**Stride Model**

|  |  |
| --- | --- |
| Spoofing | * Monitoring unencrypted user credentials on network * Phishing attacker can catch user credentials and spoof its identity * If DB is attack or leaked and credentials are not enctypted+salted then it would cause identity theft * SQL injection could cause the same if not handled properly * Stored XXS causing to send credentials to attacker each time its typed * If password is not hashed+salted could be vulnerable to dictionary attacks * Also if password is not complicated enough it could be broken quite easily with brute force * Lack of logic of password typing retries |
| temper | * Stored XXS means tempering with DB data for good and can cause various kinds of damage * Altering password of users stored in DB or in traffic |
| repudability | * One can delete data from DB with no records of who did it * One can add users to DB with no premission check * Attacker can store XSS with no traces except for the data which was stored * If there are permissions and one of them was changed you can’t decide who cause this |
| Information disclosure | * SQL injection can cause data leakage of details and passwords of users * The structure of DB can disclose data about the users and website * XXS can disclose to attacker about each new user who is register to the website * Also in phishing attack |
| DOS | * Deletion of partial\all of data in DB would leave users unserviced * Brute force network attacks method without any defence mechanism would cause sirious DOS * Altering passwords or email tempering with details would leave certain users unable to use the system |
| Eleavation of privilege | * Password theft of users will grant them unauthorized access to info and to alter data * SQL injection can reveal data which attacker shouldn’t know * Unencrypted password would ease on attacking and cracking it * Taking advantage of bad routing logic |

**DREAD Model**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Threat | | | D | R | E | A | D | total | | rating | |
| * Monitoring unencrypted user credentials on network   The scenario is easy to use but hard to achieve access and set up | | | 5 | 2 | 6 | 7 | 0 | 20 | | high | |
| * Phishing attacker can catch user credentials and spoof its identity   It's easy to use a prepared tools for this and it depends on social engineering | | | 5 | 7 | 8 | 5 | 8 | 32 | | medium | |
| If DB is attacked or leaked and credentials are not enctypted+salted then it would  If such an attack will occur it will severely affect many users | | | 9 | 2 | 8 | 10 | 0 | 29 | | high | |
| * Stored XXS means tempering with DB data for good and can cause various kinds of damage   This attack vector is not so complex but causes quite a damage | | | 8 | 8 | 1 | 7 | 6 | 30 | | high | |
| Altering password of users stored in DB or in traffic  Monitoring via traffic requires a lot of setup | | | 8 | 8 | 9 | 2 | 1 | 28 | | medium | |
| * One can delete data from DB with no records of who did it   Deleting data is a lot of damage and in this case very hard to discover | | | 9 | 2 | 5 | 7 | 1 | 24 | | high | |
| * One can add users to DB with no permission check.   If permission logic is insufficient one can add himself as a privileged user and from that point may cause a lot of damage to the system | | | 6 | 3 | 6 | 1 | 1 | 17 | | high | |
| * Deletion of partial or entire DB   Hard to execute but damage is significant | | | 10 | 2 | 4 | 10 | 10 | 36 | | High | |
| * Brute force network floods   Easy to execute and effective if not handled well and taken into considerations | | | 10 | 10 | 1 | 10 | 8 | 39 | | high | |

**Threats Modeling**

|  |  |  |  |
| --- | --- | --- | --- |
| # | Weakness point description | Relevant threat | score |
| 1 | Data not encrypted in TLS or other network protocol | **Temper**  changing the data as it comes to the DB  **spoofing**  attacker catches the credentials of the user  **DOS**  attacker can change the password as it comes, leaving user in DOS  **info disclosure**  can access user’s details and account  **EOP**  Can access stuff I’m not authorized to | 20 |
| 2 | Data unencrypted in DB\  Lack of logic of password complexity and retries | **Spoofing**  Can access on behalf of the user  **Info disclosure**  Can access account and find personal info  **DOS and Temper and Reputability**  Can change the password once I’m logged in as the user | 29 |
| 3 | No access control logic | **Reputability**  Change of access to different users without any traces  **Info disclosure**  Access to unauthorized data and control | 17 |
| 4 | No 2-factor authentication | **Spoofing**  If the network is monitored and password is not encrypted or too weak, it’s easier to spoof to another user. |  |
|  | No logs or documentation of who made changes | **Temper + repudiation**  Data will change with no traces of who made those changes. |  |
| 5 | No DB backup – single point of faliure | **DOS**  If a single instance of DB is deleted or not responding for some reason the entire system is incapable of serving users |  |
| 6 | No requirements for counter-mechanisms against brute-force network attacks in manner of flooding, etc. | **DOS**  If such attack succeeds it could cause a significant denial of service |  |