

Моделовање претњи је поступак коришћења хипотетичких сценарија, дијаграма система, и тестирања да би се повећала сигурност система и података. Раним откривањем рањивости

* њиховом превенцијом, смањујемо потенцијалне губитке који би се десили занемаривањем или неоткривањем истих.

Знајући рањивости нашег система самим тим потпомажемо смањењу ризика, као и предлагањем мера, како би се побољшала *Cyber* сигурност и поверење у кључне пословне системе.

* процесу креирања модела претњи, пожељно је испратити следеће кораке:
  + Неопходно је идентификовати ресурс. Ресурси у нашем систему могу бити подаци рачуна, интелектуална својина и слично
  + Формирање дијаграма система - дијаграми омогућавају поглед на системе и токове података напада на високом нивоу
  + Анализа претњи - користе се методе моделовања претњи за даљу анализу одређених врста претњи, идентификовање потенцијалних претњи и мапирање токова података
  + Управљање ризиком и одређивање приоритета. Многи алати за моделовање претњи дају оцене опасности и податке за израчунавање ризика.
  + Идентификација могућих исправки - када су идентификовани делови система, или ресурси који су најзначајнији за организацију, будући кораци могу бити очигледни

Зависно од потреба пројекта и различитих организација, користићемо различите начине моделовања претњи. Најстарија, а уједно и доста често коришћена јесте метода Stride. Користи се за идентификацију системских ентитета, догађаја и граница система унутар којег радимо. Ова метода заправо представља скраћеницу од *Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege.*

* *Spoofing –* Лажирање идентитета
* *Tampering –* Неауторизована промена података на диску, мрежи и сл.
* *Repudiation –* Негирање одговорности за акције које су почињене
* *Information Disclosure –* Одавање информација некоме ко није ауторизован да им приступи

* *Denial of Service –* Ометање ресурса који су потребни да би сервис био доступан корисницима
* *Elevation of Privilege –* Омогућавање приступа сервису некоме ко није ауторизован

1. *Анализа апликације*

Након увода о општим карактеристикама модела претњи, у овом поглављу ставићемо акценат на анализу апликације коју смо развијали за предмет Системи електронског плаћања.

Оно што је пожељно урадити јесте декомпозиција апликације на делове и посматрање њихове интеракције са окружењем. Потребно је прикупити информације о улазним тачкама, елементима и зависностима (dependencies) нашег система.

Најпре ћемо размотрити све кориснике који имају контролу над системом. Поред администраторa, систем може користити и купац. Ниво поверења може имати вредности од 1 до 2, при чему вредност 1 представља најмањи ниво поверења, док 2 представља највиши ниво.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | ***Ниво поверења*** | | ***Улога*** |  | ***Опис улоге*** | |  |
|  | *2* |  | *Купац* |  | *Корисник има могућност* |  |  |
|  |  |  | *да излистава производе, и да их купује* |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| 1 | |  | *Администратор* |  | *Има могућност да креира* | |  |
|  |  | *Нове веб продавнице, као и да додаје методе плаћања за сваки од њих* | |  |
|  |  |  |  |  |  |

Након дефинисања корисника у систему и нивоа поверења, следећи корак јесте идентификовање ресурса система. Под ресурсом у систему подразумевамо оно што има вредност у систему, а самим тим има ризик од малициозних напада.

|  |  |
| --- | --- |
| ***Назив ресурса*** | ***Ниво поверења*** |
|  |  |
| *База података* | *4* |
|  |  |
| *Логин креденцијали* | *2,3,4* |
|  |  |
| *Лични подаци корисника* | *2,3,4* |
|  |  |
| *Подаци о картицама* | *4* |
|  |  |
| *Подаци о производима* | *1,2,3,4* |
|  |  |
| *Логови система* | *4* |
|  |  |
| *Конфигурациони фајлови* | *4* |
|  |  |
| *Подаци о чланаринама* | *2,3,4* |
|  |  |
| *Подаци о начинима плаћања* | *2,3,4* |
|  |  |

Након што смо дефинисали све битне ресурсе унутар система, сада је неопходно да дефинишемо улазне тачке, које би малициозни корисник потенцијално могао да искористи. Као улазне тачке посматраћемо све податке које корисник може да уноси/модификује и пошаље на наш сервер.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Назив** |  | **Ниво поверења** | |  |
|  |  |  |  |  |
| *Страница за пријаву* |  | *1,2,3,4* |  |  |
|  |  |  |  |  |
| *Форма за креирање нове методе плаћања* | *3* | |  |  |
|  |  |  |  |  |
| *Форма за креирање нове веб продавнице* |  | *4* |  |  |
|  |  |  |  |
|  |  |  |  |  |
|  |  | |  |  |
| *HTTP* | *1,2,3,4* | |  |  |



1. *Дијаграм тока података*

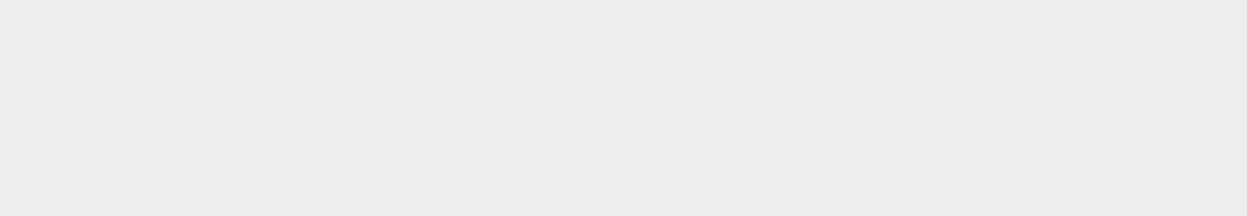
Дијаграм тока података представља графички приказ тока података у информационом систему. На овом дијаграму можемо да уочимо ток корисничких захтева који након ауторизације пристижу на систем.

Корисници се аутентификују путем својих креденцијала, где након препознавања од стране система, добијају токен који користе у даљем раду. Приликом слања захтева на систем, корисник мора да приложи свој токен, како би био ауторизован и како би систем утврдио које активности су дозвољене том кориснику. Сваки корисник има додељену улогу, која са собом носи пермисије. У зависности од пермисија које поседује, а које ће бити учитане са послатим токеном, корисник може да приступи одређеним *endpoint-*има.



* 1. *Анализа претњи*
* следећој табели, дефинисаћемо претње, односно могуће нападе на наш систем, као и начине на које је могуће одбранити се од истих.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Назив претње** | | **Начин одбране** | |  |
|  |  |  |  |  |  |
|  | *SQL Injection* |  | *Коришћење Hibernate-а и JPA слоја који* |  |  |
|  |  |  | *имају уграђене механизме за спречавање* |  |  |
|  |  |  | *ове врсте напада. Анотације на DTO* |  |  |
|  |  |  | *објекте. Дефинисан интерфејс* |  |  |
|  |  |  | *(SQLInjectionSafe) који ће препознати и* |  |  |
|  |  |  | *одбити малициозан захтев* |  |  |
|  |  |  |  |  |  |
|  | *XSS напад* | | *Валидацијом input параметара кориснику* | |  |
|  |  |  | *није дозвољен унос специјалних карактера* | |  |
|  |  |  | *или <script> тагова.* | |  |
|  |  |  |  | |  |
|  |  |  |  | |  |
|  |  | |  | |  |
|  |  | |  |  |  |
|  | *Broken Access Control* | | *Заштита од науторизованог приступа* | |  |
|  |  |  | *ресурсима, решена је коришћењем* | |  |
|  |  |  | *механизма Role Based Access Control.* | |  |
|  |  | |  | |  |
|  | *Logging & Monitoring* |  | *Лог фајлови су заштићени коришћењем* |  |  |
|  |  |  | *АCL листа. Само админ има приступ* |  |  |
|  |  |  | *истим.* |  |  |
|  |  | |  |  |  |
|  | *Using Components with Known* | | *Koришћењем OWASP Dependency Check-a* | |  |
|  | *Vulnerabilities* | | *откривене су и отклоњене рањивости* | |  |
|  |  |  | *third party библиотека* | |  |
|  |  | |  | |  |
|  | *Bruteforce напад* |  | *Систем од корисника захтева лозинку која* |  |  |
|  |  |  | *задовољава патерн јаке лозинке.* |  |  |
|  |  |  |  |
|  |  |  |  |  |  |





1. *Анализа ризика*

На основу анализе претњи, креирана је табела ризика која садржи процену утицаја претње на систем (*C* - *concequence*), вероватноће (*L - likehood*) и процену ризика (*O – overal risk*). Могуће вредности су H (*high*), M (*medium*), L (*low*).

|  |  |  |  |
| --- | --- | --- | --- |
| **Назив претње** | **Утицај претње** | **Вероватноћа** | **Процена ризика** |
|  |  |  |  |
| *SQL Injection* | *H* | *H* | *H* |
|  |  |  |  |
| *XSS напад* | *H* | *M* | *H* |
|  |  |  |  |
| *Broken Authentication* | *H* | *M* | *H* |
|  |  |  |  |
| *Broken Access Control* | *H* | *H* | *H* |
|  |  |  |  |
| *Logging & Monitoring* | *M* | *M* | *M* |
|  |  |  |  |
| *Using Components with* | *M* | *L* | *M* |
| *Known Vulnerabilities* |  |  |  |
|  |  |  |  |
| *Bruteforce напад* | *M* | *H* | *H* |
|  |  |  |  |