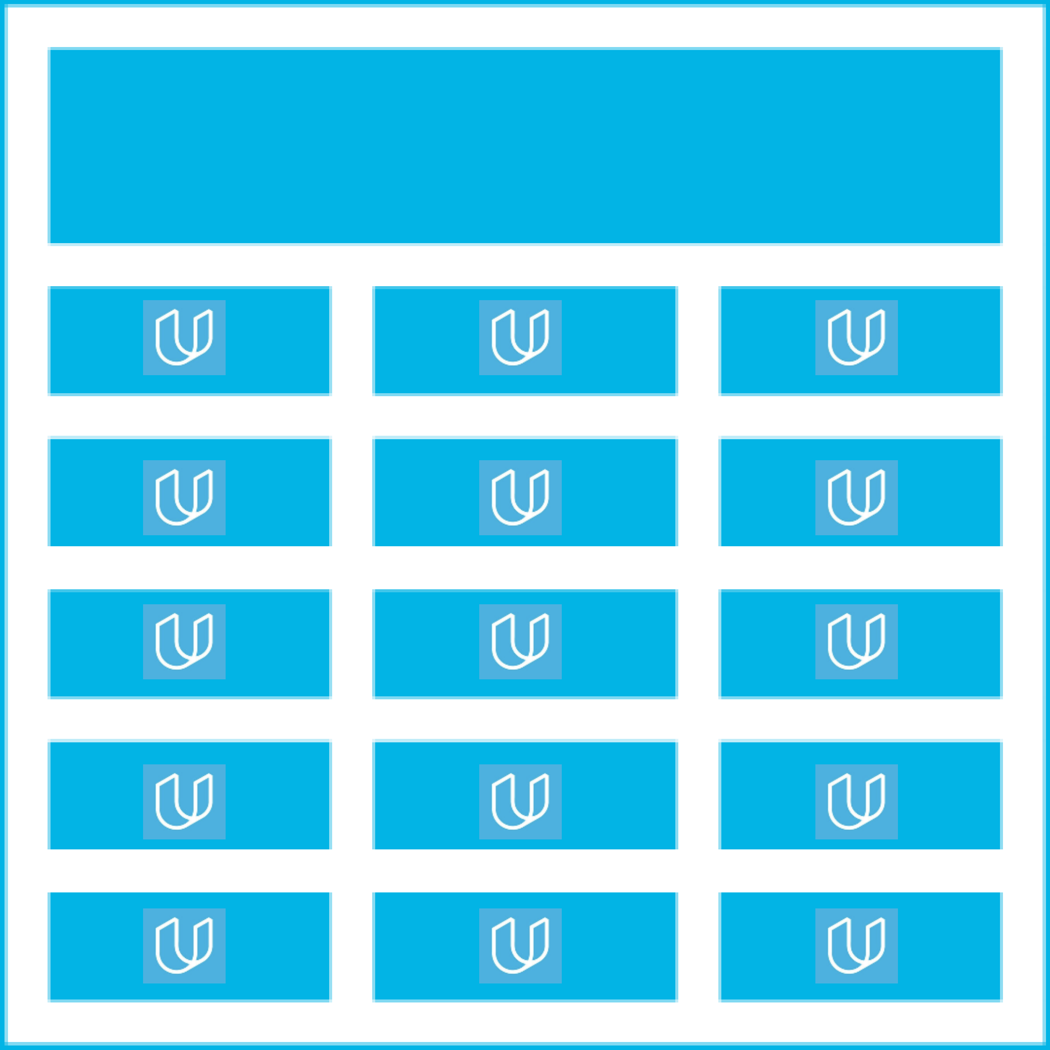
TimeSheets:

Threat Report



Jawhara Boodai

20 OCT 2022

Purpose of this Report:

This is a threat model report for **TimeSheets**. The report will describe the threats facing TimeSheets. The model will cover the following:

- Threat Assessment
  - Scoping out Asset Inventory
  - Architecture Audit
  - Threat Model Diagram
  - Threats to the Organization
  - Identifying Threat Actors
- Vulnerability Analysis
- Risk Analysis
- Mitigation Plan

Section 1

Initial Threat Assessment

Completed Asset Inventory

Components and Functions

- ***TimeSheets Web Server:*** The web server's primary role is to serve static content to a requesting client through the http protocol.
- ***TimeSheets Application Server:*** The application server handles all the business logic process and serves dynamic content.
- ***TimeSheetsDB:*** The database server stores employee data and will be queried from the application server.
- ***AuthDB:*** Stores user authentication data (credentials) and will be queried from the application server.

Completed Asset Inventory

Overview of Application Functionality

TimeSheets is used by employees to track their hours worked. Users will login to the TimeSheets application from their device.

Data Flow

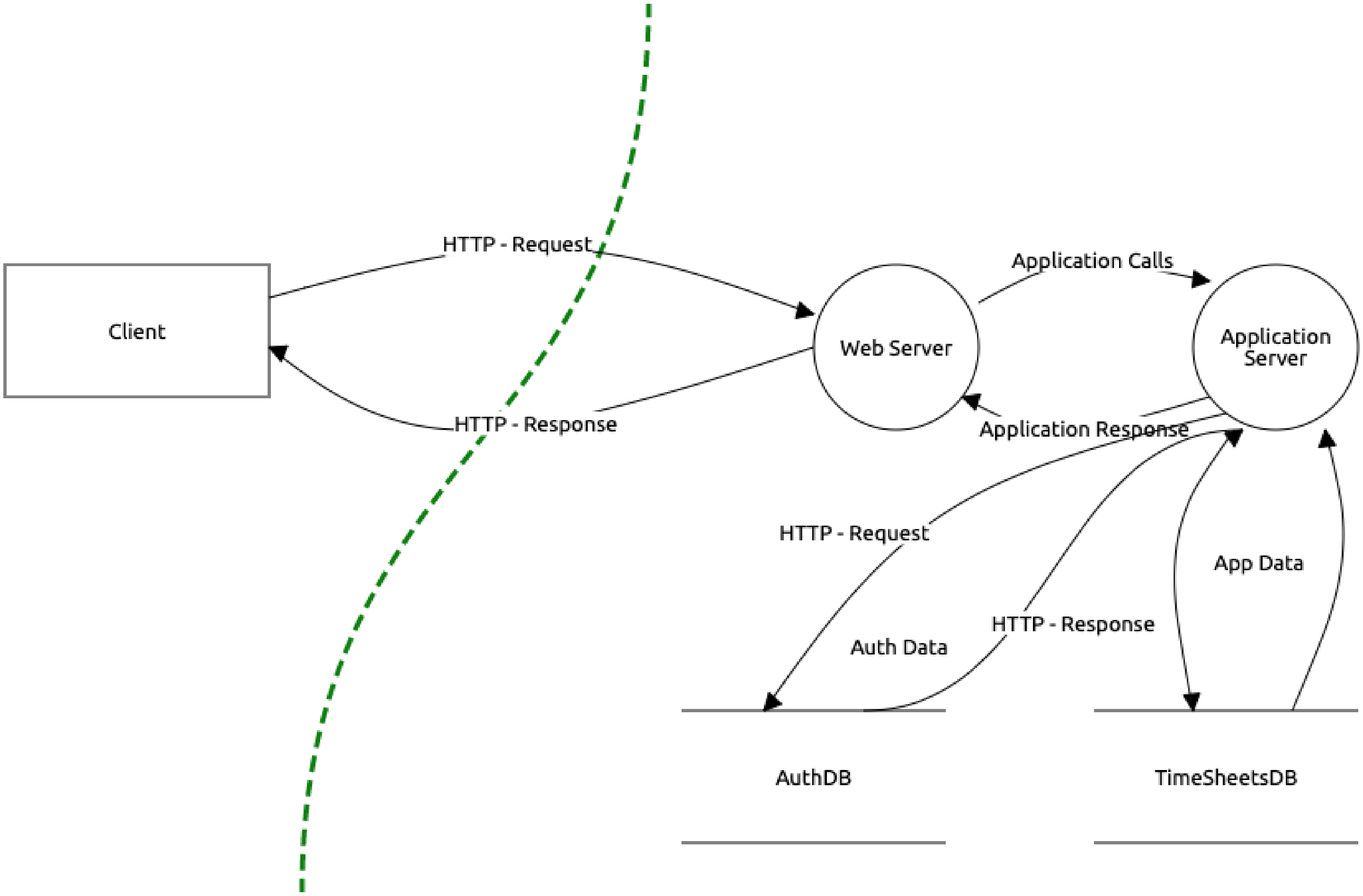
Request is generated from the client via the Internet. The request arrives at the TimeSheets web server which serves static content to the user (HTML, images, etc). Dynamic data is retrieved from the database and served to the client.

Completed Architecture Audit

Flaws

- *There is a lack of encryption at rest - database servers are storing data on unencrypted disks.*
- *There is lack of redundancy.*
- *There is no firewall that is filtering traffic coming from the Internet*

Completed Threat Model



- Employee Data Unencrypted at Rest
- Authentication data is using reversible encryption
- Authentication requests are not encrypted in transit
- Sensitive data is encrypted using DES algorithm

Completed Threat Analysis

What Type of Attack Caused the Login Alerts?

Man in the Middle (MitM)

What Proves Your Theory?

There is lack of encryption between the client and the application. A malicious actor is sniffing traffic and intercepting the requests with a valid username/password in the request. Additionally, the logs show successful login attempts from the expected IP, but also a different location at the same time.

Completed Threat Actor Analysis

Who is the Most Likely Threat Actor?

Internal User

What Proves Your Theory?

The IP address of the unexpected login matches that of an internal user. Additionally, there was no data leaked from the company, and no data changed. Just data accessed that the legitimate user typically doesn’t access.

Section 2

Vulnerability Analysis

2.1 Employee Data Unencrypted at Rest

Discovery:

During the threat model the SRE team confirmed that the database is on a server that does not have encryption at rest.

Why is this an issue?

This is a huge issue because attackers might gain access to the stored data and while data is unencrypted, the attackers can easily understand the data and the impact will be catastrophic.

Data should be encrypted at rest also to protect data confidentiality.

2.2 Authentication Data Stored Using Reversible Encryption

Discovery:

During the threat model the DBA team confirmed that the database is storing authentication data (credentials) encrypted.

Why is this an issue?

When data is stored in a reversable encryption this will create a potintial threat because the encryption can be reversed by attackers to deduce the plaint text.

Encryption should always be irreversable by using hash algorthim and salting.

2.3 Authentication Requests are Unencrypted in Transit

Discovery:

During the threat model the security team confirmed that authentication requests are being transmitted in plaintext.

Why is this an issue?

Attackers may intercept the connection between the user and the system and then they will be able observ all the messages. Also, they will be able to see the authintacation request that sent by the user including his username and password, and then use them to gain access to the system.

2.DES Algorithm in Use

Discovery:

During the threat model the security team identified sensitive data being stored using the DES algorithm.

Why is this an issue?

It is always better to use the advanced one which is AES, because Des uses only 56 key bit which is guessable by attackers when using brute force attack.

Optional Task:

Examine the threat model diagram from Section 1 and answer:

What non-encryption issues can you identify?

What recommendation would you give to solve those issues?

Why do you recommend those solutions?

- *[Issue 1 Here]*
- *[Issue 2 Here]*
- *[Add more issues as necessary]*

Section 3

Risk Analysis

3.1 Scoring Risks

Risk	**Score** ***(1 is most dangerous, 4 is least dangerous)***
Unencrypted at Rest	3
Reversible Encryption	2
Unencrypted in Transit	1
Outdated Algorithm	4

3.2 Risk Rationale

Why Did You Choose That Ranking? Make sure to include your risk ranking methodology. *(*
I did not use a tool, the ranikng was choseen logically.

1. Unencrypted in Transit: *data will be shown to attackers who intercept the medium such as man in the middle.*
2. Reversible Encryption: *When data is stored in a reversable encryption this will create a potintial threat because the encryption can be reversed by attackers to deduce the plaint text.*
3. Unencrypted at Rest: *attackers might gain access to the stored data and while data is unencrypted, the attackers can easily understand the data and the impact will be catastrophic.*
4. Outdated Algorithm: *attackers will easily find the related vulnerabilities of outdated algorithms and exploit them. Because exposed vulnerabilities are posted every where.*

Section 4

Mitigation Plan

4.1 Employee Data Unencrypted at Rest

What is Your Recommended Mitigation Plan?

The risk is: data unauthorized access.

Establish an encryption policy: data must be encrypted and salted before storing.

Database checkup every 120 days.

Use AES-256

Why Did you Recommend This Course of Action?

Data encryption is very important, so I chose AES-256 because it is harder to because there is 256 possible key permutation.

4.2 Authentication Data Stored Using Reversible Encryption

What is Your Recommended Mitigation Plan?

Reversable encryption is very risky.

System should be assaigned an irriversable encryption algorithem policy. For example hash+salt.

Update the policy every 3 months.

I recommend using agron2

Why Did you Recommend This Course of Action?

To prevenit reversing cipher text to plain text. Update the policy to make it hard to deduce for attackers.

I recommend using agron2 because it is adviesd by OWASP

4.3 Authentication Requests are Not Encrypted in Transit

What is Your Recommended Mitigation Plan?

It should be encrypted.

Each session should be encrypted and the encrypthion key should be sent to the destination via a diffirent mideum.

There should be Multi-factor authintication to prove ID.

I Recommend using TLS algorithm

Why Did you Recommend This Course of Action?

[to secure the connection and prevent any(man-in-the-middle, replay session, etc.. Sttacks). Using Multi-factor authintication make such as OTP make it harder to masqurade or impersonate to attackers.

I Recommend using TLS algorithm because it ensures the other party in a connection is who they say they are, shows whether data retains its initial integrity, and provides confidentialit

4.4 DES Algorithm in Use

What is Your Recommended Mitigation Plan?

DES can be broken by using brute force attack. So it should be upgraded to Aes to make it harder to guess.

I recommend using AES-256.

Why Did you Recommend This Course of Action?

DES has a key of 56 bit, which can be guessd by brute force attack, unlike the AES.

I recommend using AES-256 because it is harder to guess than the DES-56.

4.5 Security Audit

The audit team has been made aware of the systemic issue and wants to ensure your recommendations are followed. What steps can the audit team take?

1. *Establish an encryption policy.*
2. *Test the encryptoin tools and integrate them.*
3. *Inform employees*
4. *Train employees how to use new system.*
5. *Perform perodically system checkups.*

Optional Task:

Create an architecture diagram of a secure system.

Image of your secure architecture:

Optional Task *(Continued)*:

Additional Steps Would You Recommend to Prevent the Attack as well as Future Issues: