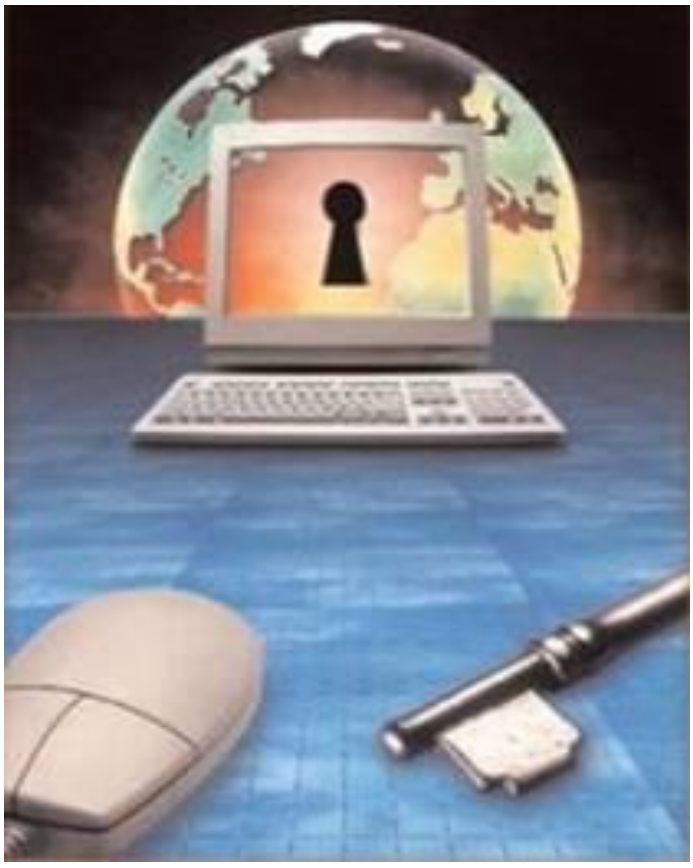
В.А. Каплун, В.П. Майданюк

ЗАХИСТ ОПЕРАЦІЙНИХ СИСТЕМ



Міністерство освіти і науки України
Вінницький національний технічний університет

В.А. Каплун, В.П. Майданюк

ЗАХИСТ ОПЕРАЦІЙНИХ СИСТЕМ

Затверджено Вченою радою Вінницького національного технічного університету як навчальний посібник для студентів напрямів підготовки 1601 – “Інформаційна безпека” та 0804 – “Комп’ютерні науки”. Протокол № від листопада 2006 року

Вінниця ВНТУ 2006

УДК 681.3.07
Д 81

Р е ц е н з е н т и :

О.Д. Азаров, доктор технічних наук, професор
Л.І. Тимченко, доктор технічних наук, професор
А.Б. Ракітянська, кандидат технічних наук, доцент

Рекомендовано до видання Вченою радою Вінницького національного технічного університету Міністерства освіти і науки України

Каплун В.А., Майданюк В.П.
Д 81 **Захист операційних систем.** Навчальний посібник. – Вінниця: ВНТУ, 2006. – 180 с.

У навчальному посібнику наведено теоретичні відомості щодо сучасних систем захисту комп'ютерних систем взагалі, особливості захисту конкретних операційних систем та їх ресурсів. Наведено характеристику можливих атак на операційні системи, види та методи захисту від них, можливості та види ідентифікації та аутентифікації користувачів. Сформовано рекомендації щодо адміністрування операційних систем з метою їх ефективного захисту від несанкціонованого проникнення. Розглянуто сучасні програми для захисту операційних систем від шкідливого програмного забезпечення, від шпигунських програм та інші.

Навчальний посібник призначений для студентів напрямів підготовки 1601 – «Інформаційна безпека» для вивчення дисципліни «Захист програмного забезпечення» та 0804 – "Комп'ютерні науки" для вивчення певних розділів дисципліни "Операційні системи".

УДК 681.3.07

ЗМІСТ

ВСТУП **7**

1 ПРОБЛЕМА ЗАХИСТУ ОПЕРАЦІЙНИХ СИСТЕМ **8**

1.1 Основні вимоги до безпеки комп'ютерних систем 8

1.2 Класифікація загроз безпеці комп'ютерних систем 8

1.2.1 Загроза за метою реалізації 8

1.2.2 Загроза за об'єктом атаки 11

1.2.3 Загроза за принципом впливу на КС 12

1.2.4 Загроза за характером впливу на КС та його об'єкти 13

1.2.5 Загроза через появи помилок захисту 14

1.2.6 Загроза за способом впливу на КС 15

1.2.7 Загроза за засобами атаки 15

1.2.8 Загроза за станом об'єкта атаки 16

1.3 Рівні захисту операційних систем 16

1.4 Об'єкти захисту в операційних системах 17

1.4.1 Захист пам'яті 17

1.4.2 Контроль доступу як захист даних та програм 18

1.6 Категорії зламщиків 20

1.7 Атаки на рівні операційних систем 20

Контрольні питання 22

2 ХАРАКТЕРИСТИКА НАЙПОШИРЕНІШИХ ЗАГРОЗ БЕЗПЕЦІ КОМП'ЮТЕРНИХ СИСТЕМ **23**

2.1 Несанкціонований доступ 23

2.2 Незаконне використання привілеїв 24

2.3 Атаки "салямі" (salami attack) 24

2.4 Приховані канали 25

2.5 Атаки типу "маскарад" 26

2.6 "Збір сміття" 27

2.7 "Зламування системи" 27

2.8 Шкідливе програмне забезпечення 28

2.8.1 Люки 29

2.8.2 Логічні бомби 30

2.8.3 Троянські коні (Trojan Horse) 31

2.8.4 Віруси 33

2.8.5 Черв'яки 34

2.8.6 Програми-зомбі 35

2.8.7 "Жадібні" програми (greedy program) 35

2.8.8 Захоплювачі паролів (password grabber) 36

2.8.9 Утиліти схованого адміністрування (backdoor) 37

2.8.10 Intended-віруси 37

2.8.11 Конструктори вирусів 38

2.8.12 Поліморфні генератори 38
2.9 Програмні закладки і їх руйнуючий вплив 39
2.9.1 Поняття програмних закладок 39
2.9.2 Моделі поведінки програмних закладок 40
2.9.3 Основні принципи роботи закладок та їх класифікація 41
Контрольні питання 43
3 ВІРУСИ ЯК ШКІДЛИВЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ 44
3.1 Класифікація комп'ютерних вірусів 44
3.1.1 Віруси за середовищем їх існування 44
3.1.2 Віруси за способом зараження 45
3.1.3 Віруси за особливостями використовуваних алгоритмів 45
3.1.4 Віруси за деструктивними можливостями 45
3.2 Систематизація комп'ютерних вірусів 46
3.2.1 Класифікаційний код вірусу 46
3.2.2 Дескриптор вірусу 47
3.2.3 Сигнатура вірусу 48
3.3 Файлові віруси 50
3.3.1 Класифікаційний код файлового вірусу 50
3.3.2 Дескриптор файлового вірусу 53
3.3.3 Сигнатура файлового вірусу 53
3.3.4 Види файлових вірусів 53
3.3.5 Алгоритм роботи файлового вірусу 60
3.3.6 Приклади файлових вірусів 61
3.4 Завантажувальні (бутові) віруси 63
3.4.1 Класифікаційний код завантажувального вірусу 63
3.4.2 Дескриптор завантажувального вірусу 64
3.4.3 Сигнатура бутового вірусу 64
3.4.4 Принцип дії завантажувальних вірусів 65
3.4.5 Розташування завантажувального вірусу 67
3.4.6 Алгоритм роботи завантажувального вірусу 68
3.4.7 Приклади завантажувальних вірусів 69
3.5 Макро-віруси 71
3.5.1 Причини зараження макро-вірусами 71
3.5.2 Загальні відомості про віруси в MS Office 72
3.5.3 Принципи роботиWord/Excel/Office97-вірусів 73
3.5.6 Приклади макро-вірусів 77
3.6 Мережеві віруси 78
3.6.1 IRC-черв'яки 78
3.6.2 IRC-клієнти 79
3.6.3 Скрипт-черв'яки 79
3.6.4 Приклади мережевих вірусів 80
3.7 Стелс-віруси 82
3.7.1 Завантажувальні стелс-віруси 82

3.7.2 Файлові стелс-віруси 82
3.7.3 Макро-стелс-віруси 83
3.7.4 Приклади стелс-вірусів 83
3.8 Поліморфік-віруси 85
3.8.1 Поліморфні розшифровувачі 86
3.8.2 Рівні поліморфізму 87
3.8.3 Зміна виконуваного коду 88
3.8.4 Приклади поліморфік-вірусів 89
3.9 Способи захисту від вірусів 92
3.9.1 Систематичне архівування важливої інформації 92
3.9.2 Обмеження ненадійних контактів 92
3.9.3 Використання антивірусних програм 93
3.9.4 Види антивірусних програм 95
3.9.5 Огляд найпоширеніших антивірусних програм 99
Контрольні питання 104
4 НАЙПРОСТІШІ МЕТОДИ ЗАХИСТУ ІНФОРМАЦІЇ В ОПЕРАЦІЙНИХ СИСТЕМАХ **106**
4.1 Установки і налаштування системи захисту 106
4.2 Блокування доступу до комп'ютера за допомогою екранної заставки Windows 110
4.3 Використання пароля BIOS 111
4.4 Програмні продукти для найпростішого захисту 114
4.5 Відновлення інформації після сбоїв 118
4.6 Робота з реєстром Windows 120
4.6.1 Призначення і структура реєстра 120
4.6.2 Зберігання реєстру 123
4.6.3 Відновлення реєстру 124
4.6.4 Утиліти для роботи з реєстром 126
Контрольні питання 127
5 РЕКОМЕНДАЦІЇ ЩОДО ПІДВИЩЕННЯ МІР БЕЗПЕКИ ОС WINDOWS XP **128**
5.1 Додаткові обмеження на паролі 128
5.2 Додаткові міри безпеки в мережі 129
5.3 Приховування слідів роботи за комп'ютером 131
5.4 Інші міри посилення безпеки 133
Контрольні питання 143
6 ПАКУВАННЯ, АРХІВАЦІЯ І ШИФРУВАННЯ ДАНИХ В ОПЕРАЦІЙНИХ СИСТЕМАХ **144**
6.1 Історичні відомості 144
6.2 Стискання файлів під Windows 9x\NT 144
6.2.1 Стискання виконуваних файлів 145
6.2.2 Стискання динамічних бібліотек 146

6.3 Продуктивність пакування файлів 146
6.4 Принципи роботи програм-архіваторів 148
6.5 Програми архівації файлів 149
6.5.1 ZIP 150
6.5.2 ARJ 151
6.5.3 RAR 151
6.6 Шифрування файлів у програмах Microsoft 152
6.7 Encrypted File System 154
Контрольні питання 155
7 ПРОБЛЕМИ ІДЕНТИФІКАЦІЇ ТА АУТЕНТИФІКАЦІЇ КОРИСТУВАЧА **156**
7.1 Поняття ідентифікації, аутентифікації і авторизації 156
7.2 Парольний захист операційних систем 157
7.2.1 Функціональне призначення механізмів парольного захисту 157
7.2.2 Реалізація механізмів парольного захисту 158
7.2.3 Загрози подолання парольного захисту 159
7.2.4 Основні вимоги до паролів 161
7.2.5 Парольні зламщики і методи їх роботи 162
7.2.6 Парольний захист ОС UNIX та можливості його зламу 163
7.2.7 Парольний захист Windows 95/98 і його ненадійність 164
7.2.6 Парольний захист в Windows NT 167
7.3 Біометричні методи захисту інформації 170
7.4 Ідентифікація за клавіатурним почерком 172
7.4.1 Графологічні можливості комп'ютера 172
7.4.2 Використання унікальності клавіатурного почерку 173
7.4.3 Режим настроювання 174
7.4.4 Режим ідентифікації 175
7.5 Інші способи ідентифікації 176
7.6 Клавіатурні шпигуни 176
7.7 Боротьба зі spyware в Windows 179
Контрольні питання 182
ЛІТЕРАТУРА **183**

ВСТУП

Останні досягнення людської думки в області комп‘ютерних технологій пов'язані з появою не тільки персональних комп‘ютерів, мереж передавання даних і електронних грошей, але і таких понять, як хакер, інформаційна зброя, комп'ютерні віруси тощо.

Використання будь-якої операційної системи – це основа роботи з комп’ютером. Загальновідомо, що кожна з операційних систем сімейств Windows та Unix має певні, лише їй притаманні, проблеми у відношенні надійності та стійкості. Через це при роботі з комп’ютером можуть виникати деякі незручності та проблеми. Навіть у більш-менш досвідчених користувачів система часом дає збої, робота її уповільнюється, її треба перевстановлювати, шукати збійні програми, нові драйвери тощо. Крім того, сучасний програмний ринок насичений шкідливими програмами, серед яких віруси, троянські програми, шпигуни та інші. На сьогоднішній день існує безліч програм, що дозволяють будь-якому користувачу здійснити злам слабо захищеної системи. Отже, правильна побудова ефективної системи захисту комп’ютерної системи має величезне значення.

Повсюдне поширення електронних технологій збереження, обробки і передачі інформації має, як виявилося, не тільки позитивні сторони, але і породжує визначені проблеми. У сучасній юридичній мові вже затвердилося таке поняття як комп'ютерний злочин, тобто злочин, здійснений з використанням обчислювальної техніки. У зв'язку з цим проблеми ідентифікації й аутентифікації користувачів комп'ютерної техніки в даний час також стали дуже актуальними.

Для того, щоб грамотно побудувати захист сінформації у комп’ютерній системі, підібрати ефективні технічні, програмні, організаційні засоби для цього є надзвичайно важливою проблемою.

1 ПРОБЛЕМА ЗАХИСТУ ОПЕРАЦІЙНИХ СИСТЕМ

1.1 Основні вимоги до безпеки комп'ютерних систем

Для розуміння того, якого роду загрозам можуть підлягати комп'ютерні системи, визначимо вимоги до безпеки. Зазвичай висувають такі основні чотири вимоги.

Конфіденційність. Згідно цій вимозі, інформацію від комп'ютерних систем (КС) можуть отримувати тільки авторизовані особи. Це включає в себе виведення на друк або на екран та інші форми подання інформації, в тому числі і саме виявлення існування об'єкту.

Цілісність. Передбачається, що характеристики КС можуть змінювати лише авторизовані особи. Під змінами тут маються на увазі запис, редагування, зміна статусу, видалення і створення нових об'єктів.

Доступність. Необхідно, щоб характеристики КС були доступними авторизованим особам.

Аутентичність. Комп'ютерна система повинна мати можливість перевіряти ідентичність користувача.

Сучасні комп'ютерні системи – це складний механізм, що складається з великої кількості компонентів різного ступеня автономності, які пов'язані між собою, і даних, якими вони обмінюються. Практично кожний механізм може вийти з ладу або піддатися зовнішньому впливу.

Загроза безпеці - потенційно можливий вплив на КС, який може прямо або побічно завдати шкоди користувачам або власникам КС.

Атака - реалізація загрози.

1.2 Класифікація загроз безпеці комп'ютерних систем

Загрози безпеці КС, враховуючи тільки навмисні загрози, можна класифікувати за ознаками, які наведені на рис.1.

1.2.1 Загроза за метою реалізації

Розглянемо роботу КС в процесі надання інформації.

Взагалі, інформація якимось чином надходить від джерела (наприклад, від файла, від області основної пам'яті тощо) до одержувача (наприклад, до іншого файла або до користувача) (рис.2,а). В залежності від того, як ця інформація надається, розрізняють чотири категорії атак.

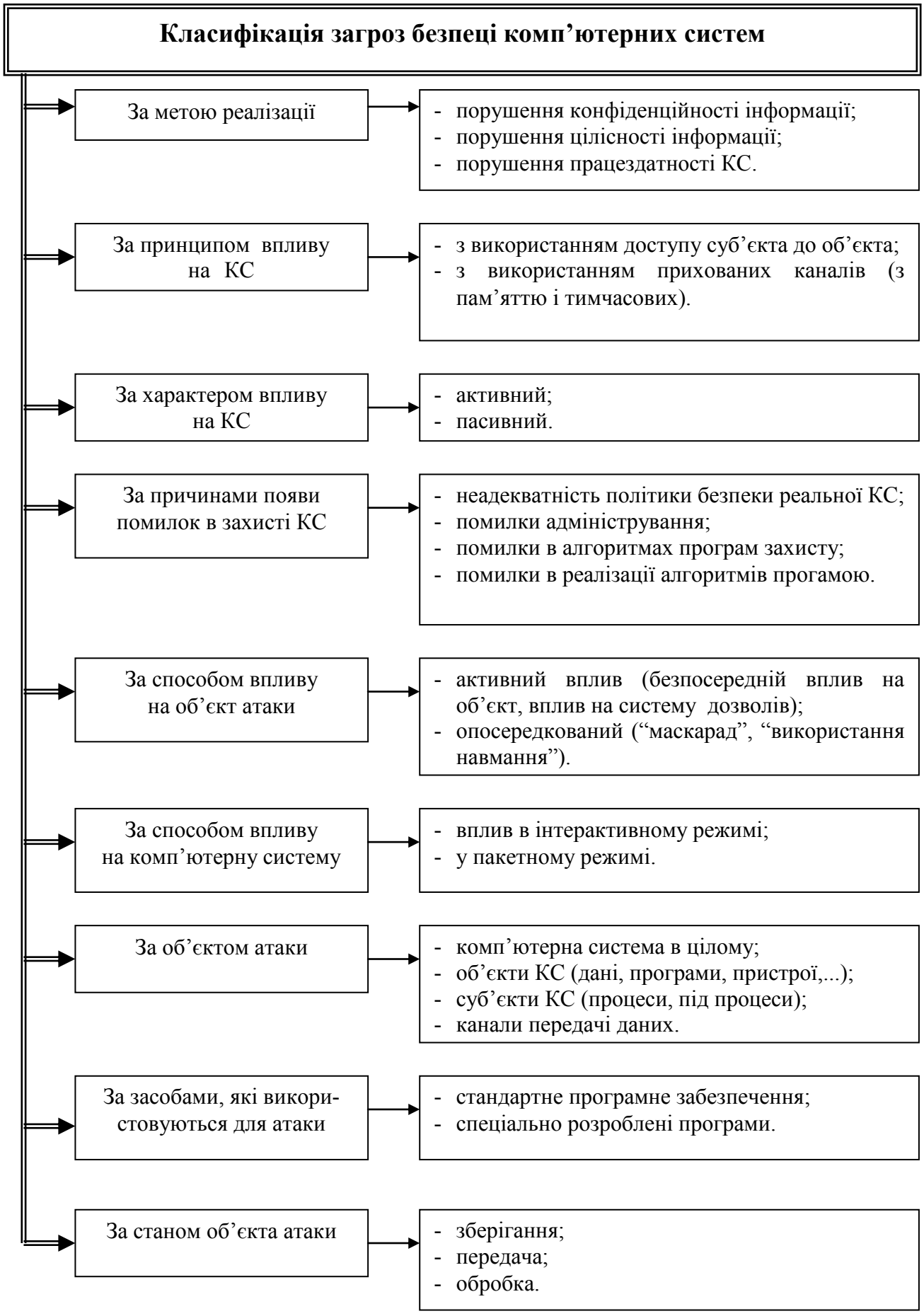
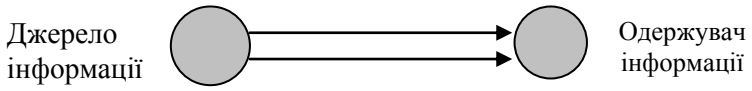
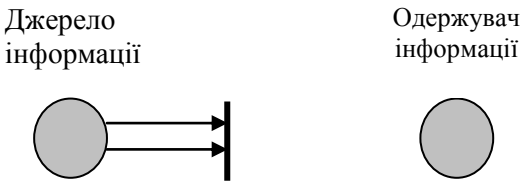


Рисунок 1 – Класифікація загроз безпеці комп'ютерних систем

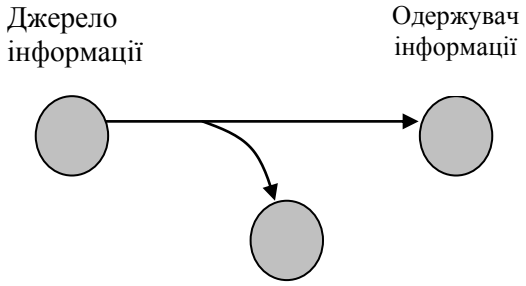
Переривання, порушення працездатності КС (часткове або повне), тобто виведення з ладу або некоректна зміна режимів роботи КС чи заміна, в результаті якої отримуються невірні результати, в наслідок чого КС не може правильно обробляти інформацію. В результаті здійснюється відмова від потоку інформації, тобто одна з взаємодіючих сторін не визнає факт передачі або прийому повідомлень, замовлень, фінансових узгоджень, донесень. Компоненти системи виходять з ладу, стають недоступними або непридатними (рис.2,б). Метою цієї атаки є *порушення доступності***.** Щодо даних, то інформація, яка зберігається і обробляється на комп'ютері, може мати велику цінність для її власника, і її використання іншими особами наносить значну шкоду власнику.



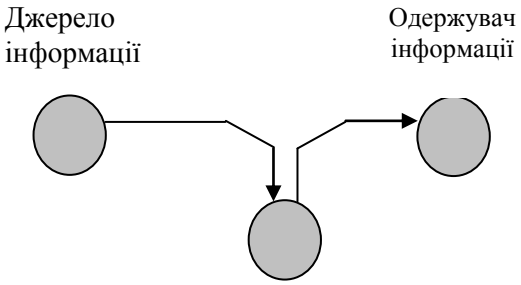
а) *Нормальна передача інформації*



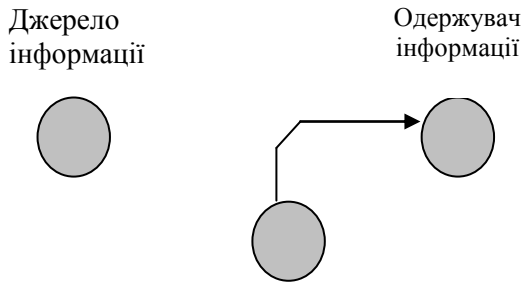
б) Переривання (порушення доступності)



в) Перехоплення (порушення конфіденційності)



г) Зміна (порушення цілісності)



д) Підробка (порушення автентичності)

Рисунок 2 – Загрози безпеці КС за метою реалізації

Перехоплення. Це атака, метою якої є *порушення конфіденційності*, в результаті чого доступ до компонентів системи отримують несанкціоновані сторони (рис.2,в). В ролі несанкціонованої сторони може бути особа, програма або комп'ютер. Прикладом цього можуть бути перехоплення повідомлень по мережі, незаконне копіювання файлів або програм.

Зміна (повна або часткова, компрометація або дезінформація). Несанкціонована сторона не тільки отримує доступ до системи та її об'єктів, але й втручається в роботу її компонентів (рис.2,г). Метою цієї атаки є *порушення цілісності.* Приклади: заміна значень у файлі даних, зміна програми таким чином, що вона працюватиме по-іншому, зміна вмісту переданих по мережі повідомлень. Цінна інформація може бути втрачена або знецінена шляхом її незаконного вилучення або модифікації;

*Підробка***.** Несанкціонована сторона розміщує в системі підроблені об'єкти (рис.2,д). Метою цієї атаки є *порушення аутентичності*. Прикладами можуть служити розміщення в мережі підробних повідомлень або додавання записів у файл.

1.2.2 Загроза за об'єктом атаки

Впливу можуть піддаватися такі об'єкти комп'ютерної системи.

Комп'ютерна система в цілому – зловмисник намагається проникнути в систему для виконання несанкціонованих дій. Тут може бути використаний метод так званого "маскараду", перехоплення або підбір паролю, зламування КС або доступ до неї через мережу.

Об'єкти комп'ютерної системи – дані або програми в оперативній пам'яті (ОП) або на зовнішніх носіях, самі пристрої системи як зовнішні (дисководи, термінали, мережні пристрої), так і внутрішні (оперативна пам'ять, процесор), канали передачі даних. Метою такої загрози є або доступ до змісту інформації (порушення конфіденційності, цілісності), або порушення функціональності (заповнення всієї ОП безглуздою інформацією, завантаження процесора завданням з необмеженим часом виконання).

Розділяють такі об'єкти КС:

- *апаратне забезпечення***.** Основна загроза для нього пов'язана з його доступністю. Апаратне забезпечення найбільше підлягає атакам і найменше піддається автоматичному керуванню. В число загроз входять випадкове або наперед сплановане виведення обладнання з ладу, а також його крадіжка. Розповсюдженість ПК та використання мереж призводять до збільшення потенційних можливостей втрат в цій області. Для запобігання загрозам подібного роду потрібні адміністративні міри щодо попередження фізичного доступу до систем;
- *програмне забезпечення* (ПЗ)**.** Основна небезпека для ПЗ полягає в її доступності. Програми, особливо прикладні, надзвичайно легко знищити. Крім того, ПЗ може бути спотворено або змінено, в результаті

чого воно стане неприйнятним для використання. Акуратне управління налаштуванням конфігурації програм, зберігання резервних копій допоможе підвищити надійність їх роботи. Складніше розв'язати проблему, якщо зміна програми призводить до того, що вона продовжує працювати, але поводить себе не так, як потрібно. Ця категорія атак пов'язана з *комп'ютерними вірусами*. Крім того, неабияке значення має захист програмного за безпечення від *несанкціонованого доступу* (НСД) та *несанкціонованого копіювання* (НСК);

- *дані*. Безпека даних охоплює широкий круг питань, що включає себе і доступність, і секретність, і цілісність. Коли йдеться про доступність, мається на увазі захист від випадкового або передбаченого спотворення файлів з даними. Для забезпечення секретності даних необхідно турбуватись про несанкціоноване зчитування файлів даних і баз даних. Особливої уваги заслуговують статистичні бази даних, в яких зберігається інформація індивідуального характеру або інформація підприємств та відомств, що не повинна підлягати загальному перегляду чи розголошенню. Крім того, важливою задачею є зберігання цілісності даних, оскільки зміна файлів даних може мати різні наслідки – від незначних до катастрофічних;
- *лінії зв'язку та канали передач***.** Це пакети даних, які передаються по каналах зв'язку (атака на об'єкт мережі) або самі канали. Це може бути і підслуховування каналу (порушення конфідентційності) та аналіз трафіка (потік повідомлень), заміна або модифікація повідомлень в каналах зв'язку (порушення цілісності), заміна топології і характеристик мережі, правил комутації і адресації.

Суб'єкти комп'ютерної системи – процеси і підпроцеси користувачів, підсистеми, мережі. Мета таких атак – або прямий вплив на роботу процесу (призупинення, зміна привілеїв та характеристик), або зворотний вплив (використання зловмисником привілеїв, характеристик процесу для своєї мети), або впровадження зловмисником вірусу в середовище другого процесу і його виконання від імені цього процесу.

1.2.3 Загроза за принципом впливу на КС

Загрози з використанням доступу суб'єкта КС (користувача, процеса) *до об'єкта* (файла даних, канала зв'яку і т.д.).

Загрози з використанням прихованих каналів. Прихований канал (convert channel) – це шлях передачі інформації, що дозволяє двом взаємодіючим процесам обмінюватись інформацією способом, який порушує системну політику безпеки КС. Вони бувають 2 видів:

- *канали з пам'яттю* (convert storage channel), коли здійснюється читання або записування інформації другого каналу за допомогою проміжних об'єктів для зберігання інформації (тимчасова пам'ять);

- *тимчасові канали* (convert timing channel), коли один процес може отримувати інформацію про хід другого, використовуючи інтервали між якими-небудь подіями (наприклад, аналіз часового інтервалу між запитом на введення-виведення дозволяє зробити висновок про розмір введеної інформації).

Взаємодія, яка базується на першому принципі, простіша, більш інформативна, тут відбувається взаємодія суб'єкта і об'єкта, що змінює стан КС. Від загрози такого роду легше захиститися, оскільки її легше виявити.

Взаємодія на основі другого принципу менш інформативна, використовуються лише побічні ефекти, що не впливає на стан КС, і тому така загроза більш складна для її виявлення та усунення.

1.2.4 Загроза за характером впливу на КС та його об'єкти

Активний вплив пов'язаний з виконанням користувачем будь-яких дій, що виходять за рамки його обов'язків і порушують існуючу політику безпеки (доступ до певних наборів даних, програм, підбір пароля,...). Цей вплив призводить до зміни стану КС. Він може здійснюватись як з використанням доступу, так і при допомозі прихованих каналів.

Активний вплив проявляється у випадках:

- *безпосереднього впливу* на об'єкт атаки (у тому числі з використанням привілеїв). Наприклад, безпосередній доступ до набору даних, програми, служби, каналу зв'язку, якщо використовувати яку-небудь помилку в захисті КС. Цим діям можна запобігти, використовуючи контроль доступу. Активні атаки викликають деяку зміну потоків даних і розділяються на 4 категорії:
  а) імітація – має місце, коли деякий об'єкт видає себе за інший Як правило, атака з імітацією використовується разом з активними атаками інших видів. Наприклад, може перехоплюватись, а потім використовуватись послідовність повідомлень, що передаються в процесі аутентифікації, в результаті чого авторизовані сторони з малими привілеями отримують додаткові привілеї, які їм не було надано;
  б) відтворення – включає в себе пасивне перехоплення елементів даних з їх наступним повторним передаванням з метою здійснити не авторизований доступ;
  в) зміна повідомлень – зміна якоїсь частини початкового повідомлення, видалення повідомлення або зміна порядку їх отримання;
  г) відмова від обслуговування – заважає нормальному використанню засобів зв'язку або керування ними чи їх стримування. Метою цієї атаки можуть бути, наприклад, підрив роботи мережі виведенням її з ладу, перевантаження повідомленнями для зниження її працезда-

тності, знищення повідомлень, призначених конкретному адресату (приватному або офіційному).

- *впливу на систему дозволів* (у тому числі захоплення привілеїв). Тут несанкціоновані дії виконуються щодо прав користувача на об'єкт атаки, а сам доступ потім здійснюється вже законним шляхом.

Опосередкований вплив (через інших користувачів) є дуже небезпечним Для запобігання йому необхідний постійний контроль як з боку адміністраторів, операторів, так і з боку користувачів за своїми даними. Опосередкований вплив проявляється у випадках:

- *"маскараду"* – користувач присвоює собі повноваження іншого користувача, видаючи себе за нього;
- *"використання навмання"* – один користувач заставляє іншого виконувати необхідні дії, причому останній може і не підозрювати про це. Для реалізації цих загроз може використовуватись вірус (це так звані шкідливі програми: "троянський кінь" і "черв'як").

Пасивний вплив здійснюється шляхом спостереження користувачем побічних ефектів та їх аналізу (підслуховування ліній зв'язку між двома вузлами мережі), що порушує конфіденційність. Стан КС при цьому не змінюється. До пасивних атак відносяться:

- *добування вмісту повідомлень* під час телефонної розмови або за допомогою електронної пошти, під час яких може бути передано важливу або конфіденційну інформацію;
- *аналіз трафіка (traffic analysis)*. Припустимо, вміст інформації, що передається, є можливість приховати повідомлення (припустимо, за допомогою шифрування), і тоді не можна добути з нього інформацію. Але опонент може отримати відомості про характер повідомлення: визначити місце розташування і параметри вузлів, що обмінюються інформацією, зібрати відомості про частоту передавання повідомлень, їх розмір, а потім зробити висновки щодо переданої інформації.

Пасивні атаки складно виявити, оскільки вони не приводять ні до яких змін даних. Але передбачити ці атаки можливо. Таким чином, слід зосередити увагу не на виявленні пасивних атак, а на їх попередженні.

1.2.5 Загроза через появи помилок захисту

Неадекватність політики безпеки КС. Розроблена політика безпеки для даної КС настільки не відображає реальні аспекти обробки інформації, що стає можливим використати цю невідповідність для НСД. Всі КС мають деяку невідповідність, але треба планувати політику безпеки таким чином, щоб вона не могла призвести до порушень. Спосіб запобігти цій загрозі – замінити засоби захисту для реалізації нової політики безпеки.

Помилки адміністративного керування – некоректна реалізація або неадекватна підтримка прийнятої політики безпеки в даній комп'ютерний

системі. Наприклад, доступ користувачів до певного набору даних повинен бути закритий, а фактично (через неуважність адміністратора) цей набір доступний всім користувачам. Для запобігання цій загрозі досить виправити таку помилку.

Помилки в алгоритмах програм – це помилки, допущені на етапі проектування програм, завдяки чому їх можна використати зовсім не так, як описано в документації. Приклад: помилки в програмі аутентифікації користувача, коли за допомогою певних дій користувач має можливість увійти в систему без пароля. Ці помилки важко знайти. Щоб усунути таку загрозу, слід змінити програму або комплекс програм.

Помилки реалізації алгоритмів програм (помилки кодування) – виникають на етапі реалізації або наладки. Прикладом таких помилок можуть служити так звані “люки”. Їх виявити найважче.

1.2.6 Загроза за способом впливу на КС

Вплив на КС систему може здійснюватись в одному з режимів:

- *в інтерактивному режимі* – користувач може активно впливати на хід виконання програми, вводячи різні команди або дані. До них належать інтерпретатори командних мов, деякі утиліти, програми керування базами даних, тобто, програми, які орієнтовані на роботу з користувачем. При використанні програм цього класу (наприклад, для атаки на КС за допомогою командного інтерпретатора) вплив є довшим у часі і тому має більшу імовірність бути виявленим;
- *у пакетному режимі* – коли всю інформацію треба готувати заздалегідь. Це системні і прикладні програми, які орієнтовані на виконання яких-небудь суворо визначених дій без участі користувача. Вплив за допомогою цих програм є короткостроковим. Його важко діагностувати, він є більш небезпечний, вимагає більшої попередньої підготовки для того, щоб передбачити всі можливі наслідки втручання.

1.2.7 Загроза за засобами атаки

Для впливу на КС зловмисник може використовувати:

- *стандартне програмне забезпечення*. У цьому випадку результати впливу здебільшого передбачені, оскільки ці програми здебільшого добре вивчені;
- *спеціально розроблені програми*. Це може бути небезпечно, і тому в захищуваних системах бажано не допускати установлення в КС програм без дозволу адміністратора.

1.2.8 Загроза за станом об’єкта атаки

Стан об’єкта в момент атаки може зробити суттєвий вплив на результати атаки і на роботу з ліквідації її наслідків. Об’єкт може бути в одному зі станів:

- *зберігання* – на диску, в оперативній пам’яті, в іншому пасивному стані. При цьому доступ до об’єкта здійснюється з використанням доступу;
- *передачі* – по лінії зв’язку між вузлами мережі, або в середині вузла. Вплив передбачає підслуховування, перехоплення, спотворення і т.д.;
- *обробки* – це коли об’єктом стає процес користувача.

1.3 Рівні захисту операційних систем

У відповідності з якістю наданого захисту існують різни рівні (або варіанти) захисту. Конкретна операційна система (ОС) може надавати захист різного ступеню для різних об’єктів, користувачів та додатків.

Відсутність захисту. Цей варіант підходить, коли відповідні процедури виконуються за часом окремо.

Ізоляція. Кожний процес працює окремо від інших процесів, не використовуючи сумісно ніякі ресурси і не обмінюючись інформацією. Кожний процес має свій адресний простір, свої файли та інші об’єкти.

Повний розподіл або повна його відсутність. Власник об’єкта (файла, сегмента пам’яті) об’являє його відкритим або закритим. У першому випадку доступ до об’єкта може отримати будь-який процес; у другому – доступ до цього об’єкта надається тільки власнику;

Спільне використання з обмеженням доступу. ОС перевіряє дозволеність доступу кожного окремого користувача до кожного окремого об’єкта. В цьому сенсі ОС виступає в якості охоронця, гарантуючи, що доступ до об’єкта отримають тільки авторизовані користувачі.

Спільне використання за допомогою динамічних можливостей. Цей варіант розширює концепцію контролю доступу, дозволяючи динамічно створювати права спільного використання об’єкта.

Обмеження на використання об’єкта. При цьому обмежується не стільки доступ до об’єкта, скільки його використання. Наприклад, користувачеві дозволено переглядати важливий документ, але не роздруковувати, або користувач має доступ до бази даних, може брати з неї статистичні зведення, але не має можливості визначити значення певних величин.

Необхідно, щоб в операційній системі підтримувався баланс між можливостями спільного використання компонентів КС, що сприяє підвищенню ефективності її використання, і ступенем захищеності ресурсів окремих користувачів.

1.4 Об’єкти захисту в операційних системах

В основі багатозадачності лежить здатність системи надавати користувачам можливість спільного використання ресурсів. Об’єктом спільного використання є не тільки процесор, але й такі елементи як:

- пам’ять;
- пристрої введення-виведення (наприклад, диски, принтери);
- програми;
- дані.

1.4.1 Захист пам’яті

В багатозадачному середовищі захист пам’яті стає важливою проблемою. Тут виникають питання, пов’язані не тільки з безпекою, але й з правильною роботою різних активних процесів. Якщо один з процесів необережно запише що-небудь в область пам’яті іншого процесу, то цим він може порушити роботу останнього.

Розподіл простору пам’яті між різними процесами легко здійснюється за допомогою *використання схеми віртуальної пам’яті* (сегментної, сторінкової або комбінованої). Якщо треба забезпечити повну ізоляцію, то операційній системі достатньо буде впевнитись, що кожний сегмент і кожна сторінка доступні тільки тому процесу, якому вона розподілена. Цього легко досягти, слідкуючи, щоб в таблицях сторінок та/або сегментів не було елементів, що повторюються. Якщо ж спільне використання дозволено, то один і той самий сегмент або сторінка можуть з’явитися в декількох таблицях. Цей тип спільного використання легше всього здійснити в ОС, що підтримують сегментацію або комбінацію сегментації з розбиттям на сторінки. У цьому випадку програмний додаток може об’явити окремі сегменти доступними або недоступними для спільного використання, і тоді в роботу вступають механізми синхронізації.

Прикладом *апаратної підтримки захисту пам’яті* є ОС, де кожному сторінковому блоку основної пам’яті ставиться у відповідність 7-бітовий *ключ управління*, значення якого встановлює ОС. Два біти цього ключа вказують на те, чи були звернення до сторінки та чи здійснювались зміни сторінки (ці біти використовуються алгоритмами заміщення сторінок). Наступні чотири біти є розрядами керування доступом (R, W, E, A). Ще один біт – розряд захисту від вибірки, який і використовується механізмом захисту. Для отримання дозволу на доступ до якоїсь сторінки, у зверненнях процесора до пам’яті і у зверненнях пристроїв прямого доступу до пам’яті (Direct Memory Access – DMA) повинен використовуватись відповідний ключ. В розряді захисту від вибірки вказується, чи дає ключ управління доступом право тільки записування, чи право записування і читання. В

процесорі існує регістр “слово стану програми” (Program Status Word – PSW), який містить інформацію щодо виконуваного в даний момент процесу. Складовим елементом цього слова є чотирибітовий ключ PSW. Поточний ключ PSW порівнюється з кодом доступу, і дозвіл на запис буде отриманий лише при умові співпадіння ключів. Якщо біт вибірки встановлено, то ключ PSW повинен відповідати ключу доступу для читання.

1.4.2 Контроль доступу як захист даних та програм

Міри по контролю доступу можна розбити на дві категорії:

1. Контроль доступу, що здійснюється *по відношенню до користувача.* Його часто називають аутентифікацією, що не зовсім правильно, оскільки цей термін широко використовується у зв’язку з аутентифікацією повідомлень, тобто може бути використаний для різних цілей. Найбільш розповсюджений контроль доступу користувача – це процедура реєстрації, при якій користувачеві необхідно ввести свій ідентифікатор та пароль. Система ж дозволить увійти лише тоді, коли його ідентифікатор співпаде з відомим системі і коли користувач знає пароль, пов’язаний з цим ідентифікатором.
2. Контроль доступу, *орієнтований на дані*, полягає в тому, що кожному користувачеві може відповідати профіль, в якому вказуються дозволені операції і режими доступу до файлів.

Система управління файлами або базою даних базується на загальній моделі під назвою **матриці доступу** (access matrix) (рис.3). В неї входять:

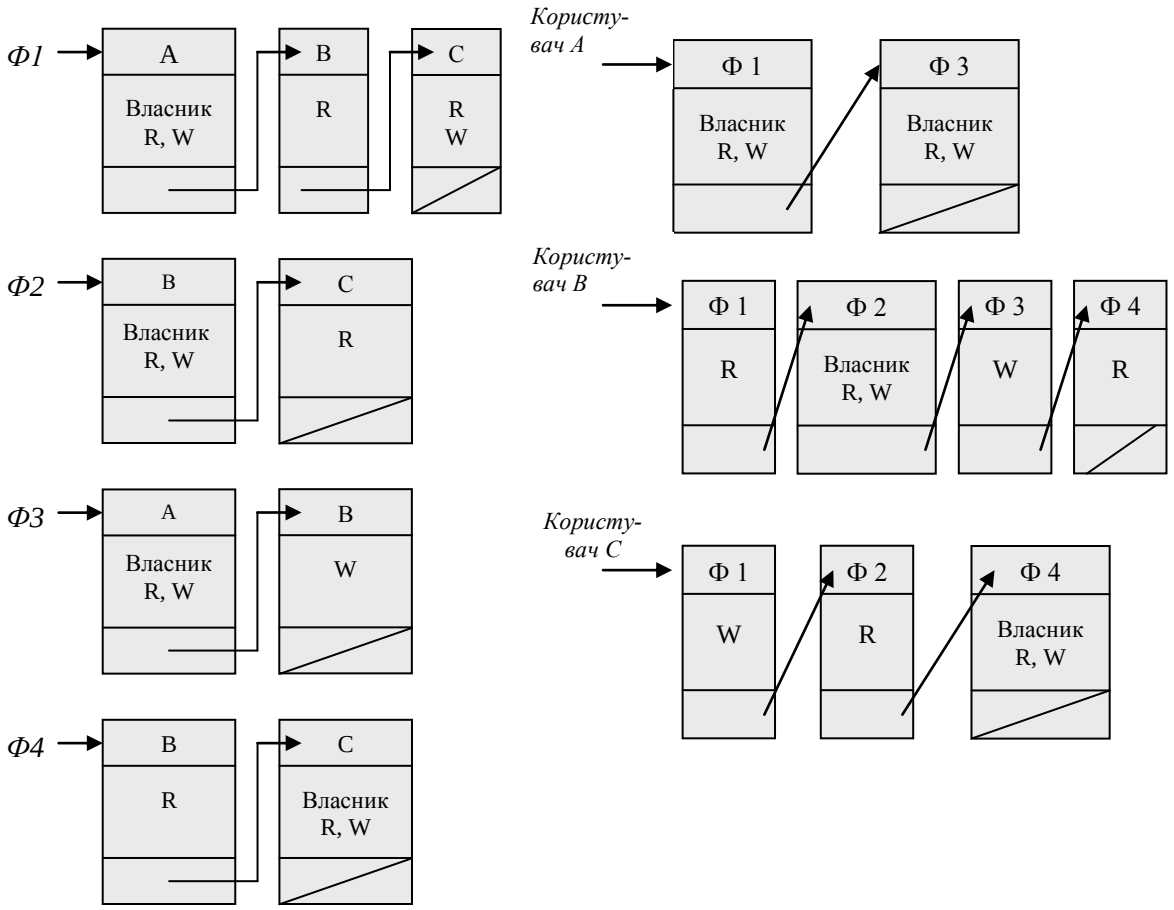
- *суб’єкти* – ідентифікатори окремих користувачів та груп користувачів, хоча контроль доступу може виконуватись і не по відношенню до користувачів, а по відношенню до терміналів, вузлів, додатків;
- *об’єкти* – все те, до чого доступ контролюється (файли, програми, сегменти пам’яті);
- *права доступу* – спосіб, за допомогою якого суб’єкт здійснює доступ до об’єкта (зчитування, записування, виконання).

	Файл 1	*Файл 2*	*Файл 3*	*Файл 4*	*Обліковий запис 1*	*Обліковий запис 2*
Користувач A	Власник R W		Власник R W		Кредит запитів	
Користувач B	R	Власник	W	R	Дебет запитів	Кредит запитів
Користувач C	R W	R		Власник R W		Дебет запитів

Рисунок 3 – Матриця доступу

На практиці матриця доступу у такому вигляді застосовується рідко. Як правило, при реалізації вона розкладається одним з двох способів: списки контроля доступу та мандати можливостей.

Списки контроля доступу (access control list) – матриця доступу розкладається за стовпчиками, і тоді кожному об'єкту відповідає список, в якому перелічені користувачі, а також вказані їх права доступу (рис.4,а).



а) Списки управління доступом для файлів з матриці доступу

б) Списки (мандати) можливостей для файлів з матриці доступу

Рисунок 4 – Приклад структур управління доступом

Мандати можливостей (capability tickets), або списки можливостей отримуються в результаті розкладання матриці доступу за рядками (рис.4,б). В мандаті можливостей вказані об'єкти і операції, санкціоновані для даного користувача.

Оскільки мандати можливостей можуть бути розсіяні по системі, вони є більшою проблемою для безпеки, ніж списки контролю доступу, а тому зберігати їх треба в області пам'яті, що є недоступною для користувача.

1.6 Категорії зламщиків

Зламщики (хакери, крекери) є найбільш відомою загрозою для безпеки операційних систем (інший вид загроз – це віруси). Ідентифікують такі класи зламщиків:

- *удавальник* (рос. "притворщик") – особа, що не має повноважень щодо використання комп'ютера, яка проникає в систему, не дивлячись на контроль доступу системи, і використовує обліковий запис законного користувача. Удавальник – це, як правило, стороння людина;
- *правопорушник* – законний користувач (зазвичай не стороння людина), що отримує доступ до даних, програм та ресурсів, до яких у нього немає доступу, або той, у якого є доступ, але він зловживає своїми привілеями;
- *таємний користувач* – особа, яка заволоділа управлінням в режимі суперкористувача і використовує його для того, щоб запобігти аудита і перебороти контроль доступу, або для запобігання збору даних щодо аудита. Це може бути як стороння людина, так і не стороння.

Наслідки атак зловмисників можуть бути різними – від незначних до досить таки серйозних. Внизу шкали зламщиків розташовані ті, які хочуть просто використати мережі і дізнатись, що і де знаходиться. На протилежному кінці шкали розміщуються індивідууми, що намагаються прочитати службові дані, використати несанкціоновану зміну даних або зруйнувати систему.

1.7 Атаки на рівні операційних систем

В загальному випадку програмне забезпечення будь-якої універсальної комп'ютерної системи складається з трьох основних компонентів: операційної системи, мереженого програмного забезпечення і системи управління базами даних. Тому і методи зламу захисту КС можна поділити на 3 групи:

- атаки на рівні операційних систем;
- атаки на рівні мереженого програмного забезпечення;
- атаки на рівня систем керування базами даних.

Внутрішня структура сучасних операційних систем є надзвичайно складною, і тому дотримання адекватної політики безпеки є досить складною задачею.

Успіх реалізації того чи іншого алгоритму хакерської атаки на практиці в значній мірі залежить від архітектури і конфігурації конкретної ОС, яка є об'єктом цієї атаки. Однак, існують атаки, яким може підлягати практично будь-яка операційна система.

1. Крадіжка пароля:

- підглядання за користувачем, коли той вводить пароль, що дає право на роботу з ОС (навіть якщо під час його введення пароль не висвітлюється на екрані, хакер може легко дізнатись про нього, просто спостерігаючи за переміщенням пальців користувача по клавіатурі);
- отримання пароля з файла, в якому цей пароль було збережено користувачем, який не бажає утруднювати себе введенням паролю при підключенні до мережі (як правило, такий пароль зберігається у файлі навіть у незашифрованому вигляді);
- пошук пароля, який користувач, щоб не забути його, записує на календарях, в записниках або на зворотній стороні клавіатури (особливо часто таке трапляється, коли адміністратори заставляють користувачів застосовувати паролі, що тяжко запам'ятати);
- крадіжка зовнішнього носія парольної інформації (дискети, електронного ключа, на яких зберігається пароль користувача, призначений для входу в систему);
- повний перебір всіх можливих варіантів паролю;
- підбір пароля за частотою зустрічаємих символів та біграм, за допомогою словників найуживаніших паролів, із залученням знань про конкретного користувача – його імені, прізвища, номера телефону, дати народження, тощо.

2. *Сканування жорстких дисків* комп'ютера, коли хакер намагається послідовно звернутись до кожного файла. Якщо об'єм диска досить великий, то можна бути впевненим, що при описі доступу до файлів і каталогів адміністратор припустив хоча б одну помилку, в результаті чого, всі такі каталоги і файли будуть прочитані хакером. Для знищення слідів хакер може організувати цю атаку під чужим іменем.
3. *Збирання "сміття"* – якщо засоби ОС дозволяють відновлювати раніше знищені об'єкти. Тоді хакер може використати цю можливість щоб отримати доступ до об'єктів, знищених іншими користувачами (наприклад, передивляючись вміст їх "сміттєвих корзин").
4. *Перевищення повноважень* – використовуючи помилки в програмному забезпеченні або в адмініструванні ОС, хакер отримує повноваження, що перевищують надані йому згідно діючій політиці безпеки:
   - запуск програм від імені користувача, що має ці необхідні повноваження, або в якості системної програми (драйвера, сервісу, демона і т.д.);
   - підміна динамічно завантажуваної бібліотеки. що використовується системними програмами, або надання інших значень змінним середовища, що описують шлях до таких бібліотек;
   - модифікація коду або даних підсистеми захисту самої операційної системи.
5. *Відмова в обслуговуванні*. Метою цієї атаки є часткове або повне виведення з ладу операційної системи:

- захоплення ресурсів (хакерська програма здійснює захоплення всіх наявних в ОС ресурсів, а потім входить в нескінчений цикл);
- бомбардування запитами (хакерська програма постійно направляє операційній системі запити, реакція на які потребує залучення значних ресурсів комп'ютера);
- використання помилок в програмному забезпеченні або адмініструванні.

Контрольні питання

1. Навести основні вимоги до безпеки комп'ютерних систем.
2. Як класифікуються загрози безпеці КС за метою реалізації?
3. Назвіть об'єкти атаки КС, які можуть підлягати загрозам.
4. Охарактеризувати загрози безпеці КС за принципом та характером впливу.
5. Яким загрозам може підлягати КС через появу помилок у захисті?
6. Дати характеристику загрозам КС за способом впливу, засобами атаки та станом об'єкта атаки.
7. Назвати рівні атаки на операційну систему і охарактеризувати кожний з рівнів.
8. Що є об'єктом захисту в операційній системі?
9. На чому базується захист оперативної пам'яті в операційній системі?
10. Яким чином здійснюється захист даних та програм в операційній системі?
11. Навести категорії зламщиків захисту ОС і наведіть приклади кожній з категорій.
12. Які види атак можуть здійснювати зламщики на рівні операційних систем в цілому?
13. Яким чином може здійснюватися "крадіжка пароля"?
14. Що означає така хакерскька атака, як відмова в обслуговуванні?
15. Що означає термін "збір сміття"?
16. Для чого хакер може використовувати сканування жорстких дисків?

2 ХАРАКТЕРИСТИКА НАЙПОШИРЕНІШИХ ЗАГРОЗ БЕЗПЕЦІ КОМП'ЮТЕРНИХ СИСТЕМ

Розглянемо наслідки, до яких може привести реалізація загроз і наведемо рекомендації щодо захисту від них. Кожна з розглянутих нижче загроз знаходить своє місце у проведеній класифікації загроз безпеці КС.

До найпоширеніших загроз можна віднести:

- несанкціонований доступ;
- незаконне використання привілеїв;
- атаки "салямі";
- приховані канали;
- атаки типу "маскарад";
- "збір сміття";
- "зламування системи" (break-in);
- шкідливе програмне забезпечення.

2.1 Несанкціонований доступ

Несанкціонований доступ (НСД) (unauthorized access). Це найпоширеніший вид комп'ютерних порушень. Він полягає в отримані користувачем доступу до об'єкта, на який у нього немає дозволу. За характером впливу НСД є активним впливом, йому може підлягати будь-який об'єкт КС. Несанкціонований доступ може бути здійснений як стандартними, так і спеціально розробленими програмними засобами до об'єктів у будь-якому стані (при зберіганні, при передачі, при обробці інформації).

Методика реалізації НСД залежить від організації обробки інформації в даній комп'ютерній системі, від організації політики безпеки, від можливостей встановлених засобів захисту. Несанкціонований доступ стає можливим через непродуманий вибір засобів захисту, їх некоректне встановлення і настроювання, контроль роботи, недбале ставлення до захисту своїх власних даних.

Для реалізації НСД використовують два способи:

- *можна подолати систему захисту,* тобто шляхом різних впливів на неї припинити її дії стосовно себе або своїх програм (це складно, трудомістко, не завжди можливо, але ефектно);
- *можна спостерігати* за тим, що "погано лежить", тобто які дані, що становлять інтерес для зловмисника, відкриті через недогляд або навмисно адміністратором. Такий НСД легко здійснити, але від нього і легше захиститись.

2.2 Незаконне використання привілеїв

Для даного способу атаки здебільшого використовується штатне програмне забезпечення (системне і прикладне), але яке функціонує в нештатному режимі. Майже будь-яка захищена система вміщує засоби, які можуть працювати з порушенням існуючої політики безпеки. У більшості випадків небезпечні засоби, які не повинні бути доступні звичайному користувачу, використовуються адміністраторами, системними програмістами та тими користувачами, які виконують деякі спеціалізовані функції.

Для того, щоб зменшити ризик від застосування таких засобів, більшість комп'ютерних систем захисту реалізує ці функції за допомогою набору привілеїв. Кожний користувач отримує свій набір привілеїв. Набори привілеїв кожного користувача є його атрибутами і зберігаються системою захисту. Отже, заповітна мрія зловмисника в такому випадку – оволодіти розширеним набором привілеїв.

Незаконне захоплення привілеїв можливе або за наявності помилок у самій системі захисту, або у випадку недобросовісного керування КС взагалі і системою привілеїв зокрема.

2.3 Атаки "салямі" (salami attack)

Атаки такого виду можливі в системах, де, наприклад, обробляються грошові рахунки для банків. Принцип атак "салямі" побудований на тому факті, що при обробці рахунків використовуються цілі одиниці (гривні, рублі, центи, копійки), а при нарахуванні процентів майже завжди виходять дробові числа.

Наприклад, 6.5% річних від суми 102.87 коп. за 31 день становлять 0.5495726 грн. Будь-яка банківська система заокруглить цю суму до 0.55 грн. Якщо робітник банку має доступ до банківських рахунків або до програм їх обробки, то він може округлити цю суму в інший бік – 0.54 грн., а різницю в 1 коп. скидати на свій рахунок. Власник рахунку ніколи не помітить цієї помилки або спише її на похибки обробки інформації. Таким чином, зловмисник при обробці за день 10 000 рахунків матиме 100 грн., або більше 30 000 грн. за рік.

Назва ж "салямі" пішла від ковбаси з такою назвою, яка виготовляється з різних сортів м'яса. Таким саме чином рахунок зловмисника поповнюється за рахунок різних вкладників.

Отже, атаки даного типу переважають у великих банках та інших фінансових організаціях. Причинами цих атак є:

- похибки обчислень, які дозволяють двояко інтерпретувати правила округлення чисел;

- великі обсяги обчислень, які необхідно виконувати при обробці рахунків клієнтури.

Атаки “салямі” досить важко розпізнаються (якщо тільки на рахунку зловмисника не накопичується величезна сума, яка може привернути увагу).

Запобігти цим атакам можна лише забезпечивши цілісність і коректність прикладних програм, розмежуванням доступу користувачів, постійним контролем рахунків на предмет їх змінювання.

2.4 Приховані канали

Приховані канали (convert channels) – це шляхи передачі інформації між процесами системи, які порушують політику безпеки. Користувач може не мати дозволу на обробку даних, які його цікавлять, але він шукає обхідні шляхи. Оскільки будь-яка дія в системі викликає зміни стану інших складових системи, то за умови спостережливості і знання цих зв’язків можна відновити першопричину події хоча б частково.

Реалізовані “приховані канали” можуть бути різними шляхами, наприклад, за допомогою закладання “троянських коней”.

Наприклад, програміст банку не завжди має доступ до імен та балансів депозитних рахунків, а програміст системи не має доступу до пропозицій про купівлю та продаж. Але при створенні таких систем він може передбачити спосіб отримання цих відомостей. В цьому разі програма встановлює таємно канал зв’язку з цим програмістом і повідомляє йому необхідні дані.

Прикладом активізації “прихованих каналів” може бути кінцевий звіт, в якому замість одного слова буде використовуватись інше. “Прихованим каналом” може стати число пропусків між двома словами, або значення третьої цифри після коми в якомусь виразі. на який ніхто не зверне увагу. “Прихованим каналом” може стати і інформація про присутність або відсутність якогось набору даних, його розміру, дати створення і модифікації тощо. Отож, існує багато способів організації зв’язку між двома процесами. Більше того, більшість ОС мають у своєму розпорядженні такі засоби, оскільки вони полегшують роботу користувачів і програмістів. Тут важливо розрізняти недозволені і дозволені “приховані канали”.

Атаки з використанням “прихованих каналів” у всіх випадках приводять до порушення конфіденційності інформації. За характером впливу вони є пасивними, для їх організації може бути використане як спеціальне, так і стандартне програмне забезпечення. Атаки здебільшого здійснюються програмним методом у пакетному режимі.

Характерними особливостями “прихованих каналів” є їх мала пропускна спроможність, оскільки по них можна передавати невелику кількість інформації, а також великі труднощі в їх організації і малі, як правило,

збитки, яких вони завдають. Найчастіше ці збитки взагалі бувають непомітними, тому спеціальний захист проти “прихованих каналів” здійснюється дуже рідко.

2.5 Атаки типу “маскарад”

Під “маскарадом” (masquerade) (симуляція, моделювання) розуміється виконання яких-небудь дій одним користувачем КС від імені іншого, тобто права і привілеї одного користувача КС присвоюються іншим з метою доступу до наборів даних першого і використання його привілеїв.

Приклади “маскараду”:

а) вхід в систему під ім’ям і паролем іншого користувача (при цьому система захисту не зможе розпізнати це порушення). В цьому випадку “маскараду” передує зламування системи або перехоплення паролю;

б) привласнення імені іншого користувача в процесі роботи за допомогою засобів операційних систем (деякі ОС дозволяють змінювати ідентифікатор користувач в процесі роботи) або за допомогою спеціальної програми, яка у визначеному місці змінює певні дані, в результаті чого користувач одержить інше ім’я;

в) передача повідомлень у мережі від імені іншого користувача. Особливо небезпечно, якщо це стосується керуючих повідомлень, які можуть змінити конфігурацію мережі, або повідомлень, які пов’язані з виконанням привілейованих операцій.

Дуже небезпечний “маскарад” у банківських системах електронних платежів, де невірна ідентифікація клієнта може призвести до великих втрат. Особливо це стосується платежів за допомогою електронних карток. Сам по собі метод ідентифікації за допомогою персонального ідентифікатора (Personal Identification Number (PIN)) досить надійний, а порушення можуть виникнути внаслідок неправильного його використання.

“Маскарад” – досить серйозне порушення, яке може призвести до тяжких наслідків (зміна конфігурації системи, відтік інформації, порушення роботи ОС).

Для запобігання “маскараду” необхідно:

- використовувати надійні методи ідентифікації і аутентифікації;
- передбачати блокування спроб зламу системи,
- здійснювати контроль входу в систему,
- передбачати фіксацію всіх подій, які можуть свідчити про “маскарад” з метою їх подальшого аналізу;
- відмовлятись від програмних продуктів, які містять помилки і можуть призвести до “маскараду”.

2.6 “Збір сміття”

Після закінчення роботи по обробці інформації частина даних може залишитися в оперативній пам’яті, на дисках, магнітних стрічках та інших носіях і зберігатися там до перезаписування або знищення. Прочитати прямим звичайним способом їх важко, але при використанні спеціальних програм і обладнання це все ж таки можна зробити. Такий процес і називається “збором сміття” (disk scavenging, garbage collection).

Для захисту від “збору сміття” використовуються спеціальні механізми, які можуть бути реалізовані в операційній системі і/або апаратурі комп’ютера чи в додаткових програмних (апаратних) засобах. Прикладами таких механізмів є стираючий зразок і мітка повноти.

Стираючий зразок (erasure pattern) – це послідовність бітів, яка записується на певне місце та стирає дані. Адміністратор може автоматично активізувати запис цієї послідовності при кожному звільнені ділянки пам’яті. При цьому стерті дані знищуються і ніхто не зможе вже їх відновити або прочитати (без спеціальної апаратури).

Мітка повноти (highewater marking) – робить неможливим читання ділянок пам’яті, відведених для процесу записування, але не використаних ним. Верхня межа пам’яті, яка використовується, і є міткою повноти. Цей спосіб використовується для захисту послідовних файлів виняткового доступу (результуючі файли редакторів, компіляторів, компоновщиків та їм подібних). Для індексних файлів і послідовних розділюваних файлів цей метод носить назву “стирання при розміщенні”, тобто пам’ять очищується при виділенні її процесу.

2.7 “Зламування системи”

“Зламування системи” (break-in) – це навмисне проникнення в комп’ютерну систему з несанкціонованими параметрами входу (іменем користувача і його паролем).

Можливими причинами зламування може бути таке: помилки в керуванні системою захисту; помилки в проектуванні систем захисту; помилки в кодуванні алгоритмів захисту. “Зламування системи”, як правило, здійснюється в інтерактивному режимі.

Оскільки ім’я користувача зазвичай не є таємницею, то об’єктом полювання для зламщиків в більшості випадків є пароль. Способи розкриття паролю можуть бути різними: перебір можливих паролів, “маскарад” з використанням іншого користувача, захоплення привілеїв.

Основне навантаження на захист КС від зламування несе програма входу. Алгоритм вводу імені і пароля, їх шифрування, правила зберігання і заміни паролів не повинні мати помилок.

2.8 Шкідливе програмне забезпечення

Цілком очевидно, що найбільш витончені загрози для КС являють собою програми, що досліджують їх вразливі місця.

Шкідливі програми (malicious software або malware) – це програми, які призначені для того, щоб чинити шкоду і використовувати ресурси комп'ютера, вибраного в якості мішені. Вона часто маскуються в легальних програмах або імітуються під них. В деяких випадках вони розповсюджуються самі по собі, переходячи по електронній пошті від одного комп'ютера до іншого, або через заражені файли і диски. Загальна класифікаційна схема шкідливих програм представлена на рис.5.

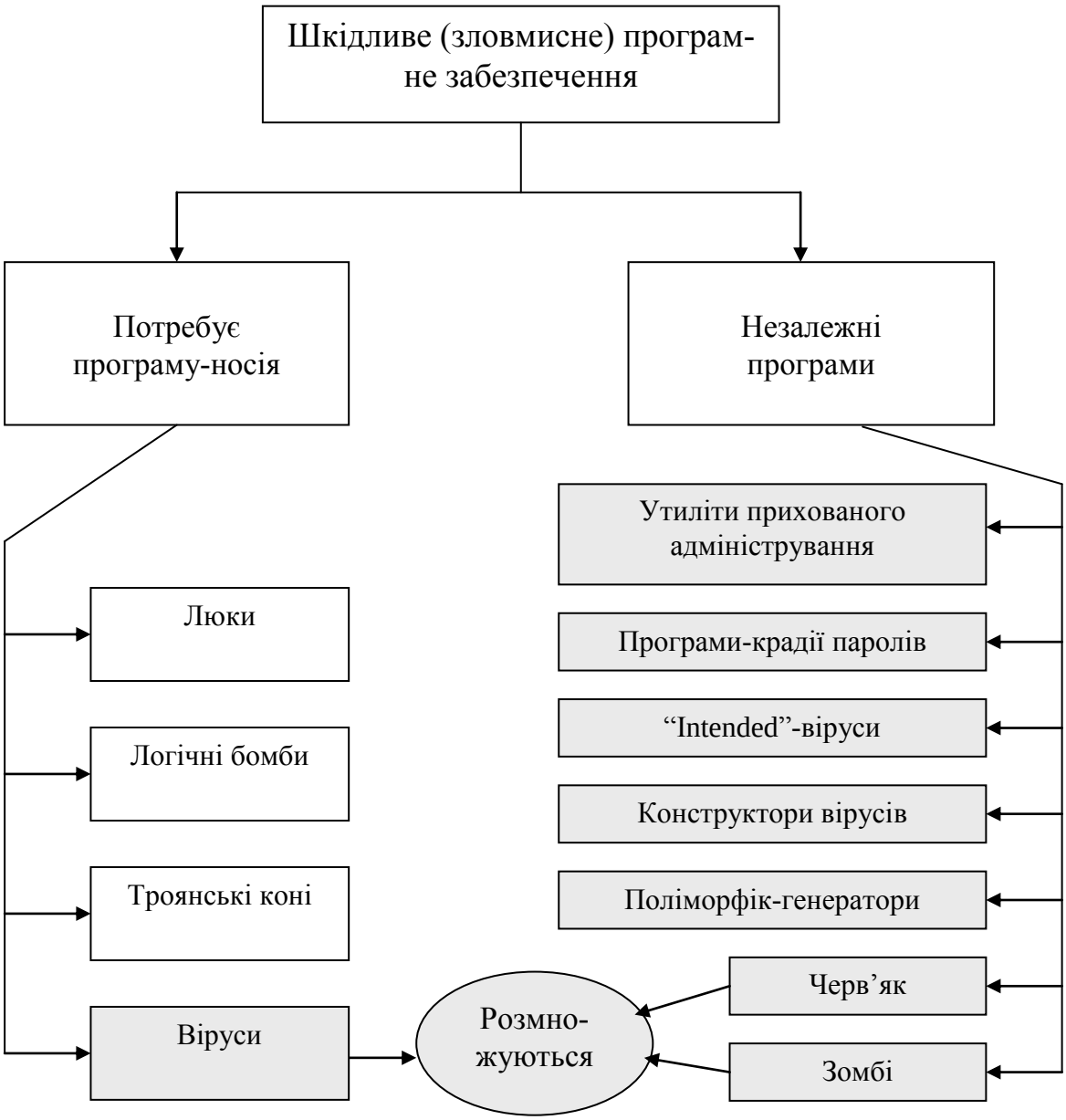


Рисунок 5 – Систематизація шкідливих програм

Загрози цього виду можна поділити на 2 групи.

Першу групу складають ті програми, що *вимагають програм-носіїв*. До них, в основному, відносяться фрагменти програм, що не можуть існувати незалежно від програм-носіїв, в ролі яких можуть виступати деякі програмні додатки, утиліти, системні програми. В цю групу входять:

- люки,
- логічні бомби,
- троянські коні,
- віруси.

У другу групу входять програми, що є *незалежними*. До них відносяться окремі назалежні програми, які можуть плануватися і запускатися операційною системою, а саме:

- черв'яки,
- зомбі;
- утиліти прихованого адміністрування;
- програми-крадії паролів;
- "intended"-віруси;
- конструктори вірусів;
- поліморфік-генератори.

Крім того, небезпечні програми поділяються на такі, що:

- *не відновлюють себе* (не розмножуються). До них відносяться фрагменти програм, які повинні активізуватися під час певних дій головної програми;/
- *розмножуються* – або фрагменти програм (віруси), або незалежні програми (черв'яки), що здатні під час запуску створювати одну або декілька копій самих себе. Ці копії пізніше також активізуються в цій самій або іншій операційній системі.

2.8.1 Люки

Люк (trapdoor) – це прихована, недокументована точка входу в програмний модуль, яка дозволяє кожному, хто про неї знає, отримати доступ до програми в обхід звичайних процедур, призначених для забезпечення безпеки КС. Люк вставляється в програму в більшості випадків на етапі налагоджування для полегшення роботи – даний модуль можна буде викликати в різних місцях, що дозволяє налагоджувати окремі його частини незалежно одна від одної. Крім того, люк може вставлятися на етапі розробки для подальшого зв'язку даного модуля з іншими модулями системи, але потім, після налагодження або внаслідок змінених умов, дана точка входу виявиться непотрібною.

Як правило, програміст розробляє програмний додаток, в який входить процедура реєстрації, або який треба довго настроювати, вводячи під час запуску багато значень. Можливо, розробник хоче надати програмі

особливі привілеї або мати можливість запобігати процесу налаштування і аутентифікації, або програмісту треба мати в своєму розпорядженні надійний метод, що дозволяє активізувати програму в разі можливих збоїв.

Наявність люка дозволяє викликати програму нестандартним способом, що може серйозно відбитися на стані системи захисту (невідомо, як у такому випадку програма буде сприймати дані, середовище системи, тощо). Крім того, не завжди можна прогнозувати її поведінку.

Отже, люки можуть з'явитися в програмах з таких причин:

- їх забули усунути (необміркований промах);
- для використання при подальшому налагоджуванні;
- для забезпечення підтримки готової програми;
- для реалізації таємного контролю доступу до даної програми після її встановлення (перший крок до навмисного проникнення в КС з використанням даної програми).

Програмні помилки не є люками. Люк – це механізм налагодження для підтримки і корегування програм. Якщо ж люки використовуються для отримання несанкціонованого доступу, тоді вони стають загрозою.

Запобігти люкам можна, провівши аналіз початкових текстів програм, міри безпеки повинні прийматися в основному ще на етапі розробки і оновлення програм.

Прикладом люку може слугувати випадок, коли при розробці системи Multics, випробування на проникнення в яку проводилось групою "Tiger team" (команда тигрів) ВПС США, що зображала противника. Один з тактичних ходів полягав в тому, щоб відправити на вузол, працюючий під керуванням Miltics, підроблену (рос. "подложную") оновлену версію ОС. Версія містила в собі троянського коня, якого можна було активізувати за допомогою люка, і який дозволив команді отримати доступ до системи. Загроза була реалізована настільки добре, що розробники системи Multics не змогли віднайти її навіть тоді, коли вже знали про її наявність.

2.8.2 Логічні бомби

Це один із самих ранішніх видів програм-загроз. Вони є попередниками вірусів і черв'яків.

Логічна бомба – це код, що поміщається в деяку легальну програму. Він влаштований таким чином, що при певних умовах "вибухає". Умовою для включення логічної бомби може бути наявність або відсутність деяких файлів, певний день тижня або певна дата, а також запуск додатку певним користувачем.

Ось приклад логічної бомби. В одному випадку логічна бомба перевіряла ідентифікаційний номер співробітника компанії, який був автором цієї бомби, і включалась, якщо цей ідентифікатор не фігурував у двох останніх нарахуваннях заробітної плати. "Вибухаючи", бомба могла зміни-

ти або видалити дані або файли, стати причиною зупинки машини або здійснитии щось інше.

Другий приклад. У бібліотечній системі графства Монтгомері (Меріленд) підрядчик, якому доручили розробку комп'ютерізованої абонентської мережі, розмістив в ній логічну бомбу. При настанні певної дати ця бомба могла вивести систему із ладу, якщо замовник відмовлявся платити. Коли ж бібліотека затримала виплату грошей, підрядчик зізнався в існуванні "бомби" і пригрозив, що в разі неперерахування йому грошей він дасть "бомбі" спрацювати.

2.8.3 Троянські коні (Trojan Horse)

Троянські коні (Trojan Horse), або просто трояни – це досить розповсюджений і, так би мовити, популярний на сьогоднішній день вид шкідливого програмного забезпечення. До даної групи шкідливих програм відносять:

- програми-вандали,
- «дроппери» вірусів,
- «злі жарти»,
- деякі види програм-люків;
- деякі логічні бомби,
- програми вгадування паролів;
- програми прихованого адміністрування.

Останні чотири групи програм можуть і не існувати у вигляді троянів, а бути цілком самостійними програмними продуктами, що також породжують шкідливі дії в операційній системі.

Троянський кінь – це програма, яка виконує на доповнення до основних (проектних і документованих) додаткові, але не описані в документацп, дії. Троянський кінь – це корисна, або така, що здається корисною, програма або процедура, в якій приховано код, здатний в разі спрацьовування виконати деяку небажану або шкідливу функцію.

Аналогія зі старогрецьким троянським конем, таким чином, цілком виправдана – і в тому, і в іншому випадку в оболонці, яка не викликає ніякої підозри, існує загроза. Програми такого типу є серйозною загрозою для безпеки комп'ютерних систем.

Троянські коні можуть використовуватись для виконання тих функцій, які несанкціонований користувач не може виконати безпосередньо.

За характером троянський кінь належить до активних загроз, які реалізуються програмними засобами і працюють у пакетному режимі. Троянський кінь є загрозою для будь-якого об'єкта комп'ютерної системи, причому ця загроза може виражатися будь-яким із способів: безпосередній вплив на об'єкт атаки, вплив на систему дозволів, опосередкований вплив. Найнебезпечнішим є опосередкований вплив, за якого троянський кінь діє

в рамках повноважень одного користувача, але в інтересах іншого користувача, особу якого встановити майже неможливо.

Небезпека троянського коня полягає в додатковому блоці команд, встановленому тим чи іншим способом у початкову нешкідливу програму, яка потім пропонується (подарунок, продаж, заміна) користувачам комп'ютерної системи. Цей блок команд може спрацювати при виконанні деякої умови (дати, часу і т. д., або по команді ззовні). Той, хто запускає таку програму, створює небезпеку як для себе і своїх файлів, так і для всієї комп'ютерної системи в цілому. Отже, у деяких випадках *логічні бомби* також можна віднести до троянських програм.

Найбільш небезпечні дії троянський кінь може виконувати, якщо користувач, який його запустив, має розширений набір привілеїв. У цьому випадку зловмисник, який склав і впровадив троянського коня, а сам цих привілеїв не має, може виконати несанкціоновані привілейовані функції чужими руками. Або, наприклад, зловмисника дуже цікавлять набори даних користувача, який запустив таку програму. Останній може навіть не мати розширеного набору привілеїв, – це не буде перешкодою для виконання несанкціонованих дій.

Наприклад, деякий користувач-зловмисник хоче отримати доступ до файлів іншого користувача. Він пише програму, яка під час запуску змінює права доступу до файлів користувача, який її викликав, таким чином, щоб ці файли могли прочитати інші користувачі. Далі, помістивши цю програму в загальний каталог і присвоївши їй ім’я, схоже на ім’я якоїсь корисної утиліти, автор програми якимось чином досягає того, щоб потрібний користувач запустив її. Прикладом такої програми може бути програма, яка ніби-то виводить лістинг файлів користувача в потрібному форматі.

Прикладом троянського коня, який важко виявити, може бути компілятор, змінений таким чином, щоб при компіляції вставляти в певні програми (наприклад, програми реєстрації в системі) додатковий код. За допомогою такого коду в програмі реєстрації можна створити люк, що дозволяє автору входити в систему за допомогою спеціального пароля. Такого троянського коня неможливо виявити в початковому тексті програми-реєстрації. Таким чином, і *люки* можна віднести до програм-троянів.

Троянський кінь – одна з найнебезпечніших загроз безпеці операційних систем. Радикальним способом захисту від цієї загрози є створення замкнутого середовища виконання програм. Бажано також, щоб привілейовані і непривілейовані користувачі працювали з різними екземплярами прикладних програм, які мають зберігатися і захищатися індивідуально. При виконанні цих заходів імовірність впровадження подібних програм буде досить низькою.

У порівнянні з вірусами троянські коні не одержують широкого поширення через досить прості причини – вони або знищують себе разом з іншими даними на диску, або демаскують свою присутність і знищуються постраждалим користувачем.

До категорії програм-троянів відносять також *програми-вандали.* Ці програми, як правило, імітують виконання якої-небудь корисної функції або маскуються під нову версію відомого програмного продукту. При цьому в якості побічного ефекту вони знищують файли, псують каталоги, форматують диски або виконують деякі інші деструктивні дії.

До троянських коней також можна віднести *"дроппери" вірусів* – заражені файли, код яких підправлений таким чином, що відомі версії антивірусів не визначають віруса у файлі. Наприклад, файл шифрується якимнебудь спеціальним чином чи упаковується рідко використовуваним архіватором, що не дозволяє антивірусу встановити факт зараження.

Слід зазначити також *"злі жарти"* (hoax). До них відносяться програми, що не заподіюють комп'ютеру якоїсь прямої шкоди, однак виводять повідомлення про те, що така шкода вже заподіяна, або буде заподіяна за певних умов, або попереджають користувача про неіснуючу небезпеку. До "злих жартів" відносяться, наприклад, програми, що "лякають" користувача повідомленнями про форматування диска (хоча самого форматування насправді не відбувається), детектують віруси в незаражених файлах (так робить відома програма ANTITIME), виводять дивні вірусоподібні повідомлення і т.д. – у залежності від почуття гумору автора.

До такої ж категорії "злих жартів" можна віднести також свідомо помилкові повідомлення про нові супер-віруси. Такі повідомлення періодично з'являються в електронних конференціях і зазвичай викликають паніку серед користувачів.

2.8.4 Віруси

Вірус – це програма. яка може заражати інші програми, змінюючи їх (копіює програму-вірус в програму, яка, в свою чергу, може заразити інші програми).

Біологічно віруси являють собою маленькі уламки генетичного коду (ДНК або РНК), які можуть переймати структуру живих клітинок і хитрістю залучити їх до виробництва тисяч точних копій початкового вірусу. Подібно цьому комп’ютерний вірус містить в собі рецепт того, як точно відтворити самого себе. Попавши в середовище комп’ютера, типовий вірус тимчасово бере на себе керування ОС і потім, при контакті зараженого комп’ютера з незараженими програмами, вірус упроваджує в ці програми свою копію. А далі він розповсюджується таким чином через магнітні носії, через мережу.

Вірус може робити те, що робить звичайна програма. Єдина відмінність полягає в тому. що він прикріплюється до іншої програми і приховано виконується під час роботи програми-хазяїна.

За час свого існування типовий вірус проходить 4 стадії:

- *фаза спокою.* Вірус не діє, а чекає події, яка його активізує. Такою подією може бути настання певної дати, наявність іншого файла або перевищення певного об'єму диска. Але не всі віруси притримуються цієї стратегії;
- *фаза розмноження.* Вірус розміщує свою копію в інші програми або в певні системні області на диску. Потім кожна заражена програмам містить клон вірусу, який також коли-небудь почне розмножуватись;
- *фаза запуску***.** Вірус активізується для отримання можливості виконувати функції, для яких його створено. Як і вихід з фази спокою, перехід в фазу запуску може бути спровокований різними системними подіями (у тому числі – перевищення деякої припустимої кількості нових копій вірусу);
- *фаза виконання***.** Вірус виконує свої функції. Ці функції можуть бути безпечними (виведення на екран повідомлення) або заподіювати шкоду (видаляти файли з програмами і даними).

Більшість вірусів робить свою справу, пристосовуючись до ОС, в деяких випадках – до певної апаратної платформи, тобто використовують особливості і слабкості операційних систем.

Більш детально про віруси, їх різновиди і характеристики, про особливості їх існування та проявлення, про можливості їх виявлення та методи боротьби з ними йтиметься далі.

2.8.5 Черв'яки

Черв'як – це програма, яка розповсюджується через мережу і не залишає своєї копії на магнітному носії. Черв'як використовує механізм підтримки мережі для визначення вузла, який може бути заражений. Потім за допомогою тих самих механізмів передає своє тіло на цей вузол й або активізується, або чекає для цього певних сприятливих умов.

Мережні програми-черв'яки використовують мережні з'єднання, щоб переходити з однієї системи в іншу. Одноразово активізувавшись в системі, черв'як може вести себе як комп'ютерний вірус, породжувати троянських коней, виконувати інші руйнівні або деструктивні дії.

Для свого самовідтворення черв'як використовує деякий транспортний засіб:

- *електронну пошту* – черв'як розсилає свою копію іншим системам;
- *можливості віддаленого запуску програм* – черв'як запускає свою копію на іншій системі;
- *можливості віддаленої реєстрації* – черв'як входить у віддалену систему під виглядом користувача, а потім за допомогою стандартних команд копіює себе із однієї системи в іншу.

Перед тим, як копіювати себе на якусь систему, мережний черв'як може спробувати визначити, чи інфікована ця система. Крім того, в багато-

задачній системі він може маскуватися, присвоюючи собі імена системних процесів або якісь інші, які важко помітити системному адміністратору.

Найбільш відомим представником цього класу є вірус Морріса (або, вірніше, "черв'як Моріса"), який вразив мережу Internet у 1988 році. Найсприятливішим середовищем для розповсюдження черв'яка є мережа, всі користувачі якої вважаються товаришами і довіряють один одному. Відсутність захисних механізмів якнайкраще сприяє вразливості мережі.

Найкращий спосіб захисту від черв'яка – вжиття заходів запобігання несанкціонованому доступу до мережі.

Отже, як віруси троянські коні і черв'яки на сьогоднішній день є однією із найнебезпечніших загроз комп'ютерній системі. Для захисту від цих різновидностей шкідливих програм необхідно створювати замкнене середовище виконання програм, розмежовувати доступ до виконуваних файлів, контролювати цілісність виконуваних файлів і системних областей, тестувати придбані програмні засоби.

2.8.6 Програми-зомбі

Зомбі – це програма, яка приховано під'єднується до інших підключених в Інтернет комп'ютерів, а потім використовує цей комп'ютер для запуску атак, що ускладнює відстеження шляхів до розробника або розповсюджувача програми-зомбі.

Зомбі використовують при атаках з відмовою в обслуговуванні, які зазвичай направляють проти Web-вузлів. Зомбі розповсюджуються на сотні комп'ютерів, що належать не підозрюючим нічого третім особам, а потім використовуються для зараження вибраного в якості мішені Web-вузла за допомогою сильно збільшеного мережного трафіка.

2.8.7 "Жадібні" програми (greedy program)

"Жадібні" програми (greedy program) – це програми, що намагаються монополізувати який-небудь ресурс, не даючи іншим програмам можливості використовувати його. Доступ таких програм до ресурсів системи призводить до порушення її доступності для інших програм. Безумовно, така атака буде активним втручанням у роботу системи. Безпосередній атаці в більшості випадків піддаються об'єкти системи: процесор, оперативна пам'ять, пристрої введення-виведення.

Багато комп'ютерів, особливо в дослідницьких центрах, мають фонові програми, які виконуються з низьким пріоритетом. Вони проводять великий обсяг обчислень, а результати їхньої роботи потрібні не так вже часто. Але при підвищенні пріоритету така програма може блокувати решту програм. Ось чому вона є "жадібною".

"Тупікова" ситуація виникає тоді, коли "жадібна" програма нескінченна (наприклад, виконує явно нескінченний цикл). Але в багатьох операційних системах існує можливість обмеження часу процесора, який використовується конкретною задачею. Це не стосується операцій, які виконуються залежно від інших програм, наприклад операцій введення-виведення, що закінчуються асинхронно до основної програми, оскільки час їх виконання не входить у час роботи програми. Перехоплюючи асинхронне повідомлення про закінчення операції введення-виведення і посилаючи знову запит на нове введення-виведення, можна досягти нескінченності програми. Такі атаки називають також асинхронними.

Другий приклад "жадібної" програми – програма, яка захоплює дуже велику ділянку оперативної пам'яті. В оперативній пам'яті послідовно розміщуються, наприклад, дані, які надходять із зовнішнього носія. Врешті-решт пам'ять може бути сконцентрована в одній програмі, і виконання інших стане неможливим.

2.8.8 Захоплювачі паролів (password grabber)

Захоплювачі паролів (password grabber) – це спеціально призначені програми для крадіжки паролів. Вони виводять на екран терміналу (один за одним): порожній екран, екран, який з'являється після катастрофи системи або сигналізує про закінчення сеансу роботи. При спробі входу імітується введення імені і пароля, які пересилаються власнику програми-захоплювача, після чого виводиться повідомлення про помилку введення і управління повертається операційній системі. Користувач думає, що зробив помилку при наборі пароля, повторює вхід і отримує доступ до системи. Отже, в результаті таких дій його ім'я і пароль стають відомими власнику програми-захоплювача.

Перехоплення пароля може здійснюватися й іншим способом – за допомогою впливу на програму, яка керує входом користувачів у систему, та її наборів даних.

Захоплення пароля є активним, безпосереднім впливом на комп'ютерну систему в цілому. Для запобігання цій загрозі перед входом в систему необхідно впевнитися, що вводиться ім'я і пароль саме системної програми входу, а не якої-небудь іншої. Крім того, необхідно суворо дотримуватися правил використання паролів і роботи з операційною системою. Слід зауважити, що більшість порушень здійснюється не через хитромудрі атаки, а через елементарну необережність.

Не слід вимикати комп'ютер, доки не будуть закриті всі робочі програми. Необхідно постійно перевіряти повідомлення про дату і час останнього входу і кількість помилкових входів. Ці прості дії допоможуть уникнути захоплення пароля.

Крім описаних вище, існують і інші можливості компрометації паро-

лів. Отже, слід дотримуватись правил, які рекомендуються для створення і використання паролів.

Не слід записувати команди, які містять пароль, у командні процедури, слід намагатись уникати явного повідомлення пароля при запитуванні доступу по мережі, оскільки ці ситуації можна простежити і захопити таким чином пароль. Не слід використовувати один і той самий пароль для доступу до різних вузлів комп'ютерної системи. Рекомендується частіше змінювати пароль. І взагалі, дотримання правил використання паролів – необхідна умова надійного захисту.

2.8.9 Утиліти схованого адміністрування (backdoor)

Цей вид шкідливого програмного забезпечення у деяких випадках також можна віднести до групи троянських коней. Вони за своєю суттю є досить могутніми утилітами віддаленого адміністрування комп'ютерів у мережі. За своєю функціональністю вони багато в чому нагадують різні системи адміністрування, розроблені і розповсюджені фірмами-розробниками програмних продуктів. Єдина особливість цих програм змушує класифікувати їх як шкідливі троянські програми – відсутність попередження про інсталяцію і запуск. Під час запуску троянська програма встановлює себе в системі і потім стежить за нею, при цьому користувачу не видається ніяких повідомлень про дії такого трояна в системі. Більш того, посилання на трояна може бути відсутнім у списку активних додатків. У результаті користувач цієї троянської програми може і не знати про її присутність у системі, у той час як його комп'ютер відкритий для віддаленого керування.

Будучи встановленими на комп'ютер, утиліти прихованого адміністрування дозволяють робити з комп'ютером усе, що в них заклав їх автор: приймати і відсилати файли, запускати і знищувати їх, виводити повідомлення, стирати інформацію, перевантажувати комп'ютер і т.д. У результаті ці трояни можуть бути використані для виявлення і передачі конфіденційної інформації, для запуску вірусів, знищення даних і т.п. У цьому випадку уражені комп'ютери виявляються відкритими для злочинних дій хакерів.

2.8.10 Intended-віруси

До intended-вірусів (intended - навмисний) відносяться програми, які на перший погляд є стовідсотковими вірусами, але не здатні розмножуватися через помилки. Наприклад, вірус, що при зараженні "забуває" помістити в початок файлів команду передачі керування на код вірусу, або записує в неї невірну адресу свого коду, або неправильно встановлює адресу перехоплюваного переривання (що в переважній більшості випадків "завішує" комп'ютер).

До категорії intended-вірусів також відносяться віруси, що за приведеними вище причинами розмножуються тільки один раз – з авторської копії. Заразивши будь-який файл, вони втрачають здатність до подальшого розмноження. З'являються intended-віруси найчастіше при невдалій перекомпіляції якогось існуючого вірусу, через недостатнє знання мови програмування, через незнання технічних тонкощів операційної системи.

2.8.11 Конструктори вирусів

Конструктор вірусів – це утиліта, призначена для виготовлення нових комп'ютерних вірусів. Відомі конструктори вірусів для DOS, Windows і макро-вірусів. Вони дозволяють генерувати вихідні тексти вірусів (ASM-файли), об'єктні модулі, і/або безпосередньо заражені файли.

Деякі конструктори (наприклад, VLC, NRLG) оздоблені стандартним віконним інтерфейсом, де за допомогою системи меню можна вибрати тип вірусу, об'єкти для зараження (COM і/або EXE), наявність або відсутність самошифрування, протидії налагоджувачу, внутрішні текстові рядки, вибрати ефекти, що супроводжують роботу вірусу і т.п.

Інші конструктори (наприклад, PS-MPC, G2) не мають інтерфейсу і зчитують інформацію про тип вірусу з конфігураційного файлу.

2.8.12 Поліморфні генератори

Поліморфні генератори (або поліморфік-генератори), як і конструкттори вірусів, не є вірусами в буквальному значенні цього слова, оскільки в їх алгоритми не закладаються функції розмноження, тобто відкриття, закриття і записування у файли, читання і записування секторів і т.д. Головною функцією подібного роду програм є шифрування тіла вірусу і генерація відповідного розшифровувача.

Звичайно поліморфні генератори поширюються авторами без обмежень у виді файла-архіву. Основним файлом в архіві будь-якого генератора є об'єктний модуль, що містить цей генератор. В усіх генераторах, що зустрічалися до цих пір, такий модуль містить зовнішню (external) функцію – виклик програми-генератора. Автору вірусу, якщо він бажає створити дійсний поліморфік-вирус, не приходиться “длубатися” над кодами власного зашифровувача/розшифровувача. При бажанні він може підключити до свого вірусу будь-який відомий поліморфік-генератор і викликати його з кодів вірусу. Фізично це досягається так: об'єктний файл вірусу лінкується з об'єктним файлом генератора, а у вихідний текст вірусу перед командами його запису у файл вставляється виклик поліморфік-генератора, що створює коди розшифровувача і шифрує тіло вірусу.

2.9 Програмні закладки і їх руйнуючий вплив

2.9.1 Поняття програмних закладок

З розвитком комп'ютерної техніки і засобів зв'язку, ускладненням процедур обміну інформацією і її обробки актуальним стає питання захисту інформаційних масивів, баз даних і програмних засобів від різних дій. Наприклад, для захисту від несанкціонованого доступу до інформації під час її передачі і зберігання використовуються криптографічні методи і, відповідно, засоби (програмні або апаратні) для їх реалізації, а для підтримки цілісності і авторизації повідомлень в електронному вигляді – системи цифрової аутентифікації (цифровий підпис). Важливим моментом при роботі засобів захисту є необхідність потенційного невтручання присутніх прикладних або системних програм в процес обробки інформації засобами захисту. Наведемо декілька прикладів.

Служба безпеки одного з комерційних банків зареєструвала дії, які могли бути виконані лише при знанні деякої конфіденційної інформації, яка зберігалася у вигляді бази даних в зашифрованому вигляді. Сумніватися в алгоритмі шифрування не доводилося – використовувалася утиліта DISKREET, що реалізовує Національний стандарт шифрування США (DES). Втрати паролів для шифрування також не було виявлено. Вивчення комп'ютерів виявило наявність в завантажувальних секторах ПЕОМ своєрідних вірусів – програм, які зберігали інформацію (у тому числі і паролі для шифрування), що вводиться з клавіатури, в декілька зарезервованих для цього секторів.

Через деякий час довелося зіткнутися ще з одним різновидом таких програм, також націлених на утиліту DISKREET. В цьому випадку програма асоціювалася з утилітою за принципом звичайного файлового вірусу. Програма ніяк не проявляла себе зовні, проте зберігала всі введення з клавіатури в прихованому файлі.

Такі програми фахівці відразу назвали закладкою – по аналогії з непомітно впроваджуваними в приміщення мініатюрними електронними системами звукового підслуховування або телевізійного спостереження. Надалі метким комп'ютерним зловмисникам вимагалося лише прочитати файл (або проглянути сектори), щоб отримати паролі і по них розшифрувати дані, що цікавили їх.

Одне з малих підприємств, яке займалось посередницькою діяльністю, і, як наслідок, володіло конфіденційною інформацією про предмети можливих операцій, також використовувало шифрування як засіб захисту своїх інтересів. В даному випадку потерпілим був Російський стандарт ГОСТ 28147-89. Для шифрування використовувалася плата Krypton-3, що реалізовує даний алгоритм шифрування (який, як відомо, забезпечує гарантований захист інформації). Через деякий час з'ясувалося,

що шифрована інформація стає відомою третій стороні. А ще через деякий час була виявлена впроваджена в систему закладка, що підмінила собою (з погляду прикладних програм шифрування файлів) плату шифрування. При цьому алгоритм ГОСТ був замінений іншим, вкрай простим і легко зчитуваним без ключа.

Багатьом пам'ятна суперечка супротивників і прихильників програми Pretty Good Privacy (PGP), яка була завершена написанням закладки, що підробляє електронний підпис під файлами, що виконується даною програмою.

При розгляді дії закладки і програм захисту інформації доречні аналогії з взаємодією вірусу і прикладної програми. Вірус може приєднатися до виконуваного файлу, відповідним чином змінивши його, може знищити деякі файли або вбудуватися в ланцюжок драйверів. Закладка відрізняється більш направленою і тонкою дією. Проте ясно, що і вірус, і закладка повинні приховувати свою присутність в операційному середовищі комп'ютерної системи. Особливістю закладок може бути і те, що вони фактично стають невіддільними від прикладних або системних програм, якщо впроваджені в них на стадії розробки або шляхом зворотного проектування (шляхом дизасемблювання прикладної програми, впровадження коду закладки і подальшої компіляції).

2.9.2 Моделі поведінки програмних закладок

Нагадаємо, що під несанкціонованим доступом до ресурсів комп'ютерної системи розуміють дії по використанню, зміні і знищенню виконуваних модулів і масивів даних вказаної системи, вироблювані суб'єктом (зловмисником), що не має права на такі дії. Якщо комп'ютерна система містить механізми захисту від НСД, то несанкціоновані дії можуть бути викликані наступними основними причинами:

- відключення або зміна захисних механізмів нелегальним користувачем;
- вхід в систему під ім'ям і з повноваженнями реального користувача.

У першому випадку зловмисник повинен видозмінити захисні механізми в системі (наприклад, відключити програму запитів паролів користувачів), в другому – яким-небудь чином з'ясувати або підроблювати ідентифікатор реального користувача (наприклад, підглянути пароль, що вводиться з клавіатури). В обох випадках НСД можна представити моделлю опосередкованого доступу – коли проникнення в систему здійснюється на основі деякої дії, здійснюваної заздалегідь впровадженою в систему програмою або декількома програмами. Наприклад, зловмисник користується інформацією, яка витягнута з деякого масиву даних, створеного роботою програмного засобу зловмисника сумісно з системою перевірки прав доступу і надання цих прав (заздалегідь впроваджена в систему

програма при здійсненні доступу легального користувача запам'ятає його пароль і збереже в наперед відомому доступному зловмиснику файлі, а потім нелегальний користувач скористається даним паролем для входу в систему), або зловмисник змінить частину системи захисту так, щоб вона перестала виконувати свої функції (наприклад, змінить програму шифрування вручну або за допомогою деякої іншої програми так, щоб вона перестала шифрувати або змінила алгоритм шифрування на простішій).

Таким чином, поняття закладки в основному пов'язане з розробкою програмного забезпечення, а конкретно, з написанням початкових текстів програм, в яких створюються додаткові функції (логічна бомба, логічний люк, троянський кінь тощо).

Отже, закладка може бути внутрішнім об'єктом захищеної системи, а може бути і зовнішнім по відношенню до захищеної системи об'єктом.

2.9.3 Основні принципи роботи закладок та їх класифікація

Програмою з потенційно небезпечними наслідками назвемо деяку самостійну програму (набір інструкцій), яка здатна виконувати будь-яку непорожню підмножину перерахованих функцій:

- приховувати ознаки своєї присутності в програмному середовищі ПЕВМ;
- володіти здібністю до самодублювання, асоціюванню себе з іншими програмами і/або перенесенню своїх фрагментів в інші області оперативної або зовнішньої пам'яті;
- руйнувати (спотворювати довільним чином) код програм в оперативній пам'яті;
- зберігати фрагменти інформації з оперативної пам'яті в деяких областях зовнішньої пам'яті прямого доступу (локальних або віддалених);
- спотворювати довільним чином, блокувати і/або підміняти той, що виводиться в зовнішню пам'ять або в канал зв'язку масив інформації, що утворився в результаті роботи прикладных програм, або масиви даних, що вже знаходяться в зовнішній пам'яті.

Самодублювання програми з потенційно небезпечними наслідками – процес відтворення свого власного коду в оперативній або зовнішній пам'яті ПЕОМ.

Асоціювання з іншою програмою – інтеграція свого коду, або його частини в код іншої програми так, щоб за деяких умов управління передавалося на код програми з потенційно небезпечними наслідками.

Програми з потенційно небезпечними наслідками можна умовно розділити на наступні *класи*:

- класичні програми-віруси. Особливістю даного класу є його неспрямованість на конкретні програми і також те, що основною зада-

чею ставиться самодублювання вірусу. Руйнування інформації вірусом не направлене на конкретні програми і зустрічається не більше ніж у 10% такого роду програм;

- програми типу програмний черв'як або троянський кінь і фрагменти програм типу логічний люк. В даному випадку має місце зворотна ситуація – самодублювання не властиве такого роду програмам, але виявляються можливості збереження конфіденційної інформації або витягання інформації з сегментів систем безпеки або обмеження доступу;
- програмні закладки або руйнуючі програмні дії – клас програм з потенційно небезпечними наслідками, обов'язково виконуючий попередньо вказані дії.

Крім того, програмні закладки можна класифікувати *по методу і місцю їх упровадження і застосування* (за способом доставки в систему):

- закладки, асоційовані з програмно-апаратним середовищем (BIOS);
- закладки, асоційовані з програмами первинного завантаження (що знаходяться в MASTER BOOT RECORD або BOOT-секторах активних розділів);
- закладки, асоційовані з завантаженням драйверів DOS, командного інтерпретатора, мережевих драйверів, тобто з завантаженням операційного середовища;
- закладки, асоційовані з прикладним програмним забезпеченням загального призначення (вбудовані в клавіатурні і екранні драйвери, програми тестування комп’ютера, утиліти і оболонки типа NORTON);
- виконувані модулі, що містять тільки код закладки (як правило, впроваджувані в пакетні файли типу .BAT);
- модулі-імітатори, що на вигляд співпадають з деякими програмами, що вимагають введення конфіденційної інформації;
- закладки, масковані під програмні засоби оптимізаційного призначення (архіватори, прискорювачі і т. д.);
- закладки, масковані під програмні засоби ігрового і розважального призначення (як правило, використовуються для первинного впровадження закладок типу “досліджуваач”).

Як бачимо, програмні закладки мають багато загального з класичними вірусами, особливо в частині асоціювання себе з виконуваним кодом (завантажувальні віруси, віруси-драйвери, файлові віруси).

Крім того, програмні закладки, як і багато відомих вірусів класичного типу, мають розвинені засоби боротьби з налагоджувачами і дизасемблерами.

Контрольні питання

1. В чому полягає несанкціонований доступ і як він реалізується?
2. Охарактеризуйте незаконне використання привілеї як одну з поширених загроз комп'ютерній системі.
3. Що таке атаки "сілямі" і за яких умов вони можуть відбутися?
4. Що таке "приховані канали", які їх види ви знаєте? В чому полягає принцип їх функціонування?
5. Що таке "маскарад" і як можна запобігти таким атакам?
6. Чому "збір сміття" можна використати як атаку на комп'ютерну систему, які механізми використовуються для захисту від них?
7. В чому полягає "зламування" системі і яким чином воно може бути реалізовано?
8. Що відносять до шкідливого програмного забезпечення і як його класифікують?
9. Що таке люки, звідки вони з'являються і як запобігти їх виникненню?
10. Охарактеризуйте логічні бомби як шкідливе програмне забезпечення. Наведіть приклади.
11. Яке шкідливе програмне забезпечення відносять до програм-"троянів" і в чому їх особливості?
12. Охарактеризуйте коротко черв'яки, "зомбі" та "жадібні програми" як шкідливе програмне забезпечення.
13. Що являють собою програми-захоплювачі паролів і як можна запобігти їх шкідливому функціонуванню?
14. В чому суть утиліт прихованого адміністрування? Навести приклади відомих програмних продуктів цього напрямку.
15. Для чого існують і як функціонують конструктори вірусів і поліморфік-генератори?

3 ВІРУСИ ЯК ШКІДЛИВЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

3.1 Класифікація комп'ютерних вірусів

Віруси можна розділити на класи за такими ознаками:

- за середовищем існування;
- за способом зараження;
- за особливостями використовуваних алгоритмів;
- за деструктивними можливостями.

3.1.1 Віруси за середовищем їх існування

За цією ознакою віруси поділяються на:

- файлові віруси;
- завантажувальні віруси;
- макро-віруси;
- мережні віруси.

Файлові віруси діють одним з таких способів:

- впроваджуються в основному у виконувані файли, тобто у файли з розширеннями COM та EXE. Вони можуть впроваджуватись і у файли інших типів, але в такому випадку, як правило, вони ніколи не отримують управління, і, як наслідок, втрачають здатність до розмноження;
- створюють файли-двійники (компаньйон-віруси);
- використовують особливості організації файлової системи (Link-віруси).

Бутовські (завантажувальні) віруси діють такими методами:

- впроваджуються в завантажувальний сектор диска (надалі Boot-сектор або бут-сектор);
- впроваджуються в сектор, що містить програму завантаження системного диска (Master Boot Record – MBR);
- змінюють покажчик на активний boot-сектор.

Існують також файлово-бутівські віруси, що заражають і файли, і завантажувальні сектори дисків.

Макро-віруси заражають файли-документи і електроні таблиці відомих програмних продуктів.

Мережні віруси використовують для свого розповсюдження протоколи або команди комп’ютерних мереж та електронної пошти.

3.1.2 Віруси за способом зараження

Резидентні віруси при зараженні (інфікуванні) комп'ютера залишають в оперативній пам'яті свою резидентну частину, яка потім перехоплює звернення операційної системи до об'єктів зараження (файлів, завантажувальних секторів тощо) і впроваджується в них. Резидентні віруси знаходяться в пам'яті і є активними до самого вимкнення комп'ютера або до його перевантаження.

Нерезидентні віруси не заражають пам'ять комп'ютера і є активними лише обмежений проміжок часу. Деякі віруси лишають в пам'яті невеликі резидентні програми, які не розповсюджують вірус. Такі віруси вважаються нерезидентними.

3.1.3 Віруси за особливостями використовуваних алгоритмів

Прості віруси – це віруси-паразити, вони змінюють вміст файлів і секторів дисків і можуть бути досить легко виявлені та знешкоджені.

Стелс-віруси (або віруси-невидимки) – це віруси, які повністю або частково приховують себе в системі. Найбільш розповсюдженим стелс-алгоритмом є перехоплення запитів ОС на читання-записування заражених об'єктів. Стелс-віруси при цьому або тимчасово виліковують їх, або підставляють замість себе незаражені ділянки інформації. Це можуть бути і макро-віруси, які можуть забороняти виклики меню перегляду макросів. Це можуть бути і файлові (наприклад, вірус "Frodo"), і бутовські (вірус "Brain") стелс-віруси.

Віруси-мутанти, які можуть використовувати алгоритми шифровки-розшифровки та поліморфізму, завдяки яким копії одного і того самого вірусу не мають жодного ланцюжка байтів, що повторюються. Поліморфік-віруси досить важко виявити, вони не мають сигнатури, тобто не містять жодної сталої ділянки коду

3.1.4 Віруси за деструктивними можливостями

Опис деструктивних можливостей вірусу можна представити у такому розрізі:

- *нешкідливі віруси*, які ніяким чином не впливають на роботу комп'ютера, крім зменшення вільної пам'яті на диску в результаті свого поширення;
- *безпечні віруси*, вплив яких обмежується зменшенням вільної пам'яті на диску і графічними, звуковими та іншими ефектами;
- *небезпечні віруси*, робота яких може привести до серйозних збійних

ситуацій у роботі комп’ютера;
- *дуже небезпечні віруси*, активність яких може приводити до втрати програми, знищення даних, стирання необхідної для роботи комп’ютера інформації, яка записана в системних областях пам’яті, і навіть сприяти прискореному зносу рухомих частин механізмів, наприклад, головок вінчестера.

3.2 Систематизація комп’ютерних вірусів

Важливість і довгостроковий характер проблеми захисту від комп’ютерних вірусів на практиці не викликає сумніву. Для зручності ідентифікації було би зручно, щоб кожний вірус мав своє ім’я. На превеликий жаль, старі віруси у більшості випадків мають чимало назв, які виникли історично, а нові віруси, навпаки, таких назв взагалі не мають.

Відомим дослідником комп’ютерних вірусів М.М. Безрукавим було вироблено і запропоновано схему класифікації, яка включає три основні елементи:
- класифікаційний код вірусу;
- дескриптор вірусу (формалізований список основних властивостей);
- сигнатура вірусу (рядок для контекстного пошуку даного вірусу в зараженій програмі).

3.2.1 Класифікаційний код вірусу

Кожному вірусу присвоюється код, який складається з літерного префікса, кількісної характеристики і факультативного літерного суфікса.

Наприклад, в коді RCE-1813c є такі складові: *RCE* – префікс, *1813* – корінь (характеристика), а *c* – суфікс. Крім того, факультативне розширення, що записується в кінці коду через крапку, характеризує групу, до якої належить даний вірус. Наприклад, RCE-1813.IER означає, що даний вірус належить до єрусалимської групи.

Головною вимогою до класифікаційного коду вірусу є можливість визначення більшості вхідних його властивостей на незараженому комп'ютері. Виконання будь-яких дій з дослідження вірусу на зараженому комп'ютері є найбільшою й найбільш розповсюдженою помилкою, якої припускаються недосвідчені користувачі. Необхідно підкреслити, що будь-які дії на комп'ютері, зараженому невідомим вірусом, пов'язані з певним ризиком викликати спрацьовування троянської компоненти вірусу. Крім того, резидентний вірус з метою маскування може перехоплювати запити і перекручувати інформацію, яка видається. Нині відомі віруси, що мають таку властивість. Наприклад, група файлових вірусів, відома під назвою TP-вірусів, починаючи з вірусу TP-34 (члени цієї групи мають номери, які

зберігаються в передостанньому байті вірусу в шістнадцятковому вигляді), має властивість "самовикусування" при спробі трасувати заражену програму – резидентний вірус виконує "викусування" вірусу з програми, "підсовуючи" налагоджувачу вже вилiковану програму. Так само завантажувальні віруси, які входять у пакистанську групу (віруси “Brain”, “Ashar”), при спробі переглянути бут-сектор на зараженому комп'ютері "підсовують" користувачу оригінальний бут-сектор, який зберігається вірусом в одному з секторів, позначеному як дефектний (і, тим самим, вилученому з розподілу файлам).

Літерний префікс вказує на місце розміщення голови вірусу і складається з послідовності літер і цифр, що починається з великої літери. Відповідно до цього будемо розрізняти такі типи вірусів (розглядатимемо тільки реально існуючі типи, а не всі принципово можливі):

- *файлові* – коли голова вірусу розміщується в COM-, EXE-файлах і оверлеях (символи C, E в префіксі). При цьому додаткову літеру, яка відображає зараження оверлеїв, у префікс не вводиться, щоб запобігти його ускладненню, а виноситься у дескриптор;
- *бутові* – коли голова вірусу розміщується в бут-секторі або блоці MBR (символи B, R або M у префіксі);
- *пакетні* – коли голова вірусу розміщена в пакетному файлі, тобто являє собою фрагмент або програму на будь-якій мові програмування (префікс J).

Поряд із "чистими" вірусами, які використовують лише одне середовище, нині з'явилися "гібридні" – комбінація файлових і бутових вірусів. У таких вірусах замість першої літери R використовують відповідну літеру префікса бутового вірусу, наприклад BCE або MCE (як і бутові, змішані віруси не можуть бути нерезидентними).

Характеристика вірусу являє собою кількісно вимірювану властивість вірусу, яка допускає просте визначення і розрізняється для більшості типів вірусів. Наприклад, для файлових вірусів як характеристика може використовуватися величина приросту довжини файлів при зараженні ("інфекційна довжина"), хоч тут є певні складності.

Суфікс використовується, коли два різних віруси або два штами одного і того самого вірусу мають однаковий префікс і характеристику. У цьому випадку, для отримання унікального коду використовується як суфікс латинська літера. Наприклад, в коді RC-1704f літера f означає "штам-f”.

3.2.2 Дескриптор вірусу

Безумовно, запропонований код вірусу не охоплює та й не може охопити основні властивості вірусу. Водночас систематизація властивостей вірусу становить значний інтерес як для розробників антивірусних про-

грам, так і для їх користувачів, оскільки дозволяє інтегрувати різнорідні факти, які стосуються поведінки того чи іншого вірусу в системі, тим самим полегшуючи їх запам'ятовування і порівняння. Тому як другий елемент класифікації пропонується так званий дескриптор.

Дескриптор є систематизацією основних характеристик вірусу в закодованому вигляді. Кодування складається з груп символів, що починаються з великої латинської літери, за якою йдуть маленькі латинські літери або цифри. При цьому велика латинська літера визначає вид характеристики, а наступні за нею малі літери або цифри – значення характеристики для конкретного вірусу. Наприклад, в дескрипторі "Xab Yc Zdmt" є три властивості: X – зі значенням "ab", Y – зі значенням "c" і Z – зі значенням "dmt".

3.2.3 Сигнатура вірусу

Оскільки більшість відомих нині вірусів допускає детектування за допомогою контекстного пошуку, то однією з важливих задач класифікації є складання сигнатур.

Сигнатура – це список рядків для контекстного пошуку. Знання сигнатур вірусів дозволяє перевіряти нове програмне забезпечення на їх наявність, тим самим суттєво підвищуючи ступінь захищеності ЕОМ. Стандартизація сигнатур особливо важлива, коли вірус має багато штамів, оскільки формальні схеми, подібні описаним вище за допомогою класифікаційного коду і дескриптора, мають той недолік, що деякі штами не будуть розрізнятися в заданому просторі ознак. Водночас порівняно легко забезпечується унікальність сигнатури, у крайньому разі для більшості відомих вірусів, хоч існують віруси, які не вміщують жодної постійної сигнатури, тобто які не можна знайти за допомогою контекстного пошуку.

Сигнатури вірусів можуть бути представлені як текстовими рядками або регулярними виразами, так і ділянками програмного коду.

Хоч частіше всього в якості сигнатури використовуються тільки ***текстові рядки***, для них застосовуються і ***регулярні вирази***. Вони суттєво стійкіші до деяких мутацій і, крім того, при меншій довжині забезпечують кращу якість розпізнавання (менша кількість невірних спрацьовувань). Все це робить їх кращими за прості текстові рядки.

Очевидно, що сигнатура, яка відповідає ***ділянці з командами***, надійніша за сигнатури ділянки з даними, наприклад з текстовими рядками (останні можуть бути модифіковані). Тому вибір сигнатури доцільно робити на основі аналізу дизасембльованого коду вірусу.

Довжина сигнатури не повинна бути дуже великою, оскільки довгу сигнатуру важко вірно набрати вручну. В той же час недостатня довжина або вибір нехарактерних ділянок коду сигнатури викликатимуть багато неправильних спрацьовувань, що зовсім небажано. Правильна сигнатура не

повинна бути в жодній з найбільш розповсюджених в операційних системах службових програм, включаючи, безумовно, і самі компоненти операційної системи. Отже, для вибору сигнатури відповідно до вказаних вимог необхідно провести ряд експериментів, де самі сигнатури можуть бути предметом порівняння і аналізу.

На сьогоднішній день є програми, які забезпечують детектування вірусів шляхом пошуку в файлах відповідних рядків, і використані в них сигнатури природно "прийняти за основу". Найбільшу цінність становлять рядки, які використовуються у відомому закордонному детекторі Scan, оскільки нові версії цього детектора з'являються регулярно і охоплюють практично всі віруси, які з'являються за кордоном. З інших закордонних детекторів необхідно відзначити Virscan фірми IBM і TNTVirus фірми Carnel (Ізраїль). Прийнято називати рядок, який використовується детектором Scan, *M-сигнатурою*, рядок, який використовується Virscan, – *I-сигнатурою*, а рядок, який використовується TNTVirus, – *C-сигнатурою*.

Відзначимо, що сигнатур для ряду вірусів, розроблених у нашій країні, в існуючих версіях цих програм немає, а сигнатури для болгарських вірусів часто невдалі. У таких випадках використовуються сигнатури, які позначені буквою B (*B-сигнатури*), або так звані *J-сигнатури*. Останні являють собою початкові байти кода вірусу, тобто перші виконувані команди тіла вірусу. Досвід показує, що вони досить специфічні і в більшості випадків дозволяють відрізнити один вірус від іншого. При цьому для файлових вірусів, які дописують своє тіло в кінець файла, вважають, що J-сигнатура починається з байта, на який передає керування команда JMP. Крім того, в тілі деяких вірусів зустрічаються характерні текстові рядки. Такі рядки називають *T-сигнатурами* і використовують як допоміжні.

Необхідно відзначити, що контекстний пошук може використовуватися не тільки для пошуку заражених вірусом програм, але й для пошуку програм і файлів, які знищені або пошкоджені вірусом. Наприклад, вірус C-648.VEN при певних значеннях таймера замість зараження програми знищує її, записуючи в перші п'ять байтів рядок, який відповідає переходу на підпрограму перезавантаження BIOS. Для пошуку знищених цим вірусом програм можна використовувати рядок "EAFOFFOOFO". Аналогічно вірус RCE-1800.DAV знищує сектори на вінчестері, записуючи перші байти повідомлення "Eddie lives … somewhere in time". За цим повідомленням за допомогою Norton Utilites або інших програм можна виявити пошкоджені сектори і визначити, до яких файлів вони належать.

За наявності сигнатури перевірку зараженості файлів вірусом даного типу зручно виконувати, використовуючи спеціальні програми. Наприклад, вдалою є програма TBScan, яка здійснює пошук у каталогах або заданих його гілках. У випадку виявлення заражених програм доцільно додатково проконтролювати результати за допомогою, наприклад, Norton Utilites, оскільки для перегляду всіх файлів можна використовувати режим глобального пошуку на диску.

3.3 Файлові віруси

До даної групи відносяться віруси, що при своєму розмноженні тим чи іншим способом використовують файлову систему якої-небудь ОС.

Впровадження файлового вірусу можливе практично в усі виконувані файли усіх популярних операційних систем. На сьогоднішній день відомі віруси, що вражають всі типи виконуваних об'єктів: командні файли (BAT), драйвери (SYS, у тому числі спеціальні файли IO.SYS і MSDOS. SYS) і виконувані двійкові файли (EXE, COM). Існують віруси, що вражають файли інших операційних систем – Windows, OS/2, Macintosh, UNIX, включаючи VxD-драйвери Windows.

Існують віруси, які заражають файли, що містять вихідні тексти програм, бібліотечні чи об'єктні модулі. Можливий запис вірусу й у файли даних, але це може бути або в результаті помилки вірусу, або при прояві його агресивних властивостей. Макро-віруси також записують свій код у файли даних або у документи та електронні таблиці, однак ці віруси настільки специфічні, що винесені в окрему групу.

Файлові віруси є найпоширенішим типом комп'ютерних вірусів. Вони становлять близько 80% загальної кількості вірусів, відомих для комп'ютерів, які сумісні з IBM PC. Цей клас комп'ютерних вірусів має дуже високу інфікуючу спроможність. За відсутності протидії вони викликають справжні епідемії. Так, наприклад, відбулося з вірусом RCE-1813.IER, відомого під назвами Jerusalim (Єрусалим), Black Friday (Чорна п'ятниця).

Більшість розповсюджених файлових вірусів мають штами, які не дуже відрізняються від базової версії. Тому можна говорити про групи файлових вірусів і, відповідно, групові дескритори і групові сигнатури. Нині кількість виявлених у країнах СНД файлових вірусів перевищує кілька сотень, тому запам'ятовування їх класифікаційних кодів суттєво полегшується, якщо вони використовуються з розширенням, яке показує, до якої групи належить даний вірус.

3.3.1 Класифікаційний код файлового вірусу

Файлові віруси можна розділити на резидентні і нерезидентні, оскільки це в багатьох випадках визначає поведінку вірусу і насамперед його інфікуючу спроможність (резидентні віруси мають значно вищу інфікуючу спроможність порівняно з нерезидентними).

Класифікаційний код файлових резидентних вірусів починається з префікса R, наприклад R-1701.CAS.

Префікс файлового вірусу

Крім символа R, класифікаційний код файлового вірусу може вклю-

чати символи C і E або їх комбінацію. Як уже зазначалося, символи C і E визначають типи файлів, заражених даним вірусом. Наприклад, якщо резидентний вірус заражає COM- і EXE-файли, то його класифікаційний код матиме префікс RCE.

Кількісна характеристика

У якості кількісної характеристики можна використовувати:

- нормований приріст або інфективну довжину (infective length);
- довжину кода вірусу;
- приріст довжини будь-якого зараженого файла.

До об'єктивних властивостей файлових вірусів можна віднести насамперед приріст довжини файлів при зараженні. Цей приріст, який зумовлює наявність вірусу, можна використати для визначення його типу. Тут є дві основні проблеми. По-перше, величина приросту може варіюватися залежно від довжини зараженого файла (багато вірусів при дописуванні свого коду в кінець зараженого файла вирівнює своє тіло на найближчу адресу, кратну 16, тобто на межу параграфа). По-друге, величина приросту може не збігатися для COM- і EXE-файлів. Тому як кількісну характеристику частіше використовують нормований приріст – ***інфективну довжину,*** яка визначається за такими правилами:

1) для вірусів з префіксом C і CE (RC, RKCE) характеристика класифікаційного коду має дорівнювати мінімальному приросту довжини COM-файла (для вірусів типу C і CE) або EXE-файла (для вірусів типу E);
2) для вірусів, які не змінюють довжину файла, вказується нуль, а через дефіс дійсна довжина тіла вірусу, наприклад RC-O-346.LEH;
3) для вірусів, які маскують збільшення довжини файла на зараженій програмі, до характеристики, визначеної за першим правилом, зліва додається незначущий нуль (наприклад, RCE-02000. DAV).

Відзначимо, що запропонований першим правилом підхід дозволяє зняти вплив вирівнювання на межу параграфа для вірусів, які вирівнюють свій приріст вказаним способом. Крім того, для вірусів, які змінюють свій приріст визначеним способом, наприклад шляхом підгонки до величини, кратної 51, мінімальний приріст також дає можливість позбутися впливу вставних байтів (цей випадок можна розглядати як різновидність вирівнювання). І нарешті, для вірусів, які багато разів заражують один і той самий файл, використання мінімального приросту дозволяє звільнитися від впливу багаторазового зараження.

Для визначення інфективної довжини не треба буде проводити спеціальні експерименти із зараження файлів. Здебільшого її можна досить просто визначити, порівнявши прирости довжин двох або більше заражених файлів типу COM. Найчастіше файлові віруси заражають командний процесор MS-DOS (файл COMMAND.COM) і програми, назви яких знаходяться у файлі AUTOEXEC.BAT. При аналізі кількох заражених файлів можливі два найтиповіші (хоч і не єдино можливі) випадки.

Якщо прирости довжин двох або більше заражених файлів збігаються, а залишки від ділення довжин початкових файлів на 16 відрізняються один від одного, то, ймовірно, вірус не виконує вирівнювання свого коду на межу параграфа й інфективну довжину L даного вірусу можна дістати за формулою:

$$L = D - (16 - mod\ (LEN, 16)),$$

тобто відніманням із отриманого приросту D доповнення 16 залишку від ділення початкової довжини LEN файла на 16. Наприклад, файл COMMAND.COM, який файлові віруси здебільшого пошкоджують в числі перших, у найпоширеніших нині версіях M8-DOS має довжину 25307. При цьому залишок від ділення 25307 на 16 дорівнює 11 (mod(25307,16)=11). Очевидно, що доповнення до 16 дорівнює 5, і для вирівнювання на межу параграфа необхідна вставка п'яти додаткових байтів. У цьому випадку інфективна довжина буде на 5 менша, ніж приріст довжини файла COMMAND.COM. Перевагою прийнятого підходу є те, що, за окремим винятком (наприклад, вірус RCE-1813.IER), визначена таким чином інфективна довжина збігається з довжиною коду вірусу.

Як кількісна характеристика класифікаційного коду можуть застосовуватися й інші параметри. Найпоширенішими вважають такі два підходи.

Використання як кількісної характеристики ***довжини коду вірусу***, визначеної за константою, яка вміщується у фрагменті, що забезпечує дописування коду вірусу в заражений файл (цю константу можна порівняно легко визначити, аналізуючи дизасембльований код вірусу). Така характеристика є об'єктивною, тому її часто використовують розробники антивірусних програм, які досить добре володіють мовою асемблера. Але визначена так характеристика в ряді випадків не збігається зі значенням приросту довжин файлів, який спостерігається. Це знижує її цінність з погляду використання при спробі класифікації користувачем, який не володіє мовою асемблера, нового, ще невідомого йому вірусу. Наприклад, для згаданого вище єрусалимського вірусу довжина коду вірусу становить 1808 байтів, а приріст довжини при зараженні файлів типу COM - 1813 байтів, що пояснюється додатковим записуванням в кінець зараженого файла типу COM п'ятибайтної константи "Ms-Dos" (використовується як ознака зараженості файла).

Використання як кількісної характеристики ***приросту довжини якого-небудь конкретного файла***, отриманого в результаті його зараження. Цей дійсно зручний підхід втратив свою привабливість з появою ряду вірусів, які не заражають командний процесор, з розповсюдженням MS-DOS версій 4.0 і вище, в якій довжина файла COMMAND.COM становить 37637, з появою нових сучасних операційних систем.

3.3.2 Дескриптор файлового вірусу.

Для зручності сприйняття дескриптор вірусу розбивається на декілька складових, для яких використовуються такі позначення:

DM – головний дескриптор;

DP – предикат зараження (для зручності сприйняття записаний в близькій до алгебраїчної нотації);

DR (тільки для резидентних вірусів) – положення в оперативній пам'яті, реакція на "тепле" перезавантаження і розмір зайнятої пам'яті;

DH – перехоплювані переривання (в шістнадцятковій системі числення).

3.3.3 Сигнатура файлового вірусу

Як уже відзначалося, для сигнатур доцільно використовувати рядки в шістнадцятковій системі числення, які відповідають характерним послідовностям команд у тілі вірусу. Розміщення сигнатур підпорядковується такому правилу: якщо M-сигнатура входить у V-сигнатуру, то вона додається після V-сигнатури. Як уже відзначалося раніше, T-сигнатури існують не для всіх файлових вірусів. Однією із найзручніших сигнатур для файлових вірусів є J-сигнатура. Їх можна дуже швидко визначити за допомогою будь-якого налагоджувача (Debug, Turbo Debugger, AFD і т.д.). Користувачі, які не вміють працювати з налагоджувачами, можуть використовувати для визначення J-сигнатур програму "маскошукач", яка входить в пакет VL (непоганий детектор, заснований на контекстному пошуку заданих рядків).

Необхідно відзначити, що контекстний пошук можна використовувати не тільки для пошуку заражених вірусом програм, але й для пошуку програм і файлів, які знищені або пошкоджені вірусом. Наприклад, вірус C-648.VEN при певних значеннях таймера замість зараження програми знищує її, записуючи в перші 5 байтів рядок, який відповідає переходу на підпрограму перезавантаження BIOS.

3.3.4 Види файлових вірусів

За способом зараження файлів віруси поділяються на:

- overwriting-віруси;
- паразитичні (parasitic);
- компаньйони-віруси (companion);
- link-віруси;
- віруси-черв’яки;
- віруси, що заражають об'єктні модулі (OBJ), бібліотеки компіляторів (LIB) і вихідні тексти програм.

Overwriting - віруси

Даний метод зараження є найбільш простий: вірус записує свій код замість коду файлу, що заражається, знищуючи його вміст. Природно, що при цьому файл перестає працювати і не відновлюється. Такі віруси дуже швидко виявляють себе, оскількиопераційна система і додатки досить швидко перестають працювати.

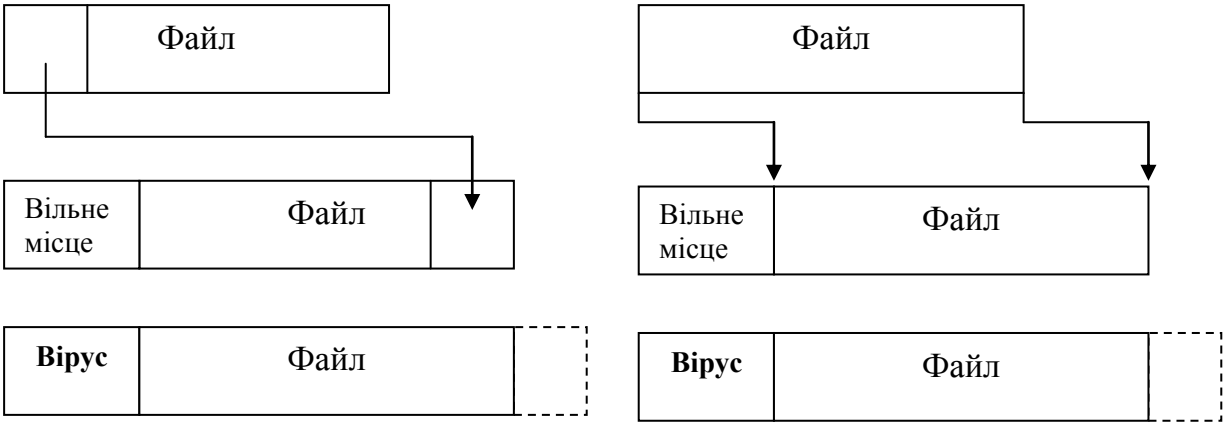
До різновиду overwriting-вірусів відносяться віруси, що записуються замість заголовка EXE-файлів. Основна частина файлу при цьому залишається без змін і продовжує нормально працювати у відповідній операційній системі, однак заголовок виявляється зіпсованим.

Parasitic-віруси

До паразитичних відносяться всі файлові віруси, які при поширенні своїх копій обов'язково змінюють вміст файлів, залишаючи самі файли при цьому цілком чи частково працездатними. Паразитичні віруси розділяються на три типи в залежності від того, куди вони записують своє тіло.

1. Впровадження вірусу ***в початок файлу*** ("prepending"-віруси) може відбуватися двома способами:

- вірус *переписує початок файлу, що заражається, у його кінець*, а сам копіюється в місце, що звільнилося (рис.6,а);
- вірус *створює в оперативній пам'яті свою копію*, дописує до неї файл, що заражається, і зберігає отриману конкатенацію на диск (рис.6,б).



а) *б)*

Рисунок 6 – Впровадження файлового вірусу в початок файла

Деякі віруси при цьому дописують у кінець файлу блок додаткової інформації (наприклад, вірус "Jerusalem" по цьому блоку відрізняє заражені файли від незаражених).

Впровадження вірусу в початок файлу застосовується в переважній більшості випадків при зараженні BAT- і COM-файлів MS DOS. Відомо кілька вірусів, що записують себе в початок EXE-файлів операційних

систем DOS, Windows і навіть Linux. При цьому віруси для збереження працездатності програми або лікують заражений файл, повторно запускають його, чекають закінчення його роботи і знову записуються в його початок (іноді для цього використовується тимчасовий файл, у який записується знешкоджений файл), або відновлюють код програми в пам'яті комп'ютера і надбудовують необхідні адреси в її тілі (тобто дублюють роботу ОС).

2. Впровадження вірусу ***в кінець файла*** ("appending"-віруси) – це найбільш розповсюджений спосіб упровадження вірусу у файл. При цьому вірус змінює початок файлу таким чином, що першими виконуваними командами програми є команди вірусу:

а) *у COM-файлі* в більшості випадків це досягається зміною його перших трьох (чи більш) байтів на коди інструкції JMP Loc_Virus (чи в більш загальному випадку – на коди програми, що передає керування на тіло вірусу) (рис.7,а);

б) *у заголовку EXE-файла* змінюються значення стартової адреси (CSIP), кількість секцій у файлі, характеристики секцій, довжина виконуваного модуля (файла), рідше – регістри-покажчики на стек (SSSP), контрольна сума файла і т.д. (рис.7,б);

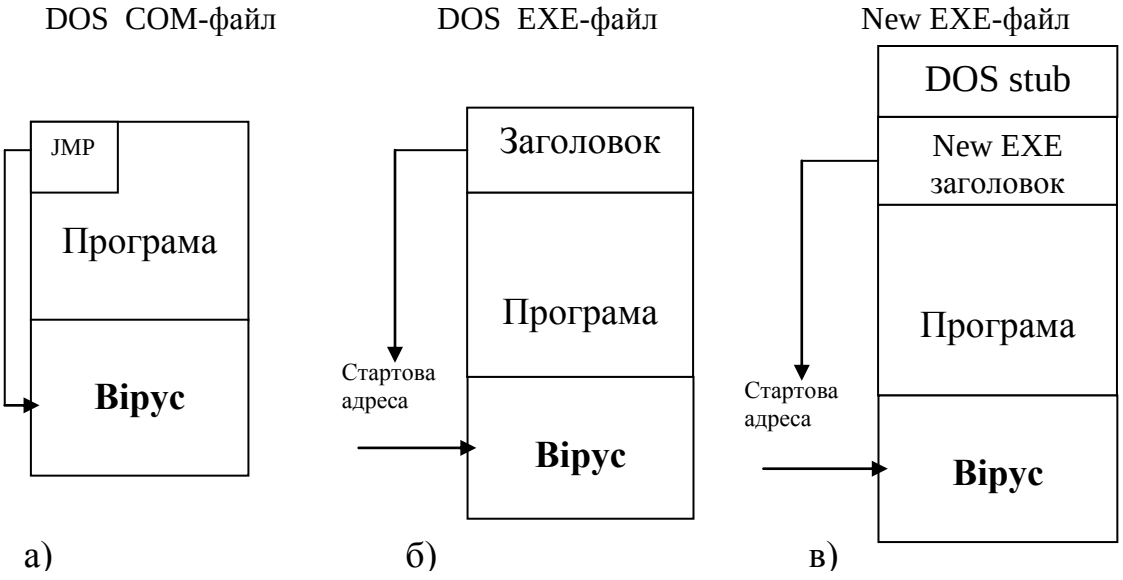


Рисунок 7 – Впровадження вірусу в кінець файла

в) *у виконуваних файлах Windows і OS/2* (NewEXE – NE, PE, LE, LX) змінюються поля в NewEXE-заголовку. Структура цього заголовка значно складніша за заголовок DOS EXE-файлів, тому зміні підлягає більше число полів – значення стартової адреси, кількість секцій у файлі, характеристики секцій і т.д. (рис.7,в). Додатково до цього довжини файлів перед зараженням можуть збільшуватися до значення, кратного параграфу (16 байт) в DOS або секції в Windows і OS/2 (розмір секції залежить від параметрів заголовка EXE-файлу);

г) віруси, що впроваджуються у *SYS-файли*, приписують свої коди до тіла файла і модифікують адреси програм стратегії (Strategy) і переривання (Interrupt) драйвера, що заражається (зустрічаються віруси, що змінюють адресу тільки однієї з цих програм). При ініціюванні зараженого драйвера вірус перехоплює відповідний запит ОС, передає його драйверу, чекає відповіді на цей запит, коректує його і залишається разом із драйвером в одному блоці оперативної пам'яті. Такий вірус може бути надзвичайно небезпечним і живучим, він впроваджується в оперативну пам'ять при завантаженні ОС раніш за будь-яку антивірусну програму, якщо вона теж не є драйвером (рис.8,а). Але існують віруси, які заражають системні драйвери іншим способом: вірус модифікує заголовок драйвера так, що DOS розглядає інфікований файл як ланцюжок з двох (або більше) драйверів (рис.8,б).

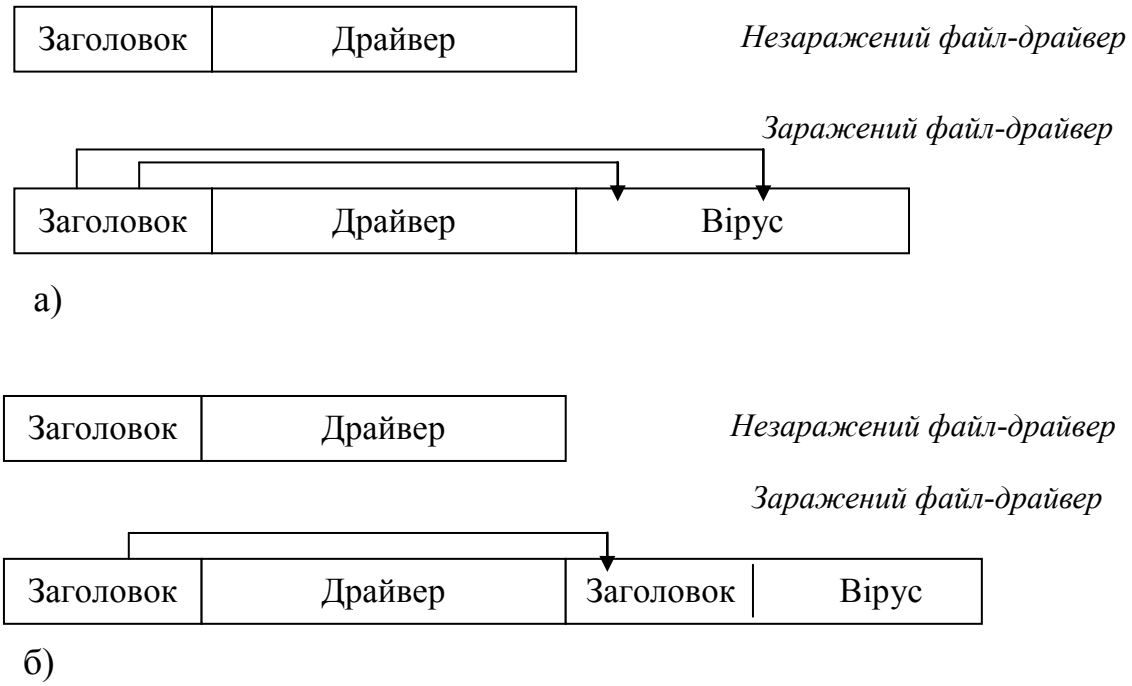


Рисунок 8 – Впровадження вірусу у SYS-файли

3. Впровадження вірусу ***в середину файла*** ("inserting"-віруси) може здійснюватись кількома методами:

а) вірус *переносить частину файлу в його кінець* або "розсовує" файл і записує свій код у простір, що звільнився. Цей спосіб багато в чому аналогічний методам, перерахованим вище. Деякі віруси при цьому компресують перенесений блок файлу так, що довжина файлу при зараженні е змінюється (вірус "Mutant");

б) метод "cavity", при якому вірус *записується у свідомо невикористовувані області файла*. Вірус може бути скопійований у незадіяні області таблиці настроювання адрес EXE-файла (вірус "BootExe"), у заголовок EXE-файлу, в область стека файла COMMAND.COM, в

область текстових повідомлень популярних компіляторів. Існують віруси, що заражають тільки ті файли, що містять блоки, заповнені яким-небудь постійним байтом, при цьому вірус записує свій код замість такого блоку;

в) копіювання вірусу в середину файлу може відбутися в результаті помилки вірусу, у цьому випадку файл може бути зіпсований.

4. Віруси ***без точки входу***. Окремо слід зазначити досить незначну групу вірусів, що не має "точки входу" (EPO-віруси – Entry Point Obscuring viruses). До них відносяться віруси, що не записують команд передачі керування в заголовок COM-файлів (JMP) і не змінюють адресу точки старту в заголовку EXE-файлів. Такі віруси записують команду переходу на свій код у будь-яке місце в середину файлу й одержують керування не безпосередньо під час запуску зараженого файлу, а під час виклику процедури, що містить код передачі керування на тіло вірусу. Причому виконуватися ця процедура може вкрай рідко (наприклад, при виведенні повідомлення про якусь специфічну помилку). Тому вірус може довгі роки "спати" усередині файлу і “вискочити на волю” тільки при деяких обмежених умовах.

Перед тим, як записати в середину файлу команду переходу на свій код, вірусу необхідно вибрати "правильну" адресу у файлі – інакше заражений файл може виявитися зіпсованим. Відомі кілька способів, за допомогою яких віруси визначають такі адреси усередині файлів.

Перший спосіб – пошук у файлі послідовності стандартного коду (віруси "Lucretia", "Zhengxi"). Ці віруси шукають у заражуваних файлах стандартні заголовки процедур C/Pascal і пишуть замість них свій код.

Другий спосіб – трасування чи дизасемблювання коду файлу (віруси "CNTV", "MidInfector", "NexivDer"). Такі віруси завантажують файл у пам'ять, потім трасують чи дизасемблюють його й у залежності від різних умов вибирають команду (чи команди), замість яких записується код переходу на тіло вірусу.

Третій спосіб застосовується тільки резидентними вірусами – при запуску файлу вони контролюють будь-яке переривання (частіше – INT 21h). Як тільки файл, що тільки заражається, викликає це переривання, вірус записує свій код замість команди виклику переривання (віруси "Avatar.Positron", "Markiz").

Четвертий спосіб базується на так званих настроюваннях коду програми. Таблиця настроювань (relocation table) в EXE-файлах вказує на адреси в тілі програми, які при завантаженні програми повинні бути приведені у відповідність до реальних адрес пам'яті. Зазвичай адреси, що надбудовуються, містять асемблерні інструкції з обмеженого набору. Вірус може легко ідентифікувати конкретну інструкцію, замінити її на виклик свого коду JMP_Virus і занулити відповідний запис у таблиці настроювань (щоб команда JMP_Virus не виявилася зіпсованою при завантаженні файла в пам'ять).

Компаньйон-віруси

До категорії "компаньйон" відносяться віруси, які не змінюють файлів, що заражаються. Алгоритм роботи цих вірусів полягає в тому, що для файла, який заражається, створюється файл-двійник, причому при запуску зараженого файлу керування одержує саме цей двійник, тобто вірус.

1. Найбільш поширені компаньйон-віруси, що використовують *особливість DOS першим виконувати .COM-файл,* якщо в одному каталозі присутні два файли з тим самим ім'ям, але різними расширеннями імені – .COM і .EXE. Такі віруси створюють для EXE-файлів файли-супутники, що мають те ж саме ім'я, але з розширенням .COM, наприклад, для файлу XCOPY.EXE створюється файл XCOPY.COM. Вірус записується в COM-файл і ніяк не змінює EXE-файл. При запуску такого файлу DOS першим знайде і виконає COM-файл, тобто вірус, який потім запустить і EXE-файл. Деякі віруси використовують не тільки варіант COM-EXE, але також і BAT-COM-EXE.
2. Другу групу складають віруси, що при зараженні *перейменовують файл у яке-небудь інше ім'я*, запам'ятовують його (для наступного запуску файла-хазяїна) і записують свій код на диск під іменем файла, що заражається. Наприклад, XCOPY.EXE перейменовується в XCOPY.EXD, а вірус записується під ім'ям XCOPY.EXE. При запуску керування одержує код вірусу, що потім запускає оригінальний XCOPY, що зберігається під ім'ям XCOPY.EXD. Цікавий той факт, що даний метод працює, напевно, у всіх операційних системах: в DOS, в Windows і OS/2.
3. У третю групу входять так називані "Path-companion" віруси, що *"грають" на особливостях PATH.* Вони або записують свій код під іменем зараженого файлу, але "вище" на один рівень PATH (ОС, таким чином, першим знайде і запустить файл-вірус), або переносять файл-жертву на один підкаталог вище і т.д.

Можливе існування й інших типи компаньйонів-вірусів, що використовують інші оригінальні чи ідеї особливості інших операційних систем.

Link-віруси

Link-віруси, як і компаньйон-віруси не змінюють фізичного вмісту файлів, однак під час запуску зараженого файла "змушують" ОС виконати свій код. Цієї мети вони досягають модифікацією необхідних полів файлової системи. На сьогоднішній день відомий єдиний тип Link-вірусів – віруси сімейства "Dir_II". При зараженні системи вони записують своє тіло в останній кластер логічного диска. При зараженні файлу віруси коректують лише номер першого кластера файла, розташований у відповідному секторі каталогу. Новий початковий кластер файла буде вказувати на кластер, що містить тіло вірусу. Отже, при зараженні файлів їх довжини і вміст кластерів диска, що містять ці файли, не змінюються, а на всі заражені файли на одному логічному диску буде приходитися тільки одна копія вірусу.

Таким чином, до зараження дані каталогу зберігають адресу першого кластера файлу (рис.9,а), а після зараження дані каталогу вказують на вірус, тобто при запуску файлу керування одержують не самі файли, а вірус (рис.9,б).

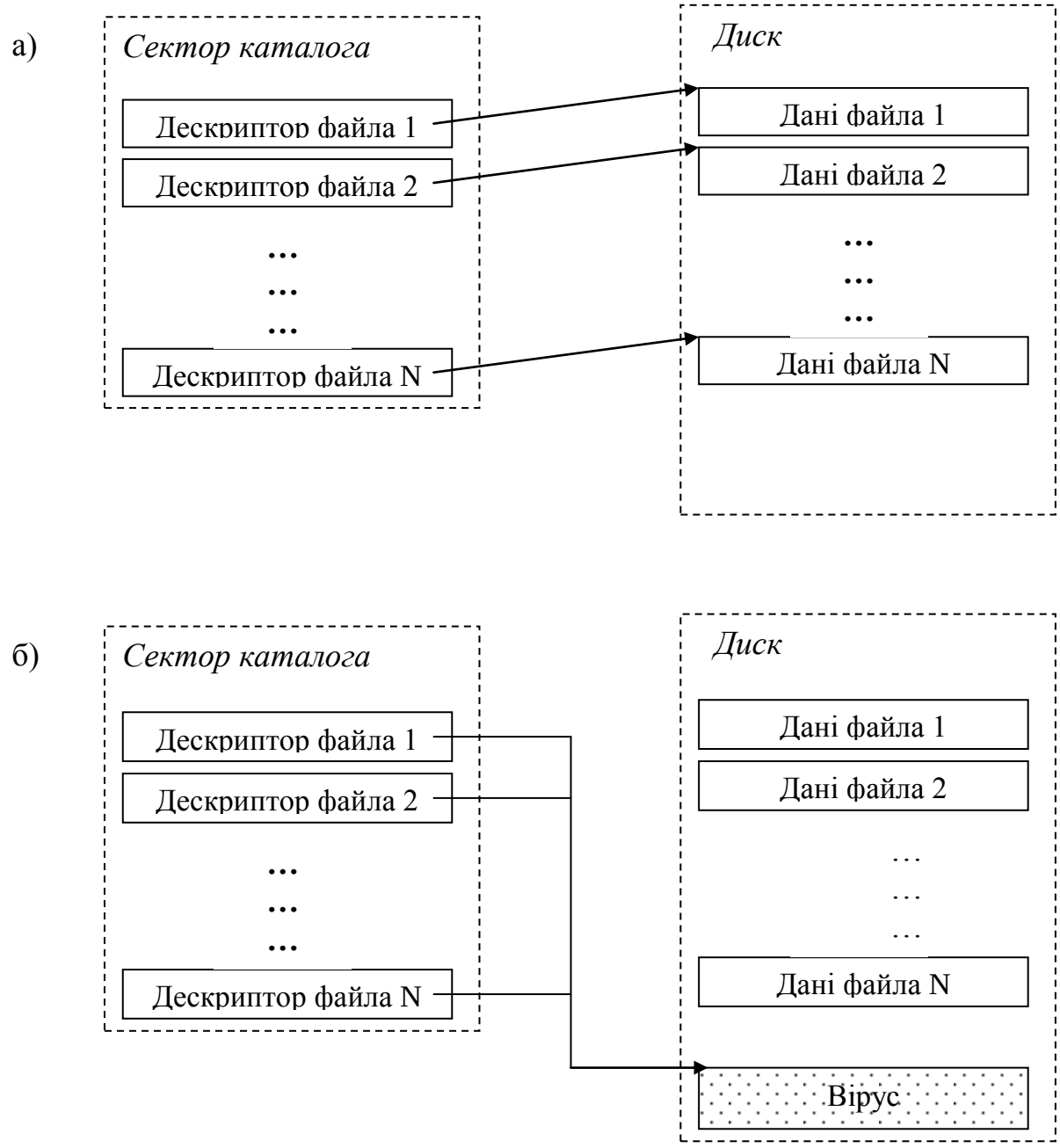


Рисунок 9 – Схема роботи Link-віруса

Файлові черв'яки

Файлові черв'яки (worms) є, у деякому сенсі, різновидом компаньйон-вірусів, але при цьому ніяким чином не пов'язують свою присутність з якимось виконуваним файлом. При розмноженні вони лише копіюють свій код у які-небудь каталоги дисків у надії, що ці нові копії будуть колись запущені користувачем. Іноді ці віруси дають своїм копіям спеціальні

імена, щоб підштовхнути користувача на запуск своєї копії, наприклад, INSTALL.EXE чи WINSTART.BAT.

Існують віруси-черв’яки, що використовують досить незвичайні прийоми, наприклад, записують свої копії в архіви (ARJ, ZIP та ін.). До таких вірусів відносяться "ArjVirus" і "Winstart". Деякі віруси записують команду запуску зараженого файлу в BAT-файли (наприклад, "Worm.Info").

Не слід плутати файлові віруси-черв’яки з мережними черв’яками. Перші використовують тільки файлові функції якої-небудь ОС, другі ж при своєму розмноженні користуються мережними протоколами.

OBJ-, LIB-віруси і віруси у вихідних текстах

Віруси, що заражають бібліотеки компіляторів, об'єктні модулі і вихідні тексти програм досить екзотичні і практично не поширені. Усього їх біля десятка.

Віруси, що заражають OBJ- і LIB-файли, записують у них свій код у форматі об'єктного модуля чи бібліотеки. Заражений файл, таким чином, не є виконуваним і нездатний на подальше поширення вірусу у своєму поточному стані. Носієм же "живого" вірусу стає COM- чи EXE-файл, одержуваний у процесі лінковки зараженого OBJ/LIB-файлу з іншими об'єктними модулями і бібліотеками. Таким чином, вірус поширюється в два етапи: на першому заражаються OBJ/LIB-файли, на другому етапі (лінковка) виходить працездатний вірус.

Зараження вихідних текстів програм є логічним продовженням попереднього методу розмноження. При цьому вірус додає до вихідних текстів свій вихідний код (у цьому випадку вірус повинен містити його у своєму тілі) чи свій шістнадцятковий дамп (що технічно легше). Заражений файл здатний на подальше поширення вірусу тільки після компіляції і лінковки (віруси "SrcVir", "Urphin").

3.3.5 Алгоритм роботи файлового вірусу

Одержавши керування, вірус здійснює такі дії (приведений список найбільш загальних дій вірусу при його виконанні, і для конкретного вірусу список може бути доповнений, скорочений, деякі пункти можуть помінятися місцями і значно розширитися):

1) резидентний вірус перевіряє оперативну пам'ять на наявність своєї копії і інфікує пам'ять комп'ютера, якщо копія вірусу не знайдена. Нерезидентний вірус шукає незаражені файли в поточному і\або кореневому каталозі, у каталогах, відзначених командою PATH, сканує дерево каталогів логічних дисків, а потім заражає виявлені файли;
2) виконує, якщо вони є, додаткові функції, деструктивні дії, графічні чи звукові ефекти і т.д. Додаткові функції резидентного вірусу можуть викликатися через деякий час після активізації в залежності від поточного

часу, конфігурації системи, внутрішніх лічильників чи вірусу інших умов; у цьому випадку вірус при активізації обробляє стан системного годинника, встановлює свої лічильники і т.д.;

3) повертає керування основній програмі (якщо вона є). Паразитичні віруси при цьому або лікують файл, виконують його, а потім знову заражають, або відновлюють програму (але не файл) у вихідному виді (наприклад, у COM-програм відновлюються декілька перших байтів, у EXE-програми обчислюється справжня стартова адреса, у драйвера відновлюються значення адрес програм стратегії і переривання). Компаньйони-віруси запускають на виконання свого "хазяїна", віруси-черв'яки і overwriting-віруси повертають керування ОС.

Метод відновлення програми у первісному вигляді залежить від способу зараження файлу.

Якщо вірус впроваджується в початок файла, то він або зрушує коди зараженої програми на число байтів, рівне довжині вірусу, або переміщає частину коду програми з її кінця в початок, або відновлює файл на диску, а потім запускає його.

Якщо вірус записався в кінець файла, то при відновленні програми він використовує інформацію, збережену у своєму тілі при зараженні файла. Це може бути довжина файла, декілька байтів початку файлу у випадку COM-файлу або декілька байтів заголовка у випадку EXE-файлу.

Якщо ж вірус записується в середину файла спеціальним чином, то при відновленні файлу він використовує ще і спеціальні алгоритми.

3.3.6 Приклади файлових вірусів

Abba.9849

Безпечний резидентний вірус. Перехоплює INT 21h і записується в кінець COM- і EXE-файлів при їх запуску. Містить рядки

```
\COMMAND.COM
Program too big to fit in memory
\ABBAл|*.*     E\ABBAл|
```

Створює на поточному диску файли ABBAл|nn з атрибутами HIDDEN і READONLY, 'nn' - число файлів, заражених на цьому диску. Це число збільшується при зараженні чергового файлу – вірус перейменовує цей файл в ABBAл|(nn+1). Залежно від числа nn вірус проявляє себе якимсь відео-ефектом на відео карті Hercules.

Lenin 943

Безпечний нерезидентний вірус. При запуску шукає EXE-файли і записується в їх кінець. При зараженні не змінює значення регістрів в EXE-заголовку, а вставляє в точку входу у файл команду CALL FAR virus і коректує Relocation Table. Залежно від своїх лічильників виводить тексти

САМЫЙ! ЧЕЛОВЕЧНЫЙ! ЧЕЛОВЕК!
Ленин и сегодня всех живых держит мертвой хваткой упыря

Також містить рядки

*.EXE PATH=

Metall.557

Дуже небезпечний нерезидентний вірус. Шукає .COM-файли (окрім COMMAND.COM) і записується в їх кінець. Коректно заражає тільки файли, на початку яких присутня команда JMP/CALL NEAR. Решта файлів після зараження виявляється зіпсованими. Залежно від системного таймера перемішує символи на екрані. Містить рядок: METALL\I

Scorpion.2278

Дуже небезпечний нерезидентний зашифрований вірус. При запуску заражає файл C\COMMAND.COM, потім шукає COM- і EXE-файли і записується в їх кінець. При зараженні COMMAND.COM записується в кінець файла в область стека COMMAND.COM і, таким чином, не збільшує його довжину. Знищує файли з ім'ям CHKLIST.MS. В деяких випадках також шукає інші файли і знищує їх. Залежно від системної дати і встановленого BIOS'a форматує вінчестер, виводить текст:

DEATH ON TWO LEGS V2.8
(c) BLACK SCORPiON, 1996
Written in Moscow

потім перехоплює INT 1Ch і програє мелодію. Вірус також містить рядки:

. *.EXE *.COM
C\COMMAND.COM
DEATH ON TWO LEGS WAS HERE

Sisters.2221

Дуже небезпечний резидентний зашифрований вірус. Перехоплює INT 21h, 16h і записується в кінець COM- і EXE-файлів при їх запуску. Знищує антивірусні файли даних CHKLIST.MS і CHKLIST.CPS.

Залежно від значень внутрішнього лічильника і поточної дати вірус відключає драйвер миші, стирає 40 секторів на диску C, видаляє CMOS пам'ять, завішує комп'ютер, виводить повідомлення:

TEMPLE OF LOVE V1.0 MS 95.
FoUnD VIRUS SYSTERS OF MERCY iN yOuR sYsTeM !!!

Вірусний обробник INT 16h (клавіатура) стирає CMOS-пам'ять комп'ютера після 700 натиснень на клавіші. Вірус також містить рядки тексту:

SyStEm is now halted.

3.4 Завантажувальні (бутові) віруси

Цей тип вірусів називають так через те, що вони впроваджуються в завантажувальний сектор диска (Boot-сектор) або в сектор, який вміщує системний завантажувач вінчестера (Master Boot Record).

Як і для файлових вірусів, виділимо групи бутових вірусів, а для кожного окремого вірусу – класифікаційний код, дискриптор і сигнатури.

3.4.1 Класифікаційний код завантажувального вірусу

Класифікаційний код бутового вірусу складається з префікса і кількісної характеристики.

Префікс

Оскільки майже всі бутові віруси є резидентними, то використання символа R у префіксі їх класифікаційного коду недоцільне. Найважливішою властивістю бутових вірусів, які порівнюються за значенням з резидентністю файлових вірусів, є спроможність деяких бутових вірусів зберігатися в пам'яті після "теплого" перезавантаження шляхом натискування комбінації клавіш Ctrl-Alt-Del. Цю властивість позначають літерою W (service Warm reboot) в префіксі. Всі бугові віруси заражають дискети, але деякі з них заражають і вінчестер. Віруси, які інфікують тільки дискети (віруси “Brain”, “Den Zuk”), позначатимемо префіксом D.

При зараженні бут-сектора можливі два випадки: зараження бут-сектора – розділу С вінчестера (префікс B) і зараження MBR – виконуваної частини таблиці розділів (префікс M). Оскільки одним з найпоширеніших випадків розміщення хвоста бутового вірусу є його розміщення в псевдозбійних кластерах (що легко визначити, переглянувши їх вміст за допомогою Norton Utilites), то для таких вірусів у суфікс включають літеру x, за якою стоїть кількість цих кластерів, наприклад, Bxl.

Кількісна характеристика

Вибір кількісної характерристики для бутових вірусів має певну специфіку: якщо для файлових вірусів найхарактернішою ознакою зараження є збільшення довжини файла, то для бутових вірусів аналогічну роль відіграє зменшення розмірів оперативної пам'яті, яка доступна ОС.

Важливою вимогою до вибору властивостей вірусу, який використовується для класифікації, є можливість їх визначення на незараженій машині. Кількість блоків пам'яті, які використовуються бутовим вірусом, цьому критерію не відповідає, тому від цієї характеристики довелося відмовитися. Отже, використовують іншу характеристику бутового вірусу – вміст зараженого бут-сектора (вірніше, вміст перших його байтів). Разом з тим аналіз об'єму пам'яті, який повідомляє ОС, є дуже ко-

рисним для діагностики. При підозрюванні на зараження тим чи іншим вірусом можна виконати програму СНКТ38К, яка повідомляє значення об'єму пам'яті, а також дає ряд інших корисних повідомлень, включаючи об'єм пам'яті, зайнятий на диску збійними кластерами. Цю програму доцільно вставляти в код програми початкового завантаження.

За характеристику вибрано значення другого байта зараженого бут-сектора. Водночас зміст цього байта записується в 16-річній системі числення, що створює певну неузгодженість з характеристикою файлових вірусів, яка є десятковим числом. Тому у варіанті класифікаційного коду вірусу префікс і характеристика розділяються знаком "-" (мінус).

Слід підкреслити, що переглядати вміст бут-сектора можна лише тоді, коли попередньо завантажитись із захищеної від запису резервної дискети з операційною системою і необхідними антивірусними програмами, оскільки сама операція перегляду на зараженій машині може або перехоплюватися вірусом для підстановки "чистого" бут-сектора (так, наприклад, маскується вірус Dx3-E9.BRN – “Brain”), або, що ще гірше, бути тріггером для яких-небудь несанкціонованих дій. Необхідно використовувати "холодне" (за допомогою клавіші RESET), а не "тепле" (за допомогою комбінації клавіш Ctrl-Alt-Del) перезавантаження. Ця вимога базується на тому факті, що ряд бутових вірусів перехоплює переривання від клавіатури і при "теплому" перезавантаженні зберігає себе в пам'яті, навіть якщо перезавантаження здійснюється із захищеної системної дискети.

3.4.2 Дескриптор завантажувального вірусу

В головному дескрипторі відображені такі властивості:

A – деструктивні дії, які використовуються вірусом;
B – прояв вірусу;
L – довжина голови і хвоста вірусу в байтах, які розділені знаком "±";
M – маскування за наявності вірусу в пам'яті;
N – номер першого байта, що не збігається при порівнянні зараженого і нормального секторів початкового завантаження;
S – стратегія зараження (метод вибору "жертви", метод зберігання хвоста вірусу і оригінальної копії бут-сектора);
R (resident) – положення в оперативній пам'яті, реакція на "тепле" перезавантаження і розмір зайнятої пам'яті;
Z – побічні прояви дій вірусу.

3.4.3 Сигнатура бутового вірусу.

Для бутових вірусів M-, I-, B-сигнатури використовуються аналогічно тому, як це було для файлових вірусів, а J-сигнатура – в дещо іншому

вигляді. На відміну від J-сигнатури для файлових вірусів, в якій байти відповідають команді переходу і не враховуються, в J-сигнатурі для бутових вірусів вони враховуються. Це пов'язано з тим, що першою командою бут-сектора завжди є команда обходу таблиці параметрів диска, розмір якої, на відміну від розміру зараженого файла, не змінюється. Тому для бутових вірусів використовують переважно J-сигнатуру, яка складається з перших трьох байтів бут-сектора, і лише при необхідності доповнюється, починаючи з байта, на якому виконується команда переходу.

Для незараженого бут-сектора (наприклад, для MS-DOS версії 3.3) J-сигнатура дорівнює EB3490h (об'єктний код команди JMP, який служить для обходу таблиці параметрів). Цінність цієї еталонної J-сигнатури в тому, що вона порівняно легко запам'ятовується. Тому невідповідність перших трьох байтів бут-сектора, що аналізується, вказаній еталонній J-сигнатурі свідчить про зараження бут-сектора.

3.4.4 Принцип дії завантажувальних вірусів

Завантажувальні віруси заражають завантажувальний сектор флоппі-диска або boot-сектор вінчестера (MBR). Принцип дії завантажувальних вірусів оснований на алгоритмах запуску операційної системи при включенні або перезавантаженні комп'ютера – після необхідних тестів встановленого устаткування (пам'яті, дисків і т.д.) програма системного завантаження зчитує перший фізичний сектор завантажувального диску (A, C чи CD у залежності від параметрів, встановлених у BIOS Setup) і передає на нього керування.

У випадку дискети чи компакт-диску керування одержує boot-сектор, що аналізує таблицю параметрів диска (BPB - BIOS Parameter Block), вираховує адреси системних файлів операційної системи, зчитує їх у пам'ять і запускає на виконання. Системними файлами звичайно є MSDOS.SYS і IO.SYS, або IBMDOS.COM і IBMBIO.COM, або інші в залежності від установленої версії DOS, Windows чи інших ОС. Якщо ж на завантажувальному диску відсутні файли операційної системи, програма, розташована в boot-секторі диска, видає повідомлення про помилку і пропонує замінити завантажувальний диск.

У випадку вінчестера керування одержує програма, розташована в MBR вінчестера. Ця програма аналізує таблицю розбиття диска (Disk Partition Table), обчислює адресу активного boot-сектора (зазвичай цим сектором є boot-сектор диску C), завантажує його в пам'ять і передає на нього керування. Одержавши керування, активний boot-сектор вінчестера здійснює певні дії.

При зараженні дисків завантажувальні віруси "підставляють" свій код замість якої-небудь програми, що одержує керування при завантаженні системи. Принцип зараження, таким чином, однаковий: у всіх описаних

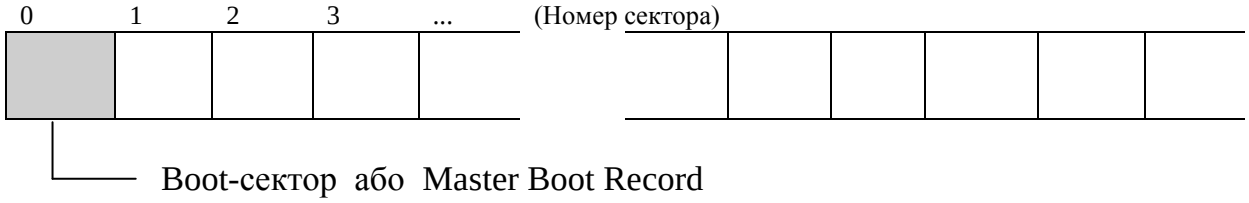
вище способах вірус "змушує" систему під час перезапуску зчитувати її в пам'ять і віддати керування не оригінальному коду завантажувальника, а коду вірусу.

Зараження дискет здійснюється єдиним відомим способом – вірус записує свій код замість оригінального коду boot-сектора дискети.

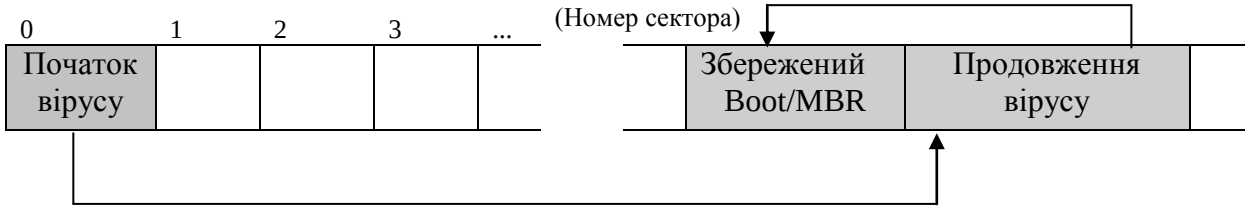
Вінчестер заражається трьома можливими способами – вірус записується або замість коду MBR, або замість коду boot-сектора завантажувального диска (звичайно диска C), або модифікує адреса активного boot-сектора в Disk Partition Table, розташованої в MBR вінчестера (рис.10).

При інфікуванні диска вірус у більшості випадків переносить оригінальний boot-сектор (чи MBR) у який-небудь інший сектор диска (наприклад, у перший вільний). Якщо довжина вірусу більше довжини сектора, то в сектор, що заражається, поміщається перша частина вірусу, інші частини розміщаються в інших секторах (наприклад, у перших вільних).

Незаражений диск



Заражений диск (підміна Boot/MBR)



Заражений диск (підміна активного Boot-сектора в Disk Partition Table)

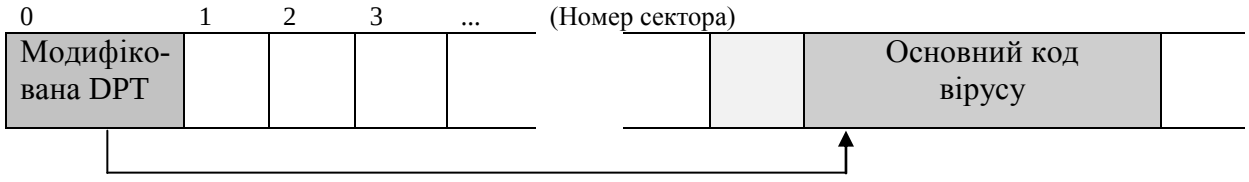


Рисунок 10 – Розміщення завантажувального вірусу

3.4.5 Розташування завантажувального вірусу

Існує декілька варіантів розміщення на диску первинного завантажувального сектора і продовження вірусу в сектори вільних кластерів логічного диска, у невикористовувані чи рідко використовувані системні сектори або у сектори, розташовані за межами диска.

1. Якщо продовження вірусу розміщається в секторах, що належать вільним кластерам диска (для пошуку цих секторів вірусу приходиться аналізувати таблицю розміщення файлів – FAT-таблицю), то, як правило, вірус ***позначає ці кластери як збійні*** (псевдозбійнй кластери). Цей спосіб використовується вірусами "Brain", "Ping-Pong" і деякими іншими.
2. У вірусах сімейства "Stoned" задіяний інший метод. Ці віруси розміщують первинний завантажувальний сектор у невикористовуваному чи рідко використовуваному секторі – в одному із секторів вінчестера (якщо такі є), розташованих ***між MBR і першим boot-сектором***, а на дискеті такий сектор вибирається з останніх секторів кореневого каталогу.
3. Деякі віруси записують свій код в ***останні сектори вінчестера***, оскільки ці сектори використовуються тільки тоді, коли вінчестер цілком заповнений інформацією (що є досить рідким явищем, якщо врахувати розміри сучасних дисків). Однак такі віруси приводять до псування файлової системи OS/2, що у деяких випадках зберігає активний boot-сектор і системні дані саме в останніх секторах вінчестера.
4. Рідше використовується метод збереження продовження вірусу ***за межами диска***. Досягається це двома способами:
   - *зменшення розмірів логічних дисків* – вірус віднімає необхідні значення з відповідних полів BPB boot-сектора і Disk Partition Table вінчестера (якщо заражається вінчестер), зменшує в такий спосіб розмір логічного диску і записує свій код у "відрізані" від нього сектори;
   - *запис даних за межами фізичної розбивки диска.* У випадку флоппі-дисків вірусу для цього приходиться форматувати на диску додатковий трек (метод нестандартного форматування), наприклад, 80-й трек на дискеті. Існують віруси, що записують свій код за межами доступного простору вінчестера, якщо, зрозуміло, це допускається встановленим устаткуванням (вірус "Hare").

Звичайно, існують і інші методи розміщення вірусу на диску, наприклад, віруси сімейства "Azusa" містять у своєму тілі стандартний завантажник MBR і при зараженні записуються поверх оригінального MBR без його збереження.

При зараженні більшість вірусів копіює в код свого завантажника системну інформацію, що зберігається в первісному завантажнику (для MBR цією інформацією є Disk Partition Table, для Boot-сектора дискет – BIOS Parameter Block). В іншому випадку система виявиться нездатною завантажити себе, оскільки дискові адреси компонентів системи вирахову-

ються на основі цієї інформації. Такі віруси досить лекго видаляються переписуванням заново коду системного завантажника в boot-секторі і MBR - для цього необхідно завантажитися з незараженої системної дискети і використовувати команди SYS для знешкодження дискет і логічних дисків вінчестера чи FDISK/MBR для лікування зараженого MBR-сектора.

Однак деякі 100%-стелс віруси не зберігають цю інформацію чи навіть, більш того, навмисно шифрують її. При звертанні системи або інших програм до заражених секторів вірус підставляє їхні незаражені оригінали, і завантаження системи відбувається без якихось збоїв, однак лікування MBR за допомогою FDISK/MBR у випадку такого вірусу приводить до втрати інформації про розбивку диска (Disk Partition Table). У цьому випадку диск може бути "оживлений" або переформатуванням із втратою всієї інформації, або відновленням Disk Partition Table "вручну", що вимагає значеної кваліфікації.

Слід також зазначити той факт, що завантажувальні віруси дуже рідко "уживаются" разом на одному диску – часто вони використовують ті самі дискові сектори для розміщення свого коду (даних). У результаті код (дані) першого вірусу виявляються зіпсованими при зараженні другим вірусом, і система або зависає при завантаженні, або зациклюється, що також приводить до її зависання.

Користувачам сучасних ОС завантажувальні віруси також можуть доставити неприємності. Незважаючи на те, що ці системи працюють з дисками напряму, минаючи виклики BIOS (що блокує вірус і унеможливлює подальше його поширення), код вірусу все-таки, хоч і дуже рідко, одержує керування при перезавантаженні системи. Тому вірус "March6", наприклад, може роками "жити" у MBR сервера і ніяк не впливати при цьому на його (сервера) роботу і продуктивність. Однак при випадковому перезавантаженні 6-го березня цей вірус цілком знищить усі дані на диску.

3.4.6 Алгоритм роботи завантажувального вірусу

Практично всі завантажувальні віруси резидентні.

Резидентні завантажувальні віруси впроваджуються в пам'ять комп'ютера при завантаженні з інфікованого диска. При цьому системний завантажник зчитує вміст першого сектора диска, з якого здійснюється завантаження, поміщає прочитану інформацію в пам'ять і передає на неї (тобто на програму-вірус) керування. Після цього починають виконуватися інструкції вірусу, що

- як правило, зменшує обсяг вільної пам'яті (слово за адресою 00400013), копіює у місце, що звільнилося, свій код і зчитує з диска своє продовження (якщо воно є). Далі деякі віруси "чекають" завантаження ОС і відновлюють це слово в його первісне значення. В результаті вони ви-

являються розташованими не за межами ОС, а в окремих блоках пам'яті;
- перехоплює необхідні вектори переривань (зазвичай INT 13H), зчитує в пам'ять оригінальний boot-сектор і передає на нього керування.

Надалі завантажувальний вірус поводиться так само, як резидентний файловий: перехоплює звертання операційної системи до дисків і інфікує їх, у залежності від деяких умов робить деструктивні дії чи викликає звукові відеоефекти.

Існують *нерезидентні* завантажувальні віруси – при завантажені вони заражають MBR вінчестера і дискети, якщо ті присутні в дисководах. Потім такі віруси передають керування оригінальному завантажнику і на роботу комп'ютера більш не впливають.

3.4.7 Приклади завантажувальних вірусів

Brain, сімейство

Складається з двох практично співпадаючих нешкідливих вірусів "Brain-Ashar" і "Brain-Singapore". Вони заражають завантажувальні сектори дискет при зверненні до них (INT 13h, AH=02h). Продовження вірусу і первинний завантажувальний сектор розміщуються у вільних кластерах диска. При пошуку цих кластерів аналізують таблицю розміщення файлів (FAT). У FAT ці кластери позначаються як збійні ("псевдозбійні" кластери). У зараженого диска встановлюється нова мітка "(C) Brain". Віруси використовують "стелс"-механізм – при спробі проглянути завантажувальний сектор зараженого диска вони "підставляють" справжній сектор.

CMOS

Небезпечний резидентний завантажувальний стелс-вірус. Псує CMOS. Копіює себе за адресою 9F800000, перехоплює INT 13h і записується в MBR вінчестера і boot-сектори дискет. Оригінальний MBR зберігає за адресою 0/0/2, оригінальний boot-сектор флопі-диска – в останньому секторі кореневого каталога.

Pentagon

Небезпечний резидентний завантажувальний вірус. Частково зашифрований. Перехоплює INT 9, 13h і вражає boot-сектор флопі-дисків при зверненні до них. При зараженні диска об’являє в FAT збійні сектори і записує туди своє продовження і первинний boot-сектор (див. вірус "Brain"). Якщо при цьому дискета вже була уражена вірусом "Brain", то "Pentagon" лікує boot-сектор цього диска, змінює його мітку і потім заражає своєю копією. На диску, що заражається, створюється файл PENTAGON.TXT. Вірус "виживає" при теплому перезавантаженні. Містить тексти:

(c) The Pentagon, Zorell Group. first sector in segment

Stoned

При завантаженні із зараженого флоппи-диска з вірогідністю 1/8 на екрані з'являється повідомлення "Your PC is now Stoned!". Крім того, містять рядок "LEGALISE MARIJUANA!". "Stoned.c" при зараженні MBR вінчестера знищує Disk Partition Table, потім комп'ютер можна завантажити тільки з флопі-диска. 1 жовтня знищує інформацію на вінчестері.

Hare

Дуже небезпечні резидентні файлово-завантажувальні стелс-поліморфік-вируси. Записуються в кінець COM- і EXE-файлів, в MBR вінчестера і boot-сектора дискет. У файлах зашифровані тричі. Застосовують поліморфізм як у файлах, так і в заражених секторах. При запуску зараженого файлу вірус розшифровує себе, заражає MBR, трасує і перехоплює INT 21h і повертає управління програмі-носію. Під Win95 він перехоплює також INT 13h. Потім вірус записується у файли при їх запуску, закритті або при виході в DOS (AH=00h,31h,4Ch). При відкритті заражених EXE-файлів лікує їх. При завантаженні із зараженої дискети вірус записується в MBR і повертає управління первинному boot-коду, при цьому вірус не залишає в пам'яті своєї резидентної копії.

При зараженні MBR вірус трасує INT 13h або напряму працює з портами контролера, потім записує своє продовження (15 секторів) в трек, що знаходиться за межами об’явленого розміру диска (LandZone). Потім затирає Disk Partition Table (в результаті цього команда FDISK/MBR може привести до повної втрати даних на диску). При завантаженні із зараженого MBR-сектора вірус відновлює Partition Table (PT) для того, щоб нормально завантажилася DOS (у цей момент стелс на рівні INT 13h ще не працює), потім зменшує розмір пам'яті (слово за адресою 00000413), копіює свій код у "відрізану" ділянку пам'яті, перехоплює INT 1Ch і передає управління первинному MBR-сектору. Перехопивши INT 1Ch, вірус чекає завантаження DOS, потім відновлює розмір системної пам'яті і перехоплює INT 13h, 21h, 28h. При першому виклику INT 28h він знову псує DPTable. При викликах INT 13h вірус перехоплює звернення до флоппі-дисків і заражає їх, для свого коду вірус форматує додатковий трек. При зверненнях до вже заражених дисків виконує стелс-програму.

Ping-Pong

Ці віруси заражають boot-сектори дисків при зверненні до них (INT 13h). На диску розташовуються способом "Brain". Містять команду MOV CS,AX (міжсегментний JMP), яка виконується тільки процесором 8086 (IBM PC і IBM PC/XT). Тому вірус не розповсюджується на комп'ютерах на базі процесорів 80x86. Не небезпечний. Перехоплює INT 8, 13h. Має байт, що містить номер версії вірусу. Якщо знаходить диск, заражений своєю попередньою версією, то "обновляє" її. Викликає відеоефект кульки

(знак 07h ASCII), що скаче, яка переміщається по екрану, відображаючись від знаків і меж екрану.

3.5 Макро-віруси

Багато табличних і графічних редакторів, системи проектування, текстові процесори мають свої макро-мови для автоматизації виконання повторюваних дій. Ці макро-мови часто мають складну структуру і розвинений набір команд. Макро-віруси є програмами на макро-мовах, вбудованих у системи обробки даних. Для свого розмноження віруси цього класу використовують можливості макро-мов і при їх допомозі переносять себе з одного зараженого файлу (документа, таблиці) в інші.

На кінець 1999 року відомо декілька систем, у яких виявлені макро-віруси. Це основні додатки Microsoft Office:

- редактор MS Word – мова WordBasic у MS Word 6/7 і VBA (Visual Basic for Applications), починаючи з MS Word 8;
- редактор таблиць MS Excel – мова VBA;
- редактор баз даних MS Access – мова VBA;
- редактор презентацій MS PowerPoint – мова VBA;
- менеджер проектів MS Project – мова VBA.

Піддався зараженню макро-вірусами також редактор AmiPro – спеціальна скрипт-мова.

Найбільше поширення одержали макро-віруси для Microsoft Office (Word, Excel і PowerPoint). Віруси в інших додатках MS Office досить рідкі, а для AmiPro відомий всего один макро-вирус. Можливе також існування макро-вірусів і для інших систем, що підтримують макро-мови достатньої потужності.

3.5.1 Причини зараження макро-вірусами

Для існування вірусів у конкретній системі (редакторі) необхідна наявність вбудованої в систему макро-мови з такими можливостями:

- програми на макромові прив'язані до документів (AmiPro) чи зберігаються в них (додатки MS Office);
- у макро-мові присутні команди копіювання макропрограм з одного файлу в інший (AmiPro) або переміщати макро-програми у службові файли системи і файли, що редагуються (MS Office);
- є можливість одержання керування макропрограмою без втручання користувача (автоматичні чи стандартні макроси), тобто при роботі з файлом за певних умов (відкриття, закриття і т.д.) викликаються макро-програми (якщо такі є), що визначені спеціальним чином (AmiPro) чи мають стандартні імена (MS Office).

Дані можливості макро-мов призначені для автоматичної обробки даних у великих організаціях чи у глобальних мережах і дозволяють організувати так званий "автоматизований документообіг". З іншого боку, можливості макро-мов таких систем дозволяють вірусу переносити свій код в інші файли, і в такий спосіб заражати їх.

Більшість макро-вирусів можна вважати резидентними, оскільки вони присутні в області системних макросів протягом усього часу роботи редактора. Вони так само, як резидентні завантажувальні і файлові віруси, перехоплюють системні події і використовують їх для свого розмноження. До подібних подій відносяться різні системні виклики, що виникають при роботі з документами Word і таблицями Excel (відкриття, закриття, створення, печатка і т.д.), виклик пункту меню, натискання на яку-небудь клавішу чи досягнення певного моменту часу. Для перехоплення подій макровируси перевизначають один чи декілька системних макросів або функцій.

При зараженні деякі макро-вируси перевіряють наявність своєї копії в об'єкті, що заражається, і повторно себе не копіюють. Інші макро-вируси не роблять цього і переписують свій код при кожному зараженні. Якщо при цьому у файлі, що заражається, чи області системних макросів уже визначений макрос, ім'я якого збігається з макросом вірусу, то такий макрос виявляється знищеним.

3.5.2 Загальні відомості про віруси в MS Office

Фізичне розташування вірусу всередині файлу залежить від його формату, що у випадку продуктів Microsoft надзвичайно складний - кожен файл-документ Word, Office чи таблиця Excel являють собою послідовність блоків даних (кожний з який також має свій формат), об'єднаних між собою за допомогою великої кількості службових даних. Цей формат називається OLE2 (Object Linking and Embedding). Структура файлів Office (OLE2) нагадує ускладнену файлову систему дисків DOS: "кореневий каталог" документа або таблиці вказує на основні підкаталоги різних блоків даних, кілька таблиць FAT містять інформацію про розташування блоків даних у документі і т.д.

Більш того, система Office Binder, що підтримує стандарти Word і Excel дозволяє створювати файли, що одночасно містять один чи декілька документів у форматі Word і одну чи декілька таблиць у форматі Excel. При цьому Word-віруси здатні вражати Word-документи, а Excel-віруси – Excel-таблиці, і все це можливо в межах одного дискового файлу. Те ж справедливо і для Office.

Слід зазначити, що Word версій 6, 7 і вище дозволяє шифрувати присутні в документі макроси. Таким чином, деякі Word-віруси присутні в заражених документах у зашифрованому (Execute only) виді.

Більшість відомих вірусів для Word несумісні з національними (у тому числі з російською) версіями Word, чи навпаки – розраховані тільки на локалізовані версії Word і не працюють під англійською версією. Однак вірус у документі все рівно залишається активним і може заражати інші комп'ютери з установленої на них відповідною версією Word.

Віруси для Word можуть заражати комп'ютери будь-якого класу, а не тільки IBM-PC. Зараження можливе у тому випадку, якщо на даному комп'ютері установлет текстовий редактор, цілком сумісний з Microsoft Word версії 6 чи 7 (наприклад, MS Word for Macintosh). Те ж справедливо для Excel і Office.

Слід також зазначити, що складність форматів документів Word, таблиць Excel і особливо Office має таку особливість: у файлах-документах і таблицях присутні "зайві" блоки даних, тобто дані, що ніяк не пов'язані з текстом, що редагується, чи таблицями, або є випадковими копіями інших даних файлу. Причиною виникнення таких блоків даних є кластерна організація даних у OLE2-документах і таблицях - навіть якщо введений всего один символ тексту, то під нього виділяється один чи навіть декілька кластерів даних. При збереженні документів і таблиць у кластерах, не заповнених "корисними" даними, залишається "сміття", що попадає у файл разом з іншими даними. Кількість "сміття" у файлах може бути зменшено скасуванням пункту настроювання Word/Excel "Allow Fast Save", однак це лише зменшує загальну кількість "сміття", але не забирає його цілком.

Наслідком цього є той факт, що при редагуванні документа його розмір змінюється незалежно від здійснених з ним дій – при додаванні нового тексту розмір файлу може зменшитися, а при видаленні частини тексту – збільшитися. Те ж і з макро-вірусом при зараженні файлу: його розмір може зменшитися, збільшитися чи залишитися незмінним.

Слід також зазначити той факт, що деякі версії OLE2.DLL містять невеликий недолік, у результаті якого при роботі з документами Word, Excel і особливо Office у блоки "сміття" можуть потрапити випадкові дані з диска, включаючи конфіденційні (вилучені файли, каталоги і т.д.). У ці блоки можуть потрапити також команди вірусу. У результаті після лікування заражених документів активний код вірусу видаляється з файлу, але в блоках "сміття" можуть залишитися частина його команд. Такі сліди присутності вірусу іноді видимі за допомогою текстових редакторів і навіть можуть викликати реакцію деяких антивірусних програм. Однак ці залишки вірусу зовсім нешкідливі Word і Excel не звертають на них ніякої уваги.

3.5.3 Принципи роботиWord/Excel/Office97-вірусів

При роботі з документом Word виконує різні дії: відкриває документ, зберігає, друкує, закриває і т.д. При цьому Word шукає і виконує відповідні вбудовані макроси – при збереженні файла по команді FILE/SAVE

викликається макрос FileSave, при збереженні по команді FILE/SAVEAS – FileSaveAs, при друці документів – FilePrint і т.д., якщо, звичайно, такі макроси визначені.

Існує також декілька "авто-макросів", що автоматично викликаються при різних умовах. Наприклад, при відкритті документа Word перевіряє його на наявність макросу AutoOpen. Якщо такий макрос присутній, то Word виконує його. При закритті документа Word виконує макрос AutoClose, при запуску Word викликається макрос AutoExec, при завершенні роботи – AutoExit, при створенні нового документа – AutoNew.

Схожі механізми (але з іншими іменами макросів і функцій) використовуються в Excel (Auto_Open, Auto_Close, Auto_Activate, Auto_Deactivate) і в Office (Document_Open, Document_Close, Document_New), у яких роль авто- і вбудованих макросів виконують авто- і вбудовані функції, що є присутнім у якому-небудь макросі чи макросах, причому в одному макросі можуть бути присутнім декілька вбудованих і функцій.

Автоматично (тобто без участі користувача) виконуються також макроси/функції, асоційовані з якою-небудь клавішею або моментом часу або датою, тобто Word/Excel викликають макрос/функцію при натисканні на яку-небудь конкретну клавішу (чи комбінацію клавіш) або при досягненні якого-небудь моменту часу. У Office97 можливості по перехопленню подій дещо розширені, але принцип використовується той самий.

Макро-віруси, що вражають файли Word, Excel чи Office, як правило, користуються одним із трьох перерахованих вище прийомів – у вірусі або присутній авто-макрос (авто-функція), або перевизначений один зі стандартних системних макросів (асоційований якимось пунктом меню), або макрос вірусу викликається автоматично при натисканні на якусь клавішу чи комбінацію клавіш. Існують також напівіруси, що не використовують цих прийомів і розмножуються, тільки коли користувач сам запускає їх.

Таким чином, якщо документ заражений, при відкритті документа Word викликає заражений автоматичний макрос AutoOpen (чи AutoClose при закритті документа) і, таким чином, запускає код вірусу, якщо це не заборонено системною змінною DisableAutoMacros. Якщо вірус містить макроси зі стандартними іменами, вони одержують керування під час виклику відповідного пункту меню (File/Open, File/Close, File/SaveAs). Якщо ж перевизначений який-небудь символ клавіатури, то вірус активізується тільки після натискання на відповідну клавішу.

Більшість макро-вірусів містять свої функції у вигляді стандартних макросів. Існують, однак, віруси, що використовують прийоми приховування свого коду і зберігають свій код у вигляді не-макросів. Відомо три подібних прийоми, усі вони використовують можливість макросів створювати, редагувати і виконувати інші макроси. Як правило, подібні віруси мають невеликий (іноді – поліморфний) макрос-завантажник вірусу, що викликає вбудований редактор макросів, створює новий макрос, заповнює його основним кодом вірусу, виконує і потім, як правило, знищує (щоб

сховати сліди присутності вірусу). Основний код таких вірусів присутній або в самому макросі вірусу у вигляді текстових рядків (іноді – зашифрованих), або зберігається в області змінних документа чи в області Autotext.

Алгоритм роботи Word-макро-вірусів

Більшість відомих Word-вірусів під час запуску переносять свій код (макроси) в область глобальних макросів ("загальні" макроси), для цього вони використовують команди копіювання макросів MacroCopy, Organizer. Copy або за допомогою редактора макросів – вірус викликає його, створює новий макрос, вставляє в нього свій код, і зберігає його в документі.

При виході з Word глобальні макроси (включаючи макроси вірусу) автоматично записуються в DOT-файл глобальних макросів (NORMAL. DOT). Таким чином, при наступному запуску редактора MS-Word вірус активізується в той момент, коли WinWord вантажить глобальні макроси, тобто відразу. Потім вірус перевизначає (чи вже містить у собі) один чи декілька стандартних макросів (наприклад, FileOpen, FileSave, FileSaveAs, FilePrint) і перехоплює в такий спосіб команди роботи з файлами. Під час виклику цих команд вірус заражає файл, до якого йде звертання. Цей вірус конвертує файл у формат Template (що унеможливлює подальші зміни формату файлу, тобто конвертування в який-небудь не-Template формат) і записує у файл свої макроси, включаючи Auto-макрос.

Таким чином, якщо вірус перехоплює макрос FileSaveAs, то заражається кожен файл, що зберігається через перехоплений вірусом макрос. Якщо перехоплений макрос FileOpen, то вірус записується у файл при його зчитуванні з диска.

Другий спосіб впровадження вірусу в систему використовується значно рідше – він базується на так званих "Add-in" файлах, тобто файлах, що є службовими доповненнями до Word. У цьому випадку NORMAL.DOT не змінюється, а Word під час запуску завантажує макроси вірусу з файлу (чи файлів), визначеного як "Add-in". Цей спосіб практично цілком повторює зараження глобальних макросів за тим виключенням, що макроси вірусу храняться не в NORMAL.DOT, а в якому-небудь іншому файлі.

Можливо також впровадження вірусу у файли, розташовані в каталозі STARTUP, – Word автоматично довантажує файли-темплейти з цього каталогу, але такі віруси поки що рідко зустрічаються.

Розглянуті вище способи впровадження в систему являють собою деякий аналог резидентных вірусів. Аналогом нерезидентности є макровіруси, що не переносять свій код в область системних макросів – для зараження інших файлів-документів вони або шукають їх за допомогою убудованих у Word функцій роботи з файлами, або звертаються до списку останніх редагованих файлів (Recently used file list). Потім такі віруси відкривають документ, заражають його і закривають.

Алгоритм роботи Excel-макро-вірусів

Методи розмноження Excel-вірусів в цілому аналогічні методам Word-вірусів. Розходження полягають у командах копіювання макросів (наприклад, Sheets.Copy) і у відсутності NORMAL.DOT – його функцію (у вірусному сенсі) виконують файли в STARTUP-каталозі Excel.

Слід зазначити, що існує два можливих варіанти розташування коду макро-вірусів у таблицях Excel. Переважна більшість таких вірусів записують свій код у форматі VBA, однак існують віруси, що зберігають свій код у старому форматі Excel версії 4.0. Такі віруси по своїй суті нічим не відрізняються від VBA-вірусів, за винятком відмінностей у форматі розташування кодів вірусу в таблицях Excel.

Незважаючи на те, що у нових версіях Excel (версія 5 і вище) використовуються досконаліші технології, можливість виконання макросів старих версій Excel була залишена для підтримки сумісності. З цієї причини всі макроси, написані у форматі Excel 4, цілком працездатні у всіх наступних версіях, незважаючи на те, що Microsoft не рекомендує використовувати їх і не включає необхідну документацію в комплект постачання Excel.

Алгоритм роботи вірусів для Access

Оскільки Access є частиною пакета Office, то віруси для Access являють собою такі ж самі макроси мовою Visual Basic, як і інші віруси, що заражають програми Office. Однак, у даному випадку замість авто-макросів у системі присутні автоматичні скрипти, що викликаються системою при різних подіях (наприклад, Autoexec). Дані скрипти потім можуть викликати різні макро-програми.

Таким чином, при зараженні баз даних Access вірусу необхідно замінити який-небудь авто-скрипт і скопіювати в базу, що заражається, свої макроси.

Зараження скриптов без додаткових макросів не є можливим, оскільки мова скриптів досить примітивна і не містить необхідних для цього функцій. Слід зазначити, що в термінах Access скрипти називаються макросами (macro), а макроси – модулями (module), однак частіше використовується уніфікована термінологія - скрипти і макроси.

Лікування баз даних Access є більш складною задачею, ніж видалення інших макро-вірусів, оскільки у випадку Access необхідно знешкодити не тільки вірусні макроси, але й авто-скрипти. А оскільки значна частина роботи Access покладена саме на скрипти і макроси, то некоректне видалення чи деактивація якого-небудь елемента може привести до неможливості операцій з базою даних. Те саме справедливо і для вірусів – некоректне заміщення авто-скриптів може привести до втрати даних, що зберігаються в базі.

3.5.6 Приклади макро-вірусів

Macro.Word.Box

Містить сім макросів AutoOpen, AutoClose, Box, Dead, FilePrint, FilePrintDefault, ToolsMacro. При викликах AutoOpen і AutoClose заражає глобальні макроси і документи. Макрос ToolsMacro використовується для заборони меню Tools/Macro. Решта макросів містить процедури зараження і різні ефекти. Виявляється декількома способами. При друці вставляє в документи рядки на китайською мовою, виводить MessageBox, записує на диск і запускає вірус "OneHalf.3544", створює і програє WAV-файл (звуковий ефект). Викликає команди DOS:

```
echo y|format c/u
echo y|format c/u/vTwnos1
```

Містить рядки:

```
Taiwan Super No.1 Macro Virus
Twno1-S
Today Is My Birthday
```

Macro.Word.Catch

Містить шість макросів AutoOpen, encrypt1, FileSave, AutoClose, infectdoc, infectnorm. Зараження системної області макросів і документів відбувається при відкритті файлів. Вірус замінює в документах букви "i" на "o", "o" на "i", "a" на "e", "e" на "a". Причому підміна символів непомітна. При відкритті заражених файлів вірус відновлює текст документа в первинному вигляді, а при збереженні документів на диск – знову замінює. В результаті після лікування вірусу текст документів може виявитися зіпсованим. Після кожної заміни вірус видає в StatusBat крапку. При закритті документів залежно від лічильника випадкових чисел (з вірогідністю 1%) вірус видає повідомлення:

```
Its a Catch 22 Situation!
```

Macro.Excel.Soldier

Поліморфний макро-вірус, заражає електронні таблиці Excel. Містить чотири функції з постійними іменами Auto_Open, Auto_Close, Delay, Poly; і декілька функцій з випадковими іменами. При відкритті зараженої таблиці видаляє рядки меню Format/Sheet/Hide і Format/Sheet/Unhide (стелс). При закритті заражає файли, що знаходяться у поточному каталозі. При зараженні залежно від системного датчика випадкових чисел вставляє в початок тексту функції з випадковими іменами і випадковими рядками. Також залежно від випадкового числа виводить в заголовок Excel рядок, що рухається по екрану:

```
Microsoft Excel
```

3.6 Мережеві віруси

Мережеві віруси поширюються по комп'ютерних мережах. Існують комбінації, наприклад, файлово-бутові віруси, які заражають і файли, і бут-сектори дисків. Крім того, у комп'ютерних мережах можуть розповсюджуватись віруси будь-яких типів. Віруси розповсюджуються від одного користувача до іншого внаслідок обміну програмними продуктами. Локальні мережі, як відомо, призначені для сумісного використання програмних пакетів кількома користувачами. Мережі дозволяють широко обмінюватися програмами і даними. Очевидно, що при цьому створюється зручне середовище для розповсюдження вірусу. Так, якщо вірус заразив програму login.exe, яка знаходиться на сервері і яку запускає кожен користувач при вході в мережу, то дуже швидко вірус з'явиться на всіх робочих станціях. Але на практиці ситуація виглядає не дуже драматично, бо мережні операційні системи мають механізми захисту і розподілу користувачів. При грамотному використанні цих можливостей можна обмежити область, в якій може розповсюдитись вірус, внесений з робочої станції, тільки робочою областю того користувача, який вніс його в систему.

Існують мережні віруси, які розраховані на спеціальні мережні диски, наприклад, на систему Netware – вони несанкціоновано входять у систему і, використовуючи максимальні повноваження супервізора, пошкоджують програми, які знаходяться на мережних дисках.

Все це свідчить про те, що рівень захищеності сучасних операційних систем поки що бажає бути кращим. Доки залишається принципова можливість такого втручання, доти існує небезпека появи таких вірусів.

3.6.1 IRC-черв’яки

IRC (Internet Relay Chat) – це спеціальний протокол, розроблений для комунікації користувачів Інтернет у реальному часі. Цей протокол надає можливість так званої Інтернет-розмови за допомогою спеціально розробленого програмного забезпечення. IRC чимось схожий на телефонну розмову, за винятком того, що в розмові можуть брати участь більш двох співрозмовників, що поєднуються по інтересах у різні групи IRC-конференцій. Для підтримки IRC-конференцій створені різні IRC-сервера, до яких підключаються учасники IRC-розмов. В усьому світі нараховується величезна кількість IRC-серверів, об'єднаних у так звані мережі. Найбільшою є мережа EFnet, сервер якої щодня одночасно відвідують кілька десятків тисяч користувачів.

Для підключення до IRC-сервера і ведення IRC-розмов розроблені спеціальні програми – IRC-клієнти. Підключившись до IRC-сервера за допомогою програми-клієнта, користувач завичай вибирає тему IRC-конфе-

ренції, командою join входить в одну або декілька конференцій ("канали" у термінах IRC) і починає спілкування з іншими "мешканцями" цих каналів.

Крім відвідування загальних (public) конференцій користувачі IRC мають можливість спілкуватися один-на-один з будь-яким іншим користувачем (private), при цьому вони навіть не обов'язково повинні бути на одному каналі. Крім цього існує досить велика кількість IRC-команд, за допомогою яких користувач може одержати інформацію про інших користувачів і канали, змінювати деякі установки IRC-клієнта та інше. Існує також можливість передавати і приймати файли – саме на цій можливості і базуються IRC-черв’яки.

3.6.2 IRC-клієнти

На комп'ютерах з MS Windows найпоширенішими клієнтами є mIRC і PIRCH. Це не дуже об'ємні, але досить складні програмні продукти, що крім надання основних послуг IRC (підключення до серверів і каналів) мають ще і масу додаткових можливостей.

До таких можливостей відносяться, наприклад, сценарії роботи (скрипти) і задання автоматичної реакції на різні події. Наприклад, з появою під час розмови визначеного слова IRC-клієнт передає повідомлення користувачу, що послав його. Можливе відключення користувача від каналу; посилка персональних повідомлень новим користувачам, що підключаються до каналу; і багато чого іншого. У PIRCH-клієнті, наприклад, подій, на які передбачена реакція, більше 50.

3.6.3 Скрипт-черв’яки

Як виявилося, могутня і розгалужена система команд IRC-клієнтів дозволяє на основі їх скриптів створювати комп'ютерні віруси, що передають свій код на комп'ютери користувачів мереж IRC, так називані IRC-черв’яки.

Перший інцидент із IRC-черв’яком зафіксований наприкінці 1997 року: користувачами mIRC-клієнта був виявлений скрипт (файл SCRIPT.INI), що переносив свій код через канали IRC і заражав mIRC-клієнтів на комп'ютерах користувачів, що підключалися до заражених каналів. Як виявилося, скрипт-черв’яки є досить простими програмами, і через досить короткий час на основі першого mIRC-черв’яка були створені і "випущені" у мережі декілька десятків різних скрипт-черв’яків.

Принцип дії таких IRC-черв’яків приблизно однаковий. За допомогою IRC-команд файл сценарію роботи (скрипт) чи реакції на IRC-події автоматично посилається з зараженого комп'ютера кожному користувачу, що під’єднується до каналу. Надісланий файл-сценарій заміщає стандартний і

при наступному сеансі роботи вже знову заражений клієнт буде розсилати черв’яка. Черв’яки при цьому використовують особливості конфігурації клієнта, завдяки якій прийняті файли всіх типів розміщуються в кореневий каталог клієнта. Цей каталог також містить і основні скрипти клієнта, включаючи завантажувальні mIRC-скрипти SCRIPT.INI, MIRC.INI і PIRCH-скрипт EVENTS.INI. Ці скрипти автоматично виконуються клієнтом при старті і далі використовуються як основний сценарій його роботи.

Деякі IRC-черв’яки також містять троянський компонент – по заданих ключових словах здійснюють руйнівні дії на уражених комп'ютерах. Наприклад, черв’як "pIRCH.Events" по визначеній команді стирає усі файли на диску користувача.

У скрипт-мовах клієнтів mIRC і PIRCH також існують оператори для запуску звичайних команд ОС і модулів, що виконуються, програм DOS і Windows. Ця можливість IRC-скриптів послужила основою для появи скрипт-черв’якв нового покоління, що крім скриптів заражали комп'ютери користувачів EXE-вірусами, установлювали "троянських коней", і т.п.

Скрипт-черв’яки працездатні тільки в тому випадку, якщо користувач дозволяє копіювання файлів з мережі на свій комп'ютер. Дана опція IRC-клієнтів називається “DCC autoget” – одержання файлів по протоколу DCC автоматично і без попереджуючого повідомлення. При відключеній опції заражений файл приймається, розміщується в каталозі клієнта і в наступному сеансі роботи продовжує своє поширення. При цьому користувач не одержує ніяких попереджуючих повідомлень.

Слід зазначити, що фірма-виготовлювач клієнта mIRC відреагувала досить оперативно і буквально через кілька днів після появи першого черв’яка випустила нову версію свого клієнта, у якій були закриті пробіли в захисті.

3.6.4 Приклади мережевих вірусів

mIRC.Acoragil і mIRC.Simpsalapim

Перші відомі mIRC-черв’яки виявлені наприкінці 1997 року. Назви одержали по кодових словах, що використовуються черв’яками: якщо в тексті, переданому в канал будь-яким користувачем, присутній рядок "Acoragil", то всі користувачі, заражені черв’яком mIRC.Acoragil автоматично відключаються від каналу. Те саме відбувається з черв’яком mIRC. Simpsalapim – він аналогічно реагує на рядок "Simpsalapim". При розмноженні черв’яки командами mIRC пересилають свій код у файлі SCRIPT. INI кожному новому користувачу, що підключається до каналу. Містять троянську частину. Черв’як mIRC.Simpsalapim містить код захоплення каналу IRC – якщо mIRC власника каналу заражений, то по введенню кодового слова "ananas", зловмисник перехоплює керування каналом. Черв’як mIRC. Acoragil по кодових словах пересилає системні файли ОС. Деякі ко-

дові слова обрані так чином, що не привертають уваги жертви – hi, cya чи the. Одна з модифікацій цього черв'яка пересилає зловмиснику файл паролів UNIX.

Win95.Fono

Небезпечний резидентный файлово-завантажувальний поліморфік-вірус. Використовує mIRC як один із способів свого поширення: перехоплює системні події Windows і при запуску файлу MIRC32.EXE активізує свою mIRC-процедуру. При цьому відкриває файл MIRC.INI і записує в його кінець команду, що знімає захист:

[fileserver]
Warning=Off .

Потім створює файли SCRIPT.INI і INCA.EXE. Файл INCA.EXE містить дроппер вірусу, скрипт файлу SCRIPT.INI пересилає себе і цей дроппер у канал IRC кожному, хто приєднується до каналу або виходить з нього.

pIRCH.Events

Перший відомий PIRCH-черв'як. Розсилає себе кожному користувачу, що приєднався. По ключових словах виконує різні дії, наприклад:

- по команді ".query" відбувається свого роду переклик, по якому заражені системи відповідають <Так, я вже заражена>;
- по команді ".exit" завершує роботу клієнта;

По інших командах черв'як видаляє всі файли з диска C, надає доступ до файлів на зараженому комп'ютері, і т.д.

IRC-Worm.Pron

Мережевий вірус-черв'як, зашифрований. Розмножується в IRC-каналах і використовує для свого розмноження mIRC-клієнта. Має дуже невелику довжину – всього 582 байти. Передається з мережі на комп'ютер у вигляді файлу PR0N.BAT. При його запуску вірус копіює себе у файл PR0N. COM і запускає його на виконання. Заголовок вірусу влаштований таким чином, що він може виконуватися як BAT-, так і COM-програма, і в результаті управління передається на основну процедуру зараження системи. При зараженні системи вірус використовує дуже простий прийом: він копіює свій BAT-файл в поточний каталог і в каталог C\WINDOWS\SYSTEM (якщо такий відсутній, то вірус не заражає систему). Потім вірус записує свій код у файл WINSTART.BAT. Для розповсюдження свого коду через mIRC вірус створює новий файл SCRIPT.INI в каталозі mIRC-клієнта. Цей каталог вірус шукає за чотирма варіантами:

C\MIRC; C\MIRC32; C\PROGRA~1\MIRC; C\PROGRA~1\MIRC32 .

Скрипт вірусу містить всього одну команду – кожному користувачу, що підключається до зараженого каналу, передається вірусний файл PR0N.BAT. Вірус містить рядок-"копірайт".

3.7 Стелс-віруси

Стелс-віруси тими чи іншими способами приховують факт своєї присутності в системі, підставляючи замість себе незаражені ділянки інформації. Крім того, такі віруси при зверненні до файлів використовують досить оригінальні алгоритми, що “обманюють” резидентні антивірусні програми.

Відомі стелс-віруси всіх типів – завантажувальні віруси, файлові DOS-віруси і навіть макро-віруси.

3.7.1 Завантажувальні стелс-віруси

Завантажувальні стелс-віруси для приховання свого коду використовують два основних способи.

Перший спосіб полягає в тому, що вірус перехоплює команди читання зараженого сектора (INT 13h) і підставляє замість нього незаражений оригінал. Цей спосіб робить вірус невидимим для будь-якої DOS-програми, включаючи антивіруси, нездатні "лікувати" оперативну пам'ять комп'ютера. Можливе перехоплення команд читання секторів на рівні більш низькому, чим INT 13h.

Другий спосіб спрямований проти антивірусів, що підтримують команди прямого читання секторів через порти контролера диска. Такі віруси при запуску будь-якої програми (включаючи антивірус) відновлюють заражені сектори, а після закінчення її роботи знову заражають диск. Оскільки для цього вірусу приходиться перехоплювати запуск і закінчення роботи програм, то він повинен перехоплювати також DOS-переривання INT 21h.

З деяким застереженням стелс-вірусами можна назвати віруси, що вносять мінімальні зміни в сектор, що заражається (наприклад, при зараженні MBR змінюють тільки активну адресу завантажувального сектора – зміні підлягають тільки 3 байти), або маскуються під код стандартного завантажника.

3.7.2 Файлові стелс-віруси

Більшість файлових стелс-вірусів використовує ті самі прийоми, що приведено вище: вони або перехоплюють DOS-виклики звертання до файлів (INT 21h), або тимчасово лікують файл при його відкритті і заражають при закритті. Так само, як і для boot-вірусів, існують файлові віруси, що використовують для своїх стелс-функцій перехоплення переривань більш низького рівня – виклики драйверів DOS, INT 25h і навіть INT 13h.

Повноцінні файлові стелс-віруси, що використовують перший спосіб приховання свого коду, здебільшого досить громіздкі, оскільки їм прихо-

дитися перехоплювати велику кількість DOS-функцій роботи з файлами: відкриття/закриття, читання/записування, пошук, запуск, перейменування і т.д., причому необхідно підтримувати обидва варіанти деяких викликів (FCB/ASCII), а після появи Windows 95/NT їм стало необхідно також обробляти третій варіант – функції роботи з довгими іменами файлів.

Деякі віруси використовують частину функцій повноцінного стелс-вірyса. Найчастіше вони перехоплюють функції DOS FindFirst і FindNext (INT 21h, AH=11h, 12h, 4Eh, 4Fh) і зменшують розмір заражених файлів. Такий вірус неможливо визначити по зміні розмірів файлів, якщо, звичайно, він резидентно знаходиться в пам'яті.

Програми, що не використовують зазначені функції DOS (наприклад, "Нортоновські утиліти"), а прямо використовують вміст секторів, що зберігають каталог, показують правильну довжину заражених файлів.

3.7.3 Макро-стелс-віруси

Реалізація стелс-алгоритмів у макро-вірусах є, напевно, найбільш простою задачею – досить усього лише заборонити виклик меню File/ Templates або Tools/Macro. Досягається це або видаленням цих пунктів меню зі списку, або їхньою підміною на макроси FileTemplates і ToolsMacro.

Частково стелс-вірусами можна назвати невелику групу макро-вірусів, що зберігають свій основний код не в самому макросі, а в інших областях документа – у його змінних чи в Auto-text.

3.7.4 Приклади стелс-вірусів

Crusher

Безпечний резидентний MBR-EXE-стелс-вірус. При запуску зараженого файла він записується в MBR вінчестера, потім перехоплює INT 21h і записується в початок EXE-файлів при їх копіюванні. При завантаженні з ураженого диска перехоплює INT 1Ch, чекає завантаження DOS, потім відновлює INT 1Ch, перехоплює INT 21h і приступає до зараження файлів. Якщо при роботі вірусу йому не вистачає пам'яті, він повідомляє "Insufficient memory" і повертається в DOS. При запуску CHKDSK вірус виводить текст:

Crusher… You are damned. Bit Addict / Trident.

Ekoterror

Резидентний небезпечний стелс-вірус, при запуску зараженого файлу записується в MBR вінчестера і передає управління програмі-носію, при завантаженні з ураженого MBR перехоплює INT 8, 13h, а потім, використ-

товуючи INT 8, перехоплює INT 21h і записується в початок .COM-файлів при їх створенні. Періодично розшифровує і виводить текст:

EkoTerror (C) 1991 ATK-toimisto P.Linkola Oy
Kovalevysi on poistettu kДytФstД luonnonsuojelun nimessД.
VihreДssД yhteiskunnassa ei saa olla ydinsДhkФllД toimivia kovalevyjД.

а потім завішує комп'ютер. В деяких випадках некоректно уражає MBR, в результаті DOS гине при завантаженні.

Rasek, сімейство

Дуже небезпечні файлово-завантажувальні віруси, що самошифруються. При запуску зараженого файлу записують себе в MBR вінчестера, потім перехоплюють INT 13h, 12h. Переривання INT 13h використовується для реалізації стелс-механізму при читанні ураженої MBR. Віруси також записують в Boot-сектори флопі-дисків програму, яка при завантаженні з такого флопі стирає FAT вінчестера. Переривання INT 21h використовується вірусом для зараження COM- і EXE-файлів при їх запуску, вірус записується в кінець файлів. У тілі вірусу міститься рядок "AND.COM", вірус шукає цей рядок в імені файла і не вражає такі файли (COMMAND.COM). У тілі вірусів також міститься і інші рядки, анприклад:

"Rasek.1310" RASEK v1.1,from LA CORUeA(SPAIN) .Jan93

Vecna

Дуже небезпечний резидентний файлово-завантажувальний стелс-вірус. Заражає boot-сектори дискет, MBR вінчестера і записується поверх EXE-файлів (псує їх). При запуску зараженого EXE-файла записується в MBR вінчестера, розшифровує і виводить текст:

Out of memory.

Потім повертає управління DOS. При завантаженні з диска перехоплює INT 13h, залишається резидентним в пам'яті і заражає дискети і EXE-файли на дискетах. Під налагоджувачем і на Pentium-комп'ютерах виводить текст:

Vecna Live ...

Має досить серйозну помилку – може повернути управління оригінальному обробнику INT 13h із зіпсованим вмістом регістра AX, що може привести до втрати даних на диску і навіть до його форматування.

Kyokushinkai

Дуже небезпечний резидентний файлово-завантажувальний вірус. При запуску зараженого записується в MBR вінчестера, перехоплює INT 12h, 13h, 1Ch, 21h і при запуску програм шукає EXE-файли і записується в їх кінець. Заражений MBR-сектор не видно при активному в пам'яті вірусі (стелс). Залежно від поточного часу стирає системні сектори рядком:

+++++++ КШФkБshЛдkДЛ +++++++.- 39-mynrazCmeroizв.....

3.8 Поліморфік-віруси

Не так давно виявлення вірусів було простою справою: кожен вірус створював точну копію самого себе при тиражуванні і інфікуванні нових файлів і завантажувальних секторів, тому антивірусним програмам необхідно було тільки знати послідовність байтів, що становлять вірус. Для кожного вірусу фахівці виявляли унікальну послідовність байтів – його сигнатуру. Наявність такої сигнатури служила високонадійним індикатором присутності небажаного коду, що і примусило авторів вірусів спробувати приховувати будь-яку послідовність байтів, здатну видати присутність їх творінь. Вони стали робити це шляхом шифрування вірусів.

Віруси, що шифрують свій код, відомі досить давно. Проте самі процедури розшифрування досить легко виявити, зокрема, тому, що далеко не всі автори вірусів мають досить знань для написання власних процедур шифрування і розшифрування, тому багато вірусів використовують для цього один і той самий код. Тепер сканери вірусів шукають певні процедури розшифрування. Хоча виявлення такої процедури ще нічого не говорить про те, який саме вірус присутній у зашифрованому вигляді, але це вже сигнал про наявність вірусу. Тому останнім прийомом зловмисників стає поліморфізм.

Перші поліморфні віруси Tequila і Maltese Amoeba з'явилися в 1991 році. Все б нічого, але в 1992 році автор, відомий під псевдонімом Dark Avenger, написав свого роду комплект «Сделай сам» для мутаційного механізму, який він зробив частиною вірусу Maltese Amoeba. До 1992 року розробники вірусів працювали насправді дарма. Абсолютно ясно, що кваліфікація професіоналів у сфері антивірусної безпеки ніяк не нижча, і тому багатомісячні зусилля “вірусописьменників” коштували в крайньому випадку зайвих годин роботи для фахівців. Адже всі зашифровані віруси обов'язково містили якийсь незашифрований фрагмент: сам розшифровувач або деяку його частину, по яких можна було б побудувати сигнатуру даного вірусу і потім вже боротися з ним звичними способами.

Ситуація змінилася, коли були придумані алгоритми, що дозволяють не тільки шифрувати код вірусу, але і міняти розшифровувач. Сама постановка такої задачі питань не викликає: ясно, що можна побудувати різні розшифровувачі. Суть у тому, що цей процес автоматизований, і кожна нова копія вірусу містить новий розшифровувач, кожен біт якого може відрізнятися від бітів розшифровувача копії, що породила її.

Отже, до поліморфік-вірусів відносяться ті з них, які неможливо (чи вкрай важко) знайти за допомогою так званих вірусних масок – ділянок постійного коду, специфічних для конкретного вірусу. Досягається це двома основними способами:

- шифруванням основного коду вірусу з непостійним ключем і випадковим набором команд розшифровувача;

- зміною самого виконуваного коду вірусу.

Існують також інші, досить екзотичні приклади поліморфізму: DOS-вірус "Bomber", наприклад, не зашифрований, однак послідовність команд, що передає керування коду вірусу, є цілком поліморфною. Поліморфізм різного ступеня складності зустрічається у вірусах усіх типів – завантажувальних, файлових і навіть у макро-вірусах.

3.8.1 Поліморфні розшифровувачі

Найпростішим прикладом частково поліморфного розшифровувача є наступний набір команд, в результаті застосування якого жоден байт коду самого вірусу і його розшифровувача не є постійним при зараженні різних файлів:

```
   MOV  reg_1, count       ; reg_1, reg_2, reg_3 вибираються з
   MOV  reg_2, key         ; AX,BX,CX,DX,SI,DI,BP
   MOV  reg_3, _offset     ; count, key, _offset також можуть мінятися
_LOOP:
        xxx     byte ptr [reg_3], reg_2  ; xor, add чи sub
        DEC   reg_1
        Jxx     _loop           ; ja чи jnc
        ; далі йдуть зашифровані код і дані вірусу
```

Більш складні поліморфік-віруси використовують значно більш складні алгоритми для генерації коду своїх розшифровувачів. Приведені вище інструкції переставляються місцями від зараження до зараження, розбавляються нічого не змінюючими командами типу

NOP, STI, CLI, STC, CLC, DEC <невикористовуваний регістр>

або

XCHG <невикористовувані регістри>.

Повноцінні ж поліморфік-віруси використовують ще більш складні алгоритми, у результаті роботи яких у розшифровувачі вірусу можуть зустрітися операції SUB, ADD, XOR, ROR, ROL і інші в довільній кількості і порядку. Завантаження і зміна ключів і інших параметрів шифровки виробляється також довільним набором операцій, у якому можуть зустрітися практично всі інструкції процесора Intel (ADD, SUB, TEST, XOR, OR, SHR, SHL, ROR, MOV, XCHG, JNZ, PUSH, POP ...) із усіма можливими режимами адресації. З'являються також поліморфік-віруси, розшифровувач яких використовує інструкції аж до Intel386. В результаті на початку файлу, зараженого подібним вірусом, йде набір безглуздих на перший погляд інструкцій, причому деякі комбінації, що цілком працездатні, не аналізуються фірмовими дизассемблерами (наприклад, сполучення CSCS чи CSNOP). І серед цієї "каші" з команд і даних зрідка прослизають MOV, XOR, LOOP, JMP – інструкції, що дійсно є "робітниками".

3.8.2 Рівні поліморфізму

Існує розподіл поліморфік-вірусів на рівні в залежності від складності коду, що зустрічається в розшифровувачах цих вірусів. Такий розподіл уперше запропонував доктор Алан Соломон, а згодом Весселин Бончев розширив його.

Рівень 1: віруси, що мають деякий набір розшифровувачів з постійним кодом і при зараженні вибирають один з них. Такі віруси є "напівполіморфіками" і носять також назву "олігоморфік" (oligomorphic). Приклади: "Cheeba", "Slovakia", "Whale".

Рівень 2: розшифровувач вірусу містить одну чи кілька постійних інструкцій, основна ж його частина непостійна.

Рівень 3: розшифровувач містить невикористовувані інструкції – "сміття" типу NOP, CLI, STI і т.д.

Рівень 4: у розшифровувачі використовуються взаємозамінні інструкції і зміна порядку проходження (перемішування) інструкцій. Алгоритм розшифрування при цьому не змінюється.

Рівень 5: використовуються всі перераховані вище прийоми, алгоритм розшифрування не постійний, можливе повторне зашифрування коду вірусу і навіть часткове зашифрування самого коду розшифровувача.

Рівень 6: permutating-віруси. Зміні підлягає основний код вірусу – він поділяється на блоки, що при зараженні переставляються в довільному порядку. Вірус при цьому залишається працездатним. Подібні віруси можуть бути незашифрованими.

Наведений вище розподіл не вільний від недоліків, оскільки створений за єдиним критерієм – можливість детектувати вірус по коду розшифровувача за допомогою стандартного прийому вірусних масок: рівень 1 – для детектування вірусу досить мати кілька масок; рівень 2 – детектування по масці з використанням "wildcards"; рівень 3 – детектування по масці після видалення інструкцій-"сміття"; рівень 4 – маска містить кілька варіантів можливого коду, тобто стає алгоритмічною; рівень 5 – неможливість детектування вірусу по масці.

Недостатність такого розподілу продемонстрована у вірусі 3-го рівня поліморфічності, що так і називається – "Level3". Цей вірус, будучи одним з найбільш складних поліморфік-вірусів, за приведеним вище розподом попадає в Рівень 3, оскільки має постійний алгоритм розшафровки, перед яким стоїть велика кількість команд-"сміття". Однак у цьому вірусі алгоритм генерування "сміття" доведений до досконалості в коді розшифровувача можуть зустрітися практично всі інструкції процесора i8086.

Якщо зробити розподіл на рівні з погляду антивірусів, що використовують системи автоматичної розшифровки коду вірусу (емулятори), то розподіл на рівні буде залежати від складності емуляції коду вірусу.

Можливо детектування вірусу й інших прийомів, наприклад, розшифровка за допомогою елементарних математичних законів і т.д.

3.8.3 Зміна виконуваного коду

Найбільш часто подібний спосіб поліморфізму використовується макро-вірусами, що при створенні своїх нових копій випадковим чином змінюють імена своїх змінних, вставляють порожні рядки чи змінюють свій код яким-небудь іншим способом. У такий спосіб алгоритм роботи вірусу залишається без змін, але код вірусу практично цілком міняється від зараження до зараження.

Рідше цей спосіб застосовується складними завантажувальними вірусами. Такі віруси впроваджують у завантажувальні сектори лише досить коротку процедуру, що зчитує з диска основний код вірусу і передає на нього керування. Код цієї процедури вибирається з декількох різних варіантів (які також можуть бути розведені "порожніми" командами), команди переставляються між собою і т.д.

Ще рідше цей прийом зустрічається у файлових вірусів – адже їм приходиться цілком змінювати свій код, а для цього вимагаються досить складні алгоритми.

На сьогоднішній день відомі всего два таких віруси, один із яких ("Ply") випадковим образом переміщає свої команди по своєму тілу і замінює їх на команди JMP чи CALL. Інший вірус ("TMC") використовує більш складний спосіб – щоразу при зараженні вірус змінює місцями блоки свого коду і даних, вставляє "сміття", у своїх асемблерних інструкціях встановлює нові значення оффсетів на дані, змінює константи і т.д. В результаті, хоча вірус і не шифрує свій код, він є поліморфік-вірусом – у коді не присутній постійний набір команд. Більш того, при створенні своїх нових копій вірус змінює свою довжину.

Деякі віруси (наприклад, віруси сімейства Eddie, Murphy) використовують частину функцій повноцінного вірусу-невидимки. Зазвичай вони перехоплюють функції DOS FindFirst і FindNext і «зменшують» розмір заражених файлів. Такий вірус неможливо визначити за зміною розмірів файлів, якщо, звичайно, він резидентно знаходиться в пам'яті. Програми, що не використовують вказані функції DOS (наприклад, Norton Commander), а напряму звертаються до вмісту секторів, які зберігають каталог, показують правильну довжину заражених файлів.

При інфікуванні файлу вірус може здійснювати дії, що маскують і прискорюють його розповсюдження. До подібних дій можна віднести обробку атрибуту Read-only, зняття його перед зараженням і подальше відновлення цього атрибуту. Багато файлових вірусів прочитують дату останньої модифікації файлу і відновлюють її після зараження. Для маскування свого розповсюдження деякі віруси перехоплюють переривання

DOS, що виникає при зверненні до диска, захищеного від запису, і самостійно обробляють його. Тому серед особливостей алгоритму файлового вірусу можна назвати наявність або відсутність обробки і швидкість його розповсюдження. Швидкість розповсюдження файлових вірусів, що заражають файли тільки під час їх запуску на виконання, буде нижчою, ніж у вірусів, що заражають файли при їх відкритті, перейменуванні, зміні їх атрибутів і т.д. Деякі віруси при створенні своєї копії в оперативній пам'яті намагаються зайняти область пам'яті з найстаршими адресами, руйнуючи тимчасову частину командного інтерпретатора COMMAND.COM. Після закінчення роботи зараженої програми тимчасова частина інтерпретатора відновлюється, при цьому відбувається відкриття файлу COMMAND.COM і його зараження, якщо вірус вражає файли при їх відкритті.

3.8.4 Приклади поліморфік-вірусів

Amoeba.2367

Дуже небезпечний резидентний поліморфик-вірус. Перехоплює INT 21h і записується в кінець COM- і EXE-файлів при їх запуску або відкритті. 21 березня і 1 листопада знищує інформацію на вінчестері. Містить тексти:

Tosee aworld in a grain of sand,
And a heaven in a wildflower
Hold Infinity in the palm of your hand
And Eternity in an hour.

"THE VIRUS 16/3/91 AMOEBA virus by the Hacker Twins (C) 1991 This is nothing, wait for the release of AMOEBA II-The universal infector, hidden to any eye but ours! Dedicated to the University of Malta- the worst educational system in the universe,and the destroyer of 5X2 years of human life.

Simulation

Безпечний нерезидентний полиморфик-вірус. Шукає .COM-файли і записується в їх кінець. Періодично виводить одне з повідомлень, після чого завішує комп'ютер:

HA HA HA YOU HAVE A VIRUS ! FRODO LIVES!
Have you ever danced with the Devil in the pale moonlight?
DATACRIME VIRUS RELEASED 1 MARCH 1989 ALIVE...
Your system is infected by the SIMULATION virus.
Have a nice day!

Predator (файлово-завантажувальні)

Нешкідливі резидентні COM-EXE-MBR-Boot-поліморфік-віруси. Під час запуску зараженого файлу трасують і перехоплюють INT 13h, 21h і записуються в MBR вінчестера. Потім записуються в кінець COM- і EXE-

файлів при зверненнях до них. Вражають Boot-сектори дискет. При завантаженні з ураженого флопі-диска перехоплюють INT 13h і чекають завантаження DOS, потім перехоплюють INT 21h і приступають до зараження. Містять текст:

THE PREDATOR. TORPNACSAELCFASVVAPC.
VANOCED

Останній рядок містить частини імен файлів (задом наперед), які не вражаються вірусом PROT, SCAN, CLEA, VSAF, CPAV, NAV, DECO.

Також містять рядки:

"Predator.2248" Predator virus #2 (c) 1993 Priest - Phalcon/Skism
"Predator.2424" Predator virus #2 (c) 1993
Here comes the Predator!

Samara.1536

Безпечний резидентний файлово-завантажувальний поліморфік-вірус. При старті з інфікованого файлу заражає MBR вінчестера, перехоплює INT 21h і записується в кінець COM- і EXE-файлів при їх запуску (окрім COMMAND.COM). Забороняє запуск антивірусів AVPLITE, AIDSTEST, AVP, DRWEB, SCAN.

При завантаженні з інфікованого MBR вірус перехоплює INT 13h, чекає завантаження DOS і потім перехоплює INT 21h. При завантаженні з boot-сектора дискети вірус ще заражає MBR. При зараженні MBR і boot-секторів не зберігає їх оригінали. Для збереження працездатності системи вірус при завантаженні із зараженого диска самостійно прочитує і запускає на виконання перший логічний сектор диска C, який містить завантажувальний код операційної системи.

OneHalf, сімейство

Дуже небезпечні резидентні файлово-завантажувальні поліморфік-віруси. При запуску заражають MBR вінчестера, при завантаженні з ураженого диска перехоплюють INT 13h, 1Ch, 21h і записуються в COM- і EXE-файли при зверненні до них. Не заражають файли SCAN, CLEAN, FINDVIRU, GUARD, NOD, VSAFE, MSAV, CHKDSK. Код-розшифровувач цих вірусів розкиданий по всьому файлу з випадковими зсувами. При зараженні вінчестера вірус прочитує його MBR і сканує таблицю розділів диска (DPT). У ній він шукає останній DOS'івський диск – логічний диск (FAT-12,16/BIGDOS) або Extended partition, і коли знаходить, підраховує номер першого і останнього циліндрів знайденого диска (або Extended partition). При цьому вірус досить грамотно обробляє диски, що мають більше 1024 циліндрів і не вписуються в стандарти INT 13h. Вірус запам'ятовує адреси цих циліндрів і заражає вінчестер.

Потім при завантаженні із зараженого вінчестера вірус шифрує два останні циліндри диска, при наступному завантаженні – ще два і т.д., поки не дійде до першого циліндра. При цьому вірус використовує адреси пер-

шого і останнього циліндрів диска, які запам'ятав при зараженні вінчестера. Коли кількість зашифрованих циліндрів перевалить за половину диска, вірус повідомляє (залежно від поточної дати і свого "покоління"):

Disк is one half.

Press any key to continue...

Таким чином, чим частіше перезавантажується заражений комп'ютер, тим більше за дані виявляються зашифрованими. Після завантаження в пам'ять вірус розшифровує/зашифровує ці сектори "на льоту", тому користувач не помічає того, що його дані зіпсовані. Проте якщо вилікувати MBR, то всі дані виявляються втраченими.

"OneHalf.3518" не шифрує себе у файлах. Виводить текст:

A20 Error !!! Press any key to continue ...

"OneHalf.3544.b" не заражає файли AIDS*.*, ADINF*.*, DRWEB*.*, ASD*.*, MSAV*.*. Виводить повідомлення:

Dis is TWO HALF. Fucks any key to Goping...

Cheeba, сімейство

Резидентні безпечні віруси. Активізуються, коли вектор INT 13h вказує на область з адресою меншою, ніж адреса першого MCB. В обробниках INT 13h, 21h, 22h замінюють перші 5 байтів на код "FAR JMP на тіло вірусу", потім записуються в кінець COM- і EXE-файлів. Містять текст:

CHEEBA Makes Уа High Harmlessly F**K THE LAMERS .

У вірусі присутні також коди, які розшифровуються і виконуються при відкритті файлу USERS.BBS, використовуючи ім'я файлу як ключ розшифровки. При цьому вірус записує у файл USERS.BBS якусь інформацію (створює ім'я з максимальними привілеями?).

Bomber

Нешкідливий резидентний полиморфик-вірус. Перехоплює INT 21h і заражає COM-файли, окрім COMMAND.COM, при їх запуску. Містить усередині себе текст:

COMMANDER BOMBER WAS HERE. [DAME]

Характерною рисою цього вірусу є те, що він використовує досить незвичайний поліморфік-алгоритм. При зараженні вірус прочитує 4096 байтів з середини файлу і переносить їх в його кінець. Себе він записує в “диру”, що утворилася, і приступає до генерації поліморфік-коду. Вірус містить декілька підпрограм генерації випадкового (але цілком працездатного!) коду, який записується у випадкові місця файлу, що заражається. У цьому коді може бути присутньою близько 90% всіх інструкцій процесора i8086. Управління з однієї ділянки в іншу передається командами CALL, JMP, RET, RET xxxx. Перша ділянка записується в початок файлу, а остання передає управління на основне тіло вірусу. У результаті заражений файл виглядає як би покритий “плямами” коду вірусу, а процедура виявлення основного тіла вірусу стає надзвичайно складною.

3.9 Способи захисту від вірусів

Насамперед, необхідно відзначити, що захистити комп'ютер від вірусів може лише сам користувач. Тільки систематичне архівування інформації, обмеження ненадійних контактів і своєчасне застосування антивірусних засобів може захистити комп'ютер від зараження або забезпечити мінімальний збиток, якщо зараження все-таки відбулося.

3.9.1 Систематичне архівування важливої інформації

Єдиним стовідсотковим по надійності методом захисту від втрати важливої інформації є її резервне копіювання на захищені від записування пристрої зберігання даних. Більше того, архівуванням також не можна нехтувати, оскільки втратити інформацію можна не лише через віруси, але й через стрибки напруги в мережі, поломки обладнання й т.д.

Жодна антивірусна програма не зрівняється по надійності з архівуванням інформації. Справа в тому, що на будь-який алгоритм антивірусу завжди найдеться алгоритм вірусу, невидимого для цього антивірусу.

3.9.2 Обмеження ненадійних контактів

Друге правило, що частково гарантує збереження інформації, – це обмеження копіювання даних з ненадійних джерел. Як би ми не старалися, обмін інформацією з іншими користувачами і робота в локальних або глобальних мережах неминучі. Однак, деякі правила для себе все-таки виділити можна.

По-перше, необхідно намагатися не запускати неперевірені файли, у тому числі отримані по комп'ютерній мережі. Бажано використовувати тільки програми, отримані з надійних джерел. Перед запуском нових програм обов'язково варто перевірити їх одним або декількома антивірусами.

По-друге, варто обов'язково користуватися тільки тими джерелами та іншими файлами, які добре зарекомендували себе, хоча це не завжди рятує (наприклад, на WWW-сервері Microsoft досить довгий час перебував документ, зараженний макровірусом “Wazzu”).

По-третє, необхідно обмежити коло людей, які допущені до роботи на конкретному комп'ютері. Практика показує, що найбільш уразливі комп'ютери – багатокористувацькі.

І нарешті, відповідно до *четвертого правила*, варто купувати тільки дистрибутивне програмне забезпечення в офіційних продавців. Безкоштовні, умовно безкоштовні або піратські копії можуть призвести до зараження.

3.9.3 Використання антивірусних програм

При існуючому різноманітті вірусів і їх мутацій запобігти зараженню може тільки повнофункціональна антивірусна система, що має в своєму арсеналі всі відомі технології боротьби з «інфекційними хворобами»: не тільки сканер-поліфаг, але і резидентний on-line-монітор, засоби контролю програмної цілісності (CRC) і евристичного пошуку вірусних сигнатур.

Кожен новий вірус необхідно знайти щонайшвидше (а деякі віруси навмисно довго себе не проявляють, щоб у них було досить часу на розповсюдження). Проблема у тому, що немає чіткого способу визначити наперед, що при своєму виконанні дана програма проявить вірусоподібну поведінку. Як немає єдиних ліків від усіх хвороб, так немаэ універсальної «вакцини» від усіх видів шкідливого програмного забезпечення. На всі 100% захиститися від вірусів практично неможливо!

У такій сфері, як виявлення атак на комп'ютерні системи, процес вдосконалення нескінченний. Хакери не втомлюються винаходити всі нові схеми проникнення в комп'ютерні системи. Розробники детектуючих додатків, що стоять по іншу сторону барикад, відстежують новинки, що з'являються, і поспішають запропонувати свої контрзаходи. От чому продукти, що випускаються, вимагають постійної модернізації, і користувачам настійно рекомендується встановлювати оновлені сигнатури, що дозволяють ідентифікувати нові види мережевих атак.

Старе антивірусне програмне забезпечення подібно лікам з минулим терміном придатності – толку від нього мало. Якщо не обновляти файли сигнатур, то рано чи пізно можна опинитися беззахисними проти нових вірусів. Більшість фірм, що розробляють антивіруси, випускає нові файли сигнатур принаймні двічі в місяць або й частіше, якщо з'являється серйозний вірус. Для отримання нових сигнатур зручно користуватися функцією автоматичного оновлення через Web, що є в антивірусному пакеті.

Супровід через Internet програми PC-cillin, наприклад, володіє унікальною особливістю. Можна не тільки отримати консультацію по електронній пошті, але і поговорити у реальному часі з фахівцем служби супроводу Trend. (Хоча від цієї чудової послуги не багато толку, якщо комп'ютер заблокований і увійти в Internet неможливо, проте вона безкоштовна.) Такі антивірусні продукти, як Norton AntiVirus 2000 і McAfee VirusScan, підтримують найкрупніші дослідницькі групи галузі: відповідно Symantec AntiVirus Research Center і AntiVirus Emergency Response Team. Тому Norton і McAfee швидко реагують на загрозу нового вірусу.

Всі основні фірми-постачальники антивірусного забезпечення регулярно і досить часто оновлюють файли сигнатур вірусів, а при появі особливо шкідливого вірусу створюють додатковий екстрений випуск. Ще зовсім недавно вважалося, що сигнатури потрібно оновлювати щомісячно, але в нашу епоху нових вірусів, можливо, буде розумним перевіряти їх

щотижня вручну або за допомогою автоматичного оновлення антивірусної програми. В утилітах McAfee, Symantec і Trend Micro для оновлення достатньо один раз клацнути кнопкою миші.

Певний набір засобів антивірусного захисту присутній у всіх утилітах основних фірм-виробників програмного забезпечення. Серед них: постійний захист від вірусів (антивірусний монітор), перевірка системи за розкладом і оновлення сигнатур через Internet, а також створення аварійної завантажувальної дискети, що дозволяє запустити комп'ютер навіть тоді, коли у нього заражений вірусом завантажувальний сектор (природно, дискету треба створити до того, як вірус потрапив в комп'ютер). Крім цих стандартних засобів, деякі пакети містять «архітектурні надмірності»: наприклад, спеціальний додатковий захист від поштових вірусів (тривога з приводу яких наростає), а також шкідливих модулів ActiveX і Java-аплетов. А такі програми, як Panda Antivirus Platinum і PC-cillin, навіть дозволяють батькам заблокувати доступ дітей до небажаних Web-сторінок.

Оскільки у нових вірусів є нові сигнатури, файли сигнатур необхідно підтримувати в актуальному стані. При виході нової версії антивіруса формат файлу сигнатур звичайно міняється, і оновлені сигнатури виявляються несумісними з попередніми версіями програми. Саме тому антивірусне програмне забезпечення вже досить давно продається по тій же схемі, що бритви і леза: одного разу купивши основну утиліту (бритву), ви потім вимушені постійно купувати оновлені файли сигнатур (леза).

Так, компанії McAfee і Symantec надають право необмеженого оновлення сигнатур протягом року з моменту придбання утиліти, але за кожен наступний рік потрібно в обох випадках заплатити 4 долари. Таку суму навряд можна вважати серйозним ударом по кишені (на відміну від підписки на оновлені файли сигнатур F-Secure, яка коштує 63 доларів); крім того, через рік ми з великою вірогідністю захочемо відновити саму програму. Їх основні конкуренти – Command AntiVirus, Inoculate IT, Panda Antivirus Platinum і PC-cillin – пропонують безкоштовне оновлення сигнатур протягом всього життя продукту.

В даний час способи надання антивірусного захисту істотно змінюються. Компанія McAfee.com вже пропонує перевірку на віруси через Internet в своїй «електронній лікарні» McAfee Clinic (разом з ще декількома видами діагностики). Послуга надається по підписці і коштує 50 доларів в рік, але часто з'являються спеціальні пропозиції, а за перші два тижні платня не береться – це випробувальний період. Перевірку віддалених комп'ютерів на віруси здійснює модуль ActiveX, який бере сигнатури з Web-серверу виробника програми.

3.9.4 Види антивірусних програм

Самими популярними й ефективними антивірусними програмами є антивірусні сканери, монітори, фаги (поліфаги), ревізори. Застосовуються також різного роду блокувальники і іммунізатори (вакцини). Розглянемо характеристики кожного з цих видів програм.

Сканери (scanner)

Сканери (детектори) здатні виявити фіксований набір суттєвих вірусів у файловій системі, секторах і системній пам'яті, а потім – негайно видалити більшість з них. Для пошуку вірусів сканери використовують так звані "маски" (або сигнатуру) – деяку постійну послідовність коду, специфічну для конкретного вірусу.

У випадку, якщо вірус не містить у собі постійної маски (наприклад, поліморфік-вируси), використовуються інші методи, засновані на описі всіх можливих варіантів коду на алгоритмічній мові.

У багатьох популярних сканерах (наприклад Антивірус Касперського, Doctor Web, Norton Antivirus, McAfee, Panda Antivir, AntiVir Personal Edition і ін.) застосовується режим **евристичного сканування**. Цей режим полягає в тому, що програма не просто шукає віруси, а проводить аналіз послідовності команд у кожному об'єкті, який перевіряється, здійснює набір деякої статистики, згодом приймає ймовірне рішення типу: "можливо заражений" або "не заражений".

Евристичне сканування являє собою ймовірнісний метод пошуку вірусів, що, в решті решт, забезпечує можливість визначення невідомих програмі вірусів, але разом з цим збільшує кількість помилкових спрацьовувань (повідомлень, знайдених вірусах у файлах, де насправді їх немає).

Основна ідея такого підходу полягає у тому, що евристика спочатку розглядає поведінку програми, а потім зіставляє його з характерним для зловмисної атаки, на зразок поведінки троянського коня. Встановити модель поведінки і ухвалити рішення щодо нього можна за допомогою декількох механізмів. Для того, щоб виявити і визначити всі можливі дії програми, використовують два підходи: сканування і емуляція.

Підхід зі скануванням припускає пошук «поведінкових штампів», наприклад, найтиповіших низькорівневих способів відкриття файлів. Або процедура сканування звичайного виконуваного файлу проглядає всі місця, де програма відкриває інший файл, і визначає, якого роду файли вона відкриває і що в них записує.

Другий метод визначення поведінки – *емуляція*. Такий підхід дещо складніший. Програма пропускається через емулятор Windows або макроемулятор Macintosh або Word з метою подивитися, що вона робитиме. Проте виникають питання, тому що в цьому випадку багато що залежить від чудасій вірусів. Наприклад, якщо вірус запрограмований на формату-

вання жорсткого диска 25 лютого о 10 годині ранку, а при емуляції цього вірусу на симуляторі дата встановлена на 24 лютого, то вірус поки не проявить свої наміри.

Вся хитрість швидкого розпізнавання полягає в поєднанні двох підходів і отриманні найдокладнішого каталога поведінкових штампів за можливо коротший час. Для перевірки факту зараження файлу вірусом фахівці можуть використовувати різні варіанти штучного інтелекту – експертні системи і нейронні мережі.

Недолік евристичного підходу полягає якраз в його еврістичності. Завжди є вірогідність, що надзвичайно підозрілий файл насправді абсолютно нешкідливий. Проте останній евристичний механізм Symantec під назвою Bloodhound дозволяє знайти до 80% невідомих вірусів виконуваних файлів і до 90% невідомих макровірусів. Варто також помітити, що програми-детектори не дуже універсальні, оскільки здатні знайти тільки відомі віруси. Деяким таким програмам можна повідомити спеціальну послідовність байт, характерну для якогось вірусу, і вони зможуть знайти інфіковані ним файли: наприклад, це уміють NotronAntiVirus або AVP-сканер.

Різновидом сканерів є так звані **таблетки** – спеціалізовані програми, орієнтовані на пошук певного типу або сімейства вірусів, наприклад, троянів, макровірусів та інших (наприклад, Anti-Trojan, Trojan Remover).

Слід зазначити, що використання спеціалізованих сканерів, розрахованих тільки на макровіруси, іноді буває більше зручним і надійним рішенням для захисту документів MS Word і MS Excel.

До недоліків сканерів варто віднести тільки те, що вони охоплюють далеко не всі відомі віруси й вимагають постійного відновлення антивірусних баз. З огляду на частоту появи нових вірусів і їх короткий життєвий цикл, для використання сканерів необхідно налагодити одержання свіжих версій не рідше одного-двох разів на місяць. В іншому випадку їхня ефективність істотно знижується.

Монітори

Монітори – це різновид сканерів, які, постійно перебуваючи в пам'яті, відслідковують вірусоподобні ситуації, які відбуваються з диском і пам'яттю (тобто виконують безперервний моніторинг). Прикладом таких антивірусів може бути програма Kaspersky Anti-Virus або SpiDer Guard.

До недоліків цих програм можна віднести, наприклад, імовірність виникнення конфліктів з іншим програмним забезпеченням, як і для сканерів – залежність від нових версій вірусних баз, а також можливість їхнього обходу деякими вірусами.

Фаги (поліфаги) (scanner/cleaner, scaner/remover)

Фаги – це програми, здатні не тільки знаходити, але і знищувати віруси, тобто лікувати «хворі» програми (поліфаг може знищити багато ві-

русів). До поліфагів відноситься і така стара програма, як Aidstest, яка знаходить і знешкоджує близько 2000 вірусів.

Основний принцип роботи традиційного фага простий і не є секретом. Для кожного вірусу шляхом аналізу його коду, способів зараження файлів і т.д. виділяється деяка характерна тільки для нього послідовність байтів – сигнатура. Пошук вірусів в простому випадку зводиться до пошуку їх сигнатур (так працює будь-який детектор).

Сучасні фаги використовують інші методи пошуку вірусів. Після виявлення вірусу в тілі програми (або завантажувального сектора, який теж містить програму початкового завантаження) фаг знешкоджує його. Для цього розробники антивірусних засобів ретельно вивчають роботу кожного конкретного вірусу: що він псує, як він псує, де він ховає те, що зіпсує (якщо ховає). В більшості випадків фаг може видалити вірус і відновити працездатність зіпсованих програм. Але необхідно добре розуміти, що це можливо далеко не завжди.

Ревізори

Ревізори – це програми, принцип роботи яких заснований на підрахунку контрольних сум (CRC-сум) для присутніх на диску файлів і системних секторів.

Прикладом такого антивірусу може бути програма ADinf32. Ці контрольні суми потім зберігаються в базі даних антивірусу (у таблицях) разом із відповідною інформацією: довжинами файлів, датами їх останньої модифікації і т.д. При наступному запуску ревізори звіряють відомості, що містяться в базі даних, з реально підрахованими значеннями. Якщо інформація про файл, записана в базі даних, не збігається з реальними значеннями, то ревізор попереджає про те, що файл, можливо, був змінений або заражений вірусом.

Ревізори вміють вчасно виявляти зараження комп'ютера практично кожним з існуючих на сьогодні вірусів, не допускаючи розвитку епідемії, а сучасні версії ревізора вміють негайно видаляти більшість навіть раніше незнайомих їм вірусів.

До недоліків ревізорів можна віднести те, що для забезпечення безпеки вони повинні використовуватися регулярно. Але безсумнівними їхніми перевагами є висока швидкість перевірок і те, що вони не вимагають частого відновлення версій.

Антивірусні блокувальники

Антивірусні блокувальники – це резидентні програми, які перехоплюють небезпечні ситуації, і повідомляють про це користувача (наприклад, AVP Office Guard). До ситуацій, що відслідковуються, належать, наприклад, відкриття виконуваних файлів для записування і записування в boot-сектори дисків або MBR вінчестера, спроби програми залишитися резиден-

тною і т.д. Доречі, відзначені події харектерні для вірусів у моменти їх розмноження.

Блокувальники дозволяють обмежити розповсюдження епідемії, поки вірус не буде знищений. Практично всі резидентні віруси визначають факт своєї присутності в пам'яті машини, викликаючи яке-небудь програмне переривання з «хитрими» параметрами. Якщо написати просту резидентну програму, яка імітуватиме наявність вірусу в пам'яті комп'ютера, правильно «відзиваючись» на певний пароль, то вірус, швидше за все, визнає цю машину вже зараженою.

Навіть якщо деякі файли на комп'ютері містять в собі код вірусу, при використанні блокувальника зараження всієї решти файлів не відбудеться. Для нормальної роботи такої програми необхідно запустити блокувальник раніше всієї решти програм, наприклад, у файлі CONFIG.SYS. Але якщо вірус встиг заразити COMMAND.COM або стартує із завантажувального сектора, то антівірус-блокировщик не допоможе.

До переваг блокувальників можна віднести вміння виявляти вірус на самій ранній стадії його розмноження, а до недоліків – здатність деяких вірусів обходити блокувальники, а також наявність неправдивих спрацьовувань.

Іммунізатори або вакцини

Іммунізатори – це невеликі програми, які змінюють файли або проникають у них. У першому випадку вірус буде приймати файли як заражені, а в другому – антивірус буде щоразу перевіряти файл на зміни. Слід зазначити, що в наш час цей тип антивірусів не має великого расповсюдження серед користувачів.

Спеціальні вакцини призначені для обробки файлів і завантажувальних секторів. Вакцини бувають пасивними і активними.

Активна вакцина, «заражаючи» файл, подібно вірусу, оберігає його від будь-якої зміни і у ряді випадків здатна не тільки знайти сам факт зараження, але і вилікувати файл. *Пасивні вакцини* застосовують тільки для запобігання зараженню файлів деякими вірусами, що використовують прості ознаки їх зараженості – «дивні» час або дату створення, певні символьні рядки і т.д. В даний час вакцинація широко не застосовується. Бездумна вакцинація всього і всіх здатна викликати цілі епідемії неіснуючих вірусних хвороб. Так, протягом декількох років на території колишнього СРСР лютувала страшна епідемія жахливого вірусу TIME. Жертвою цього вірусу стали сотні абсолютно здорових програм, оброблених антивірусною програмою ANTI-KIT.

Наведемо приклад. В даний час існує досить багато вірусів, що запобігають повторному зараженню файлів деякою «чорною міткою», якою вони мітять інфіковану програму. Існують, наприклад, віруси, що виставляють в полі секунд часу створення файлу значення 62.Уже досить давно з'явився вірус, який до всіх заражених файлів дописував п'ять байт –

MsDos. Нормальних файлів, що містять в кінці такий символьний рядок, не буває, тому вірус і використовував цю ознаку як індикатор зараження файлу. Вакцинація файлів проти такого вірусу зовсім не складна. Достатньо дописати в кінець вище згаданий символьний рядок – і зараження таким вірусом не страшне. Страшне інше – деякі антивірусні програми, зустрівши в кінці файлу нещасливу строчку, починають негайно лікувати його. Шансів на те, що після такого «лікування» «інвалід» нормально працюватиме, практично ніяких.

3.9.5 Огляд найпоширеніших антивірусних програм

Для захисту інформації на комп'ютерах системі необхідний антивірусний сканер, який би зупиняв більшість інфекцій, що потрапляють в комп'ютер через електронну пошту і файли, викачувані з Інтернету. Знешкоджувати цю інфекцію може тільки антивірусний сканер. Отже, необхідно мати в своєму арсеналі надійні антивірусні програмні продукти.

Всі ці продукти задовольняють мінімальним вимогам по "вилову" основних зловмисних програм, що зустрічаються в Інтернеті. Адже далеко не кожним вірусом дійсно можна заразитися через Глобальну Мережу: з приблизно 100000 існуючих вірусів в даний час всього близько 250 зустрічаються в Інтернеті, інші існують тільки в "лабораторних" умовах. Виявилося, що якість роботи різних сканерів суттєво залежить від типу вірусів. Наприклад, всі відібрані для огляду продукти добре справляються з вірусами і черв'яками, діючими в 32-розрядному середовищі Windows – найпоширенішим сьогодні типом мережевої "інфекції". В цьому випадку якість розпізнавання дуже висока – від 90,4 до 100%.

Проте з троянськими програмами, які розповсюджуються не самі по собі, а за допомогою інших програм, зокрема вірусів і "черв'яків", справа йде гірше: якщо сканери McAfee і Norton затримують відповідно 99% і 95% "троянів", то AVG – всього 23,5%, що навряд чи можна вважати достатнім захистом. Нарешті, дуже бажано, щоб антивірусний сканер не піднімав помилкову тривогу, підозрюючи у наявності вірусу звичні файли. У цьому значенні краще всього йдуть справи у PC-cillin, де тест пройшов без помилкових спрацьовувань; на іншому кінці шкали – Eset NOD32, який помилково спрацював на 32 з 20000 файлів. Звичайно, це настільки мало, що навіть не варто перераховувати у відсотках, але і одного такого випадку досить, якщо в результаті замість вірусу буде видалений потрібний файл. Отже, наявність евристичного аналізатора також є важливим при виборі антивірусної програми.

За даними тестів найвдалішими евристичними алгоритмами володіють McAfee і AVG, ідентифікуючі 70,1% і 65,6% файлів, заражених невідомими вірусами; гірший результат опинився у NOD32 — 41,4%. І у будь-якому випадку, це набагато менше і повільніше, ніж при розпізнаванні ві-

домих вірусів, так що основним засобом своєчасного лікування як і раніше залишається регулярне оновлення антивірусних баз.

Всі програми тестувалися в режимі максимально ретельного сканування. Особливий випадок був NOD32: у цій програмі передбачений підвищений в порівнянні з жорстким диском рівень евристичного сканування Advanced Heuristics для електронної пошти і веб-трафіку, цих основних джерел інфекцій. Для сканування жорсткого диска цей режим теж можна використовувати, але не засобами графічного інтерфейсу, а за допомогою недокументованої команди nod32.exe /AH. У режимі Advanced Heuristics якість сканування NOD32 зростає до 53,5%.

На жаль, всі розглянуті антивіруси успішно розпізнають нові інфекції тільки в тому випадку, якщо останні належать до вже відомих вірусних "сімейств", але виявляються безсилі при зустрічі з абсолютно новим вірусом. Наприклад, жоден з сканерів не розпізнав черв'яка Netsky, поки його сигнатура не була внесена в базу даних.

McAfee

Якби головною в антивірусному сканері була зручність інтерфейсу, то кращим пакетом напевно був би визнаний McAfee (рис.11). Правда, в іншому у McAfee багато недоліків – наприклад, помилки в сценарії оновлення, в результаті яких програма повідомляє, що сканер вже був оновлений, тоді як насправді це не так.

Рисунок 11– Вигляд головного вікна програми McAfee

McAfee, звичайно, не єдиний пакет з "глюками" – хоча їх у нього і багато. Жодна з розглянутих програм не пройшла гладко тест на видалення вірусу CTX, "троянського коня" Optix і червя Mydoom.A. Краще за всіх справився з очищенням комп'ютера PC-cillin — інфекція була повністю видалена, а система не пошкоджена. Після спроб McAfee, NOD32 і Panda видалити CTX (безуспішно) система прийшла в непридатність.

PC-cillin

Одним з кращих антивірусів був визнаний пакет Trend Micro PC-cillin Internet Security 2004: ця програма відрізняється не тільки якісним скану-

ванням і розумною ціною, але також логічним і зрозумілим інтерфейсом. PC-cillin – це антивірус і мережевий фільтр, покликані забезпечувати безпеку персонального комп'ютера (рис.12).

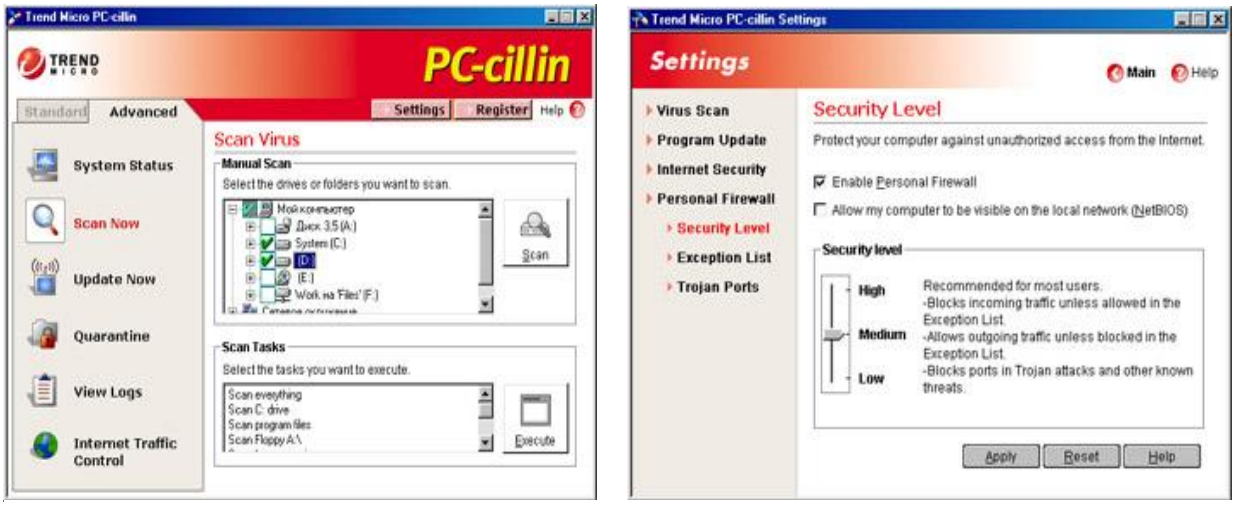


Рисунок 12 – Вигляд програми PC-Cillin у роботі

Антивірусна складова. У цієї складової пакету PC-cillin є все, що повинен мати в своєму розпорядженні сучасний антивірус. При скануванні і моніторингу можна використовувати фільтр по розширенню файла. Моніторинг можна швидко відключити/включити, клацнувши по іконі програми на панелі і вибравши опцію "Real-time Scan". Вхідна пошта (POP3) перевіряється на предмет безпеки вкладень, при цьому віруси будуть видалені навіть з архівних файлів. Вбудований планувальник задач дозволяє запланувати перевірку окремих дисків і тек з певною періодичністю. Перевіряти на віруси можна окремі теки або окремі види файлів, наприклад, тільки .doc.

Файрвол. Мережевий фільтр в PC-cillin організований так, щоб настроювати його було зручно користувачам будь-якого рівня. Досягається це тим, що "на поверхні" знаходяться прості і загальні настройки, а детально набудувати брандмауер можна, вибравши правило із списку на вкладці "Exeption List" і натиснувши на "Edit".

Internet Traffic Control. Закладка, на якій відстежується вхідний і витікаючий трафік. Тут же можна включити опцію "Wi-Fi Protection", актуальну при роботі в локальних мережах, в безпеці яких ви не упевнені.

Internet Security. Закладка, що знаходиться в меню "Settings". Тут можна заборонити виконання Java і ActiveX програм на web-сторінках. Можна набудувати фільтр, що блокує можливість переглядання сайтів.

Panda Antivirus Platinum

Одне з непоганих рішень для комплексного забезпечення безпеки комп'ютера – це Panda Antivirus Platinum.

Інтерфейс цієї програми побудований так, щоб дати можливість найледачішому користувачу без труднощів знайти необхідне (рис.13). Антивірус традиційно показує високу швидкість сканування. Монітор гідний усіляких похвал, і його присутність в пам'яті ніяк не відображається на продуктивності комп'ютера в цілому.

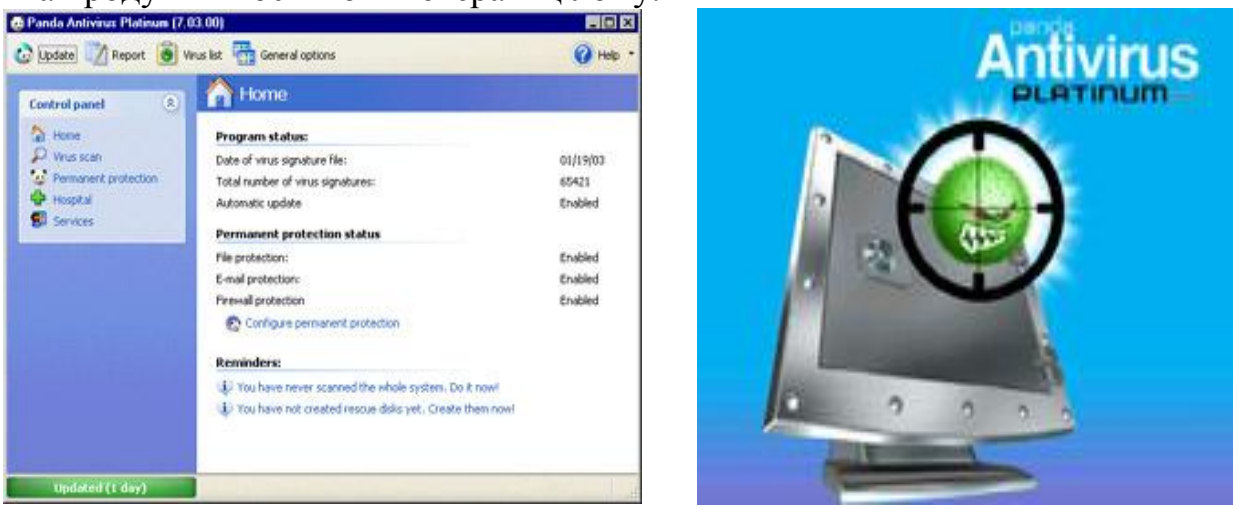


Рисунок 13 – Головне вікно і логотип програми Panda

Однією з переваг програми Panda є *комплексний захист*. Виробники антивірусного програмного забезпечення останнім часом прагнуть наділяти свої продукти декількома різносторонніми сервісами, а не тільки антивірусними засобами. Дана програма є хорошим співвідношенням продуктивність/безпека.

Firewall і проксі-сервер. Panda Antivirus Platinum має дуже простий в настройці брандмауер, який орієнтований на виявлення спроб отримання доступу в Мережу невідомими додатками. Щоб грамотно набудувати Firewall, користувачу не потрібне практично нічого знати про те, як працюють сучасні мережі. Проте, якщо необхідно детально набудувати фільтр, у розпорядженні є все необхідне для цього, навіть фільтрація по ознаці mac-адреси! Після установки пакету весь поштовий трафік прямує через локальний проксі-сервер Panda, де відбувається його перевірка, відсікання вірусів і при відповідній настройці потенційно небезпечних вкладень. Доречі, про цей факт користувачу також зовсім не обов'язково знати, сервіс чудово функціонує з настройками за замовчуванням.

Digital Patrol

Крім шпигунських модулів, життя користувачу можуть сильно зіпсувати троянські віруси. Ці шкідливі програми непомітно господарюють на наших ПК, а десь в далекій країні сидить собі такий Карабас Барабас і радіє з того, що до нього присилаються всі ваші паролі. Хоча це ще не гірша з можливих бід. Набагато гірше буде, якщо вірус "руйнуватиме" всю систему цілком. Щоб такого на відбулося, можна скористатися цією ути-

літою, що виявляє всі види троянських вірусів, черв'яків і скрипт-вірусів. Чому саме їх? Так тому що до досвідченого користувача, який ніколи не запускає файли, прислані поштою невідомими "доброзичливцями" і відразу видаляє їх, звичному вірусу потрапити досить важко.

Принцип роботи Digital Patrol такий же, як і у численних антивірусних програм. Працювати з сканером дуже просто: потрібно вибрати область сканування і натиснути на кнопку Пуск (рис.14).

Рисунок 14 – Логотип та головне вікно програми Digital Patrol

Набагато більшу цінність надає антивірусний монітор, що входить в комплект програми, і дозволяє відловлювати шкідливі програми до того, як вони встигнуть наламати дрів. Хоча програма і небезкоштовна, істотних обмежень в ній знайти не вдалося, хіба що недоступна опція, що включає евристичний аналіз. У іншому Digital Patrol повністю функціональна, добре працює оновлення антивірусної бази.

Інші програми-антивіруси

Найбільше поширення набули програми DrWeb і AVP. Дякуючи своїм новітнім детекторам, вони можуть знайти будь-які віруси: як найстаріші, так і ті, що тільки з'явилися. Ще потрібно згадати детектор ADinf. Ця антивірусна програма знаходить всі віруси, що не змінюють довжину файлів, невидимі віруси і багато інших. Таким чином, ці три програми забезпечують потужній захист проти вірусів. Всі ці програми можна вписати у файл AUTOEXEC.BAT, тоді при завантаженні комп'ютера перевірка на зараження вірусом проводитиметься автоматично. Доречі, на Заході теж вважають за краще користуватися такими російськими програмами, як DrWeb і AVP.

Кілька років тому детектори практично поступилися своїми позиціями програмам, званих поліфагамі, але сьогодні вони знов повертаються на комп'ютерний ринок. Такі антивірусні продукти, як Norton AntiVirus і McAfee VirusScan, підтримують найкрупніші дослідницькі групи галузі: відповідно Symantec AntiVirus Research Center і AntiVirus

Emergency Response Team.Поэтому Norton і McAfee виключно швидко реагують на загрозу нового вірусу.

Щоб понизити число дорогих телефонних консультацій, антивірусні компанії прагнуть підсилити підтримку через Internet. Але іноді все-таки буває потрібно поставити питання живій людині. Якщо купимо продукт, то, швидше за все, одержимо і супровід по телефону і через Internet безкоштовно, а у разі безкоштовних пакетів підтримка коштуватиме грошей.

Розповсюдження протиотрути в рамках корпоративного середовища повертає нас назад до питання часу реакції з боку постачальника антивірусних засобів. Звичайно, доставка сигнатур нових вірусів на настільні системи надзвичайно важлива, але для того, щоб їх поширювати, ці сигнатури треба спочатку одержати. У ідеалі їх хотілося б мати до того, як наш комп'ютер буде атакований. Звичайно постачальник антивірусних засобів дає відповідь на протязі не більш двох діб після надання йому підозрілого файла (термін в 48 годин став, по суті, стандартною верхньою межею). Проте в наші дні 48 годин – це дуже довго, за цей час можуть бути інфіковані тисячі робочих станцій. Компанія Symantec, наприклад, вже сьогодні має в своєму розпорядженні подібну систему для макровірусів, т.е.вирусов Word for Windows і Excel. Їй потрібно дві години, щоб надати набір визначень для протидії новому вірусу (якщо він ніколи раніше не зустрічався), – і все це винятково за допомогою комп'ютера.

Контрольні питання

1. Класифікуйте комп’ютерні віруси за середовищем їх існування та за способом зараження комп’ютерів.
2. Наведіть класифікацію вірусів за алгоритмами, які вони використовують при функціонуванні, та за своїми деструктивними можливостями?
3. З чого складається класифікаційний код вірусу?
4. Що таке дескриптор та сигнатура вірусів?
5. В чому полягають особливості файлових вірусів, якими вони бувають?
6. Де можуть бути розташовані файлові віруси?
7. Яким буває класифікаційний код файлового вірусу і які його складові?
8. Охарактеризуйте дескриптор та сигнатуру файлового вірусу.
9. Які різновиди файлових вірусів ви знаєте? Дайте характеристику "overwriting"-вірусам?
10. В чому полягає принцип функціонування та розташування паразитичних вірусів?
11. В чому різниця між вірусами типу "prepending", "appending" і "inserting"?
12. Як працюють віруси-компаньйони? В чому їх особливості?
13. Що таке файлові черв’яки? Наведіть відомі вам приклади файлових вірусів-черв’яків.

14. Link-віруси та їх особливості.
15. Охарактеризуйте групу OBJ- і LIB-вірусів. Де вони розташовуються і як себе проявляють?
16. Наведіть алгоритм роботи файлових вірусів.
17. Які можливості відновлення файлів, що заражені файловим вірусом?
18. Дайте означення завантажувального вірусу. В чому його особливості?
19. Яким може бути класифікаційний код бутівського вірусу? Наведіть приклади і поясніть значення кожної складової.
20. Що може містити дескриптор та сигнатура бутівського вірусу?
21. В чому полягає принцип дії завантажувального вірусу?
22. Наведіть можливі місця розташування завантажувальних вірусів.
23. Наведіть алгоритм роботи завантажувального вірусу: резидентного і нерезидентного.
24. Дайте загальну характеристику макро-вірусам, їх особливостям та розташуванню.
25. Якими можуть бути причини зараження макро-вірусами?
26. Які принципи роботи і алгоритм функціонування Word-вірусів?
27. Які принципи функціонування Excel-вірусів?
28. Як працюють Access-віруси і як вони себе проявляють?
29. В чому особливості мережних вірусів?
30. Що таке IRC-черв'яки і які їх різновиди ви знаєте?
31. Що таке IRC-сервери та IRC-клієнти?
32. Охарактеризуйте стелс-віруси. Які різновиди цих вірусів ви знаєте?
33. Як проявляють себе завантажувальні стелс-віруси?
34. В чому особливість файлових стелс-вірусів?
35. Що таке макро-стелс-віруси?
36. Які віруси відносять до групи поліморфічних вірусів?
37. Наведіть і охарактеризуйте рівні поліморфік-вірусів і можливості детектування вірусу на кожному з рівнів.
38. Дайте характеристику такому способу поліморфізму, як зміна виконуваного коду.
39. Що таке поліморфні розшифровувачі?
40. Наведіть перелік основних прийомів для захисту операційних систем від вірусів.
41. Доведіть необхідність систематичного архівування інформації та обмеження ненадійних контактів
42. Для чого існує антивірусне програмне забезпечення?
43. Що таке програми-сканери та які їх основні характеристики?
44. В чому полягає особливість такиї антивірусних програм, як таблетки?
45. Яка особливість антивірусних програм-моніторів?
46. Що таке програми-ревізори, як вони працюють?
47. Дайте характеристику антивірусним блокувальникам та програмам-імунізаторам.

4 НАЙПРОСТІШІ МЕТОДИ ЗАХИСТУ ІНФОРМАЦІЇ В ОПЕРАЦІЙНИХ СИСТЕМАХ

4.1 Установки і налаштування системи захисту

Існує багато установок і налаштувань, що забезпечують деякі додаткові можливості системи захисту Windows. Усі вони безкоштовні та легко реалізуються. Не зважаючи на їх простоту, не слід ігнорувати такими методами захисту.

Періодичне очищення меню Документи (Documents)

Це меню містить перелік файлів (приблизно 15 найменувань), з якими ми нещодавна працювали (рис.15). А це означає, що будь-який сторонній користувач може без проблем переглянути результати нашої роботи або наш особистий файл, навіть не використовуючи будь-якого спеціального пошуку.

Для очищення цього меню слід виконати такі дії: *Пуск – Настройка - Панель задач и меню «Пуск» - Настройка меню-Очистить.*

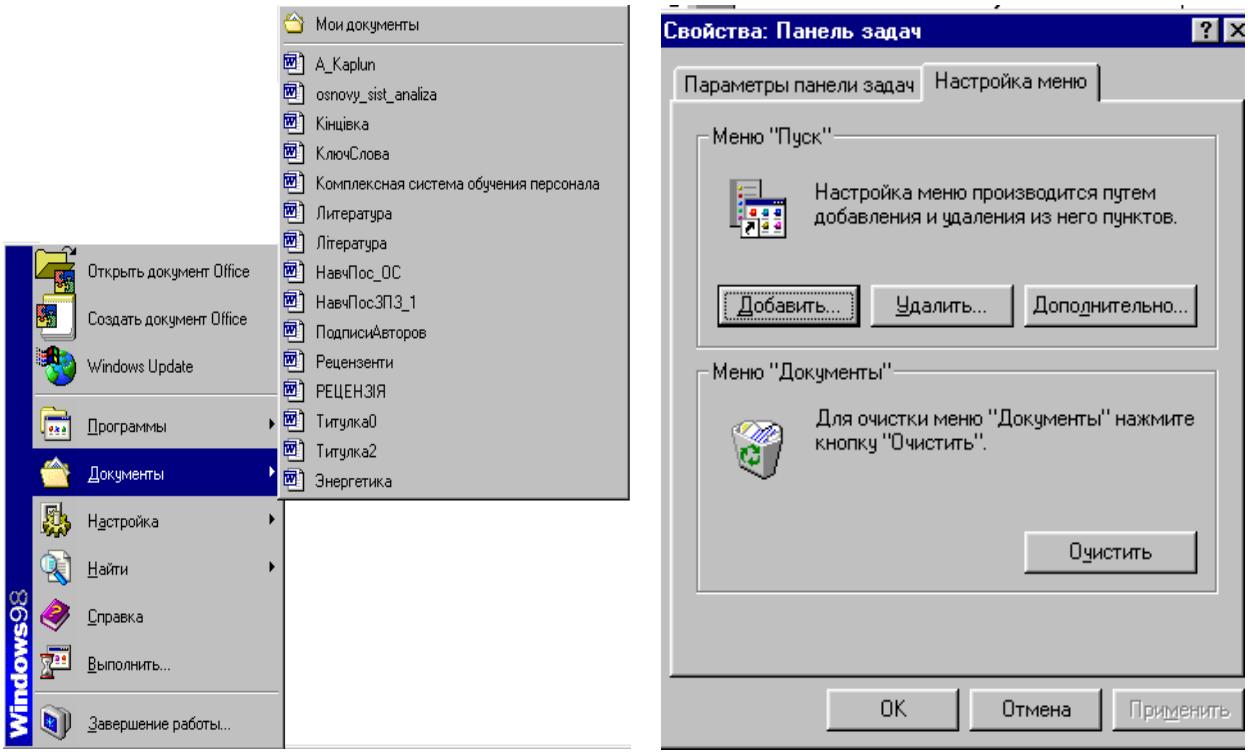


Рисунок 15 – Вигляд меню Documents та вікна очистки

Після очистки цього меню зловмисникам прийдеться витратити більше часу для пошуку наших файлів.

Очистка і установка Корзини (Recycle Bin)

Корзина – улюблене місце для зловмисників, які хочуть добути якусь інформацію. Вона зберігає для використання будь-які файли, видалені за допомогою програми *Проводник* (Windows Explorer). Є три способи розв'язати цю проблему.

1. Перший спосіб полягає у тому, щоб очищати корзину кожен день або навіть частіше. Зробити це можна за допомогою послідовності команд:
   Корзина – Очистить корзину.
2. Встановити властивості Корзини так, щоб файли видалялися з неї безперервно (рис.16), для чого виконати таку послідовність команд:
   Корзина – Свойства – Глобальные – Уничтожать файлы сразу после ... - OK.

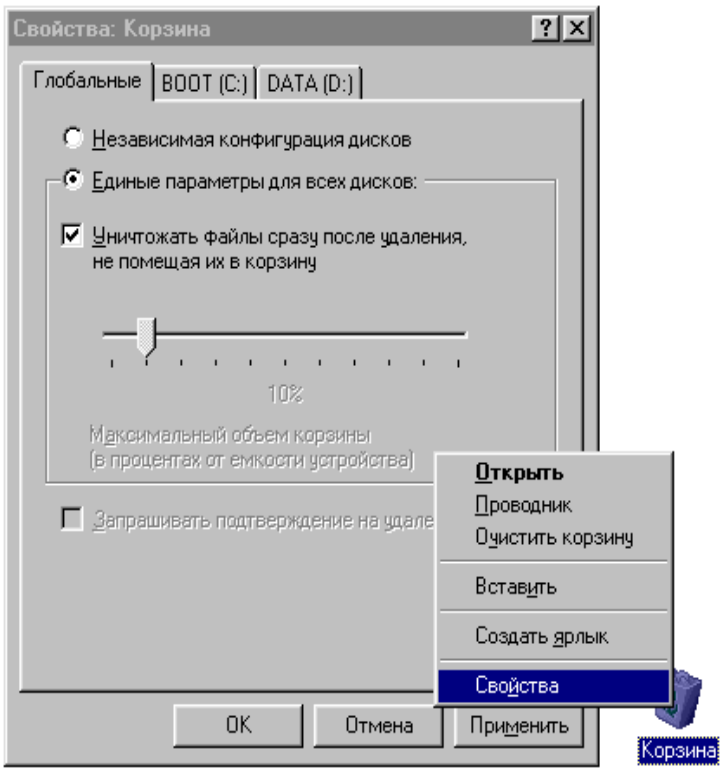


Рисунок 16 – Вікно встановлення властивостей Корзини

3. Видалити файли засобами DOS. Для цього виконати команди:
   Пуск – Програми – Сеанс MS-DOS

Далі перейти в каталог, що містить файли, які ми хочемо видалити і командою DEL видалити їх. Але слід пам'ятати, що такі файли відновити вже не так просто, якщо це взагалі можливо.

Можна заставити зловмисника повірити, що ми не видаляємо файли з корзини, оскільки нам нічого приховувати. Для цього слід видаляти лише найбільш вразливі файли (видаляти з Корзини тільки ці файли або видаляти їх засобами DOS). Складатиметься враження, що ми нічого навмисно не видаляємо, тобто,нам нічого приховувати. Можливо, це зупинить зловмисника від подальших дій.

Видалення або перейменування ярликів

Багато хто з користувачів, отримавши доступ до комп'ютерної системи, відразу ж починають клацати по значках ярликів на Робочому столі (Desktop) Windows. Можна запобігти цьому, видаляючи і/або перейменовуючи ці значки. Тоді зловмисник може не звернути увагу на програму,

ім’я якої в нього не викликає ніякого зацікавлення. Але при цьому самому слід пам’ятати нові назви ярликів.

Для видалення або перейменування значків на Робочому столі треба клацнути правою клавішею миші на вибраному ярлику і вибрати з контекстного меню, що з’явиться при цьому, пункт *Удалить* – для видалення його, пункт *Переименовать* – для перейменування.

Видалення і перейменування пунктів меню Start

Пункти меню Пуск (Start) легко можна видалити або змінити, і цими можливостями можна скористатися для запобігання деяких дій зловмисників щодо нашої інформації.

Для видалення слід виконати такі команди (рис.17):

Пуск – Панель задач и меню “Пуск” – Свойства – Настройка меню – Удалить,

вибравши зі списку ту програму або папку, яку треба видалити. При цьому компоненти, що видалені, лишаються на вінчестері. Отже, запустити програми все ж таки можна. Крім того, можна запустити програму на виконання, або скориставшись командами *Пуск – Выполнить,* або за допомогою ярликів Робочого столу, або за допомогою програми *Проводник.*

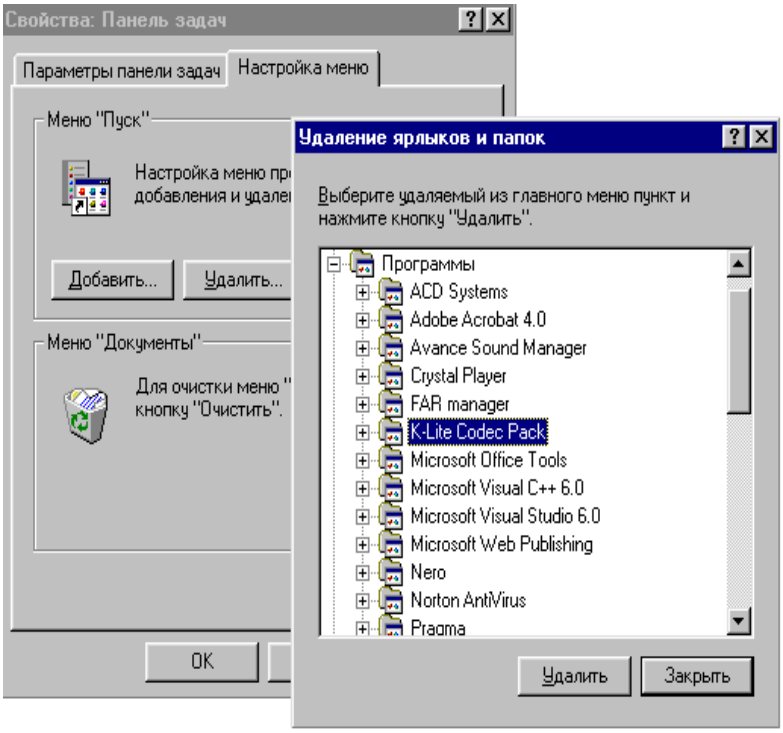


Рисунок 17 – Діалогове вікно видалення ярликів

Для перейменування пункта меню Пуск слід вибрати на вкладці Панель задач кнопку *Дополнительно* і перейменувати потрібний файл або папку за допомогою контекстного меню.

Приховування Панелі задач

У Панелі задач є параметр – *Автоматически убрать с экрана* (Auto Hide), який може дещо спантеличити недосвідченого зловмисника і не привернути увагу стороннього користувача. Цей параметр робить Панель задач практично невидимою (вона ніби виштовхується зі звичайної області перегляду, хоча і буде з’являтися, якщо курсор миші пройде над її областю розташування). Для установки цього параметра слід виконати такі дії:

Пуск – Настройка – Параметры панели задач – Автоматически убирать с экрана - ОК.

Захист від зміни і видалення файлів і папок

Якщо ми не хочемо, щоб хтось випадково або навмисно видалив або змінив наші файли, можна дещо застрахуватись від цього. Для цього використовують програму *Проводник*, де за допомогою контекстного меню можна встановити властивості файла так, що він стане недоступним для зміни.

Атрибут *Только чтение* (Read-only) дозволить запобігти спробу користувачів відредагувати і зберігти файл за допомогою додатків Windows або DOS та видалити за допомогою DOS. Хоча за допомогою програми Проводник видалити його все ж таки можна. Аналогічно поступають і з папками.

Атрибут *Скрытый* (Hidden) приховує файл або папку для перегляду програмами Windows, DOS і командами DOS для роботи з каталогами.

Звичайно, всі наведені вище міри захисту є надзвичайно ненадійними, і більш-менш досвідчений користувач з легкістю їх подолає. Це лише спроба утруднити роботу зловмисника. для більш надійного захисту слід використовувати і більш надійні інструменти.

Резервне копіювання системи та шифрування дисків

Для того, щоб захиститись від того, що сторонній користувач навмисно або випадково не знищить нашу інформацію, в систему попадк вірус або станеться повний збій системи, необхідно гарантувати збереження наших даних, тобто час від часу виконувати резервне копіювання системи. При установці програмного забезпечення рекомендується створити аварійний або системний диск. На цьому диску повинна знаходитись інформація, необхідна операційній системі для запуска комп’ютера з зовнішнього носія, а також деяа інформація про конфігурацію комп’ютера.

Звичайно, аварійний диск не допоможе відновити файли даних, створені за допомогою різних програмних додатків – текстових процесорів, редакторів електронних таблиць, графічних редакторів тощо. Для всіх цінних файлів даних слід використовувати резервне копіювання (архівацію) на інших носіях – на гнучких дисках, на CD-дисках, на ZIP-дисках.

Ще з часів DOS існують програми, що дозволяють створювати захищені диски, вміст яких стає доступним лише після того, як користувач введе правильний пароль.

Отже, створення зашифрованих дисків – це ще один з найпростіших методів захисту власної інформації, і застосуванням програм, призначених для цього, також не треба нехтувати.

4.2 Блокування доступу до комп'ютера за допомогою екранної заставки Windows

Існує певний клас програм, здатних захистити комп'ютер від мережевих атак; налаштування ОС не дадуть використати "проколи" розробників (так звані "дірки" у програмному забезпеченні). Але всі ці прийоми не врятують від елементарної людської дурості і явного халатного відношення до захисту власного комп'ютера, що в більшості випадків приводить хакера до потрібного йому результату, тобто до успішного зламу. Але сподіватись на вдачу (тобто на відсутність будь-якого інтересу до ПК з боку зловмисників) щонайменше необережно, тому варто знати, як встановити найпростіше блокування у вигляді програм-заставок з паролем (і подібних їм), щоб грамотно організувати захист свого комп'ютера.

Для того щоб захиститься від зламу, потрібно знати, яким чином буде діяти зламщик, спробувати зрозуміти логіку його вчинків і, спираючись на отримані знання, грамотно організувати оборону. Отже, основною функцією програми-заставки є так зване "запирання" монітора, тобто блокування доступу до комп'ютера.

В операційній системі Linux опція "запирання" екрана передбачена за замовчуванням. Тобто перед тим, як користувач відійде від комп'ютера, він може активізувати заставку, що допускає відображення діалогового вікна із пропозицією ввести пароль користувача для продовження роботи (якщо стороння людина спробує забрати заставку рухом миші або натисканням будь-якої клавіші).

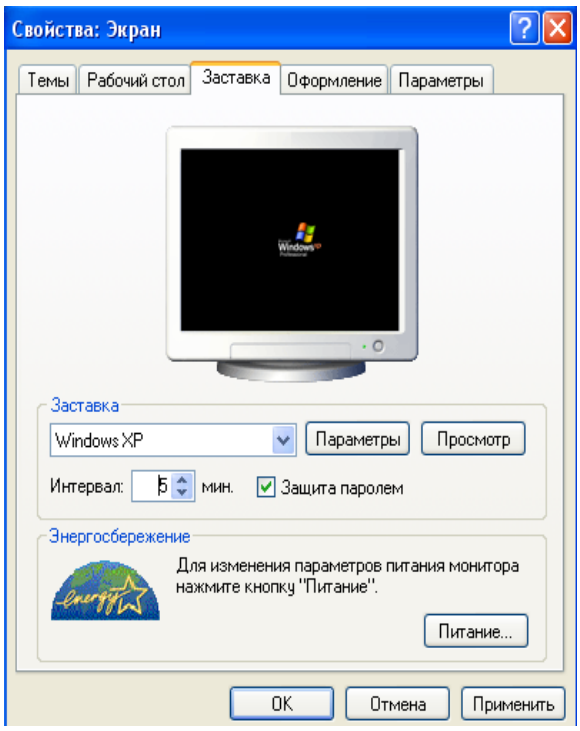


Рисунок 18 – Активація програми-заставки

В Windows ця функція активізується установкою певної опції. Для активізації заставки з паролем необхідно відкрити "Панель управління" (команди *Пуск – Настройка - Панель управления*), у якій подвійним натисканням клавіші миші вибрати пункт *Екран*. У результаті відобразиться діалогове вікно властивостей екрана. Інший спосіб активізації цього вікна – натисканням правої кнопки миші по вільній поверхні Робочого стола викликати контекстне меню і вибрати пункт *Свойства*. На закладці *Заставка* варто вибрати бажану заставку, вказати пароль за допомогою кнопки *Изменить* і встановити прапорець *Пароль* (рис.18). Далі потрібно ввести пароль. Для роз-

блокування комп'ютера, що захищається за допомогою заставки з паролем, необхідно перезавантажити комп'ютер. Після цього заставка пропадає, а для обходження вхідного пароля в Windows 95/98 досить натиснути кнопку *Отменить*. Тому для захисту в Windows 95/98 доведеться використати спеціальні програми.

У системах Windows NT/2000/XP разом із програмою-заставкою варто використати вхідний пароль, щоб навіть після перезавантаження зловмисник не зміг одержати доступ до потрібної інформації.

Перезавантаження – не єдиний метод боротьби із заставкою. На жаль, він не позбавлений недоліків. Один з них полягає в тому, що програма запускається тільки через певний інтервал часу, коли з комп'ютером не здійснюють ніяких дій. Цього інтервалу цілком може вистачити зламщику, щоб отримати доступ до комп'ютера, поки заставка не спрацювала. Вирішується ця проблема так: у закладці *Заставка* натискаємо кнопку *Просмотр*, після чого заставка функціонує в нормальному режимі.

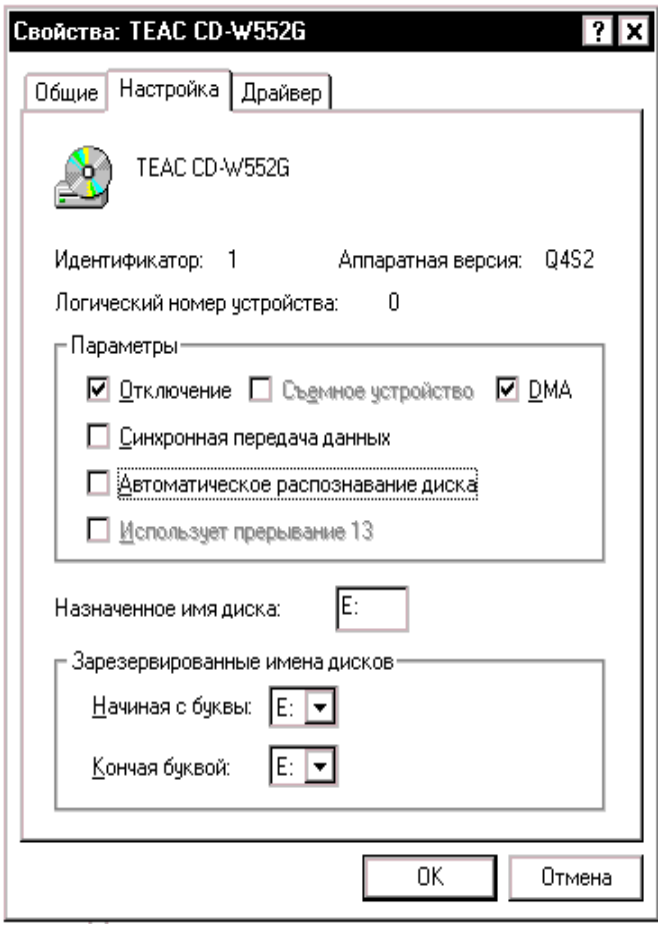


Рисунок 19 – Відключення Autorun

Наступний недолік більш істотний. Програма-заставка не відключає *Autorun*. Зловмисник може вставити в CD-ROM диск зі шкідливою програмою, що запуститься за допомогою *Autorun*. Для рішення даної проблеми необхідно відключити автозапуск. Для цього в *Панелі керування* подвійним натисканням клавіші мишки розкрити розділ *Система*, де на закладці *Оборудование* (рис.19) у властивостях параметра *Устройство чтения компакт-диска* слід зняти прапорець *Автоматическое распознавание диска*.

4.3 Використання пароля BIOS

Якщо захист за допомогою програм-заставок здається малоефективним через умови, у яких доводиться працювати, має сенс використати більш серйозні засоби безпеки. Розглянемо захист комп'ютера за допомогою BIOS і методи фізичного блокування доступу.

Захист за допомогою BIOS

На кожному ПК є вбудована в нього система BIOS (Basic Input Output System – базова система введення-виведення), що являє собою декілька низькорівневих процедур, які тестують комп'ютер після включення його живлення і запускають операційну систему. Більшість BIOS підтримують можливість задання так званого пароля включення. Якщо пароль заданий, то комп'ютер виконає будь-яку операцію тільки після введення правильного пароля.

Для того, щоб задати пароль BIOS, необхідно при завантаженні комп'ютера, коли в правому нижньому кутку з'явиться напис *Press DEL to enter Setup*, натиснути клавішу *Delete* (або іншу зазначену), після чого додержуватися підказок, що з'являються при виділенні будь-якого пункту меню. Не будемо давати рекомендації з настроювань BIOS, оскільки їх випуском займаються багато виробників: IBM, AWARD, AMI й т.д.

Розглянемо механізм задання пароля BIOS для AWARD BIOS і AMI BIOS. При цьому необхідно мати на увазі, що при роботі з BIOS необхідно бути досить обережним, оскільки неправильне настроювання може негативно позначитися на функціонуванні комп'ютера або навіть привести до виходу з ладу деяких його вузлів.

Існує два типи паролів, які можна задавати в BIOS, – пароль на завантаження комп'ютера й на завантаження меню Setup BIOS.

Встановлення пароля AWARD BIOS

Для початку розглянемо, як задається пароль в BIOS виробництва Award. При завантаженні комп'ютера необхідно натиснути клавішу Delete, після чого можна вибрати один із двох варіантів задання пароля:

Set User Password (встановити пароль користувача). Він може задаватися або на вхід у меню Setup BIOS, або на вхід у Setup і завантаження комп'ютера. Для того, щоб вибрати будь-який з перерахованих вище методів блокування, необхідно в меню *Advanced BIOS Features* вибрати рядок *Password Check*. Потім після натискання клавіші Enter у вікні, що з'явилося, варто виділити або *Setup* (тобто пароль тільки на Setup BIOS), або *System* (тобто пароль на завантаження системи й Setup BIOS);

Set Supervisor Password (встановити пароль суперпвізора). Має пріоритет перед паролем користувача. Всі інші функції й настроювання такі самі, як в User Password.

Встановлення пароля AMI BIOS

Виклик вікна настроювань AMI BIOS здійснюється натисканням клавіші F2 у процесі тестування обладнання після включення комп'ютера. Тут для задання пароля необхідно вибрати вкладку *Security*, у якій доступні *Set Supervisor Password* і *Set User Password*. Характерною рисою AMI BIOS є те, що користувач, який знає Supervisor Password, може обмежувати

можливості користувача, що знає тільки User Password. Для того, щоб задати пароль, необхідно підсвітити бажаний тип пароля. Після натискання клавіші Enter відобразиться вікно із двома полями: у першому полі необхідно ввести пароль, а в другому – продублювати його.

Обхід пароля BIOS

Незважаючи на те, що BIOS є досить потужним засобом захисту, існують різні способи обходу встановленого пароля. Користувач може потрапити в досить скрутну ситуацію, якщо забуде встановлений пароль на завантаження системи. Це може призвести до необхідності покупки нової материнської плати. У подібних випадках можна скористатися "чорним ходом" BIOS, що також застосовується хакерами. Принцип його дії такий. Встановлений пароль зберігається в пам'яті CMOS (Complimentary Metal-Oxide-Semiconductor – комплементарний метало-оксидний напівпровідник), що, у свою чергу, повинна постійно підтримуватися батарейкою, встановленою на материнській платі. Отже, якщо батарейку витягти на якийсь час, можна домогтися очищення BIOS, тобто втрати встановлених параметрів і паролів.

Але справа в тому, що витягти батарейку не завжди можливо без використання допоміжного інструмента, тому має сенс вдатися до способу, описаного в деяких інструкціях до материнської плати. На більшості материнських плат установлені виводи для очищення пам'яті CMOS. Як правило, ці виводи перебувають поруч із батарейкою. Якщо користувач у силу різних причин не може їх знайти, варто звернутися до інструкції на материнську плату, що повинна входити в комплект документів, одержаних користувачем при покупці комп'ютера. Інструкцію до материнської плати також можна скачати з Internet, із сайту фірми-виробника.

Перед початком маніпуляцій по очищенню пам'яті CMOS варто відключити живлення комп'ютера, тобто всі дії при очищенні пам'яті CMOS потрібно робити, тільки коли комп'ютер виключений, внакше це спричинить вихід материнської плати з ладу. Після відключення живлення за допомогою документації на материнську плату треба знайти виводи для очищення пам'яті CMOS і замнути їх. Далі включаємо комп'ютер, заново виставляємо настройки BIOS, запам'ятовуємо їх і перезавантажуємось.

У випадку, коли документація відсутня, на виключеному комп'ютері варто спробувати по черзі замкнути всі виводи на платі, перевіривши потім, чи був знятий пароль.

Якщо всі перераховані вище маніпуляції не привели до потрібного результату, варто спробувати застосувати так званий інженерний пароль. Але при цьому необхідно пам'ятати про те, що він працює тільки на досить старих версіях BIOS, а також про те, що він може варіюватися від версії до версії. Нижче наведені паролі для BIOS виробництва фірм AWARD і AMI (табл.1.).

Таблиця 1 – Інженерні паролі для BIOS виробництва фірм AWARD і AMI

Інженерні паролі для AWARD BIOS			Інженерні паролі для AMI BIOS
_award	AWARD_SW	Condo	A.M.I.
01322222	AWARD PW	d8on	AAAMMMIII
589589	Awkward BIOSTAR	HLT	AMI
589721	CONCAT	J262	AMI?SW
595595	SYXZ	J332	AMI_SW
ALFAROME	TTPTHA	J64	BIOS
Ally	ZJAAADC	Lkwpeter	CONDO
ALLY	Pint	LKWPETER	LKWPETER
ally	PINT	SKY_FOX	PASSWORD
aPAf	SER		SER

Використання утиліти debug

На практиці часто буває, що користувач, установлюючи пароль на BIOS Setup, робить його занадто складним і згодом забуває. При цьому витягти батарейку він з якоїсь причини не може, наприклад, через високу ймовірність ушкодити гарантійні пломби. У цьому випадку можна скористатися стандартною утилітою Windows debug.exe, яку необхідно запускати з консольного режиму.

Для запуску програми debug.exe необхідно в меню *Пуск* вибрати пункт *Виконати*, після чого у вікні, що з'явилося, ввести ім'я програми *debug.exe*. Потім у вікні, що з'явилося, необхідно ввести наступну послідовність команд, виконуючи переведення рядка клавішею Enter:

```
-o 70 33
-o 71 33
-q
```

В результаті пароль на Setup буде знятий. Необхідно відзначити, що з виконанням цієї команди знімаються деякі настроювання Windows, тому після перезавантаження для подальшої коректної роботи системи може знадобитися інсталяційний диск Windows.

4.4 Програмні продукти для найпростішого захисту

У наш час, коли недосвідчений користувач або вірус, що потихеньку пробрався на комп'ютер, можуть одним махом знищити всю систему, необхідно обов'язково мати під рукою надійні програми для створення дискового іміджу і його розпаковування, для блокування комп’ютера та приховування інформації за певними критеріями. Більшість таких програм

є дорогими західними розробками, але існує багато програм для вільного розповсюдження або умовно-безкоштовних. Розглянемо деякі з них.

Stacker

Це одна з перших програм, що дозволяє захищати диски. Точніше, вона призначена для створення стиснутих дисків, що підтримують пакування і розпакування інформації "на льоту", а одна з опцій дозволяла захищати збережену на диску інформацію паролем.

Але у програмі використана не зовсім надійна хеш-функція, внаслідок чого складність підбору пароля для розшифрування диска дорівнює 2^8, тобто пароль підбирається досить швидко.

Diskreet

Ця програма входить до складу Norton Utilites і призначена для створення зашифрованих файлів і дисків. Програма підтримує два методи шифрування, один з яких базується на алгоритмі DES і працює повільно, а другий – саморобний і більш швидкий алгоритм, хоча і є примітивним і нестійким до атак на основі відкритого текста.

BootLock

Програма дозволяє шифрувати завантажувальний диск, а отже, робить всю обчислювальну систему недоступною для зловмисника. при цьому програма шифрує не всі дані диска, а лише системні області: завантажувальний запис (Boot) і кореневу директорію (Root). А таблиці розміщення файлів (Fat-таблиця) і область даних, в якій знаходяться самі файли, виявляються незашифрованими.

Шифрування виконується шляхом накладання 512-байтової гами на кожний сектор, що підлягає шифруванню, причому для всіх секторів одного диску використовується одна й та сама гама. Через це наявність одного відкритого сектора дає можливість розшифрувати і інші.

Screen Look

Це програма від компанії iJEN Software, яка добре зарекомендувала себе, відноситься до програм-заставок.. Вона підтримує можливості "запирання" комп'ютера під час короткотермінової відсутності користувача. Крім того, вона може бути використана в якості "вхідного контролю" в операційну систему.

Black Magic

Програма Black Magic безкоштовна. Особливістю цієї програми є те, що вона не є програмою-заставкою у чистому вигляді, оскільки під час блокування не використовує екранну заставку. Слід зазначити, що подібні програми здатні забезпечити безпеку лише на досить короткий термін при відсутності користувача, і тому покладати на них великі надії не можна. Але для користувачів Windows 9X дані програми є непоганою альтернати-

вою “рідним” утилітам аналогічного призначення. З іншої сторони, для більш серйозного блокування системи застосовують захист за допомогою BIOS і деякі інші прийоми.

Backup 2001

Програма для резервного копіювання і відновлення. Для цього вибрано ZIP-формат як найпоширеніший. Резервне копіювання здійснюється на дискети, жорсткі диски, по локальній мережі, ZIP-диски, JAZZ-диски, Sparq, CD-R і ін. Працює під Windows 95(OSR2) /98/NT4/2000.

Desktop Disguise

Програма дозволяє приховати робочий стіл разом зі всіма вікнами методом перекриття заданим зображенням.

PC Security

Утиліта, призначена для приватного доступу до даних шляхом блокування файлів, каталогів, системи

Bluescreen Screen Saver

Цікавий зберігач екрану, який емулює на екрані комп'ютера системну помилку або, іншими словами, видає синій екран. За синім екраном слідує спроба перезавантаження комп'ютера з подальшою "системною помилкою". І так по кругу. Виглядає це досить реалістично, і необізнаній людині може дійсно здатися, що наш комп'ютер "висить".

HDD Password Protection

Якщо ми не хочемо, щоб у нашу відсутність хтось працював на нашому комп'ютері, слід обмежити доступ до нього. Можна, наприклад, поставити пароль на вхід у Windows – це найпростіший, але і найефективніший спосіб захисту. Ще надійніше обмежити доступ ще в BIOS, але, як показує практика, і цей варіант теж небездоганний і достатньо ризикований.

Програма HDD Password пропонує нам золоту середину. Код доступу при її використанні потрібно вводити при старті операційної системи, а не перед входом в її графічну оболонку. Правда, в цьому випадку все одно можна дістати доступ до даних на жорсткому диску шляхом завантаження з дискети або CD-ROM. Вищого рівня захисту розробники обіцяють добитися в комерційній версії HDD Password Professional. Можливо, в цьому варіанті з'явиться у програми і нормальний графічний інтерфейс. Поки ж запускається вона тільки в DOS або в досівському вікні з-під Windows. Втім, ніяких складнощів в її управлінні немає. Переміщаючись за допомогою клавіатури по рядках меню, можна задати пароль, зняти його, повернути колишній варіант завантажувального MBR-сектора, який, до речі, автоматично зберігається.

Важливо згадати, що HDD Password при установці стирає дані boot-менеджерів і дозволяє завантажувати тільки Windows.

WinGuard Pro

Програма WinGuard Pro допоможе нам забезпечити свій комп'ютер від несанкціонованого вторгнення або ж від дій невмілого користувача. Програма може заблокувати за допомогою пароля запуск певних файлів, меню Пуск, Робочий стіл і т.д. За допомогою її можна також заборонити установку і видалення програм, закрити доступ в Інтернет. По суті, можна заблокувати практично будь-який елемент операційної системи.

WinGuard Pro розповсюджується небезкоштовно, але при першому запуску у нас буде можливість вибрати тип роботи: використовувати програму безкоштовно, але з істотними обмеженнями функціональності, або ж отримати доступ до всіх функцій утиліти на термін 30 днів.

Acronis True Image 6.0

Acronis True Image – російськомовна і недорога програма, здатна створювати образи дисків прямо з операційної системи Windows.

Що стосується ціни, то ця програма коштує на декілька порядків менше, ніж її західні аналоги (всього близько $10), а по можливостях не поступається багатьом з них. Перше, що впадає в очі – це простий і продуманий інтерфейс, виконаний у вигляді "майстра". Все, що потрібно користувачу – це переходити від вкладки до вкладки, вибираючи по черзі потрібні параметри. Сьогодні це як ніколи актуально, оскільки багато користувачів починають своє комп'ютерне "життя" відразу з Windows XP, і спочатку їм складно розібратися в досить розгалуженому інтерфейсі, наприклад, такої програми, як DriveImage.

Основні характеристики програми такі. Перше – це ступінь компресії, з якою вона стискає дані. Так, логічний диск, заповнений на 3.6 Гбайти, програма стиснула в архів розміром 1,5 Гбайт, що дуже навіть непогане. При цьому можна створити імідж не на жорсткому диску, а на знімних носіях таких як ZIP-накопичувач або компакт-диск. При цьому не варто турбуватися, що розмір іміджу перевищить об'єм носія, оскільки програма уміє розбивати архіви на декілька частин.

У True Image передбачена маса корисних дрібниць, які роблять роботу з нею зручнішою. Так, наприклад, є можливість додати довгий коментар до створюваного образу.

Найголовніше зручність програми полягає у тому, що вона може створювати і відновлювати образи розділів без перезавантаження в DOS. Це дуже істотний плюс, оскільки швидкість роботи при цьому збільшується майже в два рази. Такою можливістю не володіє жодна аналогічна програма, навіть такі як PowerQuest Drive Image і Norton Ghost.

MAPBackup 1.2

Дуже зручна програма для створення резервних копій. Не дивлячись на те, що всі подібні утиліти схожі одна на одну, MAPBackup все ж таки різниться в цьому. А відрізняє програму від аналогів цікавий підхід до ар-

хівації даних. MAPBackup може створити копію робочих файлів певної програми відразу після того, як користувач її закрив. Це дуже корисно, наприклад, для створення резервної копії пошти. При настроюванні програми потрібно вказати, який процес повинен відстежуватися (наприклад, thebat.exe), який архіватор повинен бути використаний і які директорії повинні бути збережені. Після того, як процес буде завершений користувачем, MAPBackup створить архівну копію вказаних файлів або директорій.

SmartBackup

Збереження важливої інформації і файлів останнім часом стало задоволене актуальною темою. Втрата документів, настройок і лігв інтернет-пейджерів, поштових повідомлень може стати причиною серйозних неприємностей. А втратити все це багатство дуже легко. Системний збій, вірус або не уміючий працювати з комп'ютером людина можуть в два рахунки поховати результати багатомісячної праці. Врятувати від цього кошмару можуть утиліти для резервного копіювання, і SmartBackup - одна з них. Працює програма дуже просто. У призначений час вона копіює вказані файли або теки в окрему директорію. Копіювати можна не всі файли директорії, а тільки ті, які піддалися модифікації. Плюс до всього, SmartBackup володіє такою корисною можливістю, як стиснення резервної копії в архів.

4.5 Відновлення інформації після сбоїв

Важлива інформація може бути втрачена як через необмірковані дії користувачів, так і внаслідок збоїв у роботі техніки або внаслідок навмисного пошкодження даних зловмисником. Але існують програмні засоби, які дозволяють відновити інформацію після таких збоїв. Розглянемо деякі з цих програм.

Програма EasyRecovery Pro

Популярна програма для відновлення видалених або пошкоджених файлів. Усі налаштування програми EasyRecovery Pro досить прості і доступні користувачу, хоча роботу може утруднити повністю англомовний інтерфейс.

У своєму складі програма містить утиліти, за допомогою яких можна виконати діагностику жорстких дисків на предмет наявності помилок:

- DriveTests – сканування приводів усіх накопичувачі на предмет потенційних не поладок;
- SMARTTests – моніторинг жорсткого диска для виявлення можливих пошкоджень;
- SizeManeger – надання інформації про використання дискового простору;

- JumperViewer – перевірка правильного підключення пристроїв;
- PartitionTests – аналіз існуючої структури файлової системи;
- DataAdvisor – створення завантажувальної дискети для діагностики системи.

Основу даної програми складають утиліти для безпосереднього відновлення даних:

- AdvancedRecovery – відновлення даних з використанням специфічних налаштувань;
- DeletedRecovery – знаходження і відновлення пошкоджених файлів;
- FormatRecovery – відновлення інформації з відформотованого або видаленого тома;
- RawRecovery – відновлення інформації з директорії;
- ResumeRecovery – перегляд LOG-файлів, створених під час роботи по відновленню даних;
- EmergencyDiskette – створення аварійної завантажувальної дискети.

Утиліти для відновлення конкретних файлів:

- AccessRepair – відновлення бази даних Microsoft Access;
- ExcelRepair – відновлення таблиці Microsoft Excel;
- PowerPointRepair – відновлення презентації Microsoft PowerPoint;
- WordRepair – відновлення документу Microsoft Word;
- ZipRepair – відновлення пошкодженого WinZip-архіва;
- EmailRepair – засоби для відновлення e-mail листів Outlook Repair;
- Software Updates – засоби для оновлення програми;
- Crisis Center – тематичні каталоги з посиланнями, що дозволяють допомогти користувачу у кризисних ситуаціях.

Програма FinalData

Дана програма дозволяє відновлювати інформацію, втрачену в результаті дії вірусів, форматування дисків або випадкового знищення файла.

Діалог програми дозволяє вибрати диск, на якому знищено аюл пошкоджено інформацію і діапазон кластерів, в якому здійснювати пошук. В результаті сканування програма показує весь вміст диска на даний момент (як існуючі. так і видалені файли). Крім того. можна окремо вивести розділ видалених файлів і розділ втрачених файлів. Потім над файлом здійснюються певні операції по його відновленню.

Програма має зручний інтерфейс і розгалужену систему меню, зручну і зрозумілу у роботі.

Програма GetDataBack

Це також популярна програма для відновлення пошкоджених та видалених файлів. Дії в ній виконуються за допомогою покрокового Майстра, який на першому кроці пропонує ряд налаштувань для встановлення параметрів:

- вибір каталогу, в який будуть копіюватися відновлені файли;

- вибір каталогу для тимчасових файлів;
- вибір шляху до програми обробки диска DiskExplorer, яка встановлюється у разі необхідності;
- можливість відновлення знищених файлів (навіть не прив'язаних до якогось певного каталогу;
- можливість використовувати або не використовувати Fat-таблицю (якщо знайдена на момент відновлення Fat не відповідає файловій системі).

На другому кроці здійснюється сканування вказаного диску, враховуючи вибрані на першому кроці параметри. При цьому за бажанням користувача пошук і сканування може здійснюватися як по всьому вибраному диску, так і на певній його частині (вказується початковий і кінцевий сектори), в одній чи іншій файловій системі (якщо ця опція не вказана, буде здійснюватись більш детальний пошук, що збільшить час сканування).

Далі, керуючись пропозиціями Майстра, можна відновлювати певні файли і каталоги. Слід зауважити, що не слід копіювати відновлені дані на той диск, з якого виконується відновлення.

4.6 Робота з реєстром Windows

Більшість комп'ютерних вірусів, що їх застосовують зловмисники для отримання контролю над ПК, використовують реєстр операційної системи для перехоплення деяких системних функцій або для автоматичного завантаження тіла віруса при включенні комп'ютера. Крім того, зловмисник може вилучити деякі (або всі) файли реєстра, що приведе до повної непрацездатності комп'ютера. Зловмисник може скопіювати їх на свій комп'ютер, що дасть йому можливість підібрати паролі до облікових записів користувачів. Ось чому вміле використання реєстра дозволить адміністратору комп'ютера не лише захистити ПК від несанкціонованих дій користувачів, але суттєво спростить настроювання робочої станції. Отже, вивчення реєстра, правильне його використання, а також тонке налаштування параметрів операційної системи з його застосуванням дозволять звести до мінімуму імовірність зламу комп'ютера.

4.6.1 Призначення і структура реєстра

Практично всі версії Windows (окрім Windows 3.1) мають схожі реєстри, але є і суттєві відміни, основною з яких є організація кореневих каталогів. Windows 98 зберігає реєстр у двох файлах – system.dat та user.dat, що знаходяться у каталогах, в яких встановлено операційну систему.

Реєстр Windows NT – це основа операційної системи, величезна база даних настройок, що зберігаються в папках ...\System32\Config і папках

призначених для користувача профілів (Ntuser.dat). Без реєстру Windows XP була б просто набором програм, нездатних виконати навіть прості функції ОС. Все, навіть найдрібніші деталі конфігураційних даних XP, упаковано в реєстр.

Реєстр або системний реєстр – це база даних для зберігання відомостей про конфігурацію комп'ютера і настройок операційної системи. Реєстр містить дані, до яких Windows XP постійно звертається під час завантаження, роботи і її завершення, а саме:

- профілі всіх користувачів, тобто їх настройки;
- конфігурація устаткування, встановленого в операційній системі;
- дані про встановлені програми і типи документів, створюваних кожною програмою;
- властивості тек і значків програм;
- дані про використовувані порти.

Реєстр має ієрархічну деревовидну (ієрархічну) структуру, що складається з розділів, підрозділів і ключів (параметрів), і якщо ми бажаємо захистити нашу ОС якшайкраще, то знати структуру реєстра просто необхідно. Для роботи з реєстром використовується утиліта Regedit.

Розділи і підрозділи – це теки в лівому вікні regedit (рис.20). *Ключ реєстру* або параметр – це змінна, якій привласнене певне значення, це те, що ми бачимо в правому вікні regedit.

Кущ (основний розділ, стандартний розділ, в англійській документації – вулик, від англ. "hive") – це розділ реєстру, що відображається як файл на жорсткому диску. Кущ є набором розділів, підрозділів та параметрів і має корінь на верхньому рівні ієрархії реєстру. За замовчуванням більшість файлів цих кущів (Default, SAM, Security і System) зберігається у папці ...\System32\Config. Тека %SystemRoot%\Profiles містить профілі (настройки) для кожного користувача комп'ютера. Оскільки кущ є файлом, його можна переміщати з однієї системи в іншу. Для редагування цього файлу необхідно використовувати редактор реєстру.

Слід зауважити, що Windows XP, на відміну від своїх попередниць, не має обмеження за розміром реєстру.

Реєстр містить п'ять основних секцій, які називаються *кореневими розділами* (гілками, вуликами) і є аналогами кореневих розділів жорсткого диска. Кожен розділ має власне місце зберігання і файл журналу, при необхідності будь-який кореневий розділ можна відновити, не зачіпаючи решту розділів реєстру (табл.2).

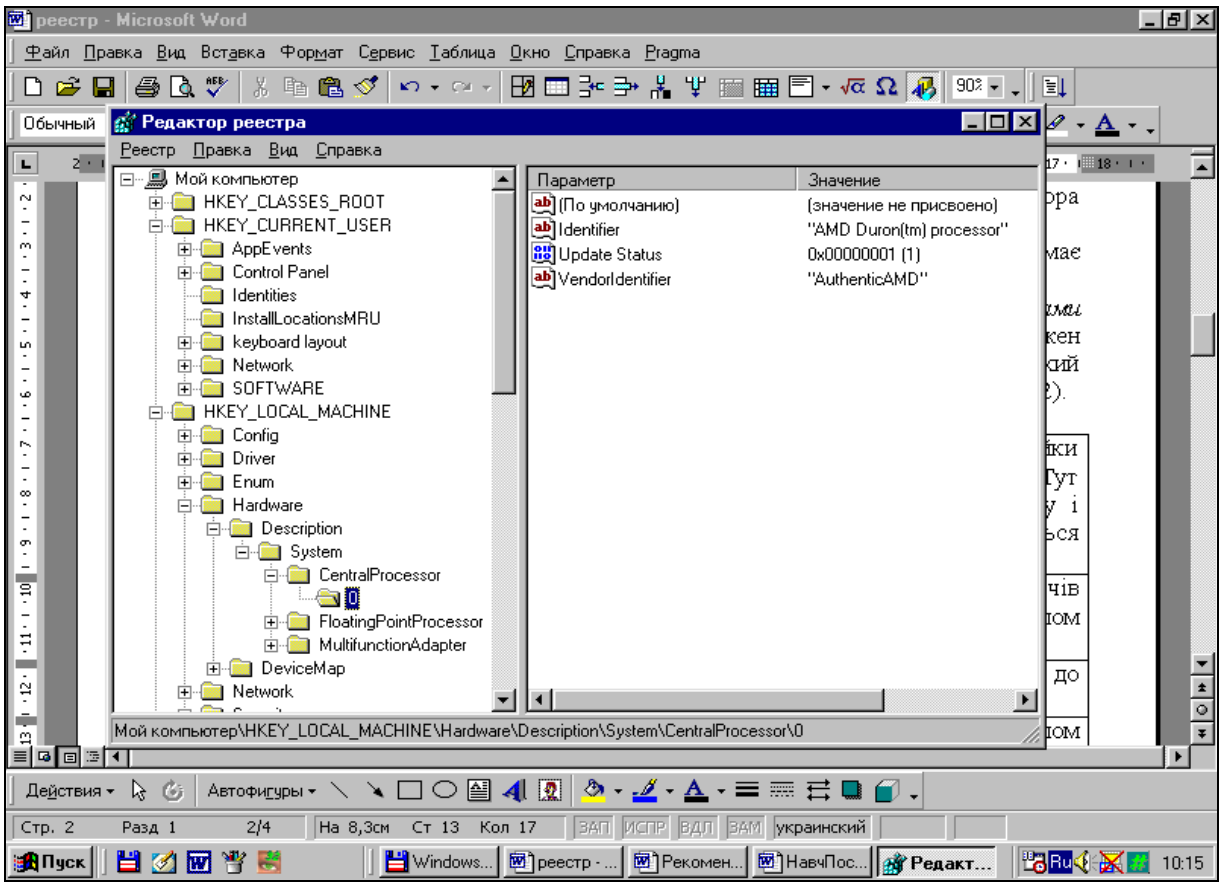


Рисунок 20 – Ієрархічна структура реєстра

Таблиця 2 – Розділи реєстру

HKEY_CURRENT_USER	Даний розділ є кореневим для даних настройки користувача, що увійшов до системи зараз. Тут зберігаються теки користувача, кольору екрану і настройки панелі управління. Ці дані називаються профілем користувача.
HKEY_USERS	Даний розділ містить всі профілі користувачів комп'ютера. HKEY_CURRENT_USER є підрозділом HKEY_USERS.
HKEY_LOCAL_MACHINE	Розділ містить дані настройки, що відносяться до даного комп'ютера (для всіх користувачів).
HKEY_CLASSES_ROOT	Даний розділ є підрозділом HKEY_LOCAL_MACHINE\Software. Відомості, що зберігаються тут, забезпечують відкриття необхідної програми при відкритті файлу за допомогою провідника.
HKEY_CURRENT_CONFIG	Даний розділ містить відомості про профіль устаткування, використовуваному локальним комп'ютером при запуску системи

Усередині кореневого розділу знаходяться розділи і підрозділи, які аналогічні каталогам і підкаталогам жорсткого диска. Розділ може містити

інформацію або дані. Розділ і підрозділ можуть не мати параметрів, або мати їх один чи декілька, параметр за замовчуванням, а також не мати або мати декілька підрозділів.

Кожен параметр має ім'я, тип і значення. Три частини параметра реєстру, або атрибути параметра (ім'я, тип даних, значення) завжди розташовуються у певному порядку:

[RegistrySizeLimit] [REG_DWORD] [0x8000000].

Тип даних може приймати декілька певних значень, які наведені у таблиці 3.

Таблиця 3 – Типи даних, що використовуються в реєстрі

REG_BINARY	Необроблені двійкові дані. Більшість відомостей про апаратні компоненти зберігається у вигляді двійкових даних і виводиться в редакторі реєстру в шістнадцятковому форматі.
REG_DWORD	Дані, представлені цілим числом (4 байти). Багато параметрів служб і драйверів пристроїв мають цей тип і відображаються в двійковому, шістнадцятковому або десятковому форматах.
REG_EXPAND_SZ	Розширюваний рядок даних. Цей рядок є текстом, що містить змінну, яка може бути замінена при виклику з боку додатку.
REG_MULTI_SZ	Багаторядкове поле. Значення, які фактично є списками текстових рядків у форматі, зручному для сприйняття людиною, звичайно мають саме цей тип даних. рядки розділені символом NULL.
REG_SZ	Текстовий рядок у форматі, зручному для сприйняття людиною. Значенням, що є описами компонентів, звичайно привласнюється саме цей тип даних.
REG_FULL_ RESOURCE_ DESCRIPTOR	Послідовність вкладених масивів, розроблена для зберігання списку ресурсів апаратного компоненту або драйвера.

4.6.2 Зберігання реєстру

Елементи реєстру зберігаються у вигляді атомарної структури. Реєстр розділяється на складові частини – кущі або вулики. Кущі зберігаються на диску у вигляді файлів. Деякі з них, такі, як HKLM\HARDWARE, не зберігаються у файлах, а створюються при кожному завантаженні, тобто є змінними (volatile). При запуску системи реєстр збирається з кущів в єдину деревовидну структуру з кореневими розділами. Наведемо кущі реєстру і їх місцеположення на диску (для NT версії 4.0 і старших) (табл.4).

У наведеній таблиці використовуються розширення:

- LOG – журнал транзакцій, в якому реєструються всі зміни реєстру.
- SAV – копії кущів в тому вигляді, в якому вони були після завершення текстової фази установки.

Таблиця 4 – Основні кущі реєстру та файли, що їх містять

Кущ (гілка реєстру)	Розташування	Імена файлів, що містять відповідні кущі
HKEY_LOCAL_MACHINE\ SYSTEM	…\system32\config\system	System, System.alt, System.log, System.sav
HKEY_CURRENT_CONFIG	…\system32\config	System, System.alt, System.log, System.sav
HKEY_LOCAL_MACHINE\ SAM	…\system32\config\SAM	Sam,Sam.log, Sam.sav
HKEY_LOCAL_MACHINE\ SECURITY	…\system32\config\SECURITY	Security, Security.log, Security.sav
HKEY_LOCAL_MACHINE\ SOFTWARE	…\system32\config\software	Software, Software.log, Software.sav
HKEY_LOCAL_MACHINE\ HARDWARE	Змінний кущ	
HKEY_LOCAL_MACHINE\ SYSTEM\Clone	Змінний кущ	
HKEY_USERS\ <SID_користувача>	…\Documents and Settings\ <Username>	ntuser.dat, ntuser.dat.log
HKEY_USERS\ <SID_користувача>_Classes	…\LocalSettings\Application Data\Microsoft\Windows\	UsrClass.dat
HKEY_USERS\.DEFAULT	… (системна директорія)	Default, Default.log, Default.sav
Файли, не асоційовані з розділами		Userdiff, Userdiff.bg

4.6.3 Відновлення реєстру

Реєстр є справжньою базою даних, тому в ньому використовується технологія відновлення, схожа з NTFS. Вже згадані LOG-файли містять журнал транзакцій, який зберігає всі зміни. Завдяки цьому реалізується атомарність реєстру – тобто в даний момент часу в реєстрі можуть бути або старі значення, або нові, навіть після збою. Як бачимо, на відміну від NTFS, тут забезпечується збереження не тільки структури реєстру, але і

даних. До того ж, реєстр підтримує такі параметри NTFS, як управління вибірковим доступом і аудит подій – система безпека пронизує всю NT .

Узгоджуючись з рекомендаціями Microsoft, сторонні розробники програмного забезпечення повинні зберігати інформацію про настройки своїх програм у реєстрі. Таким чином, у ньому зберігається інформація про апаратну конфігурацію комп'ютера, різні настройки операційної системи і настройки встановлених програм. ОС Windows XP і додатки поміщають інформацію в реєстр ОС, що добре і погано одночасно. Добре – тому що реєстр є загальною пам'яттю для ефективного зберігання інформації. Погано – тому що розміри реєстру збільшуються у міру того, як додатки і система зберігають в системі всю нову інформацію. У міру цього процесу в реєстрі з'являється величезна кількість сміття, яке необхідно час від часу видаляти.

Ніколи не слід видаляти або міняйти інформацію в реєстрі, якщо немає впевненості, що робиться саме те, що потрібно. Невірні дії при зміні реєстру можуть серйозно пошкодити систему. Перед зміною реєстру слід створити резервну копію всіх важливих даних, що є на комп'ютері. Вся відповідальність за порушення працездатності комп'ютера і операційної системи лежить на користувачах.

Відновлення реєстру Windows 98

У цій операційній системі існує така технологія захисту системного реєстра від пошкодження. У підкаталозі SYSBCKUP, що знаходиться у системному каталозі, зберігаються архіви (файли з розширенням .CAB, за замовчуванням їх п’ять), в які система після успішного завантаження комп’ютера, копіює весь реєстр. Ці архіви можна використовувати для відновлення пошкодженого реєстра. Для цього слід завантажити комп’ютер у режимі MS-DOS і виконати команду

SCANREG /RESTORE.

В результаті буде відображено перелік доступних копій, з якого можна вибрати необхідну, після чого реєстр буде повністю відновлено.

У випадку. коли необхідно створити копію реєстра (наприклад, після встановлення і настроюваннння якого-небудь обладнання), можна скористатися командою

SCANREG /BACKUP,

яка у випадно нормальної перевірки створить резервну копію.

Відновлення реєстру Windows 2000/XP

Відновлення реєстра в цих ОС є складною задачею, оскільки реєстр зберігається у багатьох файлах і для їх відновлення необхідно отримати доступ до каталогів, в яких вони знаходяться. А також треба мати повний комплект файлів реєстра, оскільки всі вони взаємопов’язані і синхронізуються в прроцесі роботи.

Передбачено функцію завантаження комп'ютера з використанням останньої вдалої конфігурації. для цього необхідно:

- включити комп'ютер:
- в момент початку завантаження ОС натиснути клавішк [F8];
- у списку варіантів завантаження. який з'явиться на екрані, вибрати пункт *Запуск последней удачной конфыгурации* і потім натиснути клавішу [ENTER].

Слід зауважити, що цей спосіб відновлення підходить лише тоді, коли на комп'ютері було встановлено нове обладнання і воно викликає якийсь конфлікт з наявними пристроями. Насправді ж тут відновлюється лише та інформація, яка зберігається у розділі HKEY_LOCAL_MACHINE\ System\CurrentComtrolSet, а решта розділів залишаються незмінними.

Разом з тим існує більш складний метод відновлення реєстра, який полягає в архівуванні каталога …\System32\Config, а також файлів Ntuser. dat та Ntuser.dat.log. Але ж просте копіюваннння даних файлів неможливе, оскільки вони постійно використовуються системою і доступ користувачів до них заборонено. Тому необхідноо використовувати спеціальну програму *Архивация данных,* яка входить в комплект службових утиліт.

Для її запуска необхідно виконати таку послідовність команд: *Пуск → Все программы →Стандартные → Служебные → Архивация данных.* Далі вибрати в меню *Сервис* команду *Создание диска аварийного восстановления* або натиснути на кнопку *Диск аварийного восстановления*. У діалогову вікні, що з'явиться, встановити прапорець *Архивировать реестр в папку восстановления* і натиснути [OK].

Після виконання операції архівування в каталог …\Repair будуть поміщені всі файли, що містять кущі реєстра, і, крім того, ці ж файли будуть скопійовані на дискету. Потім треба скопіювати вказані файли на інший носій інформаці і використовуватии їх у разі потреби.

4.6.4 Утиліти для роботи з реєстром

Безпосереднє редагування реєстра за допомогою програм regedit.exe або regedat32.exe супроводжується деякими складностями, оскільки користувач повмнен точно знати, за що відповідає той чи інший ключ і яке значення він повинен приймати. У випадку некоректної установки ключа або його додавання\видалення може виникнути збій системи чи навіть повна втрата її працездатності.

Для запобігання подібним ситуаціям рекомендується використовувати спеціалізовані утиліти, які мають такі переваги:

- зручний графічний інтерфейс;
- функцію контролю значень, яка не дозволить встановити помилкове значення;

- функцію повернення (рос. “отката”), що автоматично створює резервні копії зміненої частини реєстра;
- моодуль відслідковування змін, який відображає всі зміни, виконані програмою, та ін.

Всі утиліти для роботи з реєстром Windows можна розділити на такі категорії:

- редактори (Reg Organizer, Reg Cleaner, NBG Clean Registry, Reg View);
- діагностика і лікування;
- оптимізатори;
- монітори реєстрів (RegMon – Registry Monitor, NGB Registry Monitor);
- багатофункціональні.

Контрольні питання

1. Перелічіть загальні методи блокування доступу до комп’ютера.
2. Що таке екранна заставка, як її поставити або змінити на комп’ютері?
3. Як блокувати роботу за допомогою екранної заставки?
4. Які методи використовують хакери для обходу блокування за допомогою екранної заставки? Як цьому запобігти?
5. Які інші програми ви знаєте для блокування роботи комп’ютера? Коротко охарактеризуйте їх. Наведіть переваги та недоліки їх застосування.
6. В чому полягає блокування роботи комп’ютера за допомогою BIOS?
7. Як встановити пароль BIOS на різні версії?
8. Які методи використовують хакери для обходу парольного захисту BIOS?
9. Що можна зробити у разі втрати паролю BIOS?
10. Що таке CMOS? Яке функціональне призначення цього об’єкту комптерної системи?
11. Як можна стерти пам’ять CMOS?
12. Що таке інженерний пароль і для чого його використовують?
13. Для чого існує утиліта debug?
14. Що таке реєстр і для чого він прризначений?
15. Наведіть структуру реєстра і охарактеризуйте його складові.
16. Де розташовані складові реєстру?
17. Як можна відновити реєстр після помилок у системі?
18. Наведіть відомі програмні продукти для роботи з реєстром .

5 РЕКОМЕНДАЦІЇ ЩОДО ПІДВИЩЕННЯ МІР БЕЗПЕКИ ОС WINDOWS XP

Як вже говорилося вище, однією з найбільших проблем, що підстерігають користувачів комп'ютерів, завжди були комп'ютерні віруси, але останніми роками до них додалися і шпигунські програми. Для домашнього комп'ютера, на якому немає конфіденційної інформації, це не дуже велика проблема, але, все одно, не дуже приємно, коли інформація про власника комп'ютера, кудись передається без його відома. А от для державних установ та приватних підприємств ця проблема є дуже суттєвою.

Все ж таки можна, навіть не використовуючи спеціалізованого програмного та технічного забезпеченння, дещо підвищити рівень безпеки комп'ютера, щоб менше турбуватися про важливі дані і не дозволили нічому відправляти інформацію з нашого комп'ютера без нашого дозволу.

Підсумовуючи весь наведений вище матеріал, можна порадити прийняти деякі додаткові міри із захисту нашої інформації, які є в арсеналі кожного користувача.

5.1 Додаткові обмеження на паролі

Запитувати пароль при поверненні до роботи з режиму очікування

```
[HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\
Windows\System\Power]
"PromptPasswordOnResume"='1'
```

Вимагати паролі тільки з букв і цифр

Цей ключ примусить завжди комбінувати в паролях букви і цифри.

```
[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\
CurrentVersion\Policies\Network]
"AlphanumPwds"='1'
```

Установка мінімальної кількості символів в паролях

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\
CurrentVersion\Policies\Network]
"MinPwdLen"=hex:6
```

Відміна зберігання паролів в Internet Explorer

Якщо довіряти компанії MicroSoftwer щодо зберігання паролів і іншої конфіденційної інформації, то можна дозволити Windows зберігати пароль доступу в Інтернет на диску свого комп'ютера, але це не дуже слушна ідея.

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\
```

CurrentVersion\Internet Settings]
"DisablePasswordCaching"='1'

Заборона зберігання паролів

В ранішніх версіях Windows 9x збереження паролів було великою проблемою. Тепер це не так, Windows 2000\XP захищають цю інформацію значно краще. Але, знову ж таки, нам вирішувати, дозволити нашій ОС зберігати паролі на диску чи ні. Це стосується паролів користувачів і мережевих паролів.

[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\
CurrentVersion\Policies\Network]
"DisablePwdCaching"='1'

5.2 Додаткові міри безпеки в мережі

Заборона доступу для анонімних користувачів

Анонімний користувач може отримати доступ до списку користувачів і відкритих ресурсів. Можна це заборонитискористатися цим ключем.

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\LSA]
"RestrictAnonymous"='1'

При установці даного параметра жоден хакер не зможе віддалено підключитися до комп'ютера користувача, не пройшовши перевірку. Тільки в тому випадку, якщо користувач авторизується на комп'ютері, до якого підключається, він зможе мати доступ до відкритих ресурсів.

Не показувати паролі при введенні

При спробі доступу до захищеного паролем ресурсу, Windows не приховує пароль, який ми вводимо. Цей ключ дозволяє замінювати символи пароля зірочками.

[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\
CurrentVersion\Policies\Network]
"HideSharePwds"='1'

Приховати комп'ютер від інших користувачів в мережі

Щоб уникнути підключення до мережі стороннього користувача, бажано відключити *Network Browser,* що зробить комп'ютер «невидимим» ззовні і, отже, виключить можливість сканування на наявність відкритих портів і інших вразливих місць. Слід зауважити, що, знаючи мережеве ім'я комп'ютера, все одно можна провести сканування, і в цьому випадку відключення даної опції не дасть потрібного результату.

Отже, існує ключ дозволяє включити режим, при якому в режимі огляду мережі інші користувачі не бачитимуть нашого комп'ютера.

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\

Services\LanmanServer\Parameters]
"Hidden"='1'

Існує можливість виконання цієї операції з командної консолі операційної системи, для чого в консолі слід ввести команду

net config server /hidden: yes.

Крім того, необхідно відключити *Null-Session*, яка дозволяє іншому користувачу, навіть не знаючи логінів і паролів, отримати всю інформацію про share-директорії (тобто доступних для загального користування), про наявних локільних користувачів і т.д. Для цього в разділі HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa необхідно створити параметр RestrictAnonymous типу REG_DWORD и присвоїти йому значення 1.

Всім користувачам мережі Internet рекомендується вести повний запис всіх подій, що відбуваються з модемом. Для цього в директорії HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Rasman\Parameters треба створити параметр Logging типу REG_DWORD і присвоїти йому значення 1.

В результаті комп'ютер записуватиме всі події і зберігатиме їх у файлі Device.LOG, що знаходиться у системному каталозі ...\System32\RAS\.

Заборона на відкриття доступу до ресурсів

Дуже часто недосвідчені користувачі відкривають мережевий доступ до каталогів для обміну інформацією з іншими користувачами, а потім забувають закривати його. Це може привести до того, що буде втрачена важлива інформація. На практиці дуже часто зустрічаються ситуації, коли в мережі підприємства можна знайти комп'ютери, на яких повністю відкриті всі жорсткі диски. Адже це може бути комп'ютер секретаря директора, на якому зберігається безліч важливих документів.

Для того, щоб повністю заборонити користувачу відкривати ресурси, необхідно додати параметр NoFileSharingControl:

[HKEY_CURRENT_USER\5oftware\Microsoft\Windows\
CurrentVersion\Policies\Network]
“NoFileSharingControl”=’1’ (тип REG_DWORD).

Безпека NetBT

Якщо комп'ютер має постійну IP-адресу, то він може бути схильний до DoS-атаки (Denial of Service – відмова в обслуговуванні). Причина цьому – *NetBIOS*, що працює «поверх» протоколу TCP/IP (NetBT). На жаль, реалізація протоколу NetBIOS не включає перевірки автентичності. Для усунення цієї помилки потрібен новий файл – netbt.sys, отримати який можна за адресою: support.microsoft.com/support/kb/articles/q269/2/39.asp.

Імена і паролі в Internet Explorer

Функція автозаповнення імен користувачів і паролів у формах, які розташовуються на Internet-сторінках, прискорює роботу в Мережі, а також не вимагає від користувача запам'ятовування тієї інформації, яку він

вводив, відвідуючи сайт раніше. Проте, якщо зловмисник отримає доступ до комп'ютера, на якому дана функція включена, він зможе без труднощів скористатися різними сервісами Internet, що належать користувачу, наприклад, його електронною поштою. Щоб уникнути цієї ситуації, необхідно в розділі HKEY_CURRENT\ USER\Software\Microsoft\InternetExplorer\Main присвоїти параметру FormSuggestPasswords (тип STRING) значення "*no*". Значення *"yes"* дозволяє включити автозаповнення імен користувачів і паролів у формах

5.3 Приховування слідів роботи за комп'ютером

Очистка файла підкачки

Для підвищення безпеки слід включити опцію очищення файлу підкачки (swap-file в Windows 9X або page-file в Windows 2000, файл pagefile.sys знаходиться в кореневому каталозі кожного або тільки системного диска) після завершення роботи. Очищати цей файл необхідно, оскільки саме в ньому система в процесі роботи зберігає всілякі дані, у тому числі і паролі. Все ж таки варто помітити, що при включенні цієї опції дані не видаляються в загальному значенні цього слова.

Файл підкачки є сторінковим файлом, тобто він розбитий на сторінки. При виключенні живлення всі неактивні сторінки заповнюються нулями, а інформація активних сторінок все одно залишається в *swap*, хоча серйозними наслідками це не загрожує, оскільки всі дані, що стосуються безпеки, дана опція дозволяє затерти.

Для того, щоб очистити swap-файл, потрібно відкорегувати певні параметр у реєстрі (1 – очищувати, 0 – в іншому випадку) (рис.21).

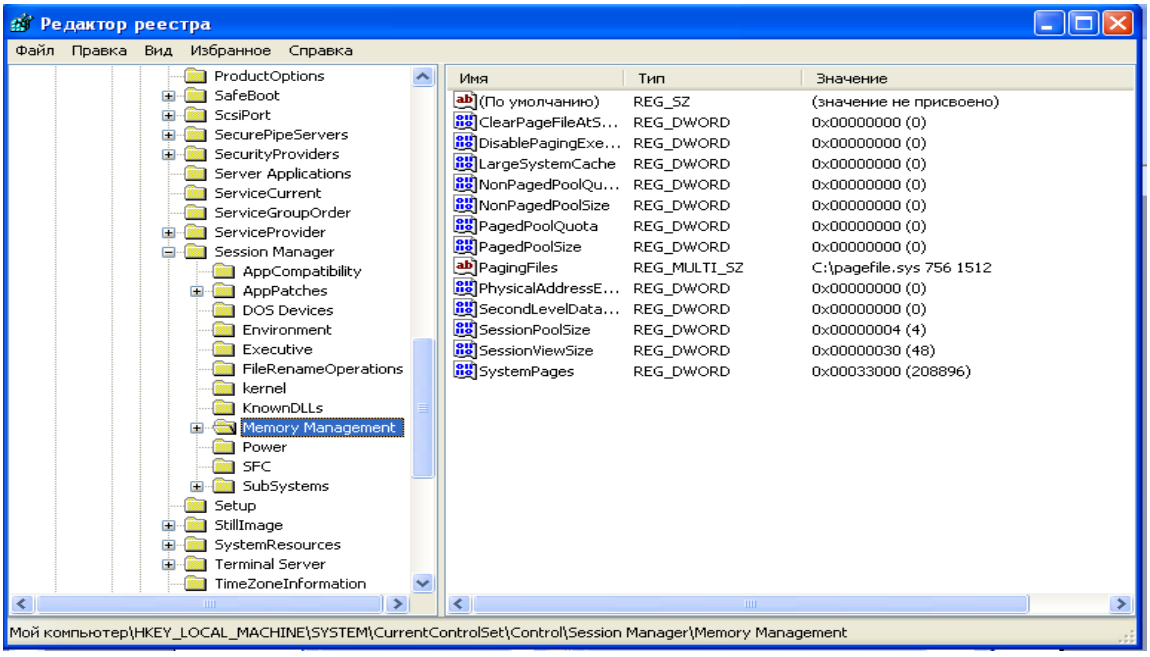


Рисунок 21 – Вікно очистки файла підкачки

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\
Session Manager\Memory Management]
"ClearPageFileAtShutdown"='1'

Слід зауважити, що включення цієї опції може декілька уповільнити процес виключення комп'ютера.

Автоматичне видалення тимчасових файлів після роботи в Інтернет

Значення ключа “0” примусить Internet Explorer видаляти всі тимчасові файли (зображення з web-сторінок та інше), що залишилася після роботи в Інтернет, а значення “1” дозволить залишити ці файли на диску.

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\
CurrentVersion\Internet Settings\Cache]
"Persistent"='0'

Відмінити збереження списку документів, з якими ми працювали

[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\
CurrentVersion\Policies\ Explorer]
"NoRecentDocsHistory"='1'

Відміна збереження інформації про дії користувача

Цей ключ забороняє записувати, з якими додатками недавно працював користувач, і до яких документів він отримував доступ.

[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\
CurrentVersion\Policies\Explorer]
"NoInstrumentation"='1'

Створити пакетний файл для видалення тимчасової інформації

Цей пакетний файл видалятиме всю тимчасову інформацію перед виключенням комп'ютера. Щоб створити його, потрібно виконати декілька простих дій:

- відкрити *Блокнот* і ввести такий текст :
  RD /S /q "C:\Documents and Settings\USERNAME\Local Settings\History"
  RD /S /q "C:\Documents and Settings\Default User\Local Settings\History"
  RD /S /q "C:\Temp\"
  (тут username – ім'я користувача, під яким він заходив в систему, C:\Temp – назва його папки з тимчасовими файлами)
- зберегти цей файл на диску C: під ім'ям deltemp.bat;
- виконати команди: *Пуск→Выполнить*. Ввести ім’я gpedit.msc;
- у лівій частині вікна Групповая политика вибрати послідовно пункти:
  Конфигурация комп'ютера → Конфігурація Windows → Сценарії (запуск/завершення).
- вибрати у правій частині вікна пункт *Завершение работы;*
- у вікні, що з'явилося, натиснути кнопку *Добавить* і вказати, де знаходиться створений нами файл.

Тепер він запускатиметься перед кожним виключенням комп'ютера.

Приховування імені користувача

При роботі з комп'ютером бажано тримати в таємниці не тільки пароль доступу до нього, але і ім'я користувача, оскільки це зменшить вірогідність входу зловмисника в комп'ютер, адже він не знатиме, до чого підбирати пароль.

[HKEY_LOCAL_MACHINE\SOFTWARE\Micro-soft\WindowsNT\
Current Version\Winlogon]
“DontDisplayLastUserName”=’1’.

5.4 Інші міри посилення безпеки

Автозавантаження програм

Для того, щоб вірус або інша шкідлива програма змогла виконати на комп'ютері користувача які-небудь дії, вона, природно, повинна бути завантажена. Зробити це може або сам користувач (наприклад, запустивши на виконання якийсь файл), або операційна система.

Windows зберігає два списки програм, які слід запускати при завантаженні системи. Перший є папкою *Автозавантаження* головного меню, куди користувач може помістити ярлик програми, внаслідок чого вона завантажуватиметься кожного разу при включенні комп'ютера. Другий список є розділами реєстру, в яких знаходяться текстові параметри, в якості значень яких зберігаються шляхи до файлів, які необхідно запустити.

Розділи, де зберігаються ці параметри, називаються:

Run,
RunOnce,
RunOnceEx,

і містяться вони у розділі реєстру HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion.

Параметри можуть мати довільні імена, але за їх значеннями, тобто за іменами файлів, можна з'ясувати, якій програмі належить параметр і, виходячи з цього, ухвалити рішення про те, чи потрібно залишати програму в автозавантаженні.

Також слід звернути увагу на підрозділ Run, розташований в розділі HKEY_CURRENT\USER\Software\Microsoft\Windows\CurrentVersion. У ньому також можуть знаходитися параметри, що запускають програми, проте всі вони поміщаються туди користувачем, а не системою.

Зауважимо, що після установки системи всі ці розділи є порожніми, окрім розділу HKEY_LOCAL_MACHINE\SOFTWARE\ Microsoft\Windows\ CurrentVersion\Run, в якому знаходиться завантажувач SystemTray.

Розділи, в назві яких присутнє слово Once, використовуються для запуску програм тільки один раз, наприклад при запуску програми конфігурації після установлення якого-небудь програмного продукту. Такі ключі після свого запуску автоматично видаляються.

Приховані адміністративні ресурси

При установці операційних систем Windows2000/XP на комп'ютер, системою створюються приховані адміністративні ресурси (ADMIN$, C$, D$ і т.д.), доступні через мережу. Вони призначені для адміністратора комп'ютера і надають повний доступ до всіх жорстких дисків. Особливістю цих ресурсів є те, що їх неможливо закрити стандартними способами (тобто через вкладку *Доступ* діалогового вікна властивостей об'єкта). Якщо ж для видалення цих ресурсів використовувати розділ *Общие папки* аплета *Управление компьютером,* то після перезавантаження вони з'являться знову. Зловмисник може зробити спробу проникнення у комп'ютер з використанням цих ресурсів, тому рекомендується їх закрити. Зробити це можна за допомогою системного реєстру, додавши деякі параметри.

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\
LanmanServer\Parameters]
“AutoShareWks”=’0’ (тип REG_DWORD).

У тому випадку, якщо використовується серверна ОС Windows, ім'я параметра слід змінити на AutoShareServer.

Приховані адміністративні ресурси не видно в таких файлових менеджерах, як *Проводник* або *Total Commander*. Для того, щоб отримати до них доступ, можна використовувати програму FAR і встановлений модуль Network, що підключається. У властивостях даного модуля необхідно задати опцію *Показывть скрытые общие ресурсы.*

Шифрування даних на диску

Як відомо, файлова система NTFS оснащена функцією шифрування даних на диску, тому, якщо у властивостях файлу поставити галочку на відповідній опції, то він буде фізично зашифрований. Крім цього, користувач може маніпулювати правами доступу до певних об'єктів (рис.22).

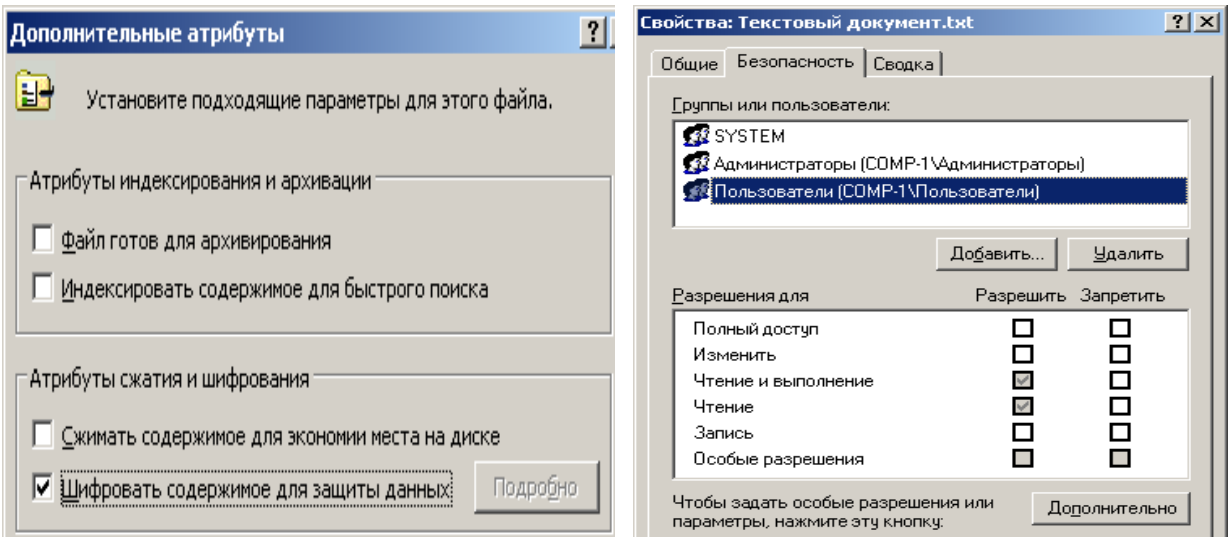


Рисунок 22 – Вікна зміни прав доступудо файлів

Проте при переустановленні Windows або підключенні жорсткого диска до іншого комп'ютера всі вказані раніше права доступу пропадуть, і файл буде доступний абсолютно всім. Якщо ж файл був зашифрований, то при скиданні прав доступу він все одно залишиться зашифрованим, і прочитати його з іншої машини буде неможливо.

Але практично самі файли ключів не зашифровані, оскільки за замовчуванням пароль до них зберігається на тому ж системному диску. Як наслідок, розкрити інформацію у такому разі – швидка і проста справа. А якщо додати сюди масу загальнодоступних утиліт хакерів, що в більшості випадків дозволяють за певний час "розпатрати" будь-який зашифрований файл, достатньо тільки підключити вінчестер до іншого ПК (або завантажитися з компакт-диска Windows PE).

«Зміна» версії Windows

В даний час велике число комп'ютерних вірусів орієнтоване на ті або інші операційні системи або версії цих систем. Наприклад, існують віруси, які перед зараженням комп'ютера перевіряють версію операційної системи і у випадку, якщо вона не співпадає з тією, для якої вони передбачені, не здійснюють на комп'ютері різних несанкціонованих дій. Цей факт можна використати для захисту комп'ютера, оскільки в Windows є можливість «зміни» версії.

Для виконання операції «зміни» версії Windows слід відкрити розділ HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion і знайти в ньому параметри CurrentBuild, CurrentType, CurrentVersion (рис. 23). Можна змінити ці параметри на довільні, забезпечивши, таким чином, захист від зараження деякими типами вірусів.

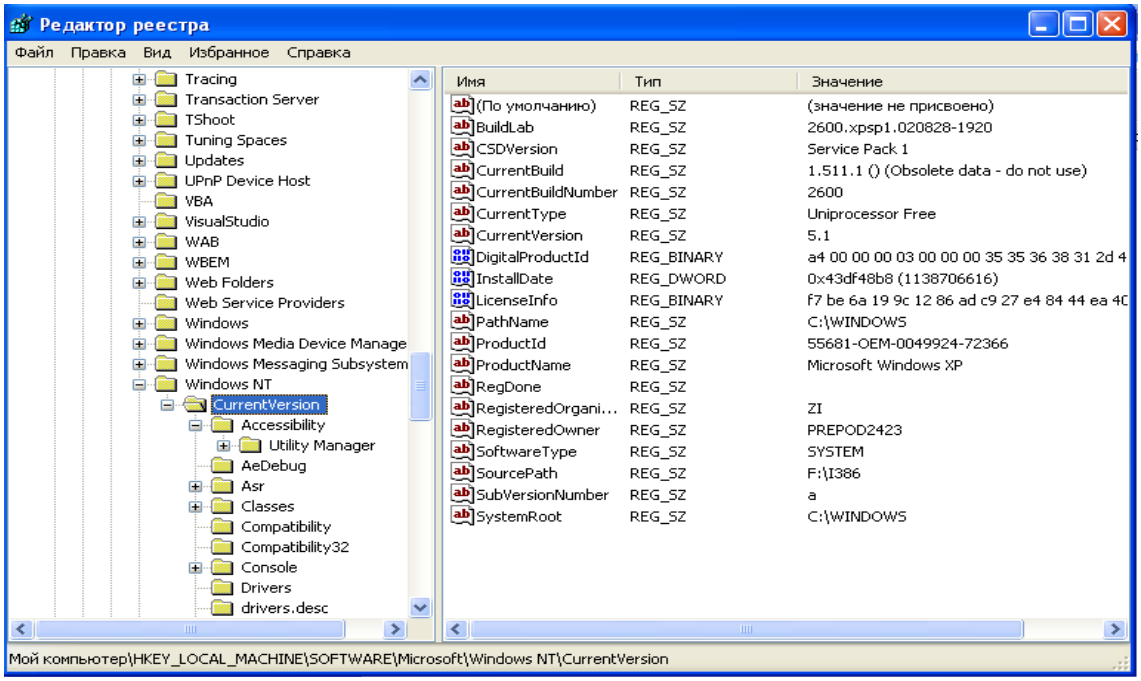


Рисунок 23 – "Зміна" у реєстрі версії Windows

У даному розділі також можна змінити параметр SourcePath, який зберігає шлях до дистрибутива Windows. Якщо даний параметр матиме помилкове значення, то при установці додаткових компонентів інсталятор Windows запрошуватиме дистрибутив.

Пароль після чекаючого режиму (Windows XP)

Можна набудувати систему так, щоб при включенні комп'ютера після *Ждущего режима* з'являлося діалогове вікно із запрошенням ввести пароль.

[HKEY_CURRENT_USER\Software\Policies\Microsoft\Windows\System\Power]
“PromptPasswonlOnResume”=’1’ (тип REG_DWORD).

Видалення зайвих аккаунтів

Уважно подивимося, які на нашому комп'ютері існують аккаунти і видалимо зайві. Для цього виконаємо команди *Мой компьютер → Управління* (рис.24). Далі – в меню *Управление компьютером → Служебные программы → Локальние пользователи и группы → Пользователи*.

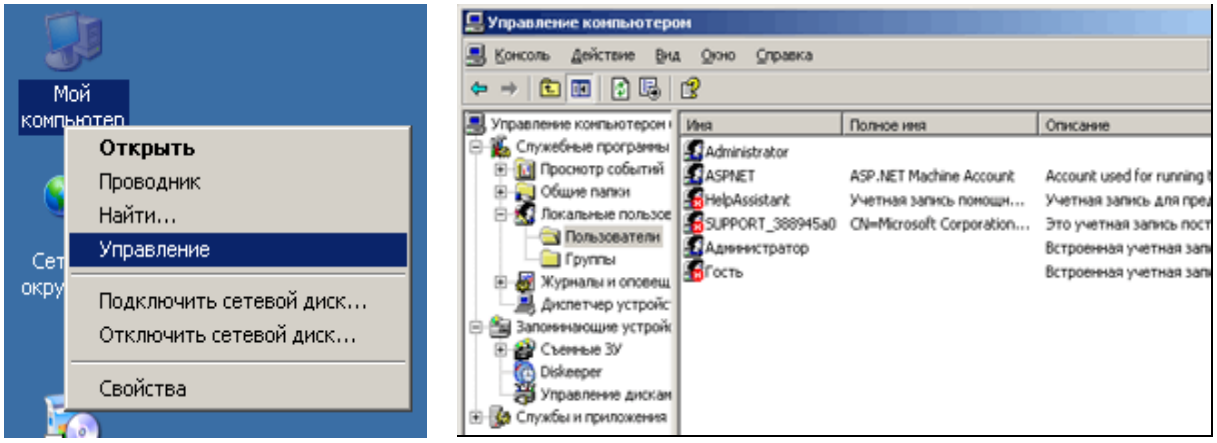


Рисунок 24 – Видалення зайвих аккаунтів

Зауважимо, що різні системні аккаунти чіпати не слід, особливо якщо ми не знаємо, навіщо вони потрібні (хоча можна їх просто відключити і подивитися результати). Видаляються явно непотрібні користувачі.

Зазвичай у Windows XP за замовчуванням виходить два адміністраторські логіни: один стандартний, а другий – першого з користувачів, що увійшли до системи відразу після установки. І через неувагу на одному з них може взагалі не бути пароля. Додатково на всіх існуючих аккаунтах слід встановити паролі не коротше 16 символів! Інакше їх легко розкрити протягом декількох днів або навіть годин.

Як знаємо, у паролі не слід використовувати словарні слова і якінебудь осмислені назви (існують бази, в яких міститься порядка трьох мільйонів популярних парольних слів). Пароль також не повинен складатися з одних тільки цифр. Змінювати пароль можна тільки в меню *Панель управления → Учетные записи пользователей*, інакше доступ до раніше зашифрованих файлів безповоротно пропаде.

Заборона автоматичного входу в Windows

Система повинна завжди запрошувати ім'я і пароль. Якщо вхід в ОС автоматичний, це треба усунути. Йдемо в меню *Пуск → Выполнить*, вводимо команду control userpasswords2 (рис.25). У вікні, що з'явилося, вибираємо опцію "Вимагати введення імені користувача і пароля".

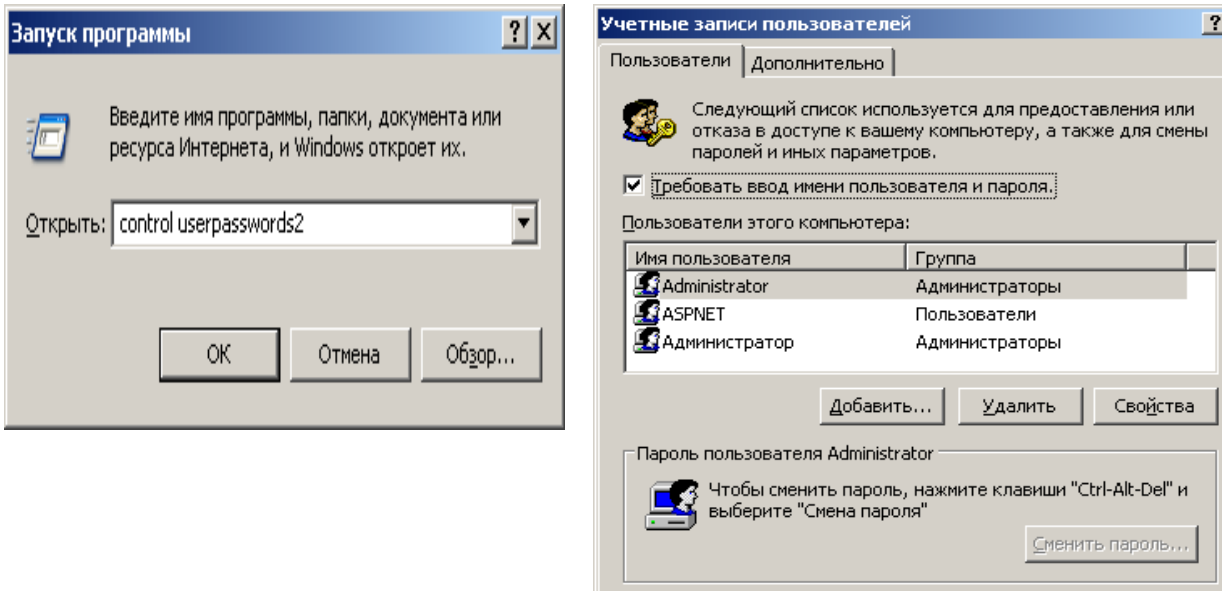


Рисунок 25 – Заборона автоматичного входу у Windows

Заставка за замовчуванням

У Windows існує заставка за замовчуванням, яка запускається, коли комп'ютер включений, але вхід в систему ще не здійснено. Існує можливість змінити як саму заставку, так і деякі її параметри, зокрема час її активації.

Для зміни заставки необхідно помістити файл нової заставки в папку ...\System32 і в розділі реєстру HKEY_USERS\.DEFAULT\ControlPanel\ Desktop змінити рядковий параметр SCRNSAVE.EXE, прописавши як його значення ім'я файлу нової заставки. Стандартним же значенням є logon.scr – файл екранної заставки, який знаходиться в каталозі ...\System32. Окрім цього параметра, можна змінювати параметр ScreenSaveTimeOut, який відповідає за період, через який з'являється заставка.

Оскільки файл з розширенням scr є таким же бінарним файлом, як і інші, він може бути заміщений на «ворожий», що також загрожує непередбаченими наслідками. Щоб уникнути цього, в розділі HKEY_USERS\ .DEFAULT\ControlPanel\Desktop необхідно створити (якщо відсутній) або змінити параметр ScreenSaveActive. Для нього потрібно вказати тип REG_SZ, значення – 0, тобто заставка відключена (за замовчуванням ScreenSaveActive = 1, тобто login.scr запускається автоматично через вказаний в настройках екрану проміжок часу).

Додаткове шифрування файла з паролями

У Windows можна задіювати додаткове шифрування файлу, що зберігає всі паролі. У принципі, його шифрування задіяне спочатку, але пароль, природно, зберігається на цьому ж диску, і програми хакерів розшифровують його "на раз". Набагато безпечніше вводити пароль з клавіатури або з ключової дискети. У таких випадках злом стає практично неможливий. Використовувати ключову дискету не рекомендується через ненадійність носія, через фізичні і людські чинники. А от використовувати введення з клавіатури цілком можна.

Переходимо в меню *Пуск→Виконати*, вводимо команду syskey. В результаті повинно з'явитися вікно управління шифруванням бази даних паролів (рис.26). Натиснемо кнопку *Відновити* для відображення настройок.

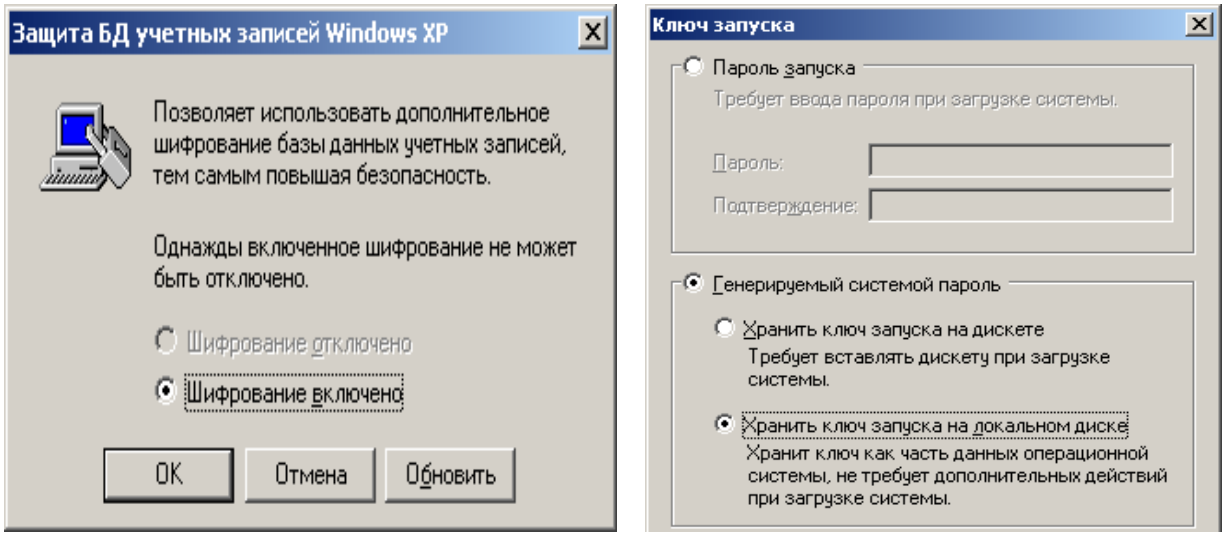


Рисунок 26 – Додаткове шифрування файла з паролями

Як бачимо, за замовчуванням ключ шифрування паролів зберігається в системі. Вибираємо режим вимоги введення пароля при запуску системи. Вказуємо пароль – теж не словарний і довгий. Готово. Тепер ключове слово запрошуватиметься при кожному завантаженні Windows, ще задовго до появи звичного вікна вибору користувача (рис.27).

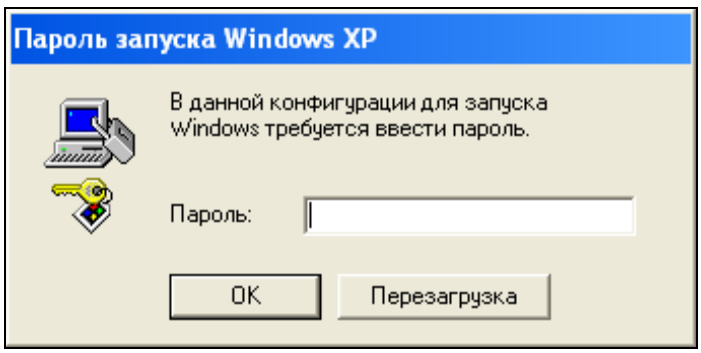


Рисунок 27 – Вікно введення пароля на запуск Windows

Цей пароль – загальний для всіх, його необхідно знати всім користувачам (що, до речі, знімає всі переваги такого додаткового захисту в тому випадку, якщо ламати систему буде один із знаючих пароль користувачів). Звичні персональні призначені для користувача логіни і паролі нікуди не подінуться, вони теж запрошуватимуться як завжди, після введення загального пароля.

Шифрування тимчасових файлів

Відзначимо ще одну потенційну уразливість зашифрованих даних. Більшість програм при роботі створює тимчасові файли, в яких зберігають оброблювані дані (або, наприклад, копію для можливості Undo), і ці тимчасові файли звичайно не зашифровані. Зловмисник може, скориставшися будь-якою з численних Unerase-програм, відновити ці файли.

Для боротьби з цим слід після роботи із зашифрованими файлами затирати порожнє місце на жорсткому диску за допомогою Wipe-утиліт (вони входять, зокрема, в ОС Windows і до складу Norton Utilities, Acronis, PGP Desktop і інших подібних пакетів). У ряді випадків обхід цієї проблеми забезпечується установкою шифрування тимчасової теки (на які посилаються системні змінні %temp% і %tmp%), застосовуючи шифрування не до окремого файлу, а до цілих директорій.

Дозвіл на запуск програм

У Windows є можливість дозволити запуск тільки певних програм, що дозволить уникнути випадкового запуска вірусу або іншої небажаної програми. Для цього програми, дозволені до запуску, повинні бути занесені в спеціальний список, сформований у реєстрі. Для створення списку необхідно виконати вказану послідовність дій:

1) Необхідно відкрити розділ HKEY_CURRENT_USER\Software\Microsoft\Windows\ CurrentVersion\Policies\Explorer.
2) Додати або відредагувати параметр RestrictRun, який має тип REG_DWORD. Для включення цієї функції його значення має дорівнювати 1.
3) У вказаному вище розділі створити розділ RestrictRun і в ньому вказати імена EXE-файлів, які можуть запускати користувачі. Для цього необхідно в розділі RestrictRun створити параметри типа REG_SZ і як їх значення вказати імена виконуваних файлів програм (при цьому можна вказати шлях до файлу).
4) Перезавантажити комп'ютер для того, щоб зміни набули чинності.

Обов'язково слід внести в цей список редактор реєстру (regedit.exe), інакше система не дасть його завантажити і неможливо буде внести в нього які-небудь зміни. Слід також врахувати те, що будь-який користувач може перейменувати будь-який файл в дозволений (наприклад game.exe в regedit.exe) і запустити будь-яку програму.

Заборона виклику Диспетчера задач

Для того, щоб позбавити користувача можливості проглядати список процесів, запущених на комп'ютері, змінювати їх пріоритет, слід використовувати параметр DisableTaskMgr типа REG_DWORD, розташований в розділі HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Po!icies\System. При установці значення цього параметра, рівного 1, користувач не зможе викликати Диспетчер задач. Окрім цього, він не зможе вимкнути, перезавантажити або припинити роботу комп'ютера за допомогою Диспетчера задач.

Заборонити запуск аплетів в Панелі управління

Якщо необхідно позбавити користувача можливості настроювати комп'ютер за допомогою *Панели управления*, необхідно в розділі HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer створити параметр NoControlPanel типа REG_DWORD. При цьому буде заборонений запуск аплетів в *Панели управления*, а також заборонений запуск CPL-файлів.

Заборонити зміну властивостей Екрану

Параметр NoDispCPL типа REG_DWORD забороняє запуск аплета настройки екрану (настройки тим, фонового малюнка, дозволу екрану і ін.). Він знаходиться в розділі HKEY_CURRENT_USER\Software\Microsoft\Windows\ CurrentVersion\Policies\ System.

Також можна просто заборонити зміну фону *Рабочего стола*, для чого слід створити параметр NoChangingWallpaper типу REG_DWORD в розділі HKEY_CURRENT_USER\Software\Micro-soft\Windows\CurrentVersion\Policies\ ActiveDesktop і встановити його рівним 1.

Зробити недоступним контекстне меню Провідника

Дуже часто системні утиліти, які встановлюються на комп'ютері, використовують контекстне меню *Проводника* для швидкого виклику програми і обслуговування того або іншого об'єкту. Це може послужити причиною випадкового видалення або пошкодження даних користувачами, що не вміють працювати з такими програмами. Для того, щоб при клацанні правою кнопкою миші, а також при натисненні [Shift+F10] контекстне меню *Проводника* не відображалося, необхідно в розділі HKEY_CURRENT_USER\ Software\Microsoft\Windows\CurrentVersion\Policies\Explorer створити параметр NoViewContextMenu типу REG_DWORD і присвоїти йому значення 1. Щоб зміни набули чинності, необхідно перезавантажити комп'ютер.

Контекстне меню папок і файлів

Якщо приховування контекстного меню небажане, є можливість його редагування. Для того, щоб залишити в ньому тільки ті програми, які дійсно необхідні, треба знайти розділ HKEY_CLASSES_ROOT*\shellex\Context-Menu-Handlers. Тут як підрозділи виступають команди, що відображаються в

контекстному меню будь-якого файлу, і для редагування меню слід просто видалити ті розділи, які є зайвими.

Запис інформації про доступ до файлу

Файлова система NTFS дозволяє вести запис інформації про останній доступ до файлів. Ці дані можуть бути корисні, якщо виникає підозра, що який-небудь файл був викрадений користувачем, що працював за комп'ютером. Також даний параметр дозволяє прискорити доступ до каталогів з великою кількістю файлів.

Щоб включити цю опцію, необхідно в розділі HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\FileSystem присвоїти "1" параметру NtfsDisableLastAccessUpdate (на роботу файлової системи FAT32 опція не впливає).

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\FileSystem]
"NtfsDisableLastAccessUpdate" = '1'

Відключення стеження Windows XP за користувачем

Даний параметр дозволяє користувачу операційної системи Windows XP відключити запис інформації про такі дії, як запуск програм, відкриття документів, зміна списків програм, що часто викликаються, списків недавно створених документів і т.д. Параметр може використовуватися в цілях безпеки користувача.

Слід зазначити, що адміністратору комп'ютера настійно рекомендується не відключати цей параметр, оскільки дані, записувані ОС, можуть допомогти в знаходженні зловмисника.

Для зміни параметра необхідно знайти розділ HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\ Policies\Explorer і змінити параметр NoInstrumentation типа REG_DWORD у відповідностями з вимогами (значення 1 дозволяє відключити стеження, значення 0 - включити).

Запис подій в системний журнал

Якщо виникає необхідність у визначенні моменту, коли відбулася яка-небудь помилка або комп'ютер був аварійно перезавантажений, слід в розділі HKEY_LOCAL_MACHINE\SYSTEM\Cur-rentControlSet\Control\CrashControl створити параметр LogEvent типа REG_DWORD і присвоїти йому значення 1. Особливо ця функція корисна на серверах, а також на робочих станціях за наявності декількох користувачів на них.

Приховування папок документів в програмі «Мой компьютер»

Якщо необхідно приховати в програмі *Мой компьютер папки* документів, такі як *Мои документи*, *Общие документы* і т.д., необхідно видалити ключ HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Exp|orer\MyComputer\Name-Space\DelegateFolders\{59031a47-3f72-44a7-89c5-5595fe6b30ee>. Створення цього ключа дозволяє повернути папки на місце.

Швидке перемикання користувачів (Windows XP)

Windows XP дозволяє працювати на комп'ютері декільком користувачам одночасно, – для цього достатньо лише перемикатися між ними за допомогою функції *Быстрое переключение пользователей.* Якщо опція включена, то при перемиканні на іншого користувача програми поточного користувача продовжуватимуть працювати, інакше програми автоматично вимикатимуться, коли користувач виходить з системи, і з наступним користувачем комп'ютер працюватиме швидше.

Проте ця опція може таїти в собі приховану загрозу. Прикладом може служити наступна ситуація: користувач А запускає на комп'ютері програму стеження за клавіатурою, а потім не завершує сеанс, а просто відображає *Экран приветствия* за допомогою натиснення комбінації клавіш Win+L. Користувач В намагається завантажити свій *Рабочий стол* і, якщо він захищений паролем, вводить цей пароль. Дана інформація буде записана програмою стеження за клавіатурою, і, таким чином, користувач А дістане доступ до профілю користувача В.

Для запобігання виникнення подібної ситуації рекомендується відключити дану функцію. Для цього в разділі HKEY_LOCAL_MACHINE\ SOFTWARE\Microsoft\Windows\CurrentVersion\Winlogon необхідно змінити параметр AllowMultipleTSSessions типу REG_DWORD з 1 на 0. Також вкрай бажано не використовувати *Экран приветствия,* який хоча і забезпечує найшвидший і простіший вхід в систему, але є уразливою функцією Windows XP. Якщо вимкнути цю опцію, використовуватиметься класичний вхід в систему. Для відключення *Экрана приветствия* необхідно в розділі HKEY_LOCAL_ MACHINE\SOFTWARE\Microsoft\WindowsNT\Cur-rentVersion\Winlogon змінити параметр LogonType типе REG_ DWORD. Значення 1 дозволяє використовувати *Экран приветствия,* 0 - використовується класичний вхід в систему.

Заборонити доступ до дисків

Якщо необхідно заборонити проглядання вмісту жорстких дисків комп'ютера за допомогою файлового менеджера *Проводник*, необхідно використовувати параметр NoViewOnDrive типа REG_ DWORD, який розташований в розділі HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\ Policies\Exp!orer. Його значенням є сума дисків, до яких забороняється доступ. При цьому кожен диск відмічається числом:

A: 1 (2°)
B: 2(2^1)
C: 4 (2^2)
D: 8 (2^3)
...Z: 33554432 (2^{25})

Наприклад, щоб заборонити доступ до дисків A: і C:, необхідно задати значення параметра NoViewOnDrive рівним 5 (що відповідає сумі 1+4=5).

Значення 0 – дозволений доступ до всіх дисків; значення 67108863 (hex: 3ffffff) – заборонити доступ до всіх дисків (A: – Z:).

AutoRun (автозапуск CD)

Можливість запуску «випадкових» програм робить доцільним відключення функції *Autorun.* Розв'язати цю проблему можна двома способами: а) при кожному запуску CD-ROM натискати і утримувати клавішу Shift; б) в розділі HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\CDRom змінити значення параметра Autorun (тип REG_DWORD) з 1 (задано за замовчуванням) на 0.

Безпека ядра

Уразливість Windows 2000 полягає у тому, що ядро цієї системи можна пошкодити як у разі дій шкідливої програми, так і внаслідок невмілих дій користувача. Але також можна і відновити як ядро, так і просто функціональність системи. Тому для попередження можливих неполадок в роботі системи, пов'язаних з пошкодженням ядра, рекомендується створити резервну копію ядра. Для цього в каталог ...\System32 слід скопіювати файли ntoscrnl.exe і hal.dll, заздалегідь трохи змінивши їх імена, наприклад, ntoscrnla-larm.exe і halalarm.dll.

У разі виникнення позаштатних ситуацій у файлі boot.ini до рядка завантаження системи, що звичайно має вигляд:

```
multi(0) disk(0) rdisk(0) partition(l)\Windows=
"Windows XP Professional" /fastdetect
```

додається (після /fastdetect) рядок:

```
/kernel= ntoscrnlalarm.exe / hal=halalarm.dll
```

після чого пошкодження ядра не викличе збою, оскільки система використовуватиме створені копії.

Слід тверезо усвідомлювати, що вищеперелічені заходи не дають стовідсоткової гарантії безпеки. Якщо диск вкрадений цілеспрямовано, і дані явно являють собою комерційний інтерес, то зловмисники, швидше за все, знайдуть можливість якось обійти ці заходи. І в такому випадку необхідне спеціалізовані програмні та технічні засоби захисту.

Контрольні питання

1. Наведіть додаткові мірі безпеки для забезпечення функціонування паролів.
2. Які додаткові міри безпеки можна прийнятиии прии роботі у мережі?
3. Як можна приховати сліди роботи на комп’ютері?
4. Чи можна зашифрувати тимчасові файли на термін безпосередньої роботи у Windows? Якщо можна, то як?
5. Як заборонити автоматичне завантаження операційної системи?

6 ПАКУВАННЯ, АРХІВАЦІЯ І ШИФРУВАННЯ ДАНИХ В ОПЕРАЦІЙНИХ СИСТЕМАХ

6.1 Історичні відомості

У ті далекі часи, коли обсяг жорстких дисків (вінчестерів) вимірювався мегабайтами – цих мегабайтів завжди не вистачало, і більшість файлів (особливо рідко використовуваних) зберігали в упакованому вигляді. Перед запуском файл розпаковували, а після завершення роботи – упаковували знову, щоб звільнити місце для розпакування інших.

Коли ці махінації всім остаточно набридли, програмісти (згадавши, що комп'ютер повинен служити людині, а не навпаки), додумалися до автоматичного розпакування файлів, що виконуються, "на льоту". Ідея полягає в дописуванні до стиснутого файла крихітного распаковщика, якому передається керування при запуску файла, і який розпаковує код, що виконується, не на диск, а безпосередньо в оперативну пам'ять. Звичайно, час завантаження при цьому відчутно збільшувався (особливо на машинах з повільними процесорами), але це з надлишком виправдовувалося простотою запуску й економією дискового простору.

Незабаром пакувальників розвелося сила-силенна (їх тоді писали всі, кому хотілося) – AINEXE, DIET, EXEPACK, LZEXE, PKLITE і інших – усіх не перелічити! І не дивно: процесори з дня у день ставали усе продуктивнішими – уже на "трійці" розпакування займало настільки незначний час, що їм можна було зневажити. До того ж приємним побічним ефектом виявився захист від дизасемблювання. Дійсно, безпосередньо дизасемблювати упакований файл неможливо – його необхідно розпакувати. Звичайно, на кожен щит знайдеться свій меч – з-під пера хакерів вийшло чимало чудових універсальних распаковщиков (UNP, Intruder, UUP, а вершиною усьому став CPU386 з вбудованим емулятором реального режиму процесора 80386), але якість автоматичного розпакування залишала бажати кращого (часом розпаковані файли зависали при запуску чи в процесі роботи), а ручним трасуванням володіли далеко не всі.

Словом, при усіх своїх достоїнствах, пакування виконуваних файлів не мало ніяких недоліків і не збиралося здавати позицій навіть із приходом ємних (по той час) одно-, двогігабайтних дисків і CD-ROM.

6.2 Стискання файлів під Windows 9x\NT

Пройшов невеликий час і світ повільно, але неминуче пересаджувався на нову операційну систему – Windows 95. Користувачі обережно ос-

воювали мишу і графічний інтерфейс, а програмісти тим часом гарячково переносили старе програмне забезпечення на нову платформу. Обсяги вінчестерів на той час виросли настільки, що розробники могли забути слово "оптимізація", так вони, судячи з розміру сучасних додатків, його і забули. Сто мегабайтів туди, триста сюди, – запросто.

Тоді і згадали про розпакування виконуваних файлів "на льоту".

6.2.1 Стискання виконуваних файлів

На ринку з'явилося декілька програм-компресорів, з яких найбільшу популярність завоювала програма ASPack, що вміє стискати і розтискати не тільки "екзешники", але і динамічні бібліотеки. А до складу самої Windows 95 увійшла динамічна бібліотека LZEXPAND.DLL, яка підтримувала базові операції пакування-розпакування і "прозору" роботу зі стиснутими файлами. Користувачі і програмісти швидко скористалися новими засобами, але...

На відміну від MS-DOS, у Windows 9x\NT за автоматичне розпакування приходиться платити більше, ніж одержувати. Згадаємо, як у MS-DOS відбувалося завантаження виконуваних модулів. Файл цілком зчитувався з диска і копіювався в оперативну пам'ять, причому найбільш вузьким місцем була саме операція читання з диска. Пакування навіть прискорювало завантаження, оскільки фізично читався менший обсяг даних, а їх розпакування займало дуже короткий час.

У Windows же завантажник читає лише заголовок і таблицю імпорту файла, а потім проектує його на адресний простір процесу так, ніби-то файл є частиною віртуальної пам'яті, що зберігається на диску (взагалі ж, все відбувається набагато складніше). Підкачування з диска відбуваються динамічно – у міру звернення до відповідних сторінок пам'яті, причому завантажуються тільки ті з них, що дійсно потрібні.

Наприклад, якщо в текстовому редакторі є модуль роботи з таблицями, він не буде завантажений з диска доти, поки користувач не захоче створити (чи відобразити) свою таблицю. Причому не має значення – чи знаходиться цей модуль у динамічній бібліотеці, чи в основному файлі. Завантаження таких "монстрів", як Microsoft Visual Studio і Word, ніби "розмазується" у часі, і до роботи з додатком можна приступати практично відразу ж після його запуску. А що ж відбудеться, якщо файл упакувати? Він повинен буде зчитуватися з диска цілком (!) і потім – знову-таки, цілком – розпакуватися в оперативну пам'ять.

Але ж нашої оперативної пам’яті явно не вистачить і розпаковані сторінки прийдеться знову скидати на диск. Причому, якщо при проектуванні неупакованого exe-файла оперативна пам'ять не виділяється (у всякому разі, доти, поки в ній не виникне необхідність), распаковщику без пам'яті ніяк не обійтися. А оскільки оперативної пам'яті ніколи не

буває забагато, вона може бути виділена лише за рахунок інших додатків. Відзначимо також, що в силу конструктивних особливостей заліза й архітектури операційної системи, операція записування на диск є помітно повільнішою за операцію зчитування.

Важливо зрозуміти: Windows ніколи не скидає на диск не модифіковані сторінки файла, що проектується. Навіщо це? Адже в будь-який момент їх можна знову зчитати з оригінального файла. Але при розпакуванні модифікуються всі сторінки файла! Виходить, система буде змушена "ганяти" їх між диском і пам'яттю, що істотно знизить загальну продуктивність усіх додатків у цілому.

6.2.2 Стискання динамічних бібліотек

Ще більші накладні витрати спричиняє стискання динамічних бібліотек. Для економії пам'яті сторінки, зайняті динамічною бібліотекою, спільно використовуються всіма процесами, що завантажили цю DLL. Але як тільки один із процесів намагається щось записати в пам'ять, зайняту DLL, система миттєво створює копію сторінки, що модифікується, і надає її в "монопольне" розпорядження процесу-“письменнику”. Оскільки розпакування динамічної бібліотеки відбувається в контексті процесу, що завантажив її, система змушена багаторазово дублювати всі сторінки пам'яті, виділені бібліотеці, фактично надаючи кожному процесору свій власний екземпляр DLL. Припустимо, одна DLL розміром у 1 Мбайт, була завантажена десятьма процесами – порахуємо, скільки пам'яті буде дарма втрачено, якщо вона стиснута!

Таким чином, під Windows 9x\NT стискати виконувані файли недоцільно – ми платимо набагато більше, ніж отримуємо. Що ж стосується захисту від дизассемблювання, то, коли ASPack тільки з'явився, він віднадив від зламу дуже багатьох некваліфікованих хакерів, але ненадовго. Сьогодні в мережі легко можна знайти посібники з так званого ручного зняття ASPack. Існує і маса готового інструментарію – від автоматичних распаковщиков до плагінів для дизасемблера IDA Pro, що дозволяють йому дизасемблювати стиснуті файли. Тому сподіватися, що ASPack цілком врятує нашу програму від зламу, не слід.

6.3 Продуктивність пакування файлів

Стосовно вимірювання зменшення продуктивності від пакування файлів, то тут, здавалося б, немає нічого складного – беремо неупакований файл, запускаємо його, заміримо час завантаження, записуємо результат на папірці, упаковуємо, запускаємо ще раз, і...

Перший камінь спотикання – що розуміти під "часом завантаження"? Якщо проектування – так воно виконується практично миттєво, і їм можна взагалі зневажити. Моментом часу, починаючи з якого з програмою можна повноцінно працювати? Так це від самої програми залежить більше, ніж від його упакування. До того ж, на час завантаження упакованих файлів дуже сильно впливає кількість вільної на момент запуску фізичної оперативної пам'яті (не слід плутати з загальним обсягом пам'яті, установленої на машині). Якщо перед запуском упакованого файлу ми завершимо один-два великі програмні додатки-"монстри", то зайнята ними пам'ять виявиться вільною, і зможе безперешкодно використовуватися распаковщиком. Але якщо вільної пам'яті немає, її прийдеться по крихтах відривати від інших додатків...

Навіть якщо ми оцінимо зміну часу завантаження (що, до речі, зробити дуже проблематично – серія вимірів на одній і тій самій машині, з тим самим набором додатків дає розкид результатів більш ніж на порядок), як вимірювати падіння продуктивності інших додатків? Адже, при недостачі пам'яті Windows у першу чергу рятується від немодифікованих сторінок, які немає необхідності зберігати на диску! У результаті пакування виконуваного файла може дещо підвищити продуктивність роботи самого цього файла, але значно погіршити стан інших, неупакованих додатків.

Тому ніяких конкретних цифр навести не можна. Наближені оцінки, виконані "на око", показують, що при наявності практично необмеженої кількості оперативної пам'яті втрати продуктивності складають менше 10%, але при її недостачі швидкість усіх додатків падає від двох до десяти разів! В експерименті брали участь файли, що виконуються, MS Word 2000, Visual Studio 6.0, Free Pascal 1.04, IDA Pro 4.17, Adobe Acrobat Reader 3.4, машина з процесором CLERION-300A, оснащена 256 МБ ОЗУ, для імітації недостачі пам'яті її обсяг зменшувався до 64 МБ; використовувалися операційні системи Windows 2000 і Windows 98.

Таким чином, можна зробити такі *висновки та рекомендації*:

1) Файли, що виконуються під Windows, краще не пакувати. У крайньому випадку – використовувати для пакування/розпакування функції операційної системи (LZInit, LZOpenFile, LZRead, LZSeek, LZClose, LZCopy), динамічно розпаковуючи в спеціально виділений буфер тільки ті частини файла, що дійсно потрібні в даний момент для роботи.
2) Динамічні бібліотеки взагалі не слід пакувати, оскільки це веде до дивовижної витрати і фізичної, і віртуальної пам'яті і псує саму концепцію DLL: один модуль – усім процесам.
3) Не слід прагнути базувати свій додаток на великій кількості DLL, – сторінки виконуваного файла не вимагають фізичної пам'яті доти, поки до них не відбувається звернень. Тому сміливо можна поміщати весь код програми в один файл.

6.4 Принципи роботи програм-архіваторів

Принцип роботи архіваторів заснований на пошуку у файлі "надлишкової" інформації і наступному її кодуванні з метою одержання мінімального обсягу.

Стискання послідовностей однакових символів

Стискання послідовностей однакових символів – найвідоміший метод архівації файлів. Наприклад, усередині нашого файлу знаходяться послідовності байтів, що часто повторюються. Замість того, щоб зберігати кожен байт, фіксується кількість повторюваних символів і їхня позиція. Наприклад, файл, що буде архівуватися, займає 15 байт і складається з наступних символів:

B B B B B L L L L L A A A A A

або у шістнадцятковій системі

42 42 42 42 42 4C 4C 4C 4C 4C 41 41 41 41 41 .

Архіватор може представити цей файл у такому вигляді (шістнадцятковому):

01 05 42 06 05 4C 0A 05 41 .

Це означає: з першої позиції п'ять разів повторюється символ "B", з позиції 6 п'ять разів повторюється символ "L" і з позиції 11 п'ять разів повторюється символ "A". Для збереження файла в такій формі буде потрібно всього 9 байт, що на 6 байт менше вихідного.

Описаний метод є простим і дуже ефективним способом стискання файлів. Однак він не забезпечує великої економії обсягу, якщо текст містить невелику кількість послідовностей повторюваних символів.

Алгоритм Хаффмана

Більш витончений метод стискання даних, використовуваний у тому чи іншому вигляді практично будь-яким архіватором, – це так званий *оптимальний префіксный код* і, зокрема, кодування символами змінної довжини (алгоритм Хаффмана). Код змінної довжини дозволяє записувати символи і групи символів, що найбільш часто зустрічаються, усього лише декількома бітами, у той час як рідкі символи і фрази будуть записані більш довгими бітовими рядками. Наприклад, у будь-якому англійському тексті буква E зустрічається частіше, ніж Z, а X і Q відносяться до тих, що найменш зустрічаються. Таким чином, використовуючи спеціальну таблицю відповідності, можна закодувати кожну букву E меншим числом біт і використовувати більш довгий код для більш рідких букв.

Алгоритму Лемпела-Зіва

Популярні архіватори ARJ, PAK, PKZIP працюють на основі алгоритму Лемпела-Зіва. Ці архіватори класифікуються як адаптивні словникові кодувальники, у яких текстові рядки заміняються покажчиками на іден-

тичні їм рядки, що зустрічаються раніше в тексті. Наприклад, усі слова якої-небудь книги можуть бути представлені у вигляді номерів сторінок і номерів рядків деякого словника. Найважливішою відмінною рисою цього алгоритму є використання граматичного розбору попереднього тексту з розбиттям його на фрази, що записуються у словник. Покажчики дозволяють зробити посилання на будь-яку фразу у вікні встановленого розміру, що передує поточній фразі. Якщо відповідність знайдена, що текст фрази заміняється покажчиком на свого попереднього двійника.

При архівації, як і при компресуванні, степінь стискання файлів сильно залежить від формату файлу. Графічні файли типу TIFF і GIF уже заздалегідь скомпресовані (хоча існує різновид формату TIFF і без компресії) і тут навіть найкращий архіватор мало що знайде для пакування. Зовсім інша картина спостерігається при архівації текстових файлів, файлів PostScript, файлів .BMP і їм подібних.

6.5 Програми архівації файлів

Архівний файл являє собою набір з одного або декількох файлів, розміщених у стисненому вигляді в одному файлі, з якого при необхідності їх можна дістати у первісному вигляді. Архівний файл мість зміст, який дозволяє побачити, які саме файли знаходяться в архіві. Для кожного стисненого і поміщеного в архів файла зберігається така інформація:

- ім’я файла;
- відомості про каталог, в якому знаходиться файл;
- дата і час останньої модифікації файла;
- розмір фала на диску і в архіві;
- код циклічного контролю для кожного файла, що використовується для перевірки цілісності архіва.

Найбільш популярні архіватори – WinZip і WinRAR – дозволяють задавати пароль на відкриття архіва. Але існують програми, за допомогою яких можна зламати архівні файли. Ці програми-зламщики архівів можна умовно розбити на дві групи?

- *спеціалізовані* – програми, які зламують паролі лише одного архіватора (наприклад, програма Azpr зламує пароль архіватора WinZip);
- *універсальні* – програми, що працюють з двома і більше видами архіваторів (наприклад, програма Archpr працює з усіма видами архівів під Windows). Недоліком таких програм є їх великий об’єм.

Більшість сучасних програм пакування даних мають *вбудовану підтримку шифрування.* Якщо користувач бажає захистити від чужих очей свою інформацію в архіві, йому необхідно при пакування ввести пароль, і архіватор далі сам виконає необхідні дії. При спробі видобути зашифрований файл архіватор вимагатиме у користувача пароль і розпакує файл лише тоді, коли пароль вказано правильно.

Слід зауважити, що шифрування відбувається завжди після компресуваня, оскільки зашифровані дані не повинні відрізнятися від випадкової послідовності, і, як наслідок, архіватор не зможе знайти в них надлишковість, за рахунок видалення якої і відбувається пакування.

6.5.1 ZIP

Одним з найпопулярніших форматів стискання даних серед користувачів операційної системи Windows був і залишається ZIP, розроблений компанією PKWARE, Inc. Широкого розповсюдження цей формат набув зовсім не через технічні особливості – швидке стискання, високий степінь упаковки. Існують архівні формати, що переважають ZIP за багатьма характеристиками. Скоріш за все, формат ZIP завдячує своїй популярності умовно безкоштовній програмі WinZIP.

WinZIP є умовно безкоштовним продуктом. Будь-який користувач має право установити WinZIP і використовувати його на протязі 30 днів у тестових і ознайомлювальних цілях. При цьому програма є повнофункціональною, але іноді з'являється вікно повідомлення з пропозицією купити WinZIP. По завершені тестового періода необхідно отримати ліцензію або видалити програму з комп'ютера. Після оплати користувач отримує реєстраційний код, що відповідає його імені. Після введення правильного кода у відповідному вікні WinZIP програма вважається зареєстроваою і припиняє турбувати користувача пропозиціями про купівлю програми.

Для шифрування архівів формату ZIP використовується потоковий алгоритм шифрування, розроблений Роджером Шлафлай. Але у 1994 році Елі Біхем і Пол Кошер опублікували статтю, присвячену атаці на алгоритм шифрування формату ZIP, в якому для знаходження ключа шифрування досить знати 13 послідовних байтів відкритого тексту і виконати певну операцію 2^{38} разів.

Для архівів, що містять 5 і більше файлів і створених на основі бібліотеки InfoZIP, можлива атака, що використовує в якості відкритого тексту дані з заголовків зашифрованих файлів. Цій атаці пядлягли архіви, створені за допомогою програми WinZIP, але в останніх версіях цього архіватора проблема була дещо виправлена.

Не дивлячись на те, що недоліки алгоритму шифрування ZIP давно відомі, він до цих пір лишається одним з найчастіше використовуваних для архівів формату ZIP. Деякий час тому в архіваторах PKZIP і WinZIP з'явилась підтримка інших, більш стійких алгоритмів шифрування, але нове шифрування не дуже популярне з деяких причин. По-перше, нові формати зашифрованих даних в PKZIP і WinZIP не сумісні між собою, що не дозволяє прочитати одним архіватором те, що створено іншим. По-друге, компанія PKWARE, що створила PKZIP, звинувачує авторів WinZIP у то-

му, що, реалізувавши своє шифрування, вони порушують патенти, що належать корпорації PKWARE.

У 2002 році компанія PKWARE випустила версію архіватора PKZIP, яка підтримувала більш стійкі алгоритми шифрування. Але через те, що PKZIP був розрахований на корпоративних користувачів, нове шифрування не отримало достатньої популярності.

Отже, алгоритм, що використовується в WinZIP для перевірки відповідності рестраційного кода імені користувача, давно відкритий, і в Інтернеті можна без особливих зусиль знайти початкові тексти і готові програми для обчислення цього кода. Молоймовірно, що в WinZIP Computing не знають про існування генератора кодів до їх програми, але на протязі багатьох версій схема реєстрації не змінювалась і, схоже, змінюватись не буде. Не зважаючи на порівняну простоту отримання повністю дієздатної копії WinZIP без оплати вартості ліцензії, утруднення схеми реєстрації навряд чи викличе різке збільшення об'ємів продаж. А от витрати на оновлення реєстраційних номерів у всіх легальних користувачів можуть виявитись зовсім не маленькими.

6.5.2 ARJ

Популярний з часів DOS, але рідко використовуваний в даний час архіватор, розроблений Робертом Янгом, базується на такому алгоритмі. З пароля, за дуже простим обертаємим алгоритмом отримувалась гама, за довжиною рівна паролю. Ця гама накладалась на дані методом додавання за модулем 2 (операція XOR). Таким чином, наявність відкритого тексту, рівного довжині пароля, дозволяла миттєво визначити гаму і використовуваний пароль.

Більш того, на самому початку упакованих даних містилась інформація компресора така, як таблиці Хаффмана, і частина цієї інформації могла бути передбачена, що дозволяло значно підвищити швидкість пошуку пороля перебором.

Починаючи з версії 2.60 (листопад1997 року), ARJ підтримує шифрування за алгоритмом ГОСТ 28147-89.

6.5.3 RAR

Архіватор, розроблений Євгеном Рошалем, є непоганим прикладом того, як можна підходити до шифрування даних.

В алгоритмі шифрування, використаному у RAR версії 1.5, є деякі недоліки. Так, ефективна довжина ключа ширування складє всього 64 біти, тобто перебором 2^{64} варіантів ключа можна гарантовано розшифрувати пароль. Більш того, наявність відкритого тексту дозволяє зменшити кількість

варіантів перебору до 2^{40}. Отже, атака успішно може бути виконана навіть на одному комп'ютері (швидкість перебору на комп'ютері з процесором Intel Pentium III 333 МГц складає приблизно 600 000 паролів в секунду).

У версії 2.0, вочевидь, була проведена серьйозна робота над помилками. Злам нового алгоритма шифрування перебором вимагав вже приблизно 2^{1023} операцій, що більше, ніж це можна було б здійснити на сучасній техніці. Про ефективні атаки, що використовують відкритий текст, офіційно нічого не відомо. Швидкість перебору паролів знизилась приблизно до 2000 штук за секунду (в 300 разів).

Але розробники RAR продовжили роботу. У версії RAR 3.0 (травень 2002 року) для шифрування став використовуватись алгоритм AES з ключем довжиною 128 бітів. Таке рішення було викликано двома причинами. По-перше, безпечніше використовувати перевірений і добре себе зарекомендувавший алгоритм, ніж дещо саморобне, а у AES тут немає конкурентів. По-друге, у AES швидкість шифрування вища, ніж у алгоритма, використовуваного в RAR 2.0.

Окрім заміни алгоритму шифрування, в RAR 3.0 використовується і інша процедура отримання ключа шифрування з пароля. Ця процедура вимагає обчилення хеш-функції SHA1 262144 рази, що дозволяє перебирати лише до 3-х паролів в секунду, тобто в 600 разів менше, ніж для RAR 2.0.

6.6 Шифрування файлів у програмах Microsoft

На протязі багатьох років програми, що входять у Microsoft Office, дозволяють шифрувати документи за паролем, який вводить користувач. Але не завжди файли виявляються захищеними надійно.

Microsoft Word і Mirosoft Excel

Шифрування файлів було реалізоване вже в Microsoft Word 2.0. З пароля за допомогою простого обертаємого алгоритма отримувалась 16-байтова гама, яка накладалась на вміст документа. Але обчислення гами баз пароля не складало труднощів, оскільки гама накладалась на службові області, які мали фіксоване значення у всіх документах.

У Word 6.0/95 та Excel 5.0/95 алгоритм шифрування не змінився, а змінився лише формат файлів – він став базуватися на OLE Structered Storage. Для відновлення пароля також треба було знайти 16-байтову гаму, використану для шифрування. А знайти цю гаму можна було, базуючись на статистичному аналізі. У будь-якому тексті найчастіше зустрічається символ " " (пробіл). Таким чином, досить визначити код найчастіше використовуваного у тексті символа в кожній з 16 позицій, що відповідають різним байтам гами. Виконавши операцію XOR кожного знайденого значення з кодом пробіла (0x20), отримуємо байт гами.

В програмах Word 97/2000 та Excel 97/2000 дані шифруються за допомогою алгоритма RC4 з ключем довжиною 40 байтів. Таке шифрування вже не дозволяє миттєво знайти пароль. Можливості обчислювальної техніки за останні роки виросли на стільки сильно, що єдиний можливий ключ шифрування документів Word (з 2^{40} можливих) може бути знайдений максимум за чотири доби на комп'ютері з двома процесорами AMD Athlon 2600+.

Починаючи з Office XP, нарешті з'явилась підтримка шифрування документів ключами довжиною більше 40 бітів. Але більшість користувачів до цих пір використовують 40-бітове шифрування, оскільки воно дозволяє відкривати захищені документи у попередніх версіях офісних програм. Та й зміна налаштувань шифрування вимагає додаткових дій з боку користувача (відкриття діалога налаштувань і вибору потрібного криптопровайдера), тоді як за замовчуванням використовуються 40-бітові ключі.

Microsoft Access

Бази даних Microsoft Access можуть мати два типи паролів: паролі на відкриття і паролі для розмежування прав доступу на рівні користувачів.

Пароль на відкриття, як правило, ніколи не був серьйозним захистом, оскільки, починаючи з Access версії 2.0, він зберігався у заголовку бази даних. Правда, сам заголовок був зашифрований алгоритмом RC4, але це не дуже посилювало стійкість, оскільки в рамках однієї весії формата завжди використовувався один і той самий 32-бітовий ключ шифрування, прошитий у данамічно завантажувану бібліотеку, що відповідає за роботу з файлом бази даних. А враховуючи те, що RC4 – синхронний потоковий шифр, достатньо було один раз знайти гаму, що породжується RC4 з відомим ключем, і після цього пароль можна було визначити, виконавши додавання за модулем 2 гами і потрібних байтів заголовку.

Починаючи з Access 2000, звичайне накладання гами вже не дозволяє відразу ж визначити пароль, оскільки необхідно виконати ще декілька додаткових нескладних дій. Але пароль все одно зберігається у заголовку, а отже, може бути звідти прочитаний.

Слід зауважити, що встановлення пароля на відкриття бази даних не приводить до шифрування її вмісту. Однак, Access підтримує таку операцію, як *шифрування бази даних*, але сам пароль у цьому шифруванні ніяк не приймає участі, а ключ шифрування зберігається у заголовку бази.

Інший тип паролів, підтримуваних Microsoft Access, використовується не для забезпечення секретності, а для *розмежування доступу*. Але виявилось, що при проектуванні було допущено декілька помилок, пов'язаних з цими паролями. Правильнішим було б зберігати не самі паролі, а їх хеш-значення. Але через незрозумілі причини в системній базі даних, що містить імена, паролі та інші атрибути всіх користувачів, можна знайти самі паролі, зашифровані потрійним використанням DES з двома ключами

в режимі EDE (Encrypt-Decrypt-Encrypt), коли перший ключ застосовується двічі, на першому і третьому кроці. Ключі зазвичай є константами і зберігаються у динамічно завантажуваній бібліотеці. Такий захист дозволяє швидко визначити пароль будь-якого користувача, хоча Microsoft й стверджує, що втрачені паролі користувачів не можуть бути відновлені.

В системній базі даних для кожного користувача зберігається унікальний ідентифікатор, що є функцією від імені користувача і деякого довільного рядка, що вводиться при створенні облікового запису. Саме цей ідентифікатор і є ключем, за яким ідентифікуються користувачі в основній базі даних.

Так, наприклад, у кожної таблиці в основній базі даних є власник, який має максимальні права. Але в основній базі даних зберігається лише ідентифікатор користувача-власника, а ім'я і вся додаткова інформація для аутентифікації користувача зберігається у системній базі. І створюється враження, що якщо системна база даних буде втрачена, то доступ до вмісту основної бази даних отримати не вдасться. Але функція обчислення ідентифікатора користувача є обертаємою, що дозволяє визначити ім'я власника і рядок, введений при створенні облікового запису. Після цього лишається тільки створити нову системну базу даних і додати в неї користувача з відомими атрибутами, але взагалі без пароля.

6.7 Encrypted File System

Починаючи з Windows 2000 операційні системи, які базуються на ядрі NT, підтримують Encrypted File System (EFS) – розширення файлової системи NTFS (New Technology File System), що дозволяє зберігати файли користувачів у зашифрованому вигляді. При цьому шифрування виконується цілком прозоро і не вимагає від користувача додаткових зусиль, окрім одноразового вказання на те, що файл має бути зашифрований.

Навіть якщо зловмисник зможе отримати фізичний доступ до файлової системи і скопіювати захищений файл, йому не вистачатиме ключа шифрування для доступу до вмісту файла.

Симетричний ключ, яким зашифровується файл (File Encryption Key, FEK), сам зашифрований на відкритому ключі, що належить користувачу, який має право доступу до файла. Ключ зберігається разом із зашифрованим файлом, і для його розширування використовується секретний ключ користувача.

З кожним файлом може бути асоційовано декілька копій ключа, зашифроваих на відкритих ключах так званих агентів відновлення даних (Recovery Agents).

Процедура отримання усієї необхідної для розшифрування інформації включає в себе багато етапів. Але у Windows 2000 реалізація EFS є

такою, що в більшості випадків усі зашифровані файли можуть бути добуті без знання пароля власника або агента відновлення.

Контрольні питання

1. Для чого існують програми-архіватори та пакувальники?
2. Для чого програми даного виду існували на початку свого створення?
3. В чому полягають переваги і недоліки стискання виконуваних файлів під Windows? У якому випадку слід стискати виконувані файли, а у якому – ні?
4. Охарактеризуйте доцільність стискання файлів динамічних бібліотек.
5. В чому полягає принцип роботи програм-архіваторів?
6. Наведіть основні відомі вам програми архівації файлів і коротко охарактеризуйте принцип їх роботи.
7. Що таке програми-зламщики архівів? Які види цих програм ви знаєте? Наведіть приклади.
8. Наведіть відомі програми-архіватори, охарактеризуйте їх і зробіть порівняльну характеристику.
9. Як здійснюється захист документів Microsoft Office?
10. Які особливості захисту файлів в операційних системах, що базуються на ядрі NT?

7 ПРОБЛЕМИ ІДЕНТИФІКАЦІЇ ТА АУТЕНТИФІКАЦІЇ КОРИСТУВАЧА

7.1 Поняття ідентифікації, аутентифікації і авторизації

Ідентифікація покликана кожному користувачу (групі користувачів) зіставити відповідну йому розмежувальну політику доступу на захищуваному об'єкті. Для цього користувач повинен себе ідентифікувати – вказати своє «ім'я» (ідентифікатор). Таким чином перевіряється, чи відноситься користувач, який реєструється, до користувачів, що ідентифікуються системою. Відповідно до введеного ідентифікатора користувачу будуть зіставлені відповідні права доступу.

Аутентифікація призначена для контролю процедури ідентифікації. Для цього користувач повинен ввести секретне слово – пароль. Правильність введеного пароля підтверджує однозначну відповідність між користувачем, що реєструється, і ідентифікованим користувачем.

У загальному випадку ідентифікуються і аутентифікуються не тільки користувачі, але і інші суб'єкти доступу до ресурсів.

Сукупність виконання процедур ідентифікації і аутентифікації прийнято називати *процедурою авторизації*. Зауважимо, що іноді не потрібно ідентифікувати користувача, а достатньо тільки виконання процедури аутентифікації. Наприклад, це відбувається коли треба підтвердити поточного (вже зареєстрованого) користувача при виконанні якихось дій, що вимагають додаткового захисту. У свою чергу, не завжди вимагається здійснювати контроль ідентифікації, тобто в деяких випадках аутентифікація може не здійснюватися. Процедура авторизації має ключове значення при захисті комп'ютерної інформації, оскільки вся розмежувальна політика доступу до ресурсів реалізується щодо ідентифікаторів користувачів. Тобто, увійшовши до системи з чужим ідентифікатором, зловмисник одержує права доступу до ресурсу того користувача, ідентифікатор якого був їм пред'явлений при вході в систему.

Формалізовані вимоги до механізмів захисту

Формалізовані вимоги до даних механізмів захисту такі:

- повинні здійснюватися ідентифікація і перевірка достовірності суб'єктів доступу при вході в систему по ідентифікатору (коду) і паролю умовно постійної дії завдовжки не менше шести буквено-цифрових символів;
- система захисту повинна вимагати від користувачів ідентифікувати себе при запитах на доступ;

- система захисту повинна піддавати перевірці достовірність ідентифікації – здійснювати аутентифікацію. Для цього вона повинна мати в своєму розпорядженні необхідні дані для ідентифікації і аутентифікації;
- система захисту повинна перешкоджати доступу до ресурсів неідентифікованих користувачів і прихованих користувачів, достовірність ідентифікації яких при аутентифікації не підтвердилася;
- система захисту повинна володіти здатністю надійно пов'язувати одержану ідентифікацію зі всіма діями даного користувача.

Основні методи ідентифікації

В сучасних системах захисту від НСД надзвичайно мало уваги приділяється ідентифікації людини, що сидить за комп'ютером. Практично завжди справа обмежується перевіркою легальності екземпляра програми і зовсім не ставиться питання про перевірку легальності користувача. В той же час цілком можлива ситуація, коли це має неабияке значення (наприклад, охорона в банках, мабуть, не повинна мати доступ до конфіденційної інформації навіть на легальних конторських терміналах). З іншої сторони, часто буває зручно не фіксувати комп'ютер, на якому виконується програма, а персоніфікувати її користувача. Безумовно, у людей існує багато індивідуальних рис, а для даної задачі треба вибрати по можливості стабільні особливості, які важко імітувати, але легко перевірити без допомоги спеціальної апаратури (відбитки пальців тут зовсім не підходять).

Існують такі статистичні методи ідентифікації користувача, як:
- ідентифікація за допомогою пароля;
- ідентифікація за клавіатурним почерком;
- ідентифікація по роботі з мишею;
- деякі інші можливості ідентифікації.

7.2 Парольний захист операційних систем

7.2.1 Функціональне призначення механізмів парольного захисту

За функціональним призначенням парольний вхід використовується у таких випадках.

1. *Контроль завантаження системи*. Тобто, може встановлюватися процедура ідентифікації і аутентифікації користувача перед початком завантаження системи, наприклад, вбудованими засобами BIOS. У цьому випадку виконати завантаження системи зможе тільки санкціонований користувач. Доступ до задання режиму завантаження контролюється штатними засобами BIOS, де після аутентифікації користувач може встановити, звідки завантажується система – з жорсткого диска або із зовнішнього носія, а також вказати черговість вибору засобів заванта-

ження. Як контроль доступу до задання режиму завантаження може встановлюватися парольний вхід на можливість завантаження в безпечному режимі. Наприклад, для завантаження в режимі Safe Mode для ОС Windows NT/2000/XP користувачу необхідно пройти авторизацію. Завантажитися в аналогічному безпечному режимі в ОС сімейства UNIX після авторизації може тільки користувач з правами «root».

2. *Контроль функціонування.* Для вирішення задачі контролю функціонування обчислювальної системи виділяють:
   - *контроль користувача при вході в систему* (реалізується зокрема штатними засобами ОС);
   - *контроль під час запуску процесу*. Завдяки цьому під час запуску деяких додатків може бути встановлений парольний захист. Перш за все, тут інтерес представляє встановлення пароля відповідальної особи, наприклад, начальника підрозділу;
   - *контроль під час доступу до локальних ресурсів*. Наприклад, під час доступу до локального принтера і т.д. також може використовуватися аутентифікація відповідальної особи;
   - *контроль під час доступу до мережевих ресурсів*. Реалізується, зокрема, штатними засобами ОС. Наприклад, доступ до ресурсів можна розділити паролем. Так здійснюється мережевий доступ до загальних ресурсів по протоколу NETBIOS для ОС сімейства Windows.
3. *З метою блокування.*

7.2.2 Реалізація механізмів парольного захисту

Введення ідентифікатора і пароля може здійснюватися

- *із застосуванням штатних засобів комп'ютера* – клавіатури, пристроїв введення (наприклад, з дискети),
- *з використанням спеціалізованих пристроїв аутентифікації* – різних апаратних ключів, біометричних пристроїв введення параметрів і т.д.

Природно, що для порівняння інформації, яка вводиться, і еталонної, еталонні облікові дані користувачів повинні десь зберігатися. Можливе *зберігання еталонних облікових даних безпосередньо на об'єкті*, що захищається. Тоді при введенні облікових даних з пам'яті прочитуються еталонні значення і порівнюються з даними, що вводяться.

Крім того, *еталонні дані можуть розташовуватися на сервері*. Тоді еталонні значення на об'єкті, що захищається, не зберігаються, а дані, що вводяться, передаються на сервер, де і порівнюються з еталоном. При цьому саме з серверу дозволяється або забороняється доступ суб'єкту, який ввів облікові дані.

Очевидно, що зберігати еталонний пароль як на захищуваному об'єкті, так і на сервері у відкритому вигляді неприпустимо. Тому для зберігання пароля використовується *необоротне перетворення (хеш-функція),*

що дозволяє створювати деякий образ пароля – пряме перетворення. Цей образ однозначно відповідає паролю, але не дозволяє здійснити зворотне перетворення – з образу відновити пароль. Образи паролів вже можуть зберігатися на об'єкті, що захищається, оскільки їх знання не дозволяє зловмиснику відновити початковий пароль. Для реалізації необоротного перетворення найчастіше на сьогоднішній день використовується алгоритм хешування MD5.

7.2.3 Загрози подолання парольного захисту

Узагальнена класифікація основних загроз парольному захисту представлена на рис.28.

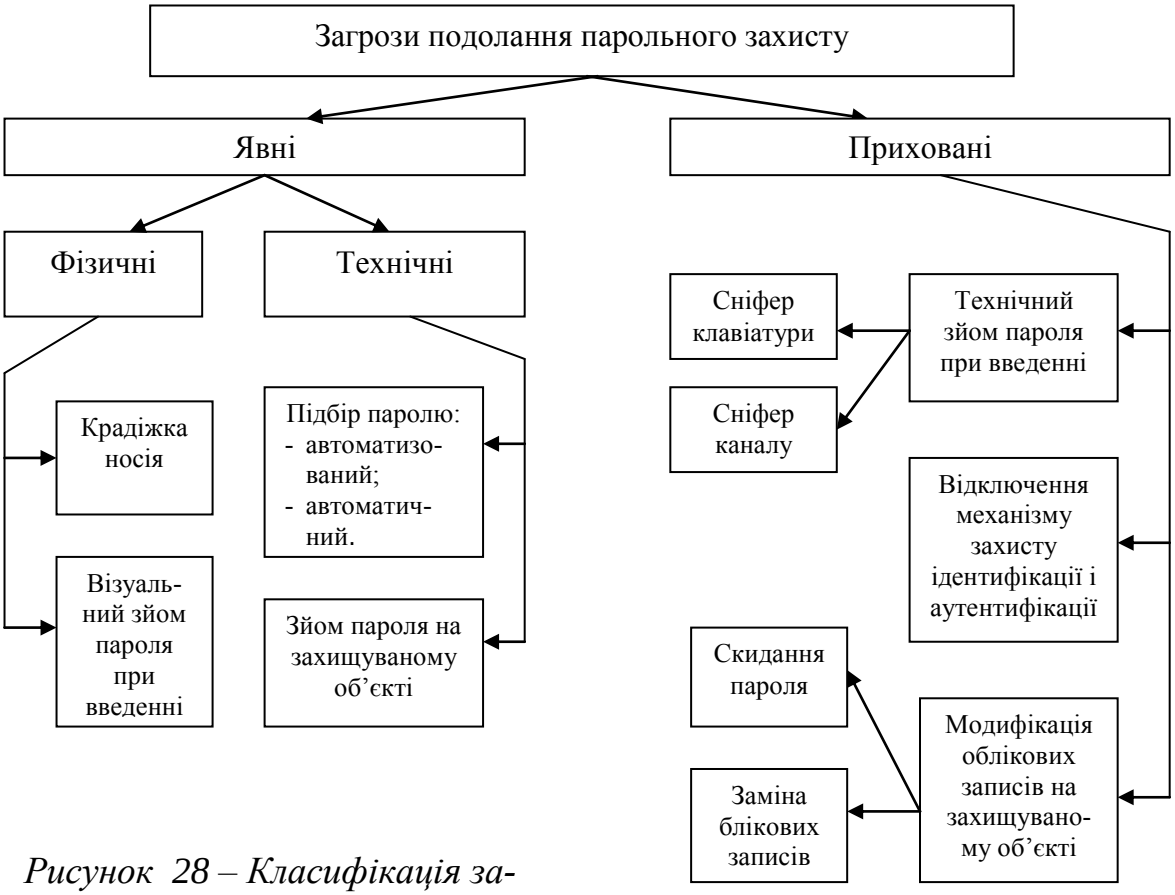


Рисунок 28 – Класифікація загроз парольного захисту

Дана класифікація введена як у відповідності зі статистикою відомих загроз, так і відповідно до потенційно можливих загроз. Крім того, при побудові даної класифікації враховувався аналіз принципів роботи механізмів ідентифікації і аутентифікації.

Розглянемо представлені загрози. Загрози можуть бути явними і прихованими.

Найочевиднішими **явними фізичними загрозами** є:

- *викрадання носія* (наприклад, дискети з паролем, електронного ключа з парольною інформацією і т.д.);
- *візуальне знімання пароля при введенні* (з клавіатури, або з монітора). Крім того, при використовуванні довгих складних паролів користувачі часом записують свій пароль, що також є об'єктом фізичного викрадання.

До **технічних явних загроз** можна віднести *підбір пароля* – або автоматизований (вручну користувачем), або автоматичний, що припускає запуск користувачем спеціальної програми підбору паролів.

Щодо зйому пароля на захищуваному об’єкті, то для порівняння значень введеного пароля і еталонного еталонне значення пароля повинно зберігатися на об'єкті, що захищається (або на сервері в мережі). Це еталонне значення без дотримання відповідних заходів по зберіганню паролів (хешування, розмежування доступу до області пам'яті або реєстру, де зберігаються паролі), може бути викрадено зловмисником.

Природно, що найбільш небезпечними є **приховані загрози**, наприклад, такі:

- *технічне знімання пароля при введенні*. Пароль повинен бути якимось чином введений в систему – з клавіатури, з вбудованого або додаткового пристрою введення, з мережі (по каналу зв'язку) . При цьому зловмисником може бути встановлена відповідна програма, що дозволяє перехоплювати поступаючу на захищуваний об'єкт інформацію. Розвинені подібні програми дозволяють автоматично фільтрувати перехоплювану інформацію за визначеними ознаками – зокрема, з метою виявлення паролів. Прикладом таких програм можуть служити сніфери клавіатури і каналу зв'язку. Наприклад, сніфер клавіатури дозволяє запам'ятовувати всі послідовності натиснень кнопок на клавіатурі (тут пароль вводиться в явному вигляді), а потім фільтрувати події по типах додатків. Зловмисник, встановивши подібну програму і задавши режим її запуску при вході в систему певного користувача, одержить його пароль у відкритому вигляді. Потім, наприклад, троянська програма може видати цей пароль по мережі на іншу робочу станцію. Таким чином, якщо в системі зареєстровано декілька користувачів, то один користувач може взнати пароль іншого користувача, а потім здійснити доступ в систему з правами останнього і т.д.;
- *модифікація механізму парольного захисту*. Цей тип прихованих загроз припускає можливість відключити механізм парольного захисту зловмисником, наприклад, завантажити систему із зовнішнього носія (дисковода або CD-ROM). Якщо механізм парольного захисту є якимсь процесом (у додатковій системі захисту), то виконання даного процесу можна зупинити засобами системного монітора, або монітора додатків, наприклад, засобами, вбудованими в оболонку Far. Подібна можливість існує для ОС Windows 9X/Me;

- *модифікація облікових даних на захищуваному об'єкті*. Ця група прихованих загроз полягає в модифікації облікових даних на об'єкті, що захищається. Це здійснюється або шляхом їх заміни, або шляхом скидання в початковий стан настройок механізму захисту. Прикладом може служити відома програмна атака на BIOS – скидання настройок BIOS в початковий стан за допомогою зміни контрольних сум BIOS.

З усього сказаного може бути зроблений важливий висновок: яким би надійним не був механізм парольного захисту, він сам по собі окремо, без застосування інших механізмів захисту, не може забезпечити високого рівня безпеки захищуваного об'єкта.

Інший висновок полягає у тому, що неможливо порівнювати між собою альтернативні підходи до реалізації механізму захисту (зокрема, механізму парольного захисту), оскільки можна оцінювати лише рівень захищеності, забезпечуваний всією системою захисту в цілому, тобто забезпечуваний сукупністю механізмів захисту (з урахуванням їх реалізації), застосований у системі.

7.2.4 Основні вимоги до паролів

Основні вимоги до паролів і до роботи з ними такі:

- при зчитуванні пароля з клавіатури треба мінімізувати можливість “підглядування” натиснутих клавіш за допомогою резидент них модулів (наприклад, тих, що протоколюють INT 16h – переривання драцверів BIOS клавіатури або INT 9 – переривання від клавіатури);
- захист від “підглядування” одночасно вирішує і ще одну проблему: швидкий перебір паролів за допомогою програм-“черв’яків”, що імітують введення з клавіатури. Але можливість атак все одно треба мати на увазі;
- пароль не повинен бути видимий (і навіть не повинен зберігатися!) в програмі в явному вигляді, а сама процедура перевірки його повинна бути надійно захищена від відлагодження;
- пароль повинен перевірятися повністю, а не частково (мало перевірити тільки суму букв пароля!);
- краще за все перевіряти якусь складну, бажано необоротну функцію пароля, але ця функція не повинна звужувати простір паролів;
- краще не використовувати порівнянь паролів, а використовувати перевірки в тілі програми, про що буде йтися далі.

7.2.5 Парольні зламщики і методи їх роботи

Основним захисним кордоном проти зловмисних атак в КС є система парольного захисту, яка є в усіх сучасних операційних системах. Відповідно встановленої практики перед початком сеансу роботи в ОС користувач повинен зареєструватися, повідомивши свої ім'я та пароль. Ім'я необхідне операційній системі для ідентифікації користувача, а пароль - для підтвердження правильності здійсненої ідентифікації

Існує дві можливості для зловмисника зламати пароль.

Перша можливість полягає в тому, щоб спробувати *підібрати ім'я і пароль*, по черзі перебираючи в голові всі можливі варіанти і вводячи їх з клавіатури. Швидкість такого підпору надзвичайно низька, і, крім того, в системі з добре продуманим парольним захистом кількість повторних введень конкретного імені користувача завжди можна обмежити по кількості і по часу введення.

Інша більш ефективна можливість зламу парольного захисту полягає в тому, що *атаці піддадає системний файл*, що містить інформацію про її легальних користувачів. На допомогу зловмисникам тоді приходять парольні зламними – спеціалізовані програми, які слугують для зламу паролів операційних системи.

Парольні зламними використовують у своїй роботі декілька методів.

1. **Криптографічні алгоритми**, які використовують для шифрування паролів користувачів в сучасних ОС, є досить стійкими для того, щоб відшукати методи їх дешифрації. Тому парольні зламщики просто шифрують всі паролі з використанням того ж самого криптографічного алгоритму, який застосовувався для засекречування в ОС, що атакується, і порівнюють результати такого шифрування з тим, що записано у системному файлі з шифрованими паролями. При цьому в якості варіантів паролів зламщики беруть символьні послідовності, що автоматично генеруються з деякого набору символів. Даний спосіб дозволяє зламати всі паролі, якщо відоме їх представлення у зашифрованому вигляді і вони містять тільки символи з даного набору.

Але кількість комбінацій, які при цьому необхідно перебрати, росте еспоненційно із збільшенням числа символів в початковому наборі, і тому такі атаки парольного захисту можуть займати забагато часу і придатні тоді. коли користувачі ОС не утруднюють себе вибором стійких паролів.

2. **Використання словників,** які являють собою заздалегідь сформовані списки найбільш вживаних слів, що використовуються для паролів. для кожного слова з словника зламщик застосовує одне або декілька правил, у відповідності з якими слово змінюється і породжує додаткову множину паролів, що випробовуються. Враховуючи те, що звичайні словники мов складаються всього з декількох сотень тисяч слів, а швидкість шифрування досить висока, парольні зламними працюють досить швидко.

7.2.6 Парольний захист ОС UNIX та можливості його зламу

В операційній системі Unix інформацію про паролі користувачів можна відшукати у файлі passwd в каталозі etc**.** Ця інформація зберігається там у зашифрованому вигляді і розташовується через "**:**" після імені відповідного користувача. Наприклад,

sasha:5gfkl7lkjhggfcv:12:sasha Spenser:/home/fsg/will:/bin/bash

Тут 5gfkl7lkjhggfcv – це і є інформація про пароль користувача.

При початковому введенні або зміні користувацького паролю операційна система Unix генерує два випадкових байти (в наведеному прикладі "5" і "g"), до яких додаються два байти паролю. Отриманий в результаті байтів рядок шифрується за допомогою спеціальної криптографічної процедури Crypt, де а якості ключа використовується пароль користувача, і в зашифрованому вигляді (fkl7lkjhggfcv) разом із двома випадковими байтами (5f) записується в файл /etc/passwd після імені користувача і двокрапки.

Якщо зловмисник має доступ до парольного файла операційної системи. то він може скопіювати цей файл на свій комп'ютер і потім скористуватись однією з програм для зламу парольного захисту.

Crack – є досить ефективною і популярною програмою цього плану. Хоча вона призначена для запуску під управлінням Unix, процес пошуку паролів, який вона ініціює, може без особливих зусиль розповсюдити між різними платформами, підключеними до єдиної комп'ютерної мережі.

CrackJack - ще одна відома програма для зламу паролів Unix. На жаль, вона працює лише під управлінням DOS, але досить невибаглива щодо ресурсів. Крім того, ця програма не може використовувати одночасно декілька словників. І ще одним "недоліком" цієї програми є те, що вона не може бути запущена під Windows 95/98.

PaceCrack95 – працює під Windows 95/98 як повноцінний DOS-додаток. До того ж він є швидкісним, компактним і досить ефективним.

Q-Crack ma John the Ripper – парольні зламщик, призначені для роботи не тільки на DOS/Windows-платформах, але й на комп'ютерах з операційною системою Linux.

Hades - парольний зламщик, який краще за інші документований, містить ряд корисних утиліт, що дозволяють злити декілька словників в один великий, видаляти із утвореного словника слова, що повторюються, додавати в словник нові паролі, зламані в процесі роботи програми.

Існує багато інших програм зламу паролів операційної системи Unix. Вони стійкі до збоїв (XiT), дозволяють планувати час своєї роботи (Star Cracker), при виконанні монополізують процесор для досягнення максимальної продуктивності (Killer), не тільки зламують паролі ОС Unix, але допомагають подолати парольний захист інших програм, які вимагають щоб користувач реєструвався шляхом введення свого імені і паролю (Cracker Claymore).

Щодо захисту від зламу паролів в операційних система сімейства Unix, то *рекомендується*:

- застосовувати тільки *стійкі паролі*, а в якості стійкості використовувати їх відсутність у словниках. призначених для парольних зламщиків;
- використовувати *затінення (Shadowing),* коли у файлі passwd шифровані паролі користувачів замінюються службовими символами (наприклад, зірочками), а вся парольна інформація зберігається десь у затишному місці. І хоча існують програми спеціально для пошуку прихованих парольних файлів Unix, вони далеко не універсальні і спрацьовують успішно не для всіх ОС сімейства Unix.

7.2.7 Парольний захист Windows 95/98 і його ненадійність

В наш час Windows корпорації Microsoft є найбільш розповсюдженою операційною системою.

Процедура встановлення парольного захисту Windows 95/98 складається з таких кроків:

1. Відкриваємо піктограму “Мой комп’ютер”.
2. Знаходимо піктограму “Панель управления”.

Відкриваємо піктограму “Пароли”. При цьому відображується діалогове вікно, в якому буде задано питання про те, чи хочемо ми одночасно з цим змінити пароль екранної заставки Windows 95/98 (тобто, чи однаковий пароль для відкриття сеансу роботи з Windows 95/98 і для екранної заставки). Це вікно з’являється тільки у тому випадку, якщо вже використовується опція екранної заставки.

3. Знаходимо опцію “Сменить пароль Windows».
4. Вводимо новий пароль в полі “Новый пароль” діалогового вікна “Изменение пароля”. На поле “Старый пароль” не звертати увагу, воно лишається незаповненим. Але якщо пароль вже був введений, і ми хочемо його змінити, то спочатку вводиться старий пароль, а потім новий.
5. В полі “Подтверждение пароля” слід знову ввести новий пароль (для запобігання випадкової помилки) і натиснути “ОК”.

В основі функціонування Windows 95/98 лежать принципи, які використовувались при створенні більш примітивної ОС корпорації Microsoft – DOS. Не зважаючи на те, що Windows 95/98 намагається “приховати” DOS якомога далі, вона і далі використовує стару систему для забезпечення працездатності старих програм, які розроблялись під DOS. А в DOS відсутні захисні механізми, оскільки ця система розроблялась саме для виконання протилежної задачі – надати всім можливість звертатись до будь-яких файлів. Як наслідок, існує декілька можливостей обійти парольний захист Windows 95/98, завантажуючи на комп’ютері операційну систему DOS замість Windows 95/98.

Як правило, обійти парольний захист можна а) або просто при вході в Windows 95/98, зареєструвавшись по-новому, б) або відповівши <Esc> на запит про пароль, в) або завантажившись з завантажувальної дискети, г) або завантажившись у захищеному режимі.

Таким чином, парольний захист Windows 95/98, можна сказати, зовсім ненадійний. Це викликано тим, що корпорація Microsoft офіційно заявила, що призначення паролів Windows 95/98 полягає в наданні можливості по-різному використовувати один і той самий комп'ютер для різних користувачів. І тому, якщо потрібен повноцінний парольний захист, слід звернутись до Windows NT або скористатись додатковими програмними засобами, спеціально призначеними для безпечної роботи з Windows 95/98 (наприклад, пакетом програм Norton Your Eyes Only).

Захист від несанкціонованого завантаження Windows 95/98

Обійти парольний захист Windows 95/98 можна за допомогою *завантажувального диска.* Парольний захист Windows 95/98 активізується тільки при завантаженні Windows 95/98. При використанні завантажувального диска спрацьовує захист не Windows 95/98, а її підсистеми, функціонально еквівалентній DOS, яка не має ніяких засобів забезпечення безпеки роботи з комп'ютером. Потім зловмисник може не тільки встановити власний пароль для наступних сеансів роботи з Windows 95/98, а й зробити так, щоб решта користувачів нічого не помітили і подовжували реєструватись шляхом своїх власних паролів.

Для запобігання несанкціонованого доступу до комп'ютера через завантажувальний диск, необхідно застосувати:

- додаткові фізичні міри захисту (наприклад, закривати двері на замок);
- блокувати клавіатуру або кнопку включення комп'ютера за допомогою ключа,
- використати установки BIOS для запобігання неконтрольованого завантаження ОС з гнучкого диска;
- введення пароля включення живлення та зміни установок BIOS.

Зловмисник може використати можливість завантаження Windows 95/98 *в режимі захисту від збоїв* – захищений режим (Safe Mode). Цей режим призначений запустити ОС, незважаючи ні на які "дрібниці" типа парольного захисту. Для цього достатньо після включення тримати натиснутою клавішу <F5> на клавіатурі.

Крім того, натискаючи при завантаженні на клавіші <Ctrl> та <F8>, можна відразу запустити операційну систему DOS та завантажитись в захищеному режимі.

Через це використання названих клавіш (можливо, і ще деяких) в ході завантаження слід заборонити, заставивши систему ніяк на них не реагувати. Для цього достатньо відредагувати системний файл msdos.sys. Цей файл має приблизно такий вигляд:

```
;FORMAT
[Paths]
WinDir=C:\WINDOWS
WinBootDir=C:\WINDOWS
HostWinBootDrv=C

[Options]
BootGUI=1
DoubleBuffer=1
;
;The following lines are required for compatibility with other programs.
;Do not remove them (MSDOS.SYS needs to be >1024 bytes).
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxa
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxb
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxc
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxd
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxe
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxf
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxg
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxh
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxi
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxj
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxk
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxь
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxn
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxo
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxp
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxq
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxr
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxs
```

Слід додати в розділ [Options] *BootKeys=0* або *BootDelay=0*. Обидві ці команди дозволяють дезактивувати роботу клавіш <F5>,<F8>,<Ctrl> на час завантаження. А для того, щоб зламщик не міг непомітно видалити ці команди, слід захистити його за допомогою антивірусного сканера, який своєчасно попередить про всі спроби внести несанкціоновані зміни в нього.

Запобігання кешуванню паролів в Windows 95/98

Кешування паролів – це метод, який був розроблений корпорацією Microsoft для їх зберігання в Windows 95/98. В цьому випадку пароль користувача розміщується операційною системою в окремий файл на диску в каталог C:\Windows з розширення ".pwl". Пароль записується в цей файл у зашифрованому вигляді, але стійкість цього шифрування не відповідає сучасним вимогам. В глобальній мережі Internet легко віднайти програми, призначені для подолання такого шифрування.

Існує 2 шляхи для розв'язання цієї проблеми.

1. Можна отримати від Microsoft "заплату", яка дозволяє використовувати більш досконалий метод шифрування паролів.
2. Кешування паролів можна відключити. Для цього слід виконати такі дії:

- встановити програму *Редактор системних правил* (System Policy Editor) (вона зазвичай входить в комплект Windows 95/98 Resource Kit на CD) з каталогу E:\Admin\Apptools\Poledit або E:\. Tools\ Reskit\Netadmin\Poledit;
- після установки запустити її на виконання;
- виконати таку послідовність дій: *Создать –> Файл -> Стандартный компьютер –> Сеть –> Пароли -> Отключить кэширование.*
- внесені зміни зберегти на диску у вигляді файлу системних правил, давши йому довільне ім'я і розширення ".pol".

При наступному завантажені Windows 95/98 файл системних правил буде завантажуватись автоматично і визначати роботу комп'ютера..

7.2.6 Парольний захист в Windows NT

Формування, зберігання і використання паролів в Windows NT

Одним з основних компонентів системи безпеки Windows NT є *диспетчер облікових записів* користувачів. Він забезпечує взаємодію інших компонентів системи безпеки, додатків і служб Windows NT з *базою даних облікових записів користувачів* (Security Account Management Database, SAM). Ця база є в наявності на кожному комп'ютері. де встановлено Windows NT. В ній зберігається ін формація, що використовується для аутентифікації користувачів Windows NT при інтерактивному вході в систему і при віддаленому доступі до неї по мережі. Ця база є одним з компонентів системного реєстру і розташована в підкаталозі ...\System32\ Config в файлі з назвою SAM. Інформація в цьому файлі зберігається у двійковому форматі, а доступ до неї здійснюється через диспетчер облікових записів. Змінювати записи цієї бази даних при допомозі програм, що дозволяють напряму здійснювати редагування реєстру Windows NT (Regedit або Regedit32), не рекомендується. За замовчуванням цього робити і не можна, оскільки доступ до бази SAM заборонений для всіх без виключення категорій користувачів операційної системи Windows NT.

Довжина пароля в Windows NT обмежена 14 символами. Це обмеження накладається диспетчером облікових записів, який не дозволяє вводити паролі довжиною більше 14 символів. В базі даних SAM кожний пароль зазвичай представляється у вигляді двох 16-байтових послідовностей, отриманих різними методами.

При використанні першого метода рядок символів користувацького паролю хешується за допомогою функції MD4. В результаті з символьного паролю, введеного користувачем, виходить 16-байтова послідовність – хешований пароль Windows NT. Ця послідовність потім шифрується за допомогою DES-алгоритма (блочний алгоритм шифрування з симетричним ключем у 64 біти, з яких лише 56 безпосередньо використовуються при шифрації, а решта 8 призначена для контролю парності байтів ключа), і ре-

зультат шифрування зберігається в базі SAM. При цьому в якості ключа використовується так названий *відносний ідентифікатор користувача* (Relative Identifier – RID), який являє собою порядковий номер облікового запису даного користувача в базі SAM, який автоматично збільшується.

Для суміщення з іншими програмними продуктами корпорації Microsoft (Windows for Workgroups, Windows 95/98, Lan Manager) в базі SAM зберігається інформація про пароль в стандарті Lan Manager. Для її формування використовується другий метод. Всі літери користувацького паролю приводяться до верхнього регістра, і, якщо пароль має менше 14 символів, доповнюється нулями. З кожної 7-байтової половини перетвореного таким чином пароля окремо формується ключ для шифрування фіксованої 8-байтової послідовності за DES-алгоритмом. Отримані в результаті дві 8-байтові половини хешованого пароля ще раз шифруються за DES-алгоритмом (при цьому в якості ключа використовують RID користувача) і тоді вносять в базу SAM.

Інформація про паролі, занесена в SAM, слугує для аутентифікації користувачів Windows NT. При інтерактивному або мереженому вході в систему користувач вводить пароль, який спочатку хешується і шифрується, далі порівнюється з 16-байтовою послідовністю в базі SAM. При їх співпадінні користувачеві дозволяється вхід в систему.

Можливі атаки на базу даних SAM

За замовчуванням в операційній системі Windows NT доступ до бази даних SAM заблоковано для всіх без виключення її користувачів. Тим не менш його все ж можна скопіювати декількома способами:

- за допомогою програми NTBACKUP будь-який власник права на резервне копіювання файлів і каталогів Windows NT.
- створити резервну копію утилітою REGBAK з Windows NT Resource Kit;
- існує резервна копія файла SAM (SAM.sav) в каталозі *…\Systerm32\Config*;
- існує стиснена архівна копія цього файлу (SAM._) в каталозі *…\Repair*.

При наявності фізичної копії файла SAM видобути з нього інформацію є справою не дуже складною. Завантаживши його в реєстр будь-якого іншого комп’ютера з Windows NT (за допомогою команди Load Hive програми Regedit32), можна вивчити облікові записи користувачів, щоб визначити значення RID користувачів і шифровані варіанти їх хешованих паролів. Знаючи RID і маючи зашифрований хешований пароль, зламщик може отримати мережений доступ до іншого комп’ютера, але для інтерактивного входу в систему цього все одно не достатньо, – треба мати символічне представлення цього пароля.

Існують спеціальні парольні зламщики для відновлення користувацьких паролів операційної системи Windows NT. Вони виконують як прямий підбір паролів, так і пошук за словниками, а також використовують

комбіновані методи зламу парольного захисту, коли в якості словника використовується файл із заздалегідь вирахованими хешованими паролями, що відповідають символьним послідовностям, які часто використовуються в якості паролів. Серед таких програм слід відзначити такі: LophtCrack; ANTExp (Advanced NT Security Explorer, що має досить зручний інтерфейс.

Деякі методи захисту від парольних зламщиків

Одною з головних задач системного адміністратора Windows NT є захист від несанкціонованого доступу тієї інформації, що зберігається в базі даних SAM. Для цього він може передбачити такі дії.

1. *Обмежити фізичний доступ* до комп'ютерів мережі:
   - встановити паролі BIOS на включення комп'ютера і на зміну настройок BIOS;
   - відключити можливість завантаження комп'ютерів з гнучких та компакт-дисків;
   - для забезпечення контролю доступу до файлів і папок Windows NT системний розділ жорсткого диску повинен мати формат NTFS.
2. *Каталог ...\repair засобами операційної системи закрити* для доступу всіх користувачів, включаючи і адміністраторів. Дозволити доступ лише під час роботи утиліти RDISK, що створює в цьому каталозі архівні копії реєстру Windows NT. Слід надійно ховати дискети аварійного відновлення і носії з архівними копіями.
3. Для захисту бази SAM *використати утиліту SYSKEY*, яка дозволяє включити режим додаткового шифрування інформації про паролі. Унікальний 128-бітовий ключ для додаткового шифрування (ключ шифрування паролів – Password Encryption Key, PEK) автоматично зберігається в системному реєстрі для подальшого використання. Перед розміщенням його в системний реєстр ключ PEK шифрується за допомогою іншого 128-бітового ключа, який називається системним ключем (System Key), і може зберігатися в системному реєстрі або у файлі STARTUP.KEY в кореневому каталозі на окремій дискеті. Можна не зберігати його на дискеті, тоді він кожний раз буде розраховуватись за допомогою алгоритму MD5 на основі пароля, введеного з клавіатури в запропоноване утилітою поле.

   Цей спосіб зберігання системного ключа забезпечують максимальний захист паролів в базі SAM. Але це приводить до неможливості автоматичного перевантаження ОС, оскільки для завершення процесу перевантаження необхідно вставити дискету з системних ключем або вручну ввести системний ключ з клавіатури.
4. Для підвищення стійкості паролів рекомендується при допомозі утиліти *"Диспетчер пользователей"* (User Manager)
   - задати мінімальну довжину користувацьких паролів не меншою за 8,

- включити режим “старіння” паролів, щоб користувачі їх періодично оновлювали (чим вища вірогідність атак, тим частіше треба міняти паролі),
- включити режим зберігання певної кількості раніше використовуваних паролів (щоб користувачі не вводили старі свої паролі).

5. Можна скористатись *утилітою PASSPROP* зі складу Windows NT Resource Kit (з ключем /COMPLEX). Це заставить користувачів вводити більш стійкі паролі, які би мали літери в різних регістрах, або сполучення літер і цифр, або літер і спеціальних символів. Для цього ж можна скористатись можливостями пакетів оновлення Windows NT.

7.3 Біометричні методи захисту інформації

Коротко зупинимося на розгляді нових властивостей парольного захисту, реалізовуваних на основі контролю біометричних характеристик користувача. Гіпотетично можлива загроза, пов'язана з тим, що один користувач передає свої парольні дані іншому користувачу, а той скористається ними для несанкціонованого входу в систему останнім (у певному значенні це можна трактувати, як зміна режимів доступу до ресурсів користувача не адміністратором безпеки, що суперечить формальним вимогам до системи захисту).

У загальному випадку механізми біометричної ідентифікації користувача (природно, при їх коректній реалізації) запобігають можливості якої-небудь передачі парольної інформації між користувачами. А це достатньо важливо при реалізації централізованої (без участі користувача) схеми адміністрування механізмів захисту. У цьому і полягає безперечна перевага даних підходів парольного захисту в порівнянні із застосуванням зовнішніх апаратних носіїв парольних даних (всіляких ключів, смарт-карт і т.д.). Іншими словами, коректно в загальному випадку концепція централізованого адміністрування системи захисту може бути реалізована із застосуванням біометричних систем контролю доступу.

Процедури ідентифікації і аутентифікації користувача можуть базуватися не тільки на секретній інформації, якою володіє користувач (пароль, секретний ключ, персональний ідентифікатор і т.п.). Останнім часом все більше поширення набуває біометрична ідентифікація і аутентифікація, що дозволяє упевнено ідентифікувати потенційного користувача шляхом вимірювання фізіологічних параметрів і характеристик людини, особливостей його поведінки.

Основні достоїнства біометричних методів ідентифікації і аутентифікації:

- висока міра достовірності ідентифікації після біометричних ознакам через їх унікальність;
- невіддільність біометричних ознак від дієздатної особи;

- трудність фальсифікації біометричних ознак.

Як біометричні ознаки, які можуть бути використані для ідентифікації потенційного користувача, використовуються:

- узор радужної оболонки і сітківки очей;
- відбитки пальців;
- геометрична форма руки;
- форма і розміри особи;
- особливості голосу;
- біомеханічні характеристики рукописного підпису;
- біомеханічні характеристики "клавіатурного почерку".

При реєстрації користувач повинен продемонструвати один або кілька разів свої характерні біометричні ознаки. Ці ознаки (відомі як справжні) реєструються системою як контрольний "образ" законного користувача. Цей образ користувача зберігається в електронній формі і використовується для перевірки ідентичності кожного, хто видає себе за відповідного законного користувача.

Системи ідентифікації за узором радужної оболонки і сітківки очей можуть бути розділені на два класи:

- ті, що використовують малюнок радужної оболонки ока;
- ті, що використовують малюнок кровоносних судин сітківки ока.

Оскільки вірогідність повторення даних параметрів дорівнює 10-78%, ці системи є найнадійнішими серед всіх біометричних систем. Такі засоби застосовуються, наприклад, в США в зонах військових і оборонних об'єктів.

Системи ідентифікації за відбитками пальців є найпоширенішими. Одна з основних причин широкого розповсюдження таких систем полягає у наявності великих банків даних за відбитками пальців. Основними користувачами таких систем у всьому світі є поліція, різні державні і деякі банківські організації.

Системи ідентифікації за геометричною формою руки використовують сканери форми руки, звичайно встановлювані на стінах. Слід зазначити, що переважна більшість користувачів віддає перевагу системам саме цього типу, а не описані вище.

Системи ідентифікації за обличчям і голосом є найдоступнішими через їх дешевизну, оскільки більшість сучасних комп'ютерів має відео- і аудіозасоби. Системи даного класу широко застосовуються при віддаленій ідентифікації в телекомунікаційних мережах.

Системи ідентифікації за динамікою рукописного підпису враховують інтенсивність кожного зусилля підписується, частотні характеристики написання кожного елементу підпису і зображення підпису в цілому.

Системи ідентифікації за біомеханічними характеристиками клавіатурного почерку грунтуються на тому, що моменти натиснення і відпуску клавіш при наборі тексту на клавіатурі істотно розрізняються у різних ко-

ристувачів. Цей динамічний ритм набору (клавіатурний почерк) дозволяє побудувати достатньо надійні засоби ідентифікації.

Слід зазначити, що застосування біометричних параметрів при ідентифікації суб'єктів доступу автоматизованих систем поки не одержало належного нормативно-правового забезпечення, зокрема у вигляді стандартів. Тому застосування систем біометричної ідентифікації допускається тільки в системах, які обробляють і зберігають персональні дані, що становлять комерційну і службову таємницю.

7.4 Ідентифікація за клавіатурним почерком

Одним з методів боротьби з парольними зламщиками є ідентифікація користувача за почерком під час введення ним імені і пароля.

7.4.1 Графологічні можливості комп’ютера

Рукопис у своєрідності накреслення букв доносить до нас щось особистісне. Графологи, продершись крізь частокіл завитків, багато чого можуть розповісти про їх автора. За допомогою комп'ютера також можна довідатись про людину, що "довбає" по клавішах? Пошукаємо аналогії.

Д.М.Зуєв-Інсаров, автор фундаментальних робіт із графології, не тільки переконливо демонструє методи визначення статі, віку, освіти, роду занять досліджуваного, але і достатню увагу приділяє експериментальним підставам цього наукового напрямку. Його класифікація містить такі формальні ознаки почерку, як: сила натискання, динамічність і напруженість руху, витягнутість, нахил і ступінь зв'язаності букв, напрямок рядка, розташування і змістовність тексту, спосіб тримання знаряддя писання, рівномірність і співрозмірність букв і слів, ритм і виразність писання.

Дещо з цього може запозичити "клавіатуровед", оскільки аналогії є, це безсумнівно. Наприклад, часові інтервали між введенням символів із клавіатури по інформативності нічим не поступаються зв'язаності букв у словах на папері. Дійсно, якщо не новачок при гарному темпі набору раптом почав помилятися, то швидше за все він охоплений внутрішніми переживаннями, що не відносяться до виконуваної роботи. А рівний темп, взятий на початку і збережений до кінця, не гірше графологічної ознаки "рівномірність писання" свідчить про пунктуальність й акуратність людини, яку тестують.

Звичайно, багато особливостей рукописного почерку при роботі на комп'ютері зіставляти даремно, адже клавіатура і драйвери стандартизують написання букв. Зате тут можливий аналіз нових ознак: залежність швидкості введення слів від їхнього змісту, відносний час натискання клавіш різних полів клавіатури й інші. Причому вони в деяких випадках навіть

більш інформативні – наприклад, реакція тестуємого на різні терміни вкаже сферу його інтересів.

Дійсно, хімік швидше набере "водень", "з'єднання", якими він постійно оперує, ніж "програма", "екскаватор". А модельєру будуть звичнішими слова "манекен", "викрійка". (До речі, цю властивість "м'язової" пам'яті, не контрольованої свідомістю, можна використовувати для програми "Де детектор неправди".) У даному випадку з'являється можливість окремо аналізувати ліву і праву півкулі мозку (що відповідають за образне й абстрактне мислення), оскільки вони зв'язані з правою і лівою руками людини, а писання являє собою лише "однобічну" інформацію.

Графологічні дослідження сприяють також діагностуванню хворого, оскільки почерк міняється при захворюванні, повертаючи до нормальному виду в міру видужування. Можливо, подібний аналіз підійде і при постановці попереднього діагнозу працюючого на комп'ютері.

7.4.2 Використання унікальності клавіатурного почерку

Почерк є унікальним, це знають всі. Під час спілкування з компютером індивідуальність користувача на предмет “свій”-“чужий” можна виявити за певними характеристиками:

- швидкість набору;
- звичка використовувати основну чи додаткову частину клавіатури,
- характер "здвоєних" і "строєних" натискань клавіш,
- улюблені прийоми керування комп'ютером, за допомогою яких можна виділити конкретну людини серед усіх працюючих на даній машині.

І нічого дивного, – це те саме, що здатність меломанів розрізняти на слух піаністів, що виконують один якийсь музичний твір.

Для того, щоб виявити індивідуальні особливості клавіатурного почерку, так само, як і при графологічній експертизі, потрібні *еталонний і досліджуваний* зразки тексту. Краще, якщо їхній зміст буде однаковим (так звана парольна чи ключова фраза). Зрозуміло, що за двома-трьома, навіть за десятьма натиснутими клавішами відрізнити користувача неможливо, - потрібна статистика.

При наборі ключової фрази комп'ютер дозволяє зафіксувати багато різних параметрів, але для ідентифікації найбільше зручно використовувати час, витрачений на введення окремих букв. А повторивши введення фрази кілька разів, в результаті матимемо безліч тимчасових інтервалів для кожного символу. На базі отриманих значень завжди можна розрахувати середній час введення кожного символу, припустиме відхилення від середнього, і зберігати ці результаті в якості еталонів для кожного користувача.

Унікальні особливості клавіатурного почерку виявляються двома методами:

- за набором ключової фрази;
- за "довільним" текстом.

Обидві методики розрізняються лише вибором парольної фрази. У першому випадку це завжди одне й те саме, а в другому – нарізноманітніший текст, що має свої переваги, оскільки дозволяє одержувати ті самі характеристики непомітно, не акцентуючи увагу на парольній фразі.

Втім, на вибір схеми перевірки впливає тематика захисту ПЗ. Припустимо, деякий власник фірми, бажаючи довідатися про поточний фінансовий оборот, запустив програму бухгалтерського обліку, а комп'ютер, замість короткої довідки з комерційною і тому секретною інформацією, пропонує набрати 2-3 сторінки "вільного тексту", щоб переконатися, що перед ним дійсно чи директор головбух. Тобто, тут краще застосувати метод "парольної фрази". З іншого боку, особа, що має допуск до секретів, може працювати з такою програмою цілий день, час від часу відволікаючись від комп'ютера, а щоб у цей момент зловмисники не скористалися розкритою системою, бажано періодично проводити "негласну перевірку" (прохання перенабрати "пароль" щопівгодини буде занадто настирливим).

Правильній ідентифікації допомагає також *рисунок почерку*. Під цим розуміється ряд значень, що являють собою різницю між сусідніми часовими інтервалами - свого роду "похідна" по почерку, що показує відносні уповільнення або прискорення при роботі з клавіатурою. Характеристика ця досить індивідуальна, що підтверджується рядом експериментів.

Кожен з методів обов'язково має режими:

- настроювання;
- ідентифікацію.

При настроюванні визначаються і запам'ятовуються еталонні характеристики введення користувачем ключових фраз, наприклад, час, витрачений на окремі букви. А в режимі ідентифікації, після виключення грубих помилок, еталонна й отримана множини зіставляються (перевіряється гіпотеза про рівність їхніх центрів розподілу).

7.4.3 Режим настроювання

Для визначення еталонних характеристик користувача необхідно вибрати ключову фразу. Бажано, щоб букви були рівномірно розподілені по клавіатурі, наприклад: "Увага – ідентифікація користувача за клавіатурним почерком". Потім разів з десять набрати її на клавіатурі, визначити час, витрачений на введення кожної букви, і виключити грубі помилки (ті значення, що різко виділяються з кожної десятки наявних). Розрахувати і запам'ятати величини математичного сподівання (M_e), дисперсії (S_e) і число спостережень (n_e). Ці значення і називаються еталонними.

7.4.4 Режим ідентифікації

Ідентифікація за набором ключової фрази. Визначення математичних параметрів (M_i, S_i, n_i) користувача при його ідентифікації проводиться аналогічно, як і в режимі настроювання. Єдина відмінність – множини для кожного символа будуть складатися з меншої кількості значень (якщо фразу набирають кілька разів) чи навіть з одиничних величин (при однократному наборі). Потім порівнюються дисперсії двох множин (еталонної і щойно розрахованої) і величини математичного сподівання – рівні чи збігаються центри розподілу цих двох сукупностей. Зрозуміло, повної рівності не буде, тому алгоритм закінчується оцінкою імовірності того, що користувач – той самий (якщо вона більше 50%, то всі невідповідності можна віднести за рахунок випадкових факторів).

Ідентифікація за "довільним" текстом. На відміну від першого методу, тут одержуваний ряд значень сильно відрізняється від еталона (будь-який символ "ключа" навіть якщо і зустрінеться, то виявиться не на "своєму" місці). Тому при складанні множин у якості базисних використовуються величини, які можна підібрати як у ключовій, так і у випадковій фразах, наприклад, - час між натисканням двох клавіш в однакових сполученнях (якщо слово еталона "Увага", то в вільному тексті шукаємо "ув", "ва", "аг" і т.д. і визначаємо розмір паузи, що пройшла з моменту натискання "У" до натискання "н"), вважаючи, що користувач буде переносити руку від однієї клавіші до інший однаково в обох випадках (і при настроюванні, і при ідентифікації).

А порівняння математичного сподівання і дисперсії з еталонними таке саме, як і раніше (якщо базисні величини двох множин обрані правильно, то вони добре корелюють), але спершу необхідно виключити грубі помилки, яких у даному випадку буде більше.

Приведені методики досить прості і спираються на відомі розділи математичної статистики, у різних варіаціях вони використовуються в багатьох системах. Зрозуміло, можна скористатися дисперсійним, регресійним і іншим видами аналізу й ускладнити рішення, довівши їх до досконалості. Але це ускладнить і життя користувача, адже важко щоразу перед початком роботи вводити солідні шматки парольних текстів.

Цілком природно, що з часом характеристики користувача змінюються. Тому рекомендується після кожної успішної ідентифікації коректувати еталони за формулою $M_i=(n*M_e+X)/(n+1)$, де M_i, M_e – характеристики виправленої й еталонної множин, X – величина, отримана в ході ідентифікації, n – кількість дослідів, що ввійшли в еталонну множину.

7.5 Інші способи ідентифікації

Люди по різному сприймають події, що відбуваються. Якщо запропонувати за короткий час прикинути кількість крапок чи голосних букв у довгих фразах, розміри горизонтальних і вертикальних ліній, то відповіді будуть різними, – скільки випробувань, стільки і думок. Ці особливості людської психіки також підходять для ідентифікації. Правда, у залежності від стану і самопочуття людини отримані значення будуть "плавати", тому в практиці розумніше покластися на інтегральний підхід, коли підсумок підводиться по декільком перевіркам, враховуючи і роботу з клавіатурою.

Результуючий тест міг би бути таким: на екрані, на кілька секунд, з'являються вертикальні лінії. Їхній розмір і кількість випадкові. Користувач набирає відповідні, на його погляд, цифри. Таким чином, з'ясовуємо: характеристики клавіатурного почерку, оцінюємо пам'ять (наскільки зазначені довжина і число ліній близькі до дійсності), увага і точність підрахунку (наскільки довжина однієї лінії правильно зіставлено із сусідньою). Порівнюємо результати з еталоном.

У цьому методі не так важливі помилки у визначенні розмірів, головне – щоб вони повторювалися і при настроюванні, і при ідентифікації.

Крім того, при ідентифікації почерку можна враховувати і звички у використанні основних чи допоміжних частин клавіатури, "улюблені" комбінації клавіш при альтернативних варіантах, виконання "здвоєних" і "строєних" натискань однією чи двома руками і т.д., за розписом з використанням мишки, за допомогою психологічних тестів і паролів, тощо.

7.6 Клавіатурні шпигуни

Мова йтиметься про так званих клавіатурних шпигунів (keyboard loggers). За допомогою цих маленьких програм можна довідатися, що робили на нашому комп'ютері, поки нас не було в офісі або вдома. Якщо ж зуміти підкласти їх на чужий комп'ютер, то можна одержати можливість дізнаватися практично про всі дії хазяїна комп'ютера.

Метод роботи таких програм дуже нагадує спосіб з якоїсь дитячої передачі про приватних детективів: під скатертину на столі кладеться лист звичайного папера, а на нього лист копірки. Тепер усе, що буде написано за цим столом, через копірку друкується на листі папера. Тут головне - непомітно підкласти копірку під скатертину, а потім також непомітно витягти результат. У нашому випадку зробити це звичайно не складає труднощів, оскільки програми ці дуже маленькі (звичайно не більше 100 кілобайт разом з описом і help'ом, але зустрічаються виключення, що дотягають майже до 500) і скопіювати їх з дискети на “вінт” “клієнта”, а також періодично (можна раз у кілька днів, але чим довше, тим сутужніше потім розі-

братися) “знімати” результати. Це хвилинна справа, досить лише на якийсь час залишитися наодинці з комп'ютером.

Spy

Програма SPY (Security Log System), написана Алексом Леменковым, призначена для запису в спеціальний текстовий Log-файл інформації про всі здійснювані на комп'ютері дії з автоматичною реєстрацією дати і часу. Запис відбувається непомітно для користувача, тобто людина, що працює на комп'ютері, швидше за все ніколи не довідається про те, що за його діями може спостерігати сторонній. Перша версія була датована в документації 1992 роком, працювала вона під MS-DOS версії 4.0 та старших на будь-якому комп'ютері, починаючи з “двійки”. SPY володіє декількома цікавими властивостями: програма завантажується в пам'ять комп'ютера по типу вірусу, маскується під DOS і не видна таким програмам як MEM, RELEASE і т.д. Крім завантаження з командного рядка або файлу autoexec. bat, SPY може запускатися з будь-якого файлу, що виконується (“.com” або “.exe”). Для того, щоб реалізувати цей метод, до складу пакета, крім власне SPY, входить програма SPYEXE, запустивши яку, автоматично приписується до обраного виконуваного файлу команда запуску SPY. Таким чином, змусивши SPY запускатися, приміром, разом із драйвером миші, не залишиться ніяких слідів в “autoexec.bat”. Тепер, якщо надійно сховати Log-файл, у який будуть виводитися результати стеження, і дати йому якенебудь непримітне ім'я, типа “mouse.log” (або зробили його невидимим), то виявити, що за усіма діями користувача спостерігають, навряд чи можна. Хіба що, “риючись” у своєму вінчестері, випадково “набрести” на дуже дивний текстовий файл з такою ж дивною інформацією.

Keylog 95

Це ще одна з програм–клавіатурних шпигунів. Призначена вона спеціально для Windows 95. Автор цієї “подглядалки” – Майк Элліс. Повний каталог програми займає приблизно 466 Кбайтів. Для запуску програми потрібно додати іконку програми в папку Автозавантаження (Startup) стартового меню Windows 95, що, як правило, далеко не самий елегантний варіант. Після завантаження Keylog на Панелі задач з'являється невеликий прямокутник, що зливається з нею по кольорі і не має ні значка, ні надпису. Натискання на цей прямокутник приводить до появи на екрані невеликого вікна з написом Minimize this window. Імовірно, це досить переконливо для недосвідченого користувача, але більш уїдлива людина обов'язково захоче довідатися, що ж це за незвичайного виду прямокутник і дивне віконце. Механізм установки Keylog 95 не дуже зручний: перед запуском програми необхідно вручну скопіювати два .dll-файли з установочного комплекту програми в директорію c:\windows\system. Запущена програма реєструє всю інформацію, що вводиться з клавіатури в текстових файлах, привласнюючи їм імена начебто 11300901.99W, де перші чотири цифри –

час створення файлу (у нашому випадку 11:30), два наступні – місяць, число і дві цифри в розширенні – рік. Істотним недоліком програми є те, що вона не дозволяє довільно вибирати директорію для збереження Log-файлів, розташовуючи їх строго по адресі c:\dos\logx. У комплект програми входить невеликий, але цілком виразний опис англійською мовою.

HookDump

HookDump 2.8, написана Іллею Осиповим, – це досить вдала програма цього класу. Вона призначена для роботи під Windows 3.1 чи 95, але без проблем запускається і працює під 98-ми *"Вікнами"*. Програма має широкий набір функцій і гнучку систему настроювання. Інтерфейс HookDump досить простий і дозволяє вказати, відзначивши відповідні пункти меню, натискання яких клавіш реєструвати (можливий вибір: буквено-цифрові клавіші + клавіші керування чи курсором же реєстрація всіх натискань, включаючи Caps Lock, Shift, Tab, всі функціональні клавіші і т.п.), яку інформацію про працюючі програми запам'ятовувати (можлива реєстрація часу, активних вікон і т.п.). Крім тексту, що набирається з клавіатури, у Log-файлі записується навіть така інформація, як схований пароль Dial-Up Networking, що взагалі не набирався. Можлива також реєстрація натискань на кнопки миші (незрозуміло, щоправда, як же потім визначити, у яких місцях і на що "клікали" мишею). Щоб встановити програму, потрібно лише цілком скопіювати її каталог (розмір каталогу ледве більше 50 кілобайт) на вінчестер. Для автоматичного запуску HookDump досить відзначити в меню Startup рядок AutoStartUp (тобто немає необхідності включати програму в папку Автозавантаження). Після цього програма буде автоматично запускатися разом з Windows, ніяк не виявляючи при цьому своєї присутності. Кінцевий текстовий файл із розширенням ".hk" може знаходитися в будь-якому каталозі на вибір. Для цього потрібно вказати у файлі "hookdump.int" бажаний каталог і ім'я файлу, що будуть створені програмою. Крім того, є досить пристойний Help (англійський). Програма HookDump проста в установці (вистачає і півхвилини), має масу настроювань, а, будучи запущеною, залишається зовсім непомітною.

Secret Agent

Існує ще ghjuhfvf SECRET AGENT Олексія Черненка (ця програма працює на будь-якому "тостері" з 8086 процесором) і багато інших.

Для того, щоб уберегтися від подібних шпигунських закладок, потрібно дотримуватись банальних правил, відомих усім: стежити за тим, щоб нашим комп'ютером не користувалися під час відсутності і "уникати випадкових зв'язків".

7.7 Боротьба зі spyware в Windows

Всім давно відомо, що на будь-яку хитрість завжди знайдеться засіб, що дозволяє цю саму хитрість знайти, і там, де діє розвідка, завжди є і контррозвідка. Тому люди, які думають, що на їх комп'ютері встановлений клавіатурний шпигун, що стежить за всіма їхніми діями, можуть використовувати відповідну програму для їх виявлення. Ці програми працюють за принципом антивірусів, виявляючи шпигунські програми за допомогою евристичного аналізу. Вони перевіряють, які процеси в даний момент виконуються в системі, і якщо знаходять, що якийсь з них стежить за клавіатурою, моментально сповіщають про це користувача.

Сьогодні проблема шкідливих «шпигунських» програм (spyware), які так легко «підчепити» в Інтернеті, стоїть гостріше нікуди. І, на жаль, поки немає причин чекати, що вона зникне в майбутньому. Популярні і безкоштовні програми Lavasoft AdAware і Spy-Bot Search and Destroy (www.safer-networking.org/en/index.html) добре справляються з видаленням шкідливих програм, що вже прокралися на комп'ютер, але не дуже допомагають запобігти їх вторгненню. Особливо це стосується безкоштовних версій. На жаль, велика частина активних антишпигунських програм коштує дуже дорого. Деякі користувачі радять перейти на альтернативні браузери на зразок Firefox або Opera, проте і там існують проблеми з безпекою. Якщо який-небудь браузер починає набирати популярність, то і привабливість створення spyware-програм під нього теж збільшується. Таким чином, перехід на альтернативний браузер – це тимчасове і часто незручне рішення. Проте сьогодні можна знайти абсолютно безкоштовні утиліти, які рівні або навіть перевершують дорогі антишпигунські програми.

SpywareBlaster

Одна з подібних безкоштовних утиліт (безкоштовна для персонального використання і для освітніх установ) – SpywareBlaster (інтернетівська адреса – www.javacool-software.com/spywareblaster.html) використовує оригінальний підхід захисту від spyware (рис.29). Після запуску вона додає в розділ заборонених сайтів браузера Internet Explorer (restricted sites) список відомих сайтів, що поширюють шкідливі програми. Також забороняються деякі функції браузера, на зразок ActiveX і Java, які дозволяють встановлювати

Рисунок 29 – Вигляд головного вікна програми SpywareBlaster

шкідливі програми у фоновому режимі, часто непомітно для користувача.

SpywareBlaster перевіряє настройки безпеки Internet Explorer і при необхідності пропонує встановити захищеніший режим, щоб шкідливі додатки не могли встановлюватися без відома користувача. Ще одна функція SpywareBlaster полягає в блокуванні настройок домашньої сторінки, які шкідлива програма може змінити для злому браузера.

Як додаткову опцію безпеки SpywareBlaster пропонує зашифроване сховище файлу HOSTS, вміст якого може бути змінений в ході деяких атак. Якщо користувач побажає відновити файл HOSTS, SpywareBlaster дозволить вибрати будь-яку копію із збережених і відсортованих по даті. Функція ця дуже близька до відновлення системи (System Restore) в Windows ME/XP за винятком того, що користувач здійснює власне резервування. Для більшого захисту ПК SpywareBlaster пропонує функцію створення образу системи System Snapshot. В образ записуються параметри браузера. Якщо в його настройках відбулися які-небудь небажані зміни або він поводиться неадекватно, настройки браузера і виходу в Інтернет можна повернути назад – до того моменту, коли створювався образ.

Також програма підтримує функцію Flash Killer, яка буде корисна при відвідинах сайту, що використовує Macromedia Flash для відображення реклами. За допомогою Flash Killer можна легко заборонити флеш, після чого знов включити підтримку цієї технології при переході на інший сайт.

Як у разі будь-якого антивірусного і антишпигунського програмного забезпечення, для забезпечення максимально високого захисту ПК важливо мати останню версію оновлень. SpywareBlaster обновляється вручну. Цю операцію рекомендується виконувати не рідше, ніж раз в два тижні.

Втім, автоматичне оновлення все-таки є. Але вже не безкоштовно – за 9,95 дол. AutoUpdate автоматично обновлятиме програму протягом року. Проте, як вже сказано вище, програма чудово працює і без цієї функції.

Важливими перевагами Spyware Blaster є малий розмір (менше 2,5 Мбайтів) і те, що на відміну від інших антишпигунських утиліт, її не вимагається постійно тримати у фоновому режимі, тому вона не споживає додаткові системні ресурси. Програма чудово працює під Windows 95/98/2000/ME/NT/XP і захищає браузери Mozilla/Netscape/Firefox/AOL, а також, звичайно, і Internet Explorer.

IE-SPYAD

Утиліта IE-SPYAD (інтернетівська адреса – netfiles.uiuc.edu/ehowes/www/resource.htm) не така функціональна і дружня до користувача, як SpywareBlaster, проте і вона досить корисна. Програма працює з реєстром і додає великий список відомих сайтів і доменів, пов'язаних з розповсюдженням шкідливих проблем, некоректною рекламою і т.д. Захист забезпечується тільки від сайтів, внесених в «чорний» список, тому утиліту дуже важливо постійно оновляти.

IE-SPYAD працює тільки з версіями Internet Explorer 6.0 (включаючи SP), 5.5 (включаючи SP1 і SP2), 5.01 (включаючи SP1 і SP2), 5.0, 4.01 (включаючи SP1 і SP2), 4.0, а також з браузерамі AOL (що використовують вбудований Internet Explorer). Якщо застосовується інший браузер, то IE-SPYAD буде абсолютно зайвою.

Microsoft Antispyware

Для активного захисту від шкідливих програм у реальному часі дуже хороша безкоштовна утиліта Microsoft Antispyware. Вона працює тільки під Windows XP/2000 і Windows Server 2003, але працює дуже ефективно! На даному етапі доступна бета-версія утиліти, проте вона цілком стабільна і проста у використанні. Крім того, утиліту легко оновляти – навіть зручніше, ніж інші антишпигунські програми, включаючи платні.

На лютневій конференції RSA 2005 Біл Гейтс оголосив, що призначена для користувача версія програми під Windows продовжить залишатися безкоштовною для користувачів ліцензійній версії цієї ОС.

WinSock XP FIX

Проте, після інфікування і видалення шкідливих програм інтернет-з'єднання може працювати некоректно. Існують безкоштовні утиліти, які повністю відновлюють WinSock і підключення до Інтернету. Для Windows XP можна рекомендувати Winsock XP Fix (адреса: www.spychecker.com/program/winsockxpfix.html), для Windows 98/ME – Winsock2 Fix (www.spychecker.com/program/winsockxpfix.html).

Запуск цих утиліт приводить до відновлення працездатності мережі. Проте, якщо використовувалися статичні IP-адреси, то їх доведеться ввести повторно. Якщо ж адреса привласнюється автоматично через DHCP-сервер, то жодних проблем не виникне. Якщо потрібен більший контроль над специфічними функціями WinSock, то варто використовувати утиліту LSP-Fix. Слід, проте, розуміти, що для роботи з нею потрібно певний рівень технічних знань і відповідних навиків.

Рисунок 30 – Вигляд вікна програми Anti-keylogger після запуска

Anti-keylogger

Розробники цієї програми (рис. 30), які, до речі, створили і PC Acme Pro, стверджують, що вона працює за допомогою хитрих математичних алгоритмів і в ній відсутня будь-яка база даних. При скануванні програма

не могла знайти тільки SpyAgent, PC AcmePro і Perfect Keylogger. Інші ж шпигуни були виявлені моментально. Програма має відмінний і зрозумілий інтерфейс. Але є в неї і недоліки: при роботі програми комп'ютер, особливо слабкий, починає сильно гальмувати, і деякі додатки зависають.

HiJackThis

Якщо користувач добре знайомий з внутрішньою роботою Windows, то при видаленні шкідливої програми можна використовувати утиліту HiJackThis (www.spywareinfo.com/~merijn/downloads.html), що допомагає знайти і знищити шкідливі компоненти. Файл журналу HiJackThis можна відправити до аналізатора онлайна HiJackThis (www.hijackthis.de). Втім, цілком покладатися на нього не слід, реально допомогти він може лише укупі з власними технічними знаннями користувача.

Keylogger Killer

Ця невелика програма "важить" всього 52 Кбайти, але зі своєю задачею справляється досить успішно. Вона легко виявляє всі програми з наведеного вище переліку, крім PC AcmePro. Але тут не обійшлося і без курйозів: у розряд клавіатурних шпигунів була зарахована і цілком нешкідлива програма – автоматичний перемикач розкладок (не дивно – адже працює він за тим же самим принципом, що і клавіатурні шпигуні: відслідковує всі натискання на кнопки). Є функції знешкодження і відновлення шпигунських модулів, що викликаються натисканням правою кнопкою мишки.

Контрольні питання

1. Дайте поняття ідентифікації і аутентифікації в комп'ютерних системах.
2. Наведіть основні вимоги до механізмів захисту в КС.
3. Назвіть основні методи ідентифікації користувачів в КС.
4. Які функції виконує введення парольного захисту в ОС?
5. Які ви знаєте методи реалізації механізмів парольного захисту?
6. Класифікуйте основні загрози парольному захисту.
7. Наведіть основні вимогами до паролів і роботи з ними.
8. Які методи використовують парольні зламники у своїй роботі?
9. В чому особливості парольного захисту ОС UNIX?
10. Наведіть причини ненадійності парольного захисту Windows 95/98 і які міри можна прийняти для його покращення?
11. Як організовано парольний захист у Windows NT?
12. Охарактеризуйте біометричні методи захисту інформації.
13. В чому сутність ідентифікація за клавіатурним почерком?
14. Які ще способи ідентифікації користувача ви можете навести?
15. Охарактеризуйте відомі вам програми-клавіатурні шпигуни.
16. Назвіть і дайте характеристику поширеним програмам протидії шпигунству.

ЛІТЕРАТУРА

1. Соколов А.В., Степанюк О.М. Методы информационной защиты объектов и компьютерных сетей (Серия «Шпионские штучки»). – СПб.: Полигон, 2000. – 272 с.
2. Соколов А.В., Степанюк О.М. Защита от компьютерного терроризма. Справочное пособие. – СПб.: БХВ-Петербург, Арлит, 2002. – 496 с.
3. Щеглов А.Ю. Защита компьютерной информации от несанкционированного доступа. – СПб.: Наука и техника, 2004. – 384 с.
4. Хипсон П. Эффективная работа: Реестр Windows XP. – СПб: Питер, 2005. – 480 с.
5. Вильям Столлингс. Операционные системы. – Москва-Санкт-Петербург-Киев: Вильямс, 2002. – 845 с.
6. Глушаков С.В., Хачиров Т.С., Соболев Р.О. Секреты хакеров. Защита и атака. – Харьков: Фолио, 2004. – 414 с.
7. Анин Б. Защита компьютерной информации. – С.-П.: BHV,2000.–370 с.
8. Вильям Столлингс. Операционные системы. – Москва-Санкт-Петербург-Киев: «Вильямс», 2002. – 845 с.
9. Д.Иртегов. Введение в операционные системы. – С.-П.: BHV, 2002. – 624 с.
10. Романец Ю.В., Тимофеев П.А., Шаньгин В.Ф. Защита информации в компьютерных системах и сетях. – М.: «Радио и связь», 2001. – 376 с.
11. Михаэль Бэнкс. Информационная защита ПК. Пер. с англ. – Киев: Век, 2001. – 272 с.
12. М.М.Коваленко. Комп’ютерні віруси і захист інформації. Навчальний посібник. – Київ: “Наукова думка”, 1999. – 268 с.
13. Домарев В.В. Безопасность информационных технологий. Методология создания систем защиты. – М.: «Диасофт».
14. Лукацкий. Обнаружение атак. - С.-П.: BHV, 2000.
15. Середа С.А. Оценка эффективности систем защиты программного обеспечения. – Материалы Международной конференции IPSIT'99, 1999.
16. Середа С.А. Программно-аппаратные системы защиты программного обеспечения. – Материалы Международной конференции аспирантов при Экономической Академии Республики Молдова, 1999.