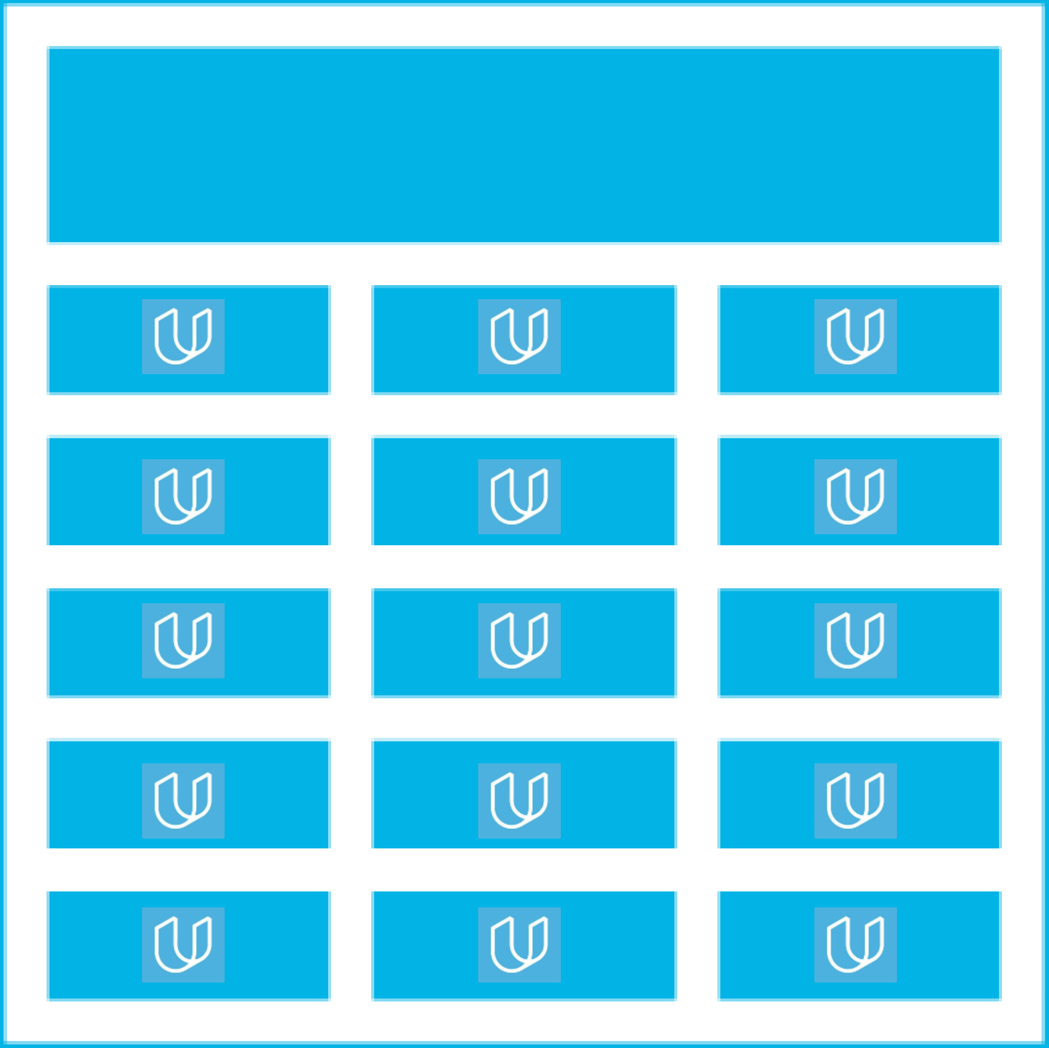
TimeSheets:
Threat Report



YOUR NAME: Jean Rodrigues Neves

DATE 05/16/2021

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

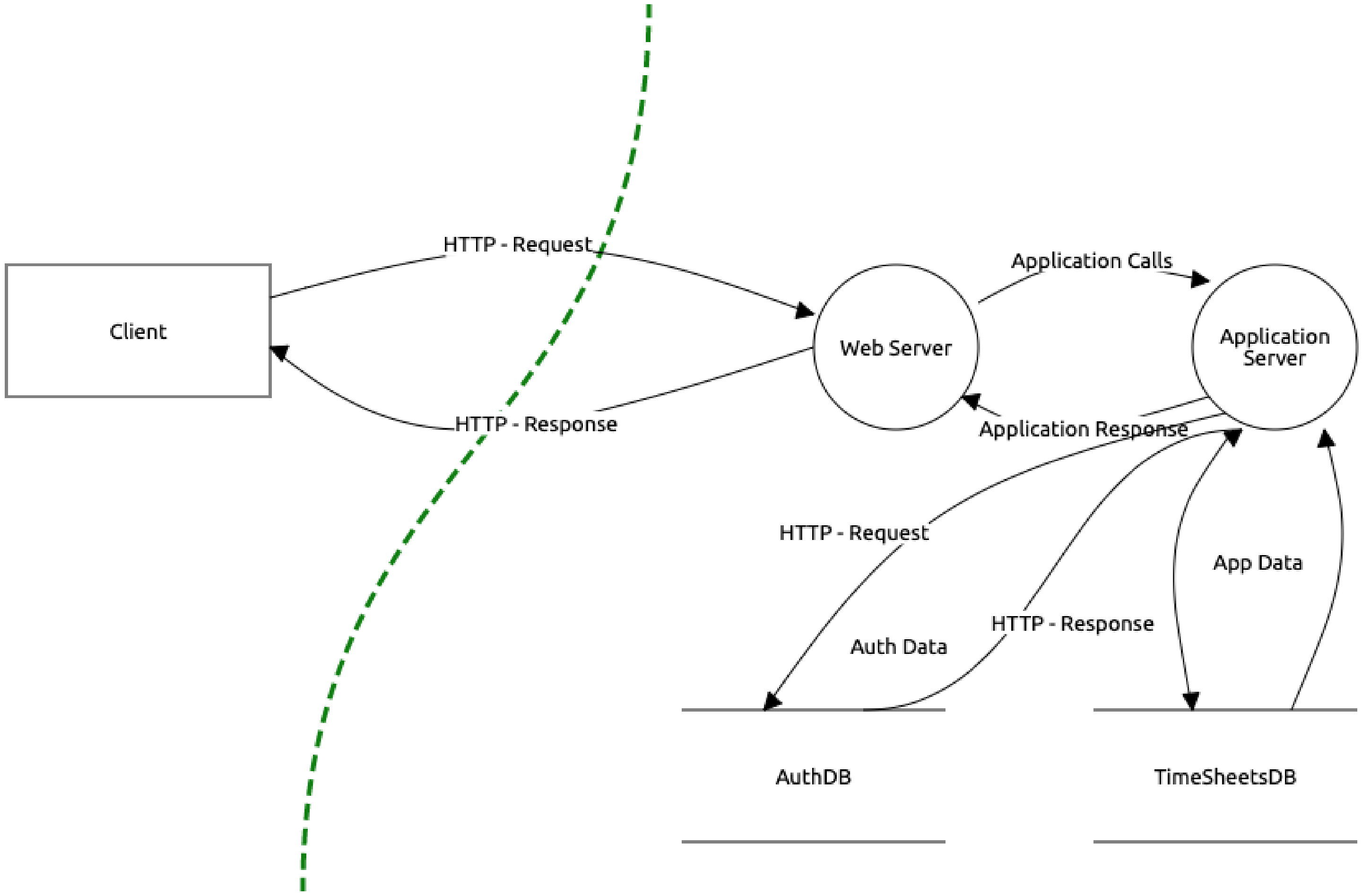
Request is generated from the client via the Internet. The request arrives at the TimeSheets web server which serves static content to the user (HTML, images, etc). Dynamic data is retrieved from the database and served to the client.

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 at reast is a key protection against a data breach. The issue with a server that does not have an encryption at reast is that your server is vulnerable for threats such as ramsonware attacks.

By encrypting data at rest, you're essentially converting your customer's sensitive data into another form of data. This usually happens through and algorith that can't be undestood by a user who does not have an encryption key to decode it. Only authorized personnel will have access to these files, this ensuring that the data stays 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?

The primary weakness of reversible encryption is simple: if the key is compromised, the encrypted data is compromised.

Storing password using reversible encryption is essentially equal to storing plaintext versions of the passwords.

The issue is that all the passwords are exposed. It’s not recommended to store password using reversible encryption.

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?

Securing data in transit is essentially securing data as it passes over a network.
Transmit data in plantext means that there is the possibility of someone monitoring or intercepting messages and being able to read their contents. This in turn could lead to unauthorized access to sensitive resources, as well as costly data breaches.

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?

DES, the Data Encryption Standard, can no longer be considered secure. While no major flaws in its innards are known, it is fundamentally inadequate because it 56-bit key is too short.

It’s vulnerable to brute-force search of the whole key space, either by large collections of general-purpose machines or even more quickly by scpecialized hardware. This also applies to any other cipher with only a 56-bit key.

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?

- *HTTP instead of HTTPS*
- *Unincrypted data in transit*

Use HTTPS protocol, it uses TLS/SSL to provide critical data protection during internet transmission. With the proper use of this protocol, all data submitted or received by the application is encrypted and cannot be read by a third party.

Section 3

Risk Analysis

3.1 Scoring Risks

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

I choosed that ranking based on the formula Risk= Threat x Vulnerability x Impact.
I did not used a tool, I defined risk scoring system.

Unencrypted at reast number one = Because it compromise the whole diagram in case of a security breach such as an Rammsonware attack.

Unencrypted in transit number two = Because transmit data in plantext means that there is the possibility of someone monitoring or intercepting messages and being able to read their contents. This in turn could lead to unauthorized access to sensitive resources, as well as costly data breaches.

Reversible Encryption number three = The primary weakness of reversible encryption is simple: if the key is compromised, the encrypted data is compromised.

Outdated Algorithm number = It's vulnerable to brute-force search of the whole key space, either by large collections of general-purpose machines or even more quickly by scpecialized hardware. This also applies to any other cipher with only a 56-bit key.

Section 4

Mitigation Plan

4.1 Employee Data Unencrypted at Rest

What is Your Recommended Mitigation Plan?

Use Antivirus software and firewalls. Or a hardware encryption.

Why Did you Recommend This Course of Action?

Encrypting data at rest on tape and disk will significantly mitigate threats and allow secure engineers secure the data while maintain the current service levels for operations.

4.2 Authentication Data Stored Using Reversible Encryption

What is Your Recommended Mitigation Plan?

Use a nonreversible hash when you store passwords so that even if your database is leaked, the passwords themselves are stil safe.

Why Did you Recommend This Course of Action?

The key for reversible encryption needs to be on disk or in memory all the time. If that program, disk or memory are somehow compromised, the all those reversibly encrypted passwords ate all comprised in one fell swoop.

In contrast, consider the use of nonreversible hashes. If the program, disk, or memory are comprised then the attacker gets the “locked” hashes, and there is no key.

MITM SSL attack

- Domain Name System (DNS) spoofing attacks (including /etc/hosts)
- Baseband Attack

4.3 Authentication Requests are Not Encrypted in Transit

What is Your Recommended Mitigation Plan?

Use HTTPS protocol, it uses TLS/SSL to provide critical data protection during internet transmission. With the proper use of this protocol, all data submitted or received by the application is encrypted and cannot be read by a third party.

Why Did you Recommend This Course of Action?

Data in transit describes data that is sent over a network or is located in the RAM. At some point, data that was recovered from the device (or data at rest) was also sent over the network. An example of this includes sending a trxt message to another user, or web browsing over a wireless connection.

Several well-known techniques are used by attackers to compromise data in transit such as:
Man in the middle, MITM SSL attack, DNS spoofing attacks, and baseband attacks.

4.4 DES Algorithm in Use

What is Your Recommended Mitigation Plan?

Use another more modern Encryption Algorithm such as RSA or Triple DES.

Why Did you Recommend This Course of Action?

DES is now outdated symmetric encryption algorithm; you use the same key to encrypt and decrypt a message. DES use a 56-bit encryption key and encrypts data in blocks of 64 bits. These sizes are typically not large enough for today's issues.

Triple DES employs three individual keys with 56 bits each. The total key length adds up to 168 bits.

RSA a popular public-key (asymmetric) encryption algorithm. It uses a pair of keys: the public key, used to encrypt the message, and the private key, used to decrypt the message.

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?

During the audit, take care to provide appropriate documentation and perform due diligence throughout the process. Monitor the progress of the audit and also the data points collected for accuracy. Use previous audits and new information as well as the guidance of your auditing team o carefully select which rabbit holes in which you descend. You will uncover details that require further examination but prioritize those new item with the team first.

Complete the audit and socialize the results with all of the stakeholders using the agreed-upon definitions. Create a list of action items based on the audit and prioritize fixes and changes to remediate the security items discovered.

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: