Безпека операційних систем і комп'ютерних мереж

Лекція 2

Модель загроз для операційної системи

Етапи побудови засобу захисту

1. Формування вимог
2. Розроблення політики безпеки, моделі безпеки
3. Розроблення ТЗ
4. Проектування
5. Реалізація
6. Тестування (сертифікація)
7. Впровадження
8. Супроводження

Модель загроз для операційної системи

- Сканування файлової системи
- Викрадення ключової інформації
- Підбирання паролів
- Збирання сміття
- Перевищення повноважень
- Програмні закладки
- Жадібні програми

Сканування файлової системи

- Ця загроза полягає в тому, що порушник переглядає (сканує) всю файлову систему, і намагається прочитати усі файли поспіль
  - Замість зчитування порушник може намагатись скопіювати або видалити файли
  - Якщо він не може отримати доступ до деякого об’єкта файлової системи (окремого файлу або каталогу), то він пропускає захищений об’єкт і продовжує сканування
  - Порушник може здійснювати цю атаку, використовуючи спеціальне програмне забезпечення
- Сканування файлової системи – це в першу чергу атака на політику безпеки
  - Будь-який користувач в системі повинен отримати доступ лише до тих файлів, до яких він повинен мати доступ згідно політики безпеки
  - Якщо порушник в результаті сканування отримує доступ до інших файлів, він порушує політику безпеки, тобто здійснює НСД
  - При цьому порушник отримує можливість несанкціоновано ознайомитись з інформацією (порушення конфіденційності), або несанкціоновано модифікувати або видалити інформацію (порушення цілісності).

Сканування файлової системи

- В подальшому ми взагалі не будемо розглядати ті ОС, які не здійснюють контроль доступу до файлів
  - Але й ті ОС, що такий контроль (керування, аудит) здійснюють, можуть бути атакованими
- Можливість НСД до об’єктів файлової системи визначається наявністю її вбудованої *системи розмежування доступу* і коректністю її адміністрування
  - Оскільки звичайна файлова система містить величезну кількість об’єктів, помилки в настроюванні виникають з великою ймовірністю
- Джерелом атаки може бути будь-який легальний користувач системи
  - Цю атаку може здійснювати анонімний користувач (гість), якщо можливість анонімного входу в систему не заблокована

Викрадення ключової інформації

- На рівні ОС використовується різна інформація, яку можна віднести до ключової. Найпоширеніший вид такої інформації – паролі доступу до системи
- Типові способи викрадення паролів:
  - викрадення пароля, що записаний на папері
  - підглядання пароля в момент, коли користувач його набирає
    - достатньо всього кілька тижнів тренувань для того, щоби фіксувати пароль за рухами рук користувача по клавіатурі
  - підглядання пароля на екрані
    - усі сучасні ОС приховують і не демонструють пароль, що вводиться
    - пароль на екрані видно, якщо його помилково вводять у інше поле введення
    - пароль видно, коли його вводять як аргумент в командному рядку
  - зчитування пароля у командних файлах (сценаріях)
    - користувачі можуть створювати командні файли для автоматизації входу на віддалені мережні ресурси
  - отримання паролів, що ненадійно зберігаються
    - іноді паролі ховаються у деякому файлі у відкритому вигляді
    - часто паролі ненадійно кодують (наприклад, за допомогою операції **XOR**
  - якщо використовуються зовнішні носії ключової інформації (дискети, пристрої флеш-пам’яті, токени), то ці носії можуть бути просто викрадені.

Підбирання паролів

- Підбирання паролів передбачає використання засобів автентифікації для знаходження того пароля, який буде прийнятий як правильний
  - Порушник отримує несанкціонований доступ до певного ресурсу, якщо пароль використовується для обмеження доступу до ресурсу
  - або можливість працювати в системі від імені іншого користувача, якщо пароль використовується для автентифікації
- Будь-яка система передбачає можливість того, що користувач під час введення пароля зробив помилку, тому введення пароля можна повторити
  - Раніше кількість таких повторів не обмежувалась, а швидкість їх введення визначалась лише можливостями користувача
    - Це надавало можливість порушникам методично перебирати ймовірні паролі
  - Ще більшу небезпеку несла в собі можливість застосування спеціальних програмних засобів, які здійснюють спроби автентифікації
    - Програмні засоби здатні тривалий час без утоми повторювати спроби автентифікації,
    - Вони можуть робити такі спроби із значно більшою швидкістю, ніж користувач-людина
    - Також вони можуть використовувати словники, відсортовані з урахуванням частоти використання тих чи інших паролів.

Підбирання паролів

- В усіх сучасних системах автентифікації вживаються заходи, які достатньо ефективно нейтралізують цю загрозу
- Основні такі заходи:
  - обмеження швидкості введення паролів шляхом введення штучних затримок в процедуру їх перевірки
  - обмеження кількості спроб введення паролів (як правило до 3...5), після чого система автентифікації на певний час блокує подальші спроби введення паролів з тієї консолі
  - реєстрація спроб автентифікації
  - негайне повідомлення адміністратора про повторні невдалі спроби автентифікації, тощо.
- В сучасних умовах крадіжка ключової інформації та підбирання паролів дуже часто реалізуються у комплексі
  - Як правило, паролі зберігаються у закодованому вигляді. Найкращий спосіб такого кодування – це обчислення хеш-функції
  - Якщо порушник викраде файл, що містить хеш-функції паролів, це не дасть йому змоги безпосередньо використати отриману інформацію для проходження автентифікації
  - Якщо хеш-функція є стійкою, то єдине, що порушник зможе зробити – це спробувати підібрати пароль, хеш-функція якого співпаде з відомою йому хеш-функцією
  - Таке підбирання, на відміну від безпосередніх спроб входу в систему, можна робити без обмежень на кількість спроб і на швидкість, використовуючи будь-яку наявну у порушника обчислювальну техніку.

Збирання сміття

- Збирання сміття – це отримання даних, які залишаються у об’єктах, що звільняються ОС після їх використання
  - Найтиповішими з таких об’єктів є файли на дисках і ділянки оперативної пам’яті
- У більшості файлових систем файли після їх видалення не знищуються фізично, а лише помічаються як знищені
  - Сектори диску, які були відведені для цього файлу, вважаються вільними, і на них можуть бути записані частини інших файлів. Але до нового запису вони продовжують містити дані, які знаходились у видаленому файлі
  - Здійснюючи прямий доступ до секторів диску, можна прочитати цю інформацію
    - Саме такий принцип використовують програми відновлення помилково видалених файлів
    - Збирання сміття можливе навіть у серверних файлових системах, де відновлення видалених файлів вкрай малоймовірне (і тому серверні ОС не мають утиліт відновлення помилково видалених файлів)
    - Наприклад, у багатозадачній серверній ОС може бути запущений окремий процес, який постійно переглядає вміст секторів диску, що звільняються

Збирання сміття — тимчасові файли

- Простий спосіб збирання сміття – копіювання і вивчення тимчасових файлів
  - Практично усі програми для роботи створюють тимчасові файли, які вилучаються по завершенні роботи програми або після закриття робочого файлу
  - Досить часто з різних причин ці файли не вилучаються
    - наприклад,добре відомо, як розростаються каталоги “Temp” у системі Windows
  - Такі файли можуть бути дуже корисними для порушників
- Велику цінність має вміст “файлу підкачування”, який створюється для реалізації механізму віртуальної пам’яті
- Деякі дуже інформативні для порушників файли можуть виникати під час збоїв
  - наприклад, дамп ядра системи (записаний на диск у вигляді файлу вміст системних областей оперативної пам’яті)
- У більшості сучасних систем для мінімізації наслідків помилкового вилучення файли, які вилучають, копіюють у спеціальне сховище – “кошик для сміття” (trash bin), звідки їх можна повністю відновити

Збирання сміття в оперативній пам'яті

- Стосовно оперативної пам’яті комп’ютера можна зазначити, що звичайні ОС не утрудняють собі життя її очищенням
  - Оскільки очищення пам'яті (заповнення її нулями) вимагає певних ресурсів, а користувачеві результатів цієї операції не видно, для більшості звичайних користувачів виконання такого очищення виглядало б як неефективна робота (зниження швидкодії) системи
- Коли прикладна програма запитує додаткову пам’ять, їй виділяють ділянки пам’яті, що продовжують зберігати дані попередніх програм, які ці ділянки використовували
  - Нормальні прикладні програми починають використання пам'яті з її ініціалізації
- Можна (і неважко) написати програму, яка буде захоплювати пам’ять і копіювати із щойно захопленої пам’яті цікаві дані
  - наприклад, здійснюючи пошук за ключовими словами

Перевищення повноважень

- Ця загроза полягає у тому, що порушник якимось чином отримує повноваження, що перевищують ті, які йому надані згідно політики безпеки
  - Перевищення повноважень можливе або через помилки в розробці й реалізації політики безпеки (наприклад, невірні настроювання системи розмежування доступу)
  - або шляхом використання наявних вразливостей в програмному забезпеченні, яке входить до складу ОС

Перевищення повноважень

- Наприклад, категорії користувачів в Unix-подібних ОС:
  - *Адміністратор*
    - У більшості класичних Unix-подібних ОС він має властивості суперкористувача, тобто повні і необмежені права в системі
  - *Спеціальний користувач*
    - Це “фіктивні” користувачі, від імені яких виконуються деякі системні процеси
    - Такі користувачі можуть мати значні повноваження стосовно окремих захищених об’єктів системи, а повноваження стосовно інших об’єктів – навпаки, менші за звичайних користувачів
    - Як правило, облікові записи спеціальних користувачів не можуть використовуватись для здійснення інтерактивного входу в систему.
  - *Звичайний користувач*
  - *Псевдокористувач*
    - До цієї категорії належать користувачі, які не реєструються в системі, але виконують в ній певні дії шляхом взаємодії з системними процесами – *демонами* (англ. – *daemon*), як правило, через мережу. Самі демони працюють в системі з великими повноваженнями, і безпека взаємодії з псевдокористувачами цілком залежить від коректності їхнього програмного коду
- Отже, перевищення повноважень реалізується при будь-якому несанкціонованому переході користувача з нижчої категорії до вищої. Типовими загрозами є перехід:
  - з категорії 3 до 1 (звичайний користувач отримав права адміністратора),
  - з 4 до 3 (псевдокористувач отримав можливість інтерактивно працювати в системі з правами користувача)
  - і з 4 до 1 (те ж саме, але з правами адміністратора)

Програмні закладки

- **До програмних закладок відносять програми або окремі модулі програм, які протягом тривалого часу функціонують в комп'ютерній системі, здійснюючи заходи щодо приховування свого існування від користувача**
- **Програмні закладки можуть впроваджуватись вірусом, "троянським конем", мережним хробаком або безпосередньо користувачем-зловмисником**
- **Функції програмних закладок:**
  - **перехоплення і передавання інформації:**
    - **крадіжка паролів;**
    - **шпигунські програми (*Spyware*);**
  - **порушення функціонування систем ("логічні бомби"):**
    - **знищення інформації;**
    - **зловмисна модифікація інформації;**
    - **блокування системи;**
  - **модифікація програмного забезпечення:**
    - **утиліти віддаленого адміністрування (люки);**
    - **Інтернет-клікери;**
    - **проксі-сервера;**
    - **дзвінки на платні ресурси;**
    - **організація DoS і DdoS атак;**
  - **психологічний тиск на користувача:**
    - **реклама (*Adware*);**
    - **злі жарти і містифікації.**

Жадібні програми

- Жадібні програми – це шкідливі програми, які захоплюють значну частину ресурсів комп’ютера, внаслідок чого робота інших користувачів та/або процесів помітно утруднюється або взагалі стає неможливою. Часто жадібні програми можуть призводити до краху ОС
- Здебільшого, жадібні програми належать до класу “троянських коней”
- Жадібні програми можуть бути Web-застосунками, які запускаються при заході браузером на певні Web-сторінки