2. Deep Generative Models [Електронний ресурс] – Режим доступу до ресурсу: https://deepgenerativemodels.github.io/.

УДК 004.492

Діденко. М. М., здобувач освіти,
Зелінська О.В., к.т.н, доцент, доцент кафедри інформаційних технологій

ТЕСТУВАННЯ СИСТЕМИ НА ПРОНИКНЕННЯ

Донецький національний університет імені Василя Стуса, м.Вінниця

З кожним днем в світі стає все більше і більше новітніх технологій, на яких дуже важно не звернути увагу. Все більше девайсів потребують нашу приватну інформацію для спрощення життя і зазвичай такі девайси захищаються одним-двома різної важкості паролями. Але виникає питання, що буде якщо хтось пройде нашу систему захисту на пристрої і отримає доступ до інформації. Саме для того, щоб ми не задавали собі такі питання кожного разу, як користуємось пристроєм, з’явився пантест або іншими словами, тестування системи на проникнення.

Кожен з нас може допуститись помилки і деякі з них можуть коштувати дуже великої ціни, як в матеріальному сенсі, так і в моральному. Але всі вони завжди матимуть наслідки. Саме для цього потрібна перевірка будь-якого продукту - тестування, перед тим як впевнено використовувати його в повсякденному житті.

Тест на проникнення або пантест – це метод оцінювання захищеності комп'ютерної системи чи мережі шляхом моделювання дій зовнішніх зловмисників з проникненням у неї. Тобто пантест передбачає моделювання реальних дій зловмисника, що дозволяє віднайти вразливості в системі та захистити подальшу експлуатацію. Цей тест дозволяє отримати оцінку про стан захищеності системи та експертний висновок[1].

Процес тестування можна розділити на такі етапи:

1. Планування та підготовка
2. Розвідка
3. Відкриття
4. Аналіз інформації і ризиків
5. Активні спроби вторгнення
6. Фінальний аналіз
7. Підготовка звіту

Планування та підготовка починається з певної цілі і задачі на тестування на зламостійкість.

Розвідка включає в себе аналіз наданої інформації. Дуже часто, тестувальник не має великої кількості інформації і весь аналіз розпочинається з перевірки доступності і можливості отримання інформації.

Відкриття. На цьому етапі використовуються автоматизовані інструменти для сканування цільових активів на вразливості. Цей етап дає змогу віднайти: мережі (додаткові мережі, сервери і т.п.), порти на девайсах.

Аналіз інформації та ризиків передбачає аналізування і оцінку інформації, що була зібрана етапами вище. Під час цього етапу враховуються такі елементи: визначення цілі на тест на проникнення, ймовірні ризики системи, час, необхідний для оцінки ризиків даної системи.

Активні спроби вторгнення являється найважливішим етапом серед усіх, який повинен бути виконаний з максимальною обережністю. В цьому етапі тестувальник починає атакувати систему, знаючи ризики, які були виявлені.

Фінальний аналіз. Цей крок в першу чергу враховує всі кроки, які були виконані до цього часу, та оцінку вразливостей у формі потенційних ризиків. Далі тестер рекомендує усунути уразливості та ризики.

Підготовка звіту повинна починатися із загальних процедур тестування, за якими слідує аналіз уразливостей та ризиків. Високі ризики та критичні уразливості повинні мати пріоритети, а потім нижній порядок. [2]

Взагалі є три типи тестування на проникнення:

- Чорний ящик
- Білий ящик
- Сірий ящик

Тестування чорного ящика – це коли тестувальник зовсім нічого не знає про систему, яку він збирається перевіряти. Через це, йому необхідно зібрати всю можливу інформацію за допомогою активного та пасивного пошуку. Код не перевіряється.

Тестування білого ящика полягає в комплексному тестуванні, оскільки тестувальнику надається інформація про мережу чи систему, чи пристрій. В даному типі, перевіряється код і проводиться тестування потоків, шляхів і т.д.

Тестування сірого ящику передбачає надання певної (частини або обмежену) інформації про об’єкт оцінки. Це розглядається як атаку зовнішнього хакера, котрий отримав несанкціонований доступ до інформації та документів певної організації.[3]

Типи тестів на проникнення:

- Соціальна інженерія (методи маніпулювання людьми)
- Веб-додаток (використання для виявлення проблемних місць веб-додатків та серверів)
- Мережева служба (тестування проникнення в мережу для виявлення можливості доступу хакерів)
- Клієнтська частина (використання тесту для тестування додатків)
- Віддалене підключення (тестування віртуальної віддаленої мережі, що може забезпечити доступ до підключеної системи)
- Бездротові мережі

• Тестування за допомогою автоматичного контролю та збору інформації.

Роблячи підсумки, потрібно пам'ятати, що навіть в самій захищеній системі можливо віднайти недоліки і проблемі місця, а як наслідок цього, потрібно час від часу перевіряти систему на проникність.

Список літератури

1. *Тест на проникнення. URL: https://inlnk.ru/1PZyY (Дата звернення: 13.11.21)*
2. *Тест на проникновение – Краткое руководство. URL: https://inlnk.ru/Jj8ZR (Дата звернення: 13.11.21)*
3. *Яцків В.В. Тестування комп'ютерних систем на проникнення. С.5-8*

УДК 519.852+519.674:004.942

Ємельянова А.О., здобувач освіти,
Зелінська О.В., к.т.н, доцент, доцент кафедри інформаційних технологій

ПОБУДОВА ГРАФІЧНОЇ МОДЕЛІ ЗАДАЧ ЛІНІЙНОГО ПРОГРАМУВАННЯ

Донецький національний університет імені Василя Стуса, м. Вінниця

Застосування задач лінійного програмування в останні роки все більше набуває широкого застосування через необхідність оптимізації процесів, підвищення ефективності роботи, наукового аналізу та впровадження автоматизації, які відіграють важливу роль в успішному функціонуванні підприємства. Це є невід'ємним процесом, який дозволяє конкретизувати інформацію на шляху до вдосконалення робочого процесу в різних галузях, таких як: економіка, енергетика, перевезення, виробництва тощо.

Задача лінійного програмування (ЗЛП) – це оптимізаційна задача, суть якої полягає у знаходженні екстремуму лінійної функції при наявності цільової функції

$$\mathrm{F}(x) = c_1x_1 + c_2x_2 + \cdots + c_nx_n \qquad (1)$$

та допустимої множини, обмеженої рівностями чи нерівностями

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n\{\leq, =, \geq\}b_1, \qquad (2)$$

і обмеженнями невід'ємності $x_1 \geq 0, x_2 \geq 0, \ldots, x_n \geq 0,$ включаючи невід'ємні праві частини.

Найпоширенішими способами вирішення таких задач є: графічний метод і симплекс-метод [2]. Тому метою даної роботи є демонстрація вирішення оптимізаційної задачі лінійного програмування шляхом побудови графічної моделі.