URL shortener (Engineer 3)

Owner: Engineer 3
Reviewer:
Contributors:
Date Generated: Sat Dec 14 2024

OWASP Threat Dragon

Executive Summary

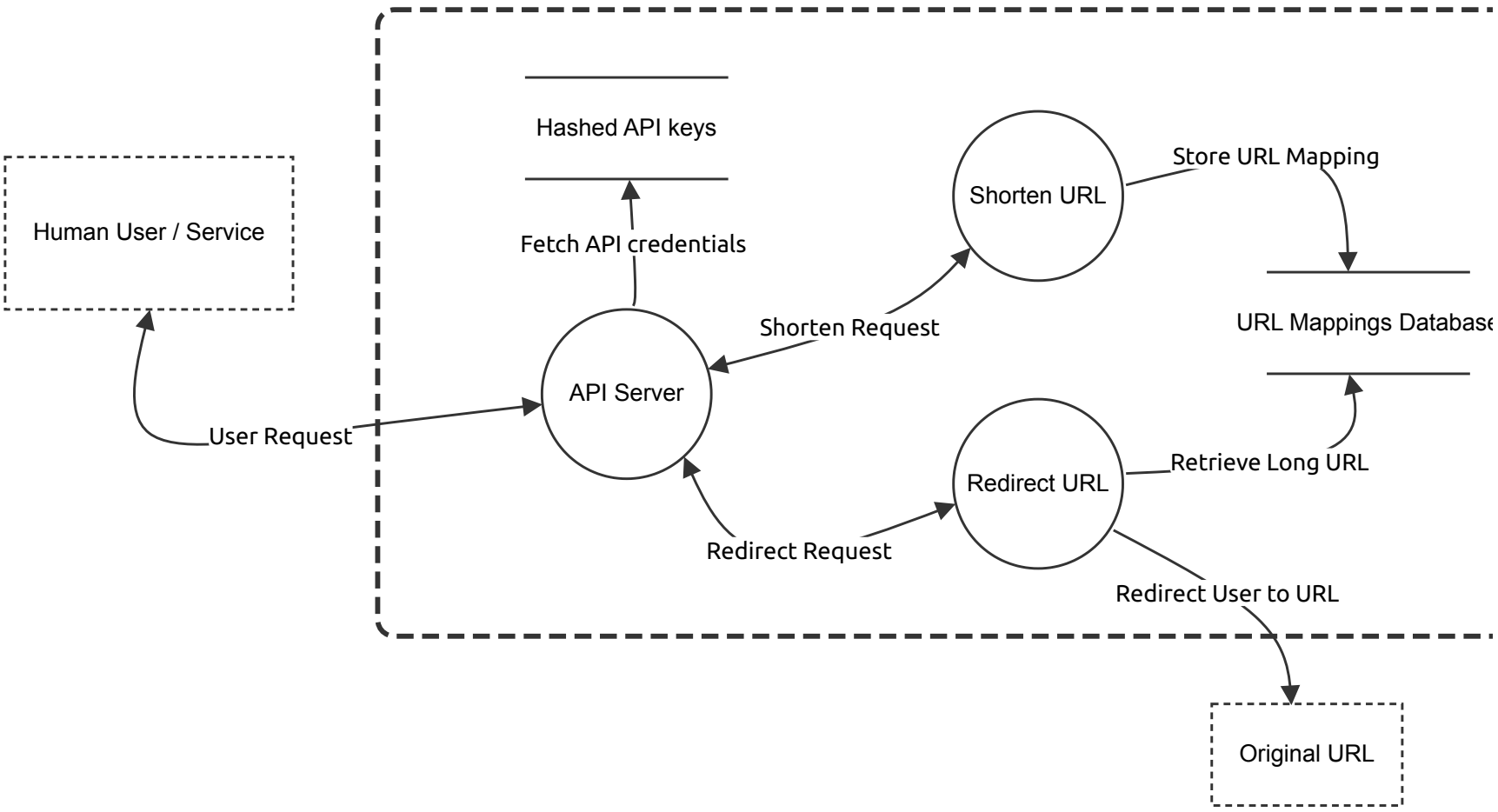
High level system description

The URL Shortener Service aims to provide a simple and efficient way for users to convert long URLs into short, easily shareable URLs. The service ensures functionality, reliability, and security while interacting with users and handling their data.

Summary

Total Threats	63
Total Mitigated	63
Not Mitigated	0
Open / High Priority	0
Open / Medium Priority	0
Open / Low Priority	0
Open / Unknown Priority	0

URL Shortener Service Architecture



URL Shortener Service Architecture

Human User / Service (Actor) - *Out of Scope*

Reason for out of scope: Incoming requests outside of application scope

Description: External services or human users may interact with the API for bulk operations or integrations.

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Redirect URL (Process)

Description: Enables users to resolve short URLs and be redirected to the corresponding long URLs via a dedicated API.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
157	Flooding	Denial of service	Medium	Mitigated		A service receives a rapid and vast number of requests and as a result becomes unusable/unavailable to the legitimate users and/or services. As the service doesn't limit the number of possible requests and/or doesn't handle throughput appropriately, an attacker can cause resource exhaustion that may result in system being unresponsive to authorized actors. Related attacks: Denial of Service, Distributed Denial of Service	A threat can be mitigated by implementing such mechanism as rate limiting.
56	Cache Poisoning	Tampering	Medium	Mitigated		An attacker may attempt to manipulate or inject incorrect data into caching mechanisms (e.g., DNS or HTTP caches) to redirect users to malicious sites instead of the intended long URLs.	A threat can be mitigated by implementing cache validation techniques, such as using cache integrity checks and ensuring that only trusted sources can update cache entries. Additionally, using secure protocols like HTTPS can help prevent cache poisoning.
57	Cross-Site Scripting	Tampering	Medium	Mitigated		An attacker might inject malicious scripts into the short URL service, which are then executed when users are redirected. This could lead to unauthorized actions or data being stolen from the user's session.	A threat can be mitigated by implementing strict input validation and output encoding to prevent the injection of malicious scripts. Additionally, employing Content Security Policy (CSP) headers can help restrict the sources from which scripts can be loaded.
60	URL Enumeration	Information disclosure	Medium	Mitigated		An attacker might attempt to discover valid short URLs through automated guessing or brute force methods, potentially leading to unauthorized access to sensitive URLs or information.	A threat can be mitigated by using sufficiently long and random identifiers for short URLs, making them difficult to predict. Implementing rate limiting and monitoring for unusual access patterns can also help detect and prevent enumeration attempts.

Shorten URL (Process)

Description: Allows users to submit long URLs and receive shortened URLs via a dedicated API.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
157	Flooding	Denial of service	Medium	Mitigated		A service receives a rapid and vast number of requests and as a result becomes unusable/unavailable to the legitimate users and/or services. As the service doesn't limit the number of possible requests and/or doesn't handle throughput appropriately, an attacker can cause resource exhaustion that may result in system being unresponsive to authorized actors. Related attacks: Denial of Service, Distributed Denial of Service	A threat can be mitigated by implementing such mechanism as rate limiting.
61	Malicious URL Submission	Tampering	Medium	Mitigated		An attacker could submit malicious or harmful long URLs with the intent of spreading malware or conducting phishing attacks through the shortened URLs.	A threat can be mitigated by implementing URL filtering and validation mechanisms that check submitted URLs against known malicious URL databases. Additionally, employing a content scanning service can help detect and block harmful URLs before they are shortened. Excessive Resource Consumption
63	Sensitive Information Exposure	Information disclosure	Medium	Mitigated		Users might accidentally submit URLs containing sensitive information (e.g., session tokens, user IDs), which could be exposed if not properly handled.	A threat can be mitigated by educating users on best practices for URL submission and implementing mechanisms to detect and redact sensitive information from URLs before storing or sharing them. Additionally, encrypting stored URLs can help protect sensitive data from unauthorized access.
64	URL Collision	Tampering	Medium	Mitigated		A poorly designed URL shortening algorithm might generate the same short URL for different long URLs, leading to incorrect redirection and potential data integrity issues.	A threat can be mitigated by using a robust algorithm that generates unique and collision-resistant short URLs. Employing a hash-based method with a sufficiently large keyspace can reduce the likelihood of collisions.
65	SQL Injection	Tampering	Medium	Mitigated		An attacker may attempt to exploit vulnerabilities in the API by injecting malicious SQL code through the URL submission process, potentially compromising the database.	A threat can be mitigated by using parameterized queries and prepared statements to interact with the database, thus preventing injection attacks. Regular security audits and code reviews can also help identify and remediate vulnerabilities.

URL Mappings Database (Store)

Description: Data store containing the mappings between short URLs and their corresponding long URLs.

DATA STORED: Long URL, Short URL, UserID, CreationDate

Number	Title	Type	Priority	Status	Score	Description	Mitigations
50	Data tampering (at rest/in transit)	Tampering	Medium	Mitigated		Information stored in the service and/or transmitted over the network can be changed by an unauthorized actor. This can be possible because there is no encryption of the data transmitted over the network and/or stored in the service or broken/weak cryptographic algorithm is being used. This can also include a scenario where authorized actor with malicious intent can manipulate data without being detected, as there is no integrity protection for the data implemented.	A threat is mitigated by using secure storage mechanisms: secure hashing algorithms.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
51	Deletion/Removal of data	Tampering	Medium	Mitigated		Data is deleted by an actor (authorized or unauthorized) and is hence not available to other authorized users. This may include, but is not limited to, clearing caches, removing stored files, whole databases, secrets, projects, artifacts, stacks. If the service doesn't have a backup system in place, deleted data might be irreversibly lost.	A threat can be mitigated by restricting access to the database and implementing mechanism such as database replication, to be able to operate in case of data removal.
52	Insider Threat	Tampering	Medium	Mitigated		Employees or individuals with access to the data store might misuse their privileges to steal, alter, or delete data, either intentionally or through negligence.	Threat can be mitigated by implementing proper authorization mechanism and data replication so system would be able to operate in case of data altering or data removal.

Original URL (Actor) - *Out of Scope*

Reason for out of scope: Original URL not in scope of URL shortener application

Description: Destination server of the original URL

Number	Title	Type	Priority	Status	Score	Description	Mitigations

API Server (Process)

Description: The API Server is the central component of the URL Shortener Service. It manages all interactions with users and external services, handling requests for both shortening URLs and redirecting from short URLs to their original long URLs.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
157	Flooding	Denial of service	Medium	Mitigated		A service receives a rapid and vast number of requests and as a result becomes unusable/unavailable to the legitimate users and/or services. As the service doesn't limit the number of possible requests and/or doesn't handle throughput appropriately, an attacker can cause resource exhaustion that may result in system being unresponsive to authorized actors. Related attacks: Denial of Service, Distributed Denial of Service	A threat can be mitigated by implementing such mechanism as rate limiting.
53	Unauthorized Access	Spoofing	Medium	Mitigated		Attackers may attempt to gain unauthorized access to the API Server to exploit its functionality, potentially creating, modifying, or deleting URL mappings without permission. This could lead to the creation of malicious short URLs or tampering with existing ones.	A threat is mitigated using API keys and enforcing access controls to ensure only authorized users can access and modify data.
54	URL Manipulation	Tampering	Medium	Mitigated		An attacker might manipulate short URLs to redirect users to malicious sites or alter the destination URL after it has been created.	A threat can be mitigated by implementing validation and integrity checks on URLs to ensure that they have not been altered.
55	Insufficient Logging and Monitoring	Repudiation	Medium	Mitigated		Provide a description for this threatWithout proper logging and monitoring, malicious activities might go undetected, and it could be difficult to trace actions back to specific users or events.	A threat can be mitigated by ensuring comprehensive logging of all API interactions and user activities, coupled with active monitoring and alerting systems to detect and respond to suspicious behavior promptly.

Hashed API keys (Store)

Description: Database containing hashed API keys for authentication purposes.

DATA STORED: Hashed API Key, UserID

Number	Title	Type	Priority	Status	Score	Description	Mitigations
40	Insecure storage of credentials	Information disclosure	Medium	Mitigated		The service stores sensitive information in an insecure manner, making it easier for an attacker to access and exploit the data. This can include storing credentials in plaintext, using weak encryption, or not using encryption at all.	A threat is mitigated by using secure storage mechanisms: secure hashing algorithms.
41	Data tampering (at rest/in transit)	Information disclosure	Medium	Mitigated		Information stored in the service and/or transmitted over the network can be changed by an unauthorized actor. This can be possible because there is no encryption of the data transmitted over the network and/or stored in the service or broken/weak cryptographic algorithm is being used. This can also include a scenario where authorized actor with malicious intent can manipulate data without being detected, as there is no integrity protection for the data implemented.	A threat is mitigated by using secure storage mechanisms: secure hashing algorithms.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
42	Deletion/Removal of data	Tampering	Medium	Mitigated		Data is deleted by an actor (authorized or unauthorized) and is hence not available to other authorized users. This may include, but is not limited to, clearing caches, removing stored files, whole databases, secrets, projects, artifacts, stacks. If the service doesn't have a backup system in place, deleted data might be irreversibly lost.	A threat can be mitigated by restricting access to the database and implementing mechanism such as database replication, to be able to operate in case of data removal.
44	Drop encryption level	Tampering	Medium	Mitigated		It is possible for an attacker to intentionally lower the encryption level, which in turn enables a successful attack against the encrypted data. This can involve an attacker forcing a communications channel to use clear text instead of strongly encrypted data. As a result, the malicious user could read the channel by sniffing, instead of going through the extra effort of trying to decrypt the data using brute force techniques. Related attacks: Traffic sniffing, Data tampering, Communication channel manipulation	A threat can be mitigated by implementing proper authorization mechanism.
48	Insider Threat	Tampering	Medium	Mitigated		Employees or individuals with access to the data store might misuse their privileges to steal, alter, or delete data, either intentionally or through negligence.	Threat can be mitigated by implementing proper authorization mechanism and data replication so system would be able to operate in case of data altering or data removal.

User Request (Data Flow)

Description: The User sends a request to the API Server to shorten a long URL or to get the redirect URL.

DATA EXCHANGED: Long URL, Short URL, API key

Number	Title	Type	Priority	Status	Score	Description	Mitigations
1	Traffic sniffing	Information disclosure	Medium	Mitigated		A malicious actor can intercept traffic between a service and another actor (human or machine). The attacker is able to read the content of the communication because the channel is not encrypted and/or is encrypted using weak encryption algorithms.\nRelated attacks: Man in the Middle, Adversary in the Middle.	A threat can be mitigated by using an encryption algorithm such as TLS.
11	Exploit of weak TLS algorithm	Information disclosure	Medium	Mitigated		*Added because using TLS was suggested as remediation for threat "Traffic sniffing"* A service protects data as it traverses through the network, but because it uses weak TLS algorithm, an attacker can gain access to the supposed secure communication channel and data that is being transmitted. Vulnerable/weak TLS algorithms include versions: TLS 1.0, TLS 1.1, TLS 1.2	A threat can be mitigated by using secure version of TLS.
18	Data tampering (at rest/in transit)	Information disclosure	Medium	Mitigated		Information stored in the service and/or transmitted over the network can be changed by an unauthorized actor. This can be possible because there is no encryption of the data transmitted over the network and/or stored in the service or broken/weak cryptographic algorithm is being used. This can also include a scenario where authorized actor with malicious intent can manipulate data without being detected, as there is no integrity protection for the data implemented.	A threat can be mitigated by using an encryption algorithm such as TLS.
25	Parameter tampering	Tampering	Medium	Mitigated		An attacker is able to modify parameters over a trusted boundary. If there is no parameter validation and the channel is unencrypted, an attacker can tamper with the parameters being exchanged in the request or path.	A threat can be mitigated by using an encryption algorithm such as TLS.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
32	Manipulation of communication channel	Tampering	Medium	Mitigated		A malicious actor is able to manipulate the communication mechanism to impersonate other users, gain access to sensitive information, or carry out additional attacks. This threat targets the flawed assumptions about the mechanism used, incorrect protocol implementations, or the utilization of vulnerable protocols. It can also include manipulating a setting or parameter on the communications channel. Related attack: Traffic sniffing	A threat can be mitigated by using an encryption algorithm such as TLS.
39	Retransmission of network traffic	Tampering	Medium	Mitigated		Provide a description for this threatAn attacker is able to sniff network traffic and replay and/or delay sending the correct data without detection in order to cause unexpected behavior or perform unauthorized actions. Related attacks: Replayed request attack.	A threat can be mitigated by using an encryption algorithm such as TLS.

Fetch API credentials (Data Flow)

Description: Hashed API keys are pulled to authenticate the user before performing any operation.

DATA EXCHANGED: Hashed API key

Number	Title	Type	Priority	Status	Score	Description	Mitigations
2	Traffic sniffing	Information disclosure	Medium	Mitigated		A malicious actor can intercept traffic between a service and another actor (human or machine). The attacker is able to read the content of the communication because the channel is not encrypted and/or is encrypted using weak encryption algorithms.\nRelated attacks: Man in the Middle, Adversary in the Middle.	A threat can be mitigated by using an encryption algorithm such as TLS.
12	Exploit of weak TLS algorithm	Information disclosure	Medium	Mitigated		*Added because using TLS was suggested as remediation for threat "Traffic sniffing"* A service protects data as it traverses through the network, but because it uses weak TLS algorithm, an attacker can gain access to the supposed secure communication channel and data that is being transmitted. Vulnerable/weak TLS algorithms include versions: TLS 1.0, TLS 1.1, TLS 1.2	A threat can be mitigated by using secure version of TLS.
20	Data tampering (at rest/in transit)	Information disclosure	Medium	Mitigated		Information stored in the service and/or transmitted over the network can be changed by an unauthorized actor. This can be possible because there is no encryption of the data transmitted over the network and/or stored in the service or broken/weak cryptographic algorithm is being used. This can also include a scenario where authorized actor with malicious intent can manipulate data without being detected, as there is no integrity protection for the data implemented.	A threat can be mitigated by using an encryption algorithm such as TLS.
26	Parameter tampering	Tampering	Medium	Mitigated		An attacker is able to modify parameters over a trusted boundary. If there is no parameter validation and the channel is unencrypted, an attacker can tamper with the parameters being exchanged in the request or path.	A threat can be mitigated by using an encryption algorithm such as TLS.
33	Manipulation of communication channel	Tampering	Medium	Mitigated		A malicious actor is able to manipulate the communication mechanism to impersonate other users, gain access to sensitive information, or carry out additional attacks. This threat targets the flawed assumptions about the mechanism used, incorrect protocol implementations, or the utilization of vulnerable protocols. It can also include manipulating a setting or parameter on the communications channel. Related attack: Traffic sniffing	A threat can be mitigated by using an encryption algorithm such as TLS.
40	Retransmission of network traffic	Tampering	Medium	Mitigated		Provide a description for this threatAn attacker is able to sniff network traffic and replay and/or delay sending the correct data without detection in order to cause unexpected behavior or perform unauthorized actions. Related attacks: Replayed request attack.	A threat can be mitigated by using an encryption algorithm such as TLS.

Shorten Request (Data Flow)

Description: The user submits a long URL to the API for shortening.

DATA EXCHANGED: Short URL, Long URL, UserID

Number	Title	Type	Priority	Status	Score	Description	Mitigations
3	Traffic sniffing	Information disclosure	Medium	Mitigated		A malicious actor can intercept traffic between a service and another actor (human or machine). The attacker is able to read the content of the communication because the channel is not encrypted and/or is encrypted using weak encryption algorithms.\nRelated attacks: Man in the Middle, Adversary in the Middle.	A threat can be mitigated by using an encryption algorithm such as TLS.
13	Exploit of weak TLS algorithm	Information disclosure	Medium	Mitigated		*Added because using TLS was suggested as remediation for threat "Traffic sniffing"* A service protects data as it traverses through the network, but because it uses weak TLS algorithm, an attacker can gain access to the supposed secure communication channel and data that is being transmitted. Vulnerable/weak TLS algorithms include versions: TLS 1.0, TLS 1.1, TLS 1.2	A threat can be mitigated by using secure version of TLS.
21	Data tampering (at rest/in transit)	Information disclosure	Medium	Mitigated		Information stored in the service and/or transmitted over the network can be changed by an unauthorized actor. This can be possible because there is no encryption of the data transmitted over the network and/or stored in the service or broken/weak cryptographic algorithm is being used. This can also include a scenario where authorized actor with malicious intent can manipulate data without being detected, as there is no integrity protection for the data implemented.	A threat can be mitigated by using an encryption algorithm such as TLS.
28	Parameter tampering	Tampering	Medium	Mitigated		An attacker is able to modify parameters over a trusted boundary. If there is no parameter validation and the channel is unencrypted, an attacker can tamper with the parameters being exchanged in the request or path.	A threat can be mitigated by using an encryption algorithm such as TLS.
34	Manipulation of communication channel	Tampering	Medium	Mitigated		A malicious actor is able to manipulate the communication mechanism to impersonate other users, gain access to sensitive information, or carry out additional attacks. This threat targets the flawed assumptions about the mechanism used, incorrect protocol implementations, or the utilization of vulnerable protocols. It can also include manipulating a setting or parameter on the communications channel. Related attack: Traffic sniffing	A threat can be mitigated by using an encryption algorithm such as TLS.
42	Retransmission of network traffic	Tampering	Medium	Mitigated		Provide a description for this threatAn attacker is able to sniff network traffic and replay and/or delay sending the correct data without detection in order to cause unexpected behavior or perform unauthorized actions. Related attacks: Replayed request attack.	A threat can be mitigated by using an encryption algorithm such as TLS.

Redirect Request (Data Flow)

Description: The user submits a short URL to the API and gets redirected to the original Long URL.

DATA EXCHANGED: Short URL, Long URL

Number	Title	Type	Priority	Status	Score	Description	Mitigations
5	Traffic sniffing	Information disclosure	Medium	Mitigated		A malicious actor can intercept traffic between a service and another actor (human or machine). The attacker is able to read the content of the communication because the channel is not encrypted and/or is encrypted using weak encryption algorithms.\nRelated attacks: Man in the Middle, Adversary in the Middle.	A threat can be mitigated by using an encryption algorithm such as TLS.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
14	Exploit of weak TLS algorithm	Information disclosure	Medium	Mitigated		*Added because using TLS was suggested as remediation for threat "Traffic sniffing"* A service protects data as it traverses through the network, but because it uses weak TLS algorithm, an attacker can gain access to the supposed secure communication channel and data that is being transmitted. Vulnerable/weak TLS algorithms include versions: TLS 1.0, TLS 1.1, TLS 1.2	A threat can be mitigated by using secure version of TLS.
19	Data tampering (at rest/in transit)	Information disclosure	Medium	Mitigated		Information stored in the service and/or transmitted over the network can be changed by an unauthorized actor. This can be possible because there is no encryption of the data transmitted over the network and/or stored in the service or broken/weak cryptographic algorithm is being used. This can also include a scenario where authorized actor with malicious intent can manipulate data without being detected, as there is no integrity protection for the data implemented.	A threat can be mitigated by using an encryption algorithm such as TLS.
27	Parameter tampering	Tampering	Medium	Mitigated		An attacker is able to modify parameters over a trusted boundary. If there is no parameter validation and the channel is unencrypted, an attacker can tamper with the parameters being exchanged in the request or path.	A threat can be mitigated by using an encryption algorithm such as TLS.
35	Manipulation of communication channel	Tampering	Medium	Mitigated		A malicious actor is able to manipulate the communication mechanism to impersonate other users, gain access to sensitive information, or carry out additional attacks. This threat targets the flawed assumptions about the mechanism used, incorrect protocol implementations, or the utilization of vulnerable protocols. It can also include manipulating a setting or parameter on the communications channel. Related attack: Traffic sniffing	A threat can be mitigated by using an encryption algorithm such as TLS.
41	Retransmission of network traffic	Tampering	Medium	Mitigated		Provide a description for this threatAn attacker is able to sniff network traffic and replay and/or delay sending the correct data without detection in order to cause unexpected behavior or perform unauthorized actions. Related attacks: Replayed request attack.	A threat can be mitigated by using an encryption algorithm such as TLS.

Store URL Mapping (Data Flow)

Description: The API Server processes the request for shortening a URL, generates a short URL, and stores the mapping between the short URL and the long URL in the URL Database.

DATA EXCHANGED: Short URL, Long URL

Number	Title	Type	Priority	Status	Score	Description	Mitigations
8	Traffic sniffing	Information disclosure	Medium	Mitigated		A malicious actor can intercept traffic between a service and another actor (human or machine). The attacker is able to read the content of the communication because the channel is not encrypted and/or is encrypted using weak encryption algorithms.\nRelated attacks: Man in the Middle, Adversary in the Middle.	A threat can be mitigated by using an encryption algorithm such as TLS.
15	Exploit of weak TLS algorithm	Information disclosure	Medium	Mitigated		*Added because using TLS was suggested as remediation for threat "Traffic sniffing"* A service protects data as it traverses through the network, but because it uses weak TLS algorithm, an attacker can gain access to the supposed secure communication channel and data that is being transmitted. Vulnerable/weak TLS algorithms include versions: TLS 1.0, TLS 1.1, TLS 1.2	A threat can be mitigated by using secure version of TLS.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
24	Data tampering (at rest/in transit)	Information disclosure	Medium	Mitigated		Information stored in the service and/or transmitted over the network can be changed by an unauthorized actor. This can be possible because there is no encryption of the data transmitted over the network and/or stored in the service or broken/weak cryptographic algorithm is being used. This can also include a scenario where authorized actor with malicious intent can manipulate data without being detected, as there is no integrity protection for the data implemented.	A threat can be mitigated by using an encryption algorithm such as TLS.
29	Parameter tampering	Tampering	Medium	Mitigated		An attacker is able to modify parameters over a trusted boundary. If there is no parameter validation and the channel is unencrypted, an attacker can tamper with the parameters being exchanged in the request or path.	A threat can be mitigated by using an encryption algorithm such as TLS.
38	Manipulation of communication channel	Tampering	Medium	Mitigated		A malicious actor is able to manipulate the communication mechanism to impersonate other users, gain access to sensitive information, or carry out additional attacks. This threat targets the flawed assumptions about the mechanism used, incorrect protocol implementations, or the utilization of vulnerable protocols. It can also include manipulating a setting or parameter on the communications channel. Related attack: Traffic sniffing	A threat can be mitigated by using an encryption algorithm such as TLS.
43	Retransmission of network traffic	Tampering	Medium	Mitigated		Provide a description for this threatAn attacker is able to sniff network traffic and replay and/or delay sending the correct data without detection in order to cause unexpected behavior or perform unauthorized actions. Related attacks: Replayed request attack.	A threat can be mitigated by using an encryption algorithm such as TLS.

Redirect User to URL (Data Flow)

Description: The user is redirected to the Long URL.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
6	Traffic sniffing	Information disclosure	Medium	Mitigated		A malicious actor can intercept traffic between a service and another actor (human or machine). The attacker is able to read the content of the communication because the channel is not encrypted and/or is encrypted using weak encryption algorithms.\nRelated attacks: Man in the Middle, Adversary in the Middle.	A threat can be mitigated by using an encryption algorithm such as TLS.
17	Exploit of weak TLS algorithm	Information disclosure	Medium	Mitigated		*Added because using TLS was suggested as remediation for threat "Traffic sniffing"* A service protects data as it traverses through the network, but because it uses weak TLS algorithm, an attacker can gain access to the supposed secure communication channel and data that is being transmitted. Vulnerable/weak TLS algorithms include versions: TLS 1.0, TLS 1.1, TLS 1.2	A threat can be mitigated by using secure version of TLS.
22	Data tampering (at rest/in transit)	Information disclosure	Medium	Mitigated		Information stored in the service and/or transmitted over the network can be changed by an unauthorized actor. This can be possible because there is no encryption of the data transmitted over the network and/or stored in the service or broken/weak cryptographic algorithm is being used. This can also include a scenario where authorized actor with malicious intent can manipulate data without being detected, as there is no integrity protection for the data implemented.	A threat can be mitigated by using an encryption algorithm such as TLS.
31	Parameter Tampering	Tampering	Medium	Mitigated		An attacker is able to modify parameters over a trusted boundary. If there is no parameter validation and the channel is unencrypted, an attacker can tamper with the parameters being exchanged in the request or path.	A threat can be mitigated by using an encryption algorithm such as TLS.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
36	Manipulation of communication channel	Tampering	Medium	Mitigated		A malicious actor is able to manipulate the communication mechanism to impersonate other users, gain access to sensitive information, or carry out additional attacks. This threat targets the flawed assumptions about the mechanism used, incorrect protocol implementations, or the utilization of vulnerable protocols. It can also include manipulating a setting or parameter on the communications channel. Related attack: Traffic sniffing	A threat can be mitigated by using an encryption algorithm such as TLS.
45	Retransmission of network traffic	Tampering	Medium	Mitigated		Provide a description for this threatAn attacker is able to sniff network traffic and replay and/or delay sending the correct data without detection in order to cause unexpected behavior or perform unauthorized actions. Related attacks: Replayed request attack.	A threat can be mitigated by using an encryption algorithm such as TLS.

Retrieve Long URL (Data Flow)

Description: Long URL mapping is pulled from the database in order to redirect user.

DATA EXCHANGED: Short URL, Long URL, UserID

Number	Title	Type	Priority	Status	Score	Description	Mitigations
7	Traffic sniffing	Information disclosure	Medium	Mitigated		A malicious actor can intercept traffic between a service and another actor (human or machine). The attacker is able to read the content of the communication because the channel is not encrypted and/or is encrypted using weak encryption algorithms.\nRelated attacks: Man in the Middle, Adversary in the Middle.	A threat can be mitigated by using an encryption algorithm such as TLS.
16	Exploit of weak TLS algorithm	Information disclosure	Medium	Mitigated		*Added because using TLS was suggested as remediation for threat "Traffic sniffing"* A service protects data as it traverses through the network, but because it uses weak TLS algorithm, an attacker can gain access to the supposed secure communication channel and data that is being transmitted. Vulnerable/weak TLS algorithms include versions: TLS 1.0, TLS 1.1, TLS 1.2	A threat can be mitigated by using secure version of TLS.
23	Data tampering (at rest/in transit)	Information disclosure	Medium	Mitigated		Information stored in the service and/or transmitted over the network can be changed by an unauthorized actor. This can be possible because there is no encryption of the data transmitted over the network and/or stored in the service or broken/weak cryptographic algorithm is being used. This can also include a scenario where authorized actor with malicious intent can manipulate data without being detected, as there is no integrity protection for the data implemented.	A threat can be mitigated by using an encryption algorithm such as TLS.
30	Parameter tampering	Tampering	Medium	Mitigated		An attacker is able to modify parameters over a trusted boundary. If there is no parameter validation and the channel is unencrypted, an attacker can tamper with the parameters being exchanged in the request or path.	A threat can be mitigated by using an encryption algorithm such as TLS.
37	Manipulation of communication channel	Tampering	Medium	Mitigated		A malicious actor is able to manipulate the communication mechanism to impersonate other users, gain access to sensitive information, or carry out additional attacks. This threat targets the flawed assumptions about the mechanism used, incorrect protocol implementations, or the utilization of vulnerable protocols. It can also include manipulating a setting or parameter on the communications channel. Related attack: Traffic sniffing	A threat can be mitigated by using an encryption algorithm such as TLS.
44	Retransmission of network traffic	Tampering	Medium	Mitigated		Provide a description for this threatAn attacker is able to sniff network traffic and replay and/or delay sending the correct data without detection in order to cause unexpected behavior or perform unauthorized actions. Related attacks: Replayed request attack.	A threat can be mitigated by using an encryption algorithm such as TLS.