CMPE 494 SP.TP. INFORMATION SECURITY

Risk Analysis with CORAS

Gözde Ünver
Sevde Sarıkaya
Ahmet Necip Görgülü

1. Description of the Organization

Bogazici University is a university in Turkey established in 1863 by Cyrus Hamlin and Christopher Robert. Today the university boasts six undergraduate faculties and six graduate institutes with a total of 35 graduate, 67 masters and 33 Ph.D. programs. As of 2020, it has a total of 16.223 students, 12.912 of them being undergraduate and the rest of them being graduate. With its 438 full time teachers (Prof, Assoc. Prof) and 562 agreements with International Universities, Bogazici is the most prestigious university in the nation.

2. Description of the Scope

Bogazici Registration system is a system designed for students and teachers for handling various actions regarding their academic career. Students can select courses for the upcoming semester, check their grades, complete surveys regarding course evaluation, apply for dorms, exchange programs and so on. Teachers use this system to grade their students.

3. Refination of the Target Description Using Asset Diagrams

3A. Asset Diagram

In the below diagram you can see the main assets and the effects they have on each other.

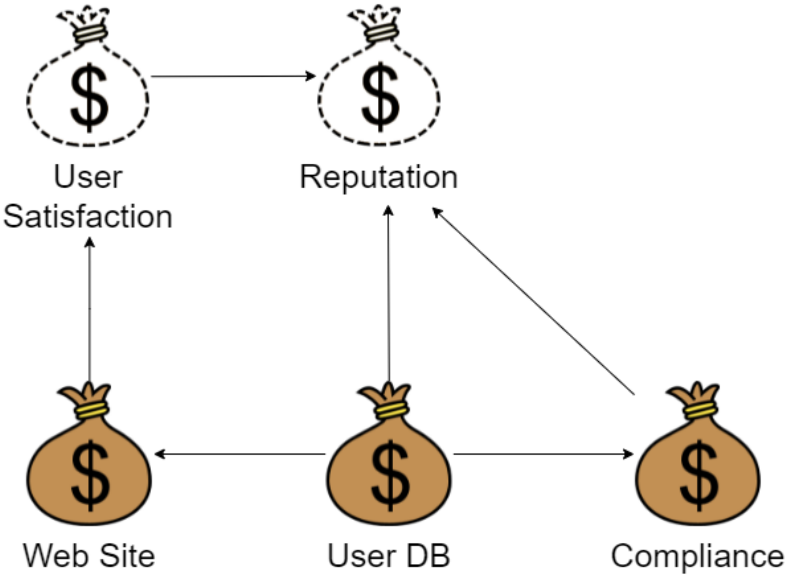

Figure 3A

3B. High Level Risk Analysis

In this part the attacker-method-vulnerability relation is explored. The table contains every single case that is possible.

Who/What is the cause?	How? What may happen? What does it harm?	What makes this possible?
Hacker	DoS Attack causes registration system to go down	Use of web application; insufficient DoS attack prevention.
Hacker	Malcode introduced by hacker via email compromises database integrity or confidentiality.	Insufficient employee training
Hacker	Hacker changes email addresses via sql injection.	Incorporating user-inputted text into a SQL query
Hacker	Hacker installs a backdoor in a company hardware via physically getting in touch with said hardware. Jeopardizes the user database integrity and-or confidentiality	Insufficient physical security and surveillance around company hardware.
Hacker	Hacker probes the web interface and find a way to access to pages when he shouldn't	Use of web application; Insufficient web security. Sloppy design
Hacker	Man in the middle attack via phishing site makes employees leak data	Incompetent employees. Insufficient training.
Hacker	While probing the web interface hacker finds data which shouldn't be on the interface in the first place	Insecure web interface design. Incompetent employees. Insufficient training.

Employee	Employee deliberately corrupts integrity of the database by modifying data they should not modify.	Insufficient work hierarchy. Insufficient tracking of employees.
Employee	Employee deliberately shares the data with unknown 3rd party	Workplace problems and trust issues among the employees.
Employee	Employee accidentally deletes the entire database.	Incompetent employees. Insufficient training.
System Failure	Server shuts down due to power outage and renders the web interface unusable	Insufficient precautions regarding power outages.
System Failure	Database connection on Amazon is lost to problems on Amazon’s ends. Web interface is unusable	Too much trust in provider company
System Failure	Too many users trying to use the system at once making the website crash.	Insufficient resources allocated to providing service rate.

Table 3B

4. Approval of the Target Description

4A. Likelihood Scale

This is the likelihood scale that is used throughout the analysis.

Likelihood	Description
Certain	More than five times a year
Likely	Two to five times a year
Possible	Once a year
Unlikely	Once per two years
Rare	Once or less per five years

Table 4A

4B. Consequence Scales

There are 5 assets in the analysis, 3 of them are direct and 2 of them are indirect. Each of them requires their own consequence scales for the analysis. Each of them can be found below.

4B.1 Consequence Scale for User Database Asset

Consequence	Description
Catastrophic	More than 20% records are affected
Serious	Range of [10%, 20%[records are affected
Moderate	Range of [5%, 10%[records are affected
Minor	Range of [1%, 5%[records are affected
Insignificant	Range of [0%, 1%[records are affected

Table 4B.1

4B.2 Consequence Scale for Web Site Asset

Consequence	**Description**
Catastrophic	Downtime is more than 1 week
Serious	Downtime is in the range of [1 day, 1 week[
Moderate	Downtime is in the range of [1 hour, 1 day[
Minor	Downtime is in the range of [1 minute, 1 hour[
Insignificant	Downtime is in the range of [0, 1 minute[

Table 4B.2

4B.3 Consequence Scale for Compliance Asset

Consequence	**Description**
Catastrophic	The incident is against government policies
Moderate	The incident is against user terms & conditions
Insignificant	The incident is against industry regulations

Table 4B.3

4B.4 Consequence Scale for User Satisfaction Asset

Consequence	Description
Catastrophic	More than 50 complaint messages are received from users
Moderate	16-50 complaint messages are received from users
Minor	6-15 complaint messages are received from users
Insignificant	0-5 complaint messages are received from users

Table 4B.4

4B.5 Consequence Scale for Reputation Asset

Consequence	Description
Catastrophic	Bogazici Registration System’s ranking among other systems is lower than 50
Moderate	Bogazici Registration System’s ranking among other systems is lower than 30
Insignificant	Bogazici Registration System’s ranking among other systems is lower than 10

Table 4B.5

4C.1 Harm Severity Description Table

Risks are classified into three categories and they are represented with the colors below throughout the analysis.

Harm severity color	Description
🟩	Acceptable
🟧	Monitor
🟥	Needs to be treated (Unacceptable)

Table 4C.1

4C.2 Risk Functions

The risk functions generated from the likelihood table of the analysis and the consequence tables for each asset are shown below.

4C.2.1 Risk Function for the User Database Asset

Consequence / Likelihood	Insignificant	Minor	Moderate	Serious	Catastrophic
Rare	🟩	🟩	🟧	🟥	🟥
Unlikely	🟩	🟩	🟧	🟥	🟥
Possible	🟩	🟧	🟧	🟥	🟥
Likely	🟩	🟧	🟥	🟥	🟥
Certain	🟩	🟧	🟥	🟥	🟥

Table 4C.2.1

4C.2.2 Risk Function for the Web Site Asset

Consequence / Likelihood	**Insignificant**	**Minor**	**Moderate**	**Serious**	**Catastrophic**
Rare	🟩	🟩	🟧	🟧	🟥
Unlikely	🟩	🟩	🟧	🟧	🟥
Possible	🟩	🟩	🟧	🟥	🟥
Likely	🟩	🟧	🟧	🟥	🟥
Certain	🟩	🟧	🟥	🟥	🟥

Table 4C.2.2

4C.2.3 Risk Function for the Compliance Asset

Consequence / Likelihood	**Insignificant**	**Minor**	**Moderate**	**Serious**	**Catastrophic**
Rare	🟩	🟩	🟧	🟧	🟥
Unlikely	🟩	🟩	🟧	🟧	🟥
Possible	🟩	🟩	🟧	🟥	🟥
Likely	🟩	🟧	🟧	🟥	🟥
Certain	🟩	🟧	🟥	🟥	🟥

Table 4C.2.3

4C.2.4 Risk Function for the User Satisfaction Asset

Consequence/ Likelihood	Insignificant	Moderate	Minor	Catastrophic
Rare	🟩	🟩	🟩	🟧
Unlikely	🟩	🟩	🟩	🟥
Possible	🟩	🟩	🟧	🟥
Likely	🟩	🟧	🟥	🟥
Certain	🟧	🟧	🟥	🟥

Table 4C.2.4

4C.2.5 Risk Function for the Reputation Asset

Consequence / Likelihood	Insignificant	Minor	Moderate	Serious	Catastrophic
Rare	🟩	🟩	🟧	🟥	🟥
Unlikely	🟩	🟩	🟧	🟥	🟥
Possible	🟩	🟩	🟧	🟥	🟥
Likely	🟩	🟧	🟧	🟥	🟥
Certain	🟩	🟧	🟥	🟥	🟥

Table 4C.2.5

5. Risk Identification

5A. Identify Assets and Threats

The assets and the threats used in the analysis are listed below. The left column represents the list of assets and the right column represents the threats.

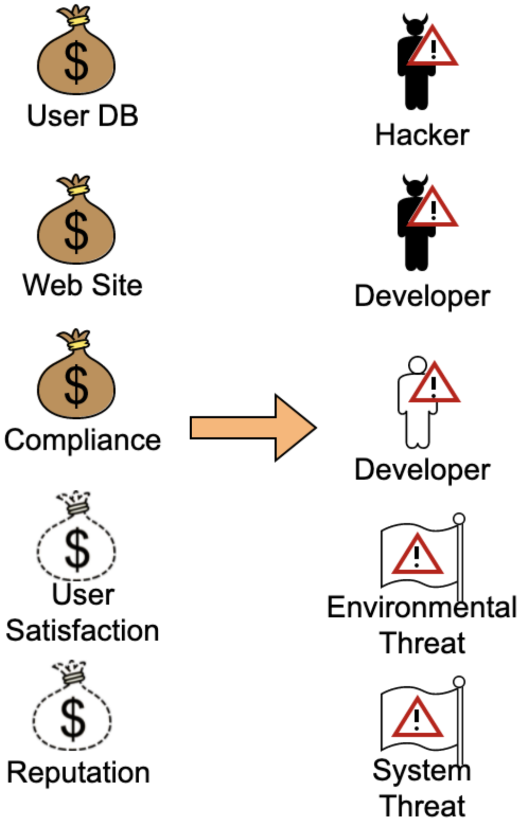

Figure 5A

5B. Identify Unwanted Incidents

Here you can see the unwanted incidents and the assets they have effect on.

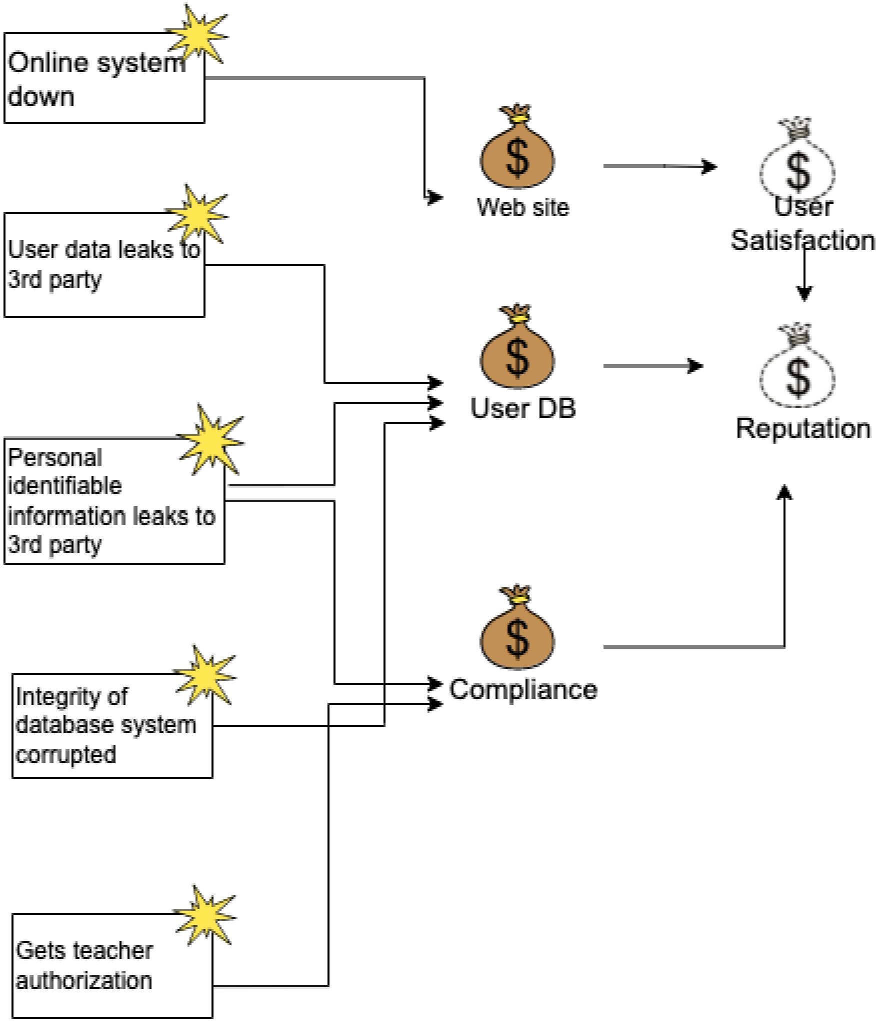
Online system down
User data leaks to 3rd party
Personal identifiable information leaks to 3rd party
Integrity of database system corrupted
Gets teacher authorization
Web site
User DB
Compliance
User Satisfaction
Reputation

Figure 5B

5C. Threat Scenarios

Here you can see the diagram showing the relation among possible threat scenarios, unwanted incidents they cause and assets.

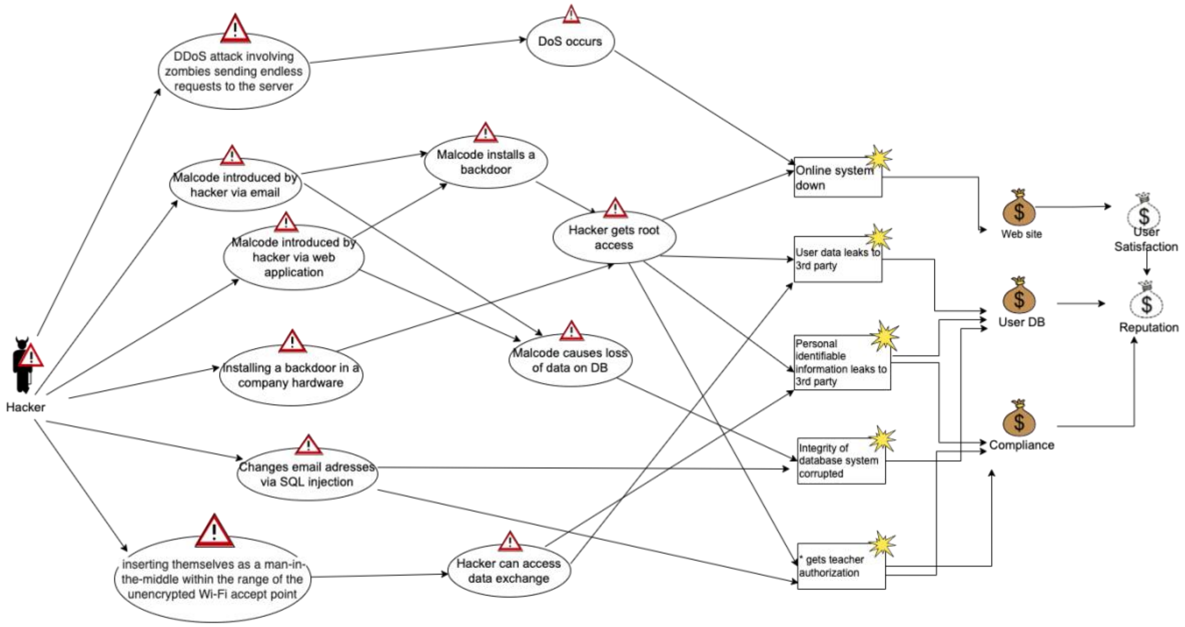
DDoS attack involving zombies sending endless requests to the server
DoS occurs
Malcode introduced by hacker via email
Malcode installs a backdoor
Malcode introduced by hacker via web application
Hacker gets root access
Hacker
Installing a backdoor in a company hardware
Malcode causes loss of data on DB
Changes email adresses via SQL injection
inserting themselves as a man-in-the-middle within the range of the unencrypted Wi-Fi accept point
Hacker can access data exchange
Online system down
User data leaks to 3rd party
Personal identifiable information leaks to 3rd party
Integrity of database system corrupted
* gets teacher authorization
Web site
User DB
Compliance
User Satisfaction
Reputation

Figure 5C.1

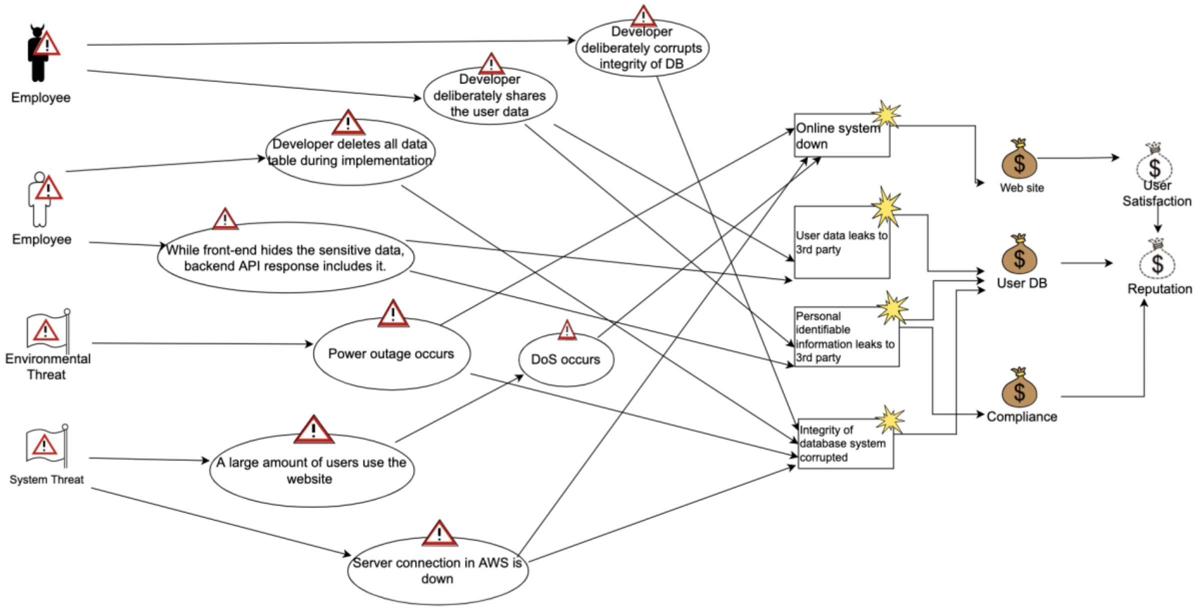
Employee
Employee
Environmental Threat
System Threat
Developer deliberately corrupts integrity of DB
Developer deliberately shares the user data
Developer deletes all data table during implementation
While front-end hides the sensitive data, backend API response includes it.
Power outage occurs
DoS occurs
A large amount of users use the website
Server connection in AWS is down
Online system down
User data leaks to 3rd party
Personal identifiable information leaks to 3rd party
Integrity of database system corrupted
Web site
User DB
Compliance
User Satisfaction
Reputation

Figure 5C.2

5D. Identify Vulnerabilities

This diagram shows the vulnerabilities enabling the threats.

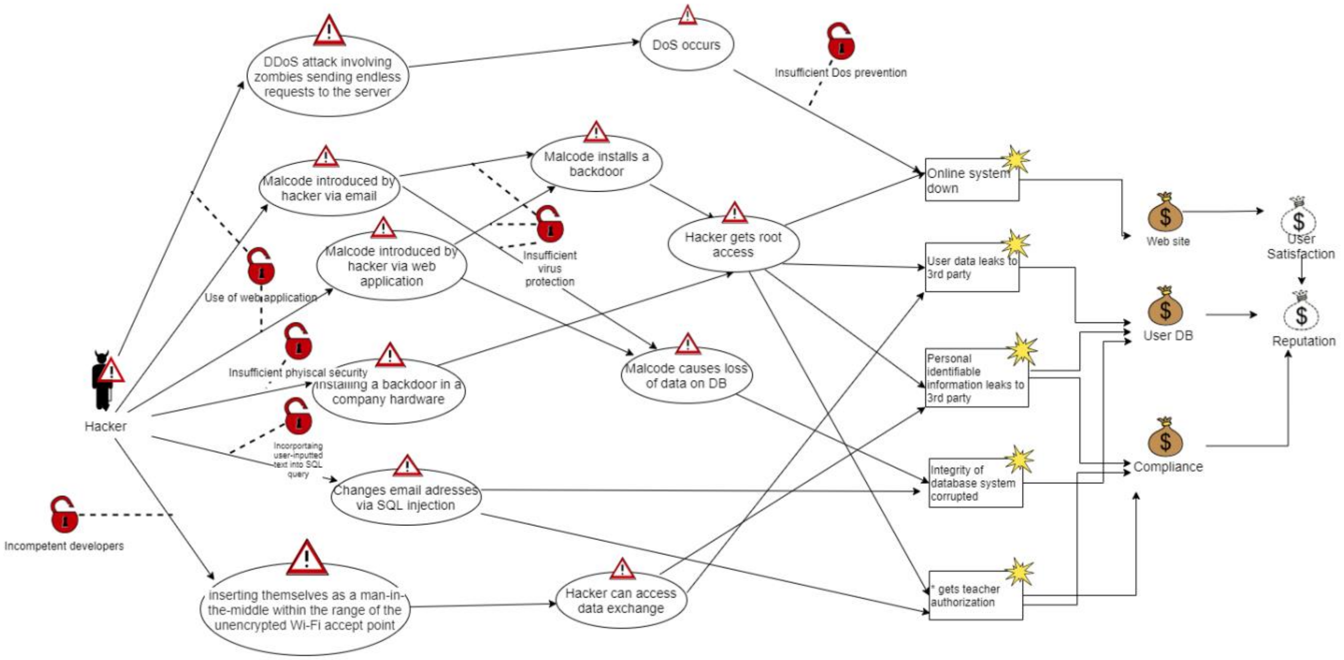
DDoS attack involving zombies sending endless requests to the server
DoS occurs
Insufficient Dos prevention
Malcode introduced by hacker via email
Malcode installs a backdoor
Malcode introduced by hacker via web application
Insufficient virus protection
Use of web application
Hacker gets root access
Online system down
User data leaks to 3rd party
Hacker
Insufficient phyiscal security
Installing a backdoor in a company hardware
Malcode causes loss of data on DB
Personal identifiable information leaks to 3rd party
Incorportaing user-inputted text into SQL query
Changes email adresses via SQL injection
Integrity of database system corrupted
Incompetent developers
inserting themselves as a man-in-the-middle within the range of the unencrypted Wi-Fi accept point
Hacker can access data exchange
* gets teacher authorization
Web site
User Satisfaction
User DB
Reputation
Compliance

Figure 5D.1

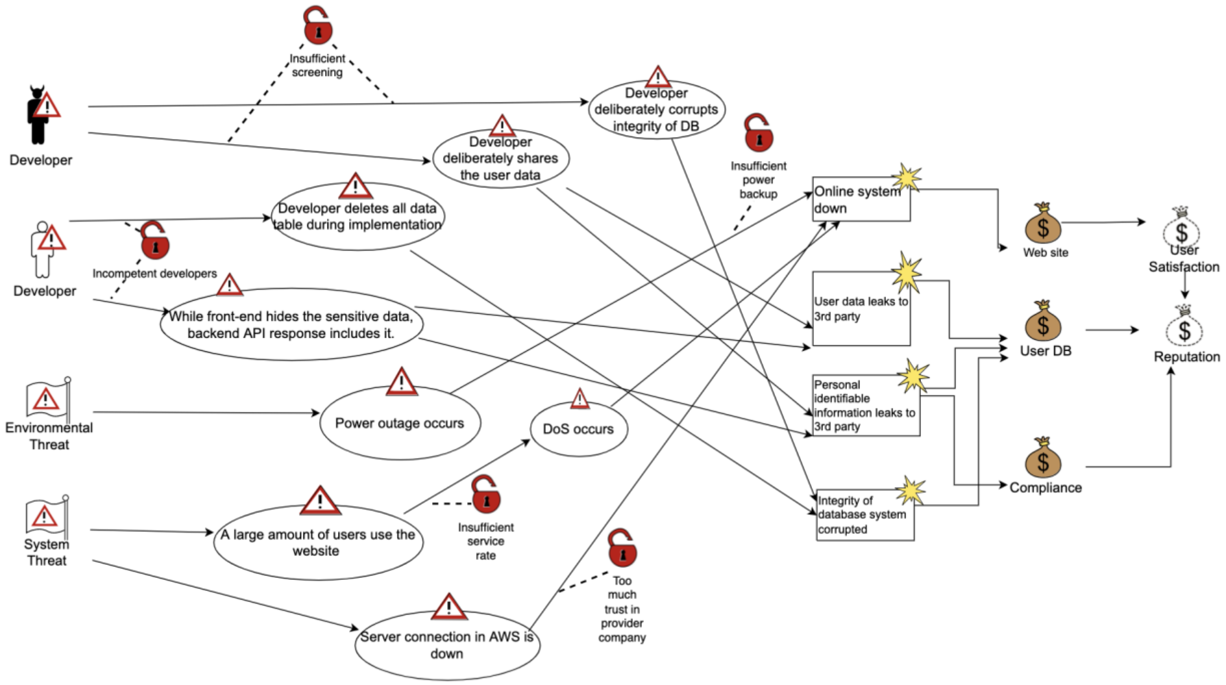
Insufficient screening
Developer
Developer deliberately corrupts integrity of DB
Developer deliberately shares the user data
Insufficient power backup
Developer deletes all data table during implementation
Online system down
Developer
Incompetent developers
Web site
User Satisfaction
User data leaks to 3rd party
While front-end hides the sensitive data, backend API response includes it.
User DB
Reputation
Environmental Threat
Power outage occurs
DoS occurs
Personal identifiable information leaks to 3rd party
Compliance
System Threat
A large amount of users use the website
Insufficient service rate
Integrity of database system corrupted
Too much trust in provider company
Server connection in AWS is down

Figure 5D.2

6. Risk Estimation Diagram

This diagram shows the severity of risks threats and vulnerabilities causing.

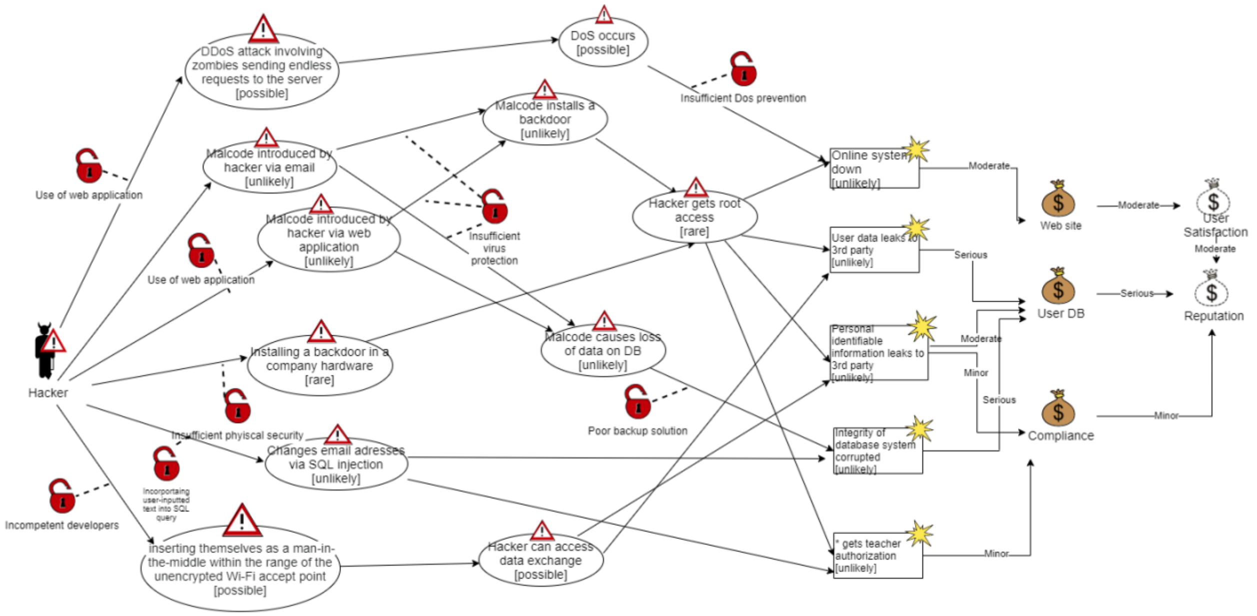
DDoS attack involving zombies sending endless requests to the server [possible]
DoS occurs [possible]
Insufficient Dos prevention
Malcode installs a backdoor [unlikely]
Malcode introduced by hacker via email [unlikely]
Use of web application
Malcode introduced by hacker via web application [unlikely]
Insufficient virus protection
Use of web application
Hacker gets root access [rare]
Online system down [unlikely]
User data leaks to 3rd party [unlikely]
Hacker
Installing a backdoor in a company hardware [rare]
Malcode causes loss of data on DB [unlikely]
Personal identifiable information leaks to 3rd party [unlikely]
Poor backup solution
Insufficient phyiscal security
Changes email adresses via SQL injection [unlikely]
Integrity of database system corrupted [unlikely]
Incorportaing user-inputted text into SQL query
Incompetent developers
Inserting themselves as a man-in-the-middle within the range of the unencrypted Wi-Fi accept point [possible]
Hacker can access data exchange [possible]
* gets teacher authorization [unlikely]
Moderate
Serious
Moderate
Minor
Serious
Minor
Web site
User DB
Compliance
Moderate
Serious
Minor
User Satisfaction
Moderate
Reputation

Figure 6.1

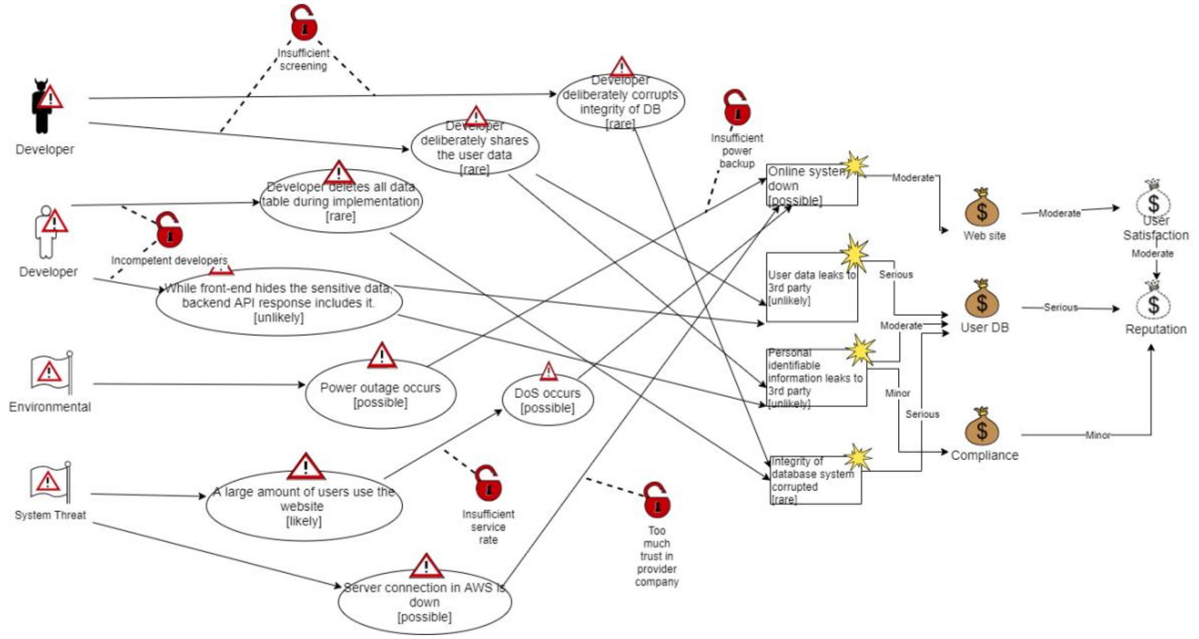
Insufficient screening
Developer
Developer deliberately corrupts integrity of DB [rare]
Developer deliberately shares the user data [rare]
Insufficient power backup
Developer
Developer deletes all data table during implementation [rare]
Incompetent developers
While front-end hides the sensitive data, backend API response includes it. [unlikely]
Online system down [possible]
User data leaks to 3rd party [unlikely]
Personal identifiable information leaks to 3rd party [unlikely]
Integrity of database system corrupted [rare]
Environmental
Power outage occurs [possible]
DoS occurs [possible]
System Threat
A large amount of users use the website [likely]
Insufficient service rate
Too much trust in provider company
Server connection in AWS is down [possible]
Moderate
Serious
Moderate
Minor
Serious
Web site
User DB
Compliance
Moderate
Serious
Minor
User Satisfaction
Moderate
Reputation

Figure 6.2

7. Risk Treatment Using Risk Diagrams

7A. Evaluation of the Identified Risks

In the below tables you can see identified risks placed into Risk Functions for each asset.

7A.1 Risk Evaluation for the User Database Asset

Consequence / Likelihood	Insignificant	Minor	Moderate	Serious	Catastrophic
Rare	🟩	🟩	🟧	Integrity of database is corrupted	🟥
Unlikely	🟩	🟩	Personal Identifiable information leaks to 3rd party	User data leaks to 3rd party	🟥
Possible	🟩	🟧	🟧	🟥	🟥
Likely	🟩	🟧	🟥	🟥	🟥
Certain	🟩	🟧	🟥	🟥	🟥

Figure 7A.1

7A.2 Risk Evaluation for the Web Site Asset

Consequence / Likelihood	Insignificant	Minor	Moderate	Serious	Catastrophic
Rare	🟩	🟩	🟧	🟧	🟥
Unlikely	🟩	🟩	🟧 Online System Down	🟧	🟥
Possible	🟩	🟩	🟧	🟥	🟥
Likely	🟩	🟧	🟧	🟥	🟥
Certain	🟩	🟧	🟥	🟥	🟥

Figure 7A.2

7A.3 Risk Evaluation for the Compliance Asset

Consequence / Likelihood	Insignificant	Minor	Moderate	Serious	Catastrophic
Rare	🟩	🟩	🟧	🟧	🟥
Unlikely	🟩	🟩 Personal Identifiable Information leaks to 3rd party	🟧	🟧	🟥
Possible	🟩	🟩	🟧	🟥	🟥
Likely	🟩	🟧	🟧	🟥	🟥
Certain	🟩	🟧	🟥	🟥	🟥

Figure 7A.3

7A.4 Risk Evaluation for the User Satisfaction Asset

Consequence/ Likelihood	Insignificant	Moderate	Minor	Catastrophic
Rare	🟩	🟩	🟩	🟧
Unlikely	🟩	🟩	🟩	🟥
Possible	🟩	🟩 Online system goes down	🟧	🟥
Likely	🟩	🟧	🟥	🟥
Certain	🟧	🟧	🟥	🟥

Figure 7A.4

7A.5 Risk Evaluation for the Reputation Asset

Consequence / Likelihood	Insignificant	Minor	Moderate	Serious	Catastrophic
Rare	🟩	🟩	🟧	🟥 Integrity of database is corrupted	🟥
Unlikely	🟩	🟩 Personal Identifiable Information leaks to 3rd party	🟧 Online system down	🟥 User data leaks to 3rd party	🟥
Possible	🟩	🟩	🟧	🟥	🟥
Likely	🟩	🟧	🟧	🟥	🟥
Certain	🟩	🟧	🟥	🟥	🟥

Figure 7A.5

7B. Summarizing the Risk Picture

Summarization of the Risk Picture containing identified risks and assets they are affecting.

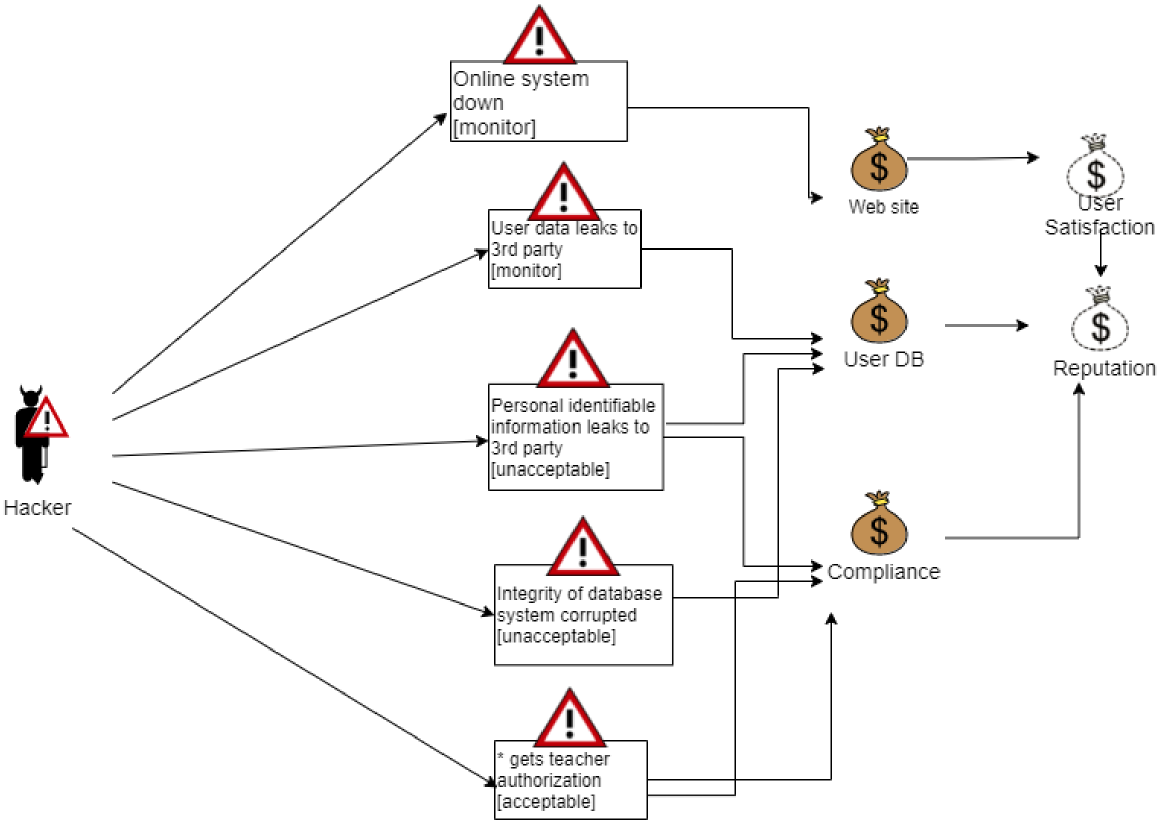
Online system down [monitor]
User data leaks to 3rd party [monitor]
Personal identifiable information leaks to 3rd party [unacceptable]
Integrity of database system corrupted [unacceptable]
* gets teacher authorization [acceptable]
Hacker
Web site
User DB
Compliance
User Satisfaction
Reputation

Figure 7B.1

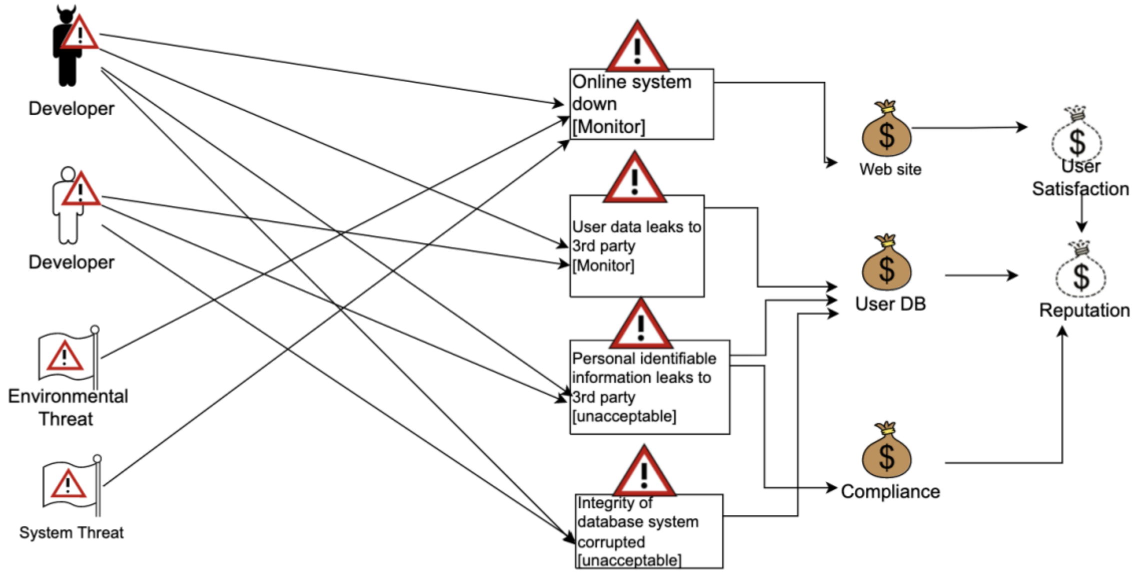
Developer
Developer
Environmental Threat
System Threat
Online system down [Monitor]
User data leaks to 3rd party [Monitor]
Personal identifiable information leaks to 3rd party [unacceptable]
Integrity of database system corrupted [unacceptable]
Web site
User DB
Compliance
User Satisfaction
Reputation

Figure 7B.2

8A. Risk Treatment Diagrams

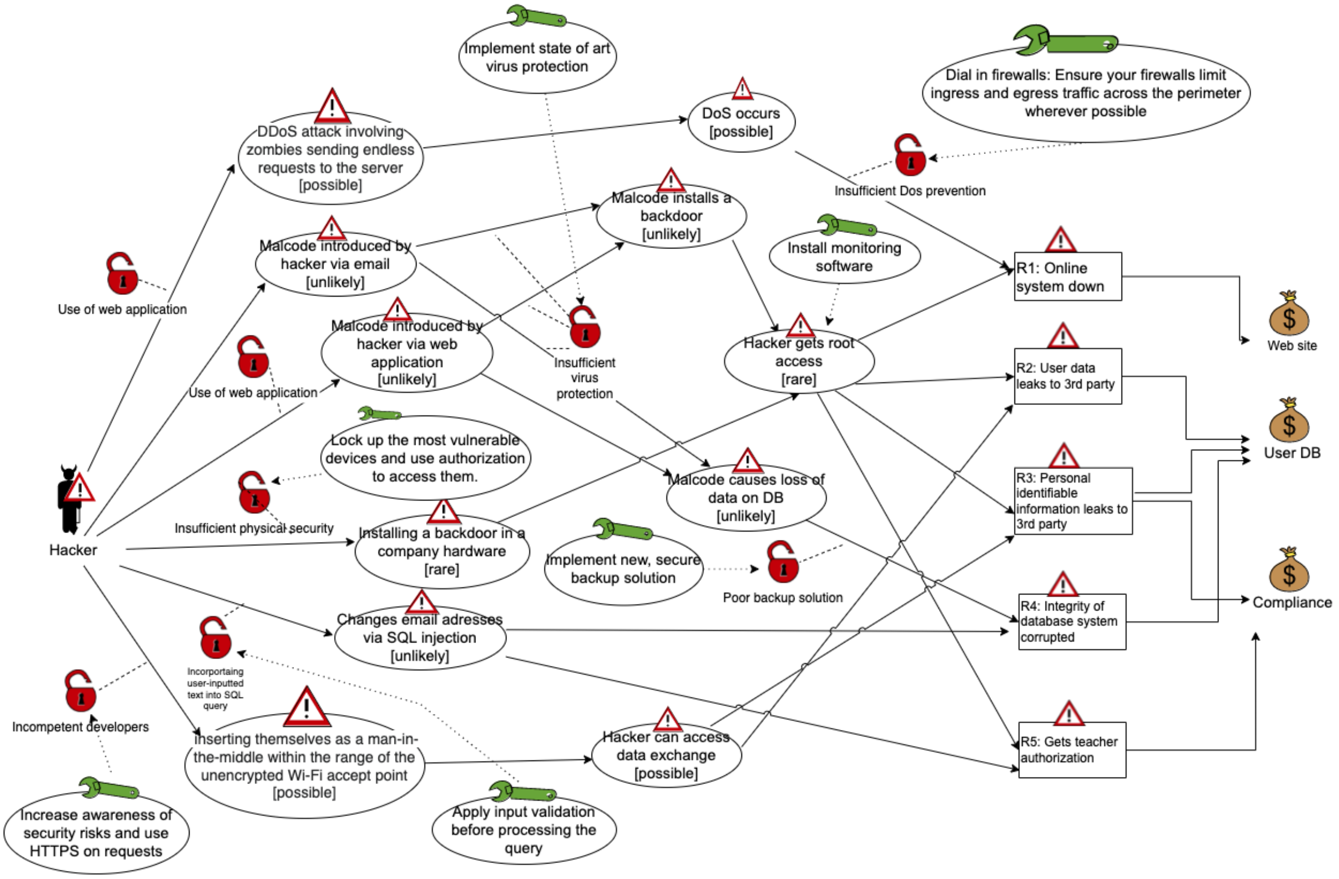
Implement state of art virus protection
Dial in firewalls: Ensure your firewalls limit ingress and egress traffic across the perimeter wherever possible
DDoS attack involving zombies sending endless requests to the server [possible]
DoS occurs [possible]
Insufficient Dos prevention
Malcode installs a backdoor [unlikely]
Malcode introduced by hacker via email [unlikely]
Install monitoring software
R1: Online system down
Web site
Use of web application
Malcode introduced by hacker via web application [unlikely]
Insufficient virus protection
Hacker gets root access [rare]
R2: User data leaks to 3rd party
Use of web application
Lock up the most vulnerable devices and use authorization to access them.
User DB
Malcode causes loss of data on DB [unlikely]
R3: Personal identifiable information leaks to 3rd party
Insufficient physical security
Hacker
Installing a backdoor in a company hardware [rare]
Implement new, secure backup solution
Poor backup solution
R4: Integrity of database system corrupted
Compliance
Changes email adresses via SQL injection [unlikely]
Incorportaing user-inputted text into SQL query
Incompetent developers
Inserting themselves as a man-in-the-middle within the range of the unencrypted Wi-Fi accept point [possible]
Hacker can access data exchange [possible]
R5: Gets teacher authorization
Increase awareness of security risks and use HTTPS on requests
Apply input validation before processing the query

Figure 8A.1

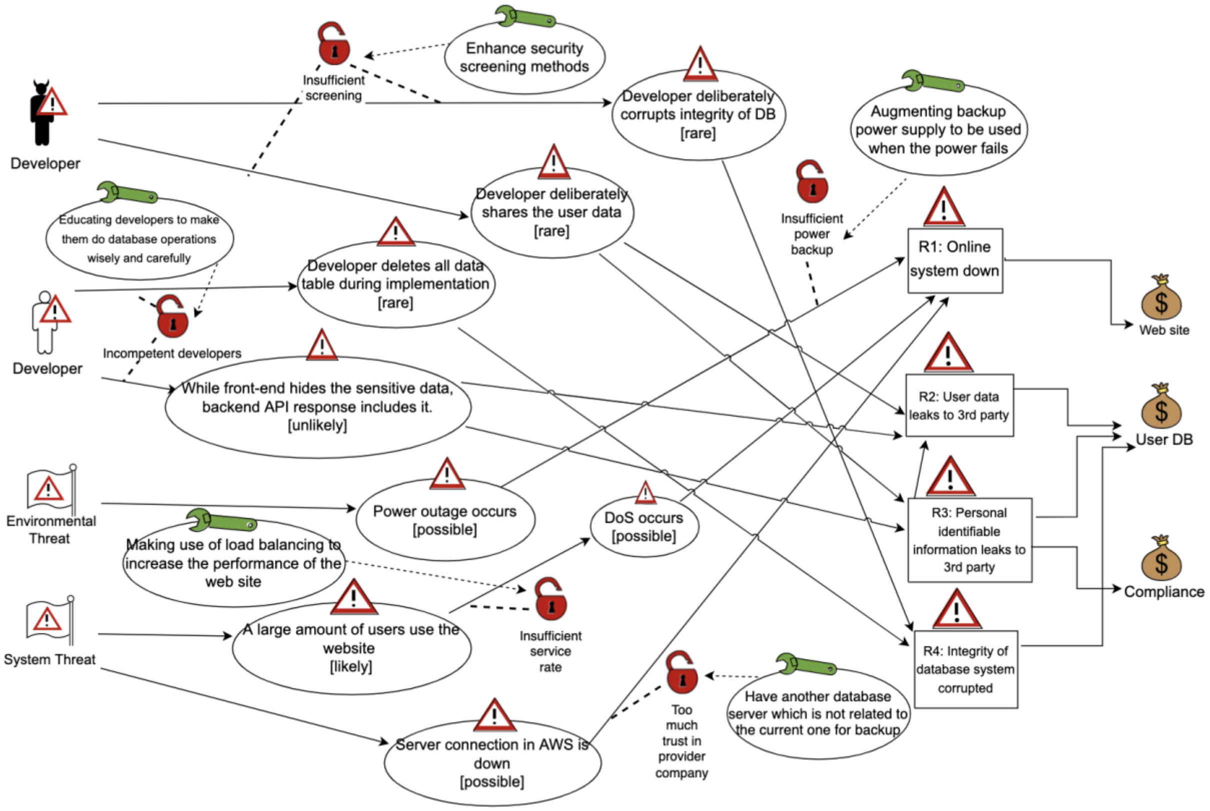
Enhance security screening methods
Insufficient screening
Developer deliberately corrupts integrity of DB [rare]
Augmenting backup power supply to be used when the power fails
Developer
Developer deliberately shares the user data [rare]
Insufficient power backup
Educating developers to make them do database operations wisely and carefully
Developer deletes all data table during implementation [rare]
R1: Online system down
Web site
Incompetent developers
Developer
R2: User data leaks to 3rd party
While front-end hides the sensitive data, backend API response includes it. [unlikely]
User DB
Environmental Threat
Power outage occurs [possible]
DoS occurs [possible]
R3: Personal identifiable information leaks to 3rd party
Making use of load balancing to increase the performance of the web site
Compliance
Insufficient service rate
System Threat
A large amount of users use the website [likely]
R4: Integrity of database system corrupted
Have another database server which is not related to the current one for backup
Too much trust in provider company
Server connection in AWS is down [possible]

Figure 8A.2

8. Risk Treatment Using Treatment Diagrams

8B. Risk Overview Diagrams

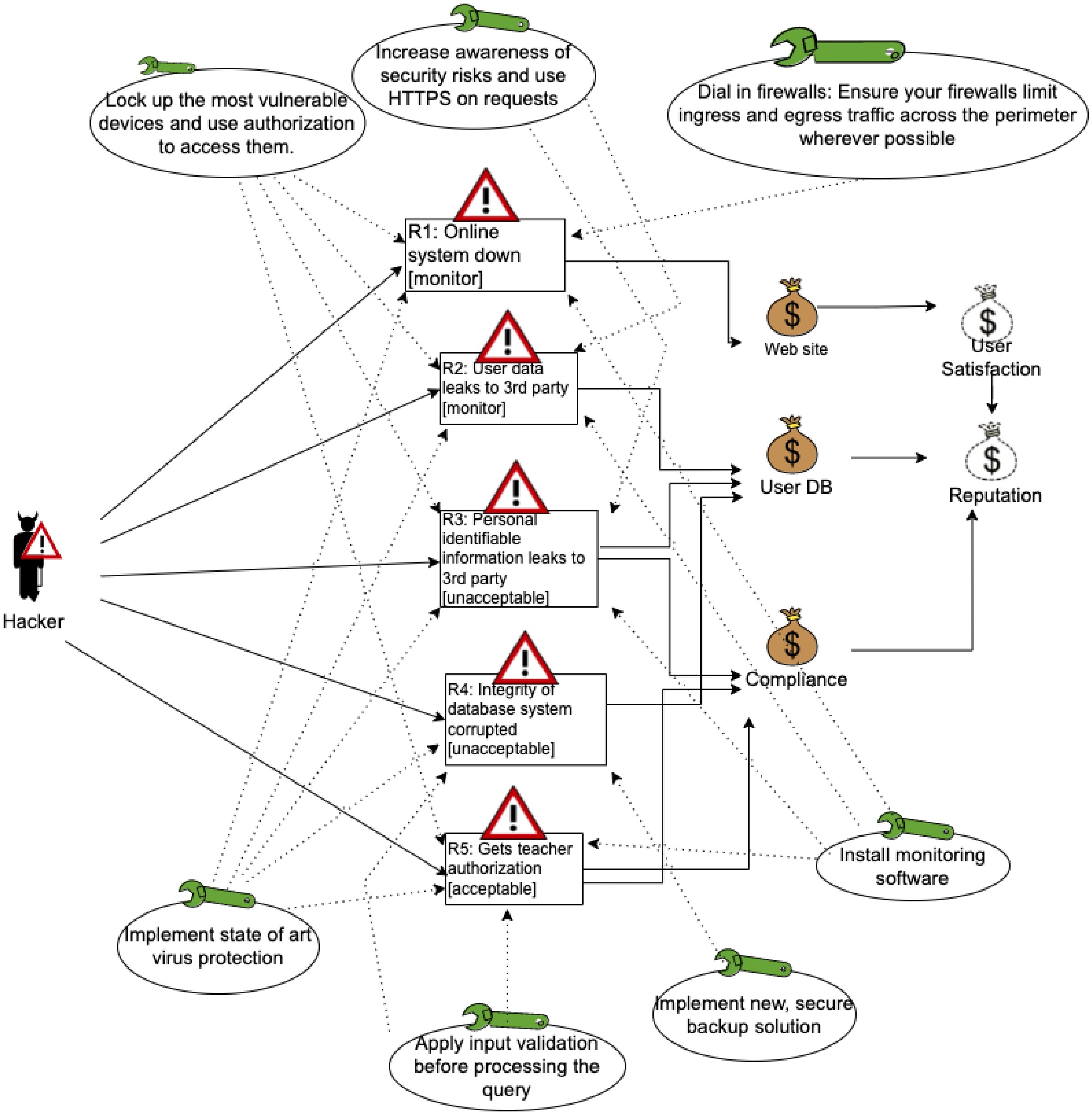
Lock up the most vulnerable devices and use authorization to access them.
Increase awareness of security risks and use HTTPS on requests
Dial in firewalls: Ensure your firewalls limit ingress and egress traffic across the perimeter wherever possible
R1: Online system down [monitor]
R2: User data leaks to 3rd party [monitor]
R3: Personal identifiable information leaks to 3rd party [unacceptable]
R4: Integrity of database system corrupted [unacceptable]
R5: Gets teacher authorization [acceptable]
Hacker
Web site
User Satisfaction
User DB
Reputation
Compliance
Install monitoring software
Implement state of art virus protection
Implement new, secure backup solution
Apply input validation before processing the query

Figure 8B.1

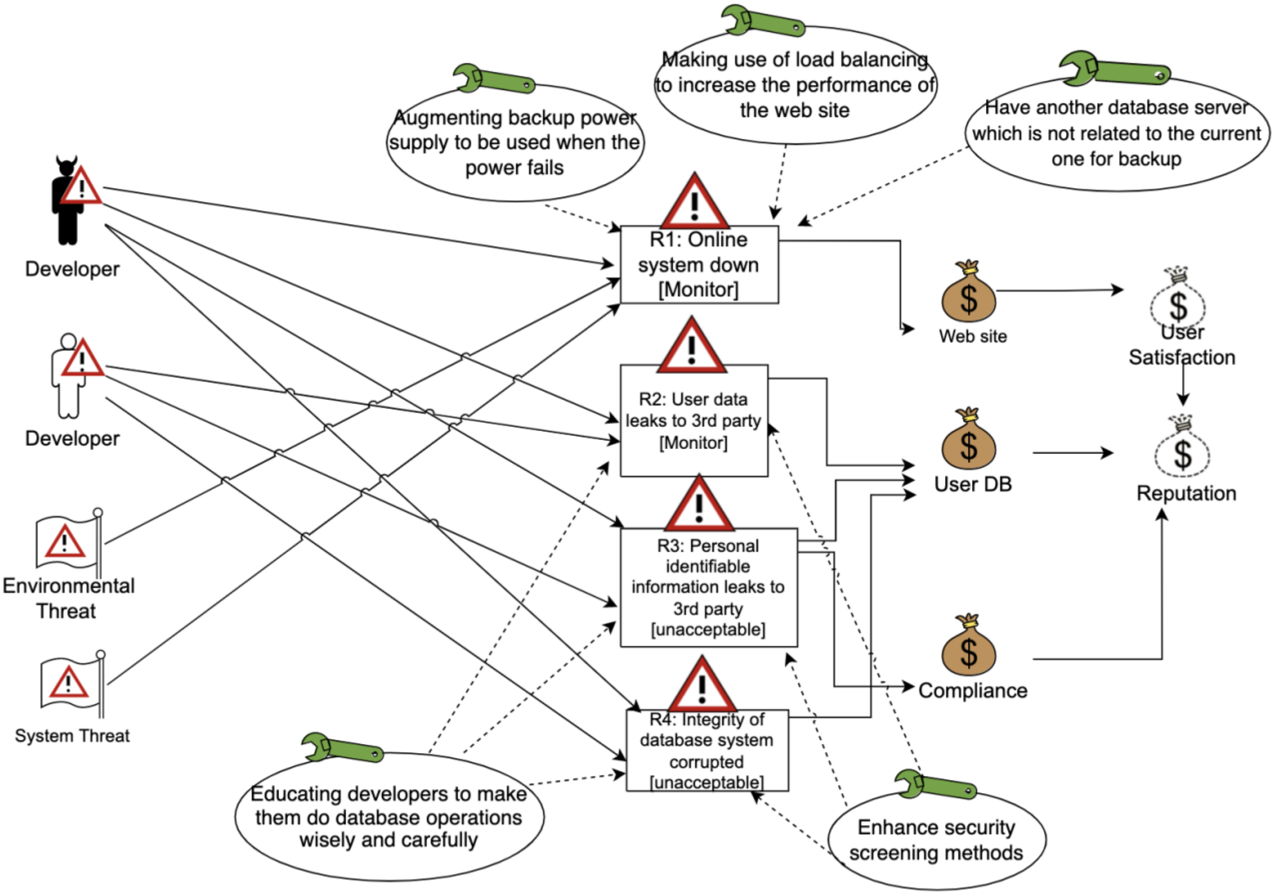
Making use of load balancing to increase the performance of the web site
Augmenting backup power supply to be used when the power fails
Have another database server which is not related to the current one for backup
Developer
Developer
Environmental Threat
System Threat
R1: Online system down [Monitor]
R2: User data leaks to 3rd party [Monitor]
R3: Personal identifiable information leaks to 3rd party [unacceptable]
R4: Integrity of database system corrupted [unacceptable]
Web site
User Satisfaction
User DB
Reputation
Compliance
Educating developers to make them do database operations wisely and carefully
Enhance security screening methods

Figure 8B.2

8C. Treatment Evaluation

For the treatment evaluation, costs and risk reduction degrees of treatments are considered to give the final answer about their implementation decision. The anticipated cost levels are categorized into three: low, medium and high. After taking into account how costly and effective the treatment is, the decision on whether it is advised to or not implement the treatment is written down. In the second table, the reasons why those treatments are advised to implement or not are given.

Treatment	Cost	Risk	Risk Reduction	Select to Implement
T1: Lock up the most vulnerable devices and use authorization to access them.	Low	R1 R2 R3 R5	R1: Monitor to Acceptable R2: Monitor to Acceptable R3: Unacceptable to Acceptable R5: Acceptable to Acceptable	Yes
T2: Increase awareness of security risks and use HTTPS on requests	Low	R2 R3	R2: Monitor to Acceptable R3: Unacceptable to Acceptable	Yes
T3:Dial in firewalls: Ensure your firewalls limit ingress and egress traffic across the perimeter wherever possible	Medium	R1	R1: Monitor to Acceptable	Yes
T4: Implement	Low	R1	R1: Monitor to	Yes

state-of-the-art virus protection		R2 R3 R4 R5	Acceptable R2: Monitor to Acceptable R3: Unacceptable to Monitor R4: Unacceptable to Monitor R5: Acceptable to Acceptable	
T5: Apply input validation before processing the query	Low	R4 R5	R4: Unacceptable to Acceptable R5: Acceptable to Acceptable	Yes
T6: Implement new, secure backup solution	High	R4	R4: Unacceptable to Monitor	No
T7: Install monitoring software	Medium	R1 R2 R3 R5	R1: Monitor to Acceptable R2: Monitor to Acceptable R3: Unacceptable to Monitor R5: Acceptable to Acceptable	Yes
T8: Have another database server which is not related to the current one for backup	Medium	R1	R1: Monitor to Acceptable	Yes
T9: Augmenting backup power supply to be used when the power fails	Medium	R1	R1: Monitor to Acceptable	Yes

T10: Making use of load balancing to increase the performance of the web site	Medium	R1	R1: Monitor to Acceptable	Yes
T11: Educating developers to make them do database operations wisely and carefully	Low	R2 R3 R4	R2: Monitor to Acceptable R3: Unacceptable to Acceptable R4: Unacceptable to Acceptable	Yes
T12: Enhance security screening methods	High	R2 R3 R4	R2: Monitor to Acceptable R3: Unacceptable to Acceptable R4: Unacceptable to Acceptable	No

Table 8C.1

Treatment	Reasons for their implementation decisions
T1: Lock up the most vulnerable devices and use authorization to access them.	The vulnerable devices that are not locked or not having authorization systems, are open to physical attacks. Attacker might be someone from inside or outside. Locking them in a room is the easiest way to prevent physical attacks. So, it should be done in the first place.
T2: Increase awareness of security risks and use HTTPS on requests	The developers should know how to make their code more secure. It also costs less to educate them. Using HTTPS on requests also costs less. Moreover, HTTPS uses the SSL/TLS protocol to encrypt the communications . In this way, you can prevent attacks in a wide range. That’s why this treatment is advised to be implemented.

T3:Dial in firewalls: Ensure your firewalls limit ingress and egress traffic across the perimeter wherever possible	Monitoring, controlling and restricting traffic leaving or incoming a network are called Egress and Ingress filtering. Therefore, having that limitation in the firewalls can help to ensure that only legitimate traffic is allowed. However, it costs to have a firewall and filtering. That’s why, it might not be implemented or it can be saved for later.
T4: Implement state-of-the-art virus protection	Implementation of state-of-the-art has low cost. Moreover, it protects the direct and indirect assets from all the risks. That’s why it should be implemented.
T5: Apply input validation before processing the query	Applying input validation doesn’t cost anything. Only thing to do, having validation functions in your backend code before processing the requests. It may help to prevent SQL-injection.
T6: Implement new, secure backup solution	It is needed to pay more to have a new backup solution such as having another cloud storage or new database server. Since the university budget is limited, it is chosen not to apply.
T7: Install monitoring software	It is needed to pay for new monitoring software. However, it doesn’t cost much. Also, there are free options. Moreover, it can help to deal with different kinds of attacks and risks. That’s why this treatment is advised to be implemented.
T8: Have another database server which is not related to the current one for backup	If the system has a backup database, it would allow the website to continue functioning even if the main database fails. This is a service whose main purpose is to save and view student and lecturer records thus having a backup database would have a big role in its functioning. Furthermore, not all database servers require payment for their usage so it wouldn’t be too much of a burden for the constitution. That’s why this treatment is advised to be implemented.
T9: Augmenting backup power supply to be used when the power fails	Backup power supply is required for almost all services if the service serves a crucial purpose for the constitution. This backup power supply would not cost too much and it can also be used for other web services of the constitution on demand. That’s why this treatment is advised to be implemented.
T10: Making use of load balancing to increase the performance of the web site	For web services like this where they often experience overload due to too much user interaction on the system, load balancing is a good solution to distribute the workload. AWS and Azure provide load balancing solutions and they require small prices thus it also wouldn’t be an expensive treatment for this web service so this treatment is advised to be implemented.

T11: Educating developers to make them do database operations wisely and carefully	Giving courses on how to better and safely use a database would be not only cheap but also very effective because this education would only be given for several times but their effects would last long and the possibility of the database losing its integrity due to the lack of knowledge and carelessness of the developer would decrease significantly. Thus it is advised to implement this treatment.
T12: Enhance security screening methods	This treatment is not advised to be implemented because even though it mitigates the risks significantly, tracking each action of the developer would have a high cost.

Table 8C.2