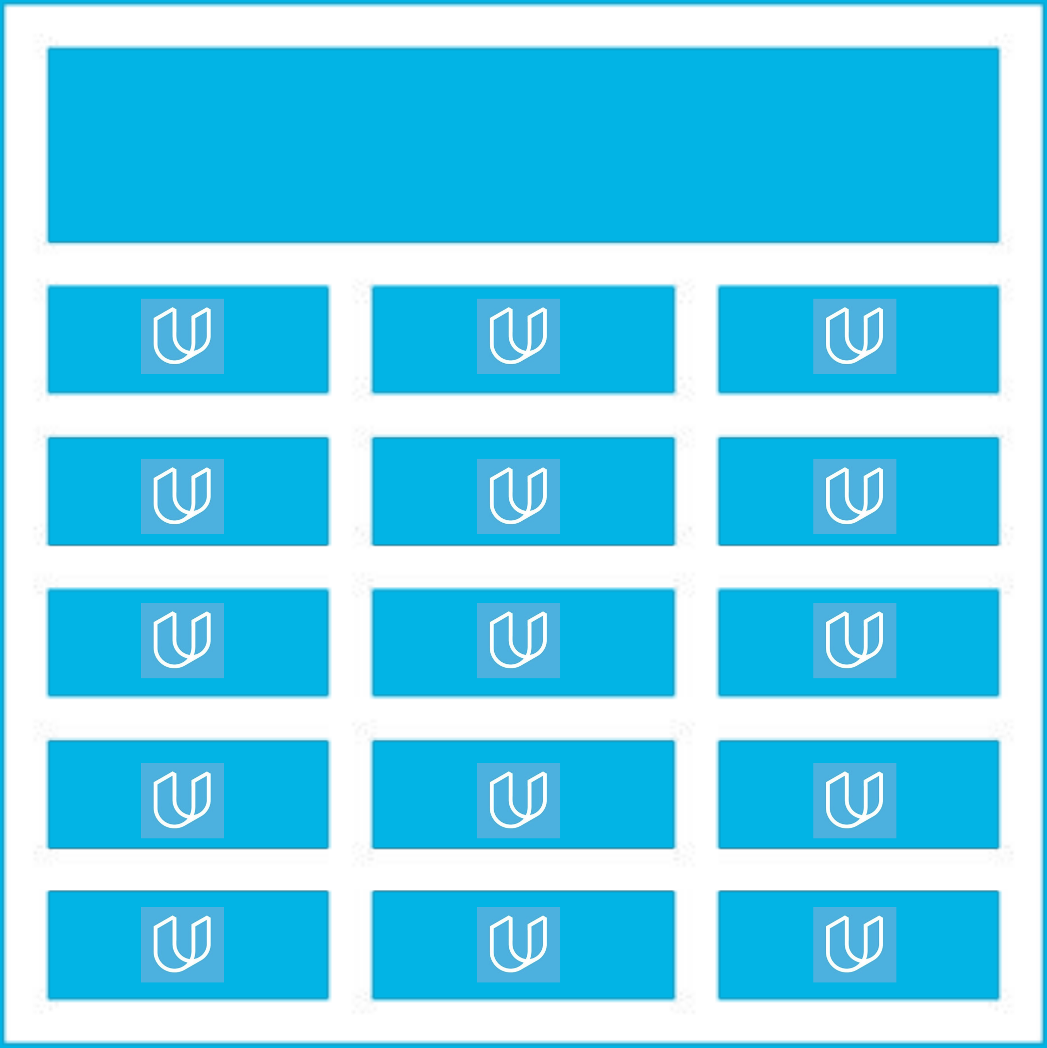
# **TimeSheets:**

## Threat Report



**YOUR NAME:** Hsin-Wen C

*DATE: 07/15/202*

# 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

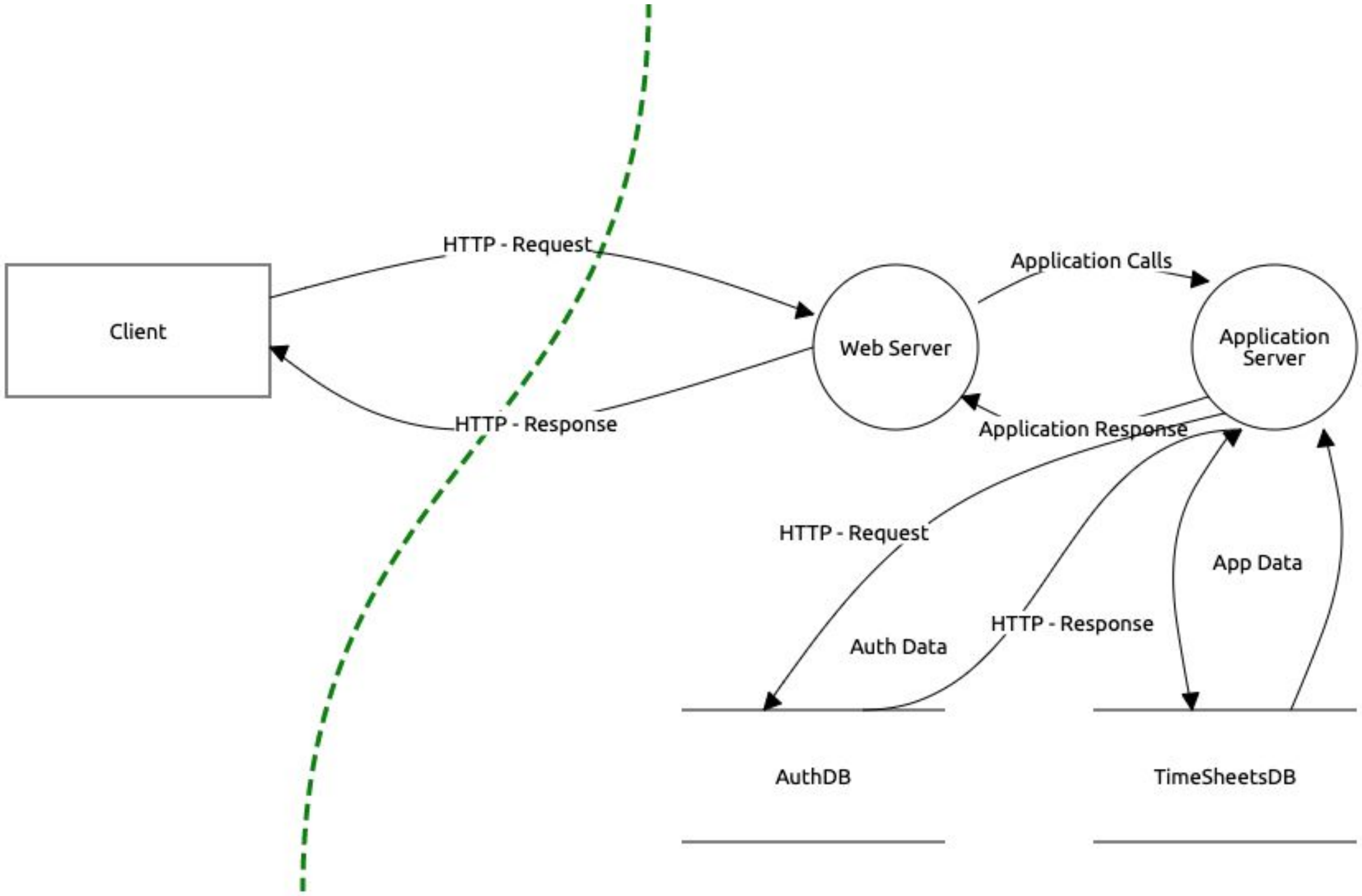
Request is generated from the client via the Internet. The request arrives at the TimeSheets web server which serves static content to the user (HTML, images, etc). Dynamic data is retrieved from the database and served to the client.

# 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?**

Encryption entails re-formatting the original data in order to prevent unauthorized users from decrypting it. For instance, you might have kept on your server a copy of a paid invoice and the credit card information of a customer. At all costs, something can't end up in the wrong hands. When you encrypt data while it is in transit, you are essentially converting sensitive information about your clients into a different kind of information. This frequently happens using an algorithm that is incomprehensible to a user without the encryption key needed to decode it. Because only authorized personnel can access these files, your data will remain secure.

# 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?**

It is possible to decrypt encrypted passwords if they are stored in a manner that is reversible. Once this encryption has been cracked, a skilled attacker can log in to network resources using the compromised account. In order to prevent password information from being compromised, never activate Store password using reversible encryption for all users in the domain.

## 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?**

One element of a more comprehensive security plan is encryption. After a connection has been established and authorized, encryption in transit protects your data from possible attackers by removing the requirement for trusting the network's lower layers, which are frequently provided by outside parties the potential attack surface is minimized preventing access to information by attackers in the event that conversations are intercepted. *Prevents conversations between your site and the cloud provider or between two services from being eavesdropped while data is being transferred between them. To achieve this protection, the data is encrypted before transmission, the endpoints are authenticated, and the data is decrypted and verified to have not been altered once it has arrived. For instance, Secure/Multipurpose Internet Mail Extensions (S/MIME) is frequently used for email message encryption, while Transport Layer Security (TLS) is often used to encrypt data in transit for transport security.*

# 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?**

When a weak encryption is required, the DES algorithm is utilized. The 56-bit key size of the DES algorithm is probably its worst drawback. A million DES operations may be encrypted and decrypted by chips in a second. For $1 million, you may get a DES cracking device that will search all the keys in roughly seven hours. On hardware, DES can be readily implemented. It runs relatively slowly on it though because it wasn't made to run software.

With the advancement of technology, it has gotten simpler to crack the DES's encrypted code. AES is recommended today over DES.

As a form of symmetric encryption, DES employs a single key for both encryption and decryption. In the event that one key is misplaced, we won't be able to get any decipherable data at all.

# Section 3

## Risk Analysis

# 3.1 Scoring Risks

| **Risk** | **Score**<br>***(1 is most dangerous, 4 is least dangerous)*** |
|---|---|
| Unencrypted at Rest | 2 |
| Reversible Encryption | 3 |
| 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.** *(Did you use a tool or defined risk scoring system?)*

Unencrypted in Transit is the most dangerous since hacker can direct access those inform why intercept those packet.

# Section 4

## Mitigation Plan

# 4.1 Employee Data Unencrypted at Rest

### What is Your Recommended Mitigation Plan?

### Why Did you Recommend This Course of Action?

# 4.2 Authentication Data Stored Using Reversible Encryption

### What is Your Recommended Mitigation Plan?

*[Your recommended plan here]*

### Why Did you Recommend This Course of Action?

*[Your justification here]*

# 4.3 Authentication Requests are Not Encrypted in Transit

### What is Your Recommended Mitigation Plan?

*[Your recommended plan here]*

### Why Did you Recommend This Course of Action?

*[Your justification here]*

# 4.4 DES Algorithm in Use

**What is Your Recommended Mitigation Plan?**

*[Your recommended plan here]*

**Why Did you Recommend This Course of Action?**

*[Your justification here]*

# 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?**

*[Your answer here]*