ХАРЬКІВСКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ

ТЕСТУВАННЯ НА ПРОНИКНЕННЯ (ПЕНТЕСТИНГ)

ЛЕКЦІЯ 5

Доцент кафедри кібербезпеки та ІТ
к.т.н. Лимаренко Вячеслав Володимирович
к.т. 066-0708586 (Viber, Telegram)

Що таке тестування на проникнення ?

Тестування на проникнення (penetration testing, pentest – тести на подолання захисту) – це детальний аналіз мережі і систем з точки зору потенційного зловмисника. Суть тесту полягає в санкціонованій спробі обійти існуючий комплекс засобів захисту інформаційної системи.

Тест на проникнення, пентест (анг. Penetration Testing) – це тип комп'ютерних тестувань безпеки, спрямованих на пошук та виявлення потенційних загроз, недоліків, несправностей (багів, помилок) та вразливостей у захисті електронно-обчислювальних, інформаційних систем. Пентест проводиться лише за згодою власника електронного ресурсу, підтвердженою двостороннім офіційним документом. В основі тестування на проникнення лежить принцип імітації кібер-атак. Завдання пентестера – спробувати обійти існуючий захист системи, перевірити її на цілісність, стійкість, гнучкість, вразливість, моделюючи справжню загрозу.

Причини вразливостей

- **Помилки проектування та розробки**: у дизайні апаратного та програмного забезпечення можуть бути недоліки. Ці помилки можуть поставити критично важливі для бізнесу дані під загрозу.
- **Погана конфігурація системи**: це ще одна причина вразливості. Якщо система погано налаштована, вона може мати лазівки, через які зловмисники можуть увійти в систему та викрасти інформацію.
- **Людські помилки**: людські фактори, такі як неналежне розпорядження документами, залишення документів без нагляду, помилки кодування, інсайдерські загрози, обмін паролями з фішинговими сайтами тощо можуть призвести до порушень безпеки.
- **Підключення**: якщо система підключена до незахищеної мережі (відкриті підключення), вона потрапляє в зону досяжності хакерів.
- **Складність**: вразливість системи безпеки зростає пропорційно складності системи. Чим більше можливостей має система, тим більше шансів на атаку системи.
- **Паролі**: паролі використовуються для запобігання несанкціонованому доступу. Вони повинні бути достатньо сильними, щоб ніхто не вгадав пароль. Паролями не можна ділитися з ким-небудь за будь-яку ціну (☺), і паролі слід періодично міняти. Незважаючи на ці вказівки, часом люди розкривають свої паролі іншим, десь записують їх і зберігають прості паролі, про які можна здогадатися.
- **Введення користувачем**: ви, вже чули про введення SQL, переповнення буфера тощо. Дані, отримані в електронному вигляді за допомогою цих методів, можуть бути використані для атаки на приймаючу систему.
- **Управління**: безпекою важко керувати. Іноді організації відстають у правильному управлінні ризиками, а отже вразливість в системі зростає.
- **Відсутність підготовки персоналу**: це призводить до людських помилок та виникнення інших вразливих місць.
- **Спілкування**: такі канали, як мобільні мережі, Інтернет, телефон, відкривають простір для крадіжки даних.

Методології тестування на проникнення

OWASP TESTING GUIDE (завантажити: https://kr-labs.com.ua/books/owasp-testing-guide-v4.pdf)

OWASP (Open Web Application Security Project) – це міжнародна відкрита спільнота, націлена на поліпшення безпеки програмного забезпечення та електронних ресурсів. Кожен має право брати участь в розвитку методологій OWASP, а усі їх матеріали вільно розповсюджуються. OWASP Testing Guide являє собою найбільш актуальну та пропрацьовану методологію, яка дає не тільки вказівки по тестам на проникнення, але й аналізу веб-застосунків в цілому (наприклад – вихідного коду), оскільки ця методика фокусує свою увагу саме на виявлення вразливостей веб- застосунків.

OSSTMM (завантажити: https://kr-labs.com.ua/books/OSSTMM.3.pdf)

OSSTMM (The Open Source Security Testing Methodology Manual) – добре структуризована методологія, містить так-звану «Мапу безпеки» – візуальний показник безпеки. На карті вказуються основні області безпеки, які включають в себе набори елементів, які повинні бути протестовані на відповідність методиці.

NIST (завантажити: https://kr-labs.com.ua/books/nist_800-1152008.pdf)

NIST (The United States National Institute of Standards and Technology) – методологія з кібербезпеки, розроблена Національним Інститутом Технологій США. Містить рекомендації щодо способів оцінки захищеності, внутрішнього і зовнішнього аудиту, тестування на проникнення, організації процесів безпеки, аналізу результатів, використання результатів аналізу для постійного вдосконалення СУІБ організації. Документ постійно оновлюється і дорабляється. Існує кілька версій цього документу, зокрема NIST 800-115 та NIST 800-181.

Методології тестування на проникнення

PTES (завантажити: https://kr-labs.com.ua/books/ptes-guide-pentest.pdf)

PTES (Penetration Testing Execution Standard) – дуже якісно та глибинно пропрацьований стандарт проведення тестів на проникнення. Містить цілий ряд рекомендацій, описів, вимог. Фактично у PTES викладені основні ази, принципи та основи пентесту.

ISSAF (завантажити: https://kr-labs.com.ua/books/oissg-pentest.pdf)

ISSAF (Information System Security Assessment Framework) – документ охоплює величезну кількість питань, пов'язаних з інформаційною та кібербезпекою. Присутні глави, що описують оцінку безпеки міжмережевих екранів, маршрутизаторів, антивірусних систем і багато іншого.

BSI (завантажити: https://kr-labs.com.ua/books/BSI-penetration-test-model.pdf)

BSI (Study A Penetration Testing Model) – методологія з кібербезпеки, розроблена німецьким підрозділом «Federal Office for Information Security». В документі описується проведення тестувань на проникнення та випробувань системи на міцність. Детально описуються усі необхідні вимоги, правові аспекти застосування методології та процедури, які необхідно виконати для успішного проведення пентестів.

PCI-DSS (завантажити: https://kr-labs.com.ua/books/Penetration_Testing_Guidance_March_2015.pdf)

PCI DSS (PCI Data Security Standard) – документ, що містить рекомендації і вимоги безпеки з точки зору фінансово-технічних операцій. Документ добре структурований і має обширну теоретичну базу.

Як здійснюється тестування на проникнення?

Методологія з проведення тестування на проникнення містить такі етапи:

- Планування тесту на проникнення.
- Збір публічно доступних даних про цільові системи.
- Пошук вразливостей ІС (сканування).
- Проникнення до системи (експлуатація вразливостей).
- Написання та надання звіту.
- ***Очищення системи від наслідків тесту.***

1. Планування тесту на проникнення

На цьому етапі визначаються терміни, вартість робіт, що будуть проводитись, методи, які будуть застосовані, тип та кількість ІС для тестування і форма звіту.

Існує 3 підходи до проведення тесту на проникнення:

White box. Виконавець має доступ до систем і володіє повною інформацією про неї.

Grey Box. Виконавець імітує хакерів, які мають інформацію про ІС частково (наприклад, про діапазон IP-адрес, web-сайти, фізичне розташування, ідентифікатори бездротових мереж тощо).

Black Box. Виконавець імітує хакерів, які мають лише назву компанії та практично нульові відомості про цільову систему.

2. Збір публічно доступних даних про цільові системи

До обсягу робіт, як правило, входить пошук інформації в таких джерелах:

- пошукові системи;
- соціальні мережі та сайти знайомств;
- каталоги підприємств;
- сайти новин;
- корпоративні сайти компанії замовника, сайти клієнтів і партнерів;
- сайти пошуку роботи;
- бази даних WHOIS;
- DNS сервера компанії;
- аналіз маршрутів мережевого обладнання;
- аналіз e-mail листів;
- дзвінки в call-центр компанії з метою отримати інформацію про ключових співробітників компанії, про структуру компанії і технології;
- аналіз метаінформації у документах, розміщених на сайтах компанії;
- безпосереднє сканування мережі різними інструментами для виявлення IP-адрес, портів, версій сервісів, які функціонують, і операційних систем.

Часто вже на цьому етапі можливе виявлення критичних вразливостей, як-от забуті або «безгосподарні» сервіси, що не вимагають авторизації і дають доступ до внутрішньої мережі, опубліковані конфіденційні дані, паролі та інша критична інформація.

2. Збір публічно доступних даних про цільові системи

- ❑Формування переліку цілей та завдань
- ❑Розвідка з відкритих джерел (OSINT)
- ❑Розвідка даних із соцмереж (SOCMINT)
- ❑Збір інсайдів (соцінженерія)
- ❑Пошук та аналіз витоків даних, точок входу, Google HackingSAST/DAST сканування, footprinting, енумерація
- ❑Обробка та структуризація накопичених даних
- ❑Розробка стратегії, сценаріїв, методів та алгоритмів атаки
- ❑Підбір інструментів і технічної бази, формування проєктної команди

Джерела пошуку інформації:

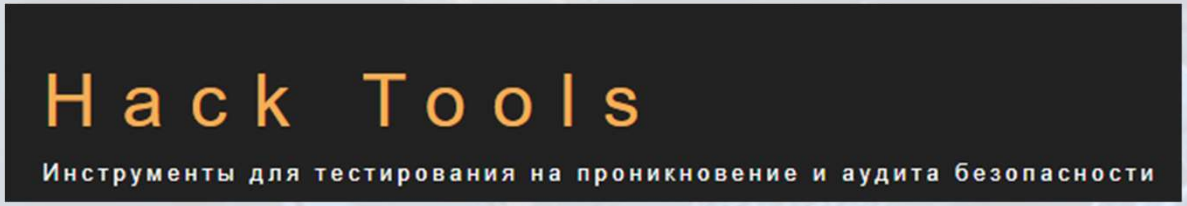
- ➢пошукові системи;
- ➢соціальні мережі та сайти знайомств;
- ➢каталоги підприємств;
- ➢сайти новин;
- ➢корпоративні сайти компанії замовника, сайти клієнтів і партнерів;
- ➢сайти пошуку роботи;
- ➢бази даних WHOIS;
- ➢DNS сервера компанії;
- ➢аналіз маршрутів мережевого обладнання;
- ➢аналіз e-mail листів;
- ➢дзвінки в call-центр компанії з метою отримати інформацію про ключових співробітників компанії, про структуру компанії і технології;
- ➢аналіз метаінформації у документах, розміщених на сайтах компанії;
- ➢безпосереднє сканування мережі різними інструментами для виявлення IP-адрес, портів, версій сервісів, які функціонують, і операційних систем.

Часто вже на цьому етапі можливе виявлення критичних вразливостей, як-от забуті або «безгосподарні» сервіси, що не вимагають авторизації і дають доступ до внутрішньої мережі, опубліковані конфіденційні дані, паролі та інша критична інформація.

3. Пошук вразливостей ІС (сканування)

Залежно від обраних систем на цьому етапі скануються вразливості різними програмами-сканерами. Спеціалізація таких сканерів може бути орієнтована на тестування периметра мережі, web-сайтів, окремих програм і сервісів: баз даних, VPN-пристроїв, пристроїв IP-телефонії тощо.

Найбільш повний та актуальний список програм та утиліт для сканування на вразливості доступний за посиланням: *https://kali.tools/all/*



4. Проникнення до системи (експлуатація вразливостей, атака)

- ❑ Експлуатація помилок і вразливостей
- ❑ Здійснення навантаження на сервер: фаззінг, парсинг, флуд
- ❑ Обхід стандартних засобів безпеки, комбінація технік
- ❑ Ескалація привілеїв
- ❑ Експлуатація доступу
- ❑ Інвентаризація мережі
- ❑ Постексплуатація і запуск експлойтів, маніпуляція даними й процесами
- ❑ Розгортання бекдорів, активне поширення та закріплення
- ❑ Досягнення поставлених цілей

Знайдені потенційні вразливості повинні бути перевірені вручну, щоб відфільтрувати всі помилкові спрацьовування. Цей етап передбачає:

- ✓ верифікацію та дослідження вразливостей;
- ✓ проведення атак на компоненти IT-інфраструктури;
- ✓ підбір паролів;
- ✓ визначення способів взаємодії застосунків;
- ✓ підтвердження виявлених вразливостей;
- ✓ збір доказів;

Для зламу вразливих IT-систем використовується різний спеціалізований інструментарій, експлойти в публічному доступі на хакерських сайтах. У деяких випадках знадобиться власна розробка вірусів і експлойтів для проникнення всередину мережі.

5. Написання та надання звіту

- ❑ Документація проведеного тестування на проникнення: збір доказів, фактів, джерел, скріншотів проникнення та компрометації системи, перелік знайдених витоків та вразливостей
- ❑ Покроковий опис усіх лацюжків атаки, використаних технік та інструментів
- ❑ Оцінка виявлених вразливостей, розрахунок рівня їх критичності, підсумковий бал захищеності системи
- ❑ Формування рекомендацій та інструкцій з оптимізації кібербезпеки, підбір оптимальних IT-рішень
- ❑ Створення фінального звіту-презентації у форматі PDF-документа

Після проведення тесту на проникнення розробляється **звіт про тестування**, який зазвичай містить:

- ✓ опис меж, у рамках яких було проведено тест на проникнення;
- ✓ методи і засоби, які використовувалися під час проведення тесту на проникнення;
- ✓ опис виявлених дефектів і недоліків, зокрема рівень їхнього ризику і можливість їхнього використання зловмисником;
- ✓ опис застосованих сценаріїв проникнення;
- ✓ опис досягнутих результатів;
- ✓ базову оцінку ризиків інформаційної безпеки Компанії;
- ✓ базову оцінку процесів забезпечення інформаційної безпеки Компанії;
- ✓ рекомендації з усунення виявлених недоліків та вдосконалення процесів забезпечення інформаційної безпеки Компанії;
- ✓ план робіт щодо усунення знайдених вразливостей і вдосконалення процесів забезпечення інформаційної безпеки Компанії, пріоритезований відповідно до критичності вразливостей.

Строки проведення тесту на проникнення

Тривалість проведення тесту на проникнення варіюється залежно від масштабу роботи і рівня захищеності об'єктів тестування.

Мінімальний термін – від 2 тижнів.

Результат (що б повинен отримати замовник)?

Основою вигодою проведення тесту на проникнення є посилення захищеності ІС, а саме:

- виявлення максимальної (?? ну, ну) кількості вразливостей;
- вжиття заходів на основі обґрунтованих рекомендацій;
- впевненість у захищеності інформації;
- виконання вимог контролюючих органів/стандартів;
- обґрунтування бюджетів підрозділу на усунення прогалин.

Ціна на пентест (ну хоч приблизно ☺)

Вартість пентесту залежить від багатьох факторів: типу досліджуваного об'єкту (інфраструктура, сервер, застосунок чи електронний ресурс), його стану та об'єму, кількості задіяних спеціалістів, тривалості виконання, переліку робіт, обраного формату/сценарію (White/Grey/BlackBox Pentest, Red Team) і методології (PCI-DSS/ISO27001/OWASP/SANS/NIST і т.д).

Пентест може бути повністю ручним, автоматичним або напівавтоматичним. Зовнішнім/внутрішнім.

Звіт може бути стандартним, розширеним, скороченим, індивідуальним.

До прикладу, середня ціна тестування на проникнення в США та Європі становить: $15 000 - $100 000. В Україні: ~ 30 000 - 100 000 грн.

Критерії вибору найкращого інструменту проникнення

- Він повинен бути простим для розгортання, налаштування та використання.
- Він повинен легко сканувати систему.
- Він повинен класифікувати вразливості на основі серйозності, яка потребує негайного виправлення.
- Він повинен мати можливість автоматизувати перевірку вразливостей.
- Він повинен повторно перевірити виявлені раніше вразливості.
- Він повинен створювати докладні звіти про вразливість та журнали.

Дізнавшись, які тести вам потрібно виконати, ви можете або створити свої тестові ресурси, або скористатися готовими, які виконуватимуть завдання проникнення за вас.

Типи тестування на проникнення

1) **Тест соціальної інженерії**: у цьому тесті намагаються змусити людину розкрити конфіденційну інформацію, таку як пароль, важливі для бізнесу дані тощо. Ці тести в основному проводяться за допомогою телефону або Інтернету і спрямовані на певні служби довідки, працівників та процеси. Помилки людини є основними причинами уразливості системи безпеки. Стандартів та політик безпеки повинні дотримуватися всі співробітники, щоб уникнути спроб проникнення в соціальну інженерію. Приклад цих стандартів включає не згадування будь-якої конфіденційної інформації в електронній пошті або телефонному спілкуванні. Аудит безпеки може проводитись для виявлення та виправлення недоліків процесу.

2) **Тест веб-застосунків**: за допомогою програмних методів можна перевірити, чи має застосунок вразливі місця безпеки. Перевіряється вразливість системи безпеки веб-програм та програм, розташованих у цільовому середовищі.

3) **Тест на фізичне проникнення**: для захисту конфіденційних даних застосовуються сильні методи фізичної безпеки. Зазвичай це використовується у військових та державних закладах. Усі фізичні мережеві пристрої та точки доступу перевірені на можливі порушення безпеки. Цей тест мало стосується сфери тестування програмного забезпечення.

4) **Тест мережевих служб**: це один із найчастіше виконуваних тестів на проникнення, де ідентифікуються проломи в мережі, за якими робиться проникнення у системи в мережі, щоб перевірити, які вразливості існують. Це можна зробити локально або віддалено.

5) **Тест на стороні клієнта**: він спрямований на пошук та використання вразливостей у програмах на стороні клієнта.

6) **Віддалений комутований номеронабирач**: він шукає модеми в оточенні та намагається увійти до систем, що підключені через ці модеми, за допомогою відгадування пароля або примусового використання.

7) **Тест безпеки бездротової мережі**: він виявляє відкриті, несанкціоновані та менш захищені точки доступу або мережі Wi-Fi і підключається через них.

Тестування на проникнення. Зразки тестових випадків (сценарії випробувань)

Пам’ятайте, що це не функціональне тестування. У Pentest ваша мета – знайти дірки в системі. Нижче наведено деякі загальні тестові приклади, які не обов’язково застосовуються до всіх програм.

1. Перевірте, чи веб-програма здатна ідентифікувати спам-атаки на контактні форми, що використовуються на веб-сайті.
2. Проксі-сервер – перевірте, чи не відстежується мережевий трафік проксі-пристроями. Проксі-сервер ускладнює отримання хакерами внутрішніх деталей мережі, захищаючи систему від зовнішніх атак.
3. Фільтри спам-електронної пошти – переконайтеся, чи вхідний та вихідний електронний трафік відфільтровано, а небажані електронні листи заблоковано.
4. Багато поштових клієнтів постачаються із вбудованими фільтрами спаму, які потрібно налаштувати відповідно до потреб. Чи можна застосувати ці правила конфігурації до заголовків електронної пошти, теми чи тексту для проникнення.
5. Брандмауер – переконайтесь, чи вся мережа або комп’ютери захищені брандмауерами. Брандмауер може бути програмним або апаратним забезпеченням для блокування несанкціонованого доступу до системи. Брандмауер може заборонити надсилання даних поза мережею без дозволу.
6. Спробуйте використати всі сервери, настільні системи, принтери та мережеві пристрої.

Тестування на проникнення. Зразки тестових випадків (сценарії випробувань)

7. Переконайтеся, що всі імена користувачів та паролі зашифровані та передані через захищені з’єднання, такі як https.
8. Перевірте інформацію, що зберігається в файли cookie веб-сайту. Він не повинен бути у читабельному форматі.
9. Перевірте знайдені раніше уразливості, щоб перевірити, чи працює виправлення.
10. Перевірте, чи немає в мережі відкритого порту.
11. Перевірте всі телефонні пристрої.
12. Перевірте безпеку мережі WIFI.
13. Перевірте всі методи HTTP. Методи PUT та Delete не повинні вмикатись на веб-сервері.
14. Переконайтеся, що пароль відповідає необхідним стандартам. Пароль повинен містити принаймні 8 символів, що містять принаймні одне число та один спеціальний символ.
15. Ім'я користувача не повинно бути таким, як «адміністратор» або «адмін».
16. Переконайтеся, чи блокується сторінка входу в програму після кількох невдалих спроб входу.
17. Повідомлення про помилки повинні бути загальними і не повинні містити конкретних деталей помилок, таких як «Недійсне ім'я користувача» або «Недійсний пароль».

Тестування на проникнення. Зразки тестових випадків (сценарії випробувань)

18. Переконайтеся, що спеціальні символи, теги HTML та сценарії обробляються належним чином як вхідне значення.

19. Внутрішні деталі системи не повинні розкриватися в жодному повідомленні про помилку чи попередженні.

20. Спеціальні повідомлення про помилки повинні відображатися кінцевим користувачам у разі аварії веб-сторінки в загальному вигляді.

21. Перевірте використання записів реєстру. Конфіденційну інформацію не слід зберігати в реєстрі.

22. Всі файли повинні бути просканові перед завантаженням їх на сервер.

23. Конфіденційні дані не повинні передаватися в URL-адресах під час спілкування з різними внутрішніми модулями веб-програми.

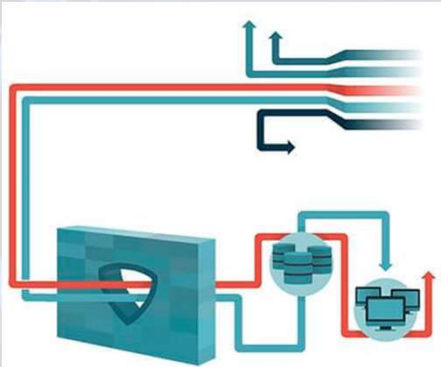
24. У системі не повинно бути жодного незакодованого імені користувача чи пароля.

25. Перевірте всі поля введення з довгим рядком введення з пробілами та без них.

26. Переконайтеся, що функція скидання пароля захищена.

27. Перевірте на SQL-ін’єкцію.

28. Перевірте на міжсайтові сценарії.



Тестування на проникнення. Зразки тестових випадків (сценарії випробувань)

29. Важливі перевірки вводу слід робити на стороні сервера, а не перевіряти JavaScript на стороні клієнта.

30. Критично важливі ресурси в системі повинні бути доступними лише уповноваженим особам та службам.

31. Усі журнали доступу повинні підтримуватися з належними дозволами доступу.

32. Переконайтеся, що сеанс користувача закінчується після виходу з системи.

33. Переконайтеся, що перегляд каталогів на сервері вимкнено.

34. Переконайтеся, що всі програми та версії баз даних оновлені.

35. Перевірте маніпуляцію з URL-адресами, щоб перевірити, чи веб-програма не відображає небажаної інформації.

36. Перевірте витік пам'яті та переповнення буфера.

37. Перевірте, чи сканується вхідний мережевий трафік, щоб знайти атаки троянських програм.

38. Переконайтеся, що система захищена від Brute Force Attacks.

Тестування на проникнення. Зразки тестових випадків (сценарії випробувань)

39. Переконайтеся, що система або мережа захищені від атак DoS (відмова в обслуговуванні). Хакер може націлити мережу або окремий комп'ютер із постійними запитами, через що ресурси цільової системи перевантажуються, що призводить до відмови в обслуговуванні законних запитів.

40. Перевірте на атаки введення сценарію HTML.

41. Перевірте наявність атак COM та ActiveX.

42. Перевірте на атаки підробки. Атаки підробки можуть бути різних типів – підробка IP-адреси, підробка ідентифікатора електронної пошти, підробка ARP, підробка рефералів, підробка ідентифікатора абонента, отруєння мереж обміну файлами, спуфінг GPS.

43. Перевірте неконтрольовану атаку рядкового формату – атаку безпеки, яка може спричинити збій програми або виконати шкідливий сценарій на ній.

44. Перевірка атаки ін’єкції XML – використовується для зміни передбачуваної логіки програми.

45. Перевірте наявність атак канонізації.

46. Переконайтеся, що сторінки помилок не відображають будь-яку інформацію, яка може бути корисною для входу хакера в систему.

Тестування на проникнення. Зразки тестових випадків (сценарії випробувань)

47. Переконайтеся, що важливі дані, такі як пароль, зберігаються в секретних файлах системи.
48. Перевірте, чи застосунок не повертає більше даних, ніж потрібно.

Це лише основні тестові сценарії для початку роботи з Pentest. Існують сотні вдосконалених методів проникнення, які можна зробити як вручну, так і за допомогою засобів автоматизації.

Тестування на проникнення. Методологія OWASP Testing Guide v.4

Розвідка та збір даних

- Розвідка: Збір даних про IP/домени
- Збір інформації через пошукові системи
- Збір банерів/відбитків серверів та веб-серверів
- Дослідження мета-файлів на можливість витоку даних
- Спроба визначення всіх програм на веб-сервері
- Визначення потенційних вхідних точок
- Збір відбитків веб програми та створення картки програми

Тестування управління конфігурацією та розгортанням

- Тестування конфігурації сервера/веб-сервера
- Тестування конфігурації платформи програми
- Тестування обробки різних розширень файлів
- Пошук старих, тимчасових або резервних копій файлів
- Пошук/підбір адміністративних інтерфейсів
- Тестування методів HTTP
- Тестування HSTS
- Перевірка міждоменної політики
- Тестування дозволів на файли/папки

OWASP

Open Web Application Security Project

Тестування на проникнення. Методологія OWASP Testing Guide v.4

Тестування механізмів визначення користувачів

- Визначення ролей
- Тестування процесу реєстрації користувачів
- Тестування створення облікового запису
- Тестування методів підбору/вгадування облікових записів

Тестування аутентифікації

- Тестування даних, що передаються по HTTPS
- Тестування облікових даних за замовчуванням
- Тестування механізмів блокування
- Тестування схем обходу аутентифікації
- Тестування функції запам'ятовування пароля
- Тестування кеша браузера на можливі недоліки
- Тестування політики на слабкі паролі
- Тестування на проблеми в процесі зміни або скидання пароля
- Тестування альтернативних способів автентифікації на слабкі місця

Тестування на проникнення. Методологія OWASP Testing Guide v.4

Тестування авторизації

- Спроби обходу каталогів/підключення файлів
- Спроби обходу авторизації
- Спроби підвищення привілеїв
- Тестування на можливість прямого звернення до об'єктів
- Тестування управління сесією користувачів
- Тестування можливостей обходу схеми керування сеансом користувача
- Перевірка атрибутів Cookie
- Тестування можливої фіксації сесії користувача
- Перевірка для відображених змінних сеансу користувача
- Тестування на CSRF
- Тестування механізму виходу із системи
- Перевірка тайм-аут активних сеансів користувачів

Тестування полів у формах

- Тестування на XSS (збережені, відбиті, DOM)
- Тестування перехоплення HTTP запитів
- Тестування на заміну параметрів HTTP запитів
- Перевірка на SQL ін'єкції з різними типами баз даних
- Тестування на ін'єкції типу: LDAP, ORM, XML, SSI, Xpath
- Перевірка ін'єкцій з IMAP/SMTP
- Тестування можливості впровадження коду
- Тестування на можливість підключення файлів локально/віддалено
- Перевірка можливості відправлення та виконання команд
- Перевірка переповнення буфера
- Перевірка на можливість розщеплення запиту
- Тестування на вхідні запити HTTP

Тестування на проникнення. Методологія OWASP Testing Guide v.4

Тестування обробки помилок

- Аналіз кодів помилок
- Трасування

Тестування шифрування

- Тестування на слабкі алгоритми шифрування, перевірка на старі версії протоколів шифрування
- Перевірка на POODLE тип вразливості
- Тестування на чутливу інформацію, що надсилається незашифрованим протоколом
- Тестування на слабке шифрування

Тестування бізнес-логіки

- Перевірка бізнес-логіки на валідацію даних
- Перевірка можливості надсилання неіснуючих типів запитів
- Тестування перевірки цілісності
- Тестування часу виконання запиту
- Перевірка обмежень, пов'язаних з використанням функцій
- Тестування захисту від нецільового використання програми/функціоналу
- Перевірка можливості завантаження непідтримуваних файлів
- Тестування можливості завантаження шкідливих файлів

Тестування на стороні клієнта

- Перевірка на DOM-based XSS
- Перевірка на виконання JavaScript
- Перевірка на HTML ін'єкції
- Перевірка на можливість перенаправлення
- Перевірка на CSS ін'єкції
- Перевірка на можливість маніпуляції ресурсами клієнта
- Перевірка CORS
- Перевірка на Clickjacking
- Перевірка роботи програми із сокетами
- Перевірка локального сховища

Дякую за увагу. Питання?