МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРКАСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
імені БОГДАНА ХМЕЛЬНИЦЬКОГО

Факультет обчислювальної техніки,
інтелектуальних та управляючих систем

Кафедра програмного забезпечення
автоматизованих систем

О.О. Супруненко, О.А. Блакова, І.М. Сиволовський

МЕТОДИЧНІ ВКАЗВІКИ ТА ІНДИВІДУАЛЬНІ ЗАВДАННЯ

ДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ

з дисципліни "Безпека програм та даних"

для студентів напрямів підготовки

050103 – Програмна інженерія,
050101 – Комп'ютерні науки,
040303 – Системний аналіз

усіх форм навчання

Черкаси 2013

УДК 004.056, 004.453, 004.49
ББК 22.19

Рецензенти: Кузьмук В.В., доктор технічних наук, професор, заступник керівника Відділення гібридних моделюючих і управляючих систем в енергетиці Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України;

Онищенко Б.О., кандидат фізико-математичних наук, доцент кафедри програмного забезпечення автоматизованих систем Черкаського національного університету імені Богдана Хмельницького.

Супруненко О.О., Блакова О.А., Сиволовський І.М.
Методичні вказівки та індивідуальні завдання до виконання лабораторних робіт з дисципліни „Безпека програм та даних” для студентів напрямів підготовки 050103 – Програмна інженерія, 050103 – Комп’ютерні науки, 040303 – Системний аналіз усіх форм навчання. – Черкаси, ЧНУ, 2013. – 66 с.

Дані методичні вказівки містять необхідний теоретичний матеріал та методичні рекомендації до виконання лабораторних робіт з дисципліни «Безпека програм та даних». Кожна лабораторна робота має теоретичні відомості та варіанти завдань до виконання роботи. Також до кожної лабораторної роботи наведені література та Interiet-джерела, які допомагають розширити знання з опрацьовуваної теми, містять приклади реалізації захисних функцій. Наведені методичні вказівки щодо оформлення звіту з лабораторних робіт. Видання призначене для студентів випускного курсу бакалаврату, які навчаються за напрямами підготовки 050103 – Програмна інженерія, 050101 – Комп’ютерні науки та 040303 – Системний аналіз.

УДК 004.056, 004.453, 004.49
ББК 22.19

Рекомендовано до друку Вченою радою
Черкаського національного університету ім. Б. Хмельницького
(протокол № 1 від 28 серпня 2012 р.)

Зміст

1. Методичні вказівки до виконання та оформлення лабораторних робіт

1.1. Методичні вказівки до виконання та оформлення лабораторних робіт для студентів стаціонару

Лабораторні роботи (ЛР) студент виконує за варіантом, який отримує на початку лабораторного заняття. Кожне завдання передбачає: 1) опрацювання теоретичного матеріалу – крипторгафічного методу чи схемо-технічного рішення реалізації захисного модуля; 2) розробку алгоритму реалізації поставленого завдання; 3) реалізацію завдання на мові програмування, що відповідає мові реалізації програмного продукту, який захищається.

Після виконання завдання з кожної лабораторної роботи студент оформлює звіт на листах формату А4, що формується у вигляді текстового документу в текстовому редакторі Word. Формат: всі поля по 2 см, кегль шрифта 14, міжрядковий інтервал 1,5.

Титульний лист єдиний на всі лабораторні роботи і формується за зразком, наведеним у додатку А.

Звіт починається з вказівки номера лабораторної роботи, її теми та завдання, яке студент отримав для виконання (додаток Б). Прізвище автора, ініціали та група вказуються у верхньому колонтитулі справа.

Зміст звіту складається з п’яти частин:

1) опис методу реалізації завдання,
2) алгоритм реалізації завдання,
3) лістинг модуля захисту з не обхідними коментарями,
4) результати виконання (чи тестування) додатку,
5) висновки.

Нумерація сторінок проставляється в межах кожної лабораторної роботи і починається з 1-ї сторінки.

У разі використання додаткових інформаційних джерел (інформації з сайтів чи друкованих видань) наприкінці звіту з лабораторної роботи (після висновку) наводиться ***список використаних джерел***, а в тексті відповідного пункту робиться посилання на джерело. Наприклад: [2, с. 37-42], що означає посилання на 2-е джерело зі списку використаних джерел, в якому використана інформація розміщується на сторінках 37-42. У списку використаних джерел друковані видання та Internet-ресурси оформлюються за ДСТУ

7.1:2006 (додаток В) та нумеруються у порядку їх використання в тексті роботи.

При виконанні завдання студент користується конспектом лекцій, методичними вказівками до лабораторних робіт, рекомендованою літературою та Internet-джерелами (с. 53). Виконані та захищені лабораторні роботи підшиваються у журнал звітів з відповідним титульним листом, і є допуском студента до іспиту за 7-й семестр з дисципліни „Безпека програм та даних ”.

1.2. Методичні вказівки до виконання та оформлення лабораторних робіт для студентів заочної та екстернатної форм навчання

Студенти заочної та екстернатної форми навчання виконують лабораторні роботи (ЛР) за варіантом, отриманим на установчій сесії. Кожне завдання передбачає: 1) опрацювання теоретичного матеріалу – крипторгафічного методу чи схемо-технічного рішення реалізації захисного модуля; 2) розробку алгоритму реалізації поставленого завдання; 3) реалізацію завдання на мові програмування, що відповідає мові реалізації програмного продукту, який захищається.

Після виконання завдання з кожної лабораторної роботи студент оформлює звіт на листах формату А4, що формується у вигляді текстового документу в текстовому редакторі Word, що дозволяє студентам-заочникам переслати роботу для попередньої перевірки викладачу по *e-mail*. Формат: всі поля по 2 см, кегль шрифта 14, міжрядковий інтервал 1,5.

Титульний лист виконується на всі лабораторні роботи за зразком, наведеним у додатку А, і підписується останнім після захисту всіх лабораторних робіт.

Звіт починається з вказівки номера лабораторної роботи, її теми та завдання, яке студент отримав для виконання (додаток Б). Прізвище автора, ініціали та група вказуються у верхньому колонтитулі справа.

Зміст звіту складається з п’яти частин:

1) опис методу реалізації завдання;
2) алгоритм реалізації завдання;
3) лістинг модуля захисту з не обхідними коментарями;
4) результати виконання (чи тестування) додатку;
5) висновки.

Нумерація сторінок проставляється в межах кожної лабораторної роботи і починається з 1-ї сторінки.

У разі використання додаткових інформаційних джерел (інформації з сайтів чи друкованих видань) наприкінці звіту з лабораторної роботи (після висновку) наводиться ***список використаних джерел***, а в тексті відповідного пункту робиться посилання на джерело. Наприклад: [2, с. 37-42], що означає посилання на 2-е джерело зі списку використаних джерел, в якому використана інформація розміщується на сторінках 37-42. У списку використаних джерел друковані видання та Internet-ресурси оформлюються за ДСТУ 7.1:2006 (додаток В) та нумеруються у порядку їх використання в тексті роботи.

При виконанні завдання студент користується конспектом лекцій, методичними вказівками до лабораторних робіт, рекомендованою літературою та Internet-джерелами (с. 53).

Виконані та захищені на екзаменаційній сесії лабораторні роботи підшиваються у журнал звітів з відповідним титульним листом, і є допуском студента до іспиту за 7-й семестр з дисципліни „Безпека програм та даних ”.

2. Навчальні матеріали до виконання лабораторних робіт

При виконанні кожної лабораторної роботи перед студентом ставиться завдання, над яким він має працювати. Студент ознайомлюється з теоретичним матеріалом, отримує від викладача варіант, опрацьовує відповідний матеріал, формує алгоритми для виконання завдання у вигляді граф-схем, реалізує програмний модуль та проводить його тестування у стандартних умовах та при порушенні стандартного режиму роботи.

За підсумками виконання лабораторної роботи студент формує звіт, роздруковує його та захищає роботу перед викладачем на лабораторному занятті. Для захисту роботи необхідно продемонструвати роботу розробленого програмного продукту та відповісти на запитання по звіту з лабораторної роботи.

Лабораторна робота № 1.

Правові аспекти безпеки інформаційної діяльності в Україні.

Мета роботи: ознайомити студента із законодавчим та нормативним забезпеченням безпеки інформаційної діяльності у організаційних структурах різної спрямованості. Набути знань із правового захисту інформації у організаціях, що займаються розробкою та технічним супроводом програмних продуктів, адмініструванням комп'ютерних мереж, та іншою діяльністю, пов'язаною з виготовленням та обслуговуванням комп'ютерної техніки та програмного забезпечення.

Завдання: Зібрати необхідну законодавчу та нормативну документацію для законного ведення робіт у певній сфері IT (за варіантом).

Теоретичні відомості

Кожен суб'єкт господарчої діяльності в Україні, що діє в рамках правового поля, стикається з питаннями інформаційної безпеки та безпеки програмного забезпечення [1], яке він використовує. Тому важливо знати сучасну законодавчу базу з цих питань, та прояснити, які права суб'єкта господарчої діяльності захищені, і які він має

обов’язки перед державою та іншими суб’єктами, результати праці яких він використовує.

Статті 17, 32, 34 Конституції України визначають забезпечення інформаційної безпеки як одну з найважливіших функцій держави і мають стати основою розвитку інформаційного законодавства. «… захист інформації, охорона державної таємниці є невід’ємними складовими національної безпеки України. В інформаційному просторі ця інформація займає дуже незначну частку і стосується чітко окреслених сфер державної діяльності – оборони, економіки, зовнішніх відносин, державної безпеки і охорони правопорядку, що прямо закріплено у **статті 6 Закону "Про державну таємницю**"» [2].

Протокольне рішення Кабінету Міністрів України "**Про захист таємної та службової інформації**" (13.04.92), затверджені постановами Кабінету Міністрів України "**Тимчасовий перелік відомостей, опублікування яких обмежується**", "**Положення про порядок підготовки матеріалів, призначених для відкритого опублікування**" (21.07.92), "**Положення про Державну службу України з питань технічного захисту інформації**" та "**Положення про Державний комітет України з питань державних секретів**" й інші є основою правового захисту інформації в Україні.

На сьогодні правову основу забезпечення технічного захисту інформації (ТЗІ) становлять **Концепція національної безпеки України**, Закони України "**Про інформацію**", "**Про державну таємницю**", "**Про захист інформації в автоматизованих системах**", інші нормативно-правові акти, а також міжнародні договори України, що стосуються сфери інформаційних відносин.

Невизначеними є положення чинного інформаційного законодавства щодо таємної інформації, яка не становить державної таємниці, її захисту, особливостей відкритої інформації як об’єкту захисту, персональних даних, комерційної таємниці та іншої конфіденційної інформації.

Сьогодні в Україні діє близько **15 законів та 60 нормативних актів** (додаток Г), що стосуються регулювання відносин в інформаційній сфері. Крім того, видано низку відомчих актів Держкомсекретів України – циркулярних листів, роз’яснень та методик (понад 15), які є обов’язковими для усіх державних органів, підприємств, установ, організацій під час здійснення ними функцій щодо забезпечення охорони інформації з обмеженим доступом, перш за все – державної таємниці.

Зростання загроз для інформації, спричинене лібералізацією суспільних та міждержавних відносин, застосуванням технічних засобів обробки інформації та засобів зв'язку іноземного виробництва, поширенням засобів несанкціонованого доступу до інформації та впливу на неї, визначає необхідність проведення робіт із розробки і впровадження **нормативних документів системи технічного захисту інформації** (додаток Г).

Завдання до лабораторної роботи

Зібрати необхідну законодавчу та нормативну документацію для законного ведення робіт у наступних організаціях та їх підрозділах:

Варіант	Організація, підрозділ	Посада
1	Компанія з розробки ліцензованих програмних продуктів (з іноземним капіталом)	Директор
2	Internet-провайдер	Голова регіонального відділення
3	Підприємство з виробництва комп'ютерного обладнання (материнських плат), відділ інформаційної безпеки	Голова відділу інформаційної безпеки
4	Невелика організація з розробки комп'ютерних ігор (до 10 осіб)	Директор
5	Фірма з постачання комп'ютеризованого медичного обладнання	Директор
6	Офіційний представник компанії-розробника програмного забезпечення (наприклад, «Бухгалтерія 1С»), що займається постачанням та сервісним обслуговуванням ПЗ	Директор регіональної філії
7	Медичний заклад (лікарня), відділ комп'ютеризації	Голова відділу комп'ютеризації
8	Компанія – оператор телекомунікацій (надає послуги файлообміну)	Фахівець із безпеки інформації
9	Обласний центр обробки даних (ЦОД), відділ безпеки даних	Голова відділу безпеки даних
10	Обчислювальний центр (ОЦ) виробничого об'єднання (5 філій)	Голова ОЦ
11	---	Вільнонайманий програміст. / Як забезпечити довіру замовника до розроблюваного ПЗ?

У **висновку** описати, яким чином зібрані законодавчі та нормативні акти дозволять уникнути організаційних труднощів, фінансових та інших законодавчих санкцій у господарчій діяльності (виробництві, формуванні послуг, реалізації продукції та послуг).

Література

1. Гундарь К.Ю., Гундарь А.Ю., Янишевский Д.А. Защита информации в компьютерных системах. – Киев: Корнійчук, 2000.
2. Конституція України // Закони України. – Т.10. – К., 1997.
3. Закон України "Про державну таємницю" від 21.01.1994 р. // Закони Ук-раїни. – Т.7. – К., 1997.
4. Закон України "Про інформацію" від 2.10.1992 р. // Закони України. – Т.4. – К., 1996.

5. Деякі питання правового захисту інформації в Україні [Електронний документ]. Режим доступу: http://www.bez-peka.com/ru/lib/spec/law/art6.html. Перевірено: 27.07.12.

Лабораторна робота № 2

Профіль особистої інформаційної безпеки.

Мета роботи: виховання відповідального відношення до інформаційної діяльності, що пов’язана з обробкою і збереженням інформації, створення профілю особистої інформаційної безпеки.

Завдання: Освоїти основні способи профілактичної й попереджувальної діяльності по відношенню до інформаційних загроз на рівні особистої інформаційної безпеки.

Теоретичні відомості

1. Фундаментальні властивості цифрової інформації.
2. Причини появи кіберзлочинності.
3. Приклади невирішених проблем.
4. Причини, з яких ідеальний захист інформації неможливий.

В часи Римської імперії була сформульована геополітична формула: "Хто володіє морем, той володіє світом" Під час другої світової війни цей вираз був модифікований таким чином: "Хто володіє повітряним простором, той володіє світом". І у другій

половині XX століття, в період становлення постіндустріального суспільства був вироблений новий геополітичний тезис: "Хто володіє інформацією, той володіє світом!", який залишається актуальним і зараз.

Настав час, коли необхідно рахуватися з тим, що перехід інформації в розряд найважливіших ресурсів людства одночасно породжує проблему володіння цим ресурсом, його знищенням чи зміненням, виходячи з державних, комерційних, приватних та інших інтересів, і, як наслідок, приводить до появи нового засобу нападу чи захисту, тобто інформаційної зброї. Причиною такої зміни стала можливість представлення інформації у цифровому вигляді. Важливо відмітити, що цифрова інформація має наступні невід'ємні властивості:

- Отчужуваність,
- Відтворюваність,
- Незнищенність,
- Можливість швидкого пошуку.

З одного боку, ці якості дозволяють істотно оптимізувати процес обробки інформації, зводячи до мінімуму втручання людини в рутинні процеси і забезпечуючи легкий та швидкий доступ до необхідних відомостей. З іншого боку, вони ж стали причиною появи наприкінці XX ст. нового виду злочинності – кіберзлочинності. Так, відчужуваність та відтворюваність інформації разом призвели до загострення проблеми захисту авторських прав. Не так давно поняття "крадіжка" означало, що суб'єкт втрачає деякі матеріальні цінності. З цифровою інформацією все по-іншому: якщо пірат копіює диск із записом ще не випущеного фільму, власність правонаступника може фізично не постраждати – однак у такий спосіб законний хазяїн втрачає над своїм твором контроль. Більш того, якщо до появи комп'ютерів створення дублікатів призводило до погіршення якості об'єкту копіювання (репродукції картин, переписування книг та аудіокасет і т.і.), то в цифровому світі копіювання може відбуватися в необмежених кількостях практично безкоштовно – і без втрат якості! До парадоксально небажаних результатів призвело і швидке зниження цін на пристрої зберігання даних. Робота по аналізу збереженої в організації інформації з метою виявлення даних, що належить знищити по причині втрати актуальності й корисності, обходиться дорожче купівлі й встановлення нового обладнання. Внаслідок цього стрічки новин переповнені заголовками про знайдених в смітті чи куплених на аукціонах жорстких дисках та магнітних стрічках, що

вміщують секретні відомості, які ніхто не видалив перед утилізацією пристроїв. Окремої уваги заслуговує проблема видалення інформації, яка хоча б одного разу з'явилася в Інтернеті й проіндексована пошуковою системою. Можливість швидкого пошуку і об'єднання по ключовим полям (наприклад, ПІБ і адреса електронної пошти) робить задачу створення портрету активного користувача комп'ютера максимально простою для зловмисника.

Як наслідок цього – гострота проблем забезпечення інформаційної безпеки (ІБ) суб'єктів інформаційних відносин, захист їх законних інтересів при використанні інформаційних систем і мереж, збереженої, оброблюваної та такої, що передається в них інформації постійно зростає. Не дивлячись на інтенсивне впровадження знов створюваних технологічних рішень в області інформаційної безпеки, рівень криміногенності в інформаційній сфері мереж передачі даних провідних країн світу постійно підвищується, що призводить до мільярдних фінансових збитків.

За даними координаційного центру миттєвого реагування CERT, організованого при університеті Карнегі Меллона, щорічно спостерігається ріст кількості зареєстрованих інформаційних атак (рис. 1.2). Однією з найбільш актуальних проблем останнім часом став захист від інсайдерів (співробітників компанії, що є порушниками, які можуть мати легальний доступ до конфіденційної інформації). Така загроза стала можливою переважно з появою портативних та дешевих пристроїв зберігання інформації з високою щільністю запису. Загострює ситуацію світова фінансова криза, в умовах якої зросла кількість співробітників, незадоволених своїм роботодавцем (внаслідок скорочення зарплати чи навіть звільнення) й таких, що бажають нанести йому шкоду чи незаконно збагатитися.

Характерно, що кількість комп'ютерних злочинів щорічно збільшується (рис. 1). Так, згідно статистики Міністерства внутрішніх справ РФ, кількість комп'ютерних злочинів, які пов'язані з несанкціонованим доступом до конфіденційної інформації, збільшилася з шестисот інцидентів у 2000 році до семи тисяч у 2004 р. До основних причин зростання кількості атак можна віднести наступні фактори:

- ✓ з кожним роком збільшується кількість користувачів загальнодоступних мереж зв'язку, таких, наприклад, як мережа Інтернет. При цьому в якості нових користувачів виступають як окремі клієнтські робочі станції, так і цілі корпоративні мережі;
- ✓ збільшується кількість вразливостей, які щоденно виявляються у існуючому загальносистемному та прикладному програмному забезпеченні;

Рис. 1. Зростання кількості атак у найближчі роки [1].

- ✓ зростає кількість можливих об’єктів атаки. Якщо кілька років тому в якості основних об’єктів несанкціонованого впливу розглядалися виключно сервери стандартних Web-служб, такі як HTTP, SMTP та FTP, то до поточного моменту розроблені засоби реалізації атак на маршрутизатори, комутатори, міжмережеві екрани та ін.;
- ✓ спрощуються методи реалізації інформаційних атак. В мережі Інтернет можна з легкістю знайти програмні реалізації атак, спрямованих на активізацію різних вразливих місць ПЗ. При цьому використання цих засобів зводиться до вводу IP-адреси об’єкту атаки й натисненню відповідної керуючої кнопки;
- ✓ збільшується кількість внутрішніх атак зі сторони користувачів автоматизованих систем (АС). Прикладами таких атак є крадіжка конфіденційної інформації чи запуск шкідливого програмного забезпечення (ПЗ) на робочих станціях користувачів.

Необхідно відмітити, що рівень складності інформаційних атак також постійно зростає. Дане твердження можна проілюструвати на прикладі еволюції комп’ютерних вірусів. В момент свого першого прояву у 1980 р. віруси були достатньо простими програмами, які самостійно розповсюджувалися в автоматизованих системах і основним завданням яких було порушення працездатності системи. Сьогодні ж комп’ютерні віруси є суттєво більш складними

програмними засобами, що здатні розповсюджуватися практично в будь-якому середовищі передачі інформації, а також маскуватися під роботу штатного ПЗ. Крім того, сучасні модифікації комп'ютерних вірусів в основному використовуються для крадіжки конфіденційної інформації, а також для отримання несанкціонованого доступу до комп'ютерів користувачів. Аналогічна тенденція характерна і для інших видів загроз безпеки, для реалізації яких постійно продукуються більш складні методи та засоби проведення атак. Змінилася й ментальність хакерів: якщо раніше основною мотивацією було розв'язання складної проблеми і можливість самоствердження, то сьогодні на перший план виходить комерційна складова, яка сприяє об'єднанню талановитих одинаків в організовані злочинні співтовариства.

Варто звернути увагу на позитивну тенденцію – деякі виробники програмного і апаратного забезпечення стали звертати увагу на безпеку продукту вже на стадії проектування, а не в останній момент, коли змінити щось в архітектурі системи вже пізно і можливо задовольнятися функціональними "латками". Однак і на цьому шляху є перепони: по-перше, виробництво продукту, що не має помилок, в реальному світі неможливе; по-друге, комп'ютер є системою з величезною кількістю компонент від різних вендорів, і тестування сукупної роботи можливих комбінацій є невирішеною задачею (задачею, що неможливо вирішити). І самий досконалий захист може бути зламаний, і причина цьому – людський фактор. Усунути цю загрозу принципово неможливо, оскільки персонал є невід'ємною частиною будь-якої інформаційної системи.

З врахуванням вищесказаного можна з упевненістю сказати, що проблема захисту АС від інформаційних атак є однією з найбільш актуальних і значимих в IT-індустрії. У світі щорічно проводиться велика кількість досліджень, спрямованих на розробку нових і більш ефективних методів протидії загрозам зловмисників.

Найбільш актуальними на даний момент є загрози безпеці персональних даних. ***Загрози безпеці персональних даних*** – це сукупність умов та факторів, що створюють небезпеку несанкціонованого, зумисного чи випадкового, доступу до персональних даних, результатом якого може бути знищення, змінення, блокування, копіювання, розповсюдження персональних даних, а також інших несанкціонованих дій при їх обробці в інформаційній системі персональних даних [2]. Простіше кажучи, загроза – це «діра» у системі захисту інформації. Вона може привести до витоку

(знищенню, модифікації) інформації. Наявність загрози свідчить про можливість несанкціонованого маніпулювання даними.

За інформацією дослідників Київського міжнародного інституту соціології, що проведене у листопаді 2011 року, українці наштовхуються на небезпеку в мережі Internet повідомляючи про себе надлишкову інформацію:

- 60% користувачів повідомляють у мережі Internet своє дійсне ім‘я,
- 46% вказують назву населеного пункту, в якому живуть,
- 38% відсилають чи розміщують свої фотографії разом зі своїм ім‘ям,
- 33% ведуть особисту переписку з людьми, з якими познайомились в мережі Internet,
- 25% вказують свою адресу проживання,
- 8% повідомляють контактні дані своїх родичів чи друзів.

Для протидії шантажу, крадіжкам, нав’язливій рекламі необхідно:

1) в мережі Internet викладати мінімум інформації у відкритих джерелах (соціальні мережі, сайти Internet-магазинів і т.і.),
2) використовувати мінімум необхідної інформації при користуванні сервісами замовлення квитків, путівок і т. і.,
3) не вводити ідентифікаційну інформацію (наприклад, номер банківської карти) через Internet,
4) користуватися захищеними банківськими сервісами (використання мережі тільки для відслідковування інформації, використовувати при доступі до даних мінімум дві мережі, наприклад Internet та мобільну),
5) користуватися захищеними каналами при передачі цінної конфіденційної інформації.

Завдання до лабораторної роботи

Для виконання роботи студенти групуються по двоє для пошуку інформації один про одного. В ході виконання лабораторної роботи потрібно скласти досьє студента з використанням Інтернет-ресурсів для оцінки впливу інформаційних комп‘ютерних технологій на недоторканість його ділового та приватного життя.

Стоїть задача зібрати й систематизувати якомога більше інформації (одни про одного) з використанням загальнодоступних Інтернет-ресурсів, оцінити загрозу зловмисного застосування інформації про особу студента, й підібрати рекомендації по забезпеченню необхідного рівня безпеки приватного життя у світі

цифрових залежностей. Обмінятися результатами роботи з студентом по групі. На основі опрацьованих матеріалів сформувати профіль особистої інформаційної безпеки.

Методичні вказівки:

1. Для виконання роботи студенти групуються по двоє.
2. Перша задача: знайти якомога більше особистої інформації про колегу, використовуючи загальнодоступні мережеві ресурси:
   - пошукові системи: bing.ru, google.com.ua, yandex.ru, rambler.ru, aport.ru та ін.,
   - соціальні мережі: vkontakte.ru, odnoklassniki.ru, moikrug.ru, professionali.ru, linkedin.com, facebook.com та ін.,
   - сервіси онлайн-блогів: livejournal.com, blogs.mail.ru, blogs.yandex.ru, blog.ru, www.blogdir.ru ,
   - сайти професійних співтовариств,
   - сайти ВУЗів,
   - і т.і.
3. Створити з використанням зібраної інформації досьє з наступними пунктами:
   - ПІБ, дата народження, сімейний стан, місце проживання, контакти,
   - професія, області професійних інтересів, життєві цілі,
   - коло спілкування: родичі, друзі, колеги, знайомі,
   - відвідувані місця, пристрасті в їжі, одягу, музиці та ін.,
   - чи є машина,
   - розпорядок дня,
   - фотографії,
   - інше.
4. Оцінити можливість використання виділеної інформації зловмисниками, наприклад:
   - Телефонними терористами,
   - Зловмисниками,
   - Крадіями номерів банківських карт,
   - Розповсюджувачами рекламної продукції і т.і.
5. Передати зібрані матеріали "колезі" та отримати досьє з інформацією про себе.
6. Оцінити рівень конфіденційності, актуальності й достовірності зібраної інформації.
7. Проаналізувати висновки колеги про можливості використання знайденої інформації зловмисниками.

8. Оцінити рівень впливу цифрових технологій на своє приватне життя і продумати кроки по забезпеченню бажаного рівня безпеки.

9. Короткі висновки:

- навчитися дивитися на свої персональні дані з позиції зловмисника,
- зрозуміти важливість забезпечення особистої інформаційної безпеки у сучасному суспільстві.

Література

1. Виды угроз безопасности информации. [Електронний документ]. Режим доступу: http://www.content-security.ru/articles/vidy-ugroz-bezopasnosti-informacii/. Перевірено: 29.07.12.
2. Модель угроз. [Електронний документ]. Режим доступу: http://pdsec.ru/model_ugroz/. Перевірено: 27.07.12.
3. Деякі питання правового захисту інформації в Україні [Електронний документ]. Режим доступу: http://www.bez-peka.com/ru/lib/spec/law/art6.html. Перевірено: 27.07.12.

Лабораторна робота № 3.

Використання шифрування у програмних засобах захисту інформаційних систем.

Мета роботи: ознайомити студента з основними підходами до організації інформаційної безпеки, з основними методами шифрування інформації. Набути знань із реалізації методів шифрування інформації у вигляді програмних модулів.

Завдання: Освоїти основні методи шифрування інформації, сформувати структуру та реалізувати модуль шифрування для захисту інформаційної системи, яку власне розробив студент.

Теоретичні відомості

В технології захисту інформаційних систем створена велика кількість теоретичних моделей, що дозволяють описати практично всі аспекти безпеки і забезпечувати засоби захисту формально підтвердженою алгоритмічною базою. Практично всі існуючі теоретичні розробки основані на різних підходах до проблеми

реалізації безпеки, тому задачі реалізації безпеки та методи їх розв'язання суттєво різняться.

Найбільше розповсюдження отримали два підходи до проблеми реалізації механізмів безпеки ІС, що доповнюють один одного [1]:

- формальне моделювання політики безпеки,
- криптографія.

Формальні моделі безпеки [1] надають розробникам захисних підсистем основні принципи, що покладені в основу архітектури захисної підсистеми (Лабораторна робота №5) і визначають концепцію її побудови. ***Криптографія*** надає конкретні методи захисту інформації у вигляді алгоритмів ідентифікації, аутентифікації, шифрування та контролю цілісності.

Ідентифікацією називається процедура розпізнавання об'єкта по його ідентифікатору. ***Авторизація*** – процедура надання певних прав суб'єкту. ***Аутентифікація*** – перевірка приналежності певному суб'єкту доступа по пред'явленому ним ідентифікатору, або підтвердження достовірності (справжності чи автентичності).

Ще до появи комп'ютерів використовувалися різні відмінні риси суб'єкта, його характеристики. Зараз використання певної характеристики в системі залежить від рівня надійності, захищеності та вартості впровадження. Розрізняють 3 фактори аутентифікації [2]:

- пароль,
- пристрій аутентифікації,
- біометричні характеристики.

Пароль – це секретна інформація, якою має володіти тільки авторизований суб'єкт. Паролем може бути слово, комбінація для замка чи персональний ідентифікаційний номер (PIN). Парольний механізм може бути досить легко реалізований і мати низьку вартість. Але є суттєві мінуси: зберегти пароль в секреті часто буває проблематично, зловмисники постійно придумують нові методи крадіжки, зламу й підбору пароля. Це обумовлює слабку захищеність парольного механізму.

Пристрій аутентифікації – це унікальний пристрій, який забезпечує на апаратно-програмному рівні перевірку автентичності суб'єкта. Це може бути особиста печатка, ключ від замка, для комп'ютера це файл даних, що вміщує характеристику, яка часто вбудовується у спеціальний пристрій аутентифікації, наприклад, у пластикову картку, смарт-картку. Для зловмисника отримати такий пристрій стає більш проблематично, ніж зламати пароль, а суб'єкт може зразу ж повідомити у випадку крадіжки пристрою. Тому цей

метод більш захищений, ніж парольний механізм, однак, вартість такої системи більш висока.

Біометрика використовує у якості унікальної характеристики фізичної особливості суб'єкта. Це може бути портрет, відбиток пальця чи долоні, голос чи особливість очей. З точки зору суб'єкта, даний метод є найбільш простим: не треба ні запам'ятовувати пароль, ні переносити з собою пристрій аутентифікації. Однак, біометрична система повинна мати високу чутливість, щоб підтверджувати авторизованого користувача і мати можливість розпізнавати зловмисника зі схожими біометричними параметрами. Вартість такої системи доволі велика. Але не дивлячись на мінуси, біометричні характеристики залишаються сучасним перспективним фактором систем захисту інформації.

Криптографія (від др.-грец. κρυπτός – прихований та γράφω – пишу) – наука про методи забезпечення конфіденційності (неможливості прочитання інформації стороннім) і аутентичності (цілісності й справжності авторства, а також неможливості відмови від авторства) інформації.

Спочатку криптографія вивчала методи шифрування інформації – оберненого перетворення відкритого (початкового) тексту на основі секретного алгоритму і/чи ключа в шифрований текст (шифротекст). Традиційна криптографія утворює розділ симетричних криптосистем, у яких шифрування і розшифрування проводиться з використанням одного і того ж секретного ключа. Крім цього розділу сучасна криптографія включає в себе асиметричні криптосистеми, системи електронного цифрового підпису (ЕЦП), хеш-функції, управління ключами, отримання прихованої інформації, квантову криптографію.

Мова, як носій інформації, найбільш піддавалася впливу з зовні, в системі захисту інформації саме вона відіграє надзвичайно важливу роль. У давні часи її використовували для передачі військової та політичної інформації, що вимагала виключної конфіденційності. Для цього використовувався відомий шифр заміни Цезаря, який полягає у заміні літер основного повідомлення так, щоб в шифрованому варіанті не складалося жодного слова [5]. Кожна літера текстового повідомлення в шифротексті замінювалася на літеру, яка зміщена по алфавіту на певну кількість літер. Наприклад: повідомлення до сенату «Прийшов побачив переміг» - Veni Vidi Vici, зроблене Цезарем після одноденної війни з понтійським царем Фарнаком виглядає після шифрування зі зміщенням 4 так: yhqmymgmymfm.

Український варіант шифрованого тексту: уфкнятє утегюкє уифирлж.

Недоліком цього методу є низька стійкість, що обумовлена невеликою кількістю варіантів – на 1 менше, ніж кількість літер в алфавіті. Але в першому сторіччі до нашої ери, коли більшість населення була неграмотна використання цього методу вважалося досить надійним.

Ще одним методом шифрування є Полібіанський квадрат, який винайдений грецьким письменником та істориком Полібієм у третьому сторіччі до нашої ери. Квадрат Полібія був розміром 5×5, заповнювався абеткою у випадковому порядку. Для шифрування літеру відкритого тексту знаходили в квадраті й замінювали її літерою, що розташована в тому ж стовпчику, але на рядок нижче. Якщо літера розміщувалась в останньому рядку, то її заміняли відповідною літерою з першого рядка. При розшифровуванні такого повідомлення літеру зашифрованого тексту заміняли на літеру, що розташовувалася на рядок вище в тому ж стовпчику Полібіанського квадрату. Недоліком даного методу була складність в запам’ятовуванні розташування літер в квадраті.

Набагато пізніше, на початку шістнадцятого сторіччя Іоганн Трисеміус [5] знов винайшов шифрувальний квадрат і першим зробив систематичний опис шифрувальних таблиць, що заповнюються алфавітом у випадковій послідовності. Для отримання такого шифру найчастіше використовується ключове слово, або фразу і таблицю, яка для російської абетки має розмір 5×6. Ключове слово записується в таблиці по рядкам, відкидаючи літери, що повторюються, а потім дописуються в алфавітному порядку ті літери, які не були використані в ключовому слові. Дякуючи такому методу було легко запам’ятовувати такі шифрувальні таблиці. Наприклад, для ключового слова РЕСПУБЛІКА шифрувальна таблиця виглядає так:

Р	Е	С	П	У
Б	Л	И	К	А
В	Г	Д	Ж	З
М	Н	О	Т	Ф
Х	Ц	Ч	Ш	Щ
Ь	Ы	Э	Ю	Я

Для шифру Трисеміуса за наведеною таблицею шифрування повідомлення «Пришёл увидел победил» дає наступні результати: «Кбдюлг амдолг кчвлодг». Оскільки шифрування проводиться по одній літері, то такі шифри називаються *моногрампними*. Вони дають досить велику кількість варіантів, наприклад, якщо абетка складається з 30 літер, то кількість варіантів побудови шифрувальної таблиці буде

$30! = 2{,}65 \times 10^{32}$. Але якщо довжина шифрованого повідомлення перевищує 20-30 літер, моноалфавітний шифр досить легко можна зламати, використовуючи частоту появи символів у тексті. Так у таблиці 1 наведені ймовірності появи літер у російський абетці. При довжині повідомлення більше 100 літер криптоаналіз не викликає перешкод.

Таблиця 1

Ймовірності появи літер у російський абетці.

Літера	Ймовірності без пропуску	Ймовірності з пропуском
А	0.07922	0.063522
Б	0.01651	0.013242
В	0.04519	0.036238
Г	0.01799	0.014428
Д	0.02965	0.023775
Е	0.08363	0.067062
Ж	0.00894	0.007168
З	0.01718	0.013775
И	0.06789	0.054435
Й	0.01297	0.010401
К	0.03458	0.027731
Л	0.05028	0.040318
М	0.03147	0.025238
Н	0.06700	0.053725
О	0.10835	0.086881
П	0.02852	0.02287
Р	0.04834	0.038758
С	0.05569	0.044655
Т	0.05527	0.04432
У	0.02909	0.023329
Ф	0.00189	0.001519
Х	0.0106	0.0085
Ц	0.00330	0.002647
Ч	0.01367	0.010962
Ш	0.00971	0.00779
Щ	0.00406	0.003257
Ъ	0.00026	0.000206
Ы	0.02200	0.017638
Ь	0.01770	0.014189
Э	0.00245	0.001965
Ю	0.00569	0.004561
Я	0.02091	0.016767
Пропуск		0.198128

Тому для ускладнення аналізу повідомлення застосовуються інші способи і методи шифрування. Так для ручного шифрування російський алфавіт скорочується до 30 літер, з тексту повідомлення видаляють всі пропуски і знаки пунктуації, тексти шифрованого повідомлення розбивають на блоки однакової довжини, для того, щоб

полегшити процес розшифровки та зменшити кількість помилок шифрувальників.

Шифр «Гомоморфна підстановка» відрізняють від інших шифрів моноалфавітної заміни тим, що він містить більше символів, ніж алгоритм повідомлення. Для ускладнення аналізу частоти появи літер у тексті найбільш вживаним літерам ставлять у відповідність кілька кодів криптограми (див. таблицю 2). На 100 двозначних числових кодів та 30 літер абетки побудована така шифрувальна таблиця:

Таблиця 2

Літера	Коди криптограми
А	00 33 65 86 90 70
Б	01 34
В	02 35 66
Г	03 36
Д	04 37
Е	05 38 67 87 91 99 42
Ж	06 39
З	07 40
И	08 41 69 88 92 32
К	09 43 71
Л	10 44 72
М	11 45 73
Н	12 46 74 97
О	13 47 75 89 93 98
П	14 48
Р	15 49 68 94
С	16 50 53 77 95
Т	17 51 78 96
У	18 52 79
Ф	19
Х	20 54 80
Ц	21 55
Ч	22 56
Щ	23 57
Щ	24 58
Ь	25 59
Ы	26 60 81
Э	27 62 83 61
Ю	28 63 84
Я	29 64 85 82

Для шифрування повідомлення потрібно взяти літеру відкритого тексту і знайти її в шифрувальній таблиці, якщо вона зустрілася вперше, то замінити її першим двозначним числом, якщо вдруге – другим, і т.д. Коли перелік чисел закінчений, то при наступній заміні використовується знов перше число коду криптограми зі списку відповідних. Наприклад, зашифровуючи цим методом повідомлення «Совершенно секретно» отримаємо криптограму:

«16 13 02 05 15 23 38 12 46 47 50 67 09 49 87 17 74 75».

Шифр Гронсфельда, це модифікація шифру Цезаря числовим ключем. Для такої модифікації під символами повідомлення пишуть символи ключа (ключ складається з рядка цифр), якщо повідомлення довше за ключ, то його повторюють потрібну кількість разів. Шифрований текст отримують подібно до шифру Цезаря, але зміщення символів відносно перетворюваного не постійне, а вказується поточною цифрою ключа: якщо ключ «197», то першу літеру замінюють на літеру, що розташована праворуч (зміщення 1), другу літеру – на літеру що розташована праворуч зі зміщенням на 9 символів, третю літеру – на літеру, що розташована праворуч зі зміщенням на 7 символів і т.д. Наприклад, зашифруємо повідомлення з ключем «314»:

повідомлення	СОВЕРШЕННОСЕКРЕТНО
ключ	3143143143143143 14
шифровка	ФПЖИСЭИОССТКНСКХОТ

Щоб зашифрувати першу літеру повідомлення від «С», використовуючи першу цифру ключа «3», відраховують третю літеру за алфавітом, в результаті чого отримуємо літеру «Ф». Відповідно, шифруючи літеру «О», відраховують по зміщенню «1» наступну літеру – «П» і т.і.

Шифр Гронсфельда використовується і в наш час. Відомі різновиди і модифікації цього шифру, які покращують його стійкість, це наприклад, використання іншої абетки, або більш «сильний» шифр, що полягає у послідовному шифруванні тексту двома ключами різної довжини.

Шифри *багатоалфавітної заміни*, до яких належить шифр Гронсфельда, не зберігають у явному вигляді статистику літер вихідного повідомлення. Але існує метод криптоаналізу таких шифрів. Шифр Гронсфельда можна зламати, якщо на кожну цифру ключа припадає більше 20-30 літер тексту. Якщо на кожну цифру ключа припадає більше 100 літер, то криптоаналіз стає простою задачею.

Одним з найважливіших етапів розвитку архітектури ренесансу пов’язане ім’я Леона Батіста Альберті, який написав десять книг про будівництво, побудував палаццо Ручеллаї, церкву Іль-Джезо і багато інших прекрасних витворів мистецтва середньовічної Італії [1]. Як мистецтвознавець, він узагальнив досвід гуманістичної науки у вивченні античної спадщини, написав трактати «Про живопис», «Про будівництво», «Про статую». З іншого боку криптологи всього світу вважають його батьком своєї науки. Альберті у криптології винайшов багатоалфавітну заміну, зробивши шифрування дуже стійким до криптоаналізу. Крім самого шифру він ще навів опис пристрою з коліс, що обертаються, для його реалізації. Цей шифр можна описати за допомогою шифрувальної таблиці, що отримала назву «таблиця Віжинера» від імені Блеза Віжинера, дипломата XVI сторіччя, котрий розвивав і удосконалював криптографічні системи.

Шифрувальна таблиця Віжинера має вигляд, що показаний на рис. 2. Кожен рядок цієї таблиці відповідає одному шифру заміни на зразок шифру Цезаря.

Чим більше довжина ключа, тим стійкіший шифр. Суттєвого покращення властивостей шифротексту можливо досягти, використовуючи шифри з авто ключем.

Шифр, в якому сам відкритий текст чи отримана криптограма використовуються в якості «ключа», називається шифром з авто ключем. Шифрування у такому випадку починається з ключа, що носить назву первинного, і продовжується за допомогою відкритого ключа чи криптограми, яка зміщена на довжину первинного ключа.

Наприклад: Маємо відкритий текст: «ШИФРОВАНИЕ_ЗАМЕНОЙ». Для шифрування використаємо первинний ключ: «КЛЮЧ».

АБВГДЕЖЗИКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯ
БВГДЕЖЗИКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯА
ВГДЕЖЗИКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБ
ГДЕЖЗИКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВ
ДЕЖЗИКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГ
ЕЖЗИКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГД
ЖЗИКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕ
ЗИКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖ
ИКЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗ
КЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИ
ЛМНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИК
МНОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛ
НОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛМ
ОПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛМН
ПРСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛМНО
РСТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛМНОП
СТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛМНОПР
ТУФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛМНОПРС
УФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛМНОПРСТ
ФХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛМНОПРСТУ
ХЦЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛМНОПРСТУФ
ЦЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛМНОПРСТУФХ
ЧШЩЬЫЭЮЯАБВГДЕЖЗИКЛМНОПРСТУФХЦ
ШЩЬЫЭЮЯАБВГДЕЖЗИКЛМНОПРСТУФХЦЧ
ЩЬЫЭЮЯАБВГДЕЖЗИКЛМНОПРСТУФХЦЧШ
ЬЫЭЮЯАБВГДЕЖЗИКЛМНОПРСТУФХЦЧШЩ
ЫЭЮЯАБВГДЕЖЗИКЛМНОПРСТУФХЦЧШЩЬ
ЭЮЯАБВГДЕЖЗИКЛМНОПРСТУФХЦЧШЩЬЫ
ЮЯАБВГДЕЖЗИКЛМНОПРСТУФХЦЧШЩЬЫЭ
ЯАБВГДЕЖЗИКЛМНОПРСТУФХЦЧШЩЬЫЭЮ

Рис. 2. Шифрувальна таблиця Віжинера.

Схема шифрування з автоключем при використанні відкритого тексту виглядає так:

Ш И Ф Р О В А Н И Е _ З А М Е Н О Й
К Л Ю Ч Ш И Ф Р О В А Н И Е _ З А М
36 21 52 41 40 12 22 31 24 09 34 22 10 19 39 22 16 23
В Ф Т З Ж Л Х Ю Ч И А Х Й Т Е Х П Ц

Схема шифрування з автоключем при використанні криптограми:

Ш И Ф Р О В А Н И Е _ З А М Е Н О Й
К Л Ю Ч В Ф Т З С Ч У Х Ъ Э У Э Ы Й
36 21 52 41 18 24 20 22 27 30 53 30 24 43 26 44 39 20
В Ф Т З С Ч У Х Ъ Э У Э Ы Й Щ К Й У

Метод бітових маніпуляцій. Методи шифрування, що розглянуті вище, є комп‘ютерними версіями шифрування, яке раніше виконувалося вручну. Але комп‘ютерні технології також слугували основою для методу кодування повідомлень шляхом маніпуляцій з бітами, що складають символи нешифрованого повідомлення. Як правило, сучасні комп‘ютеризовані шифри складають клас шифрів бітових маніпуляцій (bit manipulating ciphers). Шифри бітових маніпуляцій здобули популярності з двох причин: по-перше, вони ідеально пристосовані до використання в комп‘ютерній криптографії, оскільки використовують операції, що легко виконуються системою, по-друге, отриманий на виході зашифрований текст виглядає абсолютно нечитабельним. Це позитивно відбивається на безпеці, оскільки важливі дані маскуються під пошкоджені файли, доступ до яких просто нікому не потрібен.

Шифри бітових маніпуляцій переводять відкритий текст у шифрований за допомогою перетворення набору бітів кожного символу за певним алгоритмом з використанням наступних логічних операцій (по одній чи їх комбінацій): AND, OR, NOT, XOR.

Найпростіший шифр бітових маніпуляцій використовує тільки оператор першого доповнення, який інвертує всі біти, що складають байт. Тобто всі нулі стають одиницями, а всі одиниці – нулями. Байт, над яким така операція проведена двічі стає таким же, як початковий рядок бітів.

На практиці з цією простою схемою кодування пов‘язані дві основні проблеми. По-перше, програма шифрування для розшифровки тексту не використовує ключ. Тому будь-яка людина, яка знає, що використовується даний алгоритм і може написати програму, зможе

прочитати початковий файл. По-друге, цей метод зовсім не таємниця для досвідчених програмістів.

Покращений метод шифрування з використанням побітових маніпуляцій використовує оператор XOR – результат виконання оператора XOR отримує значення ІСТИНА тоді і тільки тоді, коли один з операндів має значення ІСТИНА, а інший – ХИБНЕ. Саме це і є унікальною властивістю оператора XOR – якщо виконати цю операцію над одним байтом, використовуючи другий байт в якості «ключа», а потім виконати над результатом ту ж операцію за допомогою того ж самого ключа, то знов отримаємо початковий байт.

Наприклад:

Початковий байт		11011001
Ключ	XOR	01010011 (ключ)
Зашифрований байт		10001010
Ключ	XOR	01010011 (ключ)
Розшифрований байт		11011001

Цей процес може використовуватися для кодування файлів, оскільки він вирішує дві основні проблеми з найпростішою версією на базі першого доповнення. По-перше, завдяки використанню ключа, розшифрувати файл при наявності тільки програми декодування неможливо. По-друге, використовувані маніпуляції з бітами не настільки прості, щоб їх можна було зразу розпізнати.

Ключ для шифру може бути різної довжини, не обов‘язково 1 байт. Можливо використовувати ключ, що складається з кількох символів, і чергувати ці символи при роботі з байтами усього файлу.

Метод аналітичних перетворень. Шифрування методами аналітичних перетворень основане на понятті односторонньої функції. Функція $y = f(x)$ є односторонньою, якщо вона за порівняно невелике число операцій перетворює елемент відкритого тексту $\boldsymbol{x}$ в елемент шифртексту $\boldsymbol{y}$ для всіх значень $\boldsymbol{x}$ з області визначення, а зворотна операція (обчислення $x = F^{-1}(y)$ при відомому шифротексті) є обчислювально трудомісткою.

В якості односторонньої функції можна використовувати наступні перетворення:

1) множення матриць;
2) розв‘язання задачі про укладку ранця,
3) обчислення значення поліному по модулю;
4) експонентні перетворення та ін.

Метод множення матриць використовує перетворення виду: $Y = CX$, де $Y = \|y_1, y_2, \ldots, y_n\|^T$ – шифр текст;

$C = \|c_{ij}\|$ – матриця шифрування;

$X = \|x_1, x_2, ..., x_n\|$ – відкритий текст.

Задача про укладку ранця формулюється так: заданий вектор $C = |c_1, c_2, ..., c_n|$, що використовується для шифрування повідомлень, кожен символ s_i якого поданий послідовністю ***n*** біт $s_i = |x_1, x_2, ..., x_n|$. Шифротекст отримується як скалярний добуток $C \cdot s_i$.

Приклад: Заданий відкритий текст: «ПРИКАЗ» («16 17 09 11 01 08»)

Вектор $C = \{3,5,7,11\}$

Запишемо код кожної букви відкритого тексту в двійковому вигляді, використовуючи п‘ять розрядів:

П	Р	И	К	А	З
10000	10001	01001	01011	00001	01000

Прорахуємо значення елементів наступним чином:

$y_1 = 1 \cdot 1 + 0 \cdot 3 + 0 \cdot 5 + 0 \cdot 7 + 0 \cdot 11 = 1$ $\quad y_2 = 1 \cdot 1 + 1 \cdot 11 = 12$

$y_3 = 1 \cdot 3 + 1 \cdot 11 = 14$ $\quad y_4 = 1 \cdot 3 + 1 \cdot 7 + 1 \cdot 11 = 21$

$y_5 = 1 \cdot 11 = 11$ $\quad y_6 = 1 \cdot 3 = 3$

Шифротекст: «01 12 14 21 11 03»

Побайтні алгоритми шифрування характерні тим, що кожен наступний байт шифрується шляхом складання з попереднім байтом. При практичній реалізації можливі різні модифікації даного алгоритму, такі, як наприклад, до поточного значення шифрованого байта додається вміст не попереднього байта, а байта, що знаходиться від нього на ***k*** байт. В «чистому» вигляді даний алгоритм є дуже нестійким по тій причині, що зашифроване повідомлення вміщує в собі ключ. Нагадує ситуацію з життя: двері замкнені на ключ, а сам ключ лежить перед дверима під килимком.

Частина розглянутих алгоритмів шифрування основані на певному методі шифрування, що є секретним. ***Сучасні методи шифрування*** можуть частково використовувати такі алгоритми, але переважно ***використовують ключ*** для управління шифруванням та дешифруванням (симетричні чи асиметричні алгоритми [6]). Повідомлення може бути успішно дешифроване тільки якщо відомий ключ.

Будь які криптографічні перетворення не збільшують обсяг інформації, а лише змінюють її представлення. Тому, якщо програма шифрування значно (більш, ніж на довжину заголовку) збільшує обсяг

вихідного файлу, то в її основі лежить на оптимальний, а можливо і взагалі некоректний крипто алгоритм [6].

Зменшення обсягу закодованого файлу можливо [6] тільки при наявності вбудованого алгоритму архівації в криптосистемі та при умові стискуваності інформації (наприклад: архіви, музичні файли формату MP3, відеозображення формату JPEG стискатися більш ніж на 2-4% не будуть).

Завдання до лабораторної роботи

Ознайомитися з методом шифрування за варіантом. Розробити блок-схему алгоритму шифрування та розшифровування, реалізувати формальні моделі у вигляді двох підсистем модуля з мінімальним інтерфейсом.

Варі-ант	Основний метод	Текст для шифрування
1	*2*	*3*
1	Метод Трисеміуса, кодове слово: ймовірне	Шифрование – процесс нормального применения криптографического преобразования открытого текста на основе алгоритма и ключа, в результате которого возникает шифрованный текст.
2	Метод Цезаря, зміщення 8	Расшифровывание – процесс нормального применения криптографического преобразования шифрованного текста в открытый.
3	Гомоморфна заміна	Криптоанализ – наука, изучающая математические методы нарушения конфиденциальности и целостности информации.
4	Шифр Гронсфельда, шифр: 3827594185493761 (з періодом повтору)	Криптографическая стойкость – способность криптографического алгоритма противостоять криптоанализу.
5	Шифр Віжинера	Открытый ключ – тот из двух ключей асимметричной системы, который свободно распространяется.
6	Шифр з автоключем (1, с. 9), ключ: принт	Шифр криптосистемаы – семейство обратимых преобразований открытого текста в шифрованный.
7	Метод бітових маніпуляцій з використанням оператора XOR	Ключ – параметр шифра, определяющий выбор конкретного преобразования данного текста.

1	2	3
8	Метод аналітичних перетворень на основі задачі про укладку ранця	Открытый текст – данные, передаваемые без использования криптографии
9	Побайтний алгоритм шифрування з методом Цезаря	Шифрованный текст – данные, полученные после применения криптосистемы, обычно – с некоторым указанным ключом).

У **висновку** описати особливості реалізації завдання та варіанти застосування розробленого модуля.

Література

1. Крыжановская Ю.А. Программные методы защиты информации. Часть 1. – Врорнеж: Изд-во ВГУ, 2002. – 36 с.
2. Басов В.Е. Методичний посібник до лабораторних робіт з курсу Захист інформації. – Одеса: Вид. Нац. акад. зв'язку. – 2003. – 26 с.

3. Аутентификация [Электронный документ]. Режим доступа: http://ru.wikipedia.org/wiki/Аутентификация. Проверено: 12.03.12.
4. Венбо Мао. Современная криптография: Теория и практика: Пер. с англ. – М.: Издательский дом «Вильямс», 2005. – 768 с.
5. Квантовая криптография [Электронный документ]. Режим доступа: http://ru.wikipedia.org/wiki/Квантовая_криптография. Проверено: 12.03.12.
6. Основные алгоритмы шифрования. [Электронный документ]. Режим доступа: http://naukoved.ru/content/view/826/35/. Проверено: 12.03.12.

Лабораторна робота № 4.

Механізми парольного захисту інформаційних систем.

Мета роботи: ознайомити студента з перевагами та недоліками парольного захисту, а також механізмами реалізації парольного захисту інформаційних систем. Набути знань із реалізації механізмів парольного захисту у вигляді програмних модулів.

Завдання: Освоїти основні механізми парольного захисту інформаційних систем, сформувати структуру та реалізувати модуль парольного захисту власного програмного продукту.

Теоретичні відомості

Пароль – це секретна інформація, якою має володіти тільки авторизований суб'єкт [1]. Паролем може бути слово, комбінація для замка чи персональний ідентифікаційний номер (PIN). Парольний механізм може бути досить легко реалізований і мати низьку вартість. Але є суттєві мінуси: зберегти пароль в секреті часто буває проблематично, зловмисники постійно придумують нові методи крадіжки, зламу й підбору пароля. Це обумовлює слабку захищеність парольного механізму.

Парольний захист є одним з найрозповсюдженіших методів однофакторної аутентифікації [2-3]. Він передбачає використання певної секретної інформації – набору символів, які вводяться для ідентифікації користувача. Після ідентифікації має виконуватися аутентифікація.

Рис. 3. Класифікація систем ідентифікація та аутентифікації [3].

Під ***ідентифікацією*** [2] розуміють присвоєння суб'єктам доступу унікальних ідентифікаторів і порівняння таких ідентифікаторів з переліком можливих. Задача ідентифікації – відповісти на питання «хто це?».

Аутентифікація розглядається як перевірка приналежності суб'єкту доступу пред'явленого ним ідентифікатора і підтвердження

його справжності (автентичності). Задача аутентифікації – відповісти на питання «чи він це насправді?».

Парольні системи аутентифікації

Не дивлячись на існування різних механізмів аутентифікації (рис. 3), найбільш розповсюдженим з них залишається парольний захист. Тому є кілька причин:

- ***Відносна простота реалізації***. Реалізація механізму парольного захисту зазвичай не потребує залучення додаткових апаратних засобів.
- ***Традиційність***. Механізми парольного захисту є звичними для більшості користувачів автоматизованих систем і не викликають психологічного відторгнення – на відміну, наприклад, від сканерів малюнку сітківки ока.

Недоліки парольного захисту:

- Для парольних систем захисту характерний парадокс, що ускладнює їх ефективну реалізацію: стійкі паролі мало придатні для використання людиною.
- Стійкість пароля виникає по мірі його ускладнення; але чим складніший пароль, тим складніше його запам’ятати, і у користувача з’являється потреба записати незручний пароль, що створює додаткові канали для його дискредитації.

Рис. 4. Блок-схема алгоритму ідентифікації користувача [2]

Загрози безпеці парольного захисту. Пароль може бути отриманий зловмисником одним з трьох основних способів:

1. ***За рахунок використання слабкості людського фактору*** [2]. Методи отримання паролів у даному випадку можуть бути самими різними: підглядання, підслуховування, шантаж, погрози, використання чужих облікових записів з дозволу їх законних власників.

2. ***Шляхом підбору***. При цьому використовуються наступні методи:

- - ***Повний перебір***. Даний метод дозволяє підібрати будь-який пароль не залежно від його складності, однак для стійкого пароля час для даної атаки, має значно перевищувати допустимі часові ресурси зловмисника.
- - ***Підбір за словником***. Значна частина використовуваних на практиці паролів є осмисленими словами чи виразами. Існують словники найбільш розповсюджених паролів, які в багатьох випадках дозволяють обійтися без повного перебору.
- - ***Підбір з використанням відомостей про користувача***. Даний інтелектуальний метод підбору паролів ґрунтується на тому факті, що якщо політика безпеки системи передбачає самостійне призначення паролів користувачами, то переважно в якості пароля буде обрана певна персональна інформація, що пов’язана з користувачем АС. І хоча такою інформацією може бути обране що завгодно, від дня народження тещі й до прізвиська улюбленого песика, наявність інформації про користувача дозволяє перевірити найбільш розповсюджені варіанти (дні народження, імена дітей і т.і.).

3. ***За рахунок використання недоліків реалізації парольних систем***. До таких недоліків реалізації належать експлуатовані вразливості мережевих сервісів, що реалізують ті чи інші компоненти парольної системи захисту, чи недекларовані можливості відповідного програмного та апаратного забезпечення.

При практичній реалізації парольних систем використовують:

- ***Встановлення мінімальної довжини пароля***. Це ускладнює реалізацію повного перебору паролів.
- ***Збільшення потужності алфавіту паролів***. Воно досягається, наприклад, шляхом обов’язкового використання спецсимволів, а також механізмів ускладнення повного перебору.

- ***Перевірка та відбраковування паролів за словником***. Даний механізм дозволяє ускладнити перебір паролів за словником за рахунок відбраковування паролів, що легко підбираються.
- ***Встановлення максимального терміну дії пароля.*** Ці дії обмежують проміжок часу, який зловмисник може витратити на підбір пароля. Таким чином, зменшується ймовірність його успішного підбору.
- ***Встановлення мінімального терміну дії пароля.*** Цей механізм попереджує спроби користувача зразу змінити новий пароль на попередній.
- ***Відбраковування за журналом історії паролів.*** Механізм попереджує повторне використання паролів – можливо, раніше скомпрометованих.
- ***Обмеження числа спроб вводу пароля.*** Цей механізм ускладнює інтерактивний підбір паролів.
- ***Примусова зміна пароля при першому вході користувача в систему.*** Якщо первинну генерацію паролів для всіх користувачів здійснює адміністратор, користувачу може бути запропоновано змінити початковий пароль при першому вході в систему – в цьому випадку новий пароль не буде відомий адміністратору.
- ***Затримка при введенні неправильного пароля.*** Механізм попереджає швидкий перебір паролів.
- ***Заборона на вибір пароля користувачем і автоматична генерація пароля.*** Даний механізм дозволяє гарантувати стійкість згенерованих паролів – однак не варто забувати, що тоді у користувачів обов'язково виникнуть проблеми запам'ятовування паролів.

Оцінка стійкості парольних систем. Оцінимо елементарні взаємозв'язки між основними параметрами парольних систем [2]. Введемо наступні позначення:

- A – потужність алфавіту паролів;
- L – довжина пароля;
- $S=AL$ – потужність простору паролів;
- V – швидкість підбору паролів;
- T – термін дії пароля;
- P – ймовірність підбору пароля на протязі його терміну дії.

Очевидно, що справедливе наступне співвідношення:

$$P = \frac{V \cdot T}{A \cdot L} = \frac{V \cdot T}{S}.$$

Зазвичай швидкість підбору паролів *V та* термін дії пароля *T* можна вважати відомими. В цьому випадку, якщо задати допустиме значення ймовірності *P* підбору пароля на протязі його терміну дії, можна визначити потрібну потужність простору паролів *S*

Механізми зберігання паролів

1. ***У відкритому вигляді.*** Даний варіант не є оптимальним, оскільки автоматично створює чимало каналів витоку парольної інформації. Реальна необхідність зберігання паролів в відкритому вигляді зустрічається дуже рідко, і зазвичай таке рішення є наслідком некомпетентності розробника.
2. ***У вигляді хеш-значення.*** Даний механізм зручний для перевірки паролів, оскільки хеш-значення однозначно пов'язані з паролем, але при цьому самі не представляють інтересу для зловмисника.
3. ***У зашифрованому вигляді.*** Паролі можуть бути зашифровані з використанням деякого криптографічного алгоритму, при цьому ключ шифрування може зберігатися:
   - на одному з постійних елементів системи;
   - на деякому носії (електронний ключ, смарт-карта і т.і.), що пред'являється при ініціалізації системи;
   - ключ може генеруватися з деяких інших параметрів безпеки АС – наприклад, з пароля адміністратора при ініціалізації системи.

Передача паролів по мережі. Найбільш розповсюджені такі варіанти реалізації:

- 1. ***Передача паролів у відкритому виді***. Підхід дуже вразливий, оскільки паролі можуть бути перехвачені у каналах зв'язку. Не дивлячись на це, багато використовуваних на практиці мережевих протоколів (наприклад, FTP) передбачають передачу паролів у відкритому виді.
- 2. ***Передача паролів у вигляді хеш-значень*** іноді зустрічається на практиці, однак зазвичай не має сенсу – хеші паролів можуть бути перехвачені й повторно передані зловмисником по каналу зв'язку.
- 3. ***Передача паролів у зашифрованому виді*** в більшості є найбільш розумним і виправданим варіантом.

Завдання до лабораторної роботи

Ознайомитися з механізмами парольного захисту. Розробити блок-схему підсистеми парольного захисту, реалізувати модель у вигляді модуля парольного захисту (реалізувати непрямий алгоритм порівняння пароля).

Варі-ант	Функції підсистеми парольного захисту	Характеристика
1	*2*	*3*
1	Перевірка мінімальної довжини пароля, Перевірка допустимої кількості разів введення пароля	10-12 символів 7 разів
2	Відбраковування паролів (при введенні) за словником найбільш розповсюджених паролів Перевірка часу введення пароля:	Список найбільш розповсюджених паролів: вибрати 20 будь-яких паролів з додатку А. Не більше 3 хвилин, далі тимчасове блокування на 5 хвилин
3	Автоматична генерація незапам'ятовуваних паролів Перевірка допустимої кількості разів введення пароля	На основі генератора випадкових чисел 5 разів
4	Відбраковування пароля за низькою потужністю алфавіту паролів А. Перевірка часу введення пароля:	Алфавіт – не менше 68 символів Не більше 2 хвилин, далі тимчасове блокування на 7 хвилин
5	Відбраковування паролів (при введенні) за словником найбільш розповсюджених паролів Перевірка допустимої кількості разів введення пароля	Список найбільш розповсюджених паролів: вибрати за TOP-100 30 перших паролів з додатку А. 4 рази
6	Встановлення мінімального терміну дії нового пароля Перевірка часу введення пароля:	 Не менше 10 хвилин Не більше 2 хвилин, далі тимчасове блокування на 5 хвилин
7	Автоматична генерація паролів з чергуванням 1:2:1 голосних, приголосних букв та цифр Перевірка допустимої кількості разів введення пароля	На основі генератора випадкових чисел у три потоки (для голосних, приголосних та цифр) 3 рази
8	Автоматична генерація паролів з чергуванням 1:2 голосних і приголосних букв Перевірка часу введення пароля:	На основі генератора випадкових чисел у два потоки (для голосних і приголосних) Не більше 5 хвилин, далі тимчасове блокування на 5 хвилин
9	Відбраковування пароля за низькою потужністю алфавіту паролів А. Перевірка допустимої кількості разів введення пароля	Алфавіт – не менше 80 символів 6 разів

У **висновку** описати особливості реалізації завдання та варіанти застосування розробленого модуля.

Література

1. Крыжановская Ю.А. Программные методы защиты информации [Текст] / Ю.А. Крыжановская: Часть 1. – Врорнеж: Изд-во ВГУ, 2002. – 36 с.
2. Венбо Мао. Современная криптография: Теория и практика [Текст] / Мао Венбо: Пер. с англ. – М.: Издательский дом «Вильямс», 2005. – 768 с.
3. Цирлов В.Л. Основы информационной безопасности автоматизированных систем [Текст] / В.Л. Цирлов : краткий курс. – М.: Феникс, 2008. – 164 с.
4. Шнайер Б. Секреты и ложь. Безопасность данных в цифровом мире [Текст] / Брюс Шнайдер. – СПб.: Питер, 2003.
5. Столингс В. Криптография и защита сетей [Текст] / Вильям Столингс. – М.: Вильямс, 2001.
6. Лебедь С.В. Межсетевое экранирование. Теория и практика защиты внешнего периметра [Текст] / С.В. Лебедь. – М.: Издательство МГТУ имени Н.Э. Баумана, 2002.
7. Корт С.С. Теоретические основы защиты информации [Текст] / С.С. Корт. – М.: Гелиос АРВ, 2004.

8. Аутентификация [Электронный документ]. Режим доступа: http://ru.wikipedia.org/wiki/Аутентификация. Проверено: 12.03.12

Лабораторна робота № 5.

Розробка елементів підсистеми захисту програмного продукту від несанкціонованого доступу.

Мета роботи: ознайомити студента зі структурою та компонентами підсистеми захисту від несанкціонованого доступу (НСД), яка розробляється як вбудований модуль. Набути практичних навичок з реалізації захисту програмного продукту, який розроблений студентом, від НСД.

Завдання: Освоїти та реалізувати на практиці алгоритми захисту власного програмного продукту від НСД.

Теоретичні відомості

Технічний захист від несанкціонованого доступу реалізується за допомогою програмних та апаратних засобів. Система захисту від НСД є комплексом засобів, що призначаються для ускладнення (в ідеальному випадку – для недопущення) нелегального використання (запуску, копіювання) програмного продукту, що захищається. Структурна схема захисту програмних систем від НСД представлена на рисунку 5.

Рис 5. Структура системи захисту від несанкціонованого доступу.

Підсистема впровадження керуючих механізмів – комплекс програмних засобів для підключення впроваджуваного захисного коду до програмного модуля, що захищається. **Впроваджуваний захисний код** є програмним модулем, що протидіє спробам запуску нелегальної копії програми, що захищається.

Підсистема реалізації захисних функцій вирішує завдання розпізнавання легальності запуску програмного продукту, що захищається. **Підсистема протидії нейтралізації захисних механізмів** служить для боротьби зі спробами нейтралізації системи захисту від НСД і (чи) її дискредитації.

Блок встановлення характеристик середовища отримує ідентифікаційні характеристики обчислювального середовища. ***Блок порівняння характеристик середовища*** встановлює факт легальності запуску захищеної програми. ***Блок зворотної реакції***

відтворює відповідь системи захисту на спроби несанкціонованого виконання програми, що захищається.

При реальному розгляді питань захисту інформації треба розуміти, що *абсолютно стійкого захисту не існує*. Існують системи захисту, на нейтралізацію яких витрачається такий же час, як на розробку аналогічного ПЗ. Виходячи з цього твердження робимо висновок – нейтралізація захисту простих програмних продуктів з середнім рівнем захищеності менш імовірна, ніж складних програм з високим рівнем захисту.

Стійкість системи захисту визначається стійкістю кожного її елементу, для порушення захисних функцій може використовуватися будь-яка її підсистема. Різнорідність і різний рівень підсистем визначає найбільш слабкі елементи в системі захисту. Такими елементами є ***підсистема впровадження керуючих механізмів*** та ***підсистема встановлення характеристик середовища***. Інші підсистеми повинні бути розроблені не менш якісно. Розробка системи захисту повинна проводитися з урахуванням психології хакера, наприклад вона не повинна явно показувати, що програма захищена, що дозволить подовжити витрати часу і зусиль на розкриття системи захисту.

Часто найслабкішою ланкою в системі захисту інформації виявляється ***блок порівняння характеристик середовища***, а саме проти нього найчастіше спрямовані атаки хакерів. Як захистити цей блок?

По-перше, неможна перевіряти легальність копії за допомогою прямого порівняння кодової константи і змінної, що отримана з ***блоку встановлення характеристик середовища***, тому що інвертуванням наступної за порівнянням команди умовного переходу можна легко нейтралізувати захисний механізм.

По-друге, не рекомендується підтверджувати легальність копії простим поверненням керування у викликаючу програму. Це положення можна проілюструвати на прикладі програми нейтралізації захисту телефонного довідника “Соната”. Ця програма встановлюється як резидентний модуль, що отримує керування по 11h перериванню. Для ідентифікації використовується ключова дискета і електронний ключ. Тому при отриманні керування по перериванню вперше у стеку перевстановлюється адреса повернення для обходу перевірки ключової дискети, а вдруге у викликаючому модулі перша команда підпрограми перевірки наявності і правильності ключа

змінюється на retf (повернення з дальньої процедури). Передивимося код резидентної частини цієї програми:

```
Int_11h_entry  proc  far
        sti
        cmp   cs:Flag,0
        jne     Second_pass
        push  bp
        mov   bp,sp
        mov   [bp+2],5F6h
        pop     bp
        mov   cs:Flag,01
        iret
Second_pass:
        cmp   cs:Fiag,1
        jne     Int_11h_exit
                push  bx
                push  ds
                mov   ah,62h   ; повертає в bx адресу PSP поточної
                               ; програми (використовується для
                              ; отримання адреси параметрів команд-
                            ; ного рядка,адреси оточення DOS, та ін)
                int     21h         ; по перериванню 21h
                add     bx,10h     ; додає 10h
                mov   ds,bx
                les     bx,ds:[020Dh]; завантажує адресу, використо-
                                     ; вуючи es
                mov   byte ptr [bx],0CBh
                pop     ds
                pop     bx
        mov   cs:Flag,2
;
Int_11h_exit:
        db      0EAh          ; JMP  FAR  OLD_11h
old_11h_ofst        dw    ?
old_11h_sgmt        dw    ?
Flag  db      0
;
Int_11h_entry       endp
```

Підсистема впровадження керуючих механізмів

Системи захисту від НСД різняться по способу впровадження керуючих механізмів:

- вбудовані (створювані при формуванні програмного продукту),
- припасовані (пристиковочні), (підключаються до готового програмного продукту).

В наш час найчастіше зустрічаються ***припасовані*** системи. Це обумовлено:

- ✓ простотою тиражування програмних систем захисту на об'єкти замовника і розробника,
- ✓ простою технологією їх застосування,
- ✓ забезпеченням достатнього рівня захисту даних (розробники захисту – професіонали),
- ✓ оптимальне співвідношення "надійність / витрати на розробку " в порівнянні з вбудованими системами, створеними непрофесіоналами.

Алгоритми встановлення захисних модулів можуть бути різноманітними. Наприклад, встановлення припасованого модуля можна проводити по принципу найпростіших вірусів – дописувати код захисту в деяку область файлу, що захищається, проводити певне перетворення, у випадку обробки EXE-файлу – модифікувати його заголовок, а в якості першої виконуваної команди встановити команду переходу на добавлений код. Після проведення необхідних перевірок, код захисту встановлює оригінальний початок файлу і передає на нього керування, програма починає нормальну роботу.

У такому алгоритмі є суттєві недоліки: 1) виконуваний код програми, що захищається, залишається практично незмінним, що значно спрощує нейтралізацію захисту, 2) якщо захищається файл великого розміру і, як правило складної структури, код, що знаходиться наприкінці програми завантажений не буде, а значить він не буде працювати і в кращому випадку призведе до зависання комп'ютера (недолік можна усунути, якщо писати у початок виконуваного коду програми, що захищається, не одну команду переходу, а весь код захисту, при цьому зашифровуючи початковий файл). Єдиною умовою успішної реалізації запропонованого алгоритму є вимога, щоб програма, що захищається, не мала міжсегментних посилань і переходів.

Ще одним недоліком захисту даного типу є можливість динамічної нейтралізації такого захисту, тобто шляхом визначення

моменту, коли захисна частина вже відпрацювала і почав виконуватися основний код (це можливо, наприклад, по факту виклику певної функції певного переривання).

Іншим важливим недоліком описаних методів впровадження є те, що вони не підтримують програми, що самомодифікуються, які в процесі виконання змінюють свій образ, що зберігається на диску.

Основні вимоги, до захисних припасованих модулів (ПМ) :

- ✓ ПМ повинен підключатися до файлів будь-якого розміру,
- ✓ результуючий файл, отриманий після підключення ПМ, повинен мати таку побудову, щоб максимально ускладнити виділення початкової програми, що захищається,
- ✓ ПМ не повинен накладати обмежень на функціонування програми, що захищається; зокрема, він повинен дозволяти модифікувати в процесі роботи особистий дисковий файл.

Реалізувати необхідні вимоги можливо шляхом вбудовування ПМ по принципу вторинного завантажувача з шифруванням початкового модуля. Такий вторинний завантажувач виконує операції по завантаженню програми і її виконанню. Крім цього, вона вміщує резидентну частину (яку називають “інтелектуальним” обробником 21h переривання), яка здійснює прозоре шифрування-дешифрування програми, що захищається, у випадку його дочитування працюючим основним модулем, який дозволяє системі захисту працювати з програмами, що самомодифікуються.

Основна ідея створення припасованого модуля полягає у “дописуванні” його перед модулем, що захищається. При запуску програми, що захищається, в пам’яті залишається тільки код завантажувача, перед яким і стоїть задача подальшої обробки основного модуля. ПМ, що функціонує за таким принципом називають ***“вторинним” завантажувачем***.

Обробник 21h переривання є важливою складовою “вторинного” ПМ, тому що без нього неможливе початкове завантаження програмного продукту, що захищається. Він також необхідний для забезпечення режиму “прозорого” шифрування і можливості самомодифікації програми на диску. Обробник складається з таблиці реєстрації відкриттів файлу, що захищається; обробника стандартних функцій операційної системи; підпрограми шифрування-дешифрування. Для нормальної роботи програми, що захищається, обробник 21h переривань повинен:

- відслідковувати факти початку і кінця роботи із програмою, що захищається,
- забезпечувати режим “прозорого” шифрування,

- забезпечувати “правильне” позиціонування при роботі програми, що захищається,
- звільнення пам’яті захопленої під буфери вводу / виводу.

Програма, що захищається, у процесі роботи може відкривати себе декілька разів, тому обробник 21h переривання повинен вміщувати таблицю відкриття програми. При обробці функцій “відкрити файл”, “створити дублікат індексу файлу”, “активізувати дублікат індексу файлу” обробник повинен перевірити наявність вільного елементу в таблиці, і при його виявленні відкрити файл з наступною модифікацією таблиці. Після цього вказівник читання-запису переміщується на перший байт за кодом ПМ, програма, що захищається, вважає, що він має нульове зміщення відносно початку файлу.

Обробка функції “закрити файл” полягає в пошуку елементу таблиці з відповідним індексом файлу, звільнення його і безпосереднє закриття файлу.

При спробі зчитування файлу обробник порівнює отриманий індекс з наявним у таблиці, якщо він зареєстрований за програмою, що захищається, – після операції зчитування проводиться розшифровка зчитаної інформації (інакше інформація при зчитуванні не розшифровується).

При запису інформації в файл перевіряється, в який файл буде проводитися запис. Якщо це буде модифікація файлу, що захищається, то поводиться шифрування інформації з наступним записом її у файл. Після цього передані дані розшифровуються.

Підсистема протидії нейтралізації захисних механізмів.

Важлива функція протидії нейтралізації захисних механізмів в першу чергу пов’язана з протидією аналізу логіки і принципів роботи захисних механізмів. Дослідження програм може бути статичним (за допомогою дизасемблерів) і динамічним (за допомогою відладчиків і емуляторів). Захист від дизасемблювання успішно виконується за допомогою шифрування коду, що не гарантує захист від динамічного дослідження. Для протидії динамічному аналізу програм існують методи, при реалізації яких основною проблемою є визначення наявності відладчика. Для вирішення цієї проблеми існують такі способи:

- шифрування коду програми в присутності відладчика при нестандартному розташуванні стеку чи при використанні

конвеєрного принципу обробки команд центральним процесором,

- підрахунок контрольної суми ділянки коду на предмет виявлення точок призупинки,
- перевірка дійсної заборони апаратних переривань,
- зміна часу виконання ділянок програми,
- використання відладочних переривань (int1 та int 3) для особистих потреб,
- використання абсолютної адресації,
- порушення інтерфейсу з користувачем – блокування клавіатури і спотворення інформації при виводі на термінал,
- виявлення піднятого Т-прапорця через втрату трасувального переривання,
- змінення у PSP програми значення вказівника стеку після звернення до системних функцій в присутності відладчика.

Більшість наведених методів виявлення відладчиків спрацьовують тільки при покроковому виконанні програми, але на практиці важливі методи, що не залежать від режиму роботи відладчика.

Такі методи використовують наступні способи виявлення відладчика. Початкове значення регістрів, яке виставляє операційна система, AX=BX – відображає коректність імені диску аргументів командного рядка, при цьому AL ідентифікує коректність першого аргументу, а AH – другого. Відповідно AX, так як і BX, може приймати 4 значення: 0000, 00FF, FF00, FFFF. Важливо знати, що відладчики не підтримують BX=AX. Тому, якщо AX не дорівнює нулю і BX не дорівнює AX, то можна робити висновок про присутність відладчика. Значення інших регістрів DX=DS, SI=IP, DI=SP.

Не дивлячись на те, що практично всі регістри загального призначення, тобто DX, SI, DI (про AX і BX було сказано вище) при вході у програму мають ненульове значення, відладчики зазвичай встановлюють нульове значення їх початкового стану. Тому, для систем захисту з програмних модулів, що отримують керування перед програмою, що захищається, має сенс перевіряти початковий стан регістрів. Інші ознаки роботи відладчиків проявляються в операційному середовищі програми, тобто у векторах переривань, системних таблицях та областях даних.

Стандартним методом [4, інт.дж.], який використовується для виявлення засобів захисту від копіювання, є дизасемблювання програми встановлення програмного пакету чи виконання його під управ-

лінням покрокового відладчика. Лістинг, що отримується в процесі дизасемблювання, надає велику допомогу при використанні відладчика, тому ці два засоби – дизасемблювання та використання відладчика, зазвичай використовують разом. Відповідно потрібні окремі засоби для боротьби з дизасемблером та для захисту від відладчиків.

Для ускладнення дизасемблювання найкраще застосовувати шифрування окремих ділянок програм чи всієї програми в цілому. Наприклад, частину програми-інсталятора можна оформити у вигляді окремої COM-програми. Після трансляції початкового тексту цієї програми її можна зашифрувати певним способом і в зашифрованому вигляді відвантажувати у пам‘ять як програмний оверлей. Після завантаження програму слід розшифрувати в оперативній пам‘яті і передати їй управління.

Ще краще виконувати динамічне розшифровування програми по мірі її виконання, коли ділянки програми розшифровуються безпосередньо перед використанням і після використання зразу ж видаляються.

При розшифровуванні можна копіювати ділянки програми у інше місце оперативної пам‘яті. Наприклад, програма складається з кількох частин. Після завантаження її в оперативну пам‘ять управління передається першій частині програми. Ця частина призначена для розшифровки другої частини, яка розташовується у пам‘яті слідом за першою. Завдання другої частини – переміщення третьої частини програми на місце вже використаної першої частини і розшифровка її там. Третя частина, отримавши управління, може перевірити своє розташування відносно префіксу програмного сегменту і, у випадку правильного розташування (зразу за PSP), почати завантаження сегментних регістрів такими значеннями, які необхідні для виконання четвертої, інсталяційної частини програми. Якщо намагатися дизасемблювати програму, складену таким чином, то з цього нічого не вийде.

Другий спосіб боротьби з дизасемблером є по суті боротьбою з людиною, яка займається дизасемблюванням. Він полягає у збільшенні розміру завантажувального модуля до сотні-другої кілобайтів та в ускладненні структури програми.

Обсяг лістингу, що отримується при дизасемблюванні програми розміром у 30-40 кілобайт, досягає 1-1,5 мегабайти. Тому великі розміри інсталяційної програми можуть сильно збільшувати час виявлення засобів захисту. Що є ускладненням структури програми, зрозуміло само собою. Існує програма, що використовується для

звернення до одної і тої ж області пам‘яті, яка вміщує численні змінні, різні сегментні адреси. Тому дуже складно здогадатися, що практично програма працює з однією і тією ж областю пам‘яті.

Наступна – боротьба з трасуванням програми покроковими відладчиками. Стандартні відладчики реального режиму використовують для роботи два вектори:

- Int 1 - покроковий режим виконання;
- Int 3 - точка призупину.

Ваша задача – непомітно, не використовуючи стандартні засоби, які легко відслідковуються як самим відладчиком, так і хакером, змінити значення адреси цих векторів. Періодично необхідно перевіряти занесені значення. Покроковий режим виконання включається при встановленому прапорці TF процесора. Переключення цього прапорця в нуль призводить до відключення відладчика. Всі відладчики відслідковують команду popf і відновлюють значення прапорця TF, але дуже мало відладчиків розуміють змодельоване повернення з переривання, коли у стеку знаходиться слово прапорців з виключеним прапорцем трасування. Аналогічно, всі відладчики відслідковують команду pushf і очищають прапорець TF, але більшість відладчиків не розуміють команду cs:pushf чи es:pushf.

Ще більш ефективно використовувати вектори відладчика у власних цілях. Наприклад, ви можете перевизначити 21 вектор на 3 і звертатися до MSDOS не через int 21h, а через int 3h, це коротше на 1 байт і тому не дозволить зламнику провести зворотну заміну [4, інф.дж.].

Слово стану процесора:

17	16	15	14	13 12	11	10	9	8	7	6	5	4	3	2	1	0
VM	R		NT	I/O PL	OF	DF	IF	TF	SF	ZF		AF		PF		FC

VM - виртуальный режим

R - ошибка отладки

NT - вложенная задача

I/O PL - привилегии: 0 - высший, 3 - низший

OF - переполнение

DF - направление

IF - прерывания

TF - ТРАССИРОВКА

SF - знак

ZF - нуль

AF - вспомогат.перенос

PF - четность

FC - перенос

Ще один спосіб виявлення відладчика – замір часу виконання частин програми. При роботі під відладчиком час виконання значно більший. Це пов‘язано з тим, що при роботі під відладчиком проводяться додаткові дії, такі як натиснення клавіш на клавіатурі для активізації деяких дій відладчика. Замір часу краще проводити звертаючись безпосередньо через порти до таймера чи використовуючи CMOS-годинник.

Більшість відгадчиків використовують стек відлагоджуваної програми. Ви можете занести у невикористовувану частину стеку певну інформацію до ділянки програми, в якій не використовується стек, а після цієї ділянки перевірити [4]:

```
mov   bp,sp
            mov   ax,'a1'
            mov   [bp-2],ax

 ...    // не використовувати стек!!!

            cmp   word ptr [bp-2],'a1'
            jne     a61    // під відладчиком
```

При роботі в реальному режимі використання інструкцій виду mov sp, 1 призводить до генерації int 06 (invalid opcode). Ви можете маскувати переривання на період виконання критичної частини програми:

```
      CS:0100 E421      in     al,21h
      S:0102 0C02       or     al,00000010b      ;IRQ 1 keyboard irq
      CS:0104 E621             out    21h,al
чи
      CS:0100 E461      in     al,61
      CS:0102 0C80      or     al,10000000b      ; bit 7 - disable kbd
      CS:0104 E661             out    61h,al
чи
      CS:0100 B4AD             mov  al, 0ADh     ; disable keyboard
      CS:0102 E664             out    64h,al
```

Маскування немаскованого переривання NMI на AT:

```
      mov   al,0ADh
      out     70h,al
Дозволити NMI:
      mov   al,2Dh
      out     70h,al
```

Дуже гарні результати дає розпаковка наступних інструкцій по складному самомодифікованому алгоритму обробником Int 8 чи Int 1.

Необхідно не забувати про можливості закоментувати перевіряючи механізми у вашій програмі. Програма має періодично прораховувати і перевіряти контрольні суми ділянок програми.

Робота у захищеному режимі. Захищений режим роботи процесора відкриває перед вами нову можливість. Візьміть будь-яку програму, що працює у захищеному режимі, і спробуйте запустити її під управлінням якогось відладчика (наприклад, Turbo Debugger чи Code View). Все буде добре до тих пір, поки ваша програма не спробує загрузити регістр IDTR за допомогою команди LIDT. Після виконання цієї команди відладчик "зависає" і єдиний спосіб знов «оживити» комп‘ютер – натиснути на кнопку перезавантаження. Причина – змінилося розташування та формат дескрипторної таблиці переривань. Вона підготована для роботи в захищеному режимі, але відладчик працює у реальному режимі. Тому обробка всіх переривань, у тому числі й від клавіатури, неможлива. Ідея використання захищеного режиму роботи процесора при створенні програм, захищених від несанкціонованого доступу, копіювання, очевидна.

Наприклад, перед переключенням у захищений режим ви зможете підготувати у пам‘яті масив контрольної інформації. Розшифровка і перевірка цього масиву, а також запис даних у нестандартні сектори інсталяційного диску можуть виконуватися у захищеному режимі. При цьому, користуючись звичайними відладчиками неможливо визначити дії, що виконуються у захищеному режимі, особливо якщо ділянка програми, що працює у зашифрованому режимі зашифрована і після виконання затирається. Далі процесор можна повернути в реальний режим і продовжити процес інсталяції.

Знаходячись у захищеному режимі ви не можете звертатися до функцій DOS та BIOS, - ви можете читати і писати сектори дискети тільки з використанням рівня портів вводу/виводу контролера флопі-диску.

По боротьбі з відладчиками можна також порекомендувати і інші способи (Борьба с отладчиками // Журнал «Хакер» №057, режим доступу: http://www.xakep.ru/magazine/xs/057/084/1.asp).

При протидії вивченню логіки роботи програми найкращий принцип “чим більше, тим краще”, чим більше застосовано способів протидії, можливо і не самих надійних, тим менша імовірність нейтралізації захисту за прийнятний час.

Блок встановлення характеристик середовища.

Реалізація цього блоку залежить від способу розповсюдження програмного забезпечення. ПО може розповсюджуватися наступними способами:

1) безкоштовно (з альтруїзму і з міркувань самореклами),
2) умовно безкоштовно (оплата проводиться добровільно тоді коли користувач згоден з реальною користю для себе програмного продукту),
3) на комерційній основі.

Останній спосіб передбачає технічний захист програмних продуктів. Вони можуть розповсюджуватись такими способами:

1) за допомогою спеціальної служби розповсюдження,
2) через торгові організації,
3) через вільне розповсюдження дистрибутивних (демонстраційних) пакетів з наступною реєстрацією (користувач отримує якимсь чином ПО, при його запуску зчитує код оплати, який разом з квитанцією про оплату посилає розробнику, і отримує код-пароль для реєстрації програми і асоціації її з комп'ютером).

Спосіб розповсюдження визначає конкретну реалізацію блоку встановлення характеристик середовища. При розповсюдженні ПО за допомогою спецслужби розповсюдження використовуються організаційні заходи захисту інформації. Незручністю даного способу вважається необхідність виїзду до замовника працівника фірми-розповсюджувача у разі виникнення збоїв. Блок встановлення характеристик середовища повинен вміти ідентифікувати параметри комп'ютера.

При розповсюдженні програм через торгові організації:

1) програма пов'язує себе з дистрибутивним носієм без ідентифікації конкретного комп'ютера,
2) програма асоціює себе з спеціальним апаратним пристроєм, що підключається до комп'ютера і входить до дистрибутивного комплекту,
3) програма пов'язує себе як з дистрибутивним носієм (при інсталяції), так і з параметрами комп'ютера (у робочому режимі).

Перші два варіанти дозволяють переносити захищені програми з комп'ютера на комп'ютер, але вимагають присутності ключового диску чи спеціального апаратного пристрою захисту.

Третій варіант вимагає, крім забезпечення надійності ключового диску, розв'язання не простої проблеми – лічильника інсталяцій.

Таким чином, по вибраному способу розповсюдження ПЗ визначаються функції блоку встановлення характеристик середовища:

- ідентифікація параметрів комп’ютера,
- ідентифікація параметрів дистрибутивного носія,
- розпізнавання спеціального апаратного пристрою.

Ідентифікація параметрів комп‘ютера

Ідентифікація параметрів IBM-сумісних комп’ютерів не є однозначною – у них відсутні унікальні характеристики (такі наприклад, як заводський номер), що дозволяють відрізнити одну машину від другої. В деяких комп’ютерах є певні однозначні характеристики, наприклад машини фірми AMI визначаються по номеру набору мікросхем на материнській платі (chipset).

Для ідентифікації IBM-сумісних ПК їх або оснащують додатковими пристроями (електронними ключами та ін.), чи виділяють характеристики, які можливо вважати унікальними (цей спосіб дешевший але менш надійний). Серед таких характеристик різнять статичні (не залежать від зовнішніх впливів) та динамічні.

Динамічні характеристики **–** швидкість обертання вінчестера, точна тактова частота процесора та ін. Ці характеристики в значній мірі залежать від температури, вологості, напруження та частоти живлення мережі, зносу окремих частин ПК, тому на даний час у системах захисту інформації не використовуються.

Статичні характеристики:

- тип мікропроцесора з розрядністю шини даних,
- тип співпроцесора для операцій з плаваючою крапкою,
- тактова частота процесора з точністю до одиниць мегагерц,
- довжина черги процесора,
- тип ПК,
- дата реєстрації BIOS,
- сигнатура виробника BIOS та її адреса у пам’яті,
- розмір основної оперативної пам’яті,
- розмір розширеної пам’яті,
- розмір додаткової відображуваної пам’яті,
- тип клавіатури,
- тип відеоадаптера,
- тип та інтерфейс маніпулятора “мишка” чи “світлове перо”,
- число паралельних портів (портів принтера),
- число послідовних портів,
- число та тип накопичувача на гнучких дисках,
- число та тип накопичувачів на жорстких дисках.

Безпосередня ідентифікація в сучасних системах захисту від НСД майже не використовується. Ідентифікація комп‘ютера потрібна

в мережі, де є імовірність того, що зловмисник може видати свою машину за легально підключений до сервера пристрій (комп‘ютер, маршрутизатор та ін.). Це можливо двома способами. По-перше, скористатися IP-адресою, що знаходиться в межах діапазону санкціонованих IP-адрес, чи авторизованою зовнішньою адресою, якій дозволений доступ до певних мережевих ресурсів. Атаки цього типу часто є відправною точкою для інших атак.

Зазвичай підміна довіреного об‘єкту мережі обмежується вставкою недостовірної інформації чи шкідливих команд у звичайний потік даних, що передається між об‘єктами мережі. Для двостороннього зв‘язку зловмисник має змінити всі таблиці маршрутизації, щоб спрямувати трафік на помилкову IP-адресу, що теж є можливим. Для ослаблення загрози (але не її ліквідації) можна скористатися наступним [5, інт.дж.]:

- **Контроль доступу** будується на відсіканні будь-якого трафіку, що поступає із зовнішньої мережі з почасовою адресою внутрішньої мережі. Цей метод є дієвим, якщо санкціоновані тільки внутрішні адреси і не працює, якщо є санкціоновані зовнішні адреси.
- **Фільтрація RFC 2827** – даний тип фільтрації дозволяє усунути спроби спуфінгу чужих мереж користувачами вашої мережі. Для цього необхідно відбраковувати будь-який вихідний трафік, початкова адреса якого не є однією з IP-адрес вашої організації. Часто цей тип фільтрації виконується провайдером. В результаті відбраковується весь трафік, який не має початкової адреси, що очікується на певному інтерфейсі. Наприклад, якщо ISP надає з‘єднання з IP-адресою 15.1.1.0/24, він може налаштувати фільтр таким чином, щоб з даного інтерфейсу на маршрутизатор ISP допускався тільки трафік, що поступає с адреси 15.1.1.0/24. При цьому, до тих пір, поки всі провайдери не запровадять цей тип фільтрації, його ефективність буде набагато нижча можливої.
- **Запровадження додаткових методів аутентифікації**. IP-spoofing можливий тільки у випадку аутентифікації на основі IP. Якщо ввести якісь додаткові заходи по аутентифікації, наприклад, криптографічні, атака стає марною.

Завдання до лабораторної роботи

На основі механізмів, реалізованих на лабораторних роботах № 3 и 4, та структури підсистеми захисту програмного продукту від НСД, розробити структурну схему та реалізувати у вигляді модуля вбудовану підсистему захисту від НСД з наступними характеристиками:

Варі-ант	Підсистема реалізації захисних функцій	Підсистема нейтралізації захисних механізмів
1	Парольний захист	1) Блокування системи після вводу 5 неправильних паролів. 2) Механізм розблокування основної програми розробником.
2	Захист з відсіканням зовнішнього трафіку	Блокування роботи програми при спробі прийняти пакет із зовнішньої мережі. (при блокуванні – механізм обираєте самостійно)
3	Парольний захист	Зашифровування основної програми з ключем не менше 20 символів (метод з лаб. №3) – тобто організація повного блокування
4	Захист з прив'язкою до характеристик локальної мережі	1) Повне блокування основної програми. 2) Механізм розблокування основної програми розробником.
5	Парольний захист	Лічильник копій. (максимум 3 копії)
6	Захист з прив'язкою до характеристик комп'ютера (мінімум дві характеристики)	Лічильник копій. (максимум 4 копії)
7	Парольний захист	Можливість перегляду демоверсії (деяких функцій основної програми) при неправильному введенні пароля. (блокування всіх інших функцій – механізм обираєте самостійно)
8	Захист з контролем внутрішнього трафіку	Зашифровування основної програми з ключем не менше 15 символів (метод з лаб. №3) – тобто організація повного блокування
9	Парольний захист	Після вводу 6 неправильних паролів – перезавантаження комп'ютера з деінсталяцією програмного продукту.

У **висновку** описати особливості реалізації завдання та варіанти застосування розробленого модуля.

Література

1. Крыжановская Ю.А. Программные методы защиты информации [Текст] / Ю.А. Крыжановская: Часть 1. – Врорнеж: Изд-во ВГУ, 2002. – 36 с.
2. Венбо Мао. Современная криптография: Теория и практика [Текст] / Мао Венбо: Пер. с англ. – М.: Издательский дом «Вильямс», 2005. – 768 с.
3. Столингс В. Криптография и защита сетей [Текст] / Вильям Столингс. – М.: Вильямс, 2001.

4. Новичков А., Сардарян Р. Анализ рынка средств защиты от копирования и взлома программных средств // PC Week. - №6. – 2004.
5. Коваленко М.М. Комп’ютерні віруси і захист інформації. – Київ: Наукова думка, 1999.

Internet – посилання

1. Бібліотека інформаційної безпеки. [Електронний документ]. Режим доступу: http://bib.pps.ru/. Перевірено 11.05.2012.
2. Хакер. [Электронный ресурс]. Режим доступа: http://www.xakep.ru/ (журнал з інформаційної безпеки).
3. Schneier.com: Applied Cryptography by Bruce Schneier. Available at: http://www.schneier.com/book-applied.html (accessed 15 May 2012).
4. Колексников Д.Г. Защита от отладчиков. [Электронный документ]. Режим доступа: http://protect.htmlweb.ru/p43.htm. Проверено 03.02.2012.
5. Скрипник Д. Общие вопросы технической защиты информации [Электронный документ]. Режим доступа: http://www.intuit.ru/studies/courses/2291/591/info. Проверено: 02.02.2012

6. Список рекомендованої літератури

1. Коваленко М.М. Комп'ютерні віруси і захист інформації. – Київ : Наукова думка, 1999.
2. Хореев О.В. Криптографические интерфейсы и их использование. – М.: Горячая линия – Телеком, 2007. – 278 с.
3. Касперски К. Техника и философия хакерских атак. – М.: СОЛОН – Р, 2001. – 272 с.
4. Касперски К. Записки исследователя компьютерных вирусов. – СПб.: Питер, 2005. -316 с.
5. Венбо Мао. Современная криптография: Теория и практика [Текст] / Мао Венбо: Пер. с англ. – М.: Издательский дом «Вильямс», 2005. – 768 с.
6. Столингс В. Криптография и защита сетей [Текст] / Вильям Столингс. – М.: Вильямс, 2001.
7. Крыжановская Ю.А. Программные методы защиты информации. Часть 1. – Врорнеж: Изд-во ВГУ, 2002. – 36 с.
8. Новичков А., Сардарян Р. Анализ рынка средств защиты от копирования и взлома программных средств // PC Week. – №6. – 2004.
9. Баланс С. Пароли для профессионалов. [Электронный документ]. Режим доступа: http://bugtraq.ru/library/security/passunleashed.html Проверено 2.02. 2011.

Internet – посилання

1. Электронная библиотека попечительского совета механико-математического факультета МГУ [Электронный документ]. Режим доступа: http://lib.mexmat.ru/books/780. Проверено 11.05.2012.
2. Бібліотека інформаційної безпеки. [Електронний документ]. Режим доступу: http://bib.pps.ru/. Перевірено 11.05.2012.
3. Planet PDF. U.S. Department of Justice selects Appligent Redax for PDF redaction. Available at: http://www.planetpdf.com/mainpage.asp?webpageid=2450. (accessed 12 May 2012).
4. Kurt Foss. Makeshift PDF Redaction Exposes 'Secret' Government Info – Again. Available at: http://www.planetpdf.com/mainpage.asp?webpageid=3177 (accessed 12 May 2012).
5. Kurt Foss. PDF Secrets Revealed. Available at: http://www.planetpdf.com/mainpage.asp?webpageid=808. (accessed 12 May 2012).

6. RARLAB. WinRAR archiver, a powerful tool to process RAR and ZIP flies. Available at: http://www.rarlab.com (accessed 15 May 2012).
7. Zero, SantMat. The Reverse-Engineering Academy. Available at: http://www.reverser-course.de (accessed 15 May 2012).
8. RSA Laboratories. Factorization of RSA-155. Available at: http://www.rsasecurity.coin/rsalabs/challenges/factoring/rsal55.html (accessed 15 May 2012).
9. Schneier.com: Applied Cryptography by Bruce Schneier. Available at: http://www.schneier.com/book-applied.html (accessed 15 May 2012).
10. SealedMedia – Complete document protection and control, even after delivery. Available at: http://www.sealedmedia.com (accessed 18 May 2012).
11. Silicon Realms. The Armadillo Software Protection System. Available at: http://www.siliconrealms.com/armadillo.shtml (accessed 18.05.2012)
12. SlySoft – CloneCD. Available at: http://www.slysoft.com/en/clonecd.html – (accessed 18 May 2012).
13. Павел Семьянов. Брюс Шнайер. Прикладная криптографія. [Электронный документ]. Режим доступа: http://www.ssl.stu.neva.ru/ psw/ crypto/appl_rus/appl_cryp.htm. Проверено 23.12.2012.
14. Закон України про основи національної безпеки від 19.06.03. [Електронний документ]. Режим доступу: http://www.nbuv.gov.ua/law/03_nbu.html. Перевірено 21.05.2012.
15. Інформаційна безпека України [Електронний документ]. Режим доступу: http://uk.wikipedia.org/wiki/Інформаційна_безпека_України. Перевірено 21.05.2012.
16. Литература по криптографии. [Электронный документ]. Режим доступа: http://www.diary.ru/~eek/p76084072.htm. Проверено 23.12.2012.
17. Список публикаций и научных работ Сергея Панасенко (по шифрованию и защите данных). [Электронный документ]. Режим доступа: http://www.panasenko.ru/page_articles.html. Проверено 25.12.12.
18. Хакер. [Электронный ресурс]. Режим доступа: http://www.xakep.ru/ (журнал з інформаційної безпеки).
19. Информационная безопасность. [Электронный ресурс]. Режим доступа: http://www.itsec.ru/main.php. Проверено 25.12.12.
20. SecuritiLab. [Электронный ресурс]. Режим доступа: http://www.securitylab.ru/. Проверено 25.12.12.
21. Независимый информационно-аналитический центр информационной безопасности. [Электронный ресурс]. Режим доступа: http://www.anti-malware.ru/. Проверено 25.12.12.

ДОДАТОК А

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРКАСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
імені БОГДАНА ХМЕЛЬНИЦЬКОГО

Факультет обчислювальної техніки,
інтелектуальних та управляючих систем

Кафедра програмного забезпечення
автоматизованих систем

Журнал звітів з лабораторних робіт з дисципліни «Безпека програм та даних»

Перевірив:
доц. Супруненко О.О.

_____
(Підпис)
"___"_____2013 р.

Виконав:
студент Вакула О.М.
група КС-081

_____
(Підпис)
"___"_____2013 р.

Черкаси 2013

ДОДАТОК Б

Приклад оформлення звіту з лабораторної роботи

Вакула О.М., гр. КС-091

Лабораторна робота № 3*.

Програмні засоби захисту інформаційних систем

Завдання до лабораторної роботи: Ознайомитися з методом шифрування Цезаря (зміщення 8). Розробити блок-схему алгоритму шифрування та розшифровування, реалізувати формальні моделі у вигляді двох підсистем модуля з мінімальним інтерфейсом.

1. Опис методу реалізації завдання

<У розділі описуються методи та алгоритми (схеми та алгоритми в графічному вигляді), що використовуються при створенні модуля захисту, пояснюються етапи програмної реалізації задачі, коротко коментуються особливості розв'язання поставленого завдання.>

2. Алгоритм реалізації завдання

<У розділі наводиться граф-схема алгоритму для реалізації програмного модуля, проводиться її короткий опис.>

3. Лістинг програми**

<У розділі наводиться лістинг програмного модуля з необхідними коментарями.>

4. Результати виконання програми.

< У розділі наводиться копії екрану, що зроблені на різних етапах роботи захисного модуля, отримані при тестування в стандартних та нестандартних умовах.>

4. Висновок.

<У розділі робиться висновок, у якому проводиться аналіз тестування захисного модуля, зазначаються сфери можливого застосування.>

---

* Звіт має бути максимально конкретним та стислим.

** Даний пункт є необов'язковим в лабораторних роботах №1 та №2.

ДОДАТОК В

ПРИКЛАД ОФОРМЛЕННЯ СПИСКУ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

Список інформаційних джерел оформлюється у відповідності з **ДСТУ ГОСТ 7.1:2006** Система стандартів з інформації, бібліотечної та видавничої справи. БІБЛІОГРАФІЧНИЙ ЗАПИС. БІБЛІОГРАФІЧНИЙ ОПИС. Загальні вимоги та правила складання.

Новий державний стандарт України (ДСТУ) введений на заміну 5-и існуючих раніше стандартів на опис нотних видань, картографічних творів, ізовидань, аудіовізуальних матеріалів, печатних видань. ДСТУ ГОСТ 7.1:2006 відображає бібліографічні описи всіх видів документів, в тому числі й електронних, а також частини документу чи групи документів.

В ДСТУ наведені приклади бібліографічних описів різних документів, зі всіма областями опису та знаками пунктуації.

Деякі приклади бібліографічного опису:

- **Книга одного автора:**

Андреев, В. В. Как организовать делопроизводство на предприятии [Текст] / В. В. Андреев. – М.: ИНФРА-М, 1997. – 94 с.

- **Книга двох авторів:**

Белов, А. В. Финансы и кредит [Текст]: учеб. / А. В. Белов, В. Н. Николаев ; КНУ им. Т. Г. Шевченко. – К.: Университет, 2004. – 215 с. – Библиогр. : с. 213-215. – ISBN 5-7042-1441-Х.

- **Книга трьох авторів:**

Агафонова, Н. Н. Гражданское право [Текст]: учеб, пособие / Н. Н. Агафонова, Т. В. Богачева, Л. И. Глушкова ; под общ. ред. А. Г. Калпина; МОН Украины. – 2-е изд., перераб. и доп. – Х.: Фактор, 2000. – 542 с. – (Университетская книга).

- **Книга чотирьох авторів:**

Елементи інформатики [Текст]: довідник / В. С. Височанський, А. І. Кардаш, В. С. Костев, В. В. Черняхівський . – К.: Наук, думка, 2003. – 192 с.

- **Книга п’яти авторів і більше:**

Коротковолновые антенны [Текст]: учеб, пособие / Г. 3. Айзенберг, С. П. Белоусов, Я. М. Журбин и др. ; под общ. ред. А. А. Стогния. – 2-е изд. – М.: Радио и связь, 2003. – 192 с.

- **Перекладене видання:**

Нойман, Э. Происхождение и развитие сознания [Текст]: пер. с англ. – К.: Ваклер ; М.: Реал-бук, 1998. – 462с.

- **Книги під заголовком:**

Информационные технологии в маркетинге [Текст]: учеб. / под ред. Г. А. Титаренко. – М.: ЮНИТИ, 2000. – 335 с. – (Texbook). – 13ВК 5-238-00154-1

- **Багатотомне видання, окремий том:**

Савельев, И.В. Курс общей физики [Текст]. Т. 1. Механика. Молекулярная физика: учеб, пособие / И. В. Савельев. – 2-е изд., перераб. – М.: Наука, 1982. – 432 с.

- **Статті з журналів:**

Гончаров, В. А. Численная схема моделирования дозвуковых течений вязкого сжимаемого газа [Текст] / В. А. Гончаров, В. М. Кравцов // Журнал вычисл. математики и мат. физики. – 1988. – Т. 28, №12. – С. 1858-1866.

Анализ направляемого движения электрической дуги по массивному электроду, покрытому тонким слоем изолятора [Текст] // Прикладная физика. – 2001. – № 3. – С. 58-67.

- **Збірники наукових праць:**

Отчет о выполнении плана научно-исследовательских работ за 2003 год [Текст]: сб. науч. тр. / Рос. Акад. мед. паук, Сиб. отд. – Новосибирск : СО РАМН, 2004. – 83 с.

- **Тези конференцій:**

Образование, наука, производство: пути углубления интеграции и повышения качества инженерного образования [Текст]: тез. докл. науч.-практ. конф. (окт. 2000) / отв. ред. В. Г. В до вен ко. – Красноярск: САА, 2000. – 53 с.

- **Матеріали конференцій:**

Проблемы экономики, организации и управления реструктуризацией и развитием предприятий промышленности, сферы услуг и коммунального хозяйства [Текст]: материалы IV междунар. науч.-практ. конф., 30 марта 2005 г. Новочеркасск / редкол.: Б. Ю. Серебряков (отв. ред.). – Новочеркасск: Темп, 2005. – 58 с.

- **Стандарти, техніко-економічні й технічні документи:**

ГОСТ Р 517721-2001. Аппаратура радиоэлектронная бытовая. Входные и выходные параметры и типы соединений. Технические требования [Текст]. – Введ. 2002-01-01. – М.: Изд-во стандартов, 2001. — 27 с.

Инструкция по проектированию, строительству и эксплуатации гидротехнических сооружений на подрабатываемых горными работами территориях [Текст]: СП 522-85. – Утв. Госстроем СССР 03.05.86. – Изд. офиц. – М.: Стройиздат. 1986. – 32 с.

Електронні ресурси:

- **Віддаленого доступу:**

Основные направления исследований, основанные на семантическом анализе текстов [Электронный ресурс] / С.-Петерб. гос. ун-т, факультет прикладной математики и процессов управления. – Режим доступа: \www/ URL: http://apmath.spdu/ru/ru/staff/tuzov/onapr.html/ – 10.12.2004 г. – Загл. с экрана.

- **Локального доступу:**

Internet шаг за шагом [Электронный ресурс] : интеракт. учеб. – Электрон, дан. и прогр. -СПб.: Питер Ком, 1997. – 1 электрон, опт. диск (CD-ROM). – Систем, требования: ПК от 486 DX 66 МГц ; RAM 1616 Мб ; Windows 95; зв. плата. – Загл. с этикетки диска.

Джерела рекомендовано розміщувати у порядку подання посилань у тексті роботи.

ДОДАТОК Г

Нормативні документи системи захисту інформації

1. **Закони України:**
   - *a)* Закон України "Про Державну службу спеціального зв'язку та захисту інформації України"
   - *b)* Закон України "Про захист інформації в інформаційно-телекомунікаційних системах"
   - *c)* Закон України "Про Національну систему конфіденційного зв'язку"
   - *d)* Закон України "Про інформацію"
   - *e)* Закон України «Про науково-технічну інформацію»
   - *f)* Закон України "Про телекомунікації"
   - *g)* Закон України «Про Національну програму інформатизації»
   - *h)* Закон України «Про захист інформації в автоматизованих системах»
   - *i)* Закон України "Про радіочастотний ресурс України"
   - *j)* Закон України "Про основні засади розвитку інформаційного суспільства в Україні на 2007-2015 роки"
   - *k)* Закон України "Про Дисциплінарний статут Державної служби спеціального зв'язку та захисту інформації України"
   - *l)* Закон України "Про державну таємницю"
   - *m)* Закон України "Про ліцензування певних видів господарської діяльності"
   - *n)* Закон України "Про електронні документи та електронний документообіг"
   - *o)* Закон України "Про електронний цифровий підпис"
   - *p)* Закон України "Про наукову і науково-технічну експертизу"
   - *q)* Закон України "Про державний контроль за міжнародними передачами товарів військового призначення та подвійного використання"
   - *r)* Закон України "Про ратифікацію Статуту і Конвенції міжнародного союзу електрозв’язку"

2. **Укази президента:**
   - *a)* Указ Президента України від 27.09.1999 № 1229 "Про Положення про технічний захист інформації в Україні"
   - *b)* Указ Президента України від 10.04.2000 №582 "Про заходи щодо захисту інформаційних ресурсів держави"

3. **Постанови КМУ:**
   - *a)* Постанова Кабінету міністрів України «Про затвердження Правил забезпечення захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах»
   - *b)* Постанова Кабінету міністрів України «Про затвердження Інструкції про порядок обліку, зберігання і використання документів, справ, видань та інших матеріальних носіїв інформації, які містять конфіденційну інформацію, що є власністю держави»

4. **Нормативні документи в галузі технічного захисту інформації (НД ТЗІ) та державні стандарти України (ДСТУ) стосовно створення і функціонування комплексної системи захисту інформації:**
   a) НД ТЗІ 3.7-003-05 Порядок проведення робіт із створення комплексної системи захисту інформації в інформаційно-телекомунікаційній системі
   b) ДЕРЖАВНИЙ СТАНДАРТ УКРАЇНИ Захист інформації. Технічний захист інформації. Основні положення. ДСТУ 3396.0-96
   c) ДЕРЖАВНИЙ СТАНДАРТ УКРАЇНИ Захист інформації. Технічний захист інформації. Порядок проведення робіт. ДСТУ 3396.1-96
   d) ДЕРЖАВНИЙ СТАНДАРТ УКРАЇНИ Захист інформації. Технічний захист інформації. Терміни та визначення. ДСТУ 3396.2-97
   e) НД ТЗІ 1.4-001-2000 Типове положення про службу захисту інформації в автоматизованій системі
   f) НД ТЗІ 2.5-004-99 Критерії оцінки захищеності інформації в комп'ютерних системах від несанкціонованого доступу
   g) НД ТЗІ 2.5-005-99 Класифікація автоматизованих систем і стандартні функціональні профілі захищеності оброблюваної інформації від несанкціонованого доступу
   h) НД ТЗІ 2.5-008-02 Вимоги із захисту конфіденційної інформації від несанкціонованого доступу під час оброблення в автоматизованих системах класу 2
   i) НД ТЗІ 2.5-010-03 Вимоги до захисту інформації WEB-сторінки від несанкціонованого доступу
   j) НД ТЗІ 3.7-001-99 Методичні вказівки щодо розробки технічного завдання на створення комплексної системи захисту інформації в автоматизованій системі
   k) НД ТЗІ 3.6-001-2000 Технічний захист інформації. Комп'ютерні системи. Порядок створення, впровадження, супроводження та модернізації засобів технічного захисту інформації від несанкціонованого доступу
   l) АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ ТРЕБОВАНИЯ К СОДЕРЖАНИЮ ДОКУМЕНТОВ РД 50-34.698
   m) ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА СОЗДАНИЕ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ГОСТ 34.602-89
   n) НД ТЗІ 1.1-003-99. Термінологія в галузі захисту інформації в комп'ютерних системах від несанкціонованого доступу;
   o) НД ТЗІ 1.1-002-99 Загальні положення щодо захисту інформації в комп'ютерних системах від несанкціонованого доступу.

5. **Нормативні документи в галузі технічного захисту інформації (НД ТЗІ) та державні стандарти України (ДСТУ) по захисту інформації на об'єктах інформаційної діяльності**
   a) НД ТЗІ 1.6-005-2013 Захист інформації на об'єктах інформаційної діяльності. Положення про категоріювання об'єктів, де циркулює інформація з обмеженим доступом, що не становить державної

таємниці”, затверджене наказом Адміністрації Держспецзв'язку від 15.04.2013 № 215.

b) ДБН А.2.2-2-96 Проектування. Технічний захист інформації. Загальні вимоги до організації проектування та проектної документації для будівництва.

c) НД ТЗІ 1.1-005-07 Захист інформації на об’єктах інформаційної діяльності. Створення комплексу ТЗІ. Основні положення.

d) НД ТЗІ 1.6-003-04 Створення комплексів технічного захисту інформації на об’єктах інформаційної діяльності. Правила розроблення, побудови, викладення та оформлення моделі загроз для інформації.

e) НД ТЗІ 2.1-002-07 Захист інформації на об’єктах інформаційної діяльності. Випробування комплексу ТЗІ. Основні положення.

f) НД ТЗІ 2.5-006-99 Класифікатор засобів копіювально-розмножувальної техніки.

g) НД ТЗІ 2.7-002-99 Методичні вказівки з використання засобів копіювально-розмножувальної техніки.

h) НД ТЗІ 3.1-001-07 Захист інформації на об’єктах інформаційної діяльності. Створення комплексів технічного захисту інформації. Перед проектні роботи.

i) НД ТЗІ 3.3-001-07 Захист інформації на об’єктах інформаційної діяльності. Створення комплексу технічного захисту інформації. Порядок розроблення та впровадження заходів із захисту інформації.

j) НД ТЗІ 2.7-011-12 Захист інформації на об’єктах інформаційної діяльності. Методичні вказівки з розробки Методики виявлення закладних пристроїв.

k) НД ТЗІ 1.1-005-07 Захист інформації на об’єктах інформаційної діяльності. Створення комплексу ТЗІ. Основні положення.

За матеріалами сайту http://www.dstszi.gov.ua

ДОДАТОК Д

Список найбільш розповсюджених паролів:

№	Top 1-100	Top 101–200	Top 201–300	Top 301–400	Top 401–500
1	*2*	*3*	*4*	*5*	*6*
1	123456	porsche	firebird	prince	rosebud
2	password	guitar	butter	beach	jaguar
3	12345678	chelsea	united	amateur	great
4	1234	black	turtle	7777777	cool
5	pussy	diamond	steelers	muffin	cooper
6	12345	nascar	tiffany	redsox	1313
7	dragon	jackson	zxcvbn	star	scorpio
8	qwerty	cameron	tomcat	testing	mountain
9	696969	654321	golf	shannon	madison
10	mustang	computer	bond007	murphy	987654
11	letmein	amanda	bear	frank	brazil
12	seball	wizard	tiger	hannah	lauren
13	master	xxxxxxxx	doctor	dave	japan
14	michael	money	gateway	eagle1	naked
15	football	phoenix	gators	11111	squirt
16	shadow	mickey	angel	mother	stars
17	monkey	bailey	junior	nathan	apple
18	abc123	knight	thx1138	raiders	alexis
19	pass	iceman	porno	steve	aaaa
20	fuckme	tigers	badboy	forever	bonnie
21	6969	purple	debbie	angela	peaches
22	jordan	andrea	spider	viper	jasmine

1	*2*	*3*	*4*	*5*	*6*
23	harley	horny	melissa	ou812	kevin
24	ranger	dakota	booger	jake	matt
25	iwantu	aaaaaa	212	lovers	qwertyui
26	jennifer	player	flyers	suckit	danielle
27	hunter	sunshine	fish	gregory	beaver
28	fuck	morgan	porn	buddy	4321
29	2000	starwars	matrix	whatever	4128
30	test	boomer	teens	young	runner
31	batman	cowboys	scooby	nicholas	swimming
32	trustno1	edward	jason	lucky	dolphin
33	thomas	charles	walter	helpme	gordon
34	tigger	girls	cumshot	jackie	casper
35	robert	booboo	boston	monica	stupid
36	access	coffee	braves	midnight	shit
37	love	xxxxxx	yankee	college	saturn
38	buster	bulldog	lover	baby	gemini
39	1234567	ncc1701	barney	cunt	apples
40	soccer	rabbit	victor	brian	august
41	hockey	peanut	tucker	mark	3333
42	killer	john	princess	startrek	canada
43	george	johnny	mercedes	sierra	blazer
44	sexy	gandalf	5150	leather	cumming
45	andrew	spanky	doggie	232323	hunting
46	charlie	winter	zzzzzz	4444	kitty
47	superman	brandy	gunner	beavis	rainbow
48	asshole	compaq	horney	bigcock	112233

1	*2*	*3*	*4*	*5*	*6*
49	fuckyou	carlos	bubba	happy	arthur
50	dallas	tennis	2112	sophie	cream
51	jessica	james	fred	ladies	calvin
52	panties	mike	johnson	naughty	shaved
53	pepper	brandon	xxxxx	giants	surfer
54	1111	fender	tits	booty	samson
55	austin	anthony	member	blonde	kelly
56	william	blowme	boobs	fucked	paul
57	daniel	ferrari	donald	golden	mine
58	golfer	cookie	bigdaddy	0	king
59	summer	chicken	bronco	fire	racing
60	heather	maverick	penis	sandra	5555
61	hammer	chicago	voyager	pookie	eagle
62	yankees	joseph	rangers	packers	hentai
63	joshua	diablo	birdie	einstein	newyork
64	maggie	sexsex	trouble	dolphins	little
65	biteme	hardcore	white	00	redwings
66	enter	666666	topgun	chevy	smith
67	ashley	willie	bigtits	winston	sticky
68	thunder	welcome	bitches	warrior	cocacola
69	cowboy	chris	green	sammy	animal
70	silver	panther	super	slut	broncos
71	richard	yamaha	qazwsx	8675309	private
72	fucker	justin	magic	zxcvbnm	skippy
73	orange	banana	lakers	nipples	marvin
74	merlin	driver	rachel	power	blondes

1	*2*	*3*	*4*	*5*	*6*
75	michelle	marine	slayer	victoria	enjoy
76	corvette	angels	scott	asdfgh	girl
77	bigdog	fishing	2222	vagina	apollo
78	cheese	david	asdf	toyota	parker
79	matthew	maddog	video	travis	qwert
80	121212	hooters	london	hotdog	time
81	patrick	wilson	7777	paris	sydney
82	martin	butthead	marlboro	rock	women
83	freedom	dennis	srinivas	freedom	dennis
84	ginger	fucking	internet	extreme	magnum
85	blowjob	captain	action	redskins	juice
86	nicole	bigdick	carter	erotic	abgrtyu
87	sparky	chester	jasper	dirty	777777
88	yellow	smokey	monster	ford	dreams
89	camaro	xavier	teresa	freddy	maxwell
90	secret	steven	jeremy	arsenal	music
91	dick	viking	11111111	access14	rush2112
92	falcon	snoopy	bill	wolf	russia
93	taylor	blue	crystal	nipple	scorpion
94	111111	eagles	peter	iloveyou	rebecca
95	131313	winner	pussies	alex	tester
96	123123	samantha	cock	florida	mistress
97	bitch	house	beer	eric	phantom
98	hello	miller	rocket	legend	billy
99	scooter	flower	theman	movie	6666
100	please	jack	oliver	success	albert

За матеріалами: Perfect Passwords, Mark Burnett.

Навчально-методичне видання

Супруненко Оксана Олександрівна
Блакова Ольга Анатолівна
Сиволовський Ігор Михайлович

МЕТОДИЧНІ ВКАЗВІКИ ТА ІНДИВІДУАЛЬНІ ЗАВДАННЯ
ДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ
з дисципліни "Безпека програм та даних"

для студентів напрямів підготовки

050103 – Програмна інженерія,
050101 – Комп'ютерні науки,
040303 – Системний аналіз

усіх форм навчання

Комп'ютерне верстання:
Л.Г.Любченко

Підписано до друку 31.05.2013. Формат 60×84/16
Ум. друк. арк. 2,8. Тираж 100 прим. Зам. № 4675.

Видавець і виготовник
Черкаський національний університет імені Богдана Хмельницького.
Адреса: 18000, м. Черкаси, бул. Шевченка, 81, кімн. 117,
тел. (0472) 37-13-16, факс (0472) 37-22-33,
e-mail: vydav@cdu.edu.ua, http://www.cdu.edu.ua
Свідоцтво про внесення до державного реєстру
суб'єкт видавничої справи ДК № 3427 від 17.03.2009 р.