhost /static/js/bundle.js 0 304 174
30 21:48:55 16 Nov 2022 Proxy GET localhost /static/js/main.chun... 0 304 175 11
31 21:48:55 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 9
32 21:48:55 16 Nov 2022 Proxy GET localhost /sockjs-node 0 101 129
33 21:48:55 16 Nov 2022 Proxy GET localhost /manifest.json 0 304 237
34 21:48:55 16 Nov 2022 Proxy GET localhost /favicon.ico 0 304 237
35 21:48:56 16 Nov 2022 Proxy GET localhost /logo192.png 0 304 238 1
36 21:49:02 16 Nov 2022 Proxy OPTIO... localhost /api/auth/login 0 204 301
37 21:49:03 16 Nov 2022 Proxy POST localhost /api/auth/login 2 200 709 455
38 21:49:03 16 Nov 2022 Proxy GET fonts.gstatic.com /s/poppins/v20/pxiB... 0 200 8678 795
39 21:49:03 16 Nov 2022 Proxy POST passwordsleakcheck... /v1/leaks:lookupSin... 9 400 639 297
40 21:49:16 16 Nov 2022 Proxy OPTIO... localhost /api/auth/save 0 204 314 1
41 21:49:16 16 Nov 2022 Proxy PUT localhost /api/auth/save 2 200 253 332
42 21:49:21 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 7

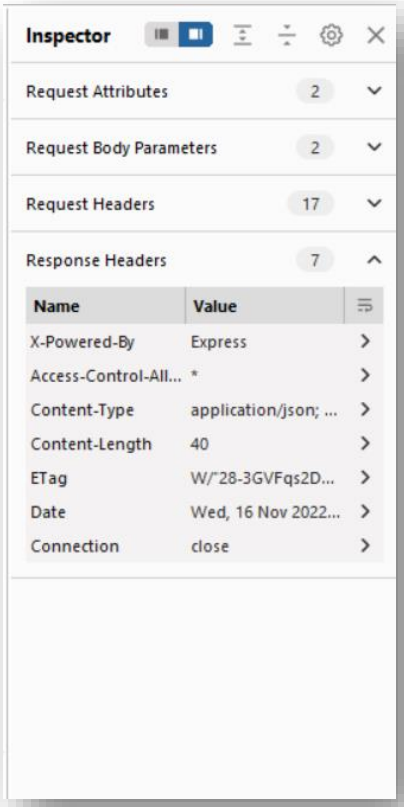

Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Capture filter: Logger memory limit set to 100MB | Capturing requests up to 1MB; capturing responses up to 1MB
Logging: On
View filter: Showing all items
Columns
Time Tool Method Host Path Query Param count Status Length Start response timer Comment
37 21:49:03 16 Nov 2022 Proxy POST localhost /api/auth/login 2 200 709 455
38 21:49:03 16 Nov 2022 Proxy GET fonts.gstatic.com /s/poppins/v20/pxiB... 0 200 8678 795
39 21:49:03 16 Nov 2022 Proxy POST passwordsleakcheck... /v1/leaks:lookupSin... 9 400 639 297
40 21:49:16 16 Nov 2022 Proxy OPTIO... localhost /api/auth/save 0 204 314 1
41 21:49:16 16 Nov 2022 Proxy PUT localhost /api/auth/save 2 200 253 332
42 21:49:21 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 7
43 21:49:21 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 5
44 21:49:21 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 5
45 21:49:21 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 4
46 21:49:24 16 Nov 2022 Proxy GET localhost / 0 304 173
47 21:49:24 16 Nov 2022 Proxy GET cdn.jsdelivr.net /npm/bootstrap@5.... 0 200 155989 313
48 21:49:24 16 Nov 2022 Proxy GET fonts.googleapis.com /css2 family=Poppins:ital,... 2 200 11592 547
49 21:49:24 16 Nov 2022 Proxy GET pro.fontawesome.c... /releases/v5.10.0/css... 0 200 156831 467
50 21:49:24 16 Nov 2022 Proxy GET localhost /static/js/bundle.js 0 304 174
51 21:49:24 16 Nov 2022 Proxy GET localhost /static/js/main.chun... 0 304 175
52 21:49:24 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176
53 21:49:25 16 Nov 2022 Proxy GET localhost /manifest.json 0 304 237 1
54 21:49:25 16 Nov 2022 Proxy GET localhost /sockjs-node 0 101 129
55 21:49:25 16 Nov 2022 Proxy GET localhost /favicon.ico 0 304 237 2
56 21:49:25 16 Nov 2022 Proxy GET localhost /logo192.png 0 304 238 1
57 21:49:46 16 Nov 2022 Proxy OPTIO... localhost /api/auth/login 0 204 301 1
58 21:49:46 16 Nov 2022 Proxy POST localhost /api/auth/login 2 200 1663 270
59 21:49:47 16 Nov 2022 Proxy POST passwordsleakcheck... /v1/leaks:lookupSin... 6 400 639 750
60 21:50:00 16 Nov 2022 Proxy OPTIO... localhost /api/auth/save 0 204 314 1
61 21:50:00 16 Nov 2022 Proxy PUT localhost /api/auth/save 2 200 253 323
62 21:50:09 16 Nov 2022 Proxy OPTIO... localhost /api/auth/upload 0 204 301
63 21:50:09 16 Nov 2022 Proxy POST localhost /api/auth/upload 3 201 284 15
64 21:50:11 16 Nov 2022 Proxy GET localhost / 0 304 173 2
65 21:50:11 16 Nov 2022 Proxy GET cdn.jsdelivr.net /npm/bootstrap@5.... 0 200 155987 359
66 21:50:11 16 Nov 2022 Proxy GET fonts.googleapis.com /css2 family=Poppins:ital,... 2 200 11592 670
67 21:50:11 16 Nov 2022 Proxy GET pro.fontawesome.c... /releases/v5.10.0/css... 0 200 156831 501
68 21:50:11 16 Nov 2022 Proxy GET localhost /static/js/bundle.js 0 304 174 1
69 21:50:11 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 6
70 21:50:11 16 Nov 2022 Proxy GET localhost /static/js/main.chun... 0 304 175 6
71 21:50:12 16 Nov 2022 Proxy GET localhost /sockjs-node 0 101 129
72 21:50:12 16 Nov 2022 Proxy GET localhost /manifest.json 0 304 237 4
73 21:50:13 16 Nov 2022 Proxy GET localhost /logo192.png 0 304 238 1
74 21:50:15 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176
75 21:50:15 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 10
76 21:50:15 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 6
77 21:50:16 16 Nov 2022 Proxy GET localhost /static/js/vendors~m... 0 304 176 6


Burp Project Intruder Repeater Window Help
Burp Suite Community Edition v2022.9.6 - Temporary Project
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Connections HTTP TLS Sessions Misc
Platform Authentication
These settings are configured within user options but can be overridden here for this specific project.
Override user options
Upstream Proxy Servers
These settings are configured within user options but can be overridden here for this specific project.
Override user options
SOCKS Proxy
These settings are configured within user options but can be overridden here for this specific project.
Override user options
Timeouts
These settings specify the timeouts to be used for various network tasks. Values are in seconds. Set an option to zero or leave it blank to never timeout that task.
Connect: 120
Normal: 120
Open-ended responses: 10
Domain name resolution: 300
Failed domain name resolution: 60
Hostname Resolution
Add entries here to override your computer's DNS resolution.
Add
Edit
Remove
Enabled Hostname IP address


Inspector
Request Attributes 2
Protocol HTTP/1 HTTP/2
Name Value
Method POST
Path /api/auth/upload
Request Body Parameters 2
Request Headers 17
Response Headers 7

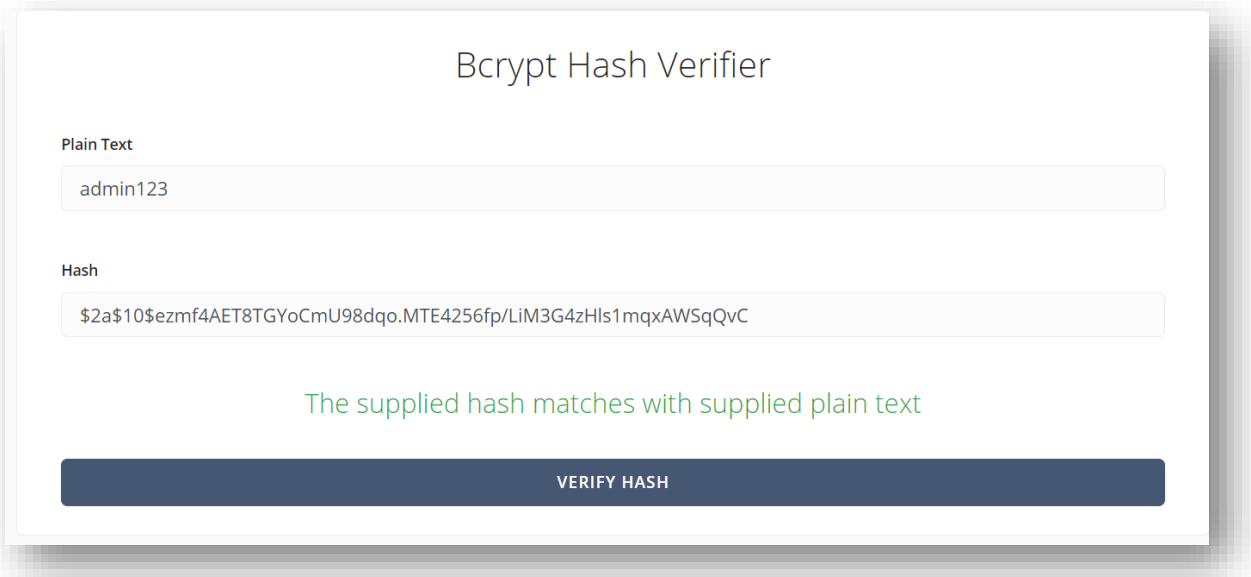

Inspector
Request Attributes 2
Request Body Parameters 2
Name Value
file □PNG□IHDR□¡□□...
userId 6374eb5c61671a1...
Request Headers 17
Response Headers 7

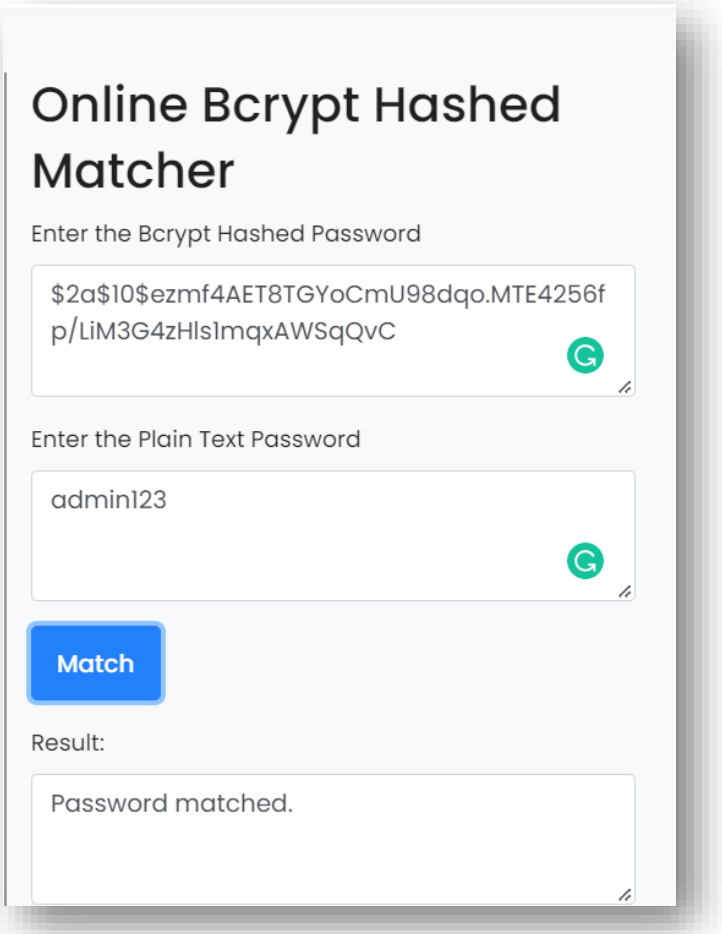

Inspector
Request Headers 17
Name Value
Host localhost:5000
Content-Length 62936
sec-ch-ua "Chromium";v="1...
Accept application/json,...
Content-Type multipart/form-d...
x-auth-token eyJhbGciOiJIUzI1...
sec-ch-ua-mobile ?0
User-Agent Mozilla/5.0 (Win...
sec-ch-ua-platform "Windows"
Origin http://localhost:...
Sec-Fetch-Site same-site
Sec-Fetch-Mode cors
Sec-Fetch-Dest empty
Referer http://localhost:...
Accept-Encoding gzip, deflate
Accept-Language en-US,en;q=0.9
Connection close
Response Headers 7


Inspector
Request Attributes 2
Request Body Parameters 2
Request Headers 17
Response Headers 7
Name Value
X-Powered-By Express
Access-Control-All... *
Content-Type application/json; ...
Content-Length 40
ETag W/"28-3GVFqs2D...
Date Wed, 16 Nov 2022...
Connection close

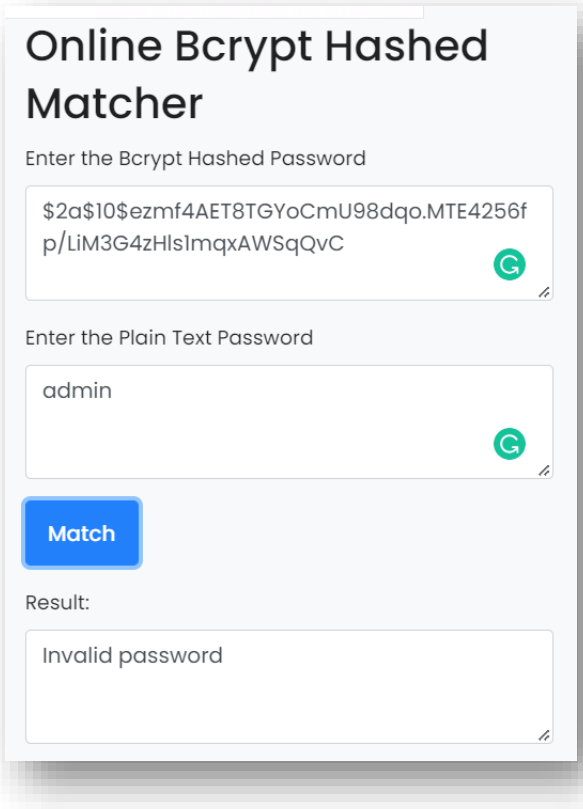
Password Hashing Testing

Valid Match

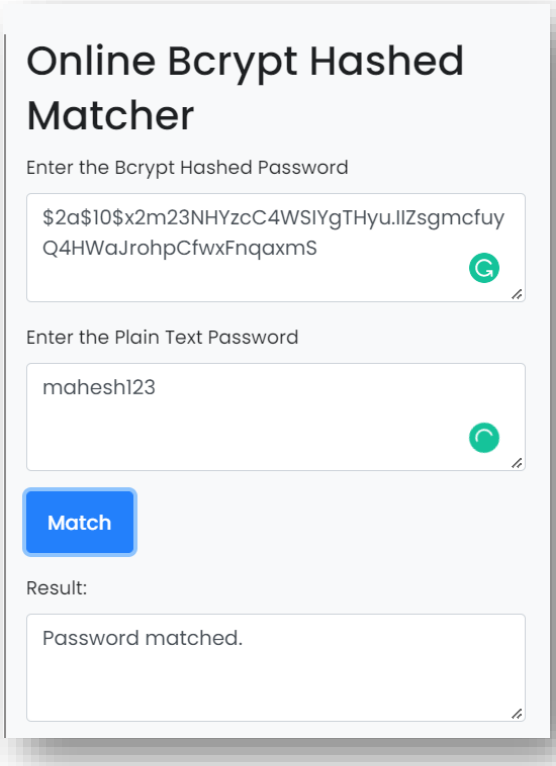




Invalid Match



Valid Match



Invalid Match