МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Р. Е. Пащенко

ЗАХИСТ ПРОСТОРОВО-РОЗПОДІЛЕНИХ ДАНИХ У КОМП'ЮТЕРНИХ СИСТЕМАХ

Конспект лекцій

Харків «ХАІ» 2020

УДК 004.056.5
П12

Рецензенти: д-р фіз.-мат. наук, с.н.с. В. К. Іванов,
д-р техн. наук, доц. В. А. Таршин

Пащенко, Р. Е.

П12 Захист просторово-розподілених даних у комп'ютерних системах [Текст] : консп. лекцій / Р. Е. Пащенко. – Харків : Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2020. – 104 с.

ISBN 978-966-662-758-5

Розглянуто відомості щодо нормативно-правового забезпечення захисту інформації і рівнів забезпечення інформаційної безпеки. Наведено класифікацію шкідливих програм, дані щодо їх впровадження у комп'ютерні системи та їх шкідливу дію на просторово-розподілені дані. Розглянуто методи виявлення шкідливих програм. Викладено основні проблеми, пов'язані з безпекою в мережі Інтернет, та основні компоненти і приклади брандмауерів. Подано дані щодо можливості тайної передачі інформації і методів криптографічного захисту просторово-розподілених даних у комп'ютерних системах, моделі цифрового підпису.

Для студентів, що навчаються за спеціальностями 193 «Геодезія та землеустрій» спеціалізації «Геоінформаційні системи і технології» та 103 «Науки про Землю» спеціалізації «Космічний моніторинг Землі».

Іл. 22. Табл. 4. Бібліогр. : 12 назв

УДК 004.056.5

ISBN 978-966-662-758-5

ЗМІСТ

ПЕРЕДМОВА

Останніми роками в результаті інтенсивного розвитку інформаційно-обчислювальних мереж відбулася інтеграція засобів обчислювальної і телекомунікаційної техніки. Це зумовило необхідність проведення єдиної політики із захисту інформації як в державних органах, так і в приватному секторі. Проблема забезпечення інформаційної безпеки на всіх рівнях може бути вирішена успішно, якщо буде створена і буде функціонувати комплексна система захисту інформації. Захист даних має здійснюватися на всіх етапах їх оброблення від збору, зберігання, перетворення до передачі інформації. Широке застосування хмарних обчислень, засобів віддаленого підключення з мобільних і віддалених стаціонарних пристроїв через мережі загального призначення приводять до значного ускладнення їх захисту. Фактично, для будь-якого повідомлення блоку даних або програмному коду потрібно забезпечити запобігання їх несанкціонованій модифікації і несанкціонованому ознайомленню, що досягається застосуванням криптографічного захисту. Забезпечення безпеки просторово-розподілених даних в інформаційно-телекомунікаційних системах стало одним з пріоритетних завдань у сучасному світі.

Під час організації захисту просторово-розподілених даних на державних і приватних підприємствах спочатку визначаються потенційні загрози та оцінюються можливі збитки стосовно кожної з них. На основі цього формується політика безпеки та впроваджуються відповідні організаційні заходи. Всі ці заходи мають відображення у нормативно-правових документах підприємств. Для реалізації цих заходів на підприємствах створюється спеціальний підрозділ з інформаційної безпеки, який, як заведено, працює за декількома напрямками. По-перше, захищає конфіденційні дані співробітників, а також таємну і конфіденційну інформацію підприємства. По-друге, запобігає несанкціонованому проникненню до комп’ютерних систем користувачів. По-третє, забезпечує цілісність інформації, з якою працюють на підприємстві. Для забезпечення захисту інформації на підприємстві можуть використовуватися електронні підписи, криптографічні способи шифрування, паролі, система аудиту і протоколювання, електронні ключі і т. д. Фахівці з інформаційної безпеки працюють над створенням строгої наукової основи методів і способів захисту інформації.

Конспект складається з дванадцяти лекцій, які охоплюють основний теоретичний матеріал навчальної дисципліни «Захист просторово-розподілених даних у комп'ютерних системах». Під час підготовки матеріалів лекцій автор користувався джерелами інформації, які наведені в кінці конспекту. Необхідно зазначити, що під час написання лекції № 1 були використані роботи [1, 4 – 6], лекції № 2 – роботи [2 – 4, 9], лекцій № 3 – 5 – роботи [4, 7, 8], лекцій № 6, 7 – роботи [4, 9], а лекцій № 8 – 12 – роботи [9 – 12].

Лекція № 1

АКТУАЛЬНІСТЬ ПРОБЛЕМИ ЗАХИСТУ ІНФОРМАЦІЇ

Навчальні цілі:

- розглянути актуальність проблеми захисту інформації;
- вивчити рівні забезпечення інформаційної безпеки.

Навчальні питання:

1. Актуальність проблеми захисту інформації в комп'ютерних системах.
2. Рівні забезпечення інформаційної безпеки.

1.1 Актуальність проблеми захисту інформації в комп'ютерних системах

У результаті інформаційних процесів, що відбуваються у світі, перше місце разом із завданнями ефективного оброблення і передачі інформації займає найважливіше завдання – забезпечення безпеки інформації. Це пояснюється особливою значущістю для розвитку держави інформаційних ресурсів, зростанням вартості інформації в умовах ринку, її високою вразливістю і нерідко значними збитками у результаті її несанкціонованого використання.

У багатьох країнах порушення безпеки в системах оброблення і передачі інформації спричиняють великі втрати. Вони є найбільш значними в системах телекомунікації, що обслуговують банківські і торгові установи. В США, наприклад, збитки від несанкціонованого проникнення у ці системи оцінюються у десятки мільйонів доларів, а у деяких випадках і мільярди доларів. Так, наприклад, економічні втрати від вірусу Petya.A і вірусу WannaCry у 2018 р. становили близько одного мільярда доларів.

Аналітичний центр компанії InfoWatch проводить щорічні дослідження джерел і каналів витоків інформації у країнах світу. Розглянемо деякі приклади зростання витоків інформації. На рисунку 1.1 показано динаміку зростання кількості витоків конфіденційної інформації за дванадцять років з 2006 р. до 2017 р. Загальна кількість (обсяг) витоків записів становила, наприклад, у 2015 р. – 965,9 мільйонів, у 2016 р. – 3147,7 мільйонів, а у 2017 р. – вже 13285,8 мільйонів, тобто за один рік (з 2016 р. до 2017 р.) кількість витоків записів (даних, що скомпрометовані) зросла майже в чотири рази.

Витоки персональних і платіжних даних у розподілі за типом інформації у 2017 р. зменшилися на 7 % порівняно з 2016 р. і становили 85,9 % (персональні дані – 64,8 %, платіжні дані – 21,1 %). Витоки даних, пов'язані з комерційною таємницею, у 2017 р. становили 8,0 %, а пов'язані з державною таємницею – 6,1 %.

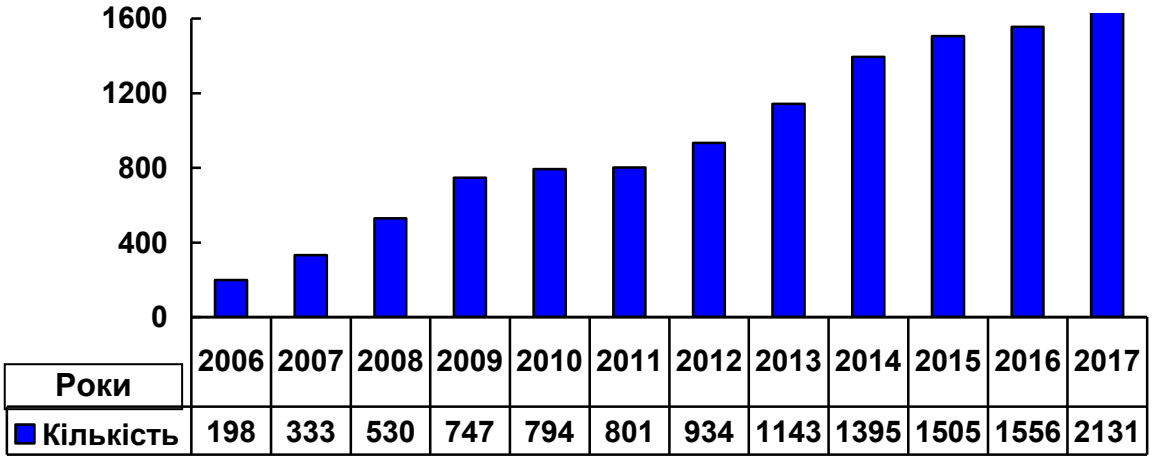

Рисунок 1.1 – Динаміка зростання кількості витоків конфіденційної інформації за роками

У 2017 р. зареєстровано 802 випадки витоків інформації (39,52 %), причиною яких був зовнішній зловмисник. У 1228 випадках (60,5 %) витоки інформації трапились з вини або неуважності внутрішнього порушника.

Кількість випадків витоків даних у країнах у 2016 і 2017 роках наведено в таблиці 1.1.

Таблиця 1.1 – Кількість витоків даних у країнах

Країна	2016 р.	2017 р.
США	838	1089
Росія	213	254
Великобританія	67	104
Канада	37	69
Україна	34	34
Германія	31	11
Австралія	25	65
Індія	22	46
Японія	20	-
Китай	16	29
Ірландія	-	15
Південна Корея	-	14

У 2018 р. розподіл витоків інформації значно не змінився. Так, 83,9 % становили некваліфіковані (прості) витоки даних, 8,5 % – витоки, пов’язані з шахрайством при використанні даних, а 7,6 % – це перевищення прав доступу. На рисунках 1.2 і 1.3 показано відповідно розподіли інцидентів за винуватцями і за каналами витоків у 2018 р. З рисунку 1.2 видно, що 69,2 % інцидентів пов’язано з діяльністю керівників і співробітників організацій і установ, тобто мають внутрішній характер, і тільки 21,9 % інцидентів – із зовнішнім зловмисником.

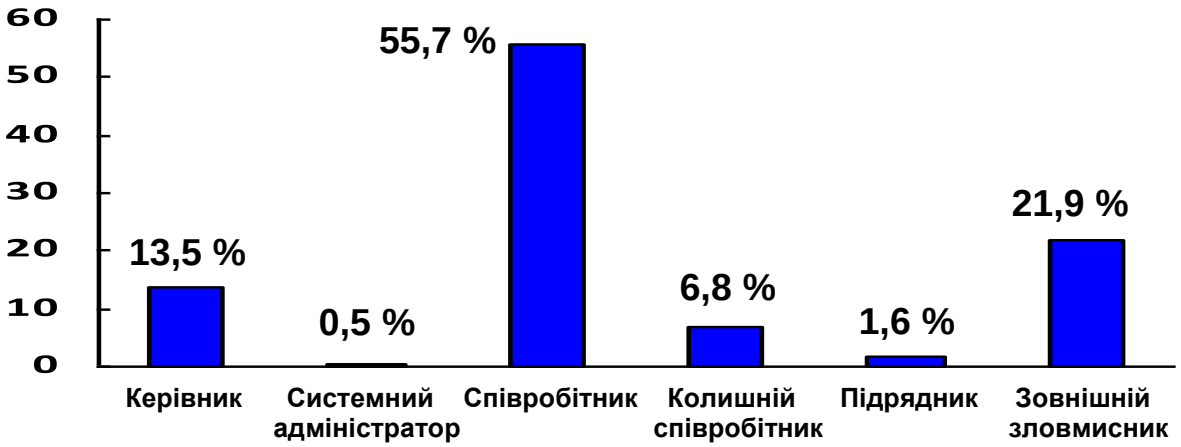

Рисунок 1.2 – Розподіл інцидентів за винуватцями у 2018 р.

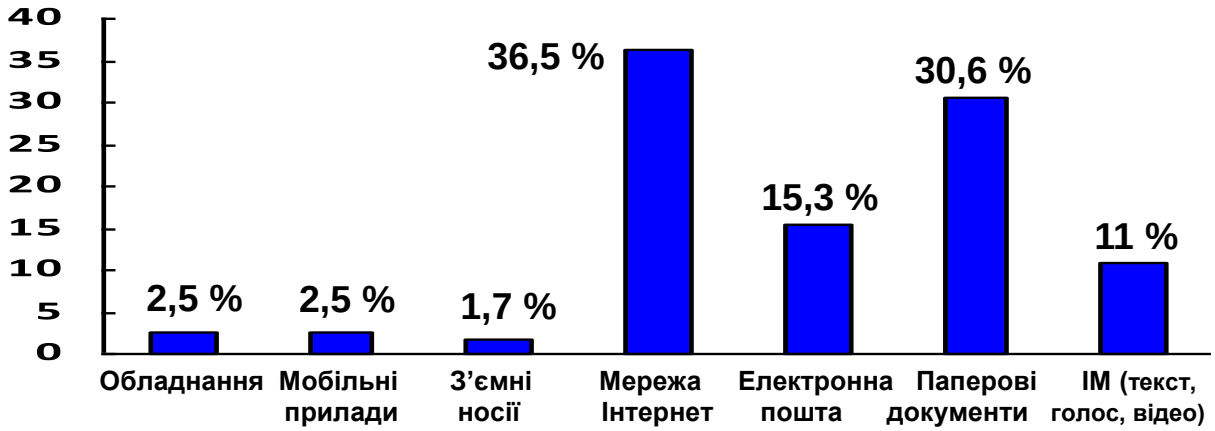

Рисунок 1.3 – Розподіл інцидентів за каналами витоків у 2018 р.

Аналіз даних, зображених на рисунку 1.3, свідчить про те, що найбільша кількість інцидентів пов'язана з мережею Інтернет (36,5 %), електронною поштою (15,3 %) та з втратою паперових документів (30,6 %). На рисунку 1.4 показано розподіл інцидентів за галузями у 2018 р.

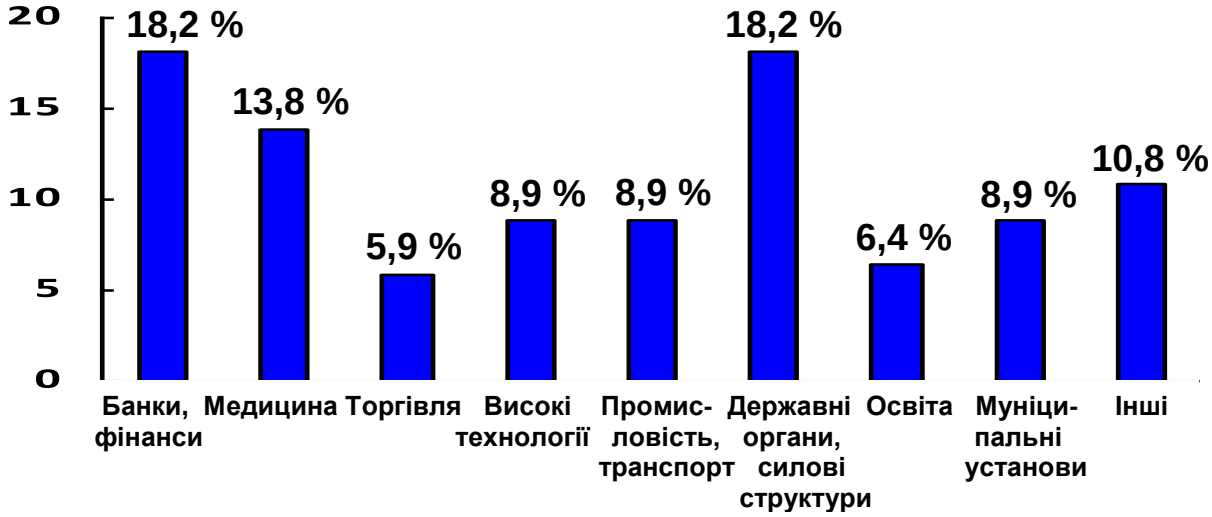

Рисунок 1.4 – Розподіл інцидентів за галузями у 2018 р.

З рисунку 1.4 видно, що найбільша кількість інцидентів, пов’язаних з витоками даних, сталася у банківській і фінансовій сферах (18,2 %), а також в органах державного керування та в силових структурах (18,2 %). Необхідно зазначити, що значний відсоток інцидентів пов'язано з медичними даними (13,8 %), коли зловмисники блокують бази даних медичних установ і для їх розблокування вимагають викуп.

Про ступінь небезпеки електронних злочинів можна робити висновок за тими витратами на засоби захисту, які вважаються припустимими і доцільними. За оцінками фахівців США, загальні витрати на захист банківської або іншої фінансової установи можуть становити всього 5 – 10 тисяч доларів. Однак вартість надійної системи захисту великої фінансової установи, яка обслуговує до 80 тисяч користувачів, становить не менше 15 мільйонів доларів, причому в цю суму входить тільки вартість апаратних і програмних засобів.

Несанкціоноване отримання інформації має різні наслідки: від нешкідливих пустощів до фінансових втрат великих розмірів. Як показав аналіз, наведений вище, найуразливішими об'єктами, які страждають від несанкціонованого доступу до даних, є системи автоматизованого перерахування грошових коштів. Останнім часом також різко почастішали випадки розкрадання програм до комп'ютерних мереж (8,9 %). Ці розкрадання мають ознаки епідемії: на кожну законну копію програми, що має досить значне поширення, існує декілька копій, отриманих незаконним шляхом.

В обчислювальних машинах є велика кількість лазівок для несанкціонованого доступу до інформації. Ніякий окремо взятий засіб захисту не може забезпечити адекватну безпеку. Скільки-небудь надійний захист може бути гарантований лише при створенні механізму комплексного забезпечення безпеки як засобів оброблення інформації, так і каналів зв'язку.

Останніми роками інтерес до питань захисту інформації значно зріс також і у зв'язку з поширенням користування персональними ЕОМ і мобільними персональними засобами телекомунікації. Пояснюється це їх масовістю, недостатньою досвідченістю користувачів при вмиканні таких засобів в мережі загального користування і відсутністю досвіду їх захисту.

Особливу небезпеку для інформаційних систем являють собою спеціальні програми, що отримали назву «комп'ютерні віруси». Їх проникнення в ЕОМ дозволяє практично вирішувати проблеми несанкціонованого отримання інформації, її фізичного знищення, модифікації програмного забезпечення та ін.

Проблеми захисту мереж телекомунікації збільшуються як під впливами специфічних характеристик сучасних мереж, так і їх внутрішніх суперечностей.

Внутрішня суперечність визначається тим, що мета створення телекомунікаційних мереж і забезпечення їх захисту абсолютно різні: якщо

в мережі є можливим максимальний доступ до інформаційних ресурсів, то для створення безпеки мережі передбачають введення обмеженого доступу в умовах жорсткого контролю. Тому, виходячи з актуальної потреби забезпечення безпеки інформаційних систем, необхідно створювати сучасні технології захисту інформації, яка обробляється в інформаційних системах і передається по телекомунікаційних каналах.

Специфічні особливості забезпечення безпеки і висока вартість технічних засобів захисту тривалий час обмежували їх комерційне впровадження і повний опис методів захисту у відкритому друку. Досягнення останніх років в області обчислювальної техніки, мікроелектроніки і зв'язку дозволяють сьогодні по-новому підійти до проблеми безпеки в інформаційних мережах, забезпечивши широке застосування технологій захисту інформації.

Поширення персональних ЕОМ і неможливість організації ефективного контролю над їх застосуванням створили умови для різкого зниження рівня безпеки інформаційних систем. При вирішенні проблеми забезпечення безпеки інформації слід застосовувати комплекс заходів і, в першу чергу, такі дії, як розроблення системи класифікації і документування інформації та способів захисту, регулювання доступу до даних і встановлення відповідальності за порушення інформаційної безпеки.

1.2 Рівні забезпечення інформаційної безпеки

Під **інформаційною безпекою** розуміють захищеність інформаційної системи від випадкового або навмисного втручання, що завдає збитків власникам або користувачам інформації.

На практиці найважливішими є **три аспекти інформаційної безпеки**:

- **доступність**, під якою розуміють можливість за певний час отримати необхідну інформаційну послугу;
- **цілісність**, яка означає актуальність і несуперечність інформації, її захищеність від руйнування і несанкціонованої зміни;
- **конфіденційність**, спрямована на захист від несанкціонованого читання.

Сучасна інформаційна система є складною системою, що містить велику кількість компонентів різного ступеня автономності, які зв'язані між собою і обмінюються даними. Практично кожен компонент може піддатися зовнішній дії або вийти з ладу.

Компоненти автоматизованої інформаційної системи можна поділити на такі групи:

- **апаратні засоби**, до яких належать комп'ютери та їх складові частини, а саме: процесори, монітори, термінали, периферійні пристрої (дисководи, принтери, контролери, кабелі, лінії зв'язку і т. д.);

- **програмне забезпечення**, що складається з програм, вихідних, об'єктних і завантажувальних модулів, операційних систем і системних програм (компіляторів, компоновщиків і т. д.), утиліт, діагностичних програм та ін.;

- **дані**, що зберігаються тимчасово і постійно на магнітних носіях, надруковані документи, архіви, системні журнали і т. д.;

- **персонал**, до якого належать обслуговуючі співробітники і користувачі.

Небезпечні дії, спрямовані на комп'ютерну інформаційну систему, можна поділити на **випадкові і навмисні**.

Причинами **випадкових дій** при експлуатації можуть бути:

- аварійні ситуації, пов’язані зі стихійним лихом і відключеннями електроживлення;
- відмови і збої апаратури;
- помилки у програмному забезпеченні;
- помилки у роботі персоналу;
- завади у лініях зв'язку, пов’язані з впливом зовнішнього середовища.

Навмисні дії – це цілеспрямовані дії порушника, яким можуть бути службовець, відвідувач, конкурент, найманець.

Дії порушника можуть бути обумовлені різними **мотивами**:

- незадоволеністю своєю кар'єрою;
- хабаром;
- цікавістю;
- конкурентною боротьбою;
- прагненням самоствердуватися за будь-яку ціну.

Найбільш поширеним видом комп'ютерних порушень є **несанкціонований доступ**, коли зловмисник використовує будь-яку помилку у системі захисту, яка може виникнути через нераціональний вибір засобів захисту, їх некоректне установлення або настройку.

Метою захисту інформації є:

- запобігання її витоку або розкраданню, втратам, спотворенню і підробленню;
- уникнення погроз безпеки особи, суспільства і держави;
- запобігання несанкціонованим діям зі знищення, модифікації, спотворення, копіювання і блокування інформації, а також іншим формам незаконного втручання в інформаційні ресурси та системи;
- створення правового режиму документування інформації як об'єкта власності;
- відстоювання конституційних прав громадян щодо збереження особистої таємниці та конфіденційності персональних даних, наявних в інформаційних системах;
- збереження державної таємниці, конфіденційності документованої інформації відповідно до законодавства;

- гарантія прав суб'єктів при створенні інформаційних процесів і при розробленні та застосуванні інформаційних систем, технологій і засобів їх забезпечення.

Формування режиму інформаційної безпеки – **проблема комплексна**. Заходи щодо забезпечення інформаційної безпеки можна поділити на **п'ять рівнів**:

- **законодавчий** – ухвалення і виконання законів, нормативних актів і стандартів, що регламентують дії учасників у сфері захисту інформації;
- **морально-етичний** – створення і впровадження у повсякденне життя людей норм поведінки щодо правильного використання інформації, недотримання яких веде до падіння престижу конкретної людини або цілої організації;
- **адміністративний** – дії загального характеру, що запроваджуються керівництвом організації з питань захисту інформації (створення інструкцій, правил, систем захисту інформації та ін.);
- **фізичний** – створення механічних, електро- та електронно-механічних перешкод на можливих шляхах проникнення потенційних порушників;
- **апаратно-програмний** – впровадження в організаціях електронних пристроїв і спеціальних (комп'ютерних) програм щодо захисту інформації.

Сукупність цих заходів на всіх рівнях **утворює систему захисту інформації**, яка спрямована на протидію погрозам безпеки з метою зведення до мінімуму можливих збитків.

Надійна система захисту інформації має відповідати таким принципам:

- вартість засобів захисту має бути менше розмірів можливого збитку;
- кожен користувач повинен мати мінімальний набір привілеїв, необхідних для роботи;
- захист тим більше ефективний, чим простіше користувачеві з ним працювати;
- можливість відключення системи захисту в екстрених випадках;
- фахівці, що мають відношення до системи захисту, повинні повністю уявляти собі принципи її функціонування і у разі виникнення скрутних ситуацій адекватно на них реагувати;
- під захистом слід тримати всю систему оброблення інформації, а не окремі її елементи;
- розробники системи захисту не мають бути у числі тих, кого ця система буде контролювати;
- особи, що займаються забезпеченням інформаційної безпеки, мають нести за це особисту відповідальність;
- система захисту має надавати докази коректності своєї роботи;
- об'єкти захисту доцільно поділяти на групи так, щоб порушення захисту в одній з груп не впливало на безпеку інших;

- надійна система захисту має бути повністю протестована і узгоджена;
- захист стає ефективнішим і більш гнучким, якщо адміністратор може змінити його параметри;
- систему захисту слід розробляти виходячи з припущення, що користувачі здійснюватимуть серйозні помилки і взагалі можуть мати якнайгірші наміри;
- найбільш важливі і критичні проблеми має вирішувати людина;
- існування механізмів захисту має бути, якщо можливо, прихованим від користувачів, робота яких знаходиться під контролем.

Дотримання зазначених принципів дозволить забезпечити підвищення захисту інформації в організації. Однак користувачам слід усвідомлювати, що абсолютного (стовідсоткового) захисту інформації не існує. Будь-який захист інформації вимірюється часом «злому». Застосування комплексу заходів із захисту інформації тільки збільшує час, за який зловмисник зможе отримати інформацію, яка зберігається у комп'ютерних системах.

Запитання для самоперевірки

1. Які тенденції щодо витоків конфіденційної інформації спостерігаються у світі ?
2. У яких сферах діяльності людини спостерігається найбільша кількість інцидентів, пов'язаних із витоками даних ?
3. У чому полягають внутрішні суперечності сучасних мереж щодо захисту інформації ?
4. Які існують основні аспекти інформаційної безпеки ?
5. У чому полягає мета захисту інформації ?
6. Які існують основні рівні щодо забезпечення інформаційної безпеки ?
7. Яким принципам має відповідати надійна система захисту інформації ?

Лекція № 2

НОРМАТИВНО-ПРАВОВА БАЗА ДЛЯ ОРГАНІЗАЦІЇ І ПРОВЕДЕННЯ ЗАХОДІВ ІЗ ЗАХИСТУ ІНФОРМАЦІЇ

Навчальні цілі:

- вивчити основні нормативні документи (закони) України щодо захисту інформації;
- розглянути зарубіжний досвід нормативно-правового забезпечення захисту інформації.

Навчальні питання:

1. Основні нормативні документи (закони) України щодо захисту інформації.
2. Зарубіжний досвід нормативно-правового забезпечення захисту інформації.

2.1 Основні нормативні документи (закони) України щодо захисту інформації

Нормативно-правове забезпечення роботи організації і заходи щодо захисту інформації базуються на сукупності законів, нормативних актів і правил, що регламентують як загальну організацію робіт, так і створення, і функціонування конкретних систем захисту інформації.

Основними законами, які регулюють відносини у сфері захисту інформації в Україні є:

- Закон України «Про інформацію» у редакції від 13.01.2011;
- Закон України «Про захист інформації в автоматизованих системах» у редакції від 11.05.2004, який був замінений Законом України «Про захист інформації в інформаційно-телекомунікаційних системах» у редакції від 27.03.2014 (Додаток).

Закон України «Про інформацію» регулює відносини щодо створення, збирання, одержання, зберігання, використання, поширення, охорони, захисту інформації. У цьому Законі визначено **основні види інформаційної діяльності**, до яких належать: створення, збирання, одержання, зберігання, використання, поширення, охорона та захист інформації.

Тут також наведено **види інформації за змістом**, а саме:

- про фізичну особу;
- довідково-енциклопедичну;
- екологічну (про стан довкілля);
- про товар (роботу, послугу);
- науково-технічну інформацію;
- податкову;

- правову;
- статистичну;
- соціологічну,

а також інші види інформації.

До **інформації про фізичну особу (персональні дані)** належать відомості чи сукупність відомостей про фізичну особу, яка ідентифікована або може бути конкретно ідентифікована. Не допустимо збирання, зберігання, використання та поширення конфіденційної інформації про особу без її згоди, крім випадків, визначених законом, і лише в інтересах національної безпеки, економічного добробуту і захисту прав людини.

До **конфіденційної інформації про фізичну особу** належать, зокрема, **дані про її національність, освіту, сімейний стан, релігійні переконання, стан здоров'я, а також адреса, дата і місце народження**.

До **довідково-енциклопедичної інформації** належать систематизовані, документовані, публічно оголошені або іншим чином поширені відомості про суспільне, державне життя і навколишнє природне середовище. Основними джерелами такої інформації є: енциклопедії, словники, довідники, рекламні повідомлення та оголошення, путівники, картографічні матеріали, електронні бази та банки даних, архіви різних довідкових інформаційних служб, мереж та систем, а також довідки, що видаються уповноваженими на те органами державної влади та органами місцевого самоврядування, об'єднаннями громадян, організаціями, їх працівниками та автоматизованими інформаційно-телекомунікаційними системами.

До **інформації про стан довкілля** (екологічної інформації) належать такі відомості:

- про стан складових довкілля та його компоненти, включаючи генетично модифіковані організми, взаємодію цих складових;
- фактори, що впливають або можуть впливати на складові довкілля (речовини, енергія, шум і випромінювання, а також діяльність або заходи, включаючи адміністративні, угоди в галузі навколишнього природного середовища, політику, законодавство, плани і програми);
- стан здоров'я та безпеку людей, умови життя людей, стан об'єктів культури і споруд щодо впливу їх на стан складових довкілля;
- інші відомості та/або дані.

До **інформації про товар (роботу, послугу)** належать відомості, які відображають кількісні, якісні та інші характеристики товару (роботи, послуги).

До **науково-технічної інформації** відносять будь-які відомості про вітчизняні та зарубіжні досягнення науки, техніки і виробництва, одержані в процесі науково-дослідної, дослідно-конструкторської, проектно-технологічної, виробничої та громадської діяльності, які можуть бути збережені на матеріальних носіях або відображені в електронному вигляді.

До **податкової інформації** належить сукупність відомостей і даних, що створені або отримані суб'єктами інформаційних відносин у процесі поточної діяльності і необхідні для реалізації покладених на контролюючі органи завдань і функцій у порядку, встановленому Податковим кодексом України.

До **правової інформації** – будь-які відомості про право, його систему, джерела, реалізацію, юридичні факти, правовідносини, правопорядок, правопорушення і боротьбу з ними та їх профілактику тощо.

До **статистичної інформації** – документована інформація про кількісну характеристику масових явищ та процесів, які відбуваються в економічній, соціальній, культурній та інших сферах життя суспільства.

До **соціологічної інформації** – будь-які документовані відомості про ставлення до окремих осіб, подій, явищ, процесів, фактів тощо.

За порядком доступу інформацію поділяють на **відкриту** та **з обмеженим доступом**.

Інформацією з обмеженим доступом є **конфіденційна**, **таємна** та **службова інформація**.

Конфіденційною є інформація про фізичну особу, а також інформація, доступ до якої обмежено фізичною або юридичною особою, крім суб'єктів владних повноважень. Конфіденційна інформація може поширюватися за бажанням (згодою) відповідної особи у визначеному нею порядку відповідно до передбачених нею умов, а також в інших випадках, визначених законом.

Порядок віднесення інформації до таємної або службової, а також порядок доступу до неї регулюється законами.

До інформації з обмеженим доступом не можуть належати такі відомості:

- про стан довкілля, якість харчових продуктів і предметів побуту;
- аварії, катастрофи, небезпечні природні явища та інші надзвичайні ситуації, що сталися або можуть статися і загрожують безпеці людей;
- стан здоров'я населення, його життєвий рівень, включаючи харчування, одяг, житло, медичне обслуговування та соціальне забезпечення, а також про соціально-демографічні показники, стан правопорядку, освіти і культури населення;
- факти порушення прав і свобод людини і громадянина;
- незаконні дії органів державної влади, органів місцевого самоврядування, їх посадових та службових осіб;
- інші відомості, доступ до яких не може бути обмежено відповідно до законів та міжнародних договорів України, згода на обов'язковість яких надана Верховною Радою України.

Порушення законодавства України про інформацію має дисциплінарну, цивільно-правову, адміністративну або кримінальну відповідальність згідно із законами України.

Найбільш сучасним законом, який регулює відносини у сфері захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах є **Закон України «Про захист інформації в інформаційно-телекомунікаційних системах»**.

У цьому Законі наведено основні визначення (терміни), що використовують під час здійснення процесу захисту інформації в інформаційно-телекомунікаційних системах. Коротко розглянемо їх значення.

Інформаційна (автоматизована) система – організаційно-технічна система, в якій реалізується технологія оброблення інформації з використанням технічних і програмних засобів.

Телекомунікаційна система – сукупність технічних і програмних засобів, призначених для обміну інформацією шляхом передавання або приймання її у вигляді сигналів, знаків, звуків, рухомих або нерухомих зображень або іншим способом.

Інформаційно-телекомунікаційна система – сукупність інформаційних і телекомунікаційних систем, які в процесі оброблення інформації діють як єдине ціле.

Володілець інформації – фізична або юридична особа, якій належать права на інформацію.

Власник системи – фізична або юридична особа, якій належить право власності на систему.

Користувач інформації в системі – фізична або юридична особа, яка в установленому законодавством порядку отримала право доступу до інформації в системі.

Захист інформації в системі – діяльність, спрямована на запобігання несанкціонованим діям щодо інформації в системі.

Комплексна система захисту інформації – взаємозв'язані організаційні та інженерно-технічні заходи, засоби і методи захисту інформації.

Криптографічний захист інформації – вид захисту інформації, що реалізується шляхом перетворення інформації з використанням спеціальних (ключових) даних з метою приховування/відновлення змісту інформації, підтвердження її справжності, цілісності, авторства тощо.

Технічний захист інформації – вид захисту інформації, спрямований на забезпечення за допомогою інженерно-технічних заходів та/або програмних і технічних засобів унеможливлення витоку, знищення та блокування інформації, порушення цілісності та режиму доступу до інформації.

Витік інформації – результат дій, внаслідок яких інформація в системі стає відомою або доступною фізичним та/або юридичним особам, які не мають права доступу до неї.

Знищення інформації – дії, внаслідок яких інформація в системі зникає.

Блокування інформації – дії, внаслідок яких унеможливлюється доступ до інформації в системі.

Порушення цілісності інформації – несанкціоновані дії щодо інформації в системі, внаслідок яких змінюється її вміст.

Порядок доступу до інформації – умови отримання користувачем можливості обробляти інформацію в системі та правила оброблення цієї інформації.

Об'єктом захисту є інформація, що обробляється в системі, та програмне забезпечення, яке призначено для оброблення цієї інформації.

Порядок доступу до інформації, перелік користувачів та їх повноваження стосовно цієї інформації визначаються володільцем інформації. Відповідальність за забезпечення захисту інформації в системі покладається на власника системи.

2.2 Зарубіжний досвід нормативно-правового забезпечення захисту інформації

У розвинених зарубіжних країнах питанням правового забезпечення безпеки інформації приділяється велика увага. Досвід інших держав свідчить, що нормативно-правова база створюється не тільки для захисту державних інформаційних ресурсів, але й для забезпечення безпеки інформаційних ресурсів незалежного сектора економіки.

Особливістю західного комп'ютерного права є те, що воно формується за прецедентом і відображається в основних законодавчих актах прямої дії, що приймаються на державному (у Великобританії, Франції, Німеччині) або федеральному і регіональному рівнях (у США).

Відповідальність за правопорушення при роботі з інформацією, комп'ютерами і комп'ютерною інформацією встановлюється у США такими законами та підзаконними актами: Законом про безпеку комп'ютерних систем; Актом про зловживання з використанням ЕОМ; Законом про свободу інформації; Законом про дотримання таємниць; Законом про фінансові таємниці; Законом про справедливі кредити; Законом про захист обчислювальних засобів невеликих фірм і підприємств. На основі закону США про безпеку комп'ютерних систем відповідальним за захист автоматизованих інформаційних систем і ліній комунікацій визнано Національне бюро стандартів, і це бюро зобов'язано використовувати у разі потреби технічні рекомендації і допомогу Агентства національної безпеки США.

У Канаді відповідальність за правопорушення при роботі з інформацією регламентується Законом про комп'ютерні та інформаційні злочини (Білль С-18), а у Франції – Законом про інформатику, картотеки і свободи. У Великобританії діє Закон про захист інформації. У Німеччині також існує низка законів та підзаконних актів, серед яких Закон про подальший розвиток електронного оброблення і захист даних; Закон про

господарську злочинність; Закон про авторське право; Німецький Кримінальний кодекс; Німецький Цивільний кодекс; Федеральний закон про захист даних.

Законопроекти про захист програм розробляються також в Японії, причому ініціатором цих законопроектів виступає міністерство зовнішньої торгівлі і промисловості. Але на відміну від США, в проектах цих законів передбачається, що програми слід захищати промисловим, а не авторським правом.

Для вирішення задач сертифікації засобів захисту інформації, захищених засобів обчислювальної техніки і телекомунікаційних засобів різного призначення створені і функціонують Національний центр комп'ютерної безпеки в США, Центральне агентство з комп'ютеризації і телекомунікацій у Великобританії, а також аналогічні державні органи в Швеції, Австралії та інших країнах.

Як зазначалося вище, у США велика увага приділяється законодавчим засобам захисту інформації. Коротко розглянемо застосування трьох видів законодавчих актів (законів про професійні таємниці, законів про авторське право і законів про патентні права) для захисту комп’ютерних програм.

Професійною таємницею у США можуть бути визнані будь-які розробки (пристрої, математичні моделі, методи і тому подібне), які не є загальновідомими і використання яких в комерційних цілях дає перевагу над тими конкурентами, яким невідомі ці розробки.

Будь-яка комп'ютерна програма може бути визнана професійною таємницею. Основна вимога, що ставиться американським законодавством до професійних таємниць, – не бути загальновідомими. Якщо ж професійна таємниця, що охороняється законом, яким-небудь чином стала загальновідомою, то охоронна дія законів на цю таємницю припиняється. Тому розробник програм має тримати в таємниці всю ту інформацію, за якою може бути розкритим зміст програми.

Щоб комп'ютерну програму було взято під захист закону про авторські права, необхідно зареєструвати її в Агентстві авторських прав, для чого має бути подана копія програми, що реєструється, а також надана копія програми або її опис у бібліотеку Конгресу США для загального користування. Після реєстрації всі копії програми, виготовлені будь-яким шляхом, вважаються копіями з авторського екземпляра. Такий спосіб захисту програми вважається дуже простим, причому досить просто встановити факт несанкціонованого використання захищених програм. Однак авторське право захищає лише конкретну реалізацію програми. Якщо яка-небудь фірма розробить свою версію програми, навіть повністю побудовану на принципах захищеної програми, то це не вважається порушенням авторського права. Більш того, таку програму може бути взято під захист закону про авторські права.

Комп'ютерні програми можуть також знаходитися під захистом законів про патентне право. Однак такий захист пов'язано з великими процедурними труднощами. Закони про патентний захист є федеральними, а регулювання професійної таємниці здійснюється законодавством штатів. Для отримання патенту необхідно подати детальний опис розробки, а публікація такого опису автоматично припиняє її захист законами про професійну таємницю.

Найбільш раціональною формою комбінованого захисту вважають таку: основні ідеї і принципи побудови програмних систем патентують, а їх конкретна програмна реалізація захищається законами про професійну таємницю.

Міністерство оборони США визначило рівні захисту інформаційних систем у книзі «Критерії оцінки безпеки комп'ютерів», так званій «Оранжевій книзі» (за кольором обкладинки):

- **D** – рівень мінімального захисту (Minimal Protection) систем, яким за іншими рівнями не гарантовано потрібного рівня безпеки;
- **C1** – рівень вибіркового захисту (Discretionary Protection), обмеження доступу користувачів до захисту приватної інформації;
- **C2** – рівень керованого доступу (Controlled Access Protection), до вимог рівня C1 додається захист процесу реєстрації у системі, облік подій захисту, ізоляційні ресурси різних процесів;
- **B1** – рівень захисту за категоріями (Labeled Protection), до вимог рівня C2 додається захист окремих файлів, записів у файлах, а також інших об'єктів системи спеціальними позначками безпеки, що зберігаються разом з цими об'єктами. Вважають, що подолати такий захист може добре підготовлений хакер, а звичайний користувач – ні;
- **B2** – рівень структурованого захисту (Structured Protection), до вимог рівня B1 додається повний захист усіх ресурсів системи прямо чи посередньо доступних користувачеві. Вважають, що хакери не зможуть проникнути у систему з таким захистом;
- **B3** – рівень доменів безпеки (Security Domains), до вимог рівня B2 додається явна специфікація користувачів, яким заборонено доступ до певних ресурсів, повніша реєстрація потенційно небезпечних подій. Вважають, що навіть досвідчені програмісти не в змозі подолати систему з таким рівнем безпеки;
- **A1** – рівень верифікованої розробки (Verified Design) забезпечує повний захист інформації, коли застосовують специфіковані та верифіковані механізми захисту. Вважають, що в систему з таким рівнем захисту без дозволу не може ввійти ніхто (навіть спеціалісти спецслужб).

Основними нормативними актами, що регламентують питання захисту інформації в Російській Федерації є Закон «Про інформацію, інформатизацію та захист інформації» і Закон «Про державну таємницю». Федеральним Законом «Про інформацію, інформатизацію та захист інформації» гарантується право власника інформації на її використання і

захист від доступу до неї інших осіб (організацій). Якщо доступ до інформації обмежується, то така інформація є конфіденційною і може містити державну або комерційну таємницю. Комерційну таємницю можуть містити відомості, що належать приватній особі, фірмі, корпорації та ін., державну таємницю – відомості, що належать державі (державній установі). Відповідно до закону «Про державну таємницю» відомостям, що являють собою цінність для держави, може бути надано один з трьох можливих ступенів секретності. При зростаючій цінності (важливості) інформація може одержати ступінь (гриф) «таємно», «цілком таємно» або «особливої важливості». У державних установах менш важливій інформації може бути надано гриф «для службового користування».

При всій відмінності національні законодавства мають ряд загальних моментів:

- практично у всіх країнах на законодавчому рівні встановлено відповідальність за порушення порядку оброблення і використання персональних даних;
- інформаційні (комп'ютерні) злочини розцінюються як особливо небезпечні для громадян, держави і суспільства в цілому і зазнають значно жорсткішого покарання, чим аналогічні злочини, здійснені без використання комп'ютерної техніки;
- дії, що створюють умови для учинення збитків, наприклад, спроба проникнення в систему, впровадження програми-вірусу та ін., кваліфікуються як злочини.

В умовах ринку і технологій зберігання, оброблення і передачі інформації доцільно створювати спеціальну інформаційно-телекомунікаційну систему, яка призначена для інформаційного забезпечення керування в межах держави і створення сучасної інформаційної інфраструктури, що сприяє розвитку ринкових відносин.

Запитання для самоперевірки

1. Які існують види інформації за змістом ?
2. Яка інформація про фізичну особу є конфіденційною ?
3. Які відомості не можуть належати до інформації з обмеженим доступом ?
4. У чому полягає різниця між інформаційною (автоматизованою) і телекомунікаційною системами ?
5. Що таке криптографічний захист інформації ?
6. На що спрямоване нормативно-правове забезпечення захисту інформації за рубежем ?
7. Як організують і здійснюють захист комп’ютерних програм у США ?

Лекція № 3

ШКІДЛИВІ ПРОГРАМИ

Навчальні цілі:
- вивчити класифікацію шкідливих програм;
- розглянути історію розвитку шкідливих програм.

Навчальні питання:
1. Класифікація шкідливих програм.
2. Коротка історія розвитку шкідливих програм.

3.1 Класифікація шкідливих програм

Сьогодні все частіше різні шкідливі коди називають вірусами, але це не зовсім правильно. За способом поширення шкідливі програми можна поділити так:

- комп'ютерні віруси;
- троянські програми;
- мережні черв'яки;
- програмні закладки.

Сучасні шкідливі програми і атаки шахраїв у наш час в основному спрямовані на отримання фінансової вигоди або керування комп'ютером користувача, але не на знищення даних. При цьому для завдавання шкоди та збитків не завжди потрібний вірус, нерідко достатньо правильно складеного листа або SMS, щоб його одержувач сам відправив відомості про кредитну картку або іншу конфіденційну інформацію зловмисникам.

Комп'ютерні віруси здатні розмножуватися самостійно, додаючи свій код до інших файлів або у службові області диска. Обов'язковою (необхідною) властивістю комп'ютерного вірусу є можливість створювати свої дублікати (не обов'язково збіжні з оригіналом) і впроваджувати їх в обчислювальні мережі і/або файли, системні області комп'ютера та інші виконувані об'єкти. При цьому дублікати зберігають здібність до подальшого поширення.

Троянські програми, або просто **Трояни** – це шкідливі програми, які самі не розмножуються, а, маскуючись під популярну програму, спонукають користувача переписати і самостійно встановити на свій комп'ютер шкідника.

Коли хакери намагаються внести вірус на чийсь комп'ютер, то в більшості випадків їм потрібно, щоб користувач особисто запустив програму. Щоб переконати власника персонального комп'ютера зробити це, троянська програма, як заведено, видається за якесь корисне програмне забезпечення, наприклад, критичне оновлення операційної

системи Windows, антивірус, кодек, необхідний для перегляду відео на сайті і т. д.

Принципова відмінність троянських програм і вірусів полягає у тому, що вірус є програмою, що самостійно розмножується, тоді як троянська програма не має можливості самостійно поширюватися. Однак у наш час досить часто зустрічаються гібриди-віруси (в основному e-mail і мережні черв'яки), разом з якими поширюються троянські програми і програмні закладки.

Мережні черв'яки не змінюють файли на дисках, а поширюються в комп'ютерній мережі, проникають в операційну систему комп'ютера, знаходять адреси інших комп'ютерів або користувачів і розсилають ним свої копії, використовуючи різне середовище поширення.

У всіх **програмних закладок** (незалежно від методу їх впровадження у комп'ютерну систему, терміну перебування в оперативній пам'яті і призначення) є одна важлива загальна риса: вони обов'язково виконують операцію запису в оперативну або зовнішню пам'ять системи.

Непрямі ознаки зараження комп'ютера є такими:

- відмова роботи однієї або декількох програм, особливо антивірусу і брандмауера;
- поява спливаючих вікон, що містять рекламу;
- періодична поява вікна з’єднання з провайдером або з невідомим сайтом;
- за відсутності активності на підключеному до мережі Інтернет комп'ютері (нічого не скачується, програми спілкувань неактивні і т. д.) індикатори підключення до мережі продовжують показувати обмін інформацією;
- стартова сторінка браузера постійно міняється, а сторінка, вказана як стартова, не зберігається;
- при спробі відвідати сайти, куди раніше легко було «заходити» (наприклад, в пошукові системи), комп'ютер переадресовує запит на незнайомий сайт, що часто містить порнографічну або рекламну інформацію.

Шкідливі програми-вимагачі, як правило, спрямовані на блокування комп'ютера, але також можуть здійснювати й інші дії:

- видаляти всі файли з комп'ютера, якщо протягом декількох годин не будуть переведені гроші на певний мобільний рахунок;
- попереджати про те, що операційна система Windows не пройшла ліцензійної перевірки і щоб її зареєструвати, необхідно отримати код за SMS;
- прибирати порнобанер тільки за допомогою SMS;
- блокувати доступ на певний сайт, наприклад, Facebook і висувати вимогу відправити SMS для розблокування;
- попереджати користувача про те, що онлайновий антивірус виявив небезпечну загрозу на комп'ютері. Вірус тимчасово блоковано (разом з

комп'ютером), але щоб остаточно його видалити, потрібна ліцензійна версія антивірусної програми, яку можна отримати, відправивши SMS;

- шифрувати файли користувача вірусом (найчастіше офісні документи) і пропонувати повернути до них доступ тільки після введення коду, отриманого за SMS.

Таким чином, основною метою шахраїв, які використовують шкідливі програми, є отримання грошей (суми можуть варіюватися в широких межах) шляхом блокування даних та викрадання фінансових даних, а також застосування комп'ютера або акаунтів його власника для розсилки спаму.

3.2 Коротка історія розвитку шкідливих програм

У **1962 р.**, ще до появи сучасних вірусів, інженери з американської компанії Bell Telephone Laboratories створили гру під назвою **«Дарвін»**, в якій були дії, які пізніше стали притаманні вірусам. В грі була реалізована ідея виживання і розвитку програмних кодів, що створювалися гравцями. Гра припускала наявність в пам'яті обчислювальної машини так званого супервізора, що визначав правила і порядок боротьби програм-суперників. Ці програми мали функції дослідження простору, розмноження і знищення. Сенс гри полягав у видаленні всіх копій програми супротивника і захопленні поля битви.

Наприкінці 60-х – початку 70-х років минулого століття з'явилися програми, які отримали назву **«Кролик»** (the rabbit). Ці програми клонували себе, займали системні ресурси і таким чином знижували продуктивність системи. Ці програми не передавалися від системи до системи, а були суто місцевим явищем – помилками або жартами системних програмістів, що обслуговували комп'ютер.

У **першій половині 70-х років** під операційну систему Tenex був створений вірус **The Creeper**, для поширення якого були використані глобальні комп'ютерні мережі. Вірус міг самостійно увійти до мережі через модем і передати свою копію віддаленій системі.

На **початку 80-х років** з’явилася велика кількість різноманітних «троянських коней» – програм, які при запуску завдавали системі будь-якої шкоди.

У **1981 р.** сталася епідемія завантажувального вірусу **Elk Cloner** на комп'ютерах Apple II. Вірус записувався у завантажувальні сектори дискет, до яких йшло звернення. Проявляв він себе по-різному – перевертав екран, примушував миготіти текст на екрані і виводив різноманітні повідомлення.

Необхідно зазначити, що до **1984 р.** всі шкідливі програми не мали назви комп'ютерного вірусу. **11 листопада 1983 року** американський студент з Університету Південної Каліфорнії Фред Коен написав програму, яка демонструвала можливість зараження комп'ютера вірусом зі

швидкістю від 5 хвилин до однієї години. Наступного року Фред Коен написав наукову роботу з цієї тематики, в якій навів визначення комп'ютерного вірусу. Можна вважати, що 11 листопада 1983 року є днем народження першого комп'ютерного вірусу.

Пандемія першого IBM-PC вірусу **Brain** сталася у **1986 р.** Вірус, що заражав 360 Кб-дискети, практично миттєво розійшовся по всьому світу. Вірус було створено в Пакистані братами Basit і Amjad Farooq Alvi, що залишили у вірусі текстове повідомлення, яке містило їх імена, адресу і телефонний номер. Як стверджували автори вірусу, вони були власниками компанії з продажу програмних продуктів і вирішили з'ясувати рівень піратського копіювання в своїй країні. На жаль, їх експеримент вийшов за межі Пакистану. Цікаво, що вірус **Brain** був також і першим **стелс-вірусом** – при спробі читання зараженого сектора він підставляв його незаражений оригінал.

У грудні **1987 р.** відбулася перша відома поголовна епідемія мережного вірусу **Cristmas Tree.** 9-го грудня вірус був запущений у мережу Bitnet в одному з університетів Західної Німеччини, проник через шлюз в European Academic Research Network (EARN) і потім – в мережу IBM VNet. Через чотири дні (13 грудня) вірус паралізував мережу – вона була забита його копіями. Під час запуску вірусу на екрані з'являлося зображення новорічної (вірніше, різдвяної) ялиночки і копії розсилалися всім користувачам мережі.

У **п'ятницю 13 травня 1988 р.** відразу кілька фірм і університетів декількох країн світу «познайомилися» з вірусом **Jerusalem.** Цього дня вірус знищив файли при запуску. Назву вірус отримав за місцем одного з інцидентів – університету в місті Єрусалимі.

У листопаді **1988 р.** вірус **«Черв'як Моріса»** «зламав» весь тодішній Інтернет. Він паралізував роботу всього Інтернету, що обернулося прямими і непрямими збитками на загальну суму 96 мільйонів доларів.

Вірус **Datacrime** мав вкрай небезпечний прояв – **з 13 жовтня до 31 грудня 1989 року** він форматував жорсткий диск (вінчестер). Цей вірус вирвався «на свободу» і спричинив великий резонанс в засобах масової інформації, особливо в Голландії і Великобританії.

У **1992 р.** вірус **Michelangelo** став стимулом для розвитку антивірусного програмного забезпечення. Проникнувши через дискети у завантажувальні сектори комп'ютера, він був бездіяльним, і лише 6 березня (день народження Мікеланджело) активізувався і почав стирати дані.

Вірус **Win95.CIH** у **1998 р.** вивів з ладу до 500 тисяч комп'ютерів. Він був розроблений тайванським студентом, CIH – його ініціали. Для проникнення вірусу в комп'ютери були використані всі способи. Він уміло «ховався» серед файлів інших програм і ніяк себе не виявляв. Датою, коли починав діяти вірус, було **26 квітня** кожного року – дата аварії на Чорнобильській АЕС, за що його прозвали ще «**Чорнобиль**». **Win95.CIH**

не просто форматував дані, а й стирав вміст BIOS, після цього комп'ютер не можна було включити.

26 березня 1999 р. вірус **Melissa** атакував поштові сервіси. Проникнувши в комп'ютер, вірус розшукував файли додатку MS Outlook і самовільно розсилав свою копію першим 50-ти адресатам із списку контактів. Швидкість розповсюдження виявилася неймовірно високою. Розсилання велося від імені власника зараженого комп'ютера, але сам він про це і не підозрював. Сумарний збиток оцінили у 100 мільйонів доларів.

У ніч на **5 травня 2000 р.** вірус **I Love You** також відомий як **Loveletter**, **The Love Bug** або «**Романтик**» був розісланий електронною поштою з Філіппін. При відкритті вірус розсилав свою копію за всіма контактами в адресній книзі Microsoft Outlook. Також він повторно записував особисті файли, включаючи документи, зображення і аудіо. Тільки у США кількість атакованих користувачів становила більше 2,5 мільйонів. Крім того, він ще привласнював секретні паролі, що збільшувало завдані збитки приблизно до 5,5 мільярдів доларів.

Вірус **Nimda** (**Admin** в зворотному порядку) з'явився у **2001 р.** Потрапляючи в комп'ютер, він отримував права адміністратора, змінював і порушував конструкцію сайтів, блокував доступ на хости, IP-адреси і т. д. Вірус був створений в Китаї.

Появою вірусу **SoBig** вважається **9 січня 2003 р.** Він поширювався через електронну пошту як вірусний спам. Користувач самостійно запускав заражений файл, який значно уповільнював роботу комп'ютера, і пересилав його копію далі поштою. **SoBig** поширився у сотнях тисяч комп'ютерів.

Вірус **MyDoom** («**Моя загибель**») став однією із самих руйнівних програм за всю історію створення вірусів. Епідемія почалася **26 січня 2004 р.**, коли вірус почав дуже швидко поширюватися в мережі Інтернет: кожен наступний комп'ютер відправляв спаму більше, ніж попередній. **MyDoom** видаляв деякі файли з жорсткого диска і організовував DoS-атаки на різні сайти. Крім того, він модифікував операційну систему, блокуючи доступ до сайтів багатьох антивірусних компаній, стрічок новин і різних розділів сайту компанії Microsoft. Антивірусна компанія MessageLabs до **4 лютого 2004 р.** зареєструвала понад 21 мільйон інфікованих листів.

Вірус **BlackEnergy 2** був створений на базі простої троянської програми **BlackEnergy**, яка застосовувалася зловмисниками з **2007 р.** для проведення DDoS-атак. Однак нова версія була значно модернізована і мала набір інструментів для самих різних деструктивних задач. Наприклад, програма могла знищити жорсткий диск, перезаписавши всю інформацію на ньому випадковим масивом даних.

Вірус **Conficker** був розроблений у **2008 р.** спеціально для оперативної системи Microsoft Windows. Використовуючи вразливості операційної системи, він був не виявленим в антивірусних програмах і блокував доступ до оновлення їх баз, відключав оновлення операційної

системи і підміняв назви служб. У світі **Conficker** поширився у 12 мільйонах комп'ютерів.

У **2010 р.** був створений перший вірус **Win32/Stuxnet** для промислових систем (спочатку він був розроблений для операційної системи Windows), який заражав не тільки комп'ютер користувача, а й промислові автоматизовані системи. Вірус по суті був призначений для диверсій і шпигунства, а у разі потреби він міг руйнувати заражену систему, віддаючи нездійсненні команди вузлам. **Win32/Stuxnet** виявляв і перехоплював потоки даних між контролерами Simatic S7 і робочими станціями SCADA-системи Simatic WINCC, розробленими компанією Siemens, підміняв значення даних, вносячи спотворення у роботу автоматизованих систем.

Вірус **Flame** (за відкритими даними) був створений фахівцями військових відомств США та Ізраїлю у **2012 р.** у межах програми «Олімпійські ігри». Цей вірус здатний збирати інформацію, змінювати параметри комп'ютера, робити знімки екрана, записувати звук, підключатися до чатів.

Складна і важко вловима шкідлива програма **DarkTequila** активна з **2013 р.** і спрямована головним чином на порушення роботи користувачів в Мексиці. Вірус створювався з метою викрадання фінансової інформації користувачів, а також логінів і паролей для входу на популярні веб-сайти. Шкідлива програма поширювалась через листи від шахраїв і USB-пристрої.

24 червня 2016 р. була виявлена нова версія шкідливого програмного забезпечення **Godless** (**«Безбожник»**), яка вразила більше 850 тисяч пристроїв. **Godless** здатний заражати пристрої з версією операційної системи Android 5.1 Lollipop і нижчих версій. Програма відкриває доступ до root (прав суперкористувача), завдяки чому може встановлювати нові компоненти, стежити за користувачем і виконувати будь-які дії. Поширюється **Godless** як складова багатьох застосувань з магазина Google Play. Її вдалося виявити як в копіях популярних ігор, так і в простих програмах, таких, як «ліхтарик».

12 травня 2017 р. з використанням вірусу **WannaCry** відбулася масштабна хакерська атака щонайменше в 150 країнах світу, були заражені до 300 тисяч комп'ютерів. Вірус блокував роботу безлічі організацій і підприємств: лікарень, аеропортів, банків, заводів. На комп'ютер встановлювався банер, який блокував доступ до даних (шифрував усі файли користувача). За відновлення даних (розшифровку) вимагали викуп – 300 доларів, або 600 біткоїнів, інакше вірус видаляв файли протягом трьох днів. Загальний збиток від масштабної атаки з використанням вірусу **WannaCry** становив більше одного мільярда доларів. **WannaCry** продовжує періодично заражати комп'ютери, хоча вже і в менших масштабах.

27 червня 2017 р. була здійснена масштабна хакерська атака з використанням вірусу **Petya**. Спочатку його класифікували як вірус-вимагач, але він мав більшу загрозу, ніж вірус **WannaCry**. Хакери вимагали викуп від користувачів заражених комп'ютерів, щоб розблоковувати особисті дані, але цього разу вірус не шифрував їх, він стирав жорсткий диск цілком, не залишаючи можливості врятувати інформацію. Експерти вважають, що зараження почалося з бухгалтерської програми M.E.Doc, якою активно користуються в Україні.

27 лютого 2018 р. фахівці виявили в мережі новий вірус-вимагач масової поразки **Data Keeper**. Програмним забезпеченням шифруються файли на комп'ютерах жертв, вимагаючи викуп в криптовалюті. Вірус не міняє розширення файлів, тому користувачі не знають, які з них заражені. Крім того, **Data Keeper** може обчислити і зашифрувати всі загальні мережі, які дістав через комп'ютер жертви.

16 травня 2018 р. було розкрито вірус-блокувальник **StalinLocker,** або **StalinScreamer**, що використовував образ Сталіна. Вірус видаляв дані з комп'ютера, якщо користувач не вводив правильний код. Унікальність вірусу-блокувальника полягала в тому, що він створював механізм свого автозапуску і намагався припинити решту процесів комп'ютера, зокрема, завершуючи роботу провідника. При запуску **StalinLocker** на екрані з’являлось зображення радянського керівника Йосипа Сталіна і лунав гімн СРСР.

15 жовтня 2018 р. було виявлено вірус **GPlayed,** здатний викрадати особисті дані користувачів Android. Троянську програму було замасковано під офіційний додаток Google Play. При скачуванні додатку на екрані з'являється значок з підписом Google Play Marketplace. Вірус дозволяє видалено керувати пристроєм. Зловмисники можуть здійснювати дзвінки і відправляти повідомлення, отримувати платіжні дані або зовсім заблоковувати пристрій.

З **11 вересня 2018 р.** хакери розіслали більше 11 тисяч листів з поштових адрес, підроблених під держустанови, які містили троянську програму **RTM**, призначену для переведення грошей із сервісів дистанційного банківського обслуговування і платіжних систем. Шкідливе розсилання здійснювалося на адреси фінансових установ і підприємств.

18 грудня 2018 р. розкрито вірус **Andr/Clickr-AD**, що вражав пристрої на базі платформи Android. Користувачі отримували троянську програму з магазина Google Play. Вона здатна генерувати постійний перехід за рекламними посиланнями незалежно від бажання власника гаджету. В зв'язку з цим шкідлива програма призводить до дуже швидкого розряджання смартфонів і планшетів.

10 червня 2019 р. було виявлено, що хакери вносили віруси **Triada** в смартфони з операційною системою Android ще до їх продажу шляхом розташування в них реклами і розсилали спам незалежно від бажання власника.

24 червня 2019 р. була здійснена масштабна атака вірусу-шифрувальника **Shade**, також відомого як **Troldesh**, **XTBL**, **Trojan.Encoder.858**, **Da Vinci**, **No_more_ransome**. Зловмисники відправляли листи від імені співробітників крупних авіакомпаній, автоділерів і засобів масової інформації.

За **перше півріччя 2019 р.** значно збільшилося застосування троянських програм, які **викрадають паролі.** Ці шкідливі програми вміють добувати інформацію безпосередньо із браузерів, зокрема логіни і паролі до акаунтів, а також збережені дані платіжних карток і вміст форм для автозаповнення. Деякі віруси уміють також добувати «куки», тобто файли, в які браузер вносить дані із сайтів, які відвідував користувач, і завдяки яким ресурси «запам'ятовують» логіни і паролі відвідувачів. Існують також шкідливі програми, які здатні викрадати файли, що зберігаються безпосередньо на пристрої. Частіше за все такі троянські програми намагалися проникнути на пристрої користувачів в Росії, Індії, Бразилії, Німеччині і США. При цьому найбільш поширеною шкідливою програмою, що викрадає паролі, стала **AZORult**.

Необхідно зазначити, що кількість атак з використанням шкідливого мобільного програмного забезпечення у всьому світі за останній рік збільшилась практично вдвічі – майже до 117 мільйонів. Головною загрозою виявилися **мобільні банківські троянські програми** (одними з найбільш поширених були **Asacub** і **Hqwar**. Ці шкідливі програми атакували 1,8 мільйона користувачів у 180 країнах світу.

Запитання для самоперевірки

1. Як поділяються шкідливі програми за способом поширення ?
2. У чому полягає принципова відмінність між троянськими програмами і вірусами ?
3. Які існують непрямі ознаки зараження комп'ютера шкідливими програмами ?
4. Які дії можуть здійснювати шкідливі програми-вимагачі ?
5. У чому полягали особливості розвитку шкідливих програм на початковому етапі формування комп’ютерних систем ?
6. Які негативні дії можуть створювати шкідливі програми в комп’ютерних системах ?
7. Які існують тенденції розвитку шкідливих програм ?

Лекція № 4

СПОСОБИ ПРОНИКНЕННЯ ШКІДЛИВИХ ПРОГРАМ

Навчальні цілі:
- вивчити типи вірусів та способи їх проникнення у програми;
- розглянути шкідливі дії троянських програм та мережних черв'яків.

Навчальні питання:
1. Типи вірусів.
2. Шкідливі дії троянських програм.
3. Мережні черв'яки.

4.1 Типи вірусів

Найбільш поширеними є три типи вірусів:
- ті, що заражають завантажувальний сектор;
- ті, що заражають програми;
- макроси (макровіруси).

Віруси, що заражають завантажувальний сектор (Boot sector) жорсткого диска комп'ютера, з'являються, коли комп'ютер намагається завантажити операційну систему із жорсткого диска. Він виконує програму, записану в завантажувальному секторі, і разом з цією програмою запускається вірус. Після цього вірус, знаходячись у пам'яті комп'ютера, виконує шкідливі задачі. Прикладом вірусу такого типу є вірус **Stoned**, який після запуску як би «заморожує» комп'ютер, він перестає реагувати на клавіатуру або мишку.

Віруси, що заражають програми. До таких вірусів найчастіше належать віруси у резидентних програмах, які завантажуються одного разу і постійно знаходяться в оперативній пам'яті комп'ютера, наприклад, драйвери. Потрапивши у пам'ять комп'ютера і ставши резидентним, вірус записується на файли, що мають розширення .com, .exe, .ovr і на файли драйверів. Нерезидентні програми, тобто ті, які після виконання звільняють займану оперативну пам'ять, також можуть бути заражені вірусом і під час їх виконання вірус «запускається» і виконує шкідливі дії. Прикладом вірусу такого типу може бути вірус **«Чорнобиль».**

Макровіруси, або **макроси** заражають документи текстового редактора Word. Такі віруси не залежать від операційної системи, вони заражають файли з конкретним розширенням, наприклад, .doc. Прикладом вірусу такого типу є вірус **Melissa**.

Розглянемо спочатку способи впровадження вірусів у файли з розширенням .com (COM-файли).

На початку COM-файла зазвичай знаходиться команда безумовного переходу JMP, що складається з трьох байтів. Перший байт містить код команди 0E9h, наступні два – адресу переходу.

На рисунку 4.1 показано етапи впровадження вірусу у COM-файл. На першому етапі тіло вірусу записується у кінець файла, у кінець файла також переноситься оригінальний JMP, а на його місце записується інструкція JMP, в якій вказано тіло вірусу. Внаслідок цього вірус отримує керування програмою (другий етап роботи). Після виконання шкідливих дій вірус відновлює оригінальний JMP (третій етап), який передає керування на початок програми (четвертий етап).

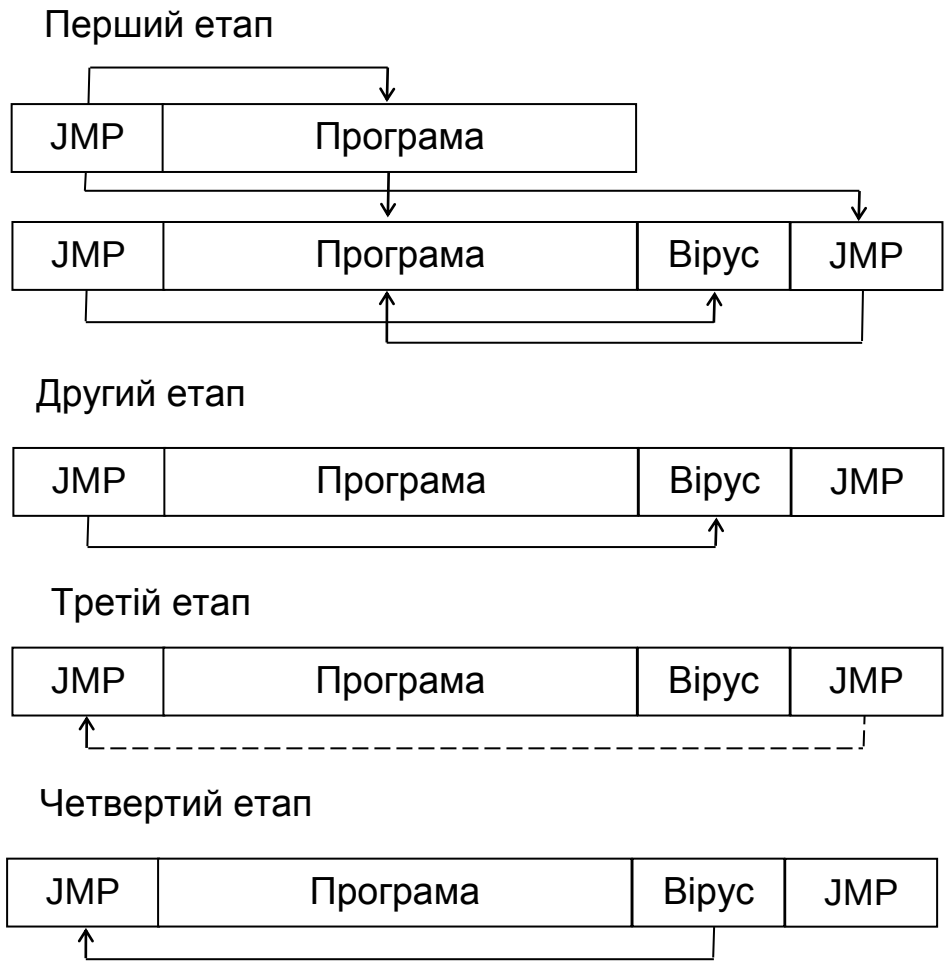

Рисунок 4.1 – Зараження COM-файла вірусом

Таким чином, у розглянутому способі впровадження вірусу здійснюється дописування вірусу у кінець файла, а у початок файла записується команда стосовно запуску вірусу. Існують й інші способи впровадження вірусів, наприклад, на початок файла.

Розглянемо два варіанти впровадження вірусу у початок COM-файла. Етапи першого варіанта впровадження зображено на рисунку 4.2.

На першому етапі початок програми переписується в кінець файла і таким чином звільняється місце для вірусу. Тіло вірусу записується у

початок файла, а його частина, що відповідає за відновлення програми, – у кінець файла. Після виконання шкідливих дій вірус передає керування своєму коду у кінці програми (другий етап). На третьому етапі здійснюється відновлення програми до початкового стану. Після цього запускається виконання програми (четвертий етап).

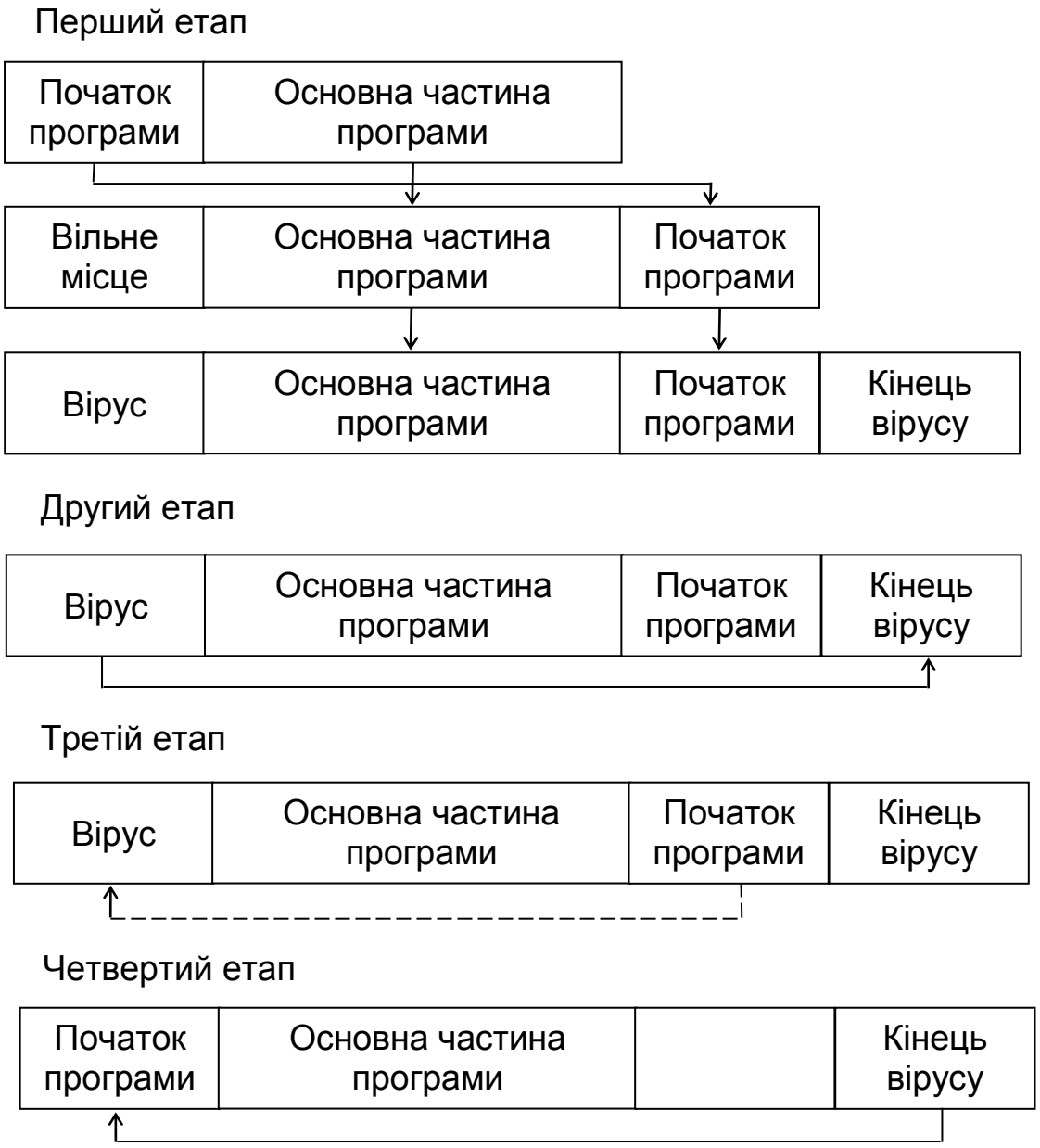

Рисунок 4.2 – Перший варіант розміщення вірусу у початок файла

Другий варіант розміщення вірусу у початок файла відрізняється від першого тим, що вірус, звільняючи для себе місце, зрушує все тіло програми, а не тільки початок файла (рисунок 4.3). На першому етапі тіло програми зсувається ближче до кінця файла, таким чином звільняється місце для вірусу. Тіло вірусу записується у початок файла, а його частина, що відповідає за відновлення програми, – у кінець файла. Після виконання шкідливих дій вірус передає керування своєму коду у кінці програми

(другий етап). На третьому етапі здійснюється відновлення програми до початкового стану – тіло програми зсувається за початковою адресою. Після цього вірус запускає програму на виконання (четвертий етап).

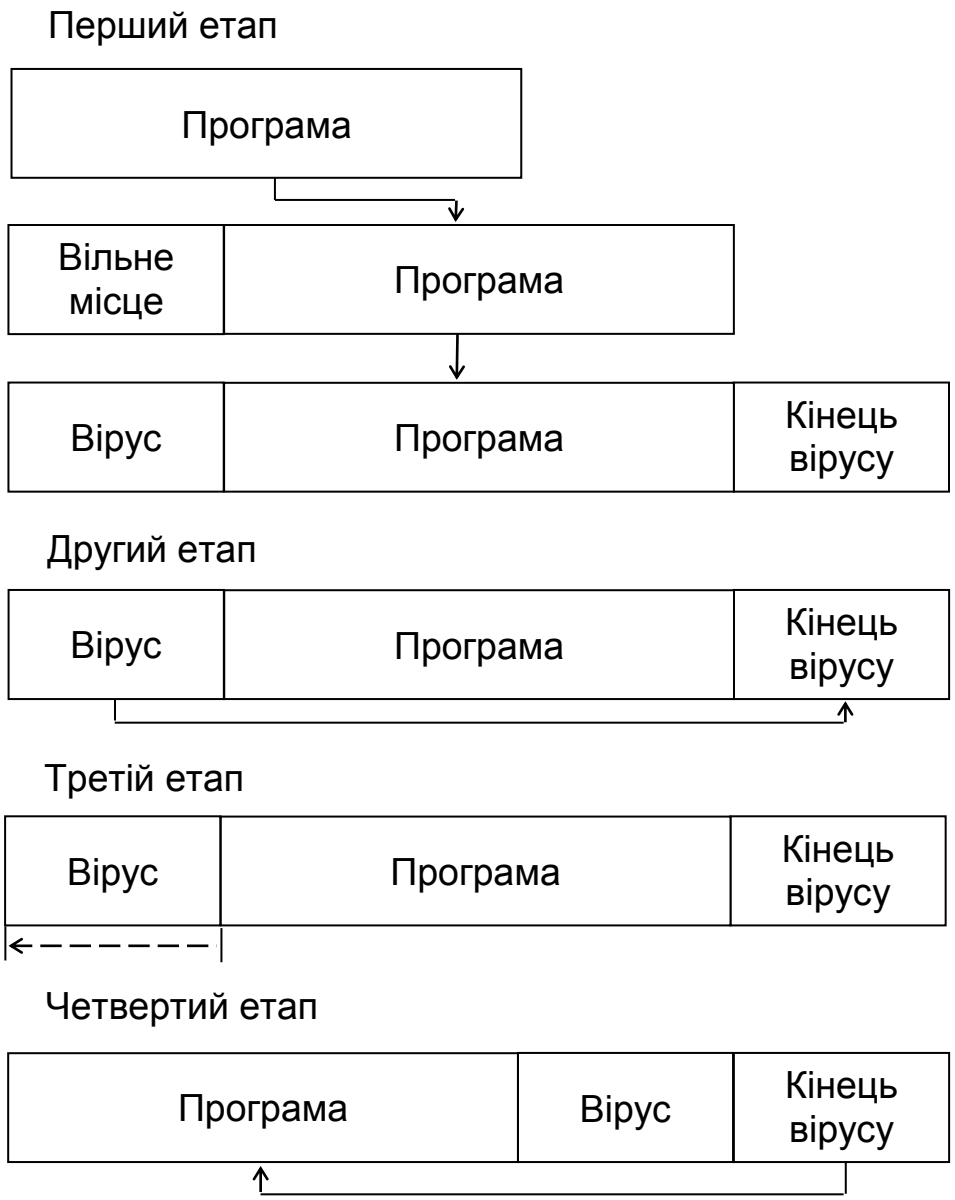

Рисунок 4.3 – Другий варіант розміщення вірусу у початок файла

Існують декілька типів вірусів, що заражають файли, які виконуються, з розширенням .exe (EXE-файли):

- ті, що заміщають програмний код;
- віруси-супутники;
- ті, що впроваджуються у програму.

Віруси, що заміщають програмний код, записуються поверх програмного коду, не зберігаючи його, тому інфіковані програми не виконуються. Під час запуску вірус шукає чергову жертву (або жертви),

відкриває знайдений файл для редагування і записує своє тіло у початок програми, не зберігаючи оригінальний код.

Віруси-супутники отримали свою назву завдяки алгоритму розмноження: до кожного інфікованого файла створюється додатковий файл-супутник.

Віруси, що впроваджуються у програму, записують свій код у програму, яка інфікується, а самі залишаються непомітними, що істотно ускладнює виявлення і «лікування» заражених файлів.

Існує декілька способів впровадження вірусів у файли з розширенням .exe, коротко розглянемо їх суть.

Під час використання першого способу тіло вірусу дописується у кінець файла, а заголовок корегується (зі збереженням оригінального) так, щоб при запуску інфікованого файла керування отримував вірус. Цей спосіб схожий із способом зараження COM-файлів, але замість завдання у коді переходу на початок вірусу корегується власна адреса точки запуску програми.

Другий метод має назву «перенесення». Під час запуску інфікованої програми тіло вірусу з неї зчитується в оперативну пам'ять. Потім ведеться пошук неінфікованої програми. В оперативну пам'ять зчитується початок програми, довжина якого дорівнює довжині тіла вірусу. На це місце записується тіло вірусу. Початок програми з оперативної пам'яті дописується у кінець файла.

Ще одним методом впровадження вірусу у початок файла є зсув коду програми. При цьому тіло програми, що інфікується, зчитується в оперативну пам'ять, на її місце записується вірусний код, а після нього – код програми, що інфікується. Таким чином, код програми як би «зсувається» у файлі на довжину коду вірусу.

4.2 Шкідливі дії троянських програм

За шкідливими виконуваними діями троянські програми можна умовно поділити на такі види:

- **крадіжка паролів.** У наш час більшість сервісів у комп’ютерних системах захищена паролями. Є паролі для зв'язку з мережею Інтернет, поштою, форумами, чатами, ICQ та ін. Хоча користувачі не зазнають прямих збитків, але наслідки, пов'язані з отриманням зловмисниками доступу до конфіденційної інформації, можуть виявитися дуже серйозними;

- **викрадення акаунтів (облікових записів).** У більшості випадків вони не блокуються, і людина може працювати на сайті, не помічаючи ніяких змін, але в цей же час від її імені іншим людям приходитимуть рекламні повідомлення. Особливо актуальною така загроза є для соціальних мереж. Шахраї мають вигоду вже від факту використання

чужого комп'ютера, тому в цьому випадку не вимагають від його власника грошової компенсації;

- **фішинг.** У багатьох випадках від користувача вимагають перейти за посиланням на підставний сайт і ввести свої облікові дані, які потім застосовуються для розсилки спаму. Як правило, на поштову скриньку приходить лист з повідомленням про злом системи захисту сайту. Щоб зберегти свій обліковий запис, вам нібито необхідно поміняти пароль, для чого пропонується зайти за посиланням на підставний ресурс, який зовні є повною копією оригіналу. Доменне ім'я зазвичай відрізняється всього на одну букву і це не відразу видно (наприклад, facedook.com замість facebook.com). Користувач реєструється і отримує повідомлення про успішну зміну пароля, а іноді і напис про те, що на сервісі ведуться технічні роботи і спробу слід повторити пізніше. Насправді облікові дані вже відправлені на комп'ютер хакерів;

- **віддалене адміністрування**. Програми цього класу аналогічні професійним утилітам віддаленого адміністрування, але встановлюються без згоди користувача і дозволяють зловмисникові тримати комп'ютер під повним контролем. Контроль над чужим комп'ютером дозволяє створювати ботнет-мережі, до складу яких іноді входять сотні тисяч комп'ютерів. Такі віртуальні армії формуються для розсилки спаму або DDoS-атак на сайти. Користувачі часто навіть не підозрюють про те, що їх комп'ютером керує хтось інший;

- **розсилка спаму**. Після проникнення у комп'ютер користувача троянська програма починає розсилати спам на заздалегідь задані адреси. Хоча від такої поведінки троянської програми страждає більше не сам користувач, а мільйони одержувачів небажаних листів з його комп'ютера, але і для користувача наслідки можуть бути неприємними (наприклад, блокування IP-адреси у більшості поштових систем);

- **proxy-сервери**. Троянська програма створює у комп'ютері один або декілька видів проксі-серверів (Socks, HTTP та ін.), за допомогою яких зловмисник може здійснювати будь-які дії в мережі Інтернет, не побоюючись виявлення дійсної IP-адреси, оскільки замість його адреси використовується адреса жертви;

- **шпигунські програми**. Програми цього класу збирають відомості із комп'ютера користувача (це може бути листування, всі клавіші, що натискають, відвідувані сторінки, встановлені програми і багато чого іншого) і пересилають їх за адресою, указаною у троянській програмі. Для отримання даних зловмисники можуть використовувати кейлогери для «зчитування» інформації з натиснутих клавіш і відправляють її шахраям;

- **програми DDoS-атаки**. Заражені такою троянською програмою комп'ютери беруть участь у DDoS-атаках, що спричиняють перевантаження атакованого сервера. З урахуванням того, на що спрямовані атаки, все виглядає так, немов один з нападаючих – це комп'ютер жертви;

- **програми розподілених обчислень**. Ці програми можна назвати одним з «найінтелігентніших» класів троянських програм. Розподілені обчислення можуть використовуватися і для менш пристойних справ: для пошуку сервісів в мережі Інтернет (наприклад, проксі-серверів), а також для підбору паролів;

- **рекламні модулі та модулі накручування реклами**. Троянські програми можуть демонструвати користувачеві комп'ютера різну рекламну інформацію, наприклад, вікна, що спливають, банери, які вбудовують у системні панелі, сторінки, що проглядаються. Інший варіант застосування зараженого комп'ютера у рекламних цілях – накручування банерних систем шляхом імітації заходів користувача на одержання ресурсу, де розміщена реклама;

- **встановлення додаткових модулів, що чекають команд від свого автора**. Такі троянські програми завантажують на заражений комп'ютер файли із запрограмованої адреси або чекають отримання команди від автора (каналом зв'язку може бути одержання електронного листа, поява повідомлення на форумі або сайті й т. д.) на здійснення яких-небудь дій (перелічених вище або будь-яких інших).

Цікаво, що деякі троянські програми не терплять конкуренції і видаляють під час установлення виявлені модулі інших «троянців».

Небезпека троянських програм полягає у такому:

- небезпеці втрати конфіденційної інформації;
- уповільненні роботи системи або взагалі в її припиненні;
- мимовільному витрачанні Інтернет-трафіку, причому нерідко у дуже великих об'ємах;
- малоприємному спілкуванні з правоохоронними органами або службами безпеки крупних організацій у тому випадку, якщо з особистої IP-адреси відбудеться зламування або DDoS-атака якого-небудь сервера.

До основних способів проникнення троянських програм у комп'ютер належать:

- **установлення її самим користувачем**, **який вважає, що перед ним потрібна програма.** Особливо часто троянські програми зустрічаються в архівах на сайтах, що поширюють зламане і неліцензійне програмне забезпечення, а також на хакерських сайтах під виглядом дистрибутивів програм, генераторів серійних номерів або хакерських утиліт. Щоб переконати власника комп'ютера особисто запустити шкідливу програму, її, як водиться, видають за якесь корисне програмне забезпечення, наприклад, критичне оновлення для Windows, антивірус, кодек, необхідний для перегляду відео на сайті і т. д. Вони також можуть поширюватися в креках і генераторах ключів;

- **троянські функції, що бувають вбудованими у ліцензійне програмне забезпечення.** Зафіксовано випадки зламування сайтів виробників програмного забезпечення, коли в архіви з програмами, що поширюються на цих сайтах, впроваджувалися троянські програми;

- **розсилка з підміною зворотної адреси.** При цьому дуже часто прикриваються назвами і адресами відомих компаній;

- **дірки, виявлені в операційних системах** і прикладних програмах роботи з електронною поштою, а також в Інтернет-браузерах;

- **використання соціального інжинірингу**, коли зловмисник, увійшовши у довіру до користувача, висилає йому троянську програму під виглядом потрібного файла (наприклад, фотографії або прайс-листа). Повідомлення зазвичай приходять в соціальних мережах у чаті, електронній пошті або на форумах. Зловмисники листами або розмовами спонукають користувачів зробити певну дію, яка відключить захист комп'ютера або якимсь іншим чином відкриє доступ до потрібної інформації.

Можливі дії зловмисників:

- використовуючи соціальну інженерію, шахраї просять перераховувати гроші на певний рахунок. Для цього SMS складають так, щоб здавалося, ніби його відправляв хтось з родичів. У ряді випадків зловмисники також спонукають жертву зробити дзвінок на мобільний номер, а потім прагнуть довго зволікати час, оскільки за розмову стягується спеціальна плата за вищим тарифом;

- пропонують програму «SMS-шпигун», нібито яка уміє встановлювати місцеположення людини за номером її мобільного телефону. Щоб скористатися послугою, абонентам радять реєструватися за допомогою SMS. Після цього користувач отримує посилання на сайт із загальнодоступною інформацією про належність того або іншого коду певному операторові зв'язку або на сервіси інтерактивних карт (Google maps або «Яндекс.Карты»). Формально такі дії навіть не є злочином, оскільки де-небудь на сайті вказуються відомості про те, які послуги будуть надані користувачеві;

- за невелику плату пропонують програму, яка нібито вміє читати SMS на будь-якому телефоні після введення потрібного номера;

- приходить SMS з описом маловідомого способу поповнити рахунок без фінансових витрат, для чого необхідно відправити повідомлення на номер.

Приклади деяких троянських програм:

- **Baker.LGC** намагається проникнути на комп'ютер, прикриваючись фальшивою історією про нещасний випадок за участю Фернандо Алонсо, іспанського гонщика Формули 1;

- **Turkojan.I** має одну з найпривабливіших масок: видає себе за новий епізод «Симпсонів»;

- **Philto.A** під маскою відео про Періс Хілтон встановлює на комп'ютер модулі, що демонструють рекламу;

- **Meteorbot.A** під виглядом супермена викрадає інформацію про комп'ютер;

- **Banbra.FXT** генерує поштове повідомлення від імені Бразильського федерального міністерства, а потім знімає гроші з банківського рахунку користувача, який повірив в розіграш.

4.3 Мережні черв’яки

Мережні черв’яки являють собою головну загрозу для всіх користувачів глобальної мережі Інтернет. Майже всі Інтернет-черв'яки – це поштові черв'яки, і лише мала частка – непоштові черв'яки, що застосовують уразливість програмного забезпечення (переважно, серверного).

Прикладами непоштових черв'яків є: **Iis-worm.CodeRed, Iis-worm.CodeBlue, Worm.SQL.Helkern**.

Поштові черв'яки можна поділити на два основних класи:

- черв'яки, які активізуються самі (без відома користувача);

- черв'яки, які активізуються, якщо користувач збереже приєднаний до листа файл і запустить його.

До першого класу належать черв'яки, які використовують уразливість (помилки) поштових клієнтів. Найчастіше такі помилки знаходяться в поштовому клієнті Outlook, а вірніше навіть не в ньому, а в Інтернет-браузері Internet Explorer. MS Outlook створює лист у вигляді HTML-сторінки і при відображенні цих сторінок використовує функції браузера Internet Explorer. При цьому, застосувавши відповідний код, шкідлива програма має можливість під час перегляду листа автоматично зберегти приєднаний до листа файл на диску і запустити його. Компанією Microsoft випущені оновлення для всіх версій браузера Internet Explorer, що виправляють цю помилку. Найбільш поширеними черв'яками цього класу є: **I-worm.Klez, I-worm.Avron, I-worm.Frethem, I-worm.Aliz.**

Поштові черв'яки другого класу розраховані на те, що користувач за якимось міркуваннями сам запустить програму, приєднану до листа. Для спонукання користувача до запуску інфікованого файла авторами черв'яків застосовуються різні психологічні ходи. Найпоширеніший прийом – видати заражений файл за якийсь важливий документ, картинку або корисну програму. Прикладами таких черв'яків є: **I-Worm.LovGate,** який створює відповіді на листи, що містяться в поштовій базі; **I-Worm.Ganda**, що маскується під інформацію про бойові дії в Іраку**.**

Практично завжди в мережних черв'яках застосовуються подвійні розширення. В цьому випадку приєднаний файл має ім'я **Doc1.doc.pif**, **pict.jpg.com**. Цей принцип розраховано на те, що поштові клієнти не відображають повне ім'я файла (якщо воно дуже довге), і користувач не побачить другого розширення, яке і є реальним. Іншими словами, користувач думає, що файл є документом або картинкою, а насправді це є виконуваний файл з розширенням: .exe, .com, .pif, .scr, .bat, .cmd і т. д. Якщо такий файл відкрити, то тіло черв'яка активізується.

Окрім основної функції – розмноження, черв'яки майже завжди мають і шкідливу дію. Вкладені функції надзвичайно різноманітні. Так, наприклад, дуже часто поштові черв'яки покликані для того, щоб встановити на заражений комп'ютер троянську програму або утиліту прихованого адміністрування і повідомити адресу комп'ютера автору черв’яка. Нерідко знищують інформацію або роблять неможливою подальшу роботу на комп'ютері. Так черв'як **I-Worm.Magistr** виконував ті ж дії, що і сумновідомий **WINCIH** – стирав вміст FLASH BIOS і забивав сміттєвими даними інформацію на жорсткому диску.

У будь-якому випадку незалежно від наявності або відсутності шкідливих функцій і їх небезпеки поштові черв'яки є шкідливими вже тільки тому, що вони існують. Це пов'язано з тим, що при розмноженні вони завантажують канали зв'язку і нерідко настільки, що повністю паралізують роботу людини або цілої організації.

Запитання для самоперевірки

1. Які типи вірусів є найбільш поширеними ?
2. Як здійснюється впровадження вірусів у файли з розширенням .com (COM-файли) ?
3. Які існують типи вірусів, що заражають файли з розширенням .exe (EXE-файли) ?
4. За якими видами шкідливих дій поділяють троянські програми ?
5. Які існують основні способи проникнення троянських програм у комп'ютер ?
6. На які основні класи поділяють поштових черв'яків ?
7. Які основні шкідливі дії спричиняють мережні черв'яки ?

Лекція № 5

МЕТОДИ ВИЯВЛЕННЯ ШКІДЛИВИХ ПРОГРАМ

Навчальні цілі:
- розглянути моделі дії програмних закладок;
- вивчити методи виявлення шкідливих програм.

Навчальні питання:
1. Моделі дії програмних закладок.
2. Методи виявлення шкідливих програм.

5.1 Моделі дії програмних закладок

Основні групи деструктивних дій, які можуть здійснюватися програмними закладками:

- копіювання інформації користувача комп'ютерної системи (паролів, криптографічних ключів, коду доступу, конфіденційних електронних документів), що знаходиться в оперативній або зовнішній пам'яті цієї системи або в пам'яті іншої комп'ютерної системи, підключеної до неї через локальну або глобальну комп'ютерну мережу;
- зміна алгоритмів функціонування системних, прикладних і службових програм (наприклад, внесення змін до програми розмежування доступу може призвести до того, що вона дозволить вхід у систему всім без виключення користувачам незалежно від правильності введеного пароля);
- нав'язування певних режимів роботи (наприклад, блокування запису на диск при видаленні інформації, при цьому інформація, яку потрібно видалити, не знищується і може бути згодом скопійована хакером).

Програмні закладки можуть діяти за такими моделями:
- перехоплення;
- спотворення;
- прибирання сміття;
- спостереження і компрометації.

Під час реалізації **моделі перехоплення** програмна закладка впроваджується у постійний запам’ятовуючий пристрій («вінчестер»), системне або прикладне програмне забезпечення і зберігає всю або вибрану інформацію, що вводиться із зовнішніх пристроїв комп'ютерної системи або виводиться на ці пристрої, у прихованій області пам'яті локальної або віддаленої комп'ютерної системи. Об'єктом збереження, наприклад, можуть бути символи, введені з клавіатури (всі повторювані двічі послідовності символів), або електронні документи, що роздруковуються на принтері. Ця модель може бути двоступінчатою. На першому етапі зберігаються тільки, наприклад, імена або початки файлів.

На другому накопичені дані аналізуються зловмисником з метою ухвалення рішення про конкретні об'єкти подальшої атаки.

У **моделі спотворення** програмна закладка змінює інформацію, яка записується у пам'ять комп'ютерної системи в результаті роботи програм, або придушує/ініціює виникнення помилкових ситуацій у комп'ютерній системі.

Статичне спотворення відбувається всього один раз. При цьому модифікуються параметри програмного середовища комп'ютерної системи, щоб згодом в ній виконувалися потрібні зловмисникові дії. Наприклад, у виконуваному EXE-модулі програми перевірки правильності цифрового підпису символьний рядок «Підпис некоректний» виправляється (замінюється) на символьний рядок «Підпис коректний». В результаті взагалі перестають фіксуватися документи з невірними цифровими підписами.

Динамічне спотворення полягає у зміні яких-небудь параметрів системних або прикладних процесів за допомогою заздалегідь активізованих закладок. Наприклад, програмна закладка дозволяє прочитувати тільки перші 512 байтів документу, і в результаті цифровий підпис визначається на основі тільки цих 512 байтів. Така ж схема може діяти і під час перевірки поставленого під документом цифрового підпису. Отже, частина цього документа, що залишилася, може бути довільним способом спотворена, і цифровий підпис під ним продовжує залишатися коректним.

Різновидом спотворення є також модель «Троянський кінь». У цьому випадку програмна закладка вбудовується у постійно використовуване програмне забезпечення і за деякою активізуючою подією спричиняє виникнення збійної ситуації в комп'ютерній системі. Тим самим паралізується її нормальне функціонування, а зловмисник, діставши доступ до комп'ютерної системи, для усунення неполадок, може, наприклад, «витягувати» з неї інформацію, перехоплену іншими програмними закладками. Як активізуюча подія зазвичай використовується настання певного моменту часу або стану деяких лічильників (наприклад, лічильника кількості запусків програми).

Під час реалізації **моделі прибирання сміття** використовуються особливості зберігання інформації на жорсткому диску комп’ютера. Під час зберігання комп'ютерних даних на зовнішніх носіях комп’ютера виділяється декілька рівнів ієрархії: сектори, кластери і файли. **Сектори** є одиницями зберігання інформації на апаратному рівні. **Кластери** складаються з одного або декількох підряд розташованих секторів. **Файл** – це безліч кластерів, об'єднаних за певним законом.

Для захисту конфіденційної інформації зазвичай використовується шифрування. Основна загроза виходить від звичайних текстових редакторів і баз даних, які застосовуються для створення і корекції конфіденційних документів. Подібні програмні засоби, як водиться, в

процесі функціонування створюють в оперативній або зовнішній пам'яті комп'ютерної системи тимчасові копії документів, з якими вони працюють. Всі ці тимчасові файли випадають з поля зору будь-яких програм шифрування і можуть бути використані зловмисником для того, щоб скласти уявлення про зміст конфіденційних документів, що зберігаються у зашифрованому вигляді.

При записі відредагованої інформації меншого обсягу у тому ж файлі, де зберігалася вихідна інформація до початку сеансу її редагування, утворюються так звані «хвостові» кластери, в яких ця вихідна інформація повністю зберігається. І тоді «хвостові» кластери не тільки не піддаються дії програм шифрування, але й залишаються не зачепленими навіть засобами гарантованого стирання інформації. Звичайно, рано чи пізно інформація з «хвостових» кластерів «затирається» даними з інших файлів (через добу можна «витягувати» до 85 % початкової інформації, а через десять діб – до 25 … 40 %).

При використанні **моделі спостереження** програмна закладка вбудовується у мережне програмне забезпечення. Впроваджена в нього програмна закладка може стежити за всіма процесами оброблення інформації у комп'ютерній системі, а також здійснювати встановлення і видалення інших програмних закладок.

Модель **компрометації** дозволяє отримувати доступ до інформації, перехопленої іншими програмними закладками.

Існує декілька непрямих ознак **виявлення програмної закладки** у комп’ютерній системі:

- змінюється склад і довжина файлів;
- старі файли кудись пропадають, а замість них з'являються нові;
- програми починають працювати повільніше або закінчують свою роботу дуже швидко, або взагалі перестають запускатися.

5.2 Методи виявлення шкідливих програм

Основними методами виявлення шкідливих програм є:

- сканування;
- виявлення змін;
- евристичний аналіз;
- використання резидентних сторожів;
- вакцинація програм;
- апаратно-програмний захист від вірусів.

Сканування здійснюється програмою-сканером, яка переглядає файли у пошуках пізнаванної частини вірусу – **сигнатури.** Програма фіксує наявність вже відомих вірусів, за винятком поліморфних, які застосовують шифрування тіла вірусу, змінюючи при цьому кожного разу і сигнатуру. Програми-сканери можуть зберігати не сигнатури відомих вірусів, а тільки їх контрольні суми, а також часто можуть видаляти

виявлені віруси. Метод сканування застосовується для виявлення вірусів, сигнатури яких вже відомі і є постійними. Необхідне регулярне оновлення відомостей про нові віруси.

Метод виявлення змін базується на використанні програм-ревізорів. Ці програми визначають і запам'ятовують характеристики всіх областей на дисках, в яких зазвичай розміщуються віруси. При періодичному застосуванні програм-ревізорів порівнюються характеристики, що зберігаються, і характеристики, що отримуються при контролі областей дисків. За результатами ревізії програма видає відомості про передбачувану наявність вірусів. Зазвичай програми-ревізори запам'ятовують у спеціальних файлах образи головного завантажувального запису, завантажувальних секторів логічних дисків, характеристики всіх контрольованих файлів, каталогів і номери дефектних кластерів. Можуть контролювати також обсяг встановленої оперативної пам'яті, кількість підключених до комп'ютера дисків та їх параметри.

Головною перевагою методу є можливість виявлення вірусів усіх типів, а також нових невідомих вірусів. Сучасні програми-ревізори виявляють навіть стелс-віруси.

Основним недоліком цього методу є те, що за допомогою програм-ревізорів неможливо визначити вірус у файлах, які надходять у систему вже зараженими. Віруси будуть виявлені тільки після розмноження у системі. Також програми-ревізори непридатні для виявлення зараження макровірусами, оскільки документи і таблиці дуже часто змінюються.

Евристичний аналіз дозволяє визначати невідомі віруси, але не потребує попереднього збору, оброблення і зберігання інформації про файлову систему. Суть евристичного аналізу полягає у перевірці можливих місць існування вірусів і виявлення в них команд (груп команд), характерних для вірусів. Такими командами можуть бути команди створення резидентних модулів в оперативній пам'яті, команди прямого звернення до дисків, минаючи операційну систему. Евристичні аналізатори при виявленні підозрілих команд у файлах або завантажувальних секторах видають повідомлення про можливе зараження. Після отримання таких повідомлень необхідно ретельно перевірити ймовірно заражені файли і завантажувальні сектори всіма наявними антивірусними засобами. Евристичний аналізатор є, наприклад, в антивірусній програмі Doctor Web.

Метод використання резидентних сторожів основано на застосуванні програм, які постійно знаходяться в оперативній пам’яті комп’ютера і відстежують всі дії решти програм. У разі виконання якою-небудь програмою підозрілих дій (звернення для запису в завантажувальні сектори, розміщення в оперативній пам’яті резидентних модулів, спроби перехоплення переривань та ін.) резидентний сторож видає повідомлення користувачу. Програма-сторож може завантажувати на виконання інші антивірусні програми для перевірки підозрілих програм, а також для контролю всіх файлів, що надходять ззовні (зі змінних дисків, за мережею).

Істотним недоліком цього методу є значний відсоток помилкових тривог, що заважають роботі користувача, спричиняють роздратування і відмовлення від використання резидентних сторожів.

Під вакцинацією програм розуміють створення спеціального модуля для контролю її цілісності. Як характеристика цілісності файла зазвичай використовується контрольна сума. При зараженні вакцинованого файла модуль контролю виявляє зміну контрольної суми і повідомляє про це користувача. Метод дозволяє виявляти всі віруси, у тому числі й незнайомі, за винятком стелс-вірусів.

Апаратно-програмні антивірусні засоби. В наш час для захисту комп'ютерів також використовують спеціальні контролери і їх програмне забезпечення. Контролер встановлюється у рознім розширення і має доступ до загальної шини. Це дозволяє йому контролювати всі звернення до дискової системи. У програмному забезпеченні контролера запам'ятовуються області на дисках, зміна яких у звичайних режимах роботи не допускається. Таким чином, можна встановити захист на зміну головного завантажувального запису, завантажувальних секторів, файлів конфігурації, виконавчих файлів та ін. При виконанні заборонених дій будь-якою програмою контролер видає відповідне повідомлення користувачу і блокує роботу комп'ютера.

Основною перевагою цього методу є те, що контролери працюють постійно і виявляють всі віруси незалежно від механізму їх дії, блокують недозволені дії, які є результатом роботи вірусу або некваліфікованого користувача. Однак цей метод має велику залежність від апаратних засобів комп'ютера. Зміна останніх веде до необхідності заміни контролера.

До основних правил зниження ризику зараження комп'ютера належать:

- не працювати в системі з правами адміністратора. Бажано працювати з обмеженими правами, а для запуску програм, що потребують більших прав, використовувати пункт «Запустити від імені» у контекстному меню;
- не завантажувати програми з неперевірених джерел – перш за все це стосується сайтів, що поширюють зламане, неліцензійне програмне забезпечення і хакерські утиліти (слід використовувати програмні продукти, отримані законним офіційним шляхом);
- не допускати до свого комп'ютера сторонніх, якщо це можливо;
- регулярно робити знімки для відновлення системи і резервні копії важливої інформації та файлів (дублювання інформації);
- користуватися малопоширеними програмами для роботи у мережі або не тими, що встановлені за умовчанням;
- не запускати програми, отримані від невідомих осіб;
- регулярно використовувати антивірусні засоби (достатньо безкоштовних AVG, avast!, Avira або Microsoft Security Essentials);

- користуватися нестандартними брандмауерами, навіть не найкращими за наслідками тестувань, оскільки зловмисник, як правило, не застосовуватиме засоби для обходу всіх існуючих брандмауерів, обмежившись декількома найпопулярнішими;

- перейменовувати виконувані файли антивірусів і брандмауерів, а також сервіси, використовувані ними, а за наявності відповідних навиків – змінювати заголовки їх вікон;

- робити знімки файлів у системних директоріях, а при появі нових спробувати визначати, що це за файли і звідки вони з'являються, або застосовувати спеціальні програми – ревізори диска, які дозволяють виявити нові підозрілі файли, а також зміну розміру тих, що існують;

- включати показ на комп'ютері всіх розширень файлів і уважно стежити за його повним ім'ям;

- регулярно встановлювати «латки» для операційної системи і використовуваних програм;

- не дозволяти браузеру запам'ятовувати паролі і зберігати їх у слабо захищених програмах зберігання паролів. Якщо зручніше не запам'ятовувати паролі, а зберігати їх у комп'ютері, слід подумати над установленням програми, яка зберігає записи, що вводяться в неї, у зашифрованому вигляді;

- особливу обережність слід виявляти при використанні нових знімних носіїв інформації і нових файлів. Нові знімні носії обов'язково мають бути перевірені на відсутність завантажувальних і файлових вірусів, а отримані файли – на наявність файлових вірусів;

- при роботі у розподілених системах або в системах колективного користування доцільно нові змінні носії інформації і файли, що вводяться в систему, перевіряти на спеціально виділених для цієї мети комп'ютерах;

- якщо не передбачається здійснювати запис інформації на носій, то необхідно блокувати виконання цієї операції;

- не погоджуватися на інсталяцію супутнього програмного забезпечення, запропонованого сайтом, якщо на 100 % немає впевненості в його необхідності.

Запитання для самоперевірки

1. Які деструктивні дії можуть здійснювати програмні закладки ?
2. Які існують моделі дії програмних закладок ?
3. У чому полягає різниця між статичною і динамічною моделями спотворення ?
4. Які особливості зберігання інформації на жорсткому диску комп'ютера використовуються у моделі прибирання сміття ?
5. Які існують методи виявлення шкідливих програм ?
6. У чому полягає суть евристичного аналізу ?
7. Які існують основні правила зниження ризику зараження комп'ютера ?

Лекція № 6

ПРОБЛЕМИ БЕЗПЕКИ У МЕРЕЖІ ІНТЕРНЕТ

Навчальні цілі:
- розглянути внутрішню організацію мережі Інтернет;
- вивчити основні проблеми безпеки у мережі Інтернет.

Навчальні питання:
1. Огляд внутрішньої структури TCP/IP.
2. Проблеми, пов’язані з безпекою в мережі Інтернет.

6.1 Огляд внутрішньої структури TCP/IP

Інтернет – це об'єднання у масштабі всієї планети групи мереж (всесвітня мережа), що використовують єдиний протокол для передачі даних (стек протоколів TCP/IP).

Принципи роботи інтернет-протоколів TCP/IP за своєю суттю нагадують роботу пошти. Спочатку на папері пишуть лист, його кладуть в конверт, на зворотному боці конверта пишуть адреси відправника і одержувача, а потім його відносять в найближче поштове відділення. Далі лист проходить через декілька поштових відділень до найближчого поштового відділення одержувача, звідки доставляється за вказаною адресою одержувача і опускається в його поштову скриньку (з номером квартири) або вручається особисто. Коли одержувач листа захоче відповісти, то він в своєму листі поміняє місцями адреси одержувача і відправника, і лист попрямує у зворотному напрямі.

Кожен комп'ютер (вузол, хост) у мережі Інтернет теж має унікальну адресу, яка називається **IP-адреса** (Internet Protocol Address), наприклад, 195.34.32.116. IP-адреса складається з чотирьох десяткових чисел (від 0 до 255), розділених крапкою (аналог номера будинку, за яким доставляється лист). Однак знати тільки IP-адресу комп'ютера ще недостатньо, оскільки обмінюються інформацією не комп'ютери самі собою, а додатки, що працюють в них. У комп'ютері може одночасно працювати декілька додатків (наприклад, поштовий сервер, веб-сервер-сервіс та ін.). Кожний програмний додаток має номер, що називається **номером порту** (аналог номера квартири, за яким доставляється лист). Порт – це програмне поняття, яке використовується клієнтом або сервером для посилання або приймання повідомлень; порт ідентифікується 16-бітовим числом. Більшість серверних додатків мають стандартні номери, наприклад, поштовий сервіс SMTP, приєднаний до порту за номером 25 (говорять: «слухає» порт або приймає на нього повідомлення), веб-сервер-сервіс, приєднаний до порту 80, FTP – до

порту 21, POP3-сервер, що забезпечує читання пошти з поштових скриньок, – до порту 110 і т. д.

Таким чином, у комп'ютерних мережах, що працюють за протоколами TCP/IP, аналогом паперового листа в конверті є **пакет**, який містить дані, що передаються, і адресну інформацію – адреси відправника і одержувача. Комбінація **IP-адреса і номер порту** – називається **сокетом**. У пакетах також присутня службова інформація, але для розуміння суті роботи протоколів TCP/IP це неважливо.

Розглянемо приклад: із сокета 82.146.49.55:2049 посилаємо пакет на сокет 195.34.32.116:53, тобто пакет з адреси відправника IP 82.146.49.55 з порту 2049 прямує у комп'ютер, що має адресу IP 195.34.32.116, у порт 53. Порту 53 відповідає **сервер розпізнавання імен** (**DNS-сервер**), який прийме цей пакет. Знаючи адресу відправника, цей сервер зможе після оброблення запиту сформувати у відповідь пакет, який відправиться у зворотному напрямі на сокет відправника 82.146.49.55:2049, який для DNS-сервера буде сокетом одержувача.

Як звичайно, взаємодія здійснюється за схемою «клієнт-сервер»: клієнт запрошує яку-небудь інформацію (наприклад, сторінку сайта), сервер приймає запит, обробляє його і посилає результат. Номери портів серверних додатків загальновідомі. Більшість програм на домашньому комп'ютері є клієнтами – наприклад, поштовий клієнт Outlook, веб-оглядувачі IE, FireFox та ін. Номери портів на клієнтові не фіксовані, як у сервера, а призначаються операційною системою динамічно. Фіксовані серверні порти переважно мають номери до 1024 (але є виключення), а клієнтські починаються після 1024.

Стек протоколів TCP/IP – це набір протоколів, який містить TCP (Transfer Control Protocol), IP (Internet Protocol), UDP (User Datagram Protocol), ICMP (Internet Control Message Protocol) і низку інших протоколів. Структура протоколів TCP/IP показана на рисунку 6.1.

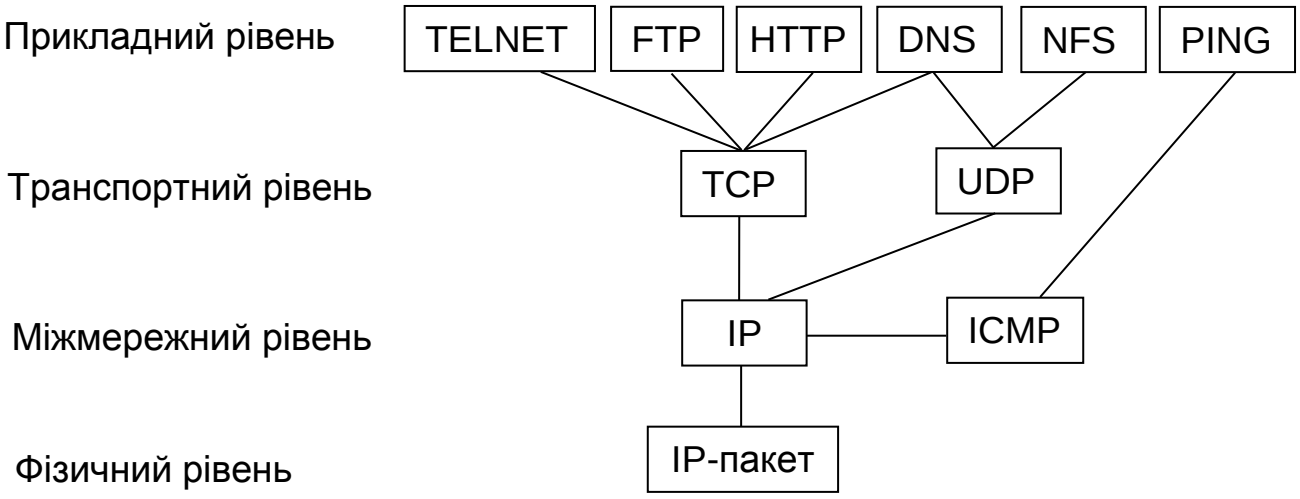

Рисунок 6.1 – Внутрішня структура TCP/IP

Протокол – це узгоджена процедура передачі даних між різними об'єктами обчислювальної системи; набір правил і форматів, семантичних і синтаксичних, які дозволяють різним компонентам системи обмінюватися інформацією (наприклад, вузлам мережі).

IP – це протокол так званого міжмережного рівня. Завдання цього рівня – доставка IP-пакетів від комп'ютера відправника до комп'ютера одержувача. Пакети цього рівня мають IP-адресу відправника і IP-адресу одержувача. Номери портів на цьому рівні не використовуються. Якому порту, тобто додатку, адресований цей пакет, чи був цей пакет доставлений або був втрачений на цьому рівні – невідомо. Це – завдання транспортного рівня.

ICMP – це розширення IP-протоколу. ICMP (протокол міжмережних керуючих повідомлень, повідомлень про помилки) призначено для передачі інформації, необхідної для керування трафіком. В основному він використовується для надання інформації про шляхи до хостів-одержувачів (інформує хости про існування коротших маршрутів до інших систем і вказує на проблеми зі знаходження шляху до одержувача). Може допомогти коректно завершити з'єднання TCP, якщо шлях став недоступним.

TCP і UDP – це протоколи так званого транспортного рівня. На цьому рівні до пакета додаються порти відправника і одержувача.

TCP – це протокол зі встановленням з'єднання і з гарантованою доставкою пакетів. Спочатку проводиться обмін спеціальними пакетами для встановлення з'єднання. Далі за цим з'єднанням туди і назад посилаються пакети, причому з перевіркою, чи дійшов пакет до одержувача. Якщо пакет не дійшов, то він посилається повторно.

UDP – це протокол без встановлення з'єднання і з негарантованою доставкою пакетів. UDP взаємодіє з прикладними програмами на тому ж рівні, що і TCP. Однак він не виконує функції виправлення помилок або повторної передачі втрачених пакетів. Тому UDP не використовується у сервісах зі встановленням з'єднання, а застосовується у сервісах типу запит-відповідь. Пакети UDP набагато простіше підроблювати, чим пакети TCP, оскільки немає етапу встановлення з'єднання. Тому використання сервісів на базі UDP пов'язано з великим ризиком.

Над транспортним рівнем знаходиться прикладний рівень. На цьому рівні працюють такі протоколи, як, наприклад, HTTP і FTP. Сервіси з встановленням з'єднання, такі, як TELNET, FTP, SMTP, потребують надійності і тому використовують TCP. Сервіс DNS застосовує TCP тільки в окремих випадках (для передачі і прийому баз даних доменних імен), а для передачі інформації про окремі хости використовує UDP.

Типові сервіси, пов'язані з TCP/IP:

PING – перевірка з'єднання у мережах;

TELNET – підключення до віддалених систем, приєднаних до мережі, застосовує базові можливості з емуляції терміналу;

FTP – передавання файлів для їх обміну між системами у мережі;

DNS – служба мережних імен, використовується TELNET і FTP та іншими сервісами для трансляції імен хостів у IP-адреси;

NFS – мережна файлова система, яка дозволяє системам спільно використовувати директорії і диски, при цьому віддалена директорія або диск здаються такими, що знаходяться на локальній машині;

NIS – мережні інформаційні сервіси, які дозволяють декільком системам спільно використовувати бази даних, наприклад, файл паролів, для централізованого керування ними;

SMTP – отримання листів від інших систем і зберігання їх у поштових скриньках користувачів;

POP – завантаження листів, отриманих від іншого поштового сервера;

IMAP – створення, видалення, перейменування поштових скриньок; перевірка надходження нових листів, оперативне видалення листів; установлення і скидання прапорів операцій; пошук серед листів, вибіркове читання листів;

HTTP – передавання гіпертекстових документів за базовим протоколом **WWW (**World Wide Web);

SSL – здійснення безпечної передачі даних – міжкінцеве шифрування трафіку на прикладному рівні;

gopher – пошук і переглядання інформації за допомогою системи меню, що може створити «дружній» інтерфейс до інших інформаційних сервісів;

X Windows – графічне віконне середовище і набір прикладних бібліотек, які використовуються на робочих станціях;

rlogin, rsh та інші **r-сервіси** – реалізують концепцію хостів, що довіряють один одному і дозволяють виконувати команди на інших комп'ютерах, не вводячи пароля.

Приєднання до Інтернету може мати величезні переваги, хоча існують серйозні проблеми з його безпекою. Перш за все проблеми виникають через уразливість мережного програмного забезпечення і відсутність засобів захисту, наявність помилок при конфігурації хосту через недосконаліть засобів керування доступом, які або погано встановлені, або дуже складні.

6.2 Проблеми, пов’язані з безпекою в мережі Інтернет

Система імен доменів (DNS) є розподіленою базою даних, яка перетворює імена користувачів і хостів в IP-адреси і навпаки. DNS також зберігає інформацію про структуру мережі, наприклад про кількість комп'ютерів з IP-адресами у кожному домені. Однією з проблем DNS є те,

що цю базу даних дуже важко приховати від неавторизованих користувачів. В результаті DNS часто використовується хакерами як джерело інформації про імена довірених хостів.

FTP забезпечує передачу текстових і двійкових файлів, тому його часто використовують у мережі Інтернет для сумісного доступу до інформації. Деякі FTP-сервери обмежують доступ користувачів до своїх архівів даних за допомогою пароля, інші ж надають вільний доступ (анонімний FTP-сервер). Якщо використовується опція анонімного FTP для сервера, то необхідно упевнитися, що на ньому зберігаються тільки файли, призначені для вільного поширення.

World Wide Web (WWW) – система, основана на мережних додатках, які дають можливість користувачам проглядати вміст різних серверів у мережі Інтернет. Найкориснішою властивістю WWW є використання гіпертекстових документів з вбудованими посиланнями на інші документи і Web-вузли, що дає можливість легко переходити від одного вузла до іншого. Ця ж властивість є і найбільш слабкою ланкою системи WWW, оскільки посилання на Web-вузли, що зберігаються у гіпертекстових документах, містять інформацію про те, як здійснюється доступ до відповідних вузлів. Використовуючи цю інформацію, хакери можуть зруйнувати Web-вузол або дістати доступ до конфіденційної інформації, що зберігається на ньому.

Слабка аутентифікація. Операційна система Unix зазвичай зберігає паролі у зашифрованій формі у файлі, який може бути прочитаним будь-яким користувачем. Цей файл паролів можна отримати простим копіюванням або яким-небудь іншим способом, що використовує зловмисник. Як тільки файл буде одержано, зловмисник зможе запустити доступні програми злому паролів для цього файла.

Спостереження за даними, що передаються. Коли користувач встановлює сеанс з віддаленим хостом, використовуючи TELNET або FTP, то пароль користувача передається у мережу Інтернет незашифрованим. Тому іншим способом проникнення у системи є спостереження за з'єднанням з метою перехоплення IP-пакетів, що містять ім'я і пароль, і подальше їх використання для нормального входу у систему. Якщо перехоплений пароль є паролем адміністратора, то завдання діставання привілейованого доступу набагато полегшується.

Маскування під інших користувачів. Маршрутизація IP-джерела – це опція, за допомогою якої можна вказати маршрут до призначення і шлях, по якому пакет повертатиметься до відправника. Цей шлях може містити інші маршрутизатори або хости, які у звичайних умовах не використовуються при передачі пакетів. Застосовуючи маршрутизацію

IP-джерела, атакуючий хост може замаскуватися під довірений хост або клієнта.

Потенційні проблеми з електронною поштою. Взаємодія хостів у мережі Інтернет при обміні поштою відбувається за допомогою простого протоколу, що використовує текстові команди. Зловмисник може легко ввести ці команди вручну, використовуючи TELNET для встановлення сеансу. Приймальний хост довіряє тому, хто заявляє про себе як хост-відправник, тому можна легко вказати помилкове джерело листа, ввівши адресу електронної пошти як адресу відправника, яка відрізнятиметься від дійсної адреси.

Складність конфігурації і заходів захисту. Системи керування доступом у хостах часто складні у настройці і важкі для перевірки правильності їх роботи. В результаті неправильно сконфігуровані заходи захисту можуть призвести до проникнення зловмисників. Помилки, що призводять до неавторизованого доступу, існують через складність програм і неможливість перевірити їх у всіх середовищах, в яких вони мають працювати.

Запитання для самоперевірки

1. Що являє собою мережа Інтернет ?
2. Які основні рівні має внутрішня структура протоколів TCP/IP ?
3. Які існують типові сервіси, пов'язані з TCP/IP ?
4. За допомогою яких полів здійснюється з'єднання у протоколах TCP або UDP ?
5. Що таке сервіс DNS і які проблеми з безпекою в мережі Інтернет, пов’язані з ним ?
6. Які проблеми з безпекою в мережі Інтернет створює система World Wide Web (WWW) ?
7. Які існують потенційні проблеми з електронною поштою ?

Лекція № 7

ЗАХИСТ ІНФОРМАЦІЇ У МЕРЕЖІ INTERNET

Навчальні цілі:

- вивчити основні компоненти брандмауерів;
- розглянути основні підходи до захисту інформації за допомогою брандмауерів.

Навчальні питання:

1. Система брандмауера.
2. Основні компоненти брандмауера.
3. Приклади брандмауерів.

7.1 Система брандмауера

Система брандмауера – це набір систем і маршрутизаторів, доданих у мережу у місцях її з'єднання з Інтернетом, а також політика доступу, що визначає правила їх роботи. Брандмауер примушує всі мережні з'єднання проходити через «шлюз», де вони можуть бути проаналізовані та оцінені щодо безпеки, і надає можливості використовувати засоби посиленої аутентифікації замість простих паролів.

Можливості брандмауерів полягають у такому:

- обмеженні доступу до тих або інших систем з мережі Інтернет або до Інтернету з цих систем;
- блокуванні певних сервісів TCP/IP або забезпеченні інших заходів безпеки;
- підвищенні мережної безпеки і зменшенні ризиків для хостів у підмережі шляхом фільтрації небезпечних за своєю природою сервісів;
- забезпеченні захисту від атак з використанням маршрутизації, такої, як маршрутизація джерела, і спроб змінити маршрути передачі даних за допомогою команд перенаправлення протоколу ICMP. Брандмауер може заблокувати всі пакети з маршрутизацією джерела і перенаправити протокол ICMP, а потім інформувати адміністраторів про інциденти;
- наданні можливості щодо керування доступом до хостів мережі. Наприклад, деякі хости можуть бути досяжними через зовнішні мережі, тоді, як доступ до інших систем ззовні буде забороненим;
- протоколюванні доступу і поданні статистичних даних про використання мережі, якщо весь доступ до мережі Інтернет і вихід з неї здійснюється через брандмауер;
- поданні детальної інформації про те, чи були брандмауер або мережа атаковані або зондовані при правильно налаштованій системі сигналів про підозрілі події;
- забезпеченні керування доступом користувачами і службами. Політика мережного доступу може бути реалізована за допомогою

брандмауера, без нього така політика залежить цілком від доброї волі користувачів.

Недоліки використання брандмауерів:

- можуть блокувати ряд сервісів, які застосовують користувачі, такі, як TELNET, FTP, X Windows, NFS та ін.;
- не захищають від чорних входів («люків») у мережі. Наприклад, якщо можна здійснити необмежений доступ за модемом у мережу, захищену брандмауером, зловмисники можуть ефективно обійти брандмауер;
- зазвичай не забезпечують захисту від внутрішніх загроз, від копіювання співробітниками даних на дискету і винесення її за межі мережі. Не слід вкладати значні ресурси у брандмауер, якщо є інші способи викрасти дані;
- інформаційні сервери і клієнти, такі, як WWW, gopher, WAIS і ряд інших не розраховані на спільну роботу з брандмауером. Є потенційна можливість атак за допомогою передачі спеціальних даних, які можуть містити команди клієнту змінити параметри засобів керування доступом або модифікувати важливі файли, пов'язані із захистом комп’ютера клієнта;
- зазвичай пропускають без перевірки пакети, в які вбудовують за допомогою сервісу MBONE групові передачі, що містять звук і зображення. Передачі типу MBONE є потенційною загрозою, якщо пакети містять команди, що змінюють параметри роботи засобів захисту і дозволяють зловмисникам дістати доступ;
- не захищають від користувачів, що завантажують програми, заражені вірусами. Ці програми можуть бути закодовані або стиснуті, і брандмауер не може сканувати такі програми, щоб виявити сигнатури вірусів. Проблема вірусів залишатиметься і має бути вирішена за допомогою інших засобів;
- є потенційно вузьким місцем, оскільки всі з'єднання мають проходити через брандмауер і в деяких випадках вивчатися ним (зниження пропускної спроможності);
- концентрують безпеку в одному місці, а не розподіляють її серед групи систем. Компрометація брандмауера може бути жахливою для погано захищених систем у підмережі.

7.2 Основні компоненти брандмауера

Основними компонентами брандмауера є:

- політика мережного доступу;
- механізми посиленої аутентифікації;
- фільтрація пакетів;
- прикладні шлюзи.

Політика мережного доступу має два рівні.

Політика **верхнього рівня (політика доступу до сервісів)** визначає:

- до яких сервісів доступ буде дозволеним або явно забороненим у мережі, що захищається;
- як будуть використовуватися ці сервіси;
- за яких умов будуть робитися виключення і коли політика не буде дотримуватися.

Політика **нижнього рівня (політика проекту брандмауера)** містить інформацію про те, як брандмауер має насправді обмежувати доступ і фільтрувати сервіси, які вказані у політиці верхнього рівня.

Політика доступу до сервісів має бути уточненням загальної політики підприємства відносно захисту інформаційних ресурсів. Якщо система брандмауера забороняє або обмежує використання деяких сервісів, то у політиці слід описати строгість, з якою це робиться, щоб запобігти зміні параметрів засобів керування доступом.

Типова політика щодо сервісів:

- забороняти доступ до внутрішньої мережі з мережі Інтернет і дозволяти тільки доступ до мережі Інтернет з внутрішньої мережі;
- дозволяти деякий доступ з мережі Інтернет, але тільки до вибраних систем (наприклад, інформаційних і поштових серверів);
- дозволяти працювати з мережі Інтернет з деякими вибраними хостами, але цей доступ надається тільки тоді, якщо він поєднується з посиленою аутентифікацією.

Політика проекту брандмауера:

- дозволяти доступ до сервісу, якщо він явно не заборонений (брандмауер пропускає всі сервіси у мережу за умовчанням, якщо тільки цей сервіс не був явно вказаний у політиці керування доступом як заборонений);
- забороняти доступ до сервісу, якщо він явно не дозволений (брандмауер за умовчанням забороняє всі сервіси, але пропускає ті, які вказані у списку дозволених сервісів).

Перша політика є менш бажаною, оскільки вона надає більше способів обійти брандмауер, наприклад, користувачі можуть дістати доступ до нових сервісів, що не забороняються політикою (або навіть не вказаних у політиці).

Друга політика є більш суворою і безпечнішою, але її важче реалізувати і вона може вплинути на роботу користувачів, оскільки деякі сервіси можуть виявитися блокованими або використання їх буде обмежено.

Посилена аутентифікація. Той факт, що зловмисники можуть спостерігати за каналами у мережі Інтернет і перехоплювати паролі, що передаються в них, робить традиційні паролі застарілими. Розроблено ряд засобів посиленої аутентифікації, таких, як **смарт-карти, біометричні**

механізми і програмні механізми. Загальним є те, що паролі, які генеруються пристроєм посиленої аутентифікації, не можуть бути повторно використані зловмисниками. Ряд найбільш популярних пристроїв посиленої аутентифікації, що використовуються сьогодні, називаються **системами з одноразовими паролями**.

Фільтрація пакетів. Фільтрація IP-пакетів зазвичай виконується за допомогою маршрутизатора з фільтрацією пакетів, що здійснює її, коли пакети передаються між його інтерфейсами. Фільтрація може застосовуватися по-різному для блокування з'єднань від (або до) окремих хостів або мереж і для блокування з'єднань до різних портів.

Рішення про те, які протоколи або групи портів фільтрувати, залежить від політики мережного доступу, тобто від того, які системи повинні мати доступ до мережі Інтернет і які типи доступу дозволені.

Потенційно вразливі до атак і зазвичай блоковані сервіси:

- Tftp, порт 69, спрощений FTP, може також бути використаний для читання будь-якого файла у системі при його неправильному установленні;

- X Windows, Open Windows, порти 6000+, 2000, можуть використовуватися для перехоплення зображення вікон, а також символів, що вводяться;

- RPC, порт 111, служби виклику віддалених процедур, які можуть застосовуватися для викрадання системної інформації, включаючи паролі, а також читання і запису файлів;

- rlogin, rsh, rexec, порти 513, 514, 512, служби, які можуть при їх неправильній конфігурації призвести до доступу в систему.

Сервіси, що дозволяються тільки для окремих систем:

- TELNET, порт 23;

- FTP, порти 20 і 21;

- SMTP, порт 25, дозволяється тільки для центрального поштового серверу;

- RIP, порт 520, протокол передачі інформації про маршрутизацію пакетів, може бути фальсифікований для перенаправлення пакетів;

- DNS, порт 53;

- NNTP, порт 119, протокол передачі мережних новин, протокол для доступу і читання мережних новин.

До недоліків фільтрації пакетів належать:

- правила фільтрації пакетів складно формулюються і зазвичай немає засобів для тестування їх коректності;

- у деяких маршрутизаторах немає засобів протоколювання, тобто небезпечні пакети не зможуть бути визначені до виявлення проникнення;

- часто потрібно зробити виключення з правил, але виключення з правил фільтрації іноді можуть зробити правила фільтрації такими складними, що вони стануть неконтрольованими.

Прикладні шлюзи. У брандмауерах можна використовувати прикладні програми для перенаправлення і фільтрації з'єднань, які називаються **проксі-службою.**

Хост, на якому працює проксі-служба, називається прикладним шлюзом.

Користувачу, який хоче з'єднатися зовні з системою у мережі, слід спочатку з'єднатися з прикладним шлюзом, а потім вже з потрібним хостом:

- спочатку користувач встановлює з'єднання з прикладним шлюзом і вводить ім'я внутрішнього хосту;
- шлюз перевіряє IP-адресу користувача і вирішує або забороняє з'єднання відповідно до того або іншого критерію доступу;
- може знадобитися аутентифікація користувача (можливо за допомогою одноразових паролів);
- проксі-сервер створює з'єднання між шлюзом і внутрішнім хостом;
- проксі-сервер передає дані між цими двома з'єднаннями;
- прикладний шлюз протоколює з'єднання.

Проксі-служби дозволяють використовувати тільки ті служби, для яких є проксі. Іншими словами, якщо прикладний шлюз містить проксі для FTP і TELNET, то у підмережі, що захищається, будуть дозволені доступи тільки до FTP і TELNET, а до інших служб доступ буде повністю блокований.

Прикладний шлюз електронної пошти призначено для централізованого збирання електронної пошти і розподілення її за внутрішніми хостами і користувачами.

Прикладні шлюзи мають такі переваги:

- приховування інформації, при якому імена внутрішніх систем необов'язково будуть відомі зовнішнім системам за допомогою DNS, оскільки прикладний шлюз може бути єдиним хостом, чиє ім'я має бути відоме зовнішнім системам;

- надійна аутентифікація і протоколювання, при якому прикладний трафік може бути попереднє аутентифікований до того, як він досягне внутрішніх хостів, і може бути запротокольований;

- оптимальне співвідношення між ціною та ефективністю через те, що додаткові програми або обладнання для аутентифікації або протоколювання потрібно встановлювати тільки на прикладному шлюзі;

- прості правила фільтрації, оскільки правила на маршрутизаторі з фільтрацією пакетів будуть менш складними, чим вони були б, якби маршрутизатор сам фільтрував прикладний трафік і відправляв його багатьом внутрішнім системам.

Основним недоліком прикладного шлюзу є те, що під час використання клієнт-серверних протоколів, наприклад, таких, як TELNET, потрібна двокрокова процедура для входження всередину або виходу назовні.

7.3 Приклади брандмауерів

Основними типами брандмауерів, які найчастіше використовуються на практиці, є: з пакетною фільтрацією; шлюзом з двома адресами; екранованим хостом; екранованою мережею.

Брандмауер з пакетною фільтрацією (рисунок 7.1) є найпоширенішим і найпростішим при реалізації в невеликих мережах з простою структурою.

Він встановлюється на маршрутизаторі з фільтрацією пакетів, через який відбувається з'єднання з мережею Інтернет (або підмережею), на якому конфігуруються правила фільтрації пакетів, що дозволяють блокувати або фільтрувати пакети на основі протоколів і адрес. Зазвичай машинам внутрішньої мережі надається повний доступ до Інтернету, а доступ з Інтернету до всіх або майже до всіх систем внутрішньої мережі блокується. Маршрутизатор може допускати вибірковий доступ до систем і сервісів (це залежить від політики).

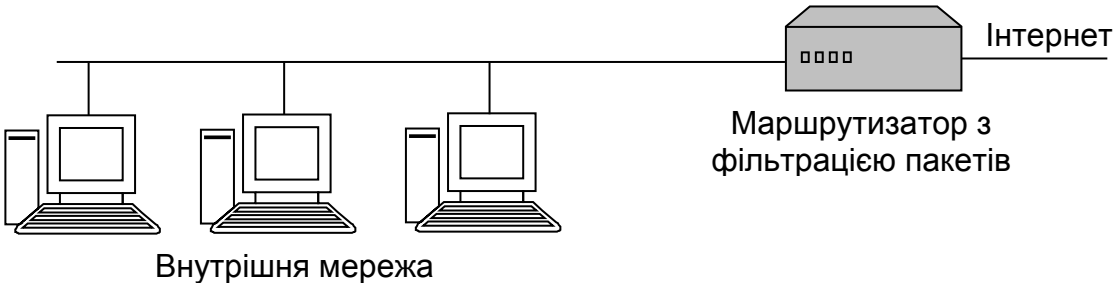

Рисунок 7.1 – Брандмауер з пакетною фільтрацією

Недоліками брандмауера з пакетною фільтрацією є:

- відсутність можливості протоколювання, нелегко виявити компрометацію маршрутизатора або атаку на мережу;
- правила фільтрації часто важко протестувати, що може призвести до виникнення вразливих місць;
- кожен хост, до якого потрібно забезпечити доступ з Інтернету, потребуватиме реалізацію заходів посиленої аутентифікації.

Брандмауер з прикладним шлюзом з двома адресами (рисунок 7.2) (на основі машини, підключеної до двох мереж) складається з хосту, що має два мережні інтерфейси, в яких відсутня функція маршрутизації IP-пакетів з одного інтерфейсу на інший (тобто хост не може маршрутизувати пакети між двома мережами). Він може повністю блокувати передачу трафіка між мережею Інтернет і захищеною мережею. Сервіси на цьому хості обслуговуються за допомогою проксі-серверів.

Брандмауер забезпечує можливість відокремити трафік, пов'язаний з інформаційним сервером, від трафіка інших систем сайту. Шлюз забезпечує використання відповідних проксі-сервісів в інформаційному сервері, маршрутизатор запобігає прямому доступу з Інтернету до серверу і примушує звернутися до брандмауера.

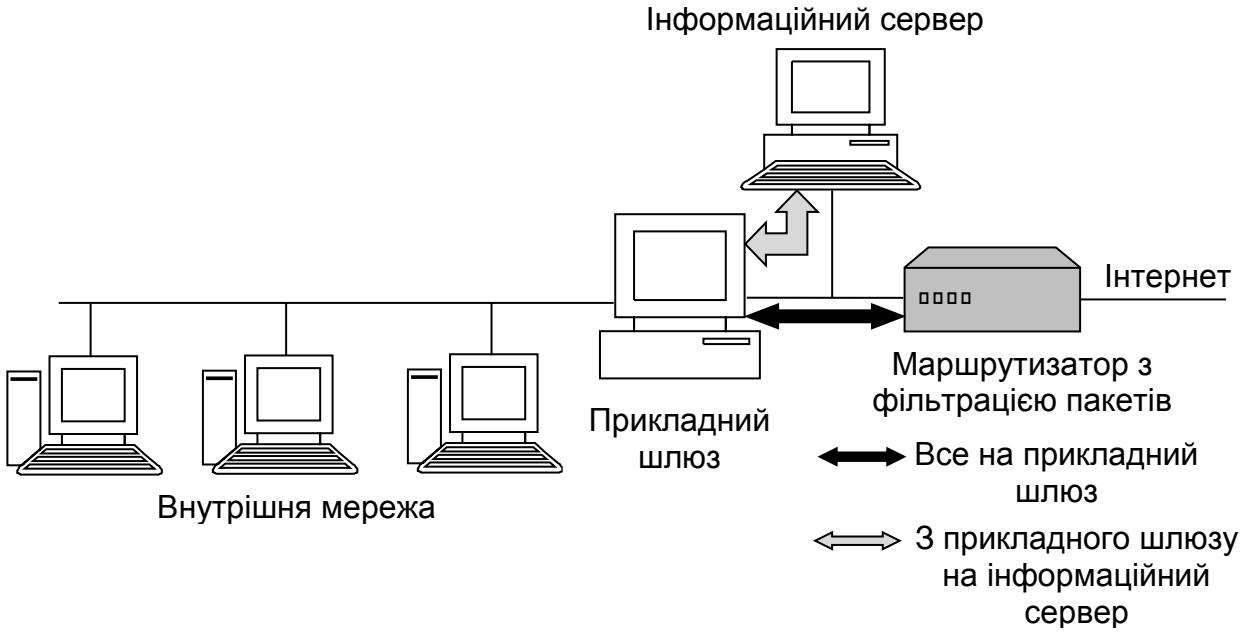

Рисунок 7.2 – Брандмауер з прикладним шлюзом з двома адресами

Жорсткість шлюзу з двома інтерфейсами може виявитися незручною для деяких мереж. Оскільки всі сервіси за умовчанням заблоковані, окрім тих, для яких є проксі-сервери, доступ до інших сервісів не може бути організовано. Системи, до яких потрібний доступ, мають бути розміщені між шлюзом і мережею Інтернет. Іншим важливим моментом є те, що безпека хосту, що використовується для організації брандмауера, має бути високою, оскільки використання уразливих сервісів і технологій може привести до проникнення у захищену мережу.

Брандмауер з екранованим (ізольованим) хостом (рисунок 7.3) складається з маршрутизатора з фільтрацією пакетів і прикладного шлюзу, розміщеного у захищеній підмережі. Прикладному шлюзу потрібний тільки один інтерфейс з мережею і не потрібна окрема підмережа між прикладним шлюзом і маршрутизатором. Проксі-сервіси шлюзу мають пропускати запити до сервісів, у яких є проксі у внутрішні машини мережі. Маршрутизатор фільтрує або блокує потенційно небезпечні протоколи, щоб вони не досягли прикладного шлюзу, і внутрішні системи.

На відміну від шлюзу з двома інтерфейсами це дозволяє брандмауеру бути гнучкішим, але менш безпечним, оскільки маршрутизатор може дозволити пропустити запити до надійних сервісів в обхід прикладного шлюзу.

Недоліками брандмауера з екранованим (ізольованим) хостом є:

- існуючі дві системи, маршрутизатор і прикладний шлюз слід конфігурувати, правила фільтрації пакетів можуть бути складними, важкими для тестування і уразливими до помилок, що ведуть до появи вразливих місць. Однак обмежування трафіка тільки для прикладного шлюзу може зробити правила не такими складними;
- гнучкість робить можливим порушення політики.

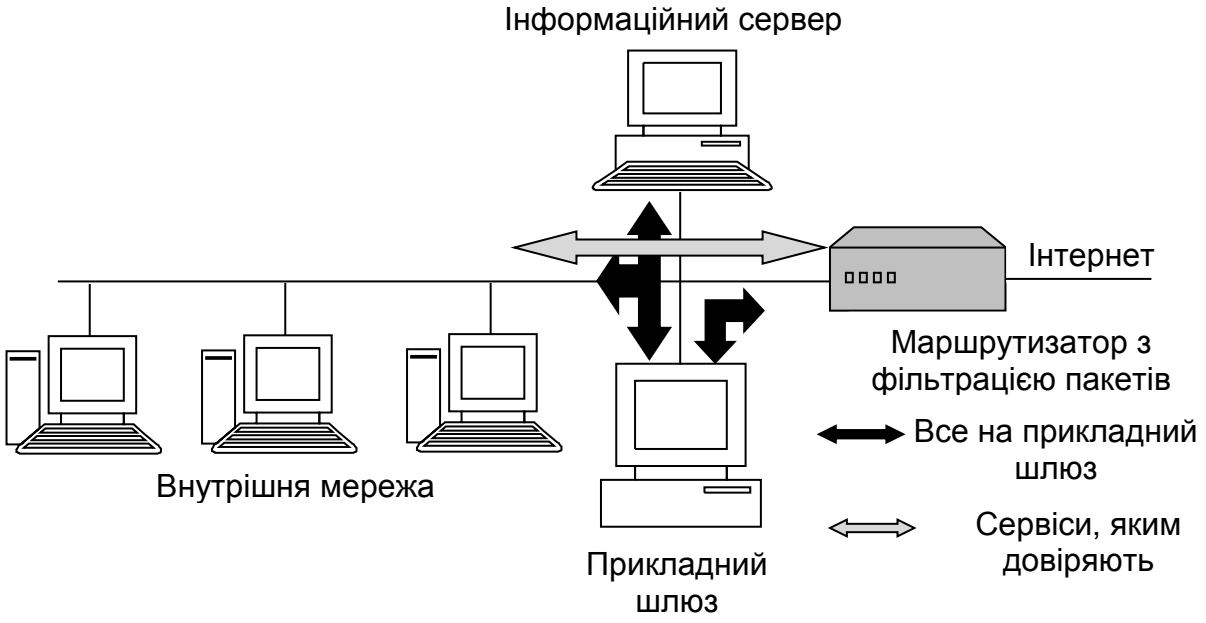

Рисунок 7.3 – Брандмауер з екранованим (ізольованим) хостом

Брандмауер з екранованою (ізольованою) мережею (рисунок 7.4) – це об'єднання шлюзу з двома інтерфейсами і брандмауера з ізольованим хостом. Він може бути використаний для того, щоб розмістити кожну компоненту брандмауера в окремій системі, забезпечивши велику пропускну спроможність і гнучкість, хоча це може призвести до деяких ускладнень.

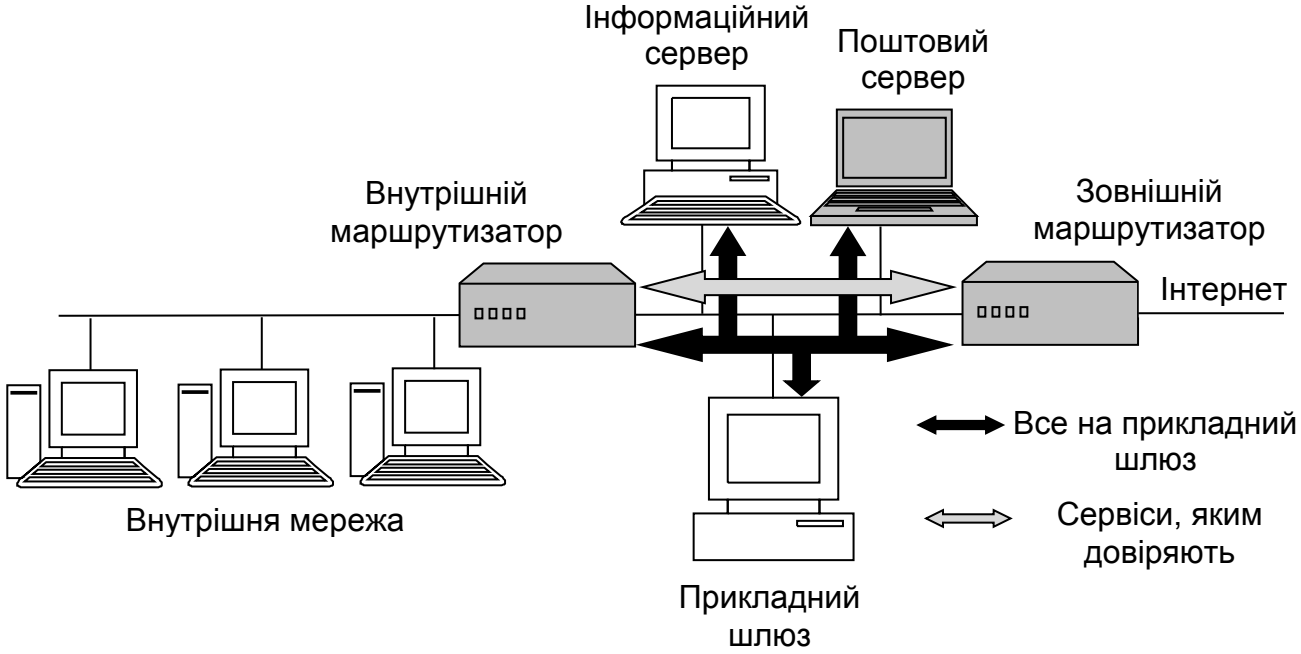

Рисунок 7.4 – Брандмауер з екранованою (ізольованою) мережею

На рисунку 7.4 показано два маршрутизатори, що використовуються для створення внутрішньої ізольованої підмережі. У цій підмережі (іноді званою DMZ – демілітаризованою зоною) знаходиться прикладний шлюз,

але в ній також можуть розміщуватися інформаційні сервери, модемні пули та інші системи.

При такій побудові не існує внутрішніх систем, безпосередньо доступних з мережі Інтернет, і навпаки. Може бути досягнута велика пропускна спроможність, якщо маршрутизатор використовується як шлюз для захищеної підмережі. Як наслідок, брандмауер з ізольованою підмережею може виявитися більш доречним варіантом для мереж з великим обсягом трафіка або мереж, яким потрібний високошвидкісний трафік.

Два маршрутизатори забезпечують додатковий шар захисту, оскільки тому, хто атакує, необхідно буде обійти засоби захисту в обох маршрутизаторах, щоб дістати доступу до внутрішніх систем. Прикладний шлюз, поштовий сервер та інформаційний сервер можуть бути встановлені так, що будуть єдиними системами, що видимі з Інтернету. На прикладному шлюзі можуть бути встановлені заходи посиленої аутентифікації для всіх вхідних з'єднань. Звичайно, це потребує додаткової конфігурації, але використання окремих систем для прикладного шлюзу і фільтрації пакетів зробить конфігурацію простішою. Як альтернативу передачі сервісів безпосередньо між мережею Інтернет і внутрішніми системами можна застосовувати системи, яким потрібні такі сервіси, прямо в ізольованій підмережі.

Недоліками брандмауера з екранованою (ізольованою) мережею є:

- можливість порушення політики, оскільки брандмауер можна конфігурувати так, що він пропускатиме довірені сервіси в обхід прикладного шлюзу;
- довірені сервіси, які передаються в обхід прикладного шлюзу, безпосередньо взаємодіють з внутрішніми системами;
- на маршрутизатори покладаються великі завдання із забезпечення безпеки і їх важко правильно конфігурувати, а помилки можуть призвести до появи вразливих місць.

Запитання для самоперевірки

1. Що таке брандмауер і які його можливості ?
2. Які проблеми можуть виникати під час застосування брандмауерів ?
3. Які основні компоненти має брандмауер ?
4. Які існують рівні політики мережного доступу ?
5. Як створюються прикладні шлюзи у комп’ютерних системах ?
6. У чому полягають особливості побудови брандмауера з прикладним шлюзом з двома адресами ?
7. У чому полягають переваги брандмауера з екранованою мережею порівняно з брандмауером з екранованим хостом ?

Лекція № 8

ОСНОВИ КРИПТОГРАФІЇ

Навчальні цілі:
- вивчити можливості тайної передачі інформації;
- розглянути історію розвитку криптографії.

Навчальні питання:
1. Можливості тайної передачі інформації.
2. Коротка історія розвитку криптографії.

8.1 Можливості тайної передачі інформації

Існує декілька способів тайної передачі інформації.

По-перше, можна створити абсолютно надійний, недоступний для інших канал зв’язку між абонентами. Однак при сучасному рівні розвитку науки і техніки зробити такий канал зв'язку між віддаленими абонентами для неодноразової передачі великих обсягів інформації практично нереально.

По-друге, можна використовувати загальнодоступний канал зв'язку, але приховати сам факт передачі інформації. Розробленням способів і методів приховування факту передачі повідомлення займається така наука, як **стеганографія**.

Перші приклади приховування факту передачі інформації були ще у Древньому Римі, коли голову раба голили, на шкірі голови писали повідомлення і після відростання волосся раба відправляли до адресата.

У різні часи використовували різноманітні способи тайнопису між рядків звичайного тексту: від молока до складних хімічних реактивів з подальшим обробленням.

У 1566 р. була запропонована **решітка Кардано** як інструмент кодування і декодування. Вона є спеціальною прямокутною (квадратною) таблицею-карткою, частина чарунок якої вирізана. Шифратор поміщає решітку на лист паперу і пише повідомлення у прямокутних чарунках, в яких поміщається окремий символ або ціле слово. Таким чином початкове повідомлення розподіляється на велику кількість маленьких фрагментів. Потім решітку прибирають, і порожні місця на папері заповнюють стороннім текстом так, щоб приховуваний текст став частиною криптотексту. Таке заповнення потребує певного літературного таланту.

Метод мікрокрапки полягає у записі повідомлення за допомогою сучасної техніки на дуже маленький носій (мікрокрапку), який пересилається із звичайним листом, наприклад, під маркою або в іншому, заздалегідь обумовленому місці.

З розвитком комп’ютерної техніки з’явилися нові методи приховування інформації.

Комп'ютерне приховування текстів – це:

- захований текстового файла у графічний файл. Файл з растровою картинкою, в якому найменш значущий біт у коді яскравості кожної точки зображення буде елементом таємного повідомлення. Одержувач листа «витягуватиме» всі такі біти і складатиме з них дійсне повідомлення. Картинка, присутня тут тільки як фон, так і залишиться для необізнаних просто картинкою;
- форматування диска під розмір секторів, яке є відмінним від прийнятого у DOS, і читати з диска стає неможливим. Були створені програми, за допомогою яких можна читати будь-яке форматування;
- захований інформації на дискетах (флешках), для чого широко використовуються їх інженерні доріжки, доступні для читання, але, які не сприймаються операційними системами;
- дописування інформації із застосуванням програми редактора диска у вільній частині хвостового кластера файла. Однак такі записи дуже просто розкривати.

Таке приховування інформації має той недолік, що воно обумовлено лише станом розвитку техніки, яка стрімко удосконалюється.

Основними недоліками стеганографії є:

- важко обґрунтувати її стійкість – раптом зловмисникам стане відомим спосіб «підмішування» таємних даних до масиву відкритих даних;
- обсяг даних, що передається або зберігається, різко збільшується, що негативно відбивається на продуктивності систем їх оброблення.

По-третє, можна використовувати загальнодоступний канал зв'язку, але передавати по ньому потрібну інформацію у перетвореному вигляді, щоб відновити її міг тільки адресат. Цим займається така наука, як **криптографія**.

Криптографія (іноді використовують термін «криптологія») – область знань, що вивчає тайнопис і методи його розкриття (криптоаналіз). Криптографія вважається розділом математики.

Шифрування (зашифровування) – процес застосування шифру до інформації, що захищається, тобто перетворення відкритого тексту у шифроване повідомлення (шифротекст, криптограму) за допомогою певних правил, що містяться у шифрі.

Дешифрування – процес, зворотний шифруванню, тобто перетворення шифрованого повідомлення в інформацію, що захищається, за допомогою певних правил, що містяться у шифрі.

До недавнього часу всі дослідження були тільки закритими. З одного боку, часткове зменшення секретності пояснюється тим, що стало вже неможливим приховувати накопичену кількість інформації. З іншого боку, криптографія все більше використовується у цивільних галузях.

Окрім забезпечення конфіденційності інформації криптографія також використовується у таких цілях:

- **перевірки достовірності.** Одержувач повідомлення може перевірити його джерело, зловмисник не зможе замаскуватися під кого-небудь;
- **збереження цілісності даних.** Одержувач може перевірити, чи не було повідомлення змінено у процесі доставки, зловмисник не зможе підмінити правильне повідомлення помилковим;
- **незаперечення авторства.** Відправник не зможе помилково заперечувати відправлення повідомлення.

Новизна комп'ютерних злочинів полягає у тому, що інформація, яка захищається, тепер зберігається не на папері, а на електронних пристроях. Не існує тестів, що дозволяють переконатися у надійності захищення інформації. Криптографія має ту особливість, що на розкриття шифру часто потрібно витратити на декілька порядків більше коштів, чим на його створення. Отже, тестові випробування системи криптозахисту не завжди можливі. Також досвід показує, що іноді багатократні невдалі спроби подолання захисту зовсім не означають, що наступна спроба не виявиться успішною. Не виключено випадок, коли професіонали не можуть розкрити шифр, а новачок, який застосує нестандартний підхід, може легко з цим справитися.

8.2 Коротка історія розвитку криптографії

Перші відомості про використання шифрів у військовій справі пов'язані з ім'ям спартанського полководця Лісандра, який використовував **шифр «Сцитали»**. Цей шифр відомий з часів війни Спарти проти Афін у V ст. до н.е. Для його реалізації використовували сциталу – жезл, що мав форму циліндра. На сциталу намотували вузьку папірусову стрічку (без просвітів і нахлестів), а потім на цій стрічці вздовж осі сцитали записувавли відкритий текст. Стрічку розмотували і виходило (для необізнаних), що впоперек стрічки безладно написані якісь букви. Потім стрічку відправляли адресатові. Адресат брав таку ж сциталу, таким же чином намотував на неї отриману стрічку і читав повідомлення вздовж осі сцитали.

У **шифрі Цезаря** застосовано таке перетворення відкритого тексту: кожну букву відкритого тексту замінюють третьою після неї буквою в алфавіті, який вважають написаним по кругу, тобто після букви «Я» йде буква «А»:

АБВГДЕЄЖЗІЙКЛМНОПРСТУФХЦЧШЩИЬЕЮЯ
ГДЕЄЖЗІЙКЛМНОПРСТУФХЦЧШЩИЬЕЮЯАБВ.

У такому шифрі можна замінювати букву не третьою після неї, а будь-якою іншою, головне, щоб той, кому посилається шифроване повідомлення, знав величину зсуву.

У стародавній Греції був винайдений вид шифру «**квадрат Полібія»**. У квадрат розміром 5х5 клітинок вписують усі букви алфавіту, при цьому букви I, J не розрізняються (J ототожнюється з буквою I):

	A	**B**	**C**	**D**	**E**
A	A	B	C	D	E
B	F	G	H	I	K
C	L	M	N	O	P
D	Q	R	S	T	U
E	V	W	X	Y	Z

Шифровану букву замінюють на координати квадрата, в якому вона записана. Так, B замінюють на AB, F на BA, R на DB і т. д. У такому шифрі можна координати квадрата визначати не буквами, а цифрами:

	1	**2**	**3**	**4**	**5**
1	A	B	C	D	E
2	F	G	H	I	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Тоді B замінюють на 12, F на 21, R на 42 і т. д. Можна змінювати і порядок цифр, що позначають координати квадрата.

Книжковий шифр. Еней – захисник Трої – запропонував проколювати малопомітні дірки у книзі або в іншому документі над буквами секретного повідомлення. У Першій світовій війні німецькі шпигуни замінили дірки на крапки, що наносили симпатичним чорнилом на букви газетного тексту. За часів Другої світової війни букву замінювали на номер рядка і номер цієї букви у рядку на заздалегідь обумовленій сторінці деякої книги. Ключем такого шифру є книга і використовувана сторінка в ній.

У 1700 р. була розроблена «**цифірна азбука»** Петра Великого. У цій азбуці шифром є лист паперу, на якому від руки виконана таблиця заміни: за горизонталлю розташовані в алфавітному порядку букви кирилиці або іншої азбуки, відповідні мові відкритого повідомлення, а під ними підписані елементи відповідного шифроалфавіту. При кодуванні букви відкритого повідомлення замінюють на елементи шифроалфавіту. Шифроалфавіт з часом може змінюватися.

Шифр Віженера є різновидом шифру Цезаря зі змінною величиною зсуву. Щоб знати, на скільки зсовувати чергову букву відкритого тексту, заздалегідь визначають спосіб запам'ятовування зсувів. Для цієї мети використовують ключове слово, кожна буква якого своїм номером в алфавіті вказує величину зсуву. Ключове слово повторюють стільки раз,

скільки потрібно для заміни всіх букв відкритого тексту. Наприклад, якщо ключове слово ВАЗА, то послідовність зсуву букв – 3191 3191 3191, а відкритий текст – КРИПТОГРАФІЯ, то після заміни він перетвориться у шифротекст НССРХПЛСГХСА.

У 20-ті роки минулого століття були створені **роторні машини,** які використовували механічні колеса (ротори) для виконання підставлення. На кожному роторі у довільному порядку розміщується алфавіт з 26 позицій і виконується просте підставлення. Наприклад, ротор може бути використаний для заміни літери A на F, B на U, C на I і т. д. Вихідні штирі одного ротора сполучаються з вхідними штирями наступного ротора.

Наприклад, у чотирироторній машині перший ротор може замінювати літеру A на F, другий – F на Y, третій – Y на E і четвертий – E на C. Літера C і буде кінцевим шифротекстом. Потім деякі ротори зміщуються, і наступного разу підставлення будуть іншими.

Найвідомішим роторним пристроєм була шифрувальна машина **Енігма** (Enigma), що використовувалася німцями у Другій світовій війні. Вона мала три ротори, які можна було вибрати з п'яти можливих, комутатор, що злегка тасував відкритий текст, і ротор, що відбивав, який примушував кожен ротор обробляти відкритий текст кожного листа двічі.

Для раннього етапу розвитку криптографії характерне таке:

- захисту піддавалися виключно текстові повідомлення, написані природними мовами;
- шифри, що використовувалися тоді, були достатньо простими і нескладними;
- науковий підхід до побудови шифрів і їх розкриття був відсутнім;
- криптографія використовувалася в дуже вузьких сферах – тільки для обслуговування вищої правлячої і військової верхівки держав;
- основним завданням криптографії був захист повідомлень, що передавалися, від несанкціонованого ознайомлення з ними.

Зміни у розвитку криптографії з появою комп’ютерів були такими:

- обсяги оброблюваної інформації зросли на декілька порядків;
- доступ до певних даних дозволив контролювати значні матеріальні і фінансові цінності;
- оброблювані дані стали надзвичайно різносторонніми і більше не зводились до виключно текстових даних;
- інформаційні взаємодії ускладнилися;
- суб'єктами інформаційних процесів тепер були не тільки люди, але й створені ними автоматичні системи, що працювали за закладеними в них програмами;
- обчислювальні «здібності» сучасних комп'ютерів підняли на абсолютно новий рівень як можливості з реалізації шифрів, так і можливості аналітиків з їх розкриття.

Найбільш перспективним напрямком розвитку криптографічних методів у наш час є квантова криптографія.

Квантова криптографія була запропонована Чарльзом Беннетом і Жилем Брессардом у 1982 р. Вони дійшли висновку, що роль фотона полягає не у зберіганні, а в передачі інформації, і можна розробити квантовий канал відкритого розподілу секретних ключів. У криптографії вважається, що лінії зв'язку завжди контролюються перехоплювачем, якому відомо зміст всіх передаваних повідомлень, про що можуть і не знати абоненти. Однак, якщо інформація кодується не ортогональними станами фотона, то порушник не може отримати відомостей навіть про наявність передачі без порушення цілісності цього процесу, що буде відразу ж виявлено. Перехопивши фотон, зловмисник не зможе зробити декілька його вимірювань, оскільки при першому ж фотон руйнуватиметься і не дозволить одержати ключову інформацію у необхідному обсязі. Тому навіть активний перехоплювач не зможе правильно передати аналогічний фотон одержувачу так, щоб перехоплення не було б виявлено.

Технологія квантової криптографії спирається на принципову невизначеність поведінки квантової системи, виражену в принципі невизначеності Гейзенберга, – неможливо одночасно отримати координати і імпульс частинки, неможливо виміряти один параметр фотона, не спотворивши інший.

Наведемо принцип роботи квантової криптографії. Спочатку відправник може послідовно посилати одержувачу фотони (кванти світла), кожен з яких має одну з чотирьох поляризацій: горизонтальну, вертикальну, діагональну справа-наліво і діагональну зліва-направо, тобто **–, |, /** і ****. У якій саме поляризації послати черговий фотон вибирають випадковим чином. Слід відзначити, що фотони з горизонтальною і діагональною справа-наліво поляризаціями (**–** і **/**) позначають біт 0, а фотони з вертикальною і діагональною зліва-направо поляризаціями (**|** і ****) – відповідно біт 1. Іншими словами, відправник посилає одержувачу послідовність бітів, при цьому кожен біт може бути поданий двома різними символами.

Під час приймання набору поляризованих фотонів одержувач має можливість вимірювати кожен з них за допомогою двох різноорієнтованих приладів, які позначають знаками «плюс» та «хрест» (**+** і **×**). Кожен спосіб вимірювання має свої особливості. Прилад **+** вимірює біти з горизонтальною і вертикальною поляризаціями (**–** і **|**) з абсолютною точністю, а при попаданні в нього фотона з діагональною справа-наліво і діагональною зліва-направо поляризаціями (**/** і ****) цей прилад з вірогідністю 0,5 приводить до результату **–**, а з вірогідністю 0,5 – до результату **|**. Те ж саме відбувається і у разі приладу **×**. Тільки він з абсолютною точністю вимірює біти з діагональною справа-наліво і діагональною зліва-направо поляризаціями (**/** і ****), а ось для фотонів з горизонтальною і вертикальною

поляризаціями (**–** і **|**) він знову ж таки з вірогідністю 0,5 і 0,5 приводить до результатів **/** або ****.

Відповідно одержувач для кожного фотона абсолютно випадково і незалежно від відправника вибирає один з приладів і вимірює поляризацію прийнятого фотона. Одержувач записує результати всіх вимірювань. Далі він зв'язується з відправником за допомогою відкритого каналу і повідомляє, які прилади він використовував для вимірювання. У відповідь відправник повідомляє одержувача, в яких вимірюваннях був вибраний правильний прилад. Разом вони викреслюють ті фотони, які були зміряні одержувачем за допомогою не того приладу, і послідовність фотонів, що залишилася, переводять в біти. Виходить, що і у відправника, і у одержувача тепер є по однаковій послідовності бітів, які можна використовувати як ключ для застосування одноразового блокнота.

Наприклад, відправник надіслав одержувачу поляризовані фотони в такій послідовності:

/	/	–	|	|	–	\	\	–	–	|	–	\	/	/	|

Одержувач провів їх вимірювання, використавши певні прилади (перший рядок таблиці показує тип використаного приладу; другий рядок таблиці – отриманий результат):

+	×	+	+	×	×	×	+	+	×	×	×	×	×	+	+
–	/	–	|	/	/	\	|	–	\	/	\	\	/	/	|

Відправник повідомляє одержувача, які вимірювання він провів правильним приладом, а які – неправильним. Одержувач викреслює неправильні вимірювання, і у результаті виходить двійковий ключ:

	/	–	|			\		–				\	/		|
	0	0	1			1		0				1	0		1

У 8/16 випадків (50 %) одержувач скористався неправильним приладом, так що з 16 переданих відправником фотонів прийшлось прибрати половину. Залишився ключ **00110101**. Цей ключ – секретний.

Наступним етапом дуже важливо оцінити спроби перехопити інформацію в квантово-криптографічному каналі зв'язку.

Нехай відправник пересилає ті ж самі фотони. Однак цього разу між ними знаходиться криптоаналітик і він перехоплює фотони. Оскільки він не знає, яка саме поляризація у фотонів відправника, йому також доводиться використовувати випадкове застосування двох різних типів приладів для вимірювання поляризації. Припустимо це зроблено так:

+	+	×	×	×	×	×	+	+	×	×	×	+	+	×	+
|	–	\	\	/	\	\	–	–	\	/	/	–	–	/	|

Потім йому слід переслати цю послідовність одержувачу. Але, як тільки криптоаналітик зміряв фотони, отримані від відправника, він зхлопував їх хвилеві функції, і тепер, навіть, якщо їх пересилають одержувачу, послідовність вже зовсім не та, яку посилав відправник, а саме та, яку отримав криптоаналітик у процесі своїх вимірювань.

Одержувач знову вимірює отримані ним фотони, випадковим чином застосовуючи свої прилади:

+	×	+	+	×	×	×	+	+	×	×	×	×	×	+	+
|	\	–	|	/	\	\	–	–	\	/	/	|	|	–	|

Після цього він дзвонить відправнику і вони звіряють використані прилади. Як і у випадку без криптоаналітика між ними, одержувач викреслює ті біти, для отримання яких він використовував неправильний прилад:

	\	–	|			\		–				|	|		|
	1	0	1			1		0				1	1		1

У одержувача вийшов рядок **10110111**, тоді як у відправника залишився рядок **00110101**. Як видно, два біти розрізняються, і це саме те, що обумовлено діями криптоаналітика з перехоплення. Для перевірки відправник і одержувач мають вибрати в своїх рядках деяку кількість бітів (наприклад, k) і порівняти їх у відкритому каналі. Якщо всі біти збіглися, то з вірогідністю $1 – 2^{-k}$ канал не прослуховувався.

Запитання для самоперевірки

1. Які існують способи тайної передачі інформації ?
2. У чому полягає різниця між стеганографією та криптографією ?
3. Для чого використовується криптографія ?
4. Що було характерним для раннього етапу розвитку криптографії ?
5. Які типи шифрів використовувалися на ранньому етапі розвитку криптографії ?
6. Які зміни у розвитку криптографії відбулися з появою комп’ютерів ?
7. У чому полягають особливості квантової криптографії ?

Лекція № 9

КРИПТОГРАФІЧНІ АЛГОРИТМИ І КЛЮЧІ

Навчальні цілі:
- вивчити види шифрів (криптографічних алгоритмів);
- розглянути поняття криптографічного алгоритму.

Навчальні питання:
1. Поняття криптографічної системи.
2. Види шифрів.
3. Основи криптоаналізу.

9.1 Поняття криптографічної системи

Криптографічна система (криптосистема) являє собою криптографічний алгоритм із всіма можливими відкритими текстами, шифротекстами і ключами.

Мета криптографічної системи полягає у тому, щоб зашифрувати осмислений початковий текст (відкритий текст), отримавши в результаті цілком безглуздий шифрований текст (шифротекст, криптограму). Одержувач, якому він призначений, має бути здатним розшифрувати (дешифрувати) цей шифротекст, відновивши, таким чином, відповідний йому відкритий текст. При цьому супротивник (криптоаналітик) має бути нездатним розкрити початковий текст.

У наш час дуже часто використовується поняття (термін) «хакер». Найближчим значенням терміна **«хакер»** є слова **трудяга**, **поденник**, **найманий робітник**. Хакерами ще деколи називають журналістів, що пишуть скандальні статті за замовленням. Цей термін почали вперше використовувати у Масачусетському технологічному інституті на початку сімдесятих років. Так називали молодих програмістів і проектувальників апаратних засобів комп’ютерів, які у гаражах і підвалах майстрували перші персональні комп'ютери і навіть намагалися продавати їх. Пізніше хакерами почали називати **всіх комп'ютерних злочинців**.

Розкриття криптосистеми – це результат роботи криптоаналітика, що приводить до можливості ефективного розкриття будь-якого зашифрованого за допомогою даної криптосистеми відкритого тексту. Неможливість розкриття криптосистеми називається її **стійкістю**.

Криптографічний алгоритм (**шифр**) є математичною функцією, яка використовується для шифрування і дешифрування.

Якщо безпека алгоритму основана на його збереженні у таємниці, то такий алгоритм називають **обмеженим**. Велика група користувачів або група, яка часто змінюється, не може використовувати такі алгоритми, оскільки всякий раз, коли користувач покидає групу, її члени мають

застосовувати інший алгоритм. Його також слід заміняти, якщо хто-небудь ззовні випадково дізнається про нього.

Обмежені алгоритми не допускають якісного контролю або стандартизації. У кожної групи користувачів має бути свій унікальний алгоритм. Ці алгоритми незвичайно популярні у додатках з низьким рівнем безпеки. Користувачі або не розуміють проблем, пов'язаних з безпекою своїх систем, або не піклуються про них.

У сучасній криптографії вирішують ці проблеми **за допомогою ключа**, що може мати будь-яке значення, вибране з великої множини. Безпека цих алгоритмів повністю основана на ключах, а не на деталях алгоритмів. Це означає, що алгоритм може бути опублікованим і проаналізованим. Незважаючи на те, що зловмисникові відомий алгоритм, він не зможе прочитати повідомлення, якщо не знає значення конкретного ключа.

Залежно від кількості використовуваних ключів розрізняють два методи шифрування і дешифрування: з одним ключем (рисунок 9.1); з двома різними ключами (рисунок 9.2).

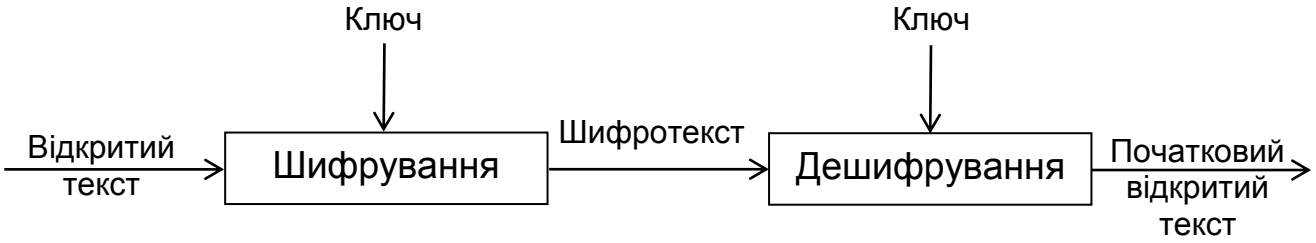

Рисунок 9.1 – Шифрування і дешифрування з одним ключем

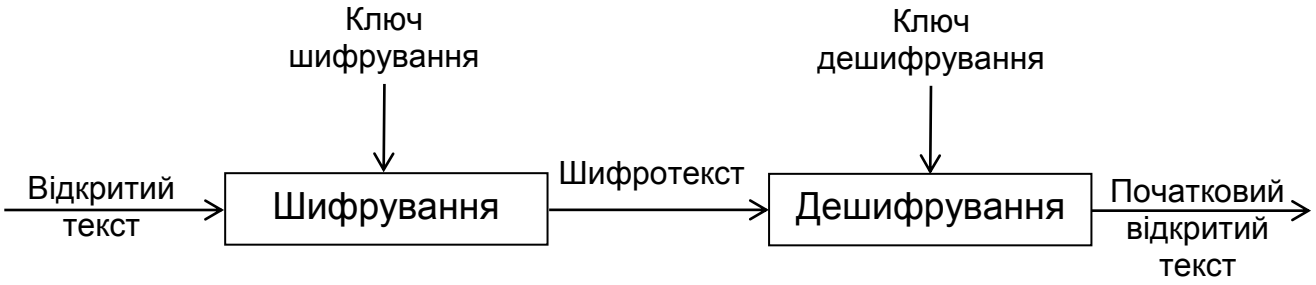

Рисунок 9.2 – Шифрування і дешифрування з двома різними ключами

Як видно з рисунка 9.1, для шифрування і дешифрування можна використовувати однаковий ключ, відправник і одержувач мають узгодити його значення перед початком обміну даними. Схема, яка зображена на рисунку 9.2, вирізняється тим, що відправник і одержувач можуть не знати ключі один одного, а знати тільки свій ключ і здійснювати шифрування або дешифрування.

9.2 Види шифрів

Усі відомі способи шифрування можна поділити на п'ять груп:

- підстановка (заміна);
- перестановка;
- аналітичне перетворення;
- гамірування;
- комбіноване шифрування.

Навіть у дуже складних шифрах як типові компоненти (складові частини) використовують прості шифри, наприклад, шифри заміни, перестановки або їх поєднання.

Шифром підстановки (заміни) здійснюють заміну букв або частин відкритого тексту на аналогічні частини шифрованого тексту. Шифр заміни математично можна описати таким чином: нехай існують два алфавіти X і Y – відкритого і шифрованого текстів відповідно, що складаються з однакового числа символів. Існує також взаємооднозначне відображення X в Y – g: X→Y.

При використанні шифру підстановки (заміни) відкритий текст $x_1x_2…x_n$ перетвориться у шифрований текст $g(x_1)g(x_2)…g(x_n)$.

Шифром перестановки здійснюють перестановку букв у відкритому тексті. Зазвичай відкритий текст розбивають на однакові частини і кожну з них шифрують незалежно. Нехай, наприклад, довжина частини дорівнює n, а σ – взаємооднозначне відображення множини {1, 2, …, n} в себе.

При застосуванні шифру перестановки частина відкритого тексту $x_1x_2…x_n$ перетвориться у частину шифрованого тексту $\sigma(x_1)\sigma(x_2)…\sigma(x_n)$.

Типовим і найбільш простим прикладом реалізації абсолютно стійкого шифру є **шифр гамірування (Вернама)**. Цим шифром здійснюється побітове складання n-бітового відкритого тексту і n-бітового ключа:

$$y_i = x_i \oplus k_i, \quad i = 1, …, n,$$

де $x_1…x_n$ – відкритий текст;

$k_1…k_n$ – ключ;

$y_1…y_n$ – шифрований текст.

Для підтримування абсолютної стійкості шифру необхідно виконувати такі вимоги:

- зберігати повну випадковість ключа (це, зокрема, означає, що ключ не можна виробляти за допомогою якого-небудь детермінованого пристрою);
- дотримувати рівність довжини ключа і довжини відкритого тексту;
- одноразово використовувати ключ.

У разі порушення хоч би однієї з цих умов шифр перестає бути абсолютно стійким і з'являються принципові можливості для його розкриття.

Однак саме ці умови і роблять абсолютно стійкий шифр дуже дорогим і непрактичним. Перш ніж користуватися таким шифром необхідно забезпечити всіх абонентів достатнім запасом випадкових ключів і виключити можливість їх повторного застосування. А це зробити незвичайно важко і дорого.

9.3 Основи криптоаналізу

Сенс криптографії полягає у збереженні відкритого тексту (або ключа, або того і іншого) у таємниці від зловмисників. При цьому передбачається, що зловмисники повністю контролюють лінії зв'язку між відправником і одержувачем.

Криптоаналіз – це наука отримання відкритого тексту, не маючи ключа. Успішно проведений криптоаналіз може розкрити відкритий текст або ключ. Він також може виявити слабкі місця у криптосистемах.

Спроба криптоаналізу називається **розкриттям.** Передбачається, що криптоаналітик знає все про використовуваний алгоритм шифрування.

Розкриття алгоритмів можна поділити за такими категоріями:

- **повне розкриття.** Криптоаналітик отримав ключ K, такий, що $D_K(C) = P$, де D – алгоритм дешифрування, C – шифротекст, P – відкритий текст;
- **глобальна дедукція**. Криптоаналітик одержав альтернативний алгоритм дешифрування A, еквівалентний $D_K(C)$ без знання ключа K;
- **місцева (або локальна) дедукція**. Криптоаналітик отримав відкритий текст P для перехопленого шифротексту C;
- **інформаційна дедукція.** Криптоаналітик одержав деяку інформацію про ключ K або відкритий текст P. Такою інформацією можуть бути декілька бітів ключа, зведення про форму відкритого тексту і т. д.

Різні алгоритми надають різні ступені безпеки залежно від того, наскільки важко розкрити алгоритм.

Алгоритм є безумовно безпечним, якщо, незалежно від обсягу шифротекстів у криптоаналітика, інформації для отримання відкритого тексту недостатньо. По суті тільки шифрування одноразовими блокнотами неможливо розкрити при нескінченних ресурсах. Решта всіх криптосистем може бути розкрита з використанням тільки шифротексту простим перебиранням можливих ключів і перевіркою смислу отриманого відкритого тексту. Це називається **розкриттям грубою силою**.

Криптографію більше застосовують до криптосистем, які важко зламати обчислювальним способом.

Алгоритм вважають обчислювально безпечним (або, як іноді називають, сильним), якщо він не може бути розкритим з використанням доступних ресурсів зараз або у майбутньому. Термін «доступні ресурси» є достатньо розпливчастим.

Під складністю розкриття розуміють:

- складність даних. Обсяг даних, які використовують на вході операції розкриття;

- вимоги до пам'яті. Обсяг пам'яті, необхідної для розкриття;

- складність оброблення. Час, потрібний для проведення розкриття, часто називають **коефіцієнтом роботи**.

Оголошення алгоритму безпечним просто тому, що його нелегко розкрити, використовуючи сучасну техніку, є ненадійним. Надійні криптосистеми проектують стійкими до розкриття з урахуванням розвитку обчислювальних засобів на багато років уперед.

Розкриття з використанням тільки шифротексту. У криптоаналітика є шифротексти декількох повідомлень, зашифрованих одним і тим же алгоритмом шифрування. Завдання криптоаналітика полягає у розкритті відкритого тексту як можна більшого числа повідомлень або, ще краще, отриманні ключа, який використовується для шифрування повідомлень, для дешифрування інших повідомлень, зашифрованих тим же ключем. Цей метод можна описати математично.

Дано: $C_1 = E_k(P_1)$, $C_2 = E_k(P_2)$, ..., $C_i = E_k(P_i)$.

Знайти: P_1, P_2, ..., P_i або k, або алгоритм отримування P_{i+1} з $C_{i+1} = E_k(P_{i+1})$.

Розкриття з використанням відкритого тексту. У криптоаналітика є доступ не тільки до шифротекстів декількох повідомлень, але і до відкритого тексту цих повідомлень. Його завдання полягає в отриманні ключа для шифрування повідомлень, дешифрування інших повідомлень, зашифрованих тим же ключем. Цей метод можна описати математично.

Дано: P_1, $C_1 = E_k(P_1)$; P_2, $C_2 = E_k(P_2)$; ...; P_i, $C_i = E_k(P_i)$.

Знайти: k або алгоритм отримування P_{i+1} з $C_{i+1} = E_k(P_{i+1})$.

Розкриття з використанням вибраного відкритого тексту. У криптоаналітика не тільки є доступ до шифротекстів і відкритих текстів декількох повідомлень, але і можливість вибирати відкритий текст для шифрування. Це надає більше варіантів ніж розкриття з використанням відкритого тексту, оскільки криптоаналітик може вибирати шифровані блоки відкритого тексту, що може містити більше інформації про ключ. Його завдання полягає в отриманні ключа, який використовується для шифрування повідомлень або алгоритму, що дозволяє дешифрувати нові повідомлення, зашифровані тим же ключем. Цей метод можна описати математично.

Дано: P_1, $C_1 = E_k(P_1)$; P_2, $C_2 = E_k(P_2)$; ...; P_i, $C_i = E_k(P_i)$,

де криптоаналітик може вибирати P_1, P_2, ..., P^i.

Знайти: k або алгоритм отримування P_{i+1} з $C_{i+1} = E_k(P_{i+1})$.

Адаптивне розкриття з використанням відкритого тексту. Це окремий випадок розкриття з використанням вибраного відкритого тексту. Криптоаналітик не тільки може вибирати шифрований текст, але також може формувати свій подальший вибір на базі отриманих результатів

шифрування. Під час розкриття з використанням вибраного відкритого тексту криптоаналітик міг вибрати для шифрування тільки один великий блок відкритого тексту, під час адаптивного розкриття з використанням вибраного відкритого тексту він може вибрати менший блок відкритого тексту, потім вибрати наступний блок, використовуючи результати першого вибору і т. д.

Розкриття з використанням вибраного ключа. Такий тип розкриття означає не те, що криптоаналітик може вибирати ключ, а що у нього є деяка інформація про зв'язок між різними ключами.

Бандитський криптоаналіз. Криптоаналітик загрожує, шантажує або катує кого-небудь, поки не отримає ключ. Хабарництво іноді називається розкриттям з купівлею ключа. Ці методи також використовуються.

Розкриття з відомим відкритим текстом та із застосуванням вибраного відкритого тексту зустрічаються частіше, ніж можна уявити. Не є неможливим для криптоаналітика добути відкритий текст шифрованого повідомлення або підкупити кого-небудь, хто зашифрує вибране повідомлення. Багато повідомлень мають стандартні початок і закінчення, що може бути відомо криптоаналітику. Особливо вразливий шифрований початковий код через часте використання ключових слів: else, return. Ті ж проблеми має і шифрований виконуваний код: функції, циклічні структури і т. д.

Криптографічний алгоритм можна вважати відносно безпечним у таких випадках:

- якщо вартість розкриття алгоритму вища, ніж вартість зашифрованих даних;
- час розкриття алгоритму більший, ніж час, протягом якого зашифровані дані мають зберігатися у таємниці;
- обсяг даних, зашифрованих одним ключем, менше, ніж обсяг даних, необхідних для розкриття алгоритму.

Запитання для самоперевірки

1. Що таке криптографічний алгоритм ?
2. У чому полягає особливість обмеженого криптографічного алгоритму ?
3. Які існують способи шифрування ?
4. У чому полягає різниця між шифром підстановки (заміни) і шифром перестановки ?
5. Що таке криптоаналіз ?
6. Які існують методи розкриття криптографічних алгоритмів ?
7. Коли можна вважати криптографічний алгоритм відносно безпечним ?

Лекція № 10

СИМЕТРИЧНІ АЛГОРИТМИ ШИФРУВАННЯ

Навчальні цілі:
- розглянути симетричні алгоритми шифрування;
- вивчити алгоритми шифрування DES і AES.

Навчальні питання:
1. Алгоритм шифрування DES.
2. Алгоритм шифрування AES.

10.1 Алгоритм шифрування DES

Симетричними алгоритмами (умовними, алгоритмами з секретним ключем, алгоритмами з одним ключем) називаються алгоритми, в яких ключ шифрування може бути розрахований щодо ключа дешифрування і навпаки (відправник і одержувач погоджують ключ перед початком безпечної передачі повідомлень). Безпека симетричного алгоритму визначається ключем, розкриття ключа означає, що хто завгодно зможе шифрувати і дешифрувати повідомлення.

Шифрування і дешифрування з використанням симетричного алгоритму позначається як

$$Ek(P) = C; \qquad Dk(C) = P.$$

Симетричні алгоритми поділяють на дві категорії:

- ті, що обробляють відкритий текст побітно (іноді побайтно), вони називаються **потоковими алгоритмами (шифрами)**;

- ті, що працюють з групами бітів відкритого тексту. Групи бітів називаються блоками, а алгоритми – **блоковими алгоритмами (шифрами)**.

Поточне покоління **блокових шифрів** працює з блоками тексту завдовжки від 128 біт (16 байт). Такий шифр приймає на вхід 128-бітовий відкритий текст і видає 128-бітовий шифрований текст. Блоковий шифр є оборотним: існує функція дешифрування, яка приймає на вхід 128-бітовий шифрований текст і видає початковий 128-бітовий відкритий текст.

Практично всі блокові шифри використовують декілька послідовних застосувань слабкого блокового шифру, який називається **раундом**. Деякі з таких слабких раундів у сукупності утворюють надійний блоковий шифр. Подібні структури значно полегшують розроблення і реалізацію шифру, але й спрощують його аналіз. Більшість атак на блокові шифри починається з атаки на версії шифрів з мінімальною кількістю раундів. З подальшим вдосконаленням атаки можуть застосовуватися для нападу на шифри з все більшою і більшою кількістю раундів.

Прикладом слабкого шифру є **просте XOR** («виключне АБО», «побітове складання», «складання без перенесення», гамірування).

XOR – це звичайна операція над бітами:

$$0 \oplus 0 = 0; \qquad 0 \oplus 1 = 1; \qquad 1 \oplus 0 = 1; \qquad 1 \oplus 1 = 0.$$

Також $a \oplus a = 0; \qquad a \oplus b \oplus b = a.$

Відкритий текст підлягає операції «виключне АБО» разом із ключовим текстом для отримання шифротекста. Оскільки повторне застосування операції XOR відновлює оригінал, для шифрування і дешифрування використовується одна і та ж процедура:

$$P \oplus K = C; \qquad C \oplus K = P.$$

Цей тип шифрування легко розкривається, навіть без комп'ютера. Його злом на комп'ютері займає декілька секунд. Однак саме цей алгоритм використовується у цифрових телефонних стільникових мережах для кодування голосу.

Одним з перших симетричних алгоритмів був **алгоритм шифрування DES** (Data Encryption Standard – стандарт шифрування даних). Цей алгоритм працює з ключами розміром 56 біт і блоками розміром 64 біт. За сучасними стандартами DES – не дуже швидкий шифр, а 3DES працює ще у три рази повільніше. Незважаючи на все це, DES досі використовується у багатьох існуючих системах, але застосовувати DES або 3DES у нових розробках не рекомендується.

Процес шифрування (рисунок 10.1) складається з двох перестановок, які називають початковою і фінальною (кінцевою) перестановками, і 16 раундів Фейстеля, які позначають цифрами від 1 до 16. Кожен раунд використовує різні 48-бітові ключі, що генеруються. Кожен раунд і перетворить пару (L, R) у нову пару (L, R) за допомогою підключа K_i.

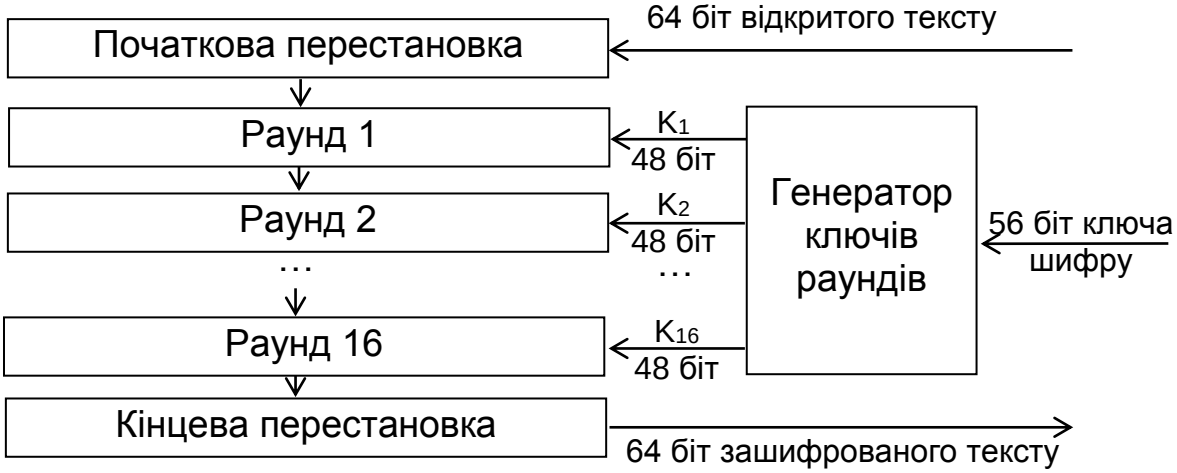

Рисунок 10.1 – Процес шифрування алгоритму DES

Початкова і кінцева перестановки. На вхід кожної з них надходить 64 біта, які потім переставляються відповідно до таблиці 10.1, біт 58 блоку стає бітом 1, біт 50 – бітом 2 і т. д.

Таблиця 10.1 – Перестановка бітів

Початкова перестановка								Кінцева перестановка							
58	50	42	34	26	18	10	2	40	8	48	16	56	24	64	32
60	52	44	36	28	20	12	4	39	7	47	15	55	23	63	31
62	54	46	38	30	22	14	6	38	6	46	14	54	22	62	30
64	56	48	40	32	24	16	8	37	5	45	13	53	21	61	29
57	49	41	33	25	17	9	1	36	4	44	12	52	20	60	28
59	51	43	35	27	19	11	3	35	3	43	11	51	19	59	27
61	53	45	37	29	21	13	5	34	2	42	10	50	18	58	26
63	55	47	39	31	23	15	7	33	1	41	9	49	17	57	25

Ці перестановки є взаємно зворотними. Іншими словами, 58-й біт на вході початкової перестановки переходить в першу позицію на виході з неї. А у фінальній перестановці 1-й вхідний біт перейде в 58 позицію на виході. Ніхто не може пояснити, навіщо розробники шифру DES вирішили переставити біти відкритого тексту – адже це не має ніякого криптографічного ефекту, але алгоритм DES визначено саме так.

Кожним раундом (рисунок 10.2) є побітове складання значення L зі значенням $F(K_i, R)$, де F – функція Фейстеля і подальший обмін місцями значень L і R.

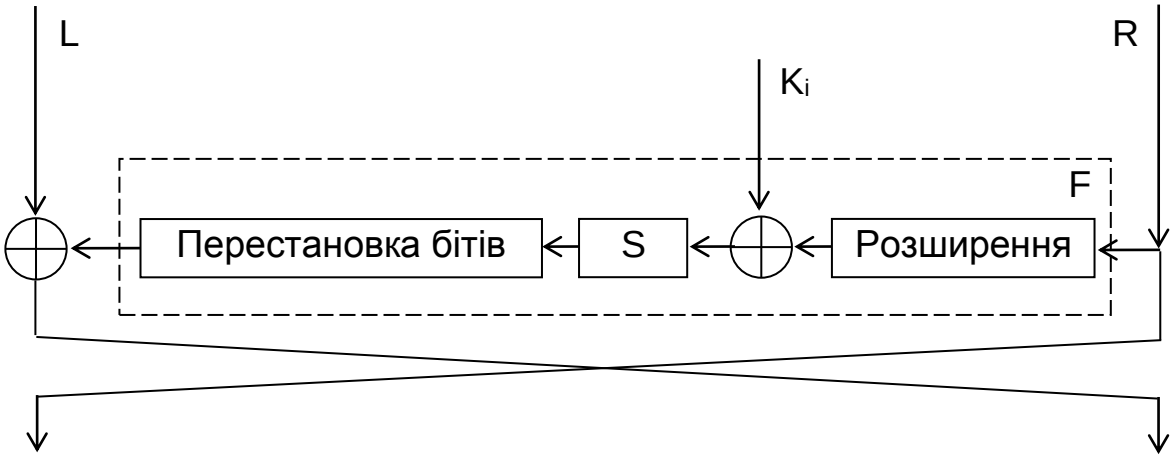

Рисунок 10.2 – Структура одного раунду DES

На вхід алгоритму DES подається 64-бітовий блок відкритого тексту, який поділяють на дві 32-бітові половини: L (ліва) і R (права). Після закінчення шифрування ліва і права половини об'єднуються і піддаються зворотній перестановці, внаслідок чого знов виходить 64-бітовий блок тексту, але цього разу шифрованого.

Спочатку до значення R застосовується функція **розширення**. Вона дублює 16 бітів значення R, внаслідок чого 32-бітове значення перетворюється на 48-бітове, щоб погоджувати його розміри з розмірами підключа раунду. Блок ділиться на 8 секцій по 4 біта. Кожна секція розширюється до 6 бітів. Число виходів – 48, діапазон значень – від 1 до 32. Деякі вхідні біти породжують декілька вихідних. Функція розширення може бути задана в таблиці 10.2.

Таблиця 10.2 – Функція розширення

Функція розширення в і-му раунді					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

Цей результат функції розширення **складається за допомогою операції XOR** з 48-бітовим підключем K_i. Кожен підключ утворюється шляхом вибору 48 бітів з 56-бітового ключа шифрування, причому для кожного раунду цей вибір виконується по-своєму. Складання тексту з підключем за допомогою операції XOR гарантує перемішування ключа і даних, в чому і полягає сенс шифрування.

Результат цієї операції подається на вхід **S-матриць**. За своєю суттю S-матриця (буква S означає substitution, тобто підстановка) – це таблиця відповідностей. Оскільки не можна побудувати таблицю відповідностей для 48-бітових даних, S-матриці складаються з восьми невеликих таблиць відповідностей (з чотирьох рядків і 16-ти стовпців), кожна з яких отримує на вхід 6-бітовий і видає 4-бітовий результат. Таким чином, після перетворення 48-бітового значення за допомогою S-матриць дані знову стають 32-бітовими. S-матриці забезпечують нелінійність. Без них процес шифрування можна було б подати у вигляді послідовності операцій двійкового складання, що дуже легко «зламати», використовуючи методи лінійної алгебри.

Після підсумовування з бітами ключа блок з 48 бітів ділиться на 8 послідовних 6-бітових векторів b1, b2, ..., b8, кожен з яких замінюється на 4-бітовий вектор за допомогою S-матриць.

Вектор b1 надходить у матрицю S1, вектор b2 – у матрицю S2 і т. д. Щоб в S-матриці знайти шифропозначення вектора, необхідно виконати такі дії (таблиця 10.3):

- з 1-го і 6-го бітів вектора утворити двійкове число, перевести його в десяткову систему. Це буде номер рядка m (m = 0,1,2,3);

- з 2-го, 3-го, 4-го і 5-го бітів вектора утворити двійкове число, перевести його в десяткову систему. Це буде номер стовпця n (n = 0...,15);

- число, що знаходиться в S-матриці на перетині m-го рядка і n-го стовпця, буде шифропозначенням b'j вектора bj;

- шифропозначення b'j перевести в двійкову систему. Відповідь готова.

Таблиця 10.3 – S-матриці

	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	
0	14 4 13 1 2 15 11 8 3 10 6 12 5 9 0 7	S1
1	0 15 7 4 14 2 13 1 10 6 12 11 9 5 3 8	
2	4 1 14 8 13 6 2 11 15 12 9 7 3 10 5 0	
3	15 12 8 2 4 9 1 7 5 11 3 14 10 0 6 13	
0	15 1 8 14 6 11 3 4 9 7 2 13 12 0 5 10	S2
1	3 13 4 7 15 2 8 14 12 0 1 10 6 9 11 5	
2	0 14 7 11 10 4 13 1 5 8 12 6 9 3 2 15	
3	13 8 10 1 3 15 4 2 11 6 7 12 0 5 14 9	
0	10 0 9 14 6 3 15 5 1 13 12 7 11 4 2 8	S3
1	13 7 0 9 3 4 6 10 2 8 5 14 12 11 15 1	
2	13 6 4 9 8 15 3 0 11 1 2 12 5 10 14 7	
3	1 10 13 0 6 9 8 7 4 15 14 3 11 5 2 12	
0	7 13 14 3 0 6 9 10 1 2 8 5 11 12 4 15	S4
1	13 8 11 5 6 15 0 3 4 7 2 12 1 10 14 9	
2	10 6 9 0 12 11 7 13 15 1 3 14 5 2 8 4	
3	3 15 0 6 10 1 13 8 9 4 5 11 12 7 2 14	
0	2 12 4 1 7 10 11 6 8 5 3 15 13 0 14 9	S5
1	14 11 2 12 4 7 13 1 5 0 15 10 3 9 8 6	
2	4 2 1 11 10 13 7 8 15 9 12 5 6 3 0 14	
3	11 8 12 7 1 14 2 13 6 15 0 9 10 4 5 3	
0	12 1 10 15 9 2 6 8 0 13 3 4 14 7 5 11	S6
1	10 15 4 2 7 12 9 5 6 1 13 14 0 11 3 8	
2	9 14 15 5 2 8 12 3 7 0 4 10 1 13 11 6	
3	4 3 2 12 9 5 15 10 11 14 1 7 6 0 8 13	
0	4 11 2 14 15 0 8 13 3 12 9 7 5 10 6 1	S7
1	13 0 11 7 4 9 1 10 14 3 5 12 2 15 8 6	
2	1 4 11 13 12 3 7 14 10 15 6 8 0 5 9 2	
3	6 11 13 8 1 4 10 7 9 5 0 15 14 2 3 12	
0	13 2 8 4 6 15 11 1 10 9 3 14 5 0 12 7	S8
1	1 15 13 8 10 3 7 4 12 5 6 11 0 14 9 2	
2	7 11 4 1 9 12 14 2 0 6 10 13 15 3 5 8	
3	2 1 14 7 4 10 8 13 15 12 9 0 3 5 6 11	

Приклад. Знайти шифропозначення вектора b1 = 101011.

Розв’язання: на вхід S1-матриці подано число 101011.

Номер рядка – $11_2 = 3_{10}$; Номер стовпця – $0101_2 = 5_{10}$.

За таблицею підстановки у S1-матриці (див. таблиця 10.3) знаходимо на перетині 3-го рядка і 5-го стовпця число $4_{10} = 0100_2$.

0100 – шифропозначення для 101011.

Таким чином, після S-матриць отримуємо вісім 4-бітових векторів b’1, b’2, ..., b’8, які знову об'єднуємо в 32-бітовий блок.

Останнє значення піддається ще одній перестановці бітів, після чого складається за допомогою операції XOR з лівою половиною L. Значення правої і лівої половини міняються місцями. Ця процедура повторюється 15 разів.

Таким чином, поєднання процедур S-матриць, функції розширення і перестановки бітів призводить до дифузії. Іншими словами, якщо змінити один біт у вхідному значенні функції F, то в її вихідному значенні зміняться відразу декілька бітів. У наступному раунді ця зміна стане ще більшою і т. д. За відсутності дифузії незначна зміна відкритого тексту приведе до незначної зміни шифрованого тексту, що можна легко відстежити.

Розшифровка складається з точно такого ж набору операцій. Необхідно поміняти місцями значення L і R і виконати побітове складання значення L з величинами $F(K_i, R)$. Це набагато спрощує реалізацію функцій шифрування і дешифрування.

Алгоритм **3DES** працює з ключем шифрування більшого розміру. На жаль, від свого попередника DES він успадкував і слабкі ключі. Крім того, 3DES має обмеження розміру блока, який не може перевищувати 64 біт. Це істотно обмежує обсяг даних, які можна зашифрувати за допомогою одного ключа.

10.2 Алгоритм шифрування AES

Крім алгоритму DES, ще одним симетричним алгоритмом, який дуже широко використовується на практиці, є **алгоритм AES**, структура якого істотно відрізняється від структури DES. Алгоритм AES не належить до шифрів Файстеля. Повний процес шифрування складається з 10 – 14 раундів залежно від розміру ключа. Як і в алгоритмі DES, підключі алгоритму AES генеруються на основі деякого ключа шифрування, але механізм генерації ключів повністю відмінний від того, що застосовувався у DES.

На рисунку 10.3 показано один раунд алгоритму AES. Подальші раунди мають аналогічну структуру.

Раунд складається з чотирьох різних перетворень:

- складання з раундовим ключем (операція XOR);
- побайтової (8 біт) підстановки в S-матриці з фіксованою таблицею замін;

- побайтового зміщення рядків S-матриці на різну кількість байтів;
- перемішування байтів в стовпцях.

На вхід алгоритму подається блок відкритого тексту завдовжки 16 байт. Відкритий текст за допомогою операції XOR підсумовується з 16-байтовим (128-бітовим) підключем (кожен байт підключа підсумовується з відповідним байтом відкритого тексту). Кожен з 16 байтів отриманого результату подається на вхід таблиці S-матриць, яка перетворює 8-бітові вхідні значення у 8-бітові вихідні значення. Всі S-матриці однакові. Отримані байти переставляють у деякому заданому порядку. На рисунку 10.3 його зображено трохи ускладненим, але насправді він має дуже просту структуру. Кожна група з чотирьох байтів піддається перемішуванню, яке здійснюється за допомогою лінійної функції перемішування (кожен біт вихідних даних отримано в результаті застосування операції XOR до декількох вхідних бітів). Використання функції перемішування завершує раунд.

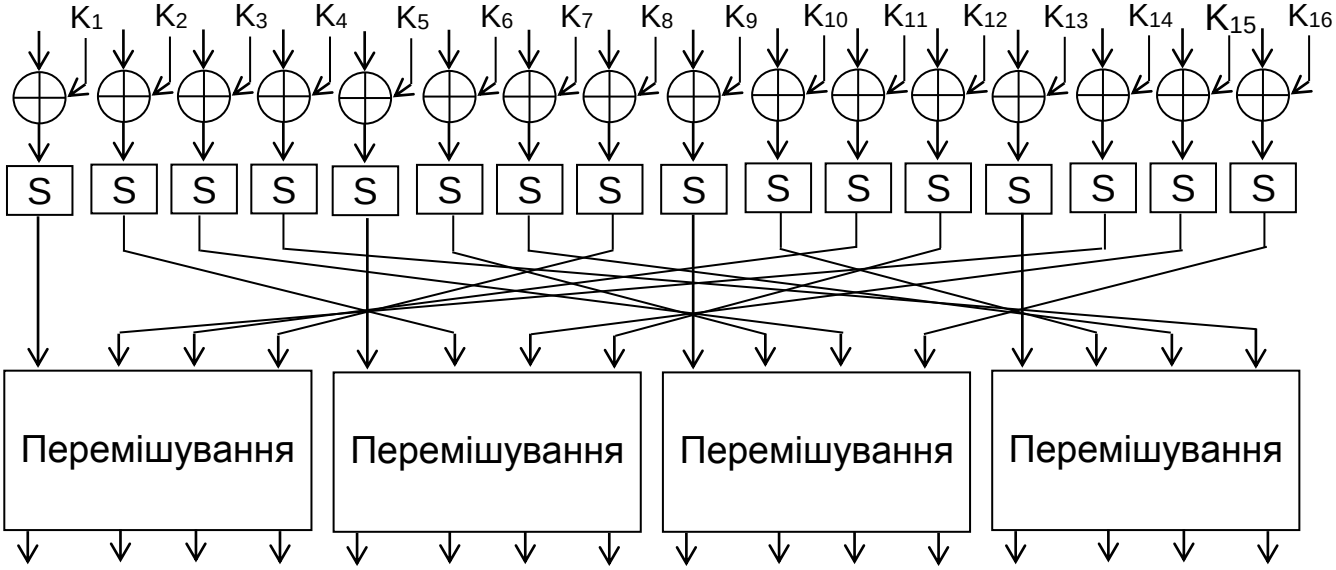

Рисунок 10.3 – Структура одного раунду AES

Основними перевагами алгоритму є:

- чітка структура, кожна частина якої виконує строго певне завдання;
- кожен крок складається з декількох операцій, які можуть виконуватися одночасно, що полегшує створення високошвидкісних реалізацій.

До недоліків алгоритму належать:

- операція дешифрування, що істотно відрізняється від операції шифрування;
- зворотні S-матриці, що використовуються для розшифровки тексту і ускладнюють цей процес;
- функція, зворотна перемішуванню, що істотно відрізняється від самої функції перемішування.

В Україні був розроблений національний криптографічний стандарт блокового симетричного перетворення ДСТУ 7624:2014, що визначає шифр «Калина» і режими його роботи для забезпечення конфіденційності і цілісності. Він підтримує розмір блока і довжину ключа шифрування 128, 256 і 512 біт (довжина ключа дорівнює розміру блока або в два рази перевищує його), забезпечуючи нормальний, високий і надвисокий рівень стійкості (зараз це єдиний в світі стандарт блокового шифрування, що підтримує 512-бітові симетричні ключі). Різні варіанти забезпечують гнучкість вибору параметрів для розробників систем криптографічного захисту, що дозволяє отримати як найвищий рівень швидкодії, так і найбільший запас стійкості перетворення.

У високорівневій конструкції використовується добре досліджена структура, яка застосовується у більшості симетричних алгоритмів, зокрема в AES. Циклове перетворення, побудоване на базі таблиць підстановки (S-матриць) і МДР-перетворення (множення/додавання/ рандомізація) над кінцевим полем, забезпечує необхідні криптографічні властивості. Кількість циклів шифрування залежить від довжини ключа: 10 циклів для 128-бітового, 14 циклів для 256-бітового і 18 циклів для 512-бітового ключа шифрування.

Порівняно з іншими симетричними алгоритмами блоковий шифр «Калина» має такі істотні конструктивні відмінності:

- початкове і кінцеве «забілення» з використанням модульного складання для підвищення складності криптоаналітичних атак;
- застосування чотирьох різних S-матриць замість однієї для захисту від атак, які забезпечують найбільшу нелінійність булевих функцій, що дає можливість одержати додатковий запас стійкості перетворення;
- збільшений розмір МДР-перетворення, що покращує криптографічні властивості і дозволяє оптимізувати швидкодію на сучасних 64-бітових платформах;
- нову однонаправлену схему формування циклових ключів, що забезпечує захист від атак, ефективність програмної і програмно-апаратної реалізації разом з додатковою стійкістю до спеціальних методів аналізу.

Запитання для самоперевірки

1. Які особливості мають симетричні алгоритми ?
2. Як здійснюється шифрування з використанням операції XOR ?
3. Які операції містить алгоритм шифрування DES ?
4. У чому полягає суть одного раунду DES ?
5. Як здійснюється перетворення 48-бітового значення у 32-бітове за допомогою S-матриць ?
6. У чому полягає суть одного раунду AES ?
7. Які особливості має український алгоритм шифрування «Калина» ?

Лекція № 11

АСИМЕТРИЧНІ АЛГОРИТМИ ШИФРУВАННЯ

Навчальні цілі:

- вивчити алгоритми з відкритим ключем та створення відкритого ключа із закритого;
- ознайомитися з алгоритмами RSA і Ель-Гамаля.

Навчальні питання:

1. Алгоритми з відкритим ключем.
2. Алгоритми RSA і Ель-Гамаля.

11.1 Алгоритми з відкритим ключем

Перший алгоритм з відкритим ключем розроблено у 1976 р. Уїтфілдом Діффі і Мартіном Хеллманом. В такому алгоритмі ключ, який використовується для шифрування, відрізняється від ключа дешифрування. Більш того, ключ дешифрування не може бути (принаймні, протягом розумного інтервалу часу) розрахованим на основі ключа шифрування. Алгоритм з відкритим ключем має такий ключ шифрування, який хто завгодно може використовувати для шифрування повідомлення, але тільки конкретна людина з відповідним ключем дешифрування може розшифрувати повідомлення. У цих системах ключ шифрування часто називається відкритим ключем, а ключ дешифрування – закритим. Шифрування з відкритим ключем K позначають як

$$E_{K1}(P) = C.$$

Хоча відкритий і закритий ключі різні, дешифрування з відповідним закритим ключем позначають як

$$D_{K2}(C) = P.$$

Іноді повідомлення шифрують закритим ключем, а дешифрують відкритим, що використовують для цифрового підпису.

Першим алгоритмом для узагальненого шифрування з відкритим ключем став **алгоритм рюкзака**. Проблема рюкзака нескладна. Маємо купу предметів різної маси. Необхідно визначити, можна чи ні покласти деякі з цих предметів у рюкзак так, щоб маса рюкзака дорівнювала певному значенню?

Цю проблему можна вирішити математично.

Дано: набір значень M_1, M_2 ..., M_n і якась сума S.

Знайти: значення b_i (може бути або нулем, або одиницею) у виразі

$$S = b_1M_1 + b_2M_2 + ... + b_nM_n.$$

Одиниця показує, що предмет кладуть у рюкзак, а нуль – що не кладуть. Наприклад, маси предметів можуть мати такі значення: $M_1 = 1$, $M_2 = 5$, $M_3 = 6$, $M_4 = 11$, $M_5 = 14$ і $M_6 = 20$. Можна упакувати рюкзак так, щоб його маса $S = 22$ складалась з мас $M_2 = 5$, $M_3 = 6$ і $M_4 = 11$. Неможливо упакувати рюкзак так, щоб його маса була $S = 24$.

В основі **алгоритму шифрування рюкзака Меркла–Хеллмана (нормального рюкзака)** лежить ідея шифрування повідомлення як вирішення проблеми рюкзака. Предмети з купи вибирають за допомогою блока відкритого тексту, який за довжиною дорівнює кількості предметів у купі (біти відкритого тексту відповідають значенням b_i, а шифротекст є отриманою сумою). Розглянемо приклад шифротексту, зашифрованого за допомогою проблеми рюкзака:

відкритий текст (P) 111001 010110 000000 011000;
рюкзак (K_1) 1 5 6 11 14 20 1 5 6 11 14 20 1 5 6 11 14 20 1 5 6 11 14 20;
шифротекст (C) 1+5+6+20=**32** 5+11+14=**30** 0 = **0** 5 + 6 = **11**.

Надзростаюча послідовність (надзростаючий рюкзак) – це послідовність, в якій кожний член більше суми всіх попередніх членів. Ця послідовність використовується для дешифрування. Послідовність {2, 3, 6, 13, 27, 52} є надзростаючою, а {2, 3, 4, 9, 15, 25} – ні.

Розв’язок надзростаючого рюкзака знайти легко.

Повну масу S слід порівняти з найбільшим числом послідовності. Якщо повна маса менша, ніж це число, то предмет не кладуть у рюкзак. Якщо повна маса більше або дорівнює цьому числу, то предмет кладуть у рюкзак.

Зменшимо масу рюкзака на це значення і перейдемо до наступного числа послідовності.

Повторюватимемо, поки процес не закінчиться.

Якщо повна маса зменшиться до нуля, то розв’язок знайдено, інакше – ні.

Наприклад, нехай повна маса рюкзака $S = 70$, а послідовність мас $M = \{2, 3, 6, 13, 27, 52\}$. Найбільша маса $52 < 70$, тому предмет масою 52 кладемо у рюкзак ($b_1 = 1$), $70 – 52 = 18$. Інша маса $27 > 18$, тому предмет масою 27 у рюкзак не кладемо ($b_2 = 0$). Наступна маса $13 < 18$, тому предмет масою 13 кладемо у рюкзак ($b_3 = 1$), $18 – 13 = 5$. Наступна маса $6 > 5$, тому предмет масою 6 не кладемо у рюкзак ($b_4 = 0$). Маса $3 < 5$, тому предмет масою 3 кладемо у рюкзак ($b_5 = 1$), $5 – 3 = 2$. Оскільки $2 = 2$, то предмет масою 2 кладемо у рюкзак ($b_6 = 1$), і повна маса зменшується до 0, що свідчить про знайдений розв’язок. Якби це був блок шифрування методом рюкзака Меркла–Хеллмана, то відкритий текст, отриманий з шифротексту $S = 70$, становив би $b_6b_5b_4b_3b_2b_1$ = **110101.**

Для здійснення ефективної комунікації (шифруваня і дешифрування) необхідно створити **відкритий ключ із закритого.** Розглянемо роботу алгоритму, не заглиблюючись у теорію чисел: щоб отримати нормальну послідовність рюкзака K_1 (відкритий ключ), візьмемо надзростаючу послідовність рюкзака K_2 (закритий ключ), наприклад, K_2 = {2, 3, 6, 13, 27, 52}, і помножимо за модулем m всі значення на число n:

$$K_1 = K_2 \cdot n \bmod m.$$

Значення модуля m має бути більше суми всіх чисел послідовності SM, наприклад, SM = 103, m > SM, m = 105. Множник має бути взаємно простим числом з модулем, наприклад, n = 31. Нормальною послідовністю рюкзака (відкритим ключем) буде:

2·31 mod 105 = 62;
3·31 mod 105 = 93;
6·31 mod 105 = 81;
13·31 mod 105 = 88;
27·31 mod 105 = 102;
52·31 mod 105 = 37.
K_1 = {62, 93, 81, 88, 102, 37}.

Надзростаюча послідовність рюкзака є закритим ключем, а нормальна послідовність рюкзака – відкритим.

Для **шифрування** повідомлення спочатку розбивається на блоки, що дорівнюють за довжиною числу елементів послідовності рюкзака. Потім, вважаючи, що одиниця указує на наявність члена послідовності, а нуль – на його відсутність, обчислюємо повні маси рюкзаків – поодинці для кожного блока повідомлення.

Наприклад, якщо повідомлення у бінарному записі має вигляд **P = 011000 110101 101110**, то шифрування послідовності рюкзака K_1 = {62, 93, 81, 88, 102, 37} відбуватиметься таким чином:

P_1 = **011000** відповідає **C_1** = 93 + 81 = **174**;
P_2 = **110101** – **C_2** = 62 + 93 + 88 + 37 = **280**;
P_3 = **101110** – **C_3** = 62 + 81 + 88 + 102 = **333.**
Шифротекстом буде послідовність **C = 174, 280, 333.**

Законний одержувач для **дешифрування** повідомлення **C** має закритий ключ: оригінальну надзростаючу послідовність K_2 = {2, 3, 6, 13, 27, 52}, а також значення n (n = 31) і m (m = 105), які використовувалися для перетворення її на нормальну послідовність рюкзака K_1.

Для дешифрування повідомлення одержувачу слід спочатку визначити n_1:

$$n \cdot n_1 \bmod m = 1.$$

Кожне значення шифротексту C помножують на $n_1 \bmod m$, а потім розподіляють за допомогою закритого ключа K_2, щоб набути значень відкритого тексту:

$$P = C \cdot n_1 \bmod m.$$

Наприклад, надзростаюча послідовність $K_2 = \{2, 3, 6, 13, 27, 52\}$, $m = 105$, $n = 31$, а шифротекстом є $C = 174, 280, 333$. У цьому випадку $n_1 = 61$ ($31 \cdot 61 \bmod 105 = 1$), а значення шифротексту C слід помножити на $61 \cdot \bmod 105$:

$P_1 = 174 \cdot 61 \bmod 105 = 9$;
$P_2 = 280 \cdot 61 \bmod 105 = 70$;
$P_3 = 333 \cdot 61 \bmod 105 = 48$.

Розподіляємо P_1, P_2, P_3 за допомогою закритого ключа K_2:

$P_1 = 9 = 3 + 6$, що відповідає **011000**;
$P_2 = 70 = 2 + 3 + 13 + 52$, що відповідає **110101**;
$P_3 = 48 = 2 + 6 + 13 + 27$, що відповідає **101110**.

Розшифрованим відкритим текстом є **P = 011000 110101 101110.**

У реальних рюкзаках необхідно розмістити не менше 250 елементів. Довжина кожного члена надзростаючої послідовності має бути на рівні 200 і 400 біт, а довжина модуля – від 100 до 200 біт. Для одержання цих значень на практиці використовують генератори випадкової послідовності. Якщо комп'ютер може перевіряти мільйон варіантів в секунду, то для перевірки всіх можливих варіантів заповнення рюкзака необхідно буде понад 10^{46} років.

11.2 Алгоритми RSA і Ель-Гамаля

Алгоритм RSA (Рівест (R), Шамір (S), Адлеман (A)). Безпека алгоритму RSA основана на трудності розкладання на множники великих чисел. Відкритий і закритий ключі є функціями двох великих (100 … 200 розрядів або навіть більше) простих чисел. Передбачено, що відновлення відкритого тексту за шифротекстом і відкритим ключем еквівалентно розкладанню на множники двох великих чисел.

Спочатку в алгоритмі здійснюється **генерування ключів:**

- використовують (вибирають) два великих випадкових простих числа **p** і **q**. Для максимальної безпеки **p** і **q** вибирають однакової довжини;
- розраховують добуток: **n** = p·q;
- обчислюють значення функції Ейлера від числа n:

$$\boldsymbol{\varphi(n)} = (p - 1)(q - 1);$$

- випадковим чином вибирають ключ шифрування **e**, такий, щоб ключ **e** і функції Ейлера **φ(n)** були взаємно простими числами. Число **e** називається **відкритою експонентою**;

- для обчислення ключа дешифрування **d** використовують розширений алгоритм Евкліда:

$$e \cdot d \bmod [(p - 1)(q - 1)] = 1.$$

Відзначимо, що **d** і **n** також мають бути взаємно простими числами.

Числа {**e, n**} – це **відкритий ключ**, а числа {**d, n**} – **закритий ключ.**

Два прості числа p і q більше не потрібні. Вони мають бути відкинутими, але не мають бути розкритими (розголошеними).

Далі здійснюється **шифрування** повідомлення:

- повідомлення M розбивають на цифрові блоки m_i, менші за n (для двійкових даних вибирають найбільший ступінь числа 2, менший n), тобто, якщо p і q – 100-розрядні прості числа, то n міститиме близько 200 розрядів. Кожному блоку повідомлення m_i слід мати близько 200 розрядів у довжину. Якщо потрібно зашифрувати фіксоване число блоків, їх можна доповнити декількома нулями зліва, щоб гарантувати, що блоки завжди будуть менші n. Зашифроване повідомлення C складатиметься з блоків c_i тієї ж самої довжини;

- зашифровують повідомлення (кожен блок m_i відкритого тексту) з використанням відкритого ключа {**e, n**} за допомогою виразу

$$c_i = E(m_i) = m_i^e \cdot \bmod n.$$

Для **дешифрування** необхідно мати:

- зашифроване повідомлення C (блоки c_i);
- закритий ключ {**d, n**}, який був згенерований;
- кожен зашифрований блок c_i розшифровують за допомогою виразу

$$m_i = D(c_i) = c_i^d \cdot \bmod n.$$

Процес роботи алгоритму RSA показано на рисунку 11.1.

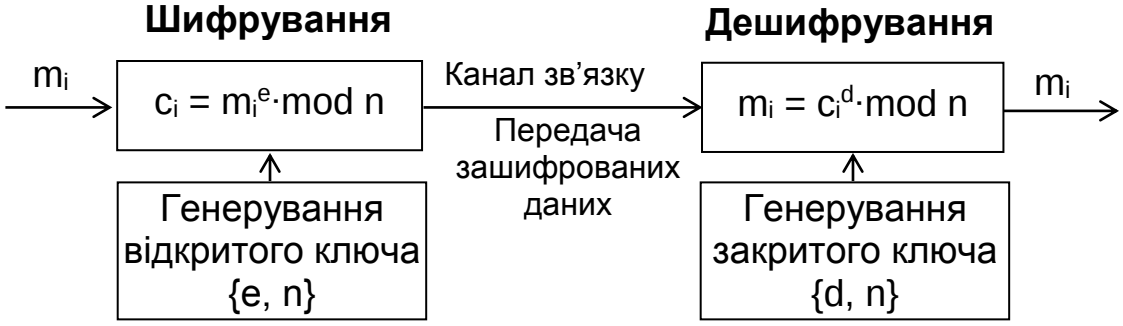

Рисунок 11.1 – Процес шифрування і дешифрування алгоритмом RSA

Повідомлення може також бути зашифровано за допомогою закритого ключа d, а розшифровано за допомогою відкритого ключа e. Цей процес використовується під час створення цифрового підпису.

Розглянемо приклад шифрування і дешифрування алгоритмом RSA.

Спочатку генеруємо ключі. Для цього вибираємо два випадкових числа: p = 47 і q = 71.

Обчислюємо n = p·q = 47·71 = 3337.

Ключ **e** не повинен мати загальних множників з функцією Ейлера:

$$\varphi(n) = (p - 1)\cdot(q - 1) = 46\cdot 70 = 3220.$$

З урахуванням цього виберемо випадкове число **e**, нехай **e** = 79.

Числа **{e, n} = {79, 3337}** є **відкритим ключем**.

Знаходимо число **d**, таке, що e·d mod [(p – 1)·(q – 1)] = 1;

$$79\cdot d \bmod 3220 = 1, \qquad 79\cdot 1019 \bmod 3220 = 1.$$

Числа **{d, n} = {1019, 3337}** є **закритим ключем**.

Опублікуємо e і n, зберігши у таємниці d. Відкинемо p і q.

Для **шифрування** повідомлення M = 688 232 687 966 668 З спочатку розподілимо його на маленькі блоки, наприклад, трибуквені блоки. Повідомлення розбиваємо на шість блоків m_i:

$$m_1 = 688;\ m_2 = 232;\ m_3 = 687;\ m_4 = 966;\ m_5 = 668;\ m_6 = 003.$$

Перший блок шифруємо відкритим ключем e = 79, n = 3337, як

$$c_1 = m_1^e \bmod n = 688^{79} \bmod 3337 = 1570.$$

Виконуючи ті ж операції з іншими блоками, створюємо шифротекст повідомлення: C = 1570 2756 2091 2276 2423 158.

Для **дешифрування** потрібно виконати таке ж піднесення до степеня, використовуючи ключ дешифрування d = 1019, n = 3337:

$$m_1 = c_1^d \bmod n = 1570^{1019} \bmod 3337 = 688.$$

Аналогічно розшифровуємо частину повідомлення, що залишилася.

Алгоритм Ель-Гамаля можна використовувати для формування електронного підпису або для шифрування даних. Він базується на трудності обчислення дискретного логарифма.

Спочатку в алгоритмі здійснюють **генерування ключів:**

- генерують випадкове просте число **p** довжиною n бітів;
- вибирають випадковим чином примітивне число **g**;
- вибирають випадкове ціле число **x** таке, що 1 < x < p – 1;

- обчислюють число

$$y = g^x \bmod p.$$

Відкритим ключем є трійка чисел **{y, g, p}**, **закритим ключем** – число **x**.

Далі здійснюють **шифрування** повідомлення M:

- вибирають **сесійний ключ** – випадкове ціле число **k** таке, що $1 < k < p - 1$;
- обчислюють числа

$$a = g^k \bmod p \quad \text{і} \quad b = M \cdot y^k \bmod p;$$

- пара чисел **(a, b)** є шифротекстом.

Довжина шифротексту в алгоритмі Ель-Гамаля вдвічі довша за початкове повідомлення.

Для **дешифрування** необхідно:

- знати закритий ключ x;
- обчислити початкове повідомлення за формулою

$$M = b \cdot (a^x)^{-1} \bmod p.$$

Для виконання обчислень краще користуватися формулою

$$M = b \cdot (a^x)^{-1} \bmod p = b \cdot a^{(p-1-x)} \bmod p.$$

Процес роботи алгоритму Ель-Гамаля показано на рисунку 11.2.

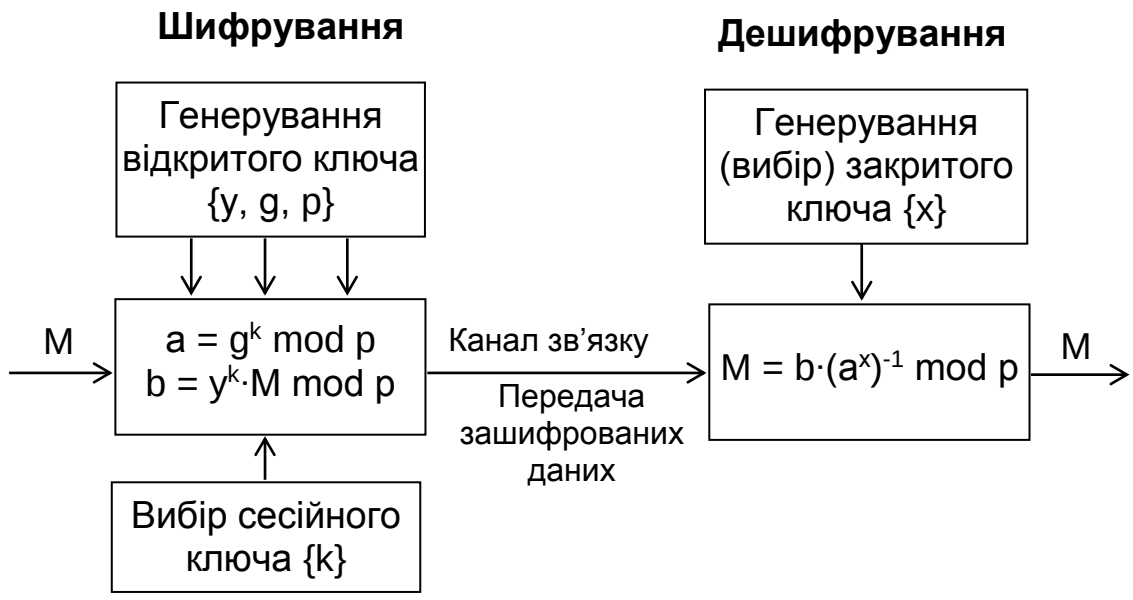

Рисунок 11.2 – Процес шифрування і дешифрування алгоритмом Ель-Гамаля

Розглянемо приклад шифрування і дешифрування алгоритмом Ель-Гамаля.

Спочатку **генеруємо ключі**. Для цього вибираємо випадкові числа: p = 11, g = 2 та випадкове ціле число таке, що 1 < x < p, наприклад, x = 8.

Обчислюємо число $y = g^x \bmod p = 2^8 \bmod 11 = 3$.

Відкритим ключем є трійка {**p, g, y} = {11, 2, 3}**, а закритим ключем є число **x = 8.**

Для **шифрування** повідомлення M = 5 спочатку вибираємо випадкове ціле число k таке, що 1 < k < (p – 1) – сесійний ключ. Припустимо вибираємо **k = 9**.

Обчислюємо перше число шифротексту

$$a = g^k \bmod p = 2^9 \bmod 11 = 512 \bmod 11 = 6.$$

Обчислюємо друге число шифротексту

$$b = M \cdot y^k \bmod p = 5 \cdot 3^9 \bmod 11 = 5 \cdot 19683 \bmod 11 = 98415 \bmod 11 = 9.$$

Отримана пара **{a, b} = {6, 9}** є шифротекстом.

Для **дешифрування** потрібно отримати шифротекст **{6, 9}** і знати закритий ключ **x = 8.**

Обчислюємо M за формулою

$$M = b \cdot a^{(p-1-x)} \bmod p = 9 \cdot 6^{(11-1-8)} \bmod 11 = 9 \cdot (6)^2 \bmod 11 = 324 \bmod 11 = 5.$$

Отримуємо початкове повідомлення **M = 5.**

Запитання для самоперевірки

1. Що таке алгоритм шифрування з відкритим ключем ?
2. Як здійснюється шифрування алгоритмом нормального рюкзака ?
3. Як створюється відкритий ключ із закритого ?
4. Як здійснюється дешифрування алгоритмом надзростаючого рюкзака ?
5. Як відбувається генерація ключів у алгоритмі RSA ?
6. Як відбувається генерація ключів у алгоритмі Ель-Гамаля ?
7. У чому полягає різниця між алгоритмами RSA і Ель-Гамаля ?

Лекція № 12

СТАНДАРТ ЦИФРОВОГО ПІДПИСУ

Навчальні цілі:

- розглянути основні поняття і напрямки розвитку технології цифрового підпису;
- вивчити моделі цифрового підпису.

Навчальні питання:

1. Основні поняття технології цифрового підпису.
2. Моделі цифрового підпису.

12.1 Основні поняття технології цифрового підпису

Одним із засобів, що забезпечує реалізацію функцій аутентифікації, цілісності і причетності, є механізм цифрового підпису.

Одним з найдавніших застосувань цифрових підписів було спрощення перевірки дотримання договорів про ядерні випробування. Сполучені Штати і Радянський Союз дозволили один одному розмістити на чужій території сейсмографи для стеження за ядерними випробуваннями. Проблема була в тому, що кожна із сторін мала бути впевненою в тому, що інша сторона не підроблювала даних цих сейсмографів. Одночасно інша сторона мала бути впевненою, що ці датчики посилають тільки ту інформацію, яка потрібна для стеження за ядерними випробуваннями. Сторона, на території якої розміщувався сейсмограф, може прочитати, але не змінити дані сейсмографа, а протилежна сторона знає, що дані не були підроблені.

В електронній кореспонденції також є розумним використання підпису перед шифруванням.

Цифровий підпис (ЦП) є рядком даних, які залежать від деякого секретного параметра (ключа), відомого тільки підписуючій особі, і від змісту підписуваного повідомлення, поданого в цифровому вигляді. Таким чином, цифровий підпис пов'язує повідомлення з деяким об'єктом або особою, підписуючою його. Цей підпис обов’язково має бути достовірним, непідробленим, його не можна використовувати повторно, а підписаний документ не можна змінити.

Звичайно ж для запобігання повторному використанню повідомлень з цим ЦП слід указати дату і час їх підписання (мітки часу), що додаються до документа разом зі всім змістом повідомлення.

Існує велика кількість алгоритмів цифрового підпису. Всі вони є алгоритмами з відкритими ключами і закритою частиною для підпису документів і з відкритою – для перевірки підпису. Цифровим підписам присвячено декілька стандартів. Велика кількість нормативних документів

ще раз указує на те, що цифровий підпис є одним з найбільш важливих механізмів безпеки.

На практиці алгоритми з відкритим ключем часто недостатньо ефективні для підпису великих документів. Для економії часу протоколи цифрового підпису нерідко використовують разом з односпрямованими хеш-функціями і підписують не документ, а значення хеш-функції для даного документа. У цьому протоколі односпрямована хеш-функція і алгоритм цифрового підпису узгоджуються заздалегідь. До переваг такого підходу належать: по-перше, підпис може бути відокремлений від документа, а по-друге, значно зменшуються вимоги до обсягу пам'яті одержувача, де зберігаються документи і підписи. В архівній системі можна використовувати цей протокол для підтвердження існування документів, не зберігаючи їх змісту. Якщо в майбутньому виникне яка-небудь розбіжність з приводу наявності автора і часу створення документа, користуючись базою даних, можна вирішити проблему за допомогою значення хеш-функції, що зберігається в базі даних.

Швидкість помітно зростає, оскільки вірогідність отримання для двох різних документів однакового 160-бітового значення хеш-функції становить тільки один шанс з 2^{160}. Можна безпечно порівняти підписи значення хеш-функції і документа. Слід використовувати тільки односпрямовану хеш-функцію. Інакше створити різні документи з одним і тим же значенням хеш-функції неважко, і підпис одного документа приведе до помилкового підпису відразу багатьох документів.

Для опису процесів оброблення інформації з використанням механізмів ЦП скористаємося наведеною нижче термінологією.

Алгоритм генерації ЦП – це метод формування ЦП.

Алгоритм перевірки (верифікації) ЦП – метод перевірки автентичного підпису, тобто дійсного і не модифікованого при передачі.

Схема ЦП (або механізм ЦП) – сукупність взаємозв'язаних алгоритмів генерування і верифікації цифрового підпису.

Процес (процедура) накладення ЦП – сукупність математичного алгоритму генерування ЦП і методів подання (форматування) підписуваних даних.

Процес (процедура) зняття ЦП – сукупність алгоритмів верифікації ЦП і методів відновлення даних.

Для побудови схеми ЦП необхідно визначити два алгоритми: **генерування ЦП** і **верифікації ЦП**. Алгоритм верифікації доступний для всіх потенційних одержувачів підписаних повідомлень, тоді як алгоритм генерування відомий тільки підписуючій особі, яка для деякого повідомлення $m \in M$ визначає відповідний підпис $s \in S$. Веріфікатор, отримавши пару (m, s) і деяку відкриту інформацію про підписуючу особу, застосовує відповідний алгоритм верифікації ЦП. Цей алгоритм видає двійковий результат: «так», якщо підпис вірний (автентичний), і «ні» – інакше.

Схеми ЦП, що існують на сьогоднішній день, поділяють на два класи (рисунок 12.1):

- з відновленням повідомлення;
- з додаванням.

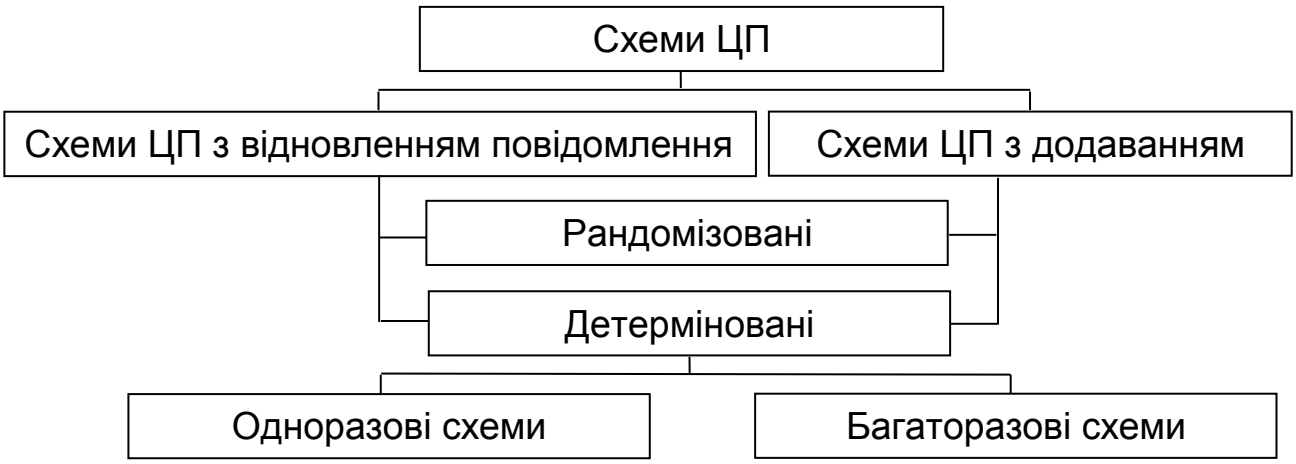

Рисунок 12.1 – Класифікація схем ЦП

У **схемах ЦП з відновленням повідомлення** все або частина підписаного повідомлення може бути відновлена безпосередньо з цифрового підпису. Таким чином, на вхід алгоритму верифікації надходить лише цифровий підпис S.

У **схемах ЦП з додаванням** цифровий підпис приєднується до повідомлення і у такому вигляді відправляється адресатові. Для верифікації такого цифрового підпису необхідно мати і підпис S, і відповідне повідомлення m.

Кожна з цих схем може бути детермінованою або рандомізованою. Застосування **детермінованих** схем характеризується тим, що цифровий підпис одного і того ж вхідного рядка даних приводить до формування однакових цифрових підписів. У **рандомізованій** схемі при генерації підпису використовується деякий випадковий параметр (число), що формує різні підписи для однакових вхідних рядків (при використанні одних і тих же ключів). У рандомізованих схемах не передбачено випадкових чисел.

Детерміновані схеми поділяють на схеми цифрового підпису одноразового застосування і схеми цифрового підпису багатократного застосування. Різниця цих цифрових підписів зрозуміла з їх назви.

12.2 Моделі цифрового підпису

До **моделі цифрового підпису з додаванням** належать такі схеми підпису, як RSA, Ель-Гамаля, Schnorr та ін.

Припустимо необхідно підписати деяке повідомлення довільної довжини $m \in M$, де M – простір повідомлень. Попереднє повідомлення m хешується з використанням однобічної і вільної від колізій хеш-функції

$$\tilde{m} = h(m); \quad \tilde{m} \in M_h,$$

де M_h – простір хеш-кодів.

Нагадаємо, що **властивість однобічності** означає, що на основі такого довільного рядка y (хеш-коду) неможливо обчислюванням знайти двійковий рядок x такий, що $h(x) = y$, незважаючи на те, що такий рядок існує і в загальному випадку він не один. **Властивість вільності від колізій** означає, що обчислюванням неможливо знайти два рядки $x \cup x'$ таких, що $h(x) = h(x')$, незважаючи на те, що такі пари рядків існують.

Використовуючи особистий ключ $k \in K$, де K – простір ключів, вибраний алгоритм генерування підпису $SIG(\bullet)$ і хеш-код повідомлення $\tilde{m}$, відправник генерує підпис $s \in S$, де S – простір підписів:

$$s = SIG_k(\tilde{m}).$$

Повідомлення m і підпис s, що додається до нього, відправляється одержувачеві.

Одержувач верифікує підпис таким чином: отримує в своє розпорядження автентичну копію ключа верифікації $k_v \in K'$. Потім за отриманим повідомленням m обчислює хеш-код $\tilde{m} = h(m)$ і, використовуючи алгоритм верифікації $VER(\bullet)$, ухвалює рішення про істинність або помилковість підпису

$$VER_{kv}(\tilde{m}, s) \to \{істина, помилка\}.$$

До створення ЦП з додаванням ставлять такі загальні вимоги:

- алгоритм генерування $SIG(\bullet)$ будь-якого ключа $k \in K$ слід ефективно обчислювати;
- ключ верифікації підпису $k_v \in K'$ необхідно захищати від підроблення і алгоритм верифікації підпису будь-якого k_v також необхідно ефективно обчислювати;
- будь-який суб'єкт, за винятком відправника, обчислювальним шляхом не може знайти $m' \cup m$, $m' \in M$ і $S' \cup S$ такі, що застосуванням алгоритму верифікації $VER_{k_v}(\tilde{m}, s')$, де $\tilde{m} = h(m)$, приведе до значення «істина».

На рисунку 12.2 показано схему моделі ЦП з додаванням.

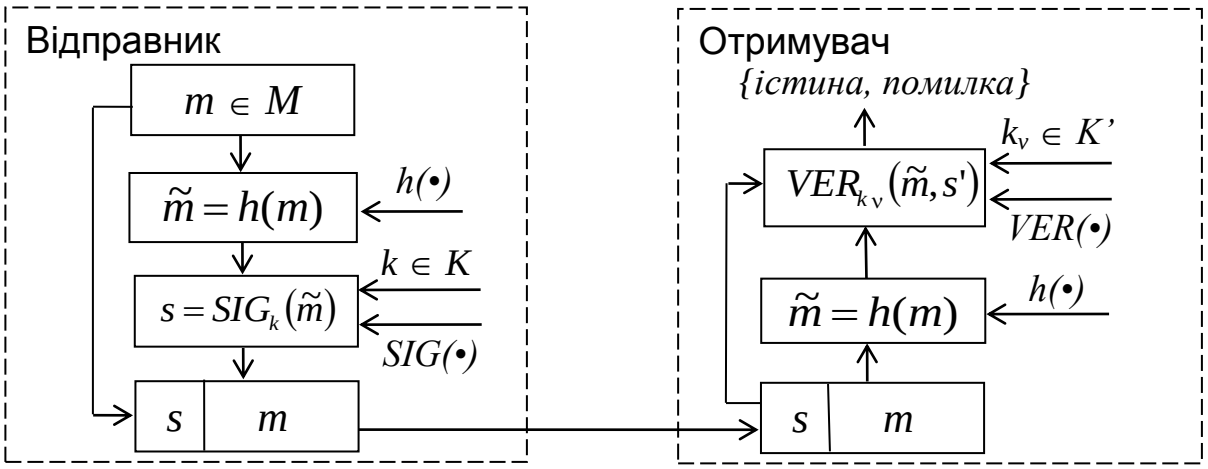

Рисунок 12.2 – Цифровий підпис з додаванням

У **моделі цифрового підпису з відновленням повідомлення** підписане повідомлення може бути відновлено безпосередньо з підпису. ЦП з відновленням звичайно використовується для роботи з короткими повідомленнями. Прикладами таких схем є схеми підпису RSA, Rabin, Nuberg-Rueppel (NR-схема).

Нехай відправник формує цифровий підпис $s \in S$ для деякого повідомлення $m \in M$. При цьому надалі одержувач може відновити повідомлення m з s. Спочатку до повідомлення m вносять надмірність шляхом застосування функції надмірності $R(\bullet)$

$$\tilde{m} = R(m); \quad \tilde{m} \in M_R,$$

де M_R – простір надмірних повідомлень, або простір надмірності.

Потім формують цифровий підпис згідно з виразом

$$s = SIG_k(\tilde{m}).$$

Цифровий підпис *s* відправляють одержувачеві.

Для верифікації одержувачу слід отримати автентичний ключ верифікації $k_v \in K'$ відправника і обчислити надмірне повідомлення щодо підпису відповідно до виразу

$$\tilde{m} = VER_{k_v}(s).$$

У випадку, якщо $\tilde{m} \notin M_R$, то підпис відкидається як недійсний, після цієї перевірки відновлюється оригінальне повідомлення m з $\tilde{m}$ шляхом обчислення:

$$m = R^{-1}(\tilde{m}).$$

На рисунку 12.3 показано процедуру накладення і верифікації ЦП з відновленням повідомлення.

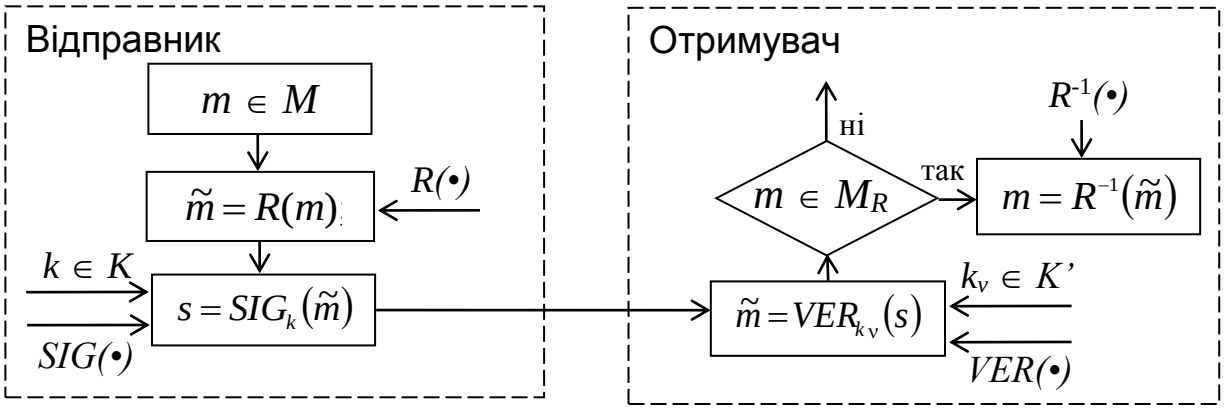

Рисунок 12.3 – Процедури накладення і верифікації ЦП з відновленням повідомлення

До створення ЦП з відновленням ставлять такі вимоги:

- алгоритм генерації $SIG_k(\bullet)$ будь-якого ключа $k_v \in K$ слід ефективно обчислювати;

- ключ верифікації підпису $k_v \in K'$ необхідно захищати від підроблення і алгоритм верифікації підпису $VER_{kv}(\bullet)$ будь-якого k_v також необхідно ефективно обчислювати;

- будь-який суб'єкт, за винятком відправника, обчислювальним шляхом не може знайти $s' \neq s$, $s' \in S$ таке, що $VER_{k_v}(s) \in M_R$.

У цифровому підписі з відновленням функція надмірності R і її інверсія R^{-1} є відкритими. Вибір відповідної функції R є критичним завданням щодо забезпечення необхідної стійкості схеми цифрового підпису. У ЦП з відновленням повідомлення потужність простору повідомлень $|m|$ менше потужності простору підписів $|S|$. Якби $|m| = |s|$, то пошук підпису s за допомогою відповідного m був би тривалим. Функцією надмірності здійснюють оборотне відображення повідомлення m в надмірне повідомлення $\tilde{m} \in M_R$. У свою чергу алгоритм генерування підпису $SIG_k(\bullet)$ застосовують до простору підписуваних повідомлень M_S, причому $|M_S| > |M_R|$, тобто $M_R \subseteq M_S$. Вважають, що безпомилкова функція надмірності має забезпечувати співвідношення потужностей таке, що $\frac{|M_R|}{|M_S|} = \left(\frac{1}{2}\right)^n$, де n – довжина підписуваного повідомлення. У схемах ЦП з

відновленням повідомлення функція надмірності є засобом виявлення модифікації цифрового підпису, тобто є аналогом хеш-функції, тільки із зворотним знаком. Функція надмірності для будь-якого повідомлення m породжує і вводить унікальну надмірність, яка залежить від вмісту інформації в повідомленні. Функція R має бути легко обчислювана і найчастіше залежати від типу алгоритму генерування підпису. Наприклад, функція введення надмірності, яка запропонована в міжнародному стандарті ISO/IEC 9796, застосовна тільки з перетвореннями в алгоритмах RSA і Rabin.

Запитання для самоперевірки

1. Що таке цифровий підпис і які вимоги ставлять до нього ?
2. На яких алгоритмах шифрування базуються алгоритми цифрового підпису ?
3. Якими є процеси (процедури) накладення та зняття цифрового підпису ?
4. У чому полягає суть алгоритмів генерування і верифікації цифрового підпису ?
5. Які існують класи схем цифрового підпису ?
6. У чому полягає суть моделі цифрового підпису з додаванням ?
7. Якою є суть моделі цифрового підпису з відновленням повідомлення ?

БІБЛІОГРАФІЧНИЙ СПИСОК

1. Аналітичний центр компанії InfoWatch [Електронний ресурс]. – Режим доступу: https://www.infowatch.ru/resources/analytics.

2. Про інформацію : Закон України від 03.04.1997 [Електронний ресурс]. – Режим доступу: https://zakon.rada.gov.ua/laws/show/2657-12.

3. Про захист інформації в інформаційно-телекомунікаційних системах : Закон України від 18.04.2006 [Електронний ресурс]. – Режим доступу: https://zakon.rada.gov.ua/laws/show/80/94-вр.

4. Завгородний, В. И. Комплексная защита информации в компьютерных системах / В. И. Завгородний. – М. : Логос, 2001. – 264 с.

5. Кузнецов, О. О. Захист інформації в інформаційних системах / О. О. Кузнецов, С. П. Євсеєв, О. Г. Король. – Харків : ХНЕУ, 2011. – 512 с.

6. Грайворонський, М. В. Безпека інформаційно-комунікаційних систем / М. В. Грайворонський, О. М. Новіков. – Київ : Видавнича група BHV, 2009. – 608 с.

7. Кавун, С. В. Основи інформаційної безпеки / С. В. Кавун, О. А. Смірнов, В. Ф. Столбов. – Кіровоград : КНТУ, 2012. – 414 с.

8. Домарев, В. В. Безопасность информационных технологий / В. В. Домарев. – М., С.-Пб., Киев : DiaSoft, 2002. – 671с.

9. Остапов, С. Е. Технології захисту інформації : навч. посіб. / С. Е. Остапов, С. П. Євсеєв, О. Г. Король. – Харків : ХНЕУ, 2013. – 476 с.

10. Кузнецов, О. О. Захист інформації в інформаційних системах. Методи традиційної криптографії : навч. посіб. / О. О. Кузнецов, С. П. Євсеєв, О. Г. Король. – Харків : ХНЕУ, 2010. – 316 с.

11. Ємець, В. Сучасна криптографія. Основні поняття / В. Ємець, А. Мельник, Р. Попович. – Львів : Бак, 2003. – 144 с.

12. Баричев, С. Г. Основы современной криптографии / С. Г. Баричев, Р. Е. Серов. – М. : Горячая линия – Телеком, 2001. – 152 с.

ДОДАТОК А

Закон України
«Про захист інформації в інформаційно-телекомунікаційних системах»

Цей Закон регулює відносини у сфері захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах (далі – системи).

Стаття 1. Визначення термінів

У Законі наведені нижче терміни застосовують у такому значенні:

блокування інформації в системі – дії, внаслідок яких унеможливлюється доступ до інформації в системі;

виток інформації – результат дій, внаслідок яких інформація в системі стає відомою або доступною фізичним та/або юридичним особам, що не мають права доступу до неї;

володілець інформації – фізична або юридична особа, якій належать права на інформацію {абзац четвертий ст. 1 в редакції Закону N 1170-VII (1170-18) від 27.03.2014};

власник системи – фізична або юридична особа, якій належить право власності на систему;

доступ до інформації в системі – отримання користувачем можливості обробляти інформацію в системі;

захист інформації в системі – діяльність, спрямована на запобігання несанкціонованим діям щодо інформації в системі;

знищення інформації в системі – дії, внаслідок яких інформація в системі зникає;

інформаційна (автоматизована) система – організаційно-технічна система, в якій реалізується технологія оброблення інформації з використанням технічних і програмних засобів;

інформаційно-телекомунікаційна система – сукупність інформаційних і телекомунікаційних систем, які у процесі оброблення інформації діють як єдине ціле;

комплексна система захисту інформації – взаємозв'язана сукупність організаційних та інженерно-технічних заходів, засобів і методів захисту інформації;

користувач інформації в системі (далі – користувач) – фізична або юридична особа, яка в установленому законодавством порядку отримала право доступу до інформації в системі;

криптографічний захист інформації – вид захисту інформації, що реалізується шляхом перетворення інформації з використанням спеціальних (ключових) даних з метою приховування/відновлення змісту інформації, підтвердження її справжності, цілісності, авторства тощо;

несанкціоновані дії щодо інформації в системі – дії, що провадяться з порушенням порядку доступу до цієї інформації, установленого відповідно до законодавства;

оброблення інформації в системі – виконання однієї або кількох операцій, зокрема, збирання, введення, записування, перетворення, зчитування, зберігання, знищення, реєстрація, приймання, отримання, передавання, які здійснюються в системі за допомогою технічних і програмних засобів;

порушення цілісності інформації в системі – несанкціоновані дії щодо інформації в системі, внаслідок яких змінюється її вміст;

порядок доступу до інформації в системі – умови отримання користувачем можливості обробляти інформацію в системі та правила оброблення цієї інформації;

телекомунікаційна система – сукупність технічних і програмних засобів, призначених для обміну інформацією шляхом передавання, випромінювання або приймання її у вигляді сигналів, знаків, звуків, рухомих або нерухомих зображень або іншим способом;

технічний захист інформації – вид захисту інформації, спрямований на забезпечення за допомогою інженерно-технічних заходів та/або програмних і технічних засобів унеможливлення витоку, знищення та блокування інформації, порушення цілісності та режиму доступу до інформації.

Стаття 2. Об'єкти захисту в системі

Об'єктами захисту в системі є інформація, що обробляється в ній, та програмне забезпечення, яке призначено для оброблення цієї інформації.

Стаття 3. Суб'єкти відносин

Суб'єктами відносин, пов'язаних із захистом інформації в системах, є: володільці інформації, власники системи, користувачі, спеціально уповноважений центральний орган виконавчої влади з питань організації спеціального зв'язку та захисту інформації і підпорядковані йому регіональні органи {абзац п'ятий ч. 1 ст. 3 в редакції Закону N 879-VI (879-17) від 15.01.2009} {абзац шостий ч. 1 ст. 3 виключено на підставі Закону N 767-VII (767-18) від 23.02.2014} {ч. 2 ст. 3 виключено на підставі Закону N 1170-VII (1170-18) від 27.03.2014}.

На підставі укладеного договору або за дорученням власник системи може надати право розпоряджатися системою іншій фізичній або юридичній особі – розпоряднику системи.

Стаття 4. Доступ до інформації в системі

Порядок доступу до інформації, перелік користувачів та їх повноваження стосовно цієї інформації визначаються володільцем інформації.

Порядок доступу до державних інформаційних ресурсів або інформації з обмеженим доступом, вимога щодо захисту якої встановлена

законом, перелік користувачів та їх повноваження стосовно цієї інформації визначаються законодавством.

У випадках, передбачених законом, доступ до інформації в системі може здійснюватися без дозволу її володільця в порядку, встановленому законом {ч. 3 ст. 4 із змінами, внесеними згідно із Законом N 1170-VII (1170-18) від 27.03.2014}.

Стаття 5. Відносини між володільцем інформації та власником системи

Власник системи забезпечує захист інформації в системі в порядку та на умовах, визначених у договорі, який укладається ним із володільцем інформації, якщо інше не передбачено законом.

Власник системи на вимогу володільця інформації надає відомості щодо захисту інформації в системі.

Стаття 6. Відносини між власником системи та користувачем

Власник системи надає користувачеві відомості про правила і режим роботи системи та забезпечує йому доступ до інформації в системі відповідно до визначеного порядку доступу.

Стаття 7. Відносини між власниками систем

Власник системи, яка використовується для оброблення інформації з іншої системи, забезпечує захист такої інформації в порядку та на умовах, що визначаються договором, який укладається між власниками систем, якщо інше не встановлено законодавством.

Власник системи, яка використовується для оброблення інформації з іншої системи, повідомляє власника зазначеної системи про виявлені факти несанкціонованих дій щодо інформації в системі.

Стаття 8. Умови оброблення інформації в системі

Умови оброблення інформації в системі визначаються власником системи відповідно до договору з володільцем інформації, якщо інше не передбачено законодавством.

Державні інформаційні ресурси або інформація з обмеженим доступом, вимога до захисту якої встановлена законом, мають оброблятися в системі із застосуванням комплексної системи захисту інформації з підтвердженою відповідністю. Підтвердження відповідності здійснюється за результатами державної експертизи в порядку, встановленому законодавством {ч. 2 ст. 8 із змінами, внесеними згідно із Законом N 1170-VII (1170-18) від 27.03.2014}.

Для створення комплексної системи захисту державних інформаційних ресурсів або інформації з обмеженим доступом, вимога до захисту якої встановлена законом, використовуються засоби захисту інформації, які мають сертифікат відповідності або позитивний експертний висновок за результатами державної експертизи у сфері технічного та/або криптографічного захисту інформації. Підтвердження відповідності та проведення державної експертизи цих засобів здійснюються в порядку, встановленому законодавством.

Стаття 9. Забезпечення захисту інформації в системі

Відповідальність за забезпечення захисту інформації в системі покладається на власника системи.

Власник системи, в якій обробляються державні інформаційні ресурси або інформація з обмеженим доступом, вимога до захисту якої встановлена законом, створює службу захисту інформації або призначає осіб, на яких покладається забезпечення захисту інформації та контролю за ним {ч. 2 ст. 9 із змінами, внесеними згідно із Законом N 1170-VII (1170-18) від 27.03.2014}.

Про спроби та/або факти несанкціонованих дій у системі щодо державних інформаційних ресурсів або інформації з обмеженим доступом, вимога до захисту якої встановлена законом, власник системи повідомляє відповідно спеціально уповноважений центральний орган виконавчої влади з питань організації спеціального зв'язку та захисту інформації або підпорядкований йому регіональний орган {ч. 3 ст. 9 із змінами, внесеними згідно із Законом N 879-VI (879-17) від 15.01.2009}.

Стаття 10. Повноваження державних органів у сфері захисту інформації в системах

Вимоги (374-2006-п) до забезпечення захисту державних інформаційних ресурсів або інформації з обмеженим доступом, вимога до захисту якої встановлена законом, встановлюються Кабінетом Міністрів України {ч. 2 ст. 10 виключено на підставі Закону N 879-VI (879-17) від 15.01.2009}.

Спеціально уповноважений центральний орган виконавчої влади з питань організації спеціального зв'язку та захисту інформації {абзац перший ч. 3 ст. 10 в редакції Закону N 879-VI (879-17) від 15.01.2009}:

- розробляє пропозиції щодо державної політики у сфері захисту інформації та забезпечує її реалізацію в межах своєї компетенції;

- визначає вимоги та порядок створення комплексної системи захисту державних інформаційних ресурсів або інформації з обмеженим доступом, вимога до захисту якої встановлена законом;

- організовує проведення державної експертизи комплексних систем захисту інформації, експертизи та підтвердження відповідності засобів технічного і криптографічного захисту інформації;

- здійснює контроль за забезпеченням захисту державних інформаційних ресурсів або інформації з обмеженим доступом, вимога до захисту якої встановлена законом;

- здійснює заходи щодо виявлення загрози державним інформаційним ресурсам від несанкціонованих дій в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах та надає рекомендації з питань запобігання такій загрозі {ч. 3 ст. 10 доповнено абзацем згідно із Законом N 1180-VI (1180-17) від 19.03.2009}.

Державні органи в межах своїх повноважень за погодженням із спеціально уповноваженим центральним органом виконавчої влади з

питань організації спеціального зв'язку та захисту інформації або підпорядкованим йому регіональним органом встановлюють особливості захисту державних інформаційних ресурсів або інформації з обмеженим доступом, вимога до захисту якої встановлена законом {ч. 4 ст. 10 із змінами, внесеними згідно із Законом N879-VI (879-17) від 15.01.2009}.

Особливості захисту інформації в системах, які забезпечують банківську діяльність, встановлюються Національним банком України.

Стаття 11. Відповідальність за порушення законодавства про захист інформації в системах

Особи, винні в порушенні законодавства про захист інформації в системах, несуть відповідальність згідно із законом.

Стаття 12. Міжнародні договори

Якщо міжнародним договором, згода на обов'язковість якого надана Верховною Радою України, визначено інші правила, ніж ті, що передбачені цим Законом, застосовуються норми міжнародного договору.

Стаття 13. Прикінцеві положення

1. Цей Закон набирає чинності з 1 січня 2006 року.

2. Нормативно-правові акти до приведення їх у відповідність із цим Законом діють у частині, що не суперечить цьому Закону.

3. Кабінету Міністрів України та Національному банку України в межах своїх повноважень протягом шести місяців з дня набирання чинності цим Законом:

- привести свої нормативно-правові акти у відповідність із цим Законом;

- забезпечити приведення міністерствами та іншими центральними органами виконавчої влади їх нормативно-правових актів у відповідність із цим Законом.

Навчальне видання

Пащенко Руслан Едуардович

ЗАХИСТ ПРОСТОРОВО-РОЗПОДІЛЕНИХ ДАНИХ У КОМП'ЮТЕРНИХ СИСТЕМАХ

Редактор В. М. Коваль

Зв. план, 2020
Підписано до друку 19.08.2020
Формат 60x84 1/16. Папір офс. № 2. Офс. друк
Ум. друк. арк. 5,8. Обл.-вид. арк. 6,5. Наклад 50 пр.
Замовлення 215. Ціна вільна

Видавець і виготовлювач
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»
61070, Харків-70, вул. Чкалова, 17
http:// www.khai.edu
Видавничий центр «ХАІ»
61070, Харків-70, вул. Чкалова, 17
izdat@khai.edu

Свідоцтво про внесення суб'єкта видавничої справи
до Державного реєстру видавців, виготовлювачів і розповсюджувачів
видавничої продукції сер. ДК № 391 від 30.03.2001