Міністерство транспорту та зв'язку України Державний університет інформаційно-комунікаційних технологій

В.О. Хорошко, В.С. Чередниченко, М.Є. Шелест

ОСНОВИ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

За редакцією професора В.О. Хорошка

Київ ДУІКТ 2008

УДК 351.746:007(0758)
ББК 67.9(4Укр)401я7+32.81я7
Х 8 0

Друкується за рішенням вченої ради Навчально-наукового інституту захисту інформації ДУІКТ /протокол №6 від 21.01.2008року)

Рецензенти: доктор юридичних наук, професор Кузьмічов В.С., доктор політичних наук, професор Соснін О.В., доктор технічних наук, професор Козловський В.В.

Хорошко В.О., Чередниченко В.С., Шелест М.Є. Х 80 Основи інформаційної безпеки / За ред. проф. В.О. Хорошка. - К.: ДУІКТ, 2008.- 186 с.
ISBN 978-966-2970-16-6

У підручнику наводяться суть, підходи та особливості інформаційної безпеки. Розглядаються загрози та різні рівні інформаційної безпеки.

Підручник призначений для студентів напряму «Інформаційна безпека» та буде корисний для спеціалістів, які працюють у цій сфері.

УДК 351.746:007(0758)
ББК 67.9(4Укр)401я7+32.81я7

ISBN **978-966-2970-16-6**

Зміст

Перелік скорочень

ІБ - інформаційна безпека;
ІС - інформаційна система;

ПЗ - програмне забезпечення;
ПЗП - постійний запам'ятовуючий пристрій;
АБС - автоматизована банківська система;
ОС - операційна система;
ПЕМІН - побічні електромагнітні випромінювання і наведення;
СУБД- системи управління базами даних;
РКД - рольове керування доступом;
ПРД - правила розмежування доступу;
ЕЦП - електронний цифровий підпис;
МЕ - міжмережевий екран;
ПЗ ПШ - програмне забезпечення проміжного шару.

Вступ

Мета заходів в області інформаційної безпеки - захистити інтереси суб'єктів інформаційних відносин. Інтереси ці різноманітні, але всі вони концентруються навколо трьох основних аспектів:

* доступність;
* цілісність;
* конфіденційність.

Перший крок при побудові системи ІБ організації - ранжування й деталізація цих аспектів.

Важливість проблематики ІБ пояснюється двома основними причинами:

* цінністю накопичених інформаційних ресурсів;
* критичною залежністю від інформаційних технологій.

Руйнування важливої інформації, крадіжка конфіденційних даних, перерва у роботі внаслідок відмови - все це виливається у великі матеріальні втрати, завдає шкоди репутації організації. Проблеми з системами керуванняабо медичними системами загрожують здоров'ю й життю людей.

Сучасні інформаційні системи складні й, виходить, небезпечні вже саміпо собі, навіть без обліку активності зловмисників. Постійно виявляються новіуразливі місця в програмному забезпеченні. Доводиться брати до увагинадзвичайно широкий спектр апаратного й програмного забезпечення, численнізв'язки між компонентами.

Змінюються принципи побудови корпоративних ІС. Використовуютьсячисленні зовнішні інформаційні сервіси; надаються зовні власні; отрималоширокого поширення явище, позначене російським словом "аутсорсинг", количастина функцій корпоративної ІС передається зовнішнім організаціям.Розвивається програмування з активними агентами.

Підтвердженням складності проблематики ІБ є паралельний (і доситьшвидкий) ріст витрат на захисні заходи й кількості порушень ІБ у сполученні зростом середнього збитку від кожного порушення. (Остання обставина - щеодин довід на користь важливості ІБ.)

Успіх в області інформаційної безпеки може принести тількикомплексний підхід, що уособлює в собі заходи чотирьох рівнів:

* законодавчого;
* адміністративного;
* процедурного;
* програмно-технічного.

Проблема ІБ - не тільки (і не стільки) технічна; без законодавчої бази, безпостійної уваги керівництва організації й виділення необхідних ресурсів, беззаходів керування персоналом і фізичного захисту вирішити її неможливо.Комплексність також ускладнює проблематику ІБ; необхідна взаємодіяфахівців з різних галузей.

Як основний інструмент боротьби зі складністю пропонується об'єктно-орієнтований підхід. Інкапсуляція, успадкування, поліморфізм, виділенняграней

об'єктів, варіювання рівня деталізації - все це універсальні поняття,знання яких необхідно всім фахівцям з інформаційної безпеки.

Законодавчий рівень є найважливішим для забезпечення інформаційноїбезпеки. Необхідно всіляко підкреслювати важливість проблеми ІБ;сконцентрувати ресурси на найважливіших напрямках досліджень;скоординувати освітню діяльність; створити й підтримувати негативневідношення до порушників ІБ - все це функції законодавчого рівня.

На законодавчому рівні особливої уваги заслуговують правові акти йстандарти.

Головне завдання засобів адміністративного рівня - сформувати програмуробіт в області інформаційної безпеки й забезпечити її виконання, виділяючинеобхідні ресурси й контролюючи стан справ.

Основою програми є політика безпеки, що відбиває підхід організації дозахисту своїх інформаційних активів.

Розробка політики й програми безпеки починається з аналізу ризиків,першим етапом якого, у свою чергу, є ознайомлення з найпоширенішимизагрозами.

Головні загрози - внутрішня складність 1С, ненавмисні помилки штатнихкористувачів, операторів, системних адміністраторів й інших осіб, щообслуговують інформаційні системи.

На другому місці по розміру збитку стоять крадіжки й підробки.

Реальну небезпеку представляють пожежі й інші аварії підтримуючоїінфраструктури.

У загальному числі порушень росте частка зовнішніх атак, але основнийзбиток як і раніше наносять "свої".

Для переважної більшості організацій досить загального знайомства зризиками; орієнтація на типові, апробовані рішення дозволить забезпечитибазовий рівень безпеки при мінімальних інтелектуальних і поміркованихматеріальних витратах.

Розробка програми й політики безпеки може бути прикладомвикористання поняття рівня деталізації. Вони повинні підрозділятися на кількарівнів, що трактують питання різного ступеня специфічності. Важливимелементом програми є розробка й підтримка в актуальному стані карти ІС.

Безпеку неможливо додати до системи; її потрібно закладати із самогопочатку й підтримувати до кінця.

Заходи процедурного рівня орієнтовані на людей (а не на технічні засоби)і підрозділяються на наступні види:

* керування персоналом;
* фізичний захист;
* підтримка працездатності;
* реагування на порушення режиму безпеки;
* планування відновлювальних робіт.

На цьому рівні застосовуються важливі принципи безпеки:

* безперервність захисту в просторі й часі;
* поділ обов'язків;

* мінімізація привілеїв.

Тут також застосовуються об'єктний підхід і поняття життєвого циклу.Перший дозволяє розділити контрольовані сутності (територію, апаратуру йт.д.) на відносно незалежні підоб'єкти, розглядаючи їх з різним ступенемдеталізації й контролюючи зв'язок між ними.

Поняття життєвого циклу корисно застосовувати не тільки доінформаційних систем, але й до співробітників. На етапі ініціації повинен бутирозроблений опис посади з вимогами до кваліфікації й виділених комп'ютернихпривілеїв; на етапі установки необхідно провести навчання, у тому числі зпитань безпеки; на етапі виведення з експлуатації варто діяти акуратно, недопускаючи завдання збитків скривдженими співробітниками.

Інформаційна безпека багато в чому залежить від акуратного веденняпоточної роботи, що включає:

* підтримку користувачів;
* підтримку програмного забезпечення;
* конфігураційне керування;
* резервне копіювання;
* керування носіями;
* документування;
* регламентні роботи.

Елементом повсякденної діяльності є відстеження інформації в областіІБ; як мінімум, адміністратор безпеки повинен підписатися на списокрозсилання по нових пробілах у захисті (і вчасно ознайомлюватися з вхіднимиповідомленнями).

Потрібно, однак, заздалегідь готуватися до неординарних подій, тобто допорушень ІБ. Заздалегідь продумана реакція на порушення режиму безпекипереслідує три головні цілі:

* локалізація інциденту й зменшення нанесеної шкоди;
* виявлення порушника;
* попередження повторних порушень.

Виявлення порушника - процес складний, ***але*** перший і третій пунктиможна й потрібно ретельно продумати й відпрацювати.

У випадку серйозних аварій необхідне проведення відбудовних робіт.Процес планування таких робіт можна розділити на наступні етапи:

* виявлення критично важливих функцій організації, установленняпріоритетів;
* ідентифікація ресурсів, необхідних для виконання критично важливихфункцій;
* визначення переліку можливих аварій;
* розробка стратегії відбудовних робіт;
* підготовка до реалізації обраної стратегії;

* перевірка стратегії.

Програмно-технічні заходи, тобто заходи, спрямовані на контролькомп'ютерних сутностей - устаткування, програм й/або даних, утворюютьостанній і найважливіший рубіж інформаційної безпеки.

На цьому рубежі стають очевидними не тільки позитивні, але й негативнінаслідки швидкого прогресу інформаційних технологій. По-перше, додатковіможливості з'являються не тільки у фахівців з ІБ, але й у зловмисників. По-друге, інформаційні системи увесь час модернізуються, перебудовуються, доних додаються недостатньо перевірені компоненти (у першу чергу програмні),що утрудняє дотримання режиму безпеки.

Заходи безпеки доцільно розділити на наступні види:

* превентивні, перешкоджаючі порушенням ІБ;
* заходи виявлення порушень;
* локалізуючі, звужуючі зону впливу порушень;
* заходи щодо виявлення порушника;
* заходи відновлення режиму безпеки.

У продуманій архітектурі безпеки всі вони повинні мати місце.Із практичної точки зору важливими також є наступні принципиархітектурної безпеки:

* безперервність захисту в просторі й часі, неможливість оминути захиснізасоби;
* наслідування визнаним стандартам, використання апробованих рішень;
* ієрархічна організація ІС із невеликою кількістю сутностей на кожномурівні;
* посилення найслабшої ланки;
* неможливість переходу в небезпечний стан;
* мінімізація привілеїв:
* поділ обов'язків:
* ешелонованість оборони;
* розмаїтність захисних засобів;
* простота й керованість інформаційної системи.

Центральним для програмно-технічного рівня є поняття сервісубезпеки. До числа таких сервісів входять:• ідентифікація й аутентифікація;

* керування доступом;
* протоколювання й аудит;
* шифрування;
* контроль цілісності;
* екранування;
* аналіз захищеності;
* забезпечення відмовостійкості;
* забезпечення безпечного відновлення;
* тунелювання;
* керування.

Ці сервіси повинні функціонувати у відкритому мережевому середовищі зрізнорідними компонентами, тобто бути стійкими до відповідних загроз, а

їхнєзастосування повинне бути зручним для користувачів й адміністраторів.Наприклад, сучасні засоби ідентифікації/аутентифікації повинні бути стійкимидо пасивного й активного прослуховування мережі й підтримувати концепціюєдиного входу в мережу.

Виділимо найважливіші моменти для кожного з перерахованих сервісівбезпеки:

1. Кращими є криптографічні методи аутентифікації, реалізованіпрограмним або апаратно-програмним способом. Парольний захист ставанахронізмом, біометричні методи мають потребу в подальшій перевірці вмережевому середовищі.
2. В умовах, коли поняття довіреного програмного забезпеченнявідходить у минуле, стає анахронізмом і найпоширеніша - похідна(дискреційна) - модель керування доступом. У її термінах неможливо навітьпояснити, що таке "троянська" програма. В ідеалі при розмежуванні доступуповинна враховуватися семантика операцій, але поки для цього є тількитеоретична база. Ще один важливий момент - простота адміністрування вумовах великої кількості користувачів, і ресурсів, і безперервних змінконфігурації. Тут може допомогти рольове керування. Протоколювання й аудитповинні бути всепроникаючими й багаторівневими, з фільтрацією даних припереході на більш високий рівень. Це необхідна умова керованості. Бажанезастосування засобів активного аудиту, однак потрібно усвідомлюватиобмеженість їхніх можливостей і розглядати ці засоби як один з рубежівешелонованої оборони, причому не найнадійніший. Варто конфігурувати їхтаким чином, щоб мінімізувати кількість фіктивних тривог і не робитинебезпечних дій при автоматичному реагуванні.

Усе, що пов'язано до криптографією, складно не стільки з технічної,скільки з юридичної точки зору; для шифрування це складніш у декілька разів.Даний сервіс є інфраструктурним, його реалізація повинна мати місце на всіхапаратно-програмних платформах і задовольняти жорстким вимогам, якістосуються не тільки безпеки, але й продуктивності. Поки ж єдиним доступнимвиходом є застосування вільно розповсюджуваного ПЗ.

Надійний контроль цілісності також базується на криптографічнихметодах з аналогічними проблемами й методами їхнього вирішення. Можливо,прийняття Закону про електронний цифровий підпис змінить ситуацію накраще, буде розширений спектр реалізацій. На щастя, до статичної цілісності єй не криптографічні підходи, засновані на використанні запам'ятовуючихпристроїв, дані на які доступні тільки для читання. Якщо в системі розділитистатичну й динамічну складові й помістити першу в ПЗП або на компакт-диск,можна в зародку придушити загрози цілісності. Розумно, наприклад,записувати реєстраційну інформацію на пристрої з одноразовим записом: тодізловмисник не зможе "приховати сліди".

Аналіз захищеності - це інструмент підтримки безпеки життєвого циклу.Схожість його з активним аудитом - еврестичність, необхідність практичнобезперервного відновлення бази знань і роль не найнадійнішого, аленеобхідного захисного рубежу, на якому можна розташувати вільнорозповсюджуваний продукт.

Місія забезпечення інформаційної безпеки складна, у багатьох випадкахнездійсненна, проте завжди шляхетна.

Автори висловлюють щиру подяку доктору юридичних наук, професоруКузьмічову Володимиру Сергійовичу «Київський національний університетвнутрішніх справ», доктору політичних наук, професору Сосніну ОлександруВасильовичу «дипломатична академія України при Міністерстві закордоннихсправ України» та доктору технічних наук, професору Козловському ВалеріюВікторовичу «інститут спеціального зв'язку та захисту інформації Національноготехнічного університету «КПІ» за уважне та доброзичливе рецензування тазауваження, яке сприяло значному поглибленню та покращенню підручника.

Крім того, автори висловлюють подяку за співробітництво співробітникамСлужби безпеки України та Державної служби спеціального зв'язку та захистуінформації.

З огляду на те, що у підручнику вперше систематизовано викладаютьсяпитання інформаційної безпеки, він не може бути без недоліків, тому авторибудуть щиро вдячні за висловлені зауваження та пропозиції щодо покращенняйого.

Розділ **1**

Поняття інформаційної безпеки. Основні складові. Важливість проблеми

1.1. Поняття інформаційної безпеки

Словосполучення "інформаційна безпека" у різних контекстах може матирізне значення.

У даному курсі наша увага буде зосереджена на зберіганні, обробці тапередачі інформації незалежно від того, якою мовою (російською чи якою-небудь іншою) вона закодована, хто або що є її джерелом та який психологічнийвплив вона має на людей. Тому термін "інформаційна безпека" використовуєтьсяу вузькому змісті, так, як це прийнято, наприклад, в англомовній літературі.

Під інформаційною безпекою ми будемо розуміти захищеністьінформації й інфраструктурою, яка її підтримує від випадкових або навмиснихвпливів природного або штучного характеру, які можуть завдати неприйнятноїшкоди суб'єктам інформаційних відносин, у тому числі власникам ікористувачам інформації й підтримуючої інфраструктури. (Далі ми пояснимо,що варто розуміти під підтримуючою інфраструктурою.)

Захист інформації - це комплекс заходів, спрямованих на забезпеченняінформаційної безпеки.

Таким чином, правильний з методологічної точки зору підхід до проблемінформаційної безпеки починається з виявлення суб'єктів інформаційнихвідносин й інтересів цих суб'єктів, пов'язаних з використанням інформаційнихсистем (ІС). Загрози інформаційної безпеки - це зворотний бік використанняінформаційних технологій.

Із цього положення можна вивести два важливі висновки:

1 Трактування проблем, пов'язаних з інформаційною безпекою, длярізних категорій суб'єктів може істотно розрізнятися. Для ілюстрації доситьзіставити режимні державні організації й навчальні інститути. У першомувипадку "нехай краще все зламається, ніж ворог довідається хоч одинсекретний біт", у другому - "так немає в нас ніяких секретів, аби тільки всепрацювало".

2 Інформаційна безпека не зводиться винятково до захисту віднесанкціонованого доступу до інформації, це принципово більш широкепоняття. Суб'єкт інформаційних відносин може постраждати (зазнати збитківй/або одержати моральний збиток) не тільки від несанкціонованого доступу,але й від поломки системи, що викликала перерву в роботі. Більше того, длябагатьох відкритих організацій (наприклад, навчальних) власне захист віднесанкціонованого доступу до інформації стоїть по важливості аж ніяк не напершому місці.

Повертаючись до питань термінології, відзначимо, що термін"комп'ютерна безпека" (як еквівалент або замінник ІБ) видається нам занадтовузьким. Комп'ютери - тільки одна зі складових інформаційних систем, і хочанаша увага буде зосереджена в першу чергу на інформації, що зберігається,обробляється й передається за допомогою комп'ютерів, її безпека визначаєтьсявсією сукупністю складових й, у першу чергу, найслабшою ланкою, якою впереважній більшості випадків є людина, яка (написала, наприклад, свій парольна "гірчичнику", наклеєному на монітор).

Відповідно до визначення інформаційної безпеки, вона залежить не тількивід комп'ютерів, але й від інфраструктури, яка її підтримує, до якої можнавіднести системи електро-, водо- і теплопостачання, кондиціонери, засобикомунікацій і, звичайно, обслуговуючий персонал. Ця інфраструктура маєсамостійну цінність, але нас цікавити не лише те, як вона впливає на виконанняінформаційною системою запропонованих їй функцій.

Звернемо увагу, що у визначенні ІБ перед іменником "збиток" стоїтьприкметник "неприйнятний". Вочевидь, застрахуватися від усіх видів збитківнеможливо, тим більше неможливо зробити це економічно доцільним чином, коливартість захисних засобів і заходів не перевищує розмір очікуваного збитку.Виходить, із чимось доводиться миритися й захищатися потрібно тільки від того, зчим змиритися ніяк не можна. Іноді таким неприпустимим збитком є нанесенняшкоди здоров'ю людей або стану навколишнього середовища, але частіше порігнеприйнятності має матеріальне (грошове) вираження, а метою захистуінформації стає зменшення розмірів збитків до припустимих значень.

1.2. Основні складові інформаційної безпеки

Інформаційна безпека - багатогранна, можна навіть сказати, багатомірнаобласть діяльності, у якій успіх може принести лише систематичний,комплексний підхід.

Спектр інтересів суб'єктів, пов'язаних з використанням інформаційнихсистем, можна розділити на наступні категорії: забезпечення доступності,цілісності й конфіденційності інформаційних ресурсів і підтримуючоїінфраструктури.

Іноді в сукупність основних складових ІБ включають захист віднесанкціонованого копіювання інформації, але, на наш погляд, це занадтоспецифічний аспект із сумнівними шансами на успіх, тому ми не будемо йоговиділяти.

Пояснимо поняття доступності, цілісності й конфіденційності.

Доступність - це можливість за прийнятний час одержати необхіднуінформаційну послугу.

Під цілісністю мається на увазі актуальність і несуперечність інформації, їїзахищеність від руйнування й несанкціонованої зміни.

Нарешті, конфіденційність - це захист від несанкціонованого доступу доінформації.

Інформаційні системи створюються (купуються) для одержання певнихінформаційних послуг. Якщо з тих або інших причин надати ці послугикористувачам стає неможливо, це, мабуть, завдає шкоди всім

суб'єктамінформаційних відносин. Тому, не протиставляючи доступність іншимаспектам, ми виділяємо її як найважливіший елемент інформаційної безпеки.

Особливо яскраво провідна роль доступності виявляється в різного родусистемах керування - виробництвом, транспортом і т.п. Зовні менш драматичні, але також досить неприємні наслідки - і матеріальні, і моральні -може бути тривала недоступність інформаційних послуг, якими користуєтьсявелика кількість людей (продаж залізничних та авіаквитків, банківськіпослуги й т.п.).

Цілісність можна підрозділити на статичну (що розуміється як незмінністьінформаційних об'єктів) і динамічну (яка стосується коректного виконанняскладних дій (транзакцій)). Засоби контролю динамічної цілісностізастосовуються, зокрема, при аналізі потоку фінансових повідомлень із метоювиявлення крадіжки, перевпорядкування або дублювання окремих повідомлень.

Цілісність виявляється є найважливішим аспектом ІБ у тих випадках, колиінформація слугує "керівництвом до дії". Рецептура ліків, запропонованімедичні процедури, набір і характеристики комплектуючих виробів, хідтехнологічного процесу - все це приклади інформації, порушення цілісностіякої може виявитися в буквальному значенні смертельним. Неприємно йперекручування офіційної інформації, будь-то текст закону або сторінка Web-сервера якої-небудь урядової організації.

Конфіденційність - найпроблемніший у нас у країні аспект інформаційноїбезпеки. На жаль, практична реалізація засобів по забезпеченнюконфіденційності сучасних інформаційних систем натрапляє в Україні насерйозні труднощі. По-перше, відомості про технічні канали витоку інформації єзакритими, так що більшість користувачів позбавлені можливості скластиуявлення про потенційні ризики. По-друге, на шляху використовуваноїкриптографії як основного засобу забезпечення конфіденційності стоятьчисленні законодавчі перепони й технічні проблеми.

Якщо повернутися до аналізу інтересів різних категорій суб'єктівінформаційних відносин, то майже для всіх, хто реально використовує ІС, напершому місці стоїть доступність. Практично не поступається їй по важливостіцілісність - який сенс в інформаційній послузі, якщо вона містить перекрученівідомості?

Нарешті, конфіденційні моменти є також у багатьох організацій (навіть узгадуваних вище навчальних інститутах намагаються не розголошувативідомості про зарплату співробітників) і окремих користувачів (наприклад,паролі).

1.3. Важливість і складність проблеми інформаційної безпеки

Інформаційна безпека є одним з найважливіших аспектів інтегральноїбезпеки, на якому б рівні ми не розглядали останню - національному,галузевому, корпоративному або персональному.

Для ілюстрації цього положення обмежимося декількома прикладами.

* За розпорядженням президента США Клінтона (від 15 липня 1996 року,номер 13010) була створена Комісія із захисту критично важливоїінфраструктури як від фізичних нападів, так і від атак, початих за допомогоюінформаційної зброї.

На початку жовтня *1997* року при підготовці доповідіпрезидентові глава вищезгаданої комісії Роберт Марш заявив, що в цей час ніуряд, ні приватний сектор не мають у своєму розпорядженні засобів захистувід комп'ютерних атак, здатних вивести з ладу комунікаційні мережі й мережіенергопостачання.

* Американський ракетний крейсер "Йорктаун" був змушений повернутисяв порт через численні проблеми із програмним забезпеченням, щофункціонувало на платформі Windows NT 4.0 (Government ComputerNews, липень 1998). Таким виявився побічний ефект програми ВМФ США по максимально широкому використанню комерційного програмногозабезпечення з метою зниження вартості військової техніки.
* Заступник начальника керування по економічних злочинах Міністерствавнутрішніх справ Росії повідомив, що російські хакери з 1994 по 1996 ріквчинили майже 500 спроб проникнення в комп'ютерну мережуЦентрального банку Росії. В 1995 році ними було викрадено 250 мільярдіврублів (ІТАР-ТАРС, АР, 17 вересня 1996 року).
* Як повідомив журнал Internet Week від 23 березня 1998 року, втратинайбільших компаній, викликані комп'ютерними вторгненнями,продовжують збільшуватися, незважаючи на ріст витрат на засобизабезпечення безпеки. Відповідно до результатів спільного дослідження

  Інституту інформаційної безпеки й ФБР. в 1997 році збиток відкомп'ютерних злочинів досяг 136 мільйонів доларів, що на 36% більше,ніж в 1996 році. Кожен комп'ютерний злочин завдає шкоди приблизно в200 тисяч доларів.
* У середині липня 1996 року корпорація General Motors відкликала 292860автомобілів марки Pontiac, Oldsmobile й Buick моделей 1996 й 1997 років,оскільки помилка в програмному забезпеченні двигуна могла спричинитипожежу.
* У лютому 2001 року двоє колишніх співробітників компанії CommerceOne, скориставшись паролем адміністратора, видалили із серверафайли, що становили великий (на кілька мільйонів доларів) проект дляіноземного замовника. На щастя, була резервна копія проекту, так щореальні втрати обмежилися витратами на наслідок і засоби захисту відподібних інцидентів у майбутньому. У серпні 2002 року злочинці посталиперед судом.
* Одна студентка втратила стипендію в 18 тисяч доларів у Мічиганськомууніверситеті через те, що її сусідка по кімнаті скористалася їх спільнимсистемним паролем і відправила від імені своєї жертви електронний лист ізвідмовою від стипендії.

Зрозуміло, що подібних прикладів безліч, можна згадати й інші випадки - недоліків в порушеннях ІБ немає й не передбачається. Чого коштує одна тільки"Проблема 2000" - сором і ганьба програмного співтовариства!

При аналізі проблематики, пов'язаної з інформаційною безпекою, необхідновраховувати специфіку даного аспекту безпеки, що полягає у тому, щоінформаційна безпека є складовою частиною інформаційних технологій -області,

що розвивається надзвичайно високими темпами. Тут важливі нестільки окремі рішення (закони, навчальні курси, програмно-технічні засоби),що перебувають на сучасному рівні, скільки механізми генерації нових рішень,що дозволяють жити в темпі технічного прогресу.

На жаль, сучасна технологія програмування не дозволяє створюватибезпомилкові програми, що не сприяє швидкому розвитку засобів забезпеченняІБ. Варто виходити з того, що необхідно конструювати надійні системи(інформаційної безпеки) із залученням ненадійних компонентів (програм). Упринципі, це можливо, але вимагає дотримання певних архітектурнихпринципів і контролю стану захищеності протягом усього життєвого циклу ІС.

Наведемо ще кілька цифр. У березні 1999 року був опублікований черговий,четвертий по рахунку, річний звіт "Комп'ютерна злочинність і безпека-1999:проблеми й тенденції" (Issues and Trends: 1999 CS1/FB1 Computer Crime and SecuritySurvey). У звіті відзначається різкий ріст числа звернень до правоохоронних органівіз приводу комп'ютерних злочинів (32% із числа опитаних); 30% респондентівповідомили про те, що їхні інформаційні системи були зламані зовнішнімизловмисниками; атакам через Internet піддавалися 57% опитаних; у 55% випадкахвідзначалися порушення з боку власних співробітників. Примітно, що 33%респондентів на питання " чи були зламані ваші Web-сервери й системи електронноїкомерції за останні 12 місяців? " відповіли "не знаю".

В аналогічному звіті, опублікованому у квітні 2002 року, цифри змінилися,але тенденція залишилася колишнюю: 90% респондентів (переважно з великихкомпаній й урядових структур) повідомили, що за останні 12 місяців у їхніхорганізаціях мали місце порушення інформаційної безпеки; 80% констатувалифінансові втрати від цих порушень; 44% (223 респондента) змогли й/або захотілиоцінити втрати кількісно, загальна сума склала більше 455 млн. доларів.Найбільший збиток нанесли крадіжки й підробки (більше 170 й 115 млн. доларіввідповідно).

Настільки ж тривожні результати втримуються в огляді information Week,опублікованому 12 липня 1999 року. Лише 22% респондентів заявили провідсутність порушень інформаційної безпеки. Поряд з поширенням вірусіввідзначається різкий ріст числа зовнішніх атак.

Збільшення кількості атак - ще не найбільша неприємність. Гірше те, щопостійно виявляються нові уразливі місця в програмному забезпеченні (вище мивказувати на обмеженість сучасної технології програмування) і, як наслідок,з'являються нові види атак.

Так, в інформаційному листі Національного центру захисту інфраструктуриСША (National Infrastructure Protection Center, NIPC) від 21 липня 1999 рокуповідомляється, що за період з 3 по 16 липня 1999 року виявлено дев'ять проблемз ПЗ, ризик використання яких оцінюється як середній або високий (загальнечисло виявлених уразливих місць дорівнює 17). Серед "потерпілих" операційнихплатформ - майже всі різновиди ОС Unix, Windows, MacOS, так що ніхто не можепочуватися спокійно, оскільки нові помилки відразу починають активновикористовуватися зловмисниками.

У таких умовах системи інформаційної безпеки повинні вміти протистоятирізноманітним атакам, як зовнішнім, так і внутрішнім, атакам автоматизованим іскоординованим. Іноді напад триває долі секунди; часом промацуванняуразливих місць ведеться повільно й розтягується на години, так що підозрілаактивність практично непомітна. Метою зловмисників може бути порушення всіхскладових ІБ - доступності, цілісності або конфіденційності.

Розділ 2. Поширення об'єктно-орієнтованого підходуна інформаційну безпеку

2.1. Про необхідність об'єктно-орієнтованого підходудо інформаційної безпеки

У наш час інформаційна безпека є відносно замкнутою дисципліною,розвиток якої не завжди синхронізований зі змінами в інших областяхінформаційних технологій. Зокрема, в ІБ поки не знайшли відображення основніположення об'єктно-орієнтованого підходу, що став основою при побудовісучасних інформаційних систем. Не враховуються в ІБ і досягнення в технологіїпрограмування, засновані на нагромадженні й багаторазовому використанніпрограмістських знань. На наш погляд, це дуже серйозна проблема, щоускладнює прогрес в області ІБ.

Спроби створення більших систем ше в 60-х роках розкрили численніпроблеми програмування, головною з яких є складність створюваних ісупроводжуваних систем. Результатами досліджень в області технологіїпрограмування стали спочатку структуроване програмування, потім об'єктно-орієнтований підхід.

Об'єктно-орієнтований підхід є основою сучасної технологіїпрограмування, випробуваним методом боротьби зі складністю систем. Єприродним й, більше того, необхідним, прагнення поширити цей підхід і насистеми інформаційної безпеки, для яких, як і для програмування в цілому, маємісце згадана проблема складності.

Складність ця має двояку природу. По-перше, складні не тільки апаратно-програмні системи, які необхідно захищати, але й самі засоби безпеки. По-друге,швидко наростає складність сімейства нормативних документів, таких,наприклад, як профілі захисту на основі "Загальних критеріїв", мова про які піседалі. Ця складність менш очевидна, але нею також не можна нехтувати; необхідноспоконвічно будувати сімейства документів по об'єктному принципу.

Будь-який розумний метод боротьби зі складністю базується на принципі"devide et impera" - "розділяй і пануй". У даному контексті цей принцип означає,що складна система (інформаційної безпеки) на верхньому рівні повиннаскладатися з невеликої кількості відносно незалежних компонентів. Відноснанезалежність тут і далі розуміється як мінімізація кількості зв'язків міжкомпонентами. Потім декомпозиції піддаються виділенню на першому етапікомпонента, і так далі до заданого рівня деталізації. У результаті системавідображається у вигляді ієрархії з декількома рівнями абстракції.

Найважливіше питання, що виникає при реалізації принципу "розділяй іпануй", - як, власне кажучи, розділяти. Згадуваний вище структурний підхідопирається на алгоритмічну декомпозицію, коли виділяються функціональніелементи системи.

Основна проблема структурного підходу полягає в тому, щовін незастосовний на ранніх етапах аналізу й моделювання предметної області,коли до алгоритмів і функцій справа ще не дійшла. Потрібний підхід "широкогоспектру", що не має такого концептуального розриву з аналізованими системами йзастосований на всіх етапах розробки й реалізації складних систем. Минамагатимемося показати, що об'єктно-орієнтований підхід задовольняє такимвимогам.

2.2. Основні поняття об'єктно-орієнтованого підходу

Об'єктно-орієнтований підхід використовує об'єктну декомпозицію, тобтоповодження системи описується в термінах взаємодії об'єктів.

Що ж розуміється під об'єктом та які інші основні поняття даногопідходу?

Насамперед, уведемо поняття класу. Клас - це абстракція безлічісутностей реального світу, об'єднаних спільністю структури й поводження.

Об'єкт - це елемент класу, тобто абстракція певної сутності.

Підкреслимо, що об'єкти активні, у них є не тільки внутрішня структура,але й поведінка, що описується так званими методами об'єкта. Наприклад, можебути визначений клас "користувач", що характеризує "користувача взагалі",тобто асоційовані з користувачами дані і їхня поведінка (методи). Після цьогоможе бути створений об'єкт "користувач Іванов" з відповідною конкретизацієюданих й, можливо, методів.

До активності об'єктів ми ще повернемося.

Наступну групу найважливіших понять об'єктного підходу складаютьінкапсуляція, успадкування й поліморфізм.

Основним інструментом боротьби зі складністю в об'єктно-орієнтованомупідході є інкапсуляція - приховування реалізації об'єктів (їхньої внутрішньоїструктури й деталей) реалізації методів) з наданням зовні чітко вираженихінтерфейсів.

Поняття "поліморфізм" може трактуватися як здатність об'єкта належатибільш ніж одному класу. Введення цього поняття відображає необхідністьдивитися на об'єкти під різними кутами зору, виділяти при побудові абстракційрізні аспекти сутностей модельованої предметної області, не порушуючи прицьому цілісності об'єкта. (Суворо кажучи, існують й інші види поліморфізму,такі як перевантаження й параметричний поліморфізм, але нас вони зараз нецікавлять.)

Успадкування означає побудову нових класів на основі існуючих зможливістю додавання або перевизначення даних і методів. Успадкування єважливим інструментом боротьби з розмноженням сутностей без необхідності.Загальна інформація не дублюється, указується тільки те, що здійснюється. Прицьому клас-нащадок пам'ятає про своє "коріння".

Дуже важливо й те, що успадкування й поліморфізм у сукупностінаділяють об'єктно-орієнтовану систему здатністю до відносно безболісноїеволюції. Засоби інформаційної безпеки доводиться постійно модифікувати йобновляти, і якщо не можна зробити так, щоб це було економічно вигідно, ІБ зінструмента захисту перетворюється на тягар.

Ми ще повернемося до механізму успадкування при розгляді рольовогокерування доступом.

Поповнимо розглянутий вище класичний набір понять об'єктно-орієнтованого підходу ще двома поняттями: межі об'єкта й рівня деталізації.

Об'єкти реального світу володіють, як правило, декількома відноснонезалежними характеристиками. Стосовно об'єктної моделі будемо називатитакі характеристики межами. Ми вже зустрічалися із трьома основнимимежами ІБ - доступністю, цілісністю й конфіденційністю. Поняття гранідозволяє більш природно, ніж поліморфізм, дивитися на об'єкти з різних точокзору й будувати різнопланові абстракції.

Поняття рівня деталізації важливе не тільки для візуалізації об'єктів, але йдля систематичного розгляду складних систем, представлених в ієрархічномувигляді. Саме по собі воно дуже просте: якщо черговий рівень ієрархіїрозглядається з рівнем деталізації n > 0, то наступний - з рівнем (n - 1). Об'єктіз рівнем деталізації 0 уважається атомарним.

Поняття рівня деталізації показу дозволяє розглядати ієрархії зпотенційно нескінченною висотою, варіювати деталізацію як об'єктів у цілому,так й їхніх меж.

Досить розповсюдженою конкретизацією об'єктно-орієнтованого підходує компонентні об'єктні середовища, до числа яких належить, наприклад,JavaBeans. Тут з'являється два нових важливих поняття: компонентів іконтейнерів.

Неформально компонент можна визначити як багаторазово використовуваний об'єкт, що допускає обробку в графічномуінструментальному оточенні й збереження в довгостроковій пам'яті.

Контейнери можуть містити в собі безліч компонентів, створюючи загальний контекст взаємодії з іншими компонентами й з оточенням.Контейнери можуть виступати в ролі компонентів інших контейнерів.

Компонентні об'єктні середовища мають всі переваги, які властивіоб'єктно-орієнтованому підходу:

* інкапсуляція об'єктних компонентів приховує складність реалізації, роблячивидимим тільки представлений зовні інтерфейс;
* успадкування дозволяє розвивати створені раніше компоненти, непорушуючи цілісність об'єктної оболонки;
* поліморфізм по суті дає можливість групувати об'єкти, характеристики якихз деякого погляду можна вважати подібними.

Поняття ж компонента й контейнера необхідні нам тому, що з їхньою допомогою ми можемо природно представити захищувані ІС і самі захиснізасоби. Зокрема, контейнер може визначати кордони контрольованої зони(задавати так званий "периметр безпеки").

На цьому ми завершуємо опис основних понять об'єктно-орієнтованогопідходу.

2.3. Застосування об'єктно-орієнтованого підходудо розгляду систем, що захищають

Спробуємо застосувати об'єктно-орієнтований підхід до питань інформаційної безпеки.

Проблема забезпечення інформаційної безпеки - комплексна, захищати доводиться складні системи, і самі захисні засоби теж складні, тому нам знадобляться всі уведені поняття. Почнемо з поняття межі.

Фактично три межі вже були уведені: це доступність, цілісність і конфіденційність. Їх можна розглядати відносно незалежно, і вважається, що якщо всі вони забезпечені, то забезпечена й ІБ у цілому (тобто суб'єктам інформаційних відносин не буде нанесений неприйнятний збиток).

Таким чином, ми структурували нашу мету. Тепер потрібно структурувати засоби її досягнення. Введемо наступні межі:

* законодавчі заходи забезпечення інформаційної безпеки;
* адміністративні заходи (накази й інші дії керівництва організацій,пов'язаних з інформаційними системами, що захищають);
* процедурні заходи (заходи безпеки, орієнтовані на людей);
* програмно-технічні заходи.

У подальшій частині курсу ми пояснимо докладніше, що розуміється під кожною з виділених меж. Тут же відзначимо, що, у принципі, їх можна розглядати і як результат варіювання рівня деталізації (із цієї причини ми буде мовживати словосполучення "законодавчий рівень", "процедурний рівень" і т.п.).
Закони й нормативні акти орієнтовані на всіх суб'єктів інформаційних відносин незалежно від їхньої організаційної приналежності (це можуть бути як юридичні, так і фізичні особи) у межах країни (міжнародні конвенції маютьнавіть більш широку область дії), адміністративні заходи - на всіх суб'єктів умежах організації, процедурні - на окремих людей (або невеликі категоріїсуб'єктів), програмно-технічні - на устаткування й програмне забезпечення. При такому трактуванні в переході з рівня на рівень можна побачити застосування успадкування (кожен наступний рівень не скасовує, а доповнює попередній), а також поліморфізму (суб'єкти виступають відразу в декількох іпостасях -наприклад, як ініціатори адміністративних заходів й як звичайні користувачі, зобов'язані цим заходам підкорятися).

Очевидно, для всіх виділених, щодо незалежних меж діє принцип інкапсуляції (це й виходить, що межі "відносно незалежні"). Більше того, ці дві сукупності меж можна назвати ортогональними, оскільки для фіксованої межі водній сукупності (наприклад, доступності) межі в іншій сукупності повинні пробігати всю кількість можливих значень (потрібно розглянути законодавчі, адміністративні, процедурні й програмно-технічні заходи). Ортогональних сукупностей не повинно бути багато; здасться, двох сукупностей із числомелементів, відповідно, 3 й 4 уже достатньо, тому що вони дають 12 комбінацій.

Продемонструємо тепер, як можна розглядати захищувану ІС, варіюючи рівень деталізації.

Нехай інтереси суб'єктів інформаційних відносин концентруються навколо ІС якоїсь організації, яка складається з двох територіально рознесених виробничих майданчиків, на кожному з яких є сервери, що обслуговують своїх і зовнішніх користувачів, а також користувачів, які користуються внутрішніми й зовнішніми

сервісами. Один з майданчиків обладнаний зовнішнім підключенням (тобто має вихід в Internet).

При погляді з нульовим рівнем деталізації ми побачимо лише те, що ворганізації є інформаційна система (див. Рис. 2.1).

ІС організації

Рис. 2.1. ІС при розгляді з рівнем деталізації 0.Подібна точка зору може здатися неспроможною, але це не так. Уже тут необхідно врахувати закони, які застосовуються до організацій, що упорядковують інформаційні системи. Можливо, яку-небудь інформацію не можна зберігати й обробляти на комп'ютерах, якщо ІС не була атестована на відповідність певним вимогам. На адміністративному рівні може бути задекларована мета, заради якої створювалася ІС. Загальні правила закупівель, впровадження нових компонентів,експлуатації й т.п. На процедурному рівні потрібно визначити вимога до фізичної безпеки ІС і шляхи їхнього виконання, правила протипожежної безпеки й т.п. На програмно-технічному рівні можуть бути визначені кращі апаратно-програмніплатформи й т.п.

За якими критеріями проводити декомпозицію ІС - у значній мірі справа смаку. Будемо вважати, що на першому рівні деталізації робляться прозорими сервіси й користувачі, точніше, поділ на клієнтську й серверну частину (Рис. 2.2).

ІС організації:
Сервіси (без конкретизації)
Користувачі (без конкретизації)

Рис. 2.2. ІС при розгляді з рівнем деталізації 1.

На цьому рівні варто сформулювати вимоги до сервісів (до самої їхньоїнаявності, до доступності, цілісності й конфіденційності надаваних інформаційних послуг), викласти способи виконання цих вимог, визначити загальні правила поведінки користувачів, необхідний рівень їхньої попередньої підготовки, методи контролю їхньої поведінки, порядок заохочення й покарання й т.п. Можуть бути сформульовані вимоги й переваги стосовно серверних і клієнтських платформ.

На другому рівні деталізації ми побачимо наступне (див. Рис. 2.3).

Internet
Сервіси, які використовуються організацією(без конкретизації)
Користувачі сервісів організації(без конкретизації)

ІС організації
Сервіси, що надаються(без конкретизації)
Внутрішні сервіси(без конкретизації)
Користувачі зовнішніх сервісів(без конкретизації)
Користувачі внутрішніх сервісів(без конкретизації)

Рис. 2.3. ІС при розгляді з рівнем деталізації 2

На цьому рівні нас усе ще не цікавить внутрішня структура ІС організації, так само як і деталі Internet. Констатується тільки існування зв'язку між цими мережами, наявність у них користувачів, а також надаваних і внутрішніх сервісів. Що це за сервіси, поки неважливо.

Перебуваючи на рівні деталізації 2, ми повинні враховувати закони, які застосовуються до організацій, ІС які забезпечені зовнішніми підключеннями. Мова йде про допустимість такого підключення, про його захист, про відповідальність користувачів, що звертаються до зовнішніх сервісів, і про відповідальність організацій, що відкривають свої сервіси для зовнішньогодоступу. Конкретизація аналогічної спрямованості, з урахуванням наявності зовнішнього підключення, повинна бути виконана на адміністративному, процедурному й програмно-технічному рівнях.

Звернемо увагу на те, що контейнер (у змісті компонентного об'єктного середовища) "ІС організації" задає межі контрольованої зони, у межах яких організація проводить певну політику. Internet існує за іншими правилами, якіорганізація повинна приймати, як щось уже існуюче.

Збільшуючи рівень деталізації, можна розглянути два рознесенених виробничих майданчика й канали зв'язку між ними, розподіл сервісів і користувачів по цих майданчиках і засоби забезпечення безпеки внутрішніх комунікацій, специфіку окремих сервісів, різні категорії користувачів і т.п. Ми,однак, на цьому зупинимося.

2.4. Недоліки традиційного підходу до інформаційної безпеки з об'єктної точки зору

Грунтуючись основними положеннями об'єктно-орієнтованого підходу, треба в першу чергу визнати застарілий традиційний розподіл на активні й пасивні сутності (суб'єкти й об'єкти у звичній для дооб'єктної ІБ термінології). Подібний розподіл застарілий, принаймні, з двох причин.

По-перше, в об'єктному підході пасивних об'єктів немає. Можна вважати, що всі об'єкти активні одночасно й при необхідності викликають методи один одного. Як реалізовані ці методи (і, зокрема, як організований доступ до змінних і їхній значень) - власна справа викликаного об'єкта; деталі реалізації приховані, інкапсульовані. Зухвалому об'єкту доступний тільки надаваний інтерфейс. По-друге, не можна сказати, що якісь програми (методи) виконуються від імені користувача. Реалізації об'єктів складні, так що останні не можна розглядатилише як інструменти виконання волі користувачів. Можна вважати, щокористувач так чи інакше, на свій страх і ризик, "просить" деякий об'єкт проконкретну інформаційну послугу. Коли активізується викликуваний метод, об'єктдіє скоріше від імені (у всякому разі, з волі) свого творця, ніж від імені його користувача, який його викликав. Можна вважати, що об'єкти мають достатню"силу волі", щоб виконувати дії, про які користувач не тільки не просив, але навіть не здогадується про їхні можливості. Особливо це справедливо в мережевому середовищі й для програмного

забезпечення (ПЗ), отриманого через Internet, але може буди придатним і для комерційного ПЗ, закупленого за всіма правилами всолідній фірмі.

Для ілюстрації наведемо наступний гіпотетичний приклад. Банк, ІС якого має доступ до Internet, придбав за кордоном автоматизовану банківську систему (АБС). Тільки через деякий час у банку вирішили, що зовнішнє з'єднання має потребу в захисті, і встановили міжмережевий екран.

Вивчення реєстраційної інформації екрана показало, що час від часу закордони відправляються IP-пакети, що містять якусь незрозумілу інформацію (напевно, закодовану, вирішили в банку). Почали розбиратися, кому ж пакети надходять, і виявилося, що йдуть вони у фірму, що розробила АБС. Виникла підозра, що в АБС вбудована закладка, щоб одержувати інформацію про діяльність банку. Зв'язалися з фірмою; там дуже здивувалися, спочатку всі заперечували, але зрештою з'ясували, що один із програмістів не забрав зівставленого в банку варіанта налагоджену відправку, що була організована через мережу (як передача IP-пакетів специфічного виду, з явно заданою IP-адресоюробочого місця цього програміста). Таким чином, ніякого злого наміру не було,однак якийсь час інформація про платежі вільно гуляла по мережі.

У подальшій частині курсу, у лекції, присвяченої розмежуванню доступу, ми обговоримо, як можна кардинальним чином вирішити подібні проблеми. Тут відзначимо лише, що при визначенні допустимості доступу важливо не тільки (і не стільки), хто звернувся до об'єкта, але й те, яка семантика дії. Без залучення семантики не можна визначити так звані "троянські програми", що виконують, крім задекларованих, деякі приховані (зазвичай негативні) дії.

Очевидно, варто визнати застарілим й положення про те, що розмежування доступу спрямоване на захист від зловмисників. Наведений вище приклад показує, що внутрішні помилки розподілених ІС несуть за собою не меншу небезпеку, а гарантувати їхню відсутність у складних системах сучасна технологія програмування не дозволяє.

У дооб'єктній ІБ однією з найважливіших вимог є безпека повторного використання пасивних сутностей (таких, наприклад, як динамічно виділені частини пам'яті). Очевидно, подібна вимога вступає в конфлікт із таким фундаментальним принципом, як інкапсуляція. Об'єкт не можна очистити зовнішнім способом (заповнити нулями або випадковою послідовністю біт), якщо тільки він сам не пропонує відповідний метод. При наявності такого методу надійність очищення залежить від коректності його реалізації й виклику.

Найміцнішим зі стереотипів серед фахівців з ІБ є трактування операційноїсистеми як домінуючої серед заходів безпеки. На розробку захищених ОС виділяються значні кошти, найчастіше на збиток іншим напрямкам захисту й, отже, на збиток реальної безпеки. У сучасних ІС, побудованих у багаторівневій архітектурі клієнт/сервер, ОС не контролює об'єкти, з якими працюють користувачі, так само як і дії самих користувачів, які реєструються й визначаються прикладними засобами. Основною функцією безпеки ОС стаєзахист можливостей, надаваних привілейованим користувачам, від атак користувачів звичайних.

Це важливо, але безпека такими заходами не вичерпується. Далі мирозглянемо підхід до побудови програмно-технічного рівня ІБ у вигляді сукупності сервісів безпеки.

Розділ 3. Найпоширеніші загрози

3.1. Основні визначення і критерії класифікації загроз

Загроза - це потенційна можливість певним чином порушити інформаційнубезпеку.

Спроба реалізації загрози називається атакою, а той, хто вчиняє таку спробу, - зловмисником. Потенційні зловмисники називаються джерелами загроз.

Найчастіше загроза є наслідком наявності уразливих місць у захистіінформаційних систем (таких, наприклад, як можливість доступу сторонніх осіб докритично важливого устаткування або помилки в програмному забезпеченні).

Проміжок часу від моменту, коли з'являється можливість використатислабке місце, і до моменту, коли прогалина ліквідується, називається вікномнебезпеки, асоційованим з даним уразливим місцем. Поки існує вікнонебезпеки, можливі успішні атаки на ІС.

Якщо мова йде про помилки в ПЗ, то вікно небезпеки "відкривається" зпоявою засобів використання помилки й ліквідується при накладенні латок, які Ївиправляють.

Для більшості уразливих місць вікно небезпеки існує порівняно довго (кількаднів, іноді - тижнів), оскільки за цей час повинні відбутися наступні події:

* повинно стати відомо про засоби використання прогалини в захисті;
* повинні бути випущені відповідні латки;
* латки повинні бути встановлені в захищуваній ІС.

Ми вже вказували, що нові уразливі місця й засоби їхнього використанняз'являються постійно; це значить, по-перше, що майже завжди існують вікнанебезпеки й, по-друге, що відстеження таких вікон повинне провадитися постійно,а випуск і накладення латок - якомога оперативніше.

Відзначимо, що деякі загрози не можна вважати наслідком якихось помилокабо прорахунків; вони існують у чинність самої природи сучасних ІС. Наприклад,загроза відключення електрики або виходу його параметрів за припустимі граніціснує внаслідок залежності апаратного забезпечення ІС від якісногоелектроживлення.

Розглянемо найпоширеніші загрози, яким піддаються сучасні інформаційнісистеми. Мати подання про можливі загрози, а також про уразливі місця, які цізагрози зазвичай експлуатують, необхідно для того, щоб вибирати найбільшекономічні засоби забезпечення безпеки. Занадто багато міфів існує в сферіінформаційних технологій (згадаємо все ту ж "Проблему 2000"), тому незнанняв цьому випадку веде до перевитрати коштів й, що ще гірше, до концентраціїресурсів там, де вони не дуже потрібні, за рахунок ослаблення дійсно уразливихнапрямків.

Підкреслимо, що саме поняття "загроза" у різних ситуаціях найчастішетрактується по-різному. Наприклад, для підкреслено відкритої організації загрозконфіденційності може просто не існувати - вся інформація вважаєтьсязагальнодоступною; однак у більшості випадків нелегальний доступ є серйозноюнебезпекою. Іншими словами, загрози, як і все в ІБ, залежать від інтересівсуб'єктів інформаційних відносин (і від того, який збиток є для нихнеприйнятним).

Ми спробуємо подивитися на предмет з погляду типової (на наш погляд)організації. Втім, багато загроз (наприклад, пожежа) небезпечні для всіх.Загрози можна класифікувати по декількох критеріях:

* по аспекту інформаційної безпеки (доступність, цілісність,конфіденційність), проти якого загрози спрямовані в першу чергу;
* по компонентах інформаційних систем, на які загрози націлені (дані,програми, апаратура, підтримуюча інфраструктура);
* по способу здійснення (випадкові/навмисні дії природного/техногенногохарактеру);
* розташуванню джерела загроз (усередині/поза розглянутим ІС).

Як основний критерій ми будемо використовувати перший (по аспекту ІБ),залучаючи при необхідності інші.

3.2. Найпоширеніші загрози доступності

Найчастішими й найнебезпечнішими (з погляду розміру збитку) єненавмисні помилки штатних користувачів, операторів, системнихадміністраторів й інших осіб, що обслуговують інформаційні системи.

Іноді такі помилки і є властиво загрозами (неправильно уведені дані абопомилка в програмі, що викликала крах системи), іноді вони створюють уразливімісця, якими

можуть скористатися зловмисники (такими є помилкиадміністрування). За деякими даними, до 65% втрат - наслідок ненавмиснихпомилок.

Пожежі й повені не приносять стільки лих, скільки безграмотність інедбалість у роботі.

Очевидно, найрадикальніший засіб боротьби з ненавмисними помилками - максимальна автоматизація й строгий контроль.

Інші загрози доступності класифікуємо по компонентах ІС, на які націленізагрози:

* відмова користувачів;
* внутрішня відмова інформаційної системи;
* відмова підтримуючої інфраструктури.

Звичайно стосовно користувачів розглядаються наступні загрози:

* небажання працювати з інформаційною системою (найчастішепроявляється при необхідності освоювати нові можливості й природзбіжності між запитами користувачів і фактичних можливостей ітехнічних характеристик);
* неможливість працювати із системою в наслідок відсутності відповідноїпідготовки (недолік загальної комп'ютерної грамотності, невмінняінтерпретувати діагностичні повідомлення, невміння працювати здокументацією й т.п.);
* неможливість працювати із системою в наслідок відсутності технічноїпідтримки (неповнота документації, недолік довідкової інформації й т.п.).

Основними джерелами внутрішніх відмов є:

• відступ (випадковий або навмисний) від установлених правилексплуатації;

* вихід системи зі штатного режиму експлуатації в наслідок випадкових абонавмисних дій користувачів або обслуговуючого персоналу (перевищеннярозрахункового числа запитів, надмірний обсяг оброблюваної інформації йтлі.);
* помилки при переконфігуруванні системи;
* відмови програмного й апаратного забезпечення;
* руйнування даних;
* руйнування або ушкодження апаратур.

Стосовно підтримуючої інфраструктури рекомендується розглядати наступнізагрози:

* порушення роботи (випадково або навмисне) систем зв'язку,електроживлення, водо- і/або теплопостачання, кондиціонування;
* руйнування або ушкодження приміщень;
* неможливість або небажання обслуговуючого персоналу й/абокористувачів виконувати свої обов'язки (цивільні безладдя, аварії натранспорті, терористичний акт або його загроза, страйк і т.п.).

Досить небезпечні так звані "скривджені" співробітники - нинішні й колишні.Як правило, вони прагнуть завдати шкоди, організації -"кривдникові",наприклад:

* зіпсувати устаткування;

* вмонтувати логічну бомбу, що згодом зруйнує програми й/або дані;

* видалити дані.

Скривджені співробітники, що були знайомі з порядками в організації й здатнізавдати чималої шкоди. Необхідно стежити за тим, щоб при звільненніспівробітника його права доступу (логічного і фізичного) до інформаційнихресурсів анулювалися.

Небезпечні, зрозуміло, стихійні лиха й події, які сприймаються як стихійнілиха: пожежі, повені, землетруси, урагани. По статистиці, на частку вогню, води йтому подібних "зловмисників" (серед яких найнебезпечніший - перебійелектроживлення) доводиться 13% втрат, нанесених інформаційним системам.

3.3. Деякі приклади загроз доступності

Загрози доступності можуть виглядати грубо - як ушкодження або навітьруйнування устаткування (у тому числі носіїв даних). Таке ушкодження можевикликатися природними причинами (найчастіше - грозами). На жаль, джерелабезперебійного живлення, що перебувають у масовому використанні незахищають від потужних короткочасних імпульсів, і випадки вигорянняустаткування - не рідкість.

У принципі, потужний короткочасний імпульс, здатний зруйнувати дані намагнітних носіях, можна згенерувати й штучним чином - за допомогою такзваних високоенергетичних радіочастотних гармат. Але, напевно, у нашихумовах подібну загрозу потрібко все-таки визнати надуманою.

Дійсно небезпечні - протікання водопроводу й опалювальної системи. Частоорганізації, щоб заощадити на орендній платі, знімають приміщення в будинкахстарої будівлі, роблять косметичний ремонт, але не міняють старі труби. Авторовікурсу довелось бути свідком ситуації, коли прорвало трубу з гарячою водою, ісистемний блок комп'ютера (це була робоча станція виробництва SunMicrosystems) виявився заповнений окропом. Коли окріп вилили, а комп'ютерпросушили, він відновив нормальну роботу, але краще таких дослідів непроводити...

Улітку, під час сильної спеки, намагаються зламатися кондиціонери, установленів серверних залах, набитих дорогим устаткуванням. У результаті значний збитокнаноситься й репутації, і гаманцю організації.

Загальновідомо, що періодично необхідно провадити резервне копіюванняданих. Однак навіть якщо ця пропозиція виконується, резервні носії найчастішезберігають недбало (до цього ми ще повернемося під час обговорення загрозконфіденційності), не забезпечуючи їхній захист від шкідливого впливу навколишнього середовища. І коли потрібно відновити дані, виявляється, що цісамі носії ніяк не бажають читатися.

Перейдемо тепер до загроз доступності, які будуть хитріші засміченьканалізації. Мова йтиме про програмні атаки на доступність.

В якості виводу системи зі штатного режиму експлуатації може використатисяагресивне споживання ресурсів (звичайно - пропускні шляхи мереж,обчислювальних можливостей процесорів або оперативної пам'яті). Порозташуванню джерела загрози таке споживання підрозділяється на локальне йвилучене. При прорахунках у конфігурації системи, локальна програма здатнапрактично монополізувати процесор й/або фізичну пам'ять, звівши швидкістьвиконання інших програм до нуля.

Найпростіший приклад вилученого споживання ресурсів - атака, що одержаланайменування "SYN-повінь". Вона являє собою спробу переповнити таблицю"напіввідчинених" TCP-з'єднань сервера (установлення з'єднань починається, алене закінчується). Така атака щонайменше ускладнює встановлення новихз'єднань з боку легальних користувачів, тобто сервер виглядає як недоступний.

Стосовно атаки "Papa Smurf' уразливі мережі, що сприймають ping-пакети ізширокомовними адресами. Відповіді на такі пакети "з'їдають" пропускні шляхи.

Вилучене споживання ресурсів останнім часом проявляється в особливонебезпечній формі - як скоординовані розподілені атаки, коли на сервер з безлічірізних адрес із максимальною швидкістю направляються цілком легальні запитина з'єднання й/або обслуговування. Часом початку "моди" на подібні атакиможна вважати лютий 2000 року, коли жертвами виявилися кілька найбільшихсистем електронної комерції (точніше - власники й користувачі систем).Відзначимо, що якщо має місце архітектурний прорахунок у виглядірозбалансованості між пропускною здатністю мережі й продуктивністю сервера,то захиститися від розподілених атак на доступність украй важко.

Для виведення систем зі штатного режиму експлуатації можутьвикористовуватися уразливі місця у вигляді програмних й апаратних помилок.Наприклад, відома помилка в процесорі Pentium I дає можливість локальномукористувачеві шляхом виконання певної команди "підвісити" комп'ютер, так щодопомагає лише апаратний RESET.

Програма "Teardrop" видалено "підвішує" комп'ютери, експлуатуючи помилкув складанні фрагментованих IP-пакетів.

3.4. Шкідливе програмне забезпечення

Одним з найнебезпечніших способів проведення атак є впровадження всистеми, **які** атакують, шкідливого програмного забезпечення. Ми виділимонаступні межі шкідливого ПЗ:

* шкідлива фікція;
* спосіб поширення;
* зовнішнє подання.

Частину, що здійснює руйнівну функцію, будемо називати "бомбою" (хоча,можливо, більш вдалими термінами були б "заряд" або "боєголовка"). Загаломкажучи, спектр шкідливих функцій необмежений, оскільки "бомба", як і

будь-якаінша програма, може володіти якою завгодно складною логікою, але звичайно"бомби" призначаються для:

* впровадження іншого шкідливого ПЗ;
* отримання контролю над системою, яку атакують;
* агресивного споживання ресурсів;
* зміни або руйнування програм й/або даних.По механізму поширення розрізняють:
  * віруси - код, що володіє здатністю до поширення (можливо, зі змінами)шляхом впровадження в інші програми;
  * "хробаки" - код, здатний самостійно, тобто без впровадження в іншіпрограми, викликати поширення своїх копій по ІС й їхнє виконання (дляактивізації вірусу потрібен запуск зараженої програми).

Віруси звичайно поширюються локально, у межах вузла мережі; дляпередачі по мережі їм потрібна зовнішня допомога, така як пересилання зараженого файлу. "Хробаки", навпаки, орієнтовані з першу чергу на подорожіпо мережі.

Іноді саме поширення шкідливого ПЗ викликає агресивне споживанняресурсів і, отже, є шкідливою функцією. Наприклад, "хробаки" "з'їдають"пропускні шляхи мережі й ресурси поштових систем. Із цієї причини для атак надоступність вони не мають потреби у вбудовуванні спеціальних "бомб".

Шкідливий код, що виглядає як функціонально корисна програма,називається троянським. Наприклад, звичайна програма, будучи ураженоювірусом, стає троянською; часом троянські програми виготовляють вручну йпідсувають довірливим користувачам у якому-небудь привабливому пакунку.

Відзначимо, що дані нами визначення й наведена класифікація шкідливогоПЗ відрізняються від загальноприйнятих.

Вікно небезпеки для шкідливого ПЗ з'являється з випуском нового різновиду"бомб", вірусів й/або "хробаків" і перестає існувати з відновленням бази данихантивірусних програм і накладенням інших необхідних латок.

За традицією із усього шкідливого ПЗ найбільша увага громадськостізосереджується на частку вірусів. Однак до березня 1999 року з повним правомможна було стверджувати, що "незважаючи на експонентний ріст числа відомихвірусів, аналогічного росту кількості інцидентів, викликаних ними, незареєстровано. Дотримання нескладних правил "комп'ютерної гігієни" практичнозводить ризик зараження до нуля. Там, де працюють, а не грають, числозаражених комп'ютерів становить лише частки відсотка".

У березні 1999 року, з появою вірусу "Melissa", ситуація кардинальним чиномзмінилася. "Melissa" - це макровірус для файлів MS-Word, що поширюється задопомогою електронної пошти в приєднаних файлах. Коли такий (заражений)приєднаний файл відкривають, він розсилає свої копії по першим 50 адресам задресної книги Microsoft Outlook. У результаті поштові сервери піддаються атаціна доступність.

У цьому випадку нам хотілося б відзначити два моменти.

1. Як уже говорилося, пасивні об'єкти відходять у минуле; так званий активний уміст стає нормою. Файли, які по всіх ознаках повинні були бвідноситься до даних (наприклад, документи у форматах MS-Word абоPostscript, тексти поштових повідомлень), здатні містити інтерпритованікомпоненти, які можуть запускатися неявним чином при відкритті файлу.Як і всяке в цілому прогресивне явище, таке "підвищення активностіданих" має свій зворотний бік (у розглянутому випадку - відставання врозробці механізмів безпеки й помилки в їхній реалізації). Пересічнікористувачі ще не швидко навчаться застосовувати інтерпритованікомпоненти "у мирних цілях" (або хоча б довідаються про їхнє існування),а перед зловмисниками відкрилося власне кажучи необмежене поледіяльності. Як не банально це звучить, але якщо для стрілянини погоробцях викочується гармата, то постраждає в основномустріляючий.
2. Інтеграція різних сервісів, наявність серед них мережевих, загальна зв'язність приклад).Образно кажучи, багато інформаційних систем, якщо не вжити захиснихзаходів, виявляються "в одному човні" (точніше - у кораблі без перебирань),так що досить однієї пробоїни, щоб "човен" відразу пішов на дно.Як це часто буває, слідом за "Melissa" з'явилися на світ ціла серія вірусів,"хробаків" і їхніх комбінацій: "Explorer.zip" (червень 1999), "Bubble Boy"(листопад 1999), "ILOVEYOU" (травень 2000) і т.д. Не те що б від них бувособливо великий збиток, але суспільний резонанс вони викликали чималий.Активний уміст, крім інтерпритованих компонентів документів й інших файлівданих, має ще одне популярне обличчя - так звані мобільні агенти. Це програми,які завантажуються на інші комп'ютери й там виконуються. Найбільш відоміприклади мобільних агентів - Java-аплети, що завантажують накористувальницький комп'ютер й інтерпритовані Internet-навігаторами.Виявилося, що розробити для них модель безпеки, що залишає досить
можливостей для корисних дій, не так вже й просто; ще складніше реалізуватитаку модель без помилок. У серпні 1999 року стали відомі недоліки в реалізаціїтехнологій Active й Java у рамках Microsoft Internet Explorer, які давалиможливість розміщати на Web-серверах шкідливі аплети, що дозволяютьодержувати повний контроль над системою-візитером.

Для впровадження "бомб" часто використовуються помилки типу"переповнення буфера", коли програма, працюючи з областю пам'яті, виходить замежі припустимого й записує в потрібні зловмисникові місця певні дані. Так діявще в 1988 році знаменитий "хробак Морріса"; у червні 1999 року хакери знайшлиспосіб використати аналогічний метод стосовно Microsoft Internet InformationServer (IIS), щоб одержати контроль над Web-сервером. Вікно небезпеки охопиловідразу біля півтора мільйона серверних систем...

Не забуті сучасними зловмисниками й випробувані троянські програми.Наприклад, "троянці" Back Orifice й Netbus дозволяють одержати контроль надкористувацькими системами з різними варіантами MS-Windows.

Таким чином, дія шкідливого ПЗ може бути спрямована не тільки протидоступності, але й проти інших основних аспектів інформаційної безпеки.

3.5. Основні загрози цілісності

На другому місці по масштабу збитку (після ненавмисних помилок інедоглядів) стоять крадіжки й підробки. За даними газети USA Today, ще в 1992році в результаті подібних протиправних дій з використанням персональнихкомп'ютерів американським організаціям був нанесений загальний збиток урозмірі 882 мільйонів доларів. Можна припустити, що реальний збиток бувнабагато більше, оскільки багато організацій по зрозумілих причинахприховують такі інциденти; не викликає сумнівів, що в наші дні збиток відтакого роду дій виріс багаторазово.

У більшості випадків винуватцями виявлялися штатні співробітникиорганізацій, відмінно знайомі з режимом роботи й заходами захисту. Це ще разпідтверджує небезпеку внутрішніх загроз, хоча говорять і пишуть про їх значноменше, ніж про зовнішні.

Раніше ми розводили поняття статичної й динамічної цілісності. З метоюпорушення статичної цілісності зловмисник (як правило, штатний співробітник)може:

* увести невірні дані;
* змінити дані.

Іноді змінюються змістовні дані, іноді - службова інформація. Показовийвипадок порушення цілісності мав місце в 1996 році. Службовець Oracle(особистий секретар віце-президента) пред'явила судовий позов,обвинувачуючи президента корпорації в незаконному звільненні після того, яквона відкинула його залицяння. На доказ своєї правоти жінка показалаелектронний лист, нібито відправлений її начальником президентові. Змістлиста для нас зараз не важливий; важливий час відправлення. Справа в тому,що віце-президент пред'явив, у свою чергу, файл із реєстраційною інформацієюкомпанії стільникового зв'язку, з якого виявлялося, що в зазначений час вінрозмовляв по мобільному телефону, перебуваючи далеко від свого робочогомісця. Таким чином, у суді відбулося протистояння "файл проти файлу".Очевидно, один з них був фальсифікований або змінений, тобто булапорушена його цілісність. Суд вирішив, що підробили електронний лист(секретарка знала пароль віце-президента, оскільки їй було доручено йогоміняти), і позов був відхилений...

(Теоретично можливо, що обидва файли, які фігурували на суді булисправжніми, коректними з погляду цілісності, а лист відправили пакетнимизасобами, однак, на наш погляд, це була б дуже дивною для віце-президента дія).

З наведеного випадку можна зробити висновок не тільки про загрозипорушення цілісності, але й про небезпеку сліпої довіри комп'ютерної інформації.Заголовки електронного листа можуть бути підроблені; лист у цілому може бутифальсифікований особою, що знає пароль відправника (ми наводили відповідніприклади). Відзначимо, що останнє можливо навіть тоді, коли цілісністьконтролюється криптографічними засобами. Тут має місце взаємодія

різнихаспектів інформаційної безпеки: якщо порушено конфіденційність, можепостраждати цілісність.

Ще один урок: загрозою цілісності є не тільки фальсифікація або змінаданих, але й відмова від зроблених дій. Якщо немає засобів забезпечити"безвідмовність", комп'ютерні дані не можуть розглядатися як доказ.

Потенційно уразливі з погляду порушення цілісності не тільки дані, але йпрограми. Впровадження розглянутого вище шкідливого ПЗ - приклад подібногопорушення.

Загрозами динамічної цілісності є порушення атомарності транзакцій,перевпорядкування, крадіжка, дублювання даних або внесення додатковихповідомлень (мережних пакетів і т.п.). Відповідні дії в мережевомусередовищі називаються активним прослуховуванням.

3.6. Основні загрози конфіденційності

Конфіденційну інформацію можна розділити на предметну й службову.Службова інформація (наприклад, паролі користувачів) не відноситься до певноїпредметної області, в інформаційній системі вона відіграє технічну роль, але їїрозкриття особливо небезпечно, оскільки воно несе в собі одержаннянесанкціонованого доступу до всієї інформації, у тому числі предметної.

Навіть якщо інформація зберігається в комп'ютері або призначена длякомп'ютерного використання, загрози її конфіденційності можуть носитинекомп'ютерний і взагалі нетехнічний характер.

Багатьом людям доводиться виконувати ролі користувачів не однієї, а цілогоряду систем (інформаційних сервісів). Якщо для доступу до таких системвикористовуються багаторазові паролі або інша конфіденційна інформація, тонапевно ці дані будуть зберігатися не тільки в голові, але й у записній книжціабо на листках паперу, які користувач часто залишає на робочому столі, а іноділпросто губить. І справа тут не в неорганізованості людей, а в споконвічнійнепридатності парольної схеми. Неможливо пам'ятати багато різних паролів;рекомендації з їх регулярного (по можливості - частої) зміні тільки збільшуютьположення, змушуючи застосовувати нескладні схеми чергування або взагалінамагатися звести справу до двох-трьох легких запам'ятовувань (і настільки жлегко вгадуваних) паролів.

Описаний клас уразливих місць можна назвати розміщенням конфіденційнихданих у середовищі, де їм не забезпечений (найчастіше - і не може бутизабезпечений) необхідний захист. Загроза ж полягає в тому, що хтось невідмовиться довідатися секрети, які самі просяться в руки. Крім паролів, щозберігаються в записних книжках користувачів, у цей клас потрапляє передачаконфіденційних даних у відкритому вигляді (у розмові, у листі, по мережі), щоуможливлює перехоплення даних. Для атаки можуть використовуватися різнітехнічні засоби (підслуховування або прослуховування розмов, пасивнепрослуховування мережі й т.п.), але ідея одна - здійснити доступ до даних у тоймомент, коли вони найменш захищені.

Загрозу перехоплення даних варто брати до уваги не тільки при початковомуконфігуруванні ІС, але й, що дуже важливо, при всіх змінах. Досить

небезпечноюзагрозою є, виставки, на які багато організацій, недовго думаючи, відправляютьустаткування з виробничої мережі, з усіма даними, що зберігаються на них.Залишаються колишніми паролі, при вилученому доступі вони продовжуютьпередаватися у відкритому вигляді. Це погано навіть у межах захищеної мережіорганізації; в об'єднаній мережі виставки - це занадто суворе випробуваннячесності всіх учасників.

Ще один приклад зміни, про яку часто забувають, - зберігання даних нарезервних носіях. Для захисту даних на основних носіях застосовуються розвиненісистеми керування доступом; копії ж нерідко просто лежать у шафах й одержатидоступ до них може багато хто.

Перехоплення даних - дуже серйозна загроза, і якщо конфіденційність дійснює критичною, а дані передаються по багатьох каналах, їхній захист може виявитисядосить складним та дорогим. Технічні засоби перехоплення добре пророблені,доступні, прості в експлуатації, а встановити їх, наприклад на кабельну мережу,може будь-хто, так що цю загрозу потрібно брати до уваги по відношенню нетільки до зовнішніх, але й до внутрішніх комунікацій.

Крадіжки устаткування є загрозою не тільки для резервних носіїв, але й длякомп'ютерів, особливо портативних. Часто ноутбуки залишають без догляду нароботі або в автомобілі, іноді просто гублять.

Небезпечною нетехнічною загрозою конфіденційності є методи морально-психологічного впливу, такі як маскарад - виконання дій під виглядом особи, щоволодіє повноваженнями для доступу до даних (див., наприклад, статтю АйреВінклера "Завдання: шпигунство" в Jet Info, I996,19).

До неприємних загроз, від яких важко захищатися, можна віднестизловживання повноваженнями. У багатьох типах систем привілейованийкористувач (наприклад системний адміністратор) здатний прочитати кожен(незашифрований) файл, одержати доступ до пошти будь-якого користувача й т.д.Інший приклад - завдання збитків при сервісному обслуговуванні. Звичайносервісний інженер одержує необмежений доступ до устаткування й маєможливість діяти в обхід програмних захисних механізмів.

Такими є основні загрози, які завдають найбільшої шкоди суб'єктамінформаційних відносин.

Розділ 4. Адміністративний рівень інформаційної безпеки

4.1. Основні поняття

До адміністративного рівня інформаційної безпеки відносять діїзагального характеру, які робляться керівництвом організації.

Головна мета заходів адміністративного рівня - сформувати програмуробіт в області інформаційної безпеки й забезпечити її виконання, виділяючинеобхідні ресурси й контролюючи стан справ.

Основою програми є політика безпеки, що відбиває підхід організації дозахисту своїх інформаційних активів. Керівництво кожної організації повинноусвідомити необхідність підтримки режиму безпеки й виділити значні ресурсивпровадження цих заходів.

Політика безпеки будується на основі аналізу ризиків, які визнаютьсяреальними для інформаційної системи організації. Коли ризики проаналізованій стратегія захисту визначена, складається програма забезпеченняінформаційної безпеки. Під цю програму виділяються ресурси, призначаютьсявідповідальні, визначається порядок контролю виконання програми й т.п.

Термін "політика безпеки" є не зовсім точним перекладом англійськогословосполучення "security policy", однак у цьому випадку калька кращевідбиває зміст цього поняття, ніж лінгвістично більш вірні "правила безпеки". Ми матимемо на увазі не окремі правила або їхні набори (такого роду рішеннявиносяться на процедурний рівень, мова про який попереду), а стратегіюорганізації в області інформаційної безпеки. Для вироблення стратегії впровадження її в життя потрібні, безсумнівно, політичні рішення, прийняті нанайвищому рівні.

Під політикою безпеки ми будемо розуміти сукупність документованихрішень, прийнятих керівництвом організації й спрямованих на захистінформації й асоційованих з нею ресурсів.

Таке трактування, звичайно, набагато ширше, ніж набір правилрозмежування доступу (саме це означав термін "security policy" в "Помаранчевій книзі" в побудованих на її основі нормативних документах інших країн).

ІС організації й пов'язані з нею інтереси суб'єктів - це складна система, для розгляду якої необхідно застосовувати об'єктно-орієнтований підхід і поняття рівня деталізації. Доцільно виділити, принаймні, три таких рівні, що ми вже робили в прикладі й зробимо ще раз далі.

Щоб розглядати ІС предметно, з використанням актуальних даних, варто скласти карту інформаційної системи. Ця карта, зрозуміло, повинна бути виготовлена в об'єктно-орієнтованому стилі, з можливістю варіювати не тільки рівень деталізації, але й видимі межі об'єктів. Технічним засобом складання, супроводу й візуалізації подібних карт може слугувати вільно розповсюджуваний каркас якої-небудь системи керування.

4.2. Політика безпеки

Із практичної точки зору політику безпеки доцільно розглядати на трьох рівнях деталізації. До верхнього рівня можна віднести рішення, що стосуються організації в цілому. Вони носять досить загальний характер й, як правило, виходять від керівництва організації. Зразковий список подібних рішень може містити в собі наступні елементи:

* рішення сформувати або переглянути комплексну програму забезпечення інформаційної безпеки, призначення відповідальних за впровадження програми;
* формулювання цілей, які переслідує організація в області інформаційної безпеки, визначення загальних напрямків у досягненні цих цілей;
* забезпечення бази для дотримання законів і правил;

* формулювання адміністративних рішень з питань реалізації програми безпеки, які повинні розглядатися на рівні організації в цілому.

Для політики верхнього рівня мети організації в області інформаційної безпеки формулюються в термінах цілісності, доступності й конфіденційності. Якщо організація відповідає за підтримку критично важливих баз даних, на першому плані може стояти зменшення числа втрат, ушкоджень або перекручувань даних. Для організації, що займається продажем комп'ютерної техніки, імовірно, важлива актуальність інформації про надавані послуги й ціни і її доступність максимальній кількості потенційних покупців. Керівництво режимного підприємства в першу чергу піклується про захист від несанкціонованого доступу, тобто про конфіденційність.

На верхній рівень виноситься керування захисними ресурсами й координація використання цих ресурсів, виділення спеціального персоналу для захисту критично важливих систем і взаємодія з іншими організаціями, що забезпечують або контролюють режим безпеки.

Політика верхнього рівня повинна чітко окреслювати сферу свого впливу. Можливо, це будуть усі комп'ютерні системи організації (або навіть більше, якщо політика регламентує деякі аспекти використання співробітниками своїх домашніх комп'ютерів). Можлива, однак, і така ситуація, коли в сферу впливу включаються лише найбільш важливі системи.

У політиці повинні бути визначені обов'язки посадових осіб з вироблення програми безпеки й впровадження її в життя. У цьому сенсі політика безпеки є основою підзвітності персоналу.

Політика верхнього рівня має справу із трьома аспектами законослухняності й виконавчої дисципліни. По-перше, організація повинна дотримуватися існуючих законів. По-друге, варто контролювати дії осіб, відповідальних за вироблення програми безпеки. Нарешті, необхідно забезпечити певний ступінь виконавчості персоналу, а для цього потрібно виробити систему заохочень і покарань.

Загалом кажучи, на верхній рівень варто виносити мінімум питань. Подібне винесення доцільно, коли воно обіцяє значну економію засобів або коли інакше вчинити просто неможливо.

Тобто необхідно включати в документ, що характеризує політику безпеки організації, наступні розділи:

* вступний, підтверджуючий занепокоєність вищого керівництва проблемами інформаційної безпеки;
* організаційний, який має опис підрозділів, комісій, груп і т.д., відповідальних за роботу в області інформаційної безпеки;
* класифікаційний, що описує наявні в організації матеріальні й інформаційні ресурси й необхідний рівень їхнього захисту;
* штатний, що характеризує заходи безпеки, які застосовуються до персоналу (опис посад з погляду інформаційної безпеки, організація навчання й перепідготовки персоналу, порядок реагування на порушення режиму безпеки й т.п.);
* розділ, що висвітлює питання фізичного захисту;
* керівний розділ, що описує підхід до керування комп'ютерами й комп'ютерними мережами;
* розділ, що описує правила розмежування доступу до виробничої інформації;
* розділ, що характеризує порядок розробки й супроводу систем;
* розділ, що описує заходи, спрямовані на забезпечення безперервної роботи організації;
* юридичний розділ, що підтверджує відповідність політики безпеки чинному законодавству.

До середнього рівня можна віднести питання, що стосуються окремих аспектів інформаційної безпеки, але важливі для різних експлуатованих організацією систем. Приклади таких питань - відношення до передових (але, можливо, недостатньо перевірених) технологій, доступ в Internet (як поєднати можливість доступу до інформації із захистом від зовнішніх загроз), використання домашніх комп'ютерів, застосування користувачами неофіційного програмного забезпечення й т.д.

Політика середнього рівня повинна для кожного аспекту висвітлювати наступні теми:

Опис аспекту. Наприклад, якщо розглянути застосування користувачами неофіційного програмного забезпечення, останнє можна визначити як ПЗ, що не було схвалено й/або закуплене на рівні організації.

Область застосування. Варто визначити, де, коли, як, стосовно кого й чому застосовується дана політика безпеки. Наприклад, чи стосується політика, пов'язана з використанням неофіційного програмного забезпечення, організацій-субпідрядників? Чи стосується вона співробітників, що користуються портативними й домашніми комп'ютерами й змушених переносити інформацію на виробничі машини?

Позиція організації з даного аспекту. Продовжуючи приклад з неофіційним програмним забезпеченням, можна уявити собі позиції повної заборони, вироблення процедури прийому подібного ПЗ й т.п. Позицію можна сформулювати у більш загальному вигляді, як набір цілей, які переслідує організація в даному аспекті. Взагалі стиль документів, що визначають політику безпеки (як і їхній перелік), у різних організаціях може сильно відрізнятися.

Ролі й обов'язки. В "політичний" документ необхідно включити інформацію про посадових осіб, відповідальних за реалізацію політики безпеки. Наприклад, якщо для використання неофіційного програмного забезпечення співробітникам потрібен дозвіл керівництва, повинно бути відомо, у кого і як його можна одержати. Якщо неофіційне програмне забезпечення використовувати не можна, варто знати, хто стежить за виконанням даного правила.

Законослухняність. Політика повинна містити загальний опис заборонених дій і покарань за них.

Крапки контакту. Повинно бути відомо, куди варто звертатися за роз'ясненнями, допомогою й додатковою інформацією. Звичайно "крапкою контакту" слугують певні посадові особи, а не конкретна людина, що займає в цей момент дану посаду.

Політика безпеки нижнього рівня стосується конкретних інформаційних сервісів. Вона містить у собі два аспекти - мету й правила їхнього досягнення, тому її часом важко відокремити від питань реалізації. На відміну від двох верхніх рівнів, розглянута політика повинна визначатися більш докладно. Є багато речей, специфічних для окремих видів послуг, які не можна одним чином регламентувати в рамках всієї організації. У той же час, ці речі на стільки важливі для забезпечення режиму безпеки, що рішення, які їх стосуються повинні прийматися на управлінському, а не технічному рівні. Наведемо кілька прикладів питань, на які варто дати відповідь щодо політики безпеки нижнього рівня:

* хто має право доступу до об'єктів, підтримуваним сервісом?
* за яких умов можна читати й модифікувати дані?
* як організований вилучений доступ до сервісу?

При формулюванні цілей політики нижнього рівня можна виходити з міркувань цілісності, доступності й конфіденційності, але не можна на цьому зупинятися. її мета повинна бути більш конкретною. Наприклад, якщо мова йде про систему

розрахунку заробітної плати, можна поставити мету, щоб тільки співробітникам відділу кадрів і бухгалтерії дозволялося вводити й модифікувати інформацію. У більш загальному випадку мета повинна зв'язувати між собою об'єкти сервісу й дії з ними.

Із цілей виводяться правила безпеки, що описують, хто, що й при яких умовах може робити. Що докладніше правила, що більш формально вони викладені, тим простіше підтримати їхнє виконання програмно-технічними засобами. З іншого боку, занадто жорсткі правила можуть заважати роботі користувачів, імовірно, їх доведеться часто переглядати. Керівництво повинно знайти розумний компроміс, коли за прийнятну ціну буде забезпечений прийнятний рівень безпеки, а співробітники не виявляться надмірно зв'язані. Зазвичай найбільш формально висуваються права доступу до об'єктів через особливу важливість даного питання.

4.3. Програма безпеки

Після того, як сформульована політика безпеки, можна приступати до складання програми її реалізації і безпосередньо до реалізації.

Щоб зрозуміти й реалізувати яку-небудь програму, її потрібно структурувати по рівнях, звичайно у відповідності зі структурою організації. У найпростішому й найпоширенішому випадку досить двох рівнів - верхнього, або центрального, котрий охоплює всю організацію, і нижнього, або службового, котрий стосується окремих послуг або груп однорідних сервісів.

Програму верхнього рівня очолює особа, відповідальна за інформаційну безпеку організації. У цієї програми наступні головні цілі:

* керування ризиками (оцінка ризиків, вибір ефективних засобів захисту);
« координація діяльності в області інформаційної безпеки, поповнення й розподіл ресурсів;
* стратегічне планування;
* контроль діяльності в області інформаційної безпеки.

У рамках програми верхнього рівня приймаються стратегічні рішення із забезпечення безпеки, оцінюються технологічні новинки. Інформаційні технології розвиваються дуже швидко, і необхідно мати чітку політику відстеження й впровадження нових засобів.

Контроль діяльності в області безпеки має двосторонню спрямованість. По-перше, необхідно гарантувати, що дії організації не суперечать законам. При цьому варто підтримувати контакти із зовнішніми контролюючими організаціями. По-

друге, потрібно постійно відслідковувати стан безпеки усередині організації, реагувати на випадки порушень і доопрацьовувати захисні заходи з урахуванням зміни обстановки.

Варто підкреслити, що програма верхнього рівня повинна займати певне місце в діяльності організації, вона повинна офіційно прийматися й підтримуватися керівництвом, а також мати певний штат і бюджет.

Ціль програми нижнього рівня - забезпечити надійний й економічний захист конкретного сервісу або групи однорідних сервісів. На цьому рівні вирішується, які варто використовувати механізми захисту; закуповуються й установлюються технічні засоби; виконується повсякденне адміністрування; відслідковується стан слабких місць і т.п. Звичайно за програму нижнього рівня відповідають адміністратори сервісів.

4.4. Синхронізація програми безпеки з життєвим циклом систем

Якщо синхронізувати програму безпеки нижнього рівня з життєвим циклом захищуваного сервісу, можна домогтися більшого ефекту з меншими витратами. Програмісти знають, що додати нову можливість до вже готової системи на порядок складніше, ніж з нуля спроектувати й реалізувати її. Та ж умова виконується для інформаційної безпеки.

У життєвому циклі інформаційного сервісу можна виділити наступні етапи:

Ініціація. На даному етапі виявляється необхідність у придбанні нового сервісу, документується його передбачуване призначення.

Закупівля. На даному етапі складаються специфікації, проробляються варіанти придбання, виконується власне закупівля.

Установка. Сервіс установлюється, конфігурується, тестується й уводиться в експлуатацію.

Експлуатація. На даному етапі сервіс не тільки працює й адмініструється, але й піддається модифікаціям.

Виведення з експлуатації. Відбувається перехід на новий сервіс.

Розглянемо дії, виконувані на кожному з етапів, більш докладно.

На етапі ініціації оформляється розуміння того, що необхідно придбати новий або значно модернізувати існуючий сервіс; визначається, якими характеристиками і якою функціональністю він повинен володіти; оцінюються фінансові й інші обмеження.

З погляду безпеки найважливішою дією тут є оцінка критичності як самого сервісу, так і інформації, що з його допомогою буде оброблятися. Потрібно сформулювати відповіді на наступні питання:

* якого роду інформація призначається для обслуговування новим сервісом?
* які можливі наслідки порушення конфіденційності, цілісності та доступності цієї інформації?
* які загрози, стосовно яких сервіс та інформація будуть найбільш уразливі?
* чи є які-небудь особливості нового сервісу (наприклад, територіальна роззосередженість компонентів), що вимагають прийняття спеціальних процедурних заходів?
* які характеристики персоналу, що має відношення до безпеки (кваліфікація, благонадійність)?
* які законодавчі положення й внутрішні правила, яким повинен відповідати новий сервіс?

Результати оцінки критичності є відправною крапкою в складанні специфікацій. Крім того, вони визначають ту міру уваги, яку служба безпеки організації повинна приділяти новому сервісу на наступних етапах його життєвого циклу.

Етап закупівлі - один із найскладніших. Потрібно остаточно сформулювати вимоги до захисних засобів нового сервісу, до компанії, що може претендувати на роль постачальника, і до кваліфікації, якою повинен володіти персонал, що використовує або обслуговує закуплений продукт. Усі ці відомості оформляються у вигляді специфікації, куди входять не тільки апаратура й програми, але й документація, обслуговування, навчання персоналу. Зрозуміло, особлива увага повинна приділятися питанням сумісності нового сервісу з існуючою конфігурацією. Підкреслимо також, що нерідко засоби безпеки є необов'язковими компонентами комерційних продуктів, і потрібно простежити, щоб відповідні пункти не випали зі специфікації.

Коли продукт закуплений, його необхідно встановити. Незважаючи на можливу простоту, установка є дуже відповідальною справою. По-перше, новий продукт треба сконфігурівати. Як правило, комерційні продукти постачаються з вимкненими засобами безпеки; їх необхідно увімкнути й належним чином налаштувати. Для великої організації, де багато користувачів і даних, початкове налаштування може стати досить працеємною й відповідальною справою.

По-друге, новий сервіс має потребу в процедурних регуляторах. Варто подбати про чистоту й охорону приміщення, про документи, шо регламентують використання сервісу, про підготовку планів на випадок екстрених ситуацій, про організацію навчання користувачів і т.п.

Після прийняття перерахованих заходів необхідно провести тестування. Його повнота й комплексність можуть бути гарантією безпеки експлуатації в штатному режимі.

Період експлуатації - найтриваліший і складний. Із психологічної точки зору найбільшу небезпеку в цей час становлять незначні зміни в конфігурації сервісу, у поводженні користувачів й адміністраторів. Якщо безпеку не підтримувати, вона слабшає. Користувачі не настільки чітко виконують посадові інструкції, адміністратори менш ретельно аналізують реєстраційну інформацію. То один, то інший користувач одержує додаткові привілеї. Здається, що в сутності нічого не змінилося; насправді ж від колишньої безпеки не залишилося й сліду.

Для боротьби з ефектом повільних змін доводиться долучатися до періодичних перевірок сервісу безпеки. Зрозуміло, після значних модифікацій подібні перевірки є обов'язковими.

При виведенні з експлуатації зачіпаються апаратно-програмні компоненти сервісу й оброблювані ними дані. Апаратура продається, утилізується або викидається. Тільки в специфічних випадках необхідно піклуватися про фізичне руйнування апаратних компонентів, що зберігають конфіденційну інформацію. Програми, імовірно, просто стираються, якщо інше не передбачено ліцензійною угодою.

При виведенні даних з експлуатації їх звичайно переносять на іншу систему, архівують, викидають або знищують. Якщо архівування робиться з наміром згодом прочитати дані в іншому місці, варто подбати про апаратно-програмну сумісність засобів читання й запису. Інформаційні технології розвиваються дуже швидко, і через кілька років пристроїв, здатних прочитати старий носій, може просто не виявитися. Якщо дані архівуються в зашифрованому вигляді, необхідно зберегти ключ і засоби розшифровки. Під час архівування й зберігання архівної інформації не можна забувати про підтримку конфіденційності даних.

Розділ 5. Керування ризиками

5.1. Основні поняття

Керування ризиками розглядається нами на адміністративному рівні ІБ, оскільки лише керівництво організації здатне виділити необхідні ресурси, ініціювати й контролювати виконання відповідних програм.

Загалом кажучи, керування ризиками, так само як і вироблення власної політики безпеки, актуально тільки для тих організацій, інформаційні системи яких й/або оброблювані дані можна вважати нестандартними. Звичайну організацію цілком улаштує типовий набір захисних заходів обраних на основі подання про типові ризики або взагалі без усякого аналізу ризиків. Можна провести аналогію між індивідуальним будівництвом й одержанням квартири в районі масової забудови. У першому випадку необхідно прийняти безліч рішень, оформити велику кількість паперів, у другому досить визначитися лише з декількома параметрами.

Використання інформаційних систем пов'язане з певною сукупністю ризиків. Коли можливий збиток неприйнятно великий, необхідно прийняти економічно виправдані заходи захисту. Періодична (пере)оцінка ризиків необхідна для контролю ефективності діяльності в області безпеки й для обліку змін обстановки.

З кількісної точки зору рівень ризику є функцією ймовірності реалізації певної загрози (використовуючи деякі уразливі місця), а також величини можливого збитку.

Таким чином, суть заходів щодо керування ризиками полягає в тому, щоб оцінити їхній розмір, виробити ефективні й економічні заходи зниження ризиків, а потім переконатися, що ризики укладені в прийнятні рамки (і залишаються такими). Отже, керування ризиками містить у собі два види діяльності, які чергуються циклічно:

* (пере)оцінка (вимір) ризиків;
* вибір ефективних та економічних захисних засобів (нейтралізація ризиків).

Стосовно виявлених ризиків можливі наступні дії:

* ліквідація ризику (наприклад, за рахунок усунення причини);
« зменшення ризику (наприклад, за рахунок використання додаткових захисних засобів);
* прийняття ризику (і вироблення плану дії у відповідних умовах);
* переадресація ризику (наприклад, шляхом висновку страхової угоди). Процес керування ризиками можна розділити на наступні етапи:

1. Вибір аналізованих об'єктів і рівня деталізації їхнього розгляду.
2. Вибір методології оцінки ризиків.
3. Ідентифікація активів.

4. Аналіз загроз та їхніх наслідків, виявлення уразливих місць у захисті.
5. Оцінка ризиків.
6. Вибір захисних заходів.
7. Реалізація й перевірка обраних заходів.
8. Оцінка залишкового ризику.

Етапи 6 та 7 відносяться до вибору захисних засобів (нейтралізації ризиків), інші - до оцінки ризиків.

Уже перелік етапів показує, що керування ризиками - процес циклічний. Власне кажучи, останній етап - це оператор кінця циклу, що пропонує повернутися до початку. Ризики потрібно контролювати постійно, періодично проводячи їхню переоцінку. Відзначимо, що сумлінно виконана й ретельно документована перша оцінка може істотно спростити наступну діяльність.

Керування ризиками, як і будь-яку іншу діяльність в області інформаційної безпеки, необхідно інтегрувати в життєвий цикл ІС. Тоді ефект виявляється найбільшим, а витрати - мінімальними. Раніше ми визначили п'ять етапів життєвого циклу. Коротко опишемо, що може дати керування ризиками на кожному з них.

На етапі ініціації відомі ризики варто врахувати при виробленні вимог до системи взагалі й засобів безпеки зокрема.

На етапі закупівлі (розробки) знання ризиків допоможе вибрати відповідні архітектурні рішення, які відіграють ключову роль у забезпеченні безпеки.

На етапі установки виявлені ризики варто враховувати при конфігуруванні, тестуванні й перевірці раніше сформульованих вимог, а повний цикл керування ризиками повинен передувати впровадженню системи в експлуатацію.

На етапі експлуатації керування ризиками повинно супроводжувати всі істотні зміни в системі.

При виведенні системи з експлуатації керування ризиками допомагає переконатися в тім, що міграція даних відбувається безпечним чином.

5.2. Підготовчі етапи керування ризиками

У цьому розділі будуть описані перші три етапи процесу керування ризиками.

Вибір аналізованих об'єктів і рівня деталізації їхнього розгляду - перший крок в оцінці ризиків. Для невеликої організації припустимо розглядати всю інформаційну інфраструктуру; однак якщо організація велика, всеохоплююча оцінка може вимагати неприйнятних витрат часу й сил. У такому випадку варто зосередитися на найбільш важливих сервісах, заздалегідь погоджуючись із

наближеністю підсумкової оцінки. Якщо важливих сервісів усе ще багато, вибираються ті з них, ризики для яких свідомо великі або невідомі.

Ми вже вказували на доцільність створення карти інформаційної системи організації. Для керування ризиками подібна карта особливо важлива, оскільки вона наочно показує, які сервіси обрані для аналізу, а якими довелося знехтувати. Якщо ІС змінюється, а карта підтримується в актуальному стані, то при переоцінці ризиків відразу стане ясно, які нові або істотно змінені сервіси мають потребу в розгляді.

Загалом кажучи, уразливим є кожен компонент інформаційної системи -від мережевого кабелю, який можуть прогризти миші, до бази даних, що може бути зруйнована через невмілі дії адміністратора. Як правило, у сферу аналізу неможливо включити кожен гвинтик і кожен байт. Доводиться зупинятися на деякому рівні деталізації, усвідомлюючи наближеність оцінки. Для нових систем кращий детальний аналіз; стара система, яка піддається незначним модифікаціям, може бути проаналізована більш поверхнево.

Дуже важливо вибрати розумну методологію оцінки ризиків. Метою оцінки є одержання відповіді на два питання: чи прийнятні існуючі ризики, і якщо ні, то які захисні засоби варто використовувати. Виходить, оцінка повинна бути кількісною, що допускає зіставлення із заздалегідь обраними межами допустимості й витратами на реалізацію нових регуляторів безпеки. Керування ризиками - типова оптимізація завдання, й існує досить багато програмних продуктів, здатних допомогти в її вирішенні (іноді подібні продукти просто додаються до книг з інформаційної безпеки). Принципові труднощі, однак, складаються в неточності вихідних даних. Можна, звичайно, спробувати одержати для всіх аналізованих величин грошовий вираз, вирахувати все з точністю до копійки, але особливого сенсу в цьому немає. Практичніше користуватися умовними одиницями. У найпростішому й цілком припустимому випадку можна користуватися трибальною шкалою. Далі ми продемонструємо, як це робиться.

При ідентифікації активів, тобто тих ресурсів і цінностей, які організація намагається захистити, треба, звичайно, ураховувати не тільки компоненти інформаційної системи, але й підтримуючу інфраструктуру, персонал, а також нематеріальні цінності, такі як репутація організації. Відправною крапкою тут є уявлення про місію організації, у будь-якому випадку напрямки діяльності, які бажано (або необхідно) зберегти в кожному разі. Висловлюючись об'єктно-

орієнтованою мовою, треба в першу чергу описати зовнішній інтерфейс організації, розглянутої як абстрактний об'єкт.

Одним з головних результатів процесу ідентифікації активів є одержання детальної інформаційної структури організації й способів її (структури) використання. Ці відомості доцільно нанести на карту ІС як межі відповідних об'єктів.

Інформаційною основою якої-небудь великої організації є мережа, тому в число апаратних активів варто включити комп'ютери (сервери, робочі станції, ПК), периферійні пристрої, зовнішні інтерфейси, кабельне господарство, активне мережеве устаткування (мости, маршрутизатори й т.п.). До програмних активів, імовірно, будуть віднесені операційні системи (мережева, сервери! й клієнтські), прикладне програмне забезпечення, інструментальні засоби, де (у яких вузлах мережі) зберігається програмне забезпечення, і з яких вузлів воно використовується. Третім видом інформаційних активів є дані, які зберігаються, обробляються й передаються по мережі. Варто класифікувати дані по типах і ступеню конфіденційності, виявити місця їхнього зберігання й обробки, способи доступу до них. Все це важливо для оцінки наслідків порушень інформаційної безпеки.

Керування ризиками - процес далеко не лінійний. Практично всі його етапи пов'язані між собою, і по завершенню майже кожного з них може виникнути необхідність повернення до попереднього. Так, при ідентифікації активів може виявитися, що обрані межі аналізу варто розширити, а ступінь деталізації - збільшити. Особливо складний первинний аналіз, коли багаторазові повернення до початку неминучі.

5.3. Основні етапи керування ризиками

Етапи, що передують аналізу загроз, можна вважати підготовчими, оскільки прямого зв'язку з ризиками вони не мають. Ризик з'являється там, де є загрози.

Короткий перелік найпоширеніших загроз був розглянутий нами раніше. На жаль, на практиці загроз набагато більше, причому далеко не всі вони носять комп'ютерний характер. Так, цілком реальною загрозою є наявність мишей і тарганів у займаних організацією приміщеннях. Перші можуть зашкодити кабелю, другі викликати коротке замикання. Як правило, наявність тієї або іншої загрози є наслідком недоліків у захисті інформаційної системи, які, у свою чергу, характеризуються відсутністю деяких сервісів безпеки або недоліками в

реалізуючих їх захисних механізмах. Небезпека прогризання кабелів виникає не просто там, де с миші, вона пов'язана з відсутністю або недостатньою міцністю захисної оболонки.

Перший крок в аналізі загроз - їхня ідентифікація. Розглянуті види загроз варто вибирати виходячи з міркувань здорового глузду (відкинувши, наприклад, землетрус, однак не забуваючи про можливості захоплення організації терористами), але в межах обраних видів провести максимально докладний аналіз.

Доцільно виявляти не тільки самі загрози, але й джерела їхнього виникнення - це допоможе у виборі додаткових засобів захисту. Наприклад, нелегальний вхід у систему може стати наслідком відтворення початкового діалогу, підбору паролю або підключення до мережі неавторизованого устаткування. Очевидно, для протидії кожному з перерахованих способів нелегального входу потрібні свої механізми безпеки.

Після ідентифікації загрози необхідно оцінити ймовірність її здійснення, Можливе використання при цьому трибальної шкали (низька (1), середня (2) і висока (3) ймовірність).

Крім ймовірності здійснення, важливим є розмір потенційного збитку. Наприклад, пожежі бувають нечасто, але збиток від кожної з них, як правило, великий. Розмір збитку також можна оцінити за трибальною шкалою.

Оцінюючи розмір збитку, необхідно мати на увазі не тільки безпосередні витрати на заміну устаткування або відновлення інформації, але й більш віддалені, такі як підрив репутації, ослаблення позицій на ринку й т.п. Нехай, наприклад, у результаті дефектів у керуванні доступом до бухгалтерської інформації співробітники одержали можливість корегувати дані про власну заробітну платню. Наслідком такого стану справ може стати не тільки перевитрата бюджетних або корпоративних засобів, але й повне розкладання колективу, що може призвести до розвалу організації.

Уразливі місця мають властивість притягати до себе не тільки зловмисників, але й порівняно чесних людей. Не кожен тримається перед спокусою дещо збільшити свою зарплатню, якщо є впевненість, що це тобі минеться. Тому, оцінюючи ймовірність здійснення загроз, доцільно виходити не тільки із середньостатистичних даних, але враховувати також специфіку конкретних інформаційних систем. Якщо в

підвалі будинку, займаного організацією, розташовується сауна, а сам будинок має дерев'яні перекриття, то ймовірність пожеж, на жаль, виявиться істотно вище середнього.

Після того, як накопичені вихідні дані й оцінений ступінь невизначеності, можна переходити до обробки інформації, тобто власне до оцінки ризиків. Цілком припустимо застосовувати такий простий метод, як множення ймовірності здійснення загрози на передбачуваний збиток. Якщо для ймовірності й збитку використовувати трибальну шкалу, то можливих добутків буде шість: 1, 2, 3, 4, 6 й 9. Перші два результати можна віднести до низького ризику, третій і четвертий - до середнього, два останні - до високого, після чого з'являється можливість знову привести їх до трибальної шкали. По цій шкалі й варто оцінювати прийнятність ризиків. Щоправда, граничні випадки, коли обчислювальна величина збіглася із прийнятною, доцільно розглядати більш ретельно через наближений характер результату.

Якщо які-небудь ризики виявилися неприпустимо високими, необхідно їх нейтралізувати, реалізувавши додаткові заходи захисту. Як правило, для ліквідації або нейтралізації уразливого місця, що зробило загрозу реальною, існує кілька механізмів безпеки, різних за ефективністю й вартістю. Наприклад, якщо велика ймовірність нелегального входу в систему, можна зажадати, щоб користувачі обирали довгі паролі (скажемо, не менше восьми символів), задіяти програму генерації паролів або закупити інтегровану систему аутентифікації на основі інтелектуальних карт. Якщо є ймовірність навмисного ушкодження сервера баз даних, що може мати серйозні наслідки, можна урізати замок у двері серверної кімнати або поставити біля кожного сервера по охоронцю.

Оцінюючи вартість засобів захисту, доводиться, зрозуміло, ураховувати не тільки прямі витрати на закупівлю устаткування й/або програм, але й витрати на впровадження новинки й, зокрема, навчання й перепідготовку персоналу. Цю вартість також можна оцінити по трибальній шкалі й потім зіставити її з різницею між обчисленим і припустимим ризиком. Якщо за цього показника новий засіб виявляється економічно вигідним, його можна взяти на замітку (підходящих засобів, імовірно, буде декілька). Однак якщо засіб виявиться дорогим, його не слід відразу відкидати, пам'ятаючи про наближення розрахунку.

Вибираючи підходящий спосіб захисту, доцільно враховувати можливість екранування одним механізмом забезпечення безпеки відразу декількох прикладних сервісів. Так зробили в Масачусетському технологічному інституті, захистивши кілька тисяч комп'ютерів сервером аутентифікації КегЬегоз.

Важливою обставиною є сумісність нового засобу зі сформованою організаційною й апаратно-програмною структурою, із традиціями організації. Заходи безпеки, як правило, носять недружній характер, що може негативно позначитися на ентузіазмі співробітників. Часом збереження духу відкритості важливіше мінімізації матеріальних втрат. Втім, такого роду орієнтири повинні бути розставлені в політиці безпеки верхнього рівня.

Можна уявити собі ситуацію, коли для нейтралізації ризику не існує ефективних і прийнятних за ціною заходів. Наприклад, компанія, що базується в сейсмічно небезпечній зоні, не завжди може дозволити собі будівництво захищеної штаб-квартири. У такому випадку доводиться піднімати межу прийнятного ризику й переносити центр ваги на пом'якшення наслідків і вироблення планів відновлення після аварій, стихійних лих та інших подій. Продовжуючи приклад із сейсмобезпекою, можна рекомендувати регулярне тиражування даних в інше місто й володіння засобами відновлення первинної бази даних.

Як і будь-яку іншу діяльність, реалізацію й перевірку нових регуляторів безпеки варто попередньо планувати. У плані необхідно врахувати наявність фінансових засобів і строки навчання персоналу. Якщо мова йде про програмно-технічний механізм захисту, потрібно скласти план тестування (автономного й комплексного).

Коли заплановані заходи впроваджені, необхідно перевірити їхню дієвість, тобто переконатися, що залишкові ризики стали прийнятними. Якщо це насправді так, виходить, можна спокійно намічати дату найближчої переоцінки. У іншому випадку доведеться проаналізувати допущені помилки й провести повторний сеанс керування ризиками негайно.

Розділ 6. Процедурний рівень інформаційної безпеки

6.1. Основні класи заходів процедурного рівня

Ми приступаємо до розгляду заходів безпеки, які орієнтовані на людей, а не на технічні засоби. Саме люди формують режим інформаційної безпеки, і вони ж виявляються головною загрозою, тому "людський чинник" заслуговує на особливу увагу.

В українських компаніях накопичений багатий досвід регламентування й реалізації процедурних (організаційних) заходів, однак справа в тому, що вони прийшли з "докомп'ютерного" минулого, тому вимагають переоцінки.

Варто усвідомити той ступінь залежності від комп'ютерної обробки даних, у яку потрапило сучасне суспільство. Без усякого перебільшення можна сказати про необхідність інформаційної цивільної оборони. Спокійно, без перебільшення, потрібно пояснювати суспільству не тільки переваги, але й небезпеки, пов'язані з використанням інформаційних технологій. Акцент варто робити не на військовій або кримінальній стороні справи, а на цивільних аспектах, пов'язаних з підтримкою нормального функціонування апаратного й програмного забезпечення, тобто концентруватися на питаннях доступності й цілісності даних.

На процедурному рівні можна виділити наступні класи заходів:

* керування персоналом;
* фізичний захист;
* підтримка працездатності;
* реагування на порушення режиму безпеки; » планування відновлювальних робіт.

6.2. Керування персоналом

Керування персоналом починається із прийому нового співробітника на роботу й навіть раніше - зі складання посадових інструкцій. Уже на даному етапі бажано підключити до роботи фахівця з інформаційної безпеки для визначення комп'ютерних привілеїв, асоційованих з посадою. Існує два загальних принципи, як: варто мати на увазі:

* розподіл обов'язків;
* мінімізація привілеїв.

Принцип розподілу обов'язків пропонує так розподіляти ролі й відповідальність, щоб одна людина не могла порушити критично важливий для організації процес. Наприклад, небажана ситуація, коли великі платежі від імені організації виконує одна людина. Надійніше доручити одному співробітникові оформлення заявок на подібні платежі, а іншому - завіряти ці заявки. Інший приклад - процедурні обмеження дій суперкористувача. Можна штучно "відокремити" пароль суперкористувача, повідомивши першу його частину одному співробітникові, а

другу - іншому. Тоді критично важливі дії з адміністрування ІС вони зможуть виконати тільки вдвох, що знижує ймовірність помилок і зловживань.

Принцип мінімізації привілеїв пропонує виділяти користувачам тільки ті права доступу, які необхідні їм для виконання службових обов'язків. Призначення цього принципу очевидно - зменшити збиток від випадкових або навмисних некоректних дій.

Попереднє складання опису посади дозволяє оцінити її критичність і спланувати процедуру перевірки й відбору кандидатів. Що відповідальніше посада, тим ретельніше потрібно перевіряти кандидатів: навести про них довідки, можливо, поговорити з колишніми товаришами по службі й т.д. Подібна процедура може бути тривалою й дорогою, тому немає сенсу додатково ускладнювати її. У той же час, нерозумно й зовсім відмовлятися від попередньої перевірки, щоб випадково не прийняти на роботу людину з карним минулим або психічним захворюванням.

Коли кандидат визначений, він, імовірно, повинен пройти навчання; принаймні, його варто докладно ознайомити зі службовими обов'язками, а також з нормами й процедурами інформаційної безпеки. Бажано, щоб заходи безпеки були ним засвоєні до вступу на посаду й до введення його системного рахунку із вхідним ім'ям, паролем та привілеями.

З моменту введення системного рахунку починається його адміністрування, а також протоколювання й аналіз дій користувача. Поступово змінюється оточення, у якому працює користувач, його службові обов'язки й т.п. Все це вимагає відповідної зміни привілеїв. Технічну складність представляють тимчасові переміщення користувача, виконання ним обов'язків замість співробітника, що пішов у відпустку, і інші обставини, коли спочатку потрібно надати повноваження, а через деякий час обмежити. У такі періоди профіль активності користувача різко змінюється, що створює труднощі при виявленні підозрілих ситуацій. Певної акуратності варто дотримуватися й при видачі нових постійних повноважень, не забуваючи ліквідувати старі права доступу.

Ліквідація системного рахунку користувача, особливо у випадку конфлікту між співробітником та організацією, повинна вироблятися максимально оперативно (в

ідеалі - одночасно з повідомленням про покарання або звільнення). Можливе й фізичне обмеження доступу до робочого місця. Зрозуміло, якщо співробітник звільняється, у нього потрібно прийняти все його комп'ютерне господарство й, зокрема, криптографічні ключ?, якщо використовувалися засоби шифрування.

До керівництва співробітниками додається адміністрування осіб, що працюють за контрактом (наприклад, фахівців фірми-постачальника, що допомагають запустити нову систему). Відповідно до принципу мінімізації привілеїв, їм потрібно виділити рівно стільки прав, скільки необхідно, і забрати ці права відразу по закінченні контракту. Проблема, однак, полягає в тому, що на початковому етапі впровадження "зовнішні" співробітники будуть адмініструвати "місцевих", а не навпаки. Тут на перший план виходить кваліфікація персоналу організації, його здатність швидко навчатися, а також оперативне проведення навчальних курсів. Важливі й принципи вибору ділових партнерів.

Іноді зовнішні організації приймають на обслуговування й адміністрування відповідальні компоненти комп'ютерної системи, наприклад, мережеве устаткування. Нерідко адміністрування виконується у вилученому режимі. Загалом кажучи, це створює в системі додаткові уразливі місця, які необхідно компенсувати посиленим контролем засобів вилученого доступу або навчанням власних співробітників.

Ми бачимо, що проблема навчання - одна з основних з погляду інформаційної безпеки. Якщо співробітник не знайомий з політикою безпеки своєї організації, він не може прагнути до досягнення сформульованих у ній цілей. Не знаючи заходів безпеки, він не зможе їх дотримуватися. Навпаки, якщо співробітник знає, що його дії протоколюються, він, можливо, утримається від порушень.

6.3. Фізичний захист

Безпека інформаційної системи залежить від оточення, у якому вона функціонує. Необхідно прийняти заходи для захисту будинків і прилягаючої території підтримуючої інфраструктури, обчислювальної техніки, носіїв даних.

Основний принцип фізичного захисту, дотримання якого варто постійно контролювати, формулюється як "безперервність захисту в просторі й часі", Раніше

ми розглядали поняття вікна небезпеки. Для фізичного захисту таких вікон бути не повинно.

Ми коротко розглянемо наступні напрямки фізичного захисту:

* фізичне керування доступом;
* протипожежні заходи;
* захист підтримуючої інфраструктури;
* захист від перехоплення даних;
* захист мобільних систем.

Засоби фізичного керування доступом дозволяють контролювати й при необхідності обмежувати вхід та вихід співробітників і відвідувачів. Контролюватися може весь будинок організації, а також окремі приміщення, наприклад, ті, де розташовані сервери, комунікаційна апаратура й т.п.

При проектуванні й реалізації заходів фізичного керування доступом доцільно застосовувати об'єктний підхід. По-перше, визначається периметр безпеки, що обмежує контрольовану територію. На цьому рівні деталізації важливо продумати зовнішній інтерфейс організації - порядок входу/виходу штатних співробітників і відвідувачів, внесення/виносу техніки. Усе, що не входить у зовнішній інтерфейс, повинно бути інкапсульовано, тобто захищене від несанкціонованих проникнень.

По-друге, виробляється декомпозиція контрольованої території, виділяються (під) об'єкти й зв'язки (проходи) між ними. За такої, більш глибокої деталізації варто виділити серед підоб'єктів найбільш критичні з погляду безпеки й забезпечити до них підвищену увага. Декомпозиція повинна бути семантично виправданою, що забезпечує розмежування різнорідних сутностей, таких як устаткування різних власників або персонал, що працює з даними різного ступеня критичності. Важливо зробити так, щоб відвідувачі, за можливості, не мали безпосереднього доступу до комп'ютерів або, у крайньому випадку, подбати про те, щоб від вікон і дверей не проглядалися екрани моніторів і принтери. Необхідно, щоб відвідувачів по зовнішньому вигляду можна було відрізнити від співробітників. Якщо відмінність полягає в тому, що відвідувачам видаються ідентифікаційні картки, а співробітники

ходять "без розпізнавальних знаків", зловмисникові досить зняти картку, щоб його вважали "своїм". Очевидно, що відповідні картки потрібно видавати всім.

Засоби фізичного керування доступом відомі давно. Це охорона, двері із замками, перегородки, телекамери, датчики руху й багато чого іншого. Для вибору оптимального (за критерієм вартість/ефективність) засобу доцільно провести аналіз ризиків (до цього ми ще повернемося). Крім того, є сенс періодично відслідковувати появу технічних новинок у даній області, намагаючись максимально автоматизувати фізичний захист.

Більш докладно дана тема розглянута в статті В. Барсукова "Фізичний захист інформаційних систем" (Jet Info, 1997, 1).

Професія пожежника - одна з найдавніших, але пожежі як і раніше трапляються й завдають великої шкоди. Ми не збираємося цитувати параграфи протипожежних інструкцій або винаходити нові методи боротьби з вогнем - на це є професіонали. Відзначимо лише необхідність установки протипожежної сигналізації й автоматичних засобів пожежогасіння. Звернемо також увагу на те, що захисні заходи можуть створювати нові слабкі місця. Якщо на роботу взяли нового охоронця, це імовірно, поліпшує фізичне керування доступом. Якщо ж він ночами палить і п'є, то через підвищену пожежонебезпеку подібний засіб захисту може лише нашкодити.

До підтримуючої інфраструктури можна віднести системи електро-, водо-і теплопостачання, кондиціонери й засоби комунікацій. У принципі, до них можна застосувати ті ж вимоги цілісності й доступності, що й до інформаційних систем. Для забезпечення цілісності потрібно захищати устаткування від крадіжок та ушкоджень. Дія підтримки доступності варто вибирати устаткування з максимальним терміном праці на відмову, дублювати відповідальні вузли й завжди мати під рукою запчастини.

Окрему проблему становлять аварії водопроводу. Вони відбуваються нечасто, але можуть завдати величезної шкоди. При розміщенні комп'ютерів необхідно взяти до уваги розташування водопровідних і каналізаційних труб і намагатися триматися

від них подалі. Співробітники повинні знати, куди варто звертатися при виявленні протікань,.

Перехоплення даних (про що ми вже писали) може здійснюватися різними способами. Зловмисник може підглядати за екраном монітора, читати пакети, передані по мережі, робити аналіз побічних електромагнітних випромінювань і наведень (ПЕМІН) і т.д. Залишається сподіватися на повселюдне використання криптографії (що, втім, поєднане в нас у країні з безліччю технічних і законодавчих проблем), намагатися максимально розширити контрольовану територію, розмістившись у тихому особнячку, віддалік від інших будинків, намагатися тримати під контролем лінії зв'язку (наприклад, помістити їх у надувну оболонку з виявленням проколювання), але найрозумніше, імовірно, - намагатися усвідомити, що для комерційних систем забезпечення конфіденційності є все-таки не головним завданням.

Бажаючим докладніше ознайомитися з питанням ми рекомендуємо прочитати статтю В. Барсукова "Блокування технологічних каналів витоку інформації" (Jet Info, 1998, 5-6).

Мобільні й портативні комп'ютери - привабливий об'єкт крадіжки. їх часто залишають без догляду, в автомобілі або на роботі, і викрасти такий комп'ютер зовсім нескладно. Раз у раз засоби масової інформації повідомляють про те, що який-небудь офіцер англійської розвідки або американський військовий втратив у такий спосіб рухомое майно. Ми настійно рекомендуємо шифрувати дані на жорстких дисках таких комп'ютерів.

Загалом кажучи, при виборі засобів фізичного захисту варто робити аналіз ризиків. Так, ухвалюючи рішення щодо закупівлі джерела безперебійного живлення, необхідно врахувати якість електроживлення в будинку, займаному організацією (втім, майже напевно воно виявиться поганим), характер і тривалість збоїв електроживлення, вартість доступних джерел і можливі втрати від аварій (поломка техніки, припинення роботи організації й т.п.) (див. також статтю В.Барсукова "Захист комп'ютерних систем від силових деструктивних впливів" в Jet Info, 2000, 2). У той же час, у багатьох випадках рішення очевидні. Засоби протипожежної

безпеки обов'язкові для всіх організацій. Вартість реалізації багатьох засобів (наприклад, установка звичайного замка на двері серверної кімнати) або мала, або хоч і помітна, але все-таки значно менша, ніж можливий збиток. Зокрема, має сенс регулярно копіювати великі бази даних.

6.4. Підтримка працездатності

Далі розглянемо ряд рутинних заходів, спрямованих на підтримку працездатності інформаційних систем. Саме тут чатує найбільша небезпека. Ненавмисні помилки системних адміністраторів і користувачів можуть призвести до ушкодження апаратур, руйнування програм і даних; у найкращому випадку вони створюють пролом у захисті, який уможливлює реалізацію загроз.

Недооцінка факторів безпеки в повсякденній роботі - ахіллєсова п'ята багатьох організацій. Коштовні засоби безпеки втрачають сенс, якщо вони погано документовані, конфліктують з іншим програмним забезпеченням, а пароль системного адміністратора не змінювався з моменту установки.

Можна виділити наступні напрямки повсякденної діяльності:

* підтримка користувачів;
* підтримка програмного забезпечення;
* конфігураційне керування;
* резервне копіювання;
* керування носіями;
* документування;
* регламентні роботи.

Підтримка користувачів має на увазі насамперед консультування й надання допомоги при вирішенні різного роду проблем. Іноді в організаціях створюють для цієї мети спеціальний "довідковий стіл", але частіше від користувачів відкараскується системний адміністратор. Дуже важливо з переліку питань уміти виявляти проблеми, пов'язані з інформаційною безпекою. Так, багато труднощів користувачів, що працюють на персональних комп'ютерах, можуть бути наслідком зараження вірусами. Доцільно фіксувати питання користувачів, щоб виявляти їхні

типові помилки й випускати пам'ятки з рекомендаціями для розповсюджених ситуацій.

Підтримка програмного забезпечення - один з найважливіших засобів забезпечення цілісності інформації. Насамперед, необхідно стежити за тим, яке програмне забезпечення встановлене на комп'ютерах. Якщо користувачі будуть установлювати програми за своїм розсудом, це може привести до зараження вірусами, а також появі утиліт, що діють в обхід захисних засобів. Цілком імовірно також, що "самодіяльність" користувачів поступово приведе до хаосу на їхніх комп'ютерах, а виправляти ситуацію доведеться системному адміністратору.

Другий аспект підтримки програмного забезпечення - контроль за відсутністю неавторизованої зміни програм і прав доступу до них. Сюди ж можна віднести підтримку еталонних копій програмних систем. Звичайно контроль досягається комбінуванням засобів фізичного й логічного керування доступом, а також використанням утиліт перевірки й забезпечення цілісності.

Конфігураційне керування дозволяє контролювати й фіксувати зміни, внесені в програмну конфігурацію. Насамперед, необхідно застрахуватися від випадкових або непродуманих модифікацій, уміти як мінімум повертатися до попередньої, працюючої, версії. Фіксація змін дозволить легко відновити поточну версію після аварії.

Кращий спосіб зменшити кількість помилок у рутинній роботі -максимально автоматизувати її. Праві ті "ледачі" програмісти й системні адміністратори, які, оглянувши поглядом море одноманітних завдань, говорять: "Я нізащо не буду робити цього; я напишу програму, що зробить усе за мене". Автоматизація й безпека залежать один від одного; той, хто піклується в першу чергу про полегшення свого завдання, насправді оптимальним чином формує режим інформаційної безпеки.

Резервне копіювання необхідно для відновлення програм і даних після аварій. І тут доцільно автоматизувати роботу, як мінімум, сформувавши комп'ютерний розклад створення повних й інкрементальних копій, а як максимум - скористатись відповідними програмними продуктами (див., наприклад, Jet Info, 2000, 12). Потрібно також налагодити розміщення копій у безпечному місці, захищеному від

несанкціонованого доступу, пожеж, протікань, тобто від усього, що може привести до крадіжки або ушкодження носіїв. Доцільно мати кілька екземплярів резервних копій і частину з них зберігати за межами території організації, захищаючись у такий спосіб від великих аварій й аналогічних інцидентів.

Час від часу в тестових цілях варто перевіряти можливість відновлення інформації з копій.

Управляти носіями необхідно для забезпечення фізичного захисту й обліку дискет, стрічок, друкованих видач і т.п. Керування носіями повинне забезпечувати конфіденційність, цілісність і доступність інформації, що зберігається за межами комп'ютерних системх. Під фізичним захистом тут розуміється не тільки відбиття спроб несанкціонованого доступу, але й запобігання від шкідливих впливів навколишнього середовища (спеки, холоду, вологи, магнетизму). Керування носіями повинно охоплювати весь життєвий цикл - від закупівлі до виведення з експлуатації. Документування - невід'ємна частина інформаційної безпеки. У вигляді документів оформляється майже все - від політики безпеки до журналу обліку носіїв. Важливо, щоб документація була актуальною, відбивала саме поточний стан справ, причому в несуперечливому вигляді.

До зберігання одних документів (які містять у собі, наприклад, аналіз уразливих місць системи й загроз) можна застосувати вимоги забезпечення конфіденційності, до інших, таких як план ЕІДНОВЛЄННЯ після аварій - вимоги цілісності й доступності (у критичній ситуації план необхідно знайти й прочитати).

Регламентні роботи - дуже серйозна загроза безпеки. Співробітник, що здійснює регламентні роботи, одержує винятковий доступ до системи, і на практиці дуже важко проконтролювати, які саме дії він робить. Тут на перший план виходить ступінь довіри до тих, хто виконує роботу.

6.5. Реагування на порушення режиму безпеки

Програма безпеки, прийнята організацією, повинна передбачати набір оперативних заходів, спрямованих на виявлення й нейтралізацію порушень режиму інформаційної безпеки. Важливо, щоб у подібних випадках послідовність дій була

спланована заздалегідь, оскільки заходи потрібно приймати термінові й скоординовані.

Реакція на порушення режиму безпеки переслідує три головні цілі:

* локалізація інциденту й зменшення нанесеної шкоди;
* виявлення порушника;
* попередження повторних порушень.

В організації повинна бути людина, яка буде на зв'язку 24 години на добу, яка б відповідача за реакцію на порушення. Усі повинні знати координати цієї людини й звертатися до неї за перших ознак небезпеки.

Важливість швидкої й скоординованої реакції можна продемонструвати на наступному прикладі. Нехай локальна мережа підприємства складається із двох сегментів, адміністрованих різними людьми. Далі, нехай в один із сегментів буде заражений вірус. Майже напевно через кілька хвилин (або, у крайньому випадку, кілька десятків хвилин) вірус поширюється й на інший сегмент. Виходить, заходи потрібно ужити негайно. "Вичищати" вірус необхідно одночасно в обох сегментах; у іншому випадку сегмент, відновлений першим, заразиться від іншого, а потім вірус повернеться й у другий сегмент.

Нерідко вимога локатізації інциденту й зменшення нанесеної шкоди вступає в конфлікт із бажанням виявити порушника. У політику безпеки організації пріоритети повинні бути розставлені заздалегідь. Оскільки, як показує практика, виявити зловмисника дуже складно, на наш погляд, у першу чергу варто піклуватися про зменшення збитку.

Щоб знайти порушника, потрібно заздалегідь з'ясувати контактні координати постачальника мережевих послуг і домовитися з ним про саму можливість і порядок виконання відповідних дій.

Щоб запобігти повторним порушенням, необхідно аналізувати кожен інцидент, виявляти причини, накопичувати відомості. Які джерела шкідливого ПЗ? Які користувачі мають звичку вибирати легкі паролі? На подібні питання й повинні дати відповідь результати аналізу.

Необхідно відслідковувати появу нових уразливих місць та якнайшвидше ліквідовувати асоційовані з ними вікна небезпеки. Хтось в організації повинен відслідковувати цей процес, уживати короткострокових заходів і корегувати програму безпеки для вживання довгострокових заходів.

6.6. Планування відновлювальних робіт

Жодна організація не застрахована від серйозних аварій, викликаних природними катаклізмами, діями зловмисника, недбалістю або некомпетентністю. У той же час, у кожній організації Є функції, які керівництво вважає критично важливими, вони повинні виконуватися незважаючи ні на що. Планування відновлювальних робіт дозволяє підготуватися до аварій, зменшити збиток від них і зберегти здатність до функціонування хоча б у мінімальному обсязі.

Відзначимо, що заходи інформаційної безпеки можна розділити на три групи, залежно від того, чи спрямовані вони на попередження, виявлення або ліквідацію наслідків атак. Більшість засобів носить попереджувальний характер. Оперативний аналіз реєстраційної інформації й деякі аспекти реагування на порушення (так званого активного аудиту) слугують для виявлення й відбиття атак. Планування відновлювальних робіт, мабуть, можна віднести до останньої із трьох перерахованих груп.

Процес планування відновлювальних робіт можна розділити на наступні етапи:

* виявлення критично важливих функцій організації, установлення пріоритетів;
* ідентифікація ресурсів, необхідних для виконання критично важливих функцій;
* визначення переліку можливих аварій;
* розробка стратегії відновлювальних робіт;
* підготовка до реалізації обраної стратегії;
* перевірка стратегії.

Плануючи відновлювальні роботи, варто усвідомлювати те, що повністю зберегти функціонування організації не завжди можливо. Необхідно виявити критично важливі функції, без яких організація втрачає свою цілісність, і навіть

серед критичних функцій розставити пріоритети, щоб якнайшвидше й з мінімальними витратами відновити роботу після аварії.

Ідентифікуючи ресурси, необхідні для виконання критично важливій функцій, варто пам'ятати, що багато хто з них мають некомп'ютерний характер. На даному етапі бажано підключати до роботи фахівців різного профілю, здатних у сукупності охопити всі аспекти проблеми. Критичні ресурси звичайно відносяться до однієї з наступних категорій:

* персонал;
* інформаційна інфраструктура;
* фізична інфраструктура.

Формуючи списки відповідальних фахівців, варто враховувати, що деякі з них можуть безпосередньо постраждати від аварії (наприклад, від пожежі), хтось може перебувати в стані стресу, частина співробітників, можливо, не буде мати змоги потрапити на роботу (наприклад, у випадку масових безладь). Бажано мати невеликий резерв фахівців або заздалегідь визначити канали, по яким можна на деякий час залучити додатковий персонал.

Інформаційна інфраструктура містить у собі наступні елементи:

* комп'ютери;
* програми й дані;
* інформаційні сервіси зовнішніх організацій;
* документацію.

Потрібно підготуватися до того, що на "запасному аеродромі", куди організація буде евакуйована після аварії, апаратна платформа може відрізнятися від вихідної. Відповідно, варто продумати заходи підтримки сумісності по програмам і даним.

Серед зовнішніх інформаційних сервісів для комерційних організацій, імовірно, найважливіше отримати оперативну інформацію й зв'язок з державними службами, що курирують даний сектор економіки.

Документація важлива хоча б тому, що не вся інформація, з якою працює організація, представлена в електронному вигляді. Швидше за все, план відновлювальних робіт надрукований на папері.

До фізичної інфраструктури відносяться будинки, інженерні комунікації, засоби зв'язку, оргтехніка й багато чого іншого. Комп'ютерна техніка не може працювати в поганих умовах, без стабільного електроживлення й т.п.

Аналізуючи критичні ресурси, доцільно врахувати часовий профіль їхнього використання. Більшість ресурсів потрібні постійно, але потреба може виникати тільки в певні періоди (наприклад, наприкінці місяця або року при складанні звіту).

При визначенні переліку можливих аварій потрібно спробувати розробити їхні сценарії. Як будуть розвиватися події? Які можуть виявитися масштаби нещастя? Що відбудеться із критичними ресурсами? Наприклад, чи зможуть співробітники потрапити на роботу? Чи будуть виведені з ладу комп'ютери? Чи можливі випадки саботажу? Чи буде працювати зв'язок? Чи постраждає будинок організації? Чи можна буде знайти й прочитати необхідні папери?

Стратегія відновлювальних робіт повинна базуватися на наявних ресурсах і бути не занадто накладною для організації. При розробці стратегії доцільно провести аналіз ризиків, яким піддаються критичні функції, і спробувати обрати найбільш економічне рішення.

Стратегія повинна передбачати не тільки роботу по тимчасовій схемі, але й повернення до нормального функціонування.

Підготовка до реалізації обраної стратегії полягає у виробленні плану дій в екстрених ситуаціях і по їхньому закінченні, а також у забезпеченні деякої надмірності критичних ресурсів. Останнє можливо й без великої витрати засобів, якщо укласти з однією або декількома організаціями угоди про взаємну підтримку у випадку аварій - ті, хто не постраждав, надають частину своїх ресурсів у тимчасове користування менш щасливим партнерам.

Надмірність забезпечується також заходами резервного копіювання, зберігання копій у декількох місцях, поданням інформації в різних виглядах (на папері й у файлах) і т.д.

Має сенс укласти угоду з постачальниками інформаційних послуг про першочергове обслуговування в критичних ситуаціях або укласти угоди з декількома постачальниками. Правда, ці заходи можуть зажадати певних витрат.

Перевірка стратегії виробляється шляхом аналізу підготовленого плану, прийнятих і намічених заходів.

Розділ 7. Основні програмно-технічні заходи

7.1. Основні поняття програмно - технічного рівня інформаційної безпеки

Програмно-технічні заходи, тобто заходи, спрямовані на контроль комп'ютерних сутностей - устаткування, програм й/або даних, утворюють останній і найважливіший рубіж інформаційної безпеки. Нагадаємо, що збиток наносять в основному дії легальних користувачів, стосовно яких процедурні регулятори малоефективні. Головні вороги - некомпетентність і неакуратність при виконанні службових обов'язків, і тільки програмно-технічні заходи здатні їм протистояти.

Комп'ютери допомогли автоматизувати багато областей людської діяльності. Цілком природним бажання покласти на них і забезпечення власної безпеки. Навіть фізичний захист все частіше доручають не охоронцям, а інтегрованим комп'ютерним системам, що дозволяє одночасно відслідковувати переміщення співробітників і в організації, і в інформаційному просторі.

Це друга причина, шо пояснює важливість програмно-технічних заходів.

Треба, однак, ураховувати, що швидкий розвиток інформаційних технологій не тільки надає оборонцям нові можливості, але й **об**'єктивно ускладнює забезпечення надійного захист)', якщо опиратися винятково на заходи програмно-технічного рівня. Причин тут декілька:

* підвищення швидкодії мікросхем, розвиток архітектур з високим ступенем паралелізму дозволяє методом грубої сили переборювати бар'єри (насамперед криптографічні), які раніше здавалися неприступними;
* розвиток мереж і мережевих технологій, збільшення кількості зв'язків між інформаційними системами, ріст пропускної здатності каналів розширюють коло зловмисників, що мають технічну можливість організовувати атаки;
* поява нових інформаційних сервісів призводить і до утворення нових уразливих місць як "усередині" сервісів, так і на їхніх з'єднаннях;

* конкуренція серед виробників програмного забезпечення змушує скорочувати строки розробки, що приводить до зниження якості тестування й випуску продуктів з дефектами захисту;
* парадигма постійного нарощування, нав'язуваня споживачам, потужності апаратного й програмного забезпечення не дозволяє довго залишатися в межах надійних, апробованих конфігурацій й, крім того, вступає в конфлікт із бюджетними обмеженнями, через що знижується частка асигнувань на безпеку.

Перераховані міркування підкреслюють важливість комплексного підходу до інформаційної безпеки, а також необхідність гнучкої позиції при виборі й супроводі програмно-технічних регуляторів.

Центральним для програмно-технічного рівня є поняття сервісу безпеки.

Дотримуючись об'єктно-орієнтованого підходу, при розгляді інформаційної системи з одиничним рівнем деталізації ми побачимо сукупність надаваних нею інформаційних сервісів. Назвемо їх основними. Щоб вони могли функціонувати й мали необхідні властивості, необхідно кілька рівнів додаткових (допоміжних) сервісів - від СУБД і моніторів транзакцій до ядра операційної системи й устаткування.

До допоміжного відносяться сервіси безпеки (ми вже зіштовхувалися з ними при розгляді стандартів і специфікацій в області ІБ); з-поміж них нас у перш}' чергу будуть цікавити універсальні, високорівневі, що допускають використання різними основними й допоміжними сервісами. Далі ми розглянемо наступні сервіси:

* ідентифікація й аутентифікація;
* керування доступом;
* протоколювання й аудит:
* шифрування;
* контроль цілісності;
* екранування;
* аналіз захищеності;
* забезпечення відмовостійкості;

* забезпечення безпечного відновлення;
* тунелювання;
* керування.

Будуть описані вимоги до сервісів безпеки, їхня функціональність, можливі методи реалізації й місце в загальній архітектурі.

Якщо зіставити наведений перелік сервісів із класами функціональних вимог "Загальних критеріїв", то впадає в око їхня істотна розбіжність. Ми не будемо розглядати питання, пов'язані з приватністю. На наш погляд, сервіс безпеки, хоча б частково, повинен перебувати в розпорядженні того, кого він захищає. У випадку ж з приватністю це не так: критично важливі компоненти зосереджені не на клієнтській, а на серверній стороні, так що приватність власне кажучи виявляється властивістю пропонованої інформаційної послуги (у найпростішому випадку, приватність досягається шляхом збереження конфіденційності серверної реєстраційної інформації й захистом від перехоплення даних, для чого досить перерахованих сервісів безпеки).

З іншого боку, наш перелік є ширшим, ніж'у "Загальних критеріях", оскільки в нього входять екранування, аналіз захищеності й тунелювання. Ці сервіси мають важливе значення самі по собі й, крім того, можуть комбінуватися з іншими сервісами для одержання таких необхідних захисних засобів, як, наприклад, віртуальні приватні мережі/'

Сукупність перерахованих вище сервісів безпеки ми будемо називати повним набором. Уважається, що його, у принципі, досить для побудови надійного захисту на програмно-технічному рівні, щоправда, при дотриманні цілого ряду додаткових умов (відсутність уразливих місць, безпечне адміністрування й т.д.).

Для проведення класифікації сервісів безпеки й визначення їхнього місця в загальній архітектурі, заходи безпеки можна розділити на наступні види:

* превентивним, перешкоджаючим порушенням ІБ;
* заходи виявлення порушень;

* локалізуючі, звужуючі зону впливу порушень;

* захід виявлення порушника;
* заходи відновлення режиму безпеки.

Більшість сервісів безпек потрапляє в число превентивних, і це, безумовно, правильно. Аудит і контроль цілісності здатні допомогти у виявленні порушень; активний аудит, крім того, дозволяє запрограмувати реакцію на порушення з метою локалізації й/або простежування. Спрямованість сервісів відмовостійкості й безпечного відновлення очевидна. Нарешті, керування відіграє інфраструктурну роль, обслуговуючи всі аспекти ІС.

7.2. Особливості сучасних інформаційних систем, істотні з погляду безпеки

Інформаційна система типової сучасної організації є досить складним утворенням, побудованим у багаторівневій архітектурі клієнт/сервер, що користується численними зовнішніми сервісами й, у свою чергу, надає власні сервіси зовні. Навіть порівняно невеликі магазини, що забезпечують розрахунок з покупцями по пластикових картах (і, звичайно, що мають зовнішній Web-сервер), залежать від своїх інформаційних систем й, зокрема, від захищеності всіх компонентів систем і комунікацій між ними.

З погляду безпеки найбільш істотними уявляються наступні аспекти сучасних ІС:

* корпоративна мережа має декілька територіально рознесених частин (оскільки організація розташовується на декількох виробничих майданчиках), зв'язки між якими перебувають у компетенції зовнішнього постачальника мережевих послуг, виходити за межі зони, яка контролюється організацією;
* корпоративна мережа має одне або кілька підключень до Internet;
* на кожному з виробничих майданчиків можуть перебувати критично важливі сервери, у доступі до яких мають потребу співробітники, що працюють на інших майданчиках, мобільні користувачі й, можливо, співробітники інших організацій;
* для доступу користувачів можуть застосовуватися не тільки комп'ютери, але й споживчі пристрої, що використовують, зокрема, бездротовий зв'язок;

* протягом одного сеансу роботи користувачу доводиться звертатися до декількох інформаційних сервісів, що спираються на різні апаратно-програмні платформи;
* щодо доступності інформаційних сервісів висуваються жорсткі вимоги, які уособлюються в необхідності цілодобового функціонування з максимальним часом простою приблизно декілька хвилин;
* інформаційна система є мережею з активними агентами, тобто в процесі роботи програмні компоненти, такі як апплети або сервлети, передаються з однієї машини на іншу й виконуються в цільовому середовищі, підтримуючи зв'язок з вилученими компонентами;
* не всі користувацькі системи контролюються мережевими й/або системними адміністраторами організації;
* програмне забезпечення, особливо отримане по мережі, не може вважатися надійним, у ньому можуть бути помилки, що створюють проблеми в захисті;
* конфігурація інформаційної системи постійно змінюється на рівнях адміністративних даних, програм й апаратур (змінюється склад користувачів, їхні привілеї й версії програм, з'являються нові сервіси, нові апаратури й т.п.).

Варто враховувати ще принаймні два моменти. По-перше, для кожного сервісу основні межі ІБ (доступність, цілісність, конфіденційність) трактуються по-своєму. Цілісність із погляду системи керування базами даних і з погляду поштового сервера - речі принципово різні. Безглуздо говорити про безпеку локальної або іншої мережі взагалі, якщо мережа містить у собі різнорідні компоненти. Варто аналізувати захищеність сервісів, що функціонують у мережі. Для різних сервісів і захист будують по-різному. По-друге, основна загроза інформаційної безпеки організацій як і раніше виходить не від зовнішніх зловмисників, а від власних співробітників.

У силу викладених причин далі будуть розглядатися розподілені, різнорідні, багатосервісні еволюціонуючі системи. Відповідно, нас буде цікавити рішення, орієнтоване на подібні конфігурації.

7.3. Архітектурна безпека

Сервіси безпеки, якими б потужними вони не були, самі по собі не можуть гарантувати надійність програмно-технічного рівня захисту. Тільки перевірена архітектура здатна зробити ефективним об'єднання сервісів, забезпечити керованість інформаційної системи, її здатність розвиватися й протистояти новим загрозам при збереженні таких властивостей як висока продуктивність, простота й зручність використання.

Теоретичною основою рішення проблеми архітектурної безпеки є наступне фундаментальне твердження, що ми вже наводили, розглядаючи інтерпретацію "Помаранчевої книги" для мережевих конфігурацій.

"Нехай кожен суб'єкт (тобто процес, що діє від імені якого-небудь користувача) укладений усередині одного компонента й може здійснювати безпосередній доступ до об'єктів тільки в межах цього компонента. Далі нехай кожен компонент містить свій монітор обігів, що відслідковує всі локальні спроби доступу, і всі монітори проводять у життя погоджену політику безпеки. Нехай, нарешті, комунікаційні канали, шо зв'язують компоненти, зберігають конфіденційність і цілісність переданої інформації. Тоді сукупність усіх моніторів утворить єдиний монітор обігів для всієї мережевої конфігурації."

Звернемо увагу на три принципи, які містяться в наведеному твердженні:

* необхідність розробки й впровадження в життя єдиної політики безпеки;
* необхідність забезпечення конфіденційності й цілісності при мережевих взаємодіях;
* необхідність формування складових сервісів по змістовному принципу, щоб кожен отриманий у такий спосіб компонентів мав повний набір захисних засобів і із зовнішньої точки зору представляв собою єдине ціле (не повинно бути інформаційних потоків, що йдуть до незахищених сервісів).

Якщо який-небудь (складовий) сервіс не має повного набору захисних засобів (склад повного набору описаний вище), необхідне залучення додаткових сервісів, які ми будемо називати екранованими. Екрановані сервіси встановлюються на шляхах доступу до не досить захищених елементів; у принципі, один такий сервіс може екранувати (захищати) велику кількість елементів.

Із практичної точки зору найбільш важливими є наступні принципи архітектурної безпеки:

* безперервність захисту в просторі й часі, неможливість оминути захисні засоби;
* наслідування визнаним стандартам, використання апробованих рішень;
* ієрархічна організація ІС із невеликою кількістю сутностей на кожному рівні;
* посилення найслабшої ланки;
* неможливість переходу в небезпечний стан;
* мінімізація привілеїв;
* поділ обов'язків;
* системність оборони;
* розмаїтість захисних засобів;
* простота й керованість інформаційної системи. Пояснимо зміст перерахованих принципів.

Якщо у зловмисника або незадоволеного користувача з'явиться можливість оминути захисні засоби, він, зрозуміло, так і зробить. Визначені вище екрановані сервіси повинні виключити подібну можливість.

Наслідування визнаним стандартам і використання апробованих рішень підвищує надійність ІС і зменшує ймовірність потрапляння в тупикову ситуацію, коли забезпечення безпеки зажадає непомірно великих витрат і принципових модифікацій.

Ієрархічна організація 1С із невеликою кількістю сутностей на кожному рівні необхідна з технологічних міркувань. При порушенні даного принципу система стане некерованою й, отже, забезпечити її безпеку буде неможливо.

Надійність будь-якої оборони визначається найслабшою ланкою. Зловмисник не буде боротися проти сили, він віддасть перевагу легкій перемозі над слабкістю. (Зазвичай найслабшою ланкою виявляється не комп'ютер або програма, а людина, і тоді проблема забезпечення інформаційної безпеки набуває нетехнічного характеру).

Принцип неможливості переходу в небезпечний стан означає, що при будь-яких обставинах, у тому числі позаштатних, захисний засіб або повністю виконує свої

функції, або повністю блокує доступ. Образно кажучи, якщо в міцності механізм звідного моста ламається, міст залишають піднятим, перешкоджаючи проходу ворога.

Стосовно програмно-технічного рівня принцип мінімізації привілеїв пропонує виділяти користувачам й адміністраторам тільки ті права доступу, які необхідні їм для виконання службових обов'язків. Цей принцип дозволяє зменшити збиток від випадкових або навмисних некоректних дій користувачів й адміністраторів.

Принцип поділу обов'язків припускає такий розподіл ролей і відповідальності, щоб одна людина не могла порушити критично важливий для організації процес або створити пролом у захисті за замовленням зловмисників. Зокрема, дотримання даного принципу особливо важливо, щоб запобігти зловмисним або некваліфікованим діям системного адміністратора.

Принцип ешелонованості оборони пропонує не покладатися на один захисний рубіж, яким би надійним він не здавався. Після засобів фізичного захисту повинні слідувати програмно-технічні засоби, за ідентифікацією й аутентифікацією - керування доступом й, як останній рубіж, - протоколювання й аудит. Ешелонована оборона здатна, принаймні, затримати зловмисника, а завдяки наявності такого рубежу, як протоколювання й аудит, його дії не залишаться непоміченими.

Принцип розмаїтості захисних засобів припускає створення різних за своїм характером оборонних рубежів, щоб від потенційного зловмисника вимагалося б оволодіння різноманітними й, по можливості, несумісними між собою навичками.

Для забезпечення високої доступності (безперервності функціонування) необхідно дотримуватися наступних принципів архітектурної безпеки:

* внесення в конфігурацію тієї або іншої форми надмірності (резервне устаткування, запасні канали зв'язку й т.п.);
* наявність засобів виявлення позаштатних ситуацій;
* наявність засобів реконфігурування для відновлення ізоляції й/або заміни компонентів, що відмовили або піддалися атаці на доступність;
* роззосередженість мережевого керування, відсутність єдиної крапки відмови;

* виділення підмереж та ізоляція груп користувачів один від одного. Даний захід, що є узагальненням поділу процесів на рівні операційної системи, обмежує зону поразки при можливих порушеннях інформаційної безпеки.

Ще один важливий архітектурний принцип - мінімізація обсягу захисних засобів, що виносять на клієнтські системи. Причин тому декілька:

* для доступу в корпоративну мережу можуть використовуватися споживчі пристрої з обмеженою функціональністю;
* конфігурацію клієнтських систем важко або неможливо контролювати. До необхідного мінімуму варто віднести реалізацію сервісів безпеки на мережевому й транспортному рівнях і підтримку механізмів аутентифікації, стійких до мережевих загроз.

Розділ 8. Ідентифікація й аутентифікація, керування доступом

8.1. Ідентифікація й аутентифікація

Ідентифікацію й аутентифікацію можна вважати основою програмно-технічних засобів безпеки, оскільки інші сервіси розраховані на обслуговування іменованих суб'єктів. Ідентифікація й аутентифікація - це перша лінія оборони, "прохідна" інформаційного простору організації.

Ідентифікація дозволяє суб'єктові (користувачеві, процесу, що діє від імені певного користувача, або іншому апаратно-програмному компоненту) назвати себе (повідомити своє ім'я). За допомогою аутентифікації друга сторона переконується, що суб'єкт дійсно той, за кого він себе видає. Як синонім слова "аутентифікація" іноді використовують словосполучення "перевірка дійсності".

(Помітимо в дужках, що походження україномовного терміна "аутентифікація" не зовсім зрозуміло. Англійське "authentication" скоріше можна прочитати як "аутентикація"; важко сказати, звідки в середині узялося ще "фі" - може, з ідентифікації? Проте, термін устоявся, він закріплений у Керівних документах України, використаний у численних публікаціях, тому виправити його вже неможливо.)

Аутентифікація буває однобічної (звичайно клієнт доводить свою дійсність серверу) і двосторонньої (взаємної). Приклад однобічної аутентифікації - процедура входу користувача в систему.

У мережевому середовищі, коли сторони ідентифікації/аутентифікації територіально рознесені, у розглянутого сервісу є два основних аспекти:

* що служить аутентифікатором (тобто використовується для підтвердження дійсності суб'єкта);
* як організований (і захищений) обмін даними ідентифікації/аутентифікації.

Суб'єкт може підтвердити свою дійсність, пред'явивши принаймні одну з наступних сутностей:

* щось, що він знає (пароль, особистий ідентифікаційний номер, криптографічний ключ і т.п.);
* щось, чим він володіє (особисту картку або інший пристрій аналогічного призначення);
* щось, що є частиною його самого (голос, відбитки пальців і т.п., тобто свої біометричні характеристики).

У відкритому мережевому середовищі між сторонами ідентифікації/аутентифікації не існує довіреного маршруту; це значить, що в загальному випадку дані, передані суб'єктом, можуть не збігатися з даними, отриманими й використаними для перевірки дійсності. Необхідно забезпечити захист від пасивного й активного прослуховування мережі, тобто від перехоплення, зміни й/або відтворення даних. Передача паролів у відкритому вигляді, мабуть, незадовільна; не рятує положення й шифрування паролів, тому що воно не захищає від відтворення. Потрібні більш складні протоколи аутентифікації.

Надійна ідентифікація й аутентифікація ускладнена не тільки через мережеві загрози, але й з цілого ряду причин. По-перше, майже всі аутентифікаційні сутності можна довідатися, украсти або підробити. По-друге, є протиріччя між надійністю аутентифікації, з одного боку, і зручностями користувача й системного адміністратора з іншогої. Так, з міркувань безпеки необхідно з певною частотою

просити користувача повторно вводити аутентифікаційну інформацію (адже на його місце могла сісти інша людина), а це не тільки клопітно, але й підвищує ймовірність того, що хтось може підглянути за уведенням даних. По-третє, чим надійніший засіб захисту, тим він дорожчий.

Сучасні засоби ідентифікації/аутентифікації повинні підтримувати концепцію єдиного входу в мережу. Єдиний вхід у мережу - це, у першу чергу, вимога зручності дтя користувачів. Якщо в корпоративній мережі багато інформаційних сервісів, що допускають незалежний обіг, то багаторазова ідентифікація/аутентифікація стає занадто обтяжною. На жаль, поки не можна сказати, що єдиний вхід у мережу став нормою, що домінантні рішення поки не сформувалися.

Таким чином, необхідно шукати компроміс між надійністю, доступністю за ціною й зручністю використання й адміністрування засобів ідентифікації й аутентифікації.

Цікаво відзначити, що сервіс ідентифікації/аутентифікації може стати об'єктом атак на доступність. Якщо система сконфігурована так, що після певного числа невдалих спроб пристрій уведення ідентифікаційної інформації (таке, наприклад, як термінал) блокується, то зловмисник може припинити роботу легального користувача буквально декількома натисканнями клавіш.

8.2. Парольна аутентифікація

Головне достоїнство парольної аутентифікації - простота й звичність. Паролі давно вбудовані в операційні системи й інші сервіси. При правильному використанні паролі можуть забезпечити прийнятний для багатьох організацій рівень безпеки. Проте, по сукупності характеристик їх варто визнати найслабшим засобом перевірки дійсності.

Щоб пароль був запам'ятовуваним, його найчастіше роблять простим (ім'я подруги, назва спортивної команди й т.п.). Однак простий пароль неважко вгадати, особливо якщо знати пристрасті даного користувача. Відома класична історія про радянського розвідника Рихарда Зорге, об'єкт уваги якого через слово говорив "карамба"; зрозуміло, цим же словом відкривався надсекретний сейф.

Іноді паролі із самого початку не зберігаються в таємниці, тому що мають стандартні значення, зазначені в документації, і далеко не завжди після установки системи виробляється їхня зміна.

Уведення пароля можна підглянути. Іноді для підглядання використовуються навіть оптичні прилади.

Паролі нерідко повідомляють колегам, щоб ті могли, наприклад, підмінити на якийсь час власника пароля. Теоретично в подібних випадках більш правильно залучити засоби керувань доступом, але на практиці так ніхто не робить: а таємниця, яку знають двоє, це вже не таємниця.

Пароль можна вгадати "методом грубої сили", використовуючи, скажемо, словник. Якщо файл паролів зашифрований, але доступний для читання, його можна скачати до себе на комп'ютер і спробувати підібрати пароль, запрограмувавши повний перебір (передбачається, що алгоритм шифрування відомий).

Проте, важливі заходи дозволяють значно підвищити надійність парольного захисту:

* накладення технічних обмежень (пароль повинен бути не занадто коротким, він повинен містити букви, цифри, знаки пунктуації й т.п.);
* керування терміном дії паролів: їхня періодична зміна;
* обмеження доступу до файлу паролів;
* обмеження числа невдалих спроб входу в систему, це утруднить застосування "методу грубої сили"):
* навчання користувачів;
* використання програмних генераторів паролів (така програма, грунтуючись на нескладних правилах, може породжувати тільки благозвучні й, отже, запам'ятовувані паролі).

Перераховані заходи доцільно застосовувати завжди, навіть якщо поряд з паролями використовуються інші методи аутентифікації.

8.3. Одноразові паролі

Розглянуті вище паролі можна назвати багаторазовими: їхнє розкриття дозволяє зловмисникові діяти від імені легального користувача. Набагато сильнішім засобом, стійким до пасивного прослуховування мережі, є одноразові паролі.

Найбільш відомим програмним генератором одноразових паролів є система S/KEY компанії Bellcore. Ідея цієї системи полягає в наступному. Нехай є однобічна функція f (тобто функція, обчислити зворотну якої за прийнятний час неможливо). Ця функція відома й користувачеві, і серверу аутентифікації. Нехай, далі, є секретний ключ К, відомий тільки користувачеві.

На етапі початкового адміністрування користувача функція f застосовується до ключа К п разів, після чого результат зберігається на сервері. Після цього процедура перевірки дійсності користувача виглядає таким чином:

* сервер надсилає на користувальницьку систему число (п-1);
* користувач застосовує функцію до секретного ключа К (п-1) разів і відправляє результат по мережі на сервер аутентифікації;
* сервер застосовує функцію f до отриманого від користувача значення й порівнює результат з раніше збереженою величиною. У випадку збігу дійсність користувача вважається встаноаченою, сервер запам'ятовує нове значення (прислане користувачем) і зменшує на одиницю лічильник (п).

Насправді реалізація влаштована навряд чи складніше (крім лічильника, сервер посилає значення, використовуване функцією f), але для нас зараз це не важливо. Оскільки функція f незворотня, перехоплення пароля, так само як і одержання доступу до сервера аутентифікації, не дозволяють довідатися секретний ключ К і передбачити наступний одноразовий пароль.

Система S/KEY мас статус Internet-стандарту (RFC 1938).

Інший підхід до надійної аутентифікації складається в генерації нового пароля через невеликий проміжок часу (наприклад, кожних 60 секунд), для чого можуть використовуватися програми або спеціальні інтелектуальні карти (із практичної точки зору такі паролі можна вважати одноразовими). Серверу аутентифікації повинен бути відомий алгоритм генерації паролів й асоційовані з ним параметри; крім того, годинники клієнта й сервера повинні бути синхронізовані.

Сервер аутентифікації Kerberos.

Kerberos - це програмний продукт, розроблений у середині 1980-х років у Масачусетському технологічному інституті і зазнав з тих часів, принципових змін. Клієнтські компоненти Kerberos присутні в більшості сучасних операційних систем.

Kerberos призначений для рішення наступного завдання. Є відкрита (незахищена) мережа, у вузлах якої зосереджені суб'єкти - користувачі, а також клієнтські й серверні програмні системи. Кожен суб'єкт має секретний ключ. Щоб суб'єкт С міг довести свою дійсність суб'єктові S (без цього S не стане обслуговувати С), він повинен не тільки назвати себе, але й продемонструвати знання секретного ключа. С не може просто надіслати S свій секретний ключ, по-перше, тому, що мережа відкрита (доступна для пасивного й активного прослуховування), а, по-друге, тому, що S не знає (і не повинен знати) секретний ключ С. Потрібен менш прямолінійний спосіб демонстрації знання секретного ключа.

Система Kerberos являє собою довірену третю сторону (тобто сторону, якій довіряють усе), яка володіє секретними ключами суб'єктів, яких обслуговують, і допомагаючої їм у попарній перевірці дійсності.

Щоб за допомогою Kerberos одержати доступ до S (звичайно це сервер), С (як правило - клієнт) посилає Kerberos запит, що містить відомості про нього (клієнта) і про запитувану послугу. У відповідь Kerberos повертає так званий квиток, зашифрований секретним ключем сервера, і копію частини інформації із квитка, зашифрований секретним ключем клієнта. Клієнт повинен розшифрувати другу порцію даних і переслати її разом із квитком серверу. Сервер, розшифрувавши квиток, може порівняти його вміст із додатковою інформацією, присланої клієнтом. Збіг свідчить про те, що клієнт зміг розшифрувати призначені йому дані (адже вміст квитка нікому, крім сервера й Kerberos, недоступний), тобто продемонстрував знання секретного ключа. Виходить, клієнт - саме той, за кого себе видає. Підкреслимо, що секретні ключі в процесі перевірки дійсності не передавалися по мережі (навіть у зашифрованому вигляді) - вони тільки використовувалися для

шифрування. Як організований первинний обмін ключами між Kerberos і суб'єктами

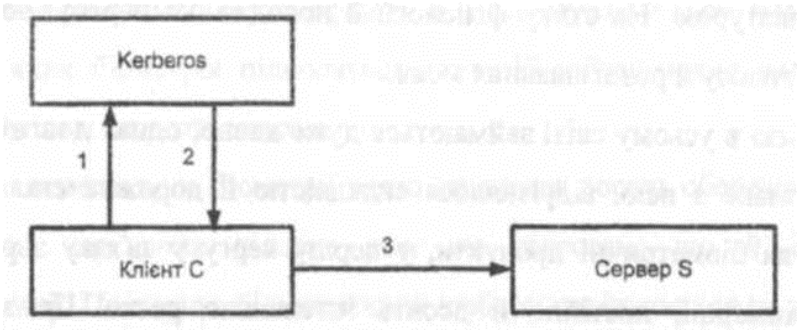


Рис. 8.1. Перевірка сервером S дійсності клієнта С

і як суб'єкти зберігають свої секретні ключі - питання окреме.

Проілюструємо описану процедуру.

1. **Клієнт С —> Kerberos: c, s, ...** (клієнт надає Kerberos інформацію про себе і про запитуваний сервіс)
2. **Kerberos —> клієнт С: (dl) Кс, (Tc.s) Ks** (Kerberos повертає білет, закодований ключем сервера, і додаткову інформацію, закодовану ключем клієнта)
3. **Клієнт С —> сервер S: d2, (Tc.s) Ks** (клієнт надає на сервер білет і додаткову інформацію)

Тут **c** та **s** - відомості (наприклад, ім'я), відповідно, про клієнта й сервер, dl та d2 - додаткова (стосовно квитка) інформація, Tc.s - квиток для клієнта С на обслуговування в сервера S, Кс й Ks - секретні ключі клієнта й сервера, {info}K - інформація info, зашифрована ключем К.

Наведена схема - украй спрощена версія реальної процедури перевірки дійсності. Більш докладний розгляд системи Kerberos можна знайти, наприклад, у статті В. Галатенко "Сервер аутентифікації Kerberos" (Jet Info, 1996, 12-13). Нам же важливо відзначити, що Kerberos не тільки стійкий до мережевих загроз, але й підтримує концепцію єдиного входу в мережу.

8.4. Ідентифікація/аутентифікація за допомогою біометричних даних

Біометрія являє собою сукупність автоматизованих методів ідентифікації й/або аутентифікації людей на основі їх фізіологічних і поведінкових характеристик. До

числа фізіологічних характеристик відносяться особливості відбитків пальців, сітківки й роговиці очей, геометрія руки й особи й т.п. До поведінкових характеристик відносяться динаміка підпису (ручний), стиль роботи із клавіатурою. На стику фізіології й поводження перебувають аналіз особливостей голосу й розпізнавання мови.

Біометрією в усьому світі займаються дуже давно, однак довгий час усе, що було пов'язане з нею, відрізнялося складністю й дорожнечею. Останнім часом попит на біометричні продукти, у першу чергу у зв'язку з розвитком електронної комерції, постійно й досить інтенсивно росте. Це зрозуміло, оскільки з погляду користувача набагато зручніше пред'явити себе самого, ніж щось запам'ятовувати. Попит народжує пропозиція, і на ринку з'явилися відносно недорогі апаратно-програмні продукти, орієнтовані в основному на розпізнавання відбитків пальців.

У загальному вигляді робота з біометричними даними організована в такий спосіб. Спочатку створюється й підтримується база даних характеристик потенційних користувачів. Для цього біометричні характеристики користувача знімаються, обробляються, і результат обробки (названий біометричним шаблоном) заноситься в базу даних (вихідні дані, такі як результат сканування пальця або роговиці, звичайно не зберігаються).

Надалі для ідентифікації (і одночасно аутентифікації) користувача процес зняття й обробки повторюється, після чого виробляється пошук у базі даних шаблонів. У випадку успішного пошуку особистість користувача і її дійсність уважаються встановленими.

Для аутентифікації досить зробити порівняння з одним біометричним шаблоном, обраним на основі попередньо уведених даних.

Звичайно біометрію застосовують разом з іншими аутентифікаторами, такими, наприклад, як інтелектуальні карти. Іноді біометрична аутентифікація є лише першим рубежем захисту й слугує для активізації інтелектуальних карт, що зберігають криптографічні секрети; у такому випадку біометричний шаблон зберігається на тій же карті.

Активність в області біометрії дуже велика. Організовано відповідний консорціум (див. http://www.biometrics.org). Активно ведуться роботи зі стандартизації різних аспектів технології (формату обміну даними, прикладного програмного інтерфейсу й т.п.), публікується маса рекламних статей, у яких біометрія підноситься як засіб забезпечення надбезпеки, що стало доступним широким масам.

На наш погляд, до біометрії варто ставитися досить обережно. Необхідно враховувати, що вона піддана тим же загрозам, що й інші методи аутентифікації. По-перше, біометричний шаблон порівнюється не з результатом первинної обробки характеристик користувача, а з тим, шо прийшло до місця порівняння. А, як відомо, під час маршрутизації багато чого може відбутися. По-друге, біометричні методи не надійніші, ніж база даних шаблонів. По-третє, варто враховувати різницю між застосуванням біометрії на контрольованій території, під пильним оком охорони, і в "польових" умовах, коли, наприклад до пристрою сканування роговиці можуть піднести муляж і т.п. По-четверте, біометричні дані людини змінюються, так що база шаблонів має потребу в супроводі, що створює певні проблеми і для користувачів, і для адміністраторів.

Але головна небезпека полягає в тому, що будь-яка "пробоїна" для біометрії виявляється фатальною. Паролі, при всій їхній ненадійності, у крайньому випадку, можна змінити. Загублену аутентифікаційну карту можна анулювати й завести нову. Палець же, око або голос змінити не можна. Якщо біометричні дані виявляться скомпрометовані, доведеться як мінімум робити істотну модернізацію всієї системи.

8.5. Керування доступом. Основні поняття

Із традиційної точки зору засоби керування доступом дозволяють специфікувати і контролювати дії, які суб'єкти (користувачі й процеси) можуть виконувати над об'єктами (інформацією й іншими комп'ютерними ресурсами). У даному розділі мова йде про логічне керування доступом, що, на відміну від фізичного, реалізується програмними засобами. Логічне керування доступом - це основний механізм багатокористувальницьких систем, покликаний забезпечити

конфіденційність і цілісність об'єктів й, певною мірою, їхня доступність (шляхом заборони обслуговування неавторизованих користувачів).

Розглянемо формальну постановку завдання в традиційному трактуванні. Є сукупність суб'єктів і набір об'єктів. Завдання логічного керування доступом полягає в тому, щоб для кожної пари "об'єкт-об'єкт-суб'єкт-об'єкт" визначити безліч припустимих операцій (залежне, може бути, від деяких додаткових умов) і контролювати виконання встановленого порядку.

Відношення "об'єкт-об'єкти-суб'єкти-об'єкти" можна представити у вигляді матриці доступу, у рядках якої перераховані суб'єкти, у стовпчиках -об'єкти, а в клітинках, розташованих на перетині рядків і стовпчик, записані додаткові умови (наприклад, час і місце дії) і дозволені види доступу. Фрагмент матриці може виглядати, наприклад так:

Таблиця 8.1. Фрагмент матриці доступу

	Файл	Програма	Лінія зв'язку	Реляційна таблиця
Користувач - 1	orw системної консолі	e	**ГУ/** з 8:00 до 18:00	
Користувач - 2				a
„o"- означає дозвіл на передачу прав доступу іншим користувачам „г"- читання, „%'"- запис, „є"- виконання, „a"- додавання інформації				

Тема логічного керування доступом - одна з найскладніших в області інформаційної безпеки. Справа в тому, що саме поняття об'єкта (а тим більше видів доступу) змінюється від сервісу до сервісу. Для операційної системи до об'єктів відносяться файли, пристрої й процеси. Стосовно файлів і пристроїв звичайно розглядаються права на читання, запис, виконання (для програмних файлів), іноді на видалення й додавання. Окремим правом може бути можливість передачі повноважень доступу іншим суб'єктам (так зване право володіння). Процеси можна створювати й знищувати. Сучасні операційні системи можуть підтримувати й інші об'єкти.

Дія систем керування реляційними базами даних об'єкт - це база даних, таблиця, подання, збережена процедура. До таблиць застосовуються операції пошуку, додавання, модифікації й видаленні даних, в інших об'єктів інші види доступу. Розмаїтість об'єктів і застосовуваних до них операцій призводить до: принципової децентралізації логічного керування доступом. Кожен сервіс повинен сам вирішувати, чи дозволити конкретному суб'єктові ту або іншу операцію. Теоретично це узгоджується із сучасним об'єктно-орієнтованим підходом, на практиці ж призводить до значних труднощів. Головна проблема в тім, що багатьом об'єктам можна одержати доступ за допомогою різних сервісів (можливо, при цьому доведеться перебороти деякі технічні труднощі). Так, до реляційних таблиць можна дістатися не тільки засобами СУБД, але й шляхом безпосереднього читання файлів або дискових розділів, підтримуваних операційною системою (розібравшись попередньо в структурі зберігання об'єктів бази даних). У результаті при завданні матриці доступу потрібно брати до уваги не тільки принцип розподілу привілеїв для кожного сервісу, але й існуючі зв'язки між сервісами (доводиться піклуватися про узгодженість різних частин матриці). Аналогічні труднощі виникають при експорті/імпорті даних, коли інформація про права доступу, як правило, губиться (оскільки на новому сервісі вона не має змісту). Отже, обмін даними між різними сервісами становить особливу небезпеку з погляду керування доступом, а при проектуванні й реалізації різнорідної конфігурації необхідно подбати про погоджений розподіл прав доступу суб'єктів до об'єктів і про мінімізацію числа способів експорту/імпорту даних.

Контроль прав доступу виробляється різними компонентами програмного середовища - ядром операційної системи, сервісами безпеки, системою керування базами даних, програмним забезпеченням проміжного шару (таким, як монітор транзакцій) і т.д. Проте, можна виділити загальні критерії, на підставі яких вирішується питання про надання доступу, і загальні методи зберігання матриці доступу.

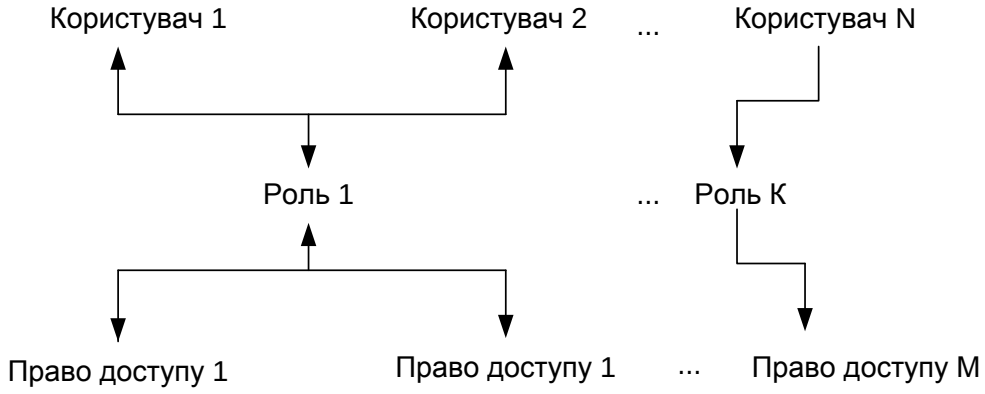
При ухваленні рішення про надання доступу звичайно аналізується наступна інформація:

• ідентифікатор суб'єкта (ідентифікатор користувача, мережна адреса комп'ютера й т.п.). Подібні ідентифікатори є основою довільного (абодискреційного) керування доступом;

• атрибути суб'єкта (мітка безпеки, група користувача й т.п.). Мітки безпеки - основа примусового (мандатного) керування доступом.

Матрицю доступу, через її розрідженість (більшість клітинок - порожні), нерозумно зберігати у вигляді двомірного масиву. Звичайно її зберігають по стовпчиках, тобто для кожного об'єкта підтримується список "допущених" суб'єктів разом з їхніми правами. Елементами списків можуть бути імена груп і шаблони суб'єктів, що слугує більшою допомогою адміністраторові. Деякі проблеми виникають тільки при видаленні суб'єкта, коли доводиться видаляти його ім'я із усіх списків доступу; втім, ця операція проробляється нечасто.

Списки доступу - винятково гнучкий засіб. З їхньою допомогою легко виконати вимогу про гранулярність прав з точністю до користувача. За допомогою списків нескладно додати права або явно заборонити доступ (наприклад, щоб покарати декількох членів групи користувачів). Безумовно, списки є кращим засобом довільного керування доступом.



Переважна більшість операційних систем і систем керування базами даних реалізують саме довільне керування доступом. Основне достоїнство довільного керування - гнучкість. Загалом кажучи, для кожної пари "об'єкт-об'єкт-суб'єкт-об'єкт" можна незалежно задавати права доступу (особливо легко це робити, якщо використовуються списки керування доступом). На жаль, у "довільного" підходу є ряд недоліків. Роззосередженість керування доступом призводить до того, що

довіреними повинні бути багато користувачів, а не тільки системні оператори або адміністратори. Через неуважність або некомпетентність співробітника, що володіє секретною інформацією, цю інформацію можуть довідатися і всі інші користувачі. Отже, довільність керування повинна бути доповнена твердим контролем за реалізацією обраної політики безпеки.

Другий недолік, що уявляється основним, полягає в тому, що права доступу існують окремо від даних. Ніщо не заважає користувачеві, що має доступ до секретної інформації, записати її в доступний усьому файл або замінити корисну утиліту її "троянським" аналогом. Подібне "розділення" прав і даних істотно ускладнює проведення декількома системами погодженої політики безпеки й, головне, робить практично неможливим ефективний контроль узгодженості.

Повертаючись до питання подання матриці доступу, укажемо, що для цього можна використовувати також функціональний спосіб, коли матрицю не зберігають у явному вигляді, а щораз обчислюють уміст відповідних клітинок. Наприклад, при примусовому керуванні доступом застосовується порівняння міток безпеки суб'єкта та об'єкта.

Зручною надбудовою над засобами логічного керування доступом є обмежуючий інтерфейс, коли користувача позбавляють самої можливості спробувати зробити несанкціоновані дії, включивши в число видимих йому об'єктів тільки ті, до яких він має доступ. Подібний підхід звичайно реалізують у межах системи меню (користувачеві показують лише припустимі варіанти вибору) або за допомогою обмежуючих оболонок, таких як restricted shell в ОС Unix.

На закінчення підкреслимо важливість керування доступом не тільки на рівні операційної системи, але й у межах інших сервісів, що входять до складу сучасних додатків, а також, наскільки це можливо, на "стиках" між сервісами. Тут на перший план виходить існування єдиної політики безпеки організації, а також кваліфіковане й погоджене системне адміністрування.

8.6. Рольове керування доступом

При великій кількості користувачів традиційні підсистеми керування доступом стають украй складними для адміністрування. Число зв'язків у них пропорційно

добутку кількості користувачів на кількість об'єктів. Необхідні рішення в об'єктно-орієнтованому стилі, здатні ці складнощі понизити.

Таким рішенням є рольове керування доступом (РКД). Суть його в тому, що між користувачами і їхніми привілеями з'являються проміжні сутності -ролі. Для кожного користувача одночасно можуть бути активними кілька ролей, кожна з яких дає йому певні права (див. рис. 8.2).

Право доступу 1 Право доступу 2 ••• Право доступу **M**

Рис. 8.2. Користувачі, об'єкти й ролі Рольовий доступ нейтральний стосовно конкретних видів прав і способів їхньої перевірки; його можна розглядати як об'єктно-орієнтований каркас, що полегшує адміністрування, оскільки він дозволяє зробити підсистему розмежування доступу керованою при якій завгодно великій кількості користувачів, насамперед за рахунок установлення між ролями зв'язків, аналогічних успадкуванню в об'єктно-орієнтованих системах. Крім того, ролей повинно бути значно менше, ніж користувачів. У результаті число адміністрованих зв'язків стає пропорційним сумі (а не добутку) кількості користувачів й об'єктів, що один по одному величини зменшити вже неможливо.

Рольовий доступ розвивається більше 10 років (сама ідея ролей, зрозуміло, є значно старшою) як на рівні операційних систем, так і у рамках СУБД й інших інформаційних сервісів. Зокрема, існують реалізації рольового доступу для Web-серверів.

У 2001 році Національний інститут стандартів і технологій США запропонував проект стандарту рольового керування доступом (див. http://csrc.nist.gov/rbac/), основні положення якого ми й розглянемо.

Рольове керування доступом оперує наступними основними поняттями: • користувач (людина, інтелектуальний автономний агент і т.п.);

* сеанс роботи користувача;
* роль (звичайно визначається відповідно до організаційної структури);
* об'єкт (сутність, доступ до якої розмежовується; наприклад, файл ОС або таблиця СУБД);
* операція (залежить від об'єкта; для файлів ОС - читання, запис, виконання й т.п.; для таблиць СУБД - вставка, видалення й т.п., для прикладних об'єктів операції можуть бути більш складними);
* право доступу (дозвіл виконувати певні операції над певними об'єктами).

Ролям приписуються користувачі й права доступу; можна вважати, що вони (ролі) іменують відносини "багато хто до багатьох" між користувачами й правами Ролі можуть бути приписані багатьом користувачам; один користувач може бути приписаний декільком ролям. Під час сеансу роботи користувача активізується підмножина ролей, яким він приписаний, у результаті чого він стає власником об'єднання прав, приписаних активним ролям. Одночасно користувач може відкрити кілька сеансів.

Між ролями може бути визначене відношення часткового порядку, так зване успадкуванням. Якщо роль г2 є спадкоємицею гі, то усі права гі приписуються г2, а всі користувачі г2 приписуються гі. Очевидно, що успадкування ролей відповідає успадкуванню класів в об'єктно-орієнтованому програмуванні, тільки правам доступу відповідають методи класів, а користувачам - об'єкти (екземпляри) класів.

Відношення успадкування є ієрархічним, причому права доступу й користувачі поширюються по рівнях ієрархії назустріч один одному. У загальному випадку успадкування є множинним, тобто в однієї ролі може бути кілька попередниць (і, природно, трохи спадкоємиць, яких ми будемо називати також спадкоємицями).

Можна уявити собі формування ієрархії ролей, починаючи з мінімуму прав (і максимуму користувачів), приписуваних ролі "співробітник", з поступовим уточненням складу користувачів і додаванням прав (ролі "системний адміністратор", "бухгалтер" і т.п.), аж до ролі "керівник" (що, втім, не

виходить, що керівникові надаються необмежені права; як й іншим ролям, відповідно до принципу мінімізації привілеїв, цієї ролі доцільно дозволити тільки те, що необхідно для виконання службових обов'язків). Фрагмент подібної ієрархії ролей показаний на рис. 8.3.

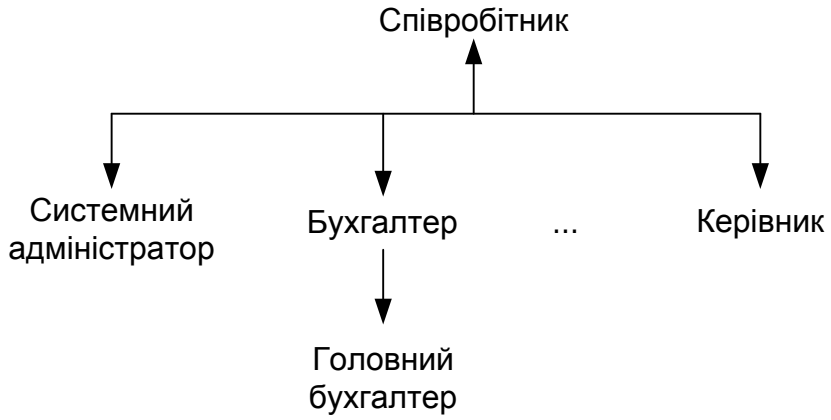


Рис. 8.3. фрагмент ієрархії ролей

Для реалізації ще одного згадуваного раніше важливого принципу інформаційної безпеки вводиться поняття поділу обов'язків, причому у двох виглядах: статичному й динамічному.

Статичний поділ обов'язків накладає обмеження на приписування користувачів ролям. У найпростішому випадку членство в деякій ролі забороняє приписування користувача певній сукупності інших ролей. У загальному випадку дане обмеження задається як пара "безліч ролей - число" (де безліч складається, принаймні, із двох ролей, а число повинне бути більше 1), так що ніякий користувач не може бути приписаний зазначеному (або більшому) числу ролей із заданої кількості. Наприклад, може існувати п'ять бухгалтерських ролей, але політика безпеки допускає членство не більш ніж у двох таких ролях (тут число=3).

При наявності успадкування ролей обмеження набуває більш складного вигляду, але суть залишається простою: при перевірці членства в ролях потрібно враховувати приписування користувачів ролям-спадкоємицям.

Динамічний поділ обов'язків відрізняється від статичного тільки тим, що розглядаються ролі, одночасно активні (у різних сеансах) для даного користувача (а не ті, котрим користувач статично приписаний). Наприклад, один користувач може відігравати роль і касира, і контролера, але не одночасно; щоб стати контролером,

він повинен спочатку закрити касу. Тим самим реалізується так зване тимчасове обмеження довіри, що є аспектом мінімізації привілеїв.

Розглянутий проект стандарту містить специфікації трьох категорій функцій, необхідних для адміністрування РКД:

1. **Адміністративні функції** (створення й супровід ролей та інших атрибутів рольового доступу): створити/видалити роль/користувача, приписати користувача/право ролі або ліквідувати існуючу асоціацію, створити/видалити відношення успадкування між існуючими ролями, створити нову роль і зробити її спадкоємицею/попередницею існуючої ролі, створити/видалити обмеження для статичного/динамічного поділу обов'язків.
2. **Допоміжні функції** (обслуговування сеансів роботи користувачів): відкрити сеанс роботи користувача з активацією якого мається на увазі набір ролей; активувати нову роль, деактивувати роль; перевірити правомірність доступу.
3. **Інформаційні функції** (одержання відомостей про поточну конфігурацію з обліком відносин успадкування). Тут проводиться поділ на обов'язкові й необов'язкові функції. До числа перших належать одержання списку користувачів, приписаних ролі, і списку ролей, яким приписаний користувач.

Усі інші функції віднесені до розряду необов'язкових. Це одержання інформації про права, які приписані ролі, про права заданого користувача (якими він володіє як член певної сукупності), про активні на данний момент сеансу ролі і права, про операції, які роль/користувач правочинні зробити з заданим об'єктом, про статичний/динамічний розподіл обов'язків.

Можна сподіватися, що запропонований стандарт допоможе сформувати єдину термінологію й, що більш важливо, дозволить оцінювати РУД-продукти з єдиних позицій, по єдиній шкалі.

8.7. Керування доступом в Java-середовищі

Java - це об'єктно-орієнтована система програмування, тому й керування доступом у ній спроектоване й реалізовано в об'єктному стилі. Із цієї причини розглянути Java-середовище для нас дуже важливо. Докладно про Java-технологію й безпеку Java-середовища розказано в статті А. Таранова й В. Цишевского "Java у три роки" (Jet Info, 1998, 11-12). З дозволу авторів далі використовуються її фрагменти.

Насамперед, зупинимося на еволюції моделі безпеки Java. В JDK 1.0 була запропонована концепція "пісочниці" (sandbox) - замкненого середовища, у якій виконуються потенційно ненадійні програми (апплети, що надійшли по мережі). Програми, що розташовуються на локальному комп'ютері, уважалися абсолютно надійними, і їм було доступно все, що доступно віртуальній Java-машині.

У число обмежень, що накладаються "пісочницею", входить заборона на доступ до локальної файлової системи, на мережеву взаємодію з усіма хостами, крім джерела апплета, і т.п. Незалежно від рівня безпеки, що досягається при цьому, (а проблеми виникали й з розподілом свій/чужий, і з визначенням джерела апплета), накладені обмеження варто визнати занадто обтяжними: можливості для змістовних дій в апплетів майже не залишається.

Щоб упоратися із цією проблемою, в JDK 1.1 увели розподіл джерел (точніше, розповсюджувачів) апплетів на надійні й ненадійні (джерело визначалося по електронному підпису). Надійні апплети прирівнювалися в правах до "рідного" коду. Зроблене послаблення вирішило проблеми тих, кому прав не вистачало, але захист залишився неешелонованим й, отже, неповним.

В JDK 1.2 сформувалася модель безпеки, використовувана й в Java 2. Від моделі "пісочниці" відмовилися. Сформувалися три основні поняття:

* джерело програми;
* право й сукупність;
* політика безпеки.

Джерело програми визначається парою (URL, розповсюджувачі програми). Останні задаються набором цифрових сертифікатів.

Право - це абстрактне поняття, згідно якого, як і покладено в об'єктному середовищі, стоять класи й об'єкти. У більшості випадків право визначається двома ланцюжками символів - ім'ям ресурсу й дією. Наприклад, ресурсом може виступати файл, а дією - читання. Найважливішим методом "правових" об'єктів є implies. Він перевіряє, чи витікає одне право (запитуване) з іншого (наявного).

Політика безпеки задає відповідність між джерелом і правами програм, що надійшли з нього (формально можна вважати, що кожному джерелу відповідає своя "пісочниця"). В JDK 1.2 "рідні" програми не мають яких-небудь привілеїв у плані безпеки, і політика стосовно них може бути будь-якою. У результаті вийшов традиційний для сучасних ОС і СУБД механізм прав доступу з наступними особливостями:

* Java-програми виступають не від імені користувача, що їх запустив, а від імені джерела програми. (Це досить глибоке й прогресивне трактування, якщо її правильно розвинути, див. наступний розділ);
* немає поняття власника ресурсів, що міг би змінювати права; останні задаються виняткова політикою безпеки (формально можна вважати, що власником усього є той, хто формує політику);
* механізми безпеки забезпечені об'єктною обгорткою.

Досить важливим поняттям у моделі безпеки JDK. 1.2 є контекст виконання. Коли віртуальна Java-машина перевіряє права доступу об'єкта до системного ресурсу, вона розглядає не тільки поточний об'єкт, але й попередні елементи стеку викликів. Доступ надається тільки тоді, коли потрібним правом володіють усі об'єкти в стеці. Розроблювачі Java називають це реалізацією принципу мінімізації привілеїв.

На перший погляд, облік контексту уявляється логічним. Не можна допускати, щоб виклик якого-небудь методу розширював права доступу хоча б з тієї причини, що доступ до системних ресурсів здійснюється не прямо, а за допомогою системних об'єктів, що мають усі права.

На жаль, подібні твердження суперечать одному з основних принципів об'єктного підходу - принципу інкапсуляції. Якщо об'єкт А звертається до об'єкту В,

він не може й не повинен знати, як реалізований У і якими ресурсами він користується для своїх цілей. Якщо А має право викликати який-небудь метод У з певним значенням аргументів, У зобов'язаний обслужити виклик. У протилежному випадку при формуванні політики безпеки доведеться враховувати можливий перелік викликів об'єктів, що, звичайно ж, нереально.

Розробники Java усвідомлювали цю проблему. Щоб упоратися з нею, вони ввели поняття привілейованого інтервалу програми. При виконанні такого інтервалу контекст ігнорується. Привілейована програма відповідає за себе, не цікавлячись передісторією. Аналогом привілейованих програм є файли з бітами переустановки ідентифікатора користувача/групи в ОС Unix, що зайвий раз підтверджує традиційність підходу, реалізованого в JDK 1.2. Відомі загрози безпеки, які привносять подібні файли. Тепер цей не найкращий засіб ОС Unix перейшов до Java.

8.8. Можливий підхід до керування доступом у розподіленому об'єктному середовищі

Уявляється, що в цей час проблема керування доступом існує в трьох майже не пов'язаних між собою проявах:

* традиційні моделі (дискреційна й мандатна);
* модель "пісочниця" (запропонована для Java-середовища й близької їй системі Safe-Tel);
* модель фільтрації (використана в міжмережевих екранах).

На наш погляд, необхідно об'єднати існуючі підходи на основі їхнього розвитку й узагальнення.

Формальна постановка завдання розмежування доступу може виглядати в такий спосіб. Розглядається сукупність об'єктів (у змісті об'єктно-орієнтованого програмування). Частина об'єктів може бути контейнерами, що групують об'єкти-компоненти, що задають для них загальний контекст, шо виконують загальні функції й реалізують перебір компонентів. Контейнери або вкладені один у одного, або не мають загальних компонентів.

З кожним об'єктом асоційований набір інтерфейсів, забезпечених дескрипторами (ДЕ). До об'єкта можна звернутися тільки за допомогою ДЕ. Різні інтерфейси можуть надавати різні методи й бути доступними для різних об'єктів. Кожен контейнер дозволяє опитати набір ДЕ об'єктів-компонентів, що задовольняють деякій умові. Результат, що повертає, у загальному випадку залежить від зухвалого об'єкта.

Об'єкти ізольовані один від одного. Єдиним видом міжоб'єктної взаємодії є виклик методу.

Передбачається, шо використовуються надійні засоби аутентифікації й захисту комунікацій. У плані розмежування доступу локальні й вилучені виклики не розрізняються.

Передбачається також, що дозвіл або заборона на доступ не залежать від можливого паралельного виконання методів (синхронізація представляє окрему проблему, що тут не розглядається).

Розмежовується доступ до інтерфейсів об'єктів, а також до методів об'єктів (з урахуванням значень фактичних параметрів виклику). Правила розмежування доступу (ПРД) задаються у вигляді предикатів над об'єктами.

Розглядається завдання розмежування доступу для виділеного контейнера СС, компонентами якого повинні бути зухвалий й/або викликуваний об'єкти. ДЕ цього контейнера є загальновідомим. Уважається також, що між зовнішніми стосовно виділеного контейнера об'єктами можливі будь-які виклики.

Виконання ПРД контролюється монітором обігів.При виклику методу ми будемо розділяти дії, вироблені зухвалим об'єктом (ініціація виклику) і викликуваним методом (прийом і завершення виклику).

При ініціації виклику може вироблятися перетворення ДЕ фактичних параметрів до вигляду, доступному викликуваному методу ("трансляція інтерфейсу"). Трансляція може мати місце, якщо викликуваний об'єкт не входить у той же контейнер, що й зухвалий.

Параметри методів можуть бути вхідними й/або вихідними. При прийомі виклику виникає інформаційний потік із вхідних параметрів у викликуваний об'єкт.

У момент завершення виклику виникає інформаційний потік з викликуваного об'єкта у вихідні параметри. Ці потоки можуть фігурувати в правилах розмежування доступу.

Структуруємо сукупність всіх ПРД, виділивши чотири групи правил:

* політика безпеки контейнера;
* обмеження на викликуваний метод;
* обмеження на зухвалий метод;
* обмеження, що накладають добровільно.

Правила, загальні для всіх об'єктів, що входять у контейнер 3, назвемо політикою безпеки даного контейнера.

Нехай метод MI об'єкта 01 у крапці PI свого виконання повинен викликати метод M об'єкта O. Правила, яким повинен задовольняти M, можна розділити на три наступні підгрупи:

* правила, що описують вимоги до формальних параметрів виклику;
* правила, що описують вимоги до семантики M;
* реалізаційні правила, що накладають обмеження на можливі реалізації M;
* правила, що накладають обмеження на викликуваний об'єкт O.

Метод M об'єкта O. потенційно доступний для виклику, може висувати до зухвалого об'єкта наступні групи вимог:

- правила, що описують вимоги до фактичних параметрів виклику;
- правила, що накладають обмеження на зухвалий об'єкт.

Можна виділити три різновиди предикатів, що відповідають семантиці й/або особливостям реалізації методів:

* твердження про фактичні параметри виклику методу M у крапці P1;
* предикат, що описує семантику методу M;
* предикат, що описує особливості реалізації методу M.

Перераховані обмеження можна назвати добровільними, оскільки вони відповідають реальному поводженню об'єктів і не пов'язані з якими-небудь зовнішніми вимогами.

Запропонована постановка завдання розмежування доступу відповідає сучасному

етапу розвитку програмування, вона дозволяє відобразити найскладнішу політику безпеки, знайти баланс між багатством виразних можливостей й ефективністю роботи монітора обігів.

Розділ 9. Протоколювання й аудит, шифрування, контроль цілісності

9.1. Протоколювання й аудит. Основні поняття

Під протоколюванням розуміється збір і нагромадження інформації про події, що відбуваються в інформаційній системі. У кожного сервісу свій набір можливих подій, але в кожному разі їх можна розділити на зовнішні (викликані діями інших сервісів), внутрішні (викликані діями самого сервісу) і клієнтські (викликані діями користувачів й адміністраторів).

Аудит - це аналіз накопиченої інформації, проведений оперативно, у реальному часі або періодично (наприклад, раз на день). Оперативний аудит з автоматичним реагуванням на виявлені позаштатні ситуації називається активним.

Реалізація протоколювання й аудиту вирішує наступні завдання:

* забезпечення підзвітності користувачів й адміністраторів;
* забезпечення можливості реконструкції послідовності подій;
* виявлення спроб порушень інформаційної безпеки;
* надання інформації для виявлення й аналізу проблем. Протоколювання вимагає для своєї реалізації здорового глузду. Які події реєструвати? З яким ступенем деталізації? На подібні питання неможливо дати універсальні відповіді. Необхідно стежити за тим, щоб, з одного боку, досягалися перераховані вище завдання, а, з іншого, витрата ресурсів залишилася в межах припустимого. Занадто велике або докладне протоколювання не тільки знижує продуктивність сервісів (що негативно позначається на доступності), але й ускладнює аудит, тобто не збільшує, а зменшує інформаційну безпеку.

Розумний підхід до згаданих питань стосовно операційних систем пропонується в "Помаранчевій книзі", де виділені наступні події:

* вхід у систему (успішний чи ні);
* вихід із системи;
* звертання до вилученої системи;

* операції з файлами (відкрити, закрити, перейменувати, видалити);

* зміна привілеїв або інших атрибутів безпеки (режиму доступу, рівня благонадійності користувача й т.п.).

При протоколюванні події рекомендується записувати, принаймні, наступну інформацію:

* дата й час події;
* унікальний ідентифікатор користувача - ініціатора дії;
  тип події;
* результат дії (успіх або невдача);
* джерело запиту (наприклад, ім'я термінала);
* імена порушених об'єктів (наприклад, відкритих або видалених файлів);
* опис змін, внесених у бази даних захисту (наприклад, нова мітка безпеки об'єкта).

Ще одне важливе поняття, що фігурує в "Помаранчевій книзі", вибіркове протоколювання, як відносно користувачів (уважно стежити тільки за підозрілими), так і відносно подій.

Характерна риса протоколювання й аудиту - залежність від інших засобів безпеки. Ідентифікація й аутентифікація слугують відправною крапкою підзвітності користувачів, логічне керування доступом захищає конфіденційність і цілісність реєстраційної інформації. Можливо, для захисту залучаються й криптографічні методи.

Повертаючись до цілей протоколювання й аудиту, відзначимо, що забезпечення підзвітності важливо в першу чергу як стримуючий засіб. Якщо користувачі й адміністратори знають, що всі їхні дії фіксуються, вони, можливо, утримаються від незаконних операцій. Очевидно, якщо є підстави підозрювати якого-небудь користувача в нечесності, можна реєструвати всі його дії, аж до кожного натискання клавіші. При цьому забезпечується не тільки можливість розслідування випадків порушення режиму безпеки, але й виявлення некоректних змін (якщо в протоколі присутні дані до й після модифікації). Тим самим захищається цілісність інформації.

Реконструкція послідовності подій дозволяє виявити слабкість у захисті сервісів, знайти винуватця вторгнення, оцінити масштаби заподіяного збитку й повернутися до нормальної роботи.

Виявлення спроб порушень інформаційної безпеки - функція активного аудиту, про яку піде мова в наступному розділі. Звичайний аудит дозволяє виявити подібні спроби із запізненням, але й це виявляється корисним. У свій час вияв німецьких хакерів, що діяли за замовленням КДБ, почалася з виявлення підозрілої розбіжності в кілька центів у щоденному звіті великого обчислювального центру.

Виявлення й аналіз проблем можуть допомогти поліпшити такий параметр безпеки, як доступність. Виявивши вузькі місця, можна спробувати переконфігурувати або переналаштувати систему, знову виміряти продуктивність і т.д.

Непросто здійснити організацію погодженого протоколювання й аудиту в розподіленій різнорідній системі. По-перше, деякі компоненти, важливі для безпеки (наприклад, маршрутизатори), можуть не мати своїх ресурсів протоколювання; у такому випадку їх потрібно екранувати іншими сервісами, які візьмуть протоколювання на себе. По-друге, необхідно погоджувати між собою події в різних сервісах.

9.2. Активний аудит. Основні поняття

Під підозрілою активністю розуміється поводження користувача або компонента інформаційної системи, що є злочинним (відповідно до заздалегідь певної політики безпеки) або нетиповим (відповідно до прийнятих критеріїв).

Завдання активного аудиту - оперативно вияшіяти підозрілу активність і надавати засоби для автоматичного реагування на неї.

Активність, що не відповідає політиці безпеки, доцільно розділити на атаки, спрямовані на незаконне одержання повноважень, і на дії, виконувані в межах наявних повноважень, але порушуючи політику безпеки.

Атаки порушують будь-яку осмислену політик)' безпеки. Іншими словами, активність атакуючого є руйнівною незалежно від політики. Отже, для опису й виявлення атак можна застосовувати універсальні методи, інваріантні шодо

політики безпеки, такі як сигнатури і їхнє виявлення у вхідному потоці подій за допомогою апарата експертних систем.

Сигнатура атаки - це сукупність умов, при виконанні яких атака вважається дієвою, яка викликає заздалегідь певну реакцію. Найпростіший приклад сигнатури - "зафіксовані три послідовні невдалі спроби входу в систему з одного термінала", приклад асоційованої реакції - блокування термінала до з'ясування ситуації.
Дії, які виконуються в межах наявних повноважень, але порушують політику безпеки, ми будемо називати зловживанням повноваженнями. Зловживання повноваженнями можливі через неадекватність засобів розмежування доступу обраній політиці безпеки. Найпростішим прикладом зловживань є неетичне поводження суперкористувача, що переглядає особисті файли інших користувачів. Аналізуючи реєстраційну інформацію, можна виявити подібні події й повідомити про них адміністратору безпеки, хоча для цього необхідні відповідні засоби виразу політики безпеки.

Виділення зловживань повноваженнями в окрему групу неправомірних дій, що є засобами активного аудиту, не є загальноприйнятим, однак, на наш погляд, подібний підхід має право на існування й ми будемо його дотримуватися, хоча найбільш радикальним рішенням був би розвиток засобів розмежування доступу (див. "Можливий підхід до керування доступом у розподіленому об'єктному середовиші").

Нетипова поведінка виявляється статистичними методами. У найпростішому випадку застосовують систему порогів, перевищення яких є підозрілим. (Втім, "граничний" метод можна трактувати і як виокремлений випадок сигнатури атаки, і як тривіальний спосіб вираження політики безпеки.) У більш розвинених системах зіставляються довгострокові характеристики роботи (названих довгостроковим профілем) з короткостроковими профілями. (Тут можна побачити аналогію біометричної аутентифікації по поведінкових характеристиках.)

Стосовно засобів активного аудиту розрізняють помилки першого й другого роду: пропуск атак і фіктивні тривоги, відповідно. Небажаність помилок першого роду очевидна; помилки другого роду не менш неприємні, оскільки відволікають

адміністратора безпеки від дійсно важливих справ, побічно сприяючи пропуску атак.

Переваги сигнатурного методу - висока продуктивність, невелика кількість помилок другого роду, обгрунтованість рішень. Основний недолік -невміння виявляти невідомі атаки й варіації відомих атак.

Основні переваги статистичного підходу - універсальність й обґрунтованість рішень, потенційна здатність виявляти невідомі атаки, тобто мінімізація кількості помилок першого роду. Мінуси полягають у відносно високій частці помилок другого роду, поганій роботі у випадку, коли неправомірне поводження є типовим, коли типова поведінка плавно змінюється від легального до неправомірного, а також у випадках, коли типового поводження немає (як показує статистика, таких користувачів приблизно 5-10%).

Засоби активного аудиту можуть розташовуватися на всіх лініях оборони інформаційної системи. На межі контрольованої зони вони можуть виявляти підозрілу активність у точках підключення до зовнішніх мереж (не тільки спроби нелегального проникнення, але й дії по "промацуванню" сервісів безпеки). У корпоративній мережі, у межах інформаційних сервісів і сервісів безпеки, активний аудит у стані виявити й припинити підозрілу активність зовнішніх і внутрішніх користувачів, виявити проблеми в роботі сервісів, викликані як порушеннями безпеки, так й апаратно-програмними помилками. Важливо відзначити, що активний аудит, у принципі, здатний забезпечити захист від атак на доступність.
На жаль, формулювання "у принципі, здатний забезпечити захист" не випадкове. Активний аудит розвивається більше десяти років і перші результати здавалися досить багатообіцяючими. Досить швидко вдалося реалізувати розпізнавання простих типових атак, однак потім була виявлена безліч проблем, пов'язаних з виявленням заздалегідь невідомих атак, атак розподілених, розтягнутих у часі й т.п. Було б наївно очікувати повного рішення подібних проблем найближчим часом. (Оперативне поповнення бази сигнатур атак таким рішенням, звичайно, не є.) Проте, і на нинішній стадії розвитку активний аудит корисний як один з рубежів (точніше, як набір прошарків) ешелонованої оборони.

9.3. Функціональні компоненти й архітектура

У складі засобів активного аудиту можна виділити наступні функціональні компоненти:

* компоненти генерації реєстраційної інформації. Вони перебувають на стику між засобами активного аудиту й контрольованих об'єктів;
* компоненти зберігання згенерованої реєстраційної інформації;
* компоненти витягу реєстраційної інформації (сенсори). Звичайно розрізняють мережні й хостові сенсори, маючи на увазі під першими виділені комп'ютери, мережеві карти яких установлені в режим прослуховування, а під другими - програми, що читають реєстраційні журнали операційної системи. На наш погляд, з розвитком комутаційних технологій це розходження поступово стирається, тому що мережеві сенсори доводиться встановлювати в активному мережному устаткуванні й, по суті, вони стають частиною мережевої ОС;
* компоненти перегляду реєстраційної інформації можуть допомогти при ухваленні рішення про реагування на підозрілу активність;
* компоненти аналізу інформації, що надійшла від сенсорів. Відповідно до наведеного визначення засобів активного аудиту, виділяють пороговий аналізатор, аналізатор порушень політики безпеки, експертну систему, яка виявляє сигнатури атак, а також статистичний аналізатор, що виявляє нетипове поводження;
* компоненти зберігання інформації, які беруть участь в аналізі. Таке зберігання необхідно, наприклад, для виявлення атак, тривалих у часі;
* компоненти прийняття рішень і реагування ("вирішувачі"). "Вирішувач" може одержувати інформацію не тільки від локальних, але й від зовнішніх аналізаторів, проводячи так званий кореляційний аналіз розподілених подій;
* компоненти зберігання інформації про контрольовані об'єкти. Тут можуть зберігатися як пасивні дані, так і методи, необхідні, наприклад, для витягу з об'єкта реєстраційної інформації або для реагування;

* компоненти, які відіграють роль організуючої оболонки для менеджерів активного аудиту, названі моніторами й об'єднуючими аналізаторами, "вирішувачі", сховище описів об'єктів й інтерфейсні компоненти. У число останніх входять компоненти інтерфейсу з іншими моніторами, як рівноправними, так і ти ми, які входять в ієрархію. Такі інтерфейси необхідні, наприклад, для виявлення розподілених, широкомасштабних атак;
* компоненти інтерфейсу з адміністратором безпеки.

Засоби активного аудиту будуються в архітектурі менеджер/агент. Основними агентськими компонентами є сенсори. Аналіз та прийняття рішень - функції менеджерів. Очевидно, між менеджерами й агентами повинні бути сформовані довірені канали.

Підкреслимо важливість інтерфейсних компонентів. Вони корисні як із внутрішньої для засобів активного аудиту точки зору (забезпечують розширюваність, підключення компонентів різних виробників), так і із зовнішньої точки зору. Між менеджерами (між компонентами аналізу й "вирішувачами") можуть існувати горизонтальні зв'язки, необхідні для аналізу розподіленої активності. Можливо також формування ієрархій засобів активного аудиту з винесенням на верхні рівні інформації про найбільш масштабну й небезпечну активність.

Звернемо також увагу на архітектурну спорідненість засобів активного аудиту й керування, що є наслідком спільності виконуваних функцій. Продумані інтерфейсні компоненти можуть істотно полегшити спільну роботу цих засобів.

9.4. Шифрування

Ми приступаємо до розгляду криптографічних сервісів безпеки, точніше, до викладу елементарних відомостей, що допомагають скласти загальне уявлення про комп'ютерну криптографію і її місце в загальній архітектурі інформаційних систем.

Криптографія необхідна для реалізації, принаймні, трьох сервісів безпеки:

* шифрування;
* контроль цілісності;

* аутентифікація (цей сервіс був розглянутий нами раніше). Шифрування - найбільш потужний засіб забезпечення конфіденційності.

У багатьох відносинах воно займає центральне місце серед програмно-технічних регуляторів безпеки, будучи основою реалізації багатьох з них, і в той же час останнім (а часом і єдиним) захисним рубежем. Наприклад, для портативних комп'ютерів тільки шифрування дозволяє забезпечити конфіденційність даних навіть у випадку крадіжки.

У більшості випадків і шифрування, і контроль цілісності відіграють глибоко інфраструктурну роль, залишаючись прозорими й для додатків, і для користувачів. Типове місце цих сервісів безпеки - на мережевому й транспортному рівнях реалізації стека мережевих протоколів.

Розрізняють два основних методи шифрування: симетричний й асиметричний. У першому з них той самий ключ (що зберігається в секреті) використовується як для шифрування, так і для розшифрування даних. Розроблені досить ефективні (швидкі й надійні) методи симетричного шифрування.

Рис. 9.1 ілюструє використання симетричного шифрування. Для визначеності ми будемо вести мову про захист повідомлень, хоча події можуть розвиватися не тільки в просторі, але й у часі, коли зашифровуються й розшифровуються нікуди не переміщені файли.

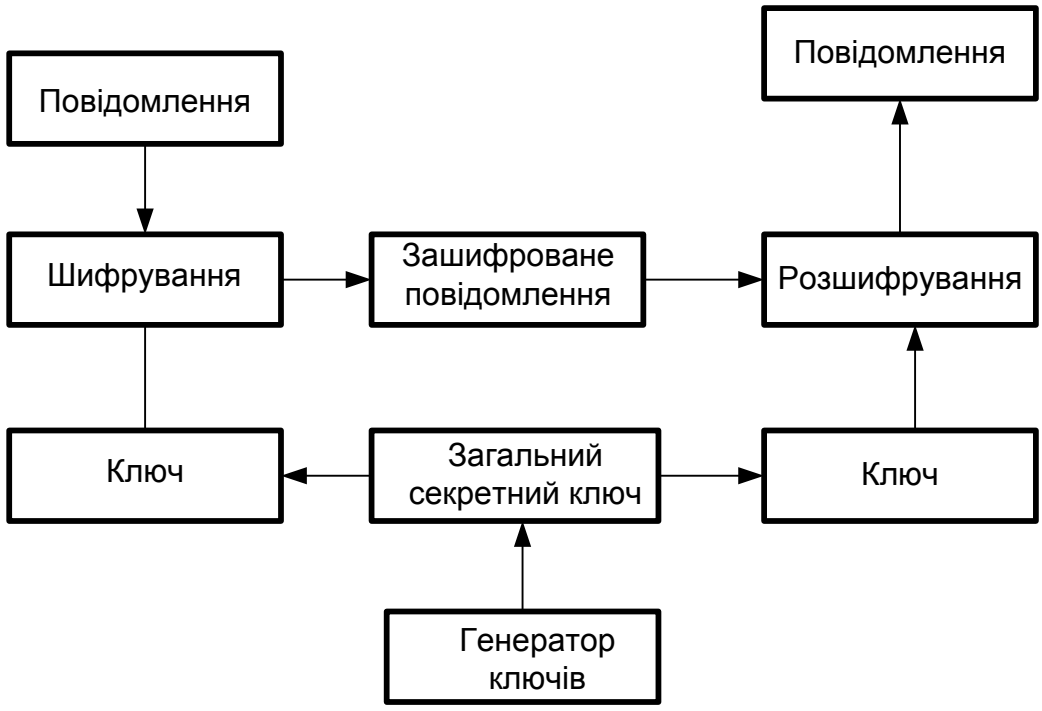


Рис. 9.1. Використання симетричного методу шифрування.

Основним недоліком симетричного шифрування є те, що секретний ключ повинен бути відомий і відправнику, і одержувачу. З одного боку, це створює нову проблему поширення ключів. З іншого боку, одержувач на підставі наявності зашифрованого й розшифрованого повідомлення не може довести, що він одержав це повідомлення від конкретного відправника, оскільки таке ж повідомлення він міг згенерувати самостійно.

В асиметричних методах використовуються два ключі. Один з них, несекретний (він може розміщуватися разом з іншими відкритими відомостями про користувача), застосовується для шифрування, інший (секретний відомий тільки одержувачу) - для розшифрування. Найпопулярнішим серед асиметричних є метод RSA (Райвест, Шамир, Алліман), заснований на операціях з більшими (скажемо, 100-значными) простими числами і їхніми добутками.

Проілюструємо використання асиметричного шифрування (див. рис. 9.2).

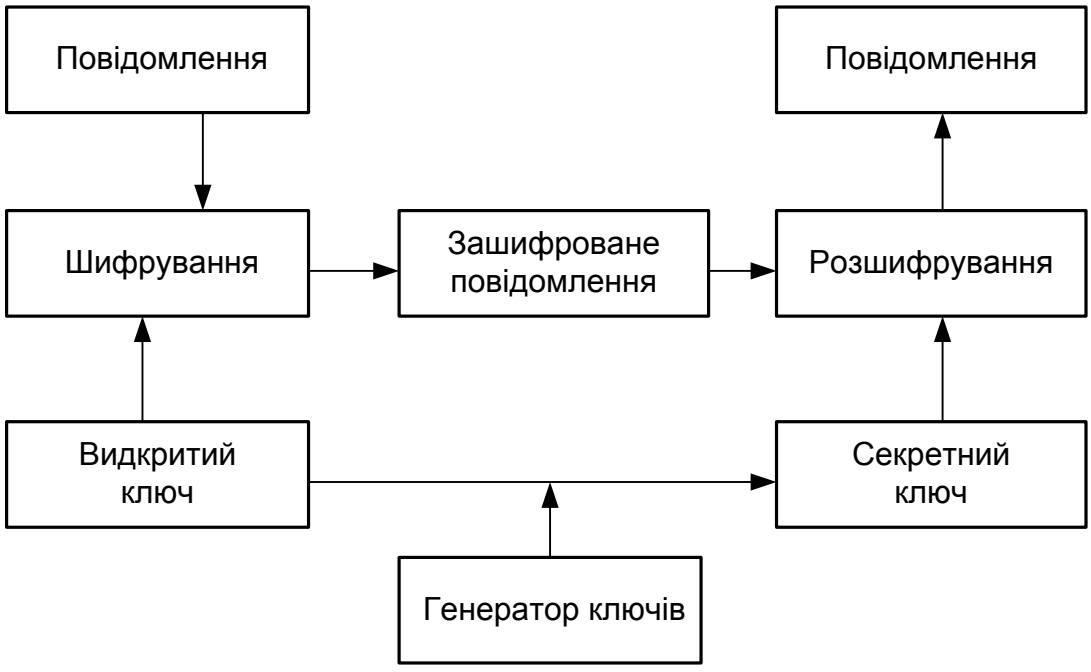


Рис. 9.2. Використання асиметричного методу шифрування

Істотним недоліком асиметричних методів шифрування є їхня низька швидкодія, тому дані методи доводиться сполучати із симетричними (асиметричні методи на 3-4 порядки повільніші). Так, для рішення завдання ефективного

шифрування з передачею секретного ключа, використаного відправником, повідомлення спочатку симетрично зашифровують випадковим ключем, потім цей ключ зашифровують відкритим асиметричним ключем одержувача, після чого повідомлення й ключ відправляються по мережі.

Рис. 9.3 ілюструє ефективне шифрування, реалізоване шляхом сполучення симетричного й асиметричного методів.

На рис. 9.4 показано розшифрування ефективно зашифрованого повідомлення. Відзначимо, що асиметричні методи дозволили вирішити важливе завдання спільного вироблення секретних ключів (це істотно, якщо сторони не довіряють один одному), що обслуговують сеанс взаємодії, при споконвічній відсутності загальних секретів. Для цього використовується алгоритм Диффи-Хелмана.

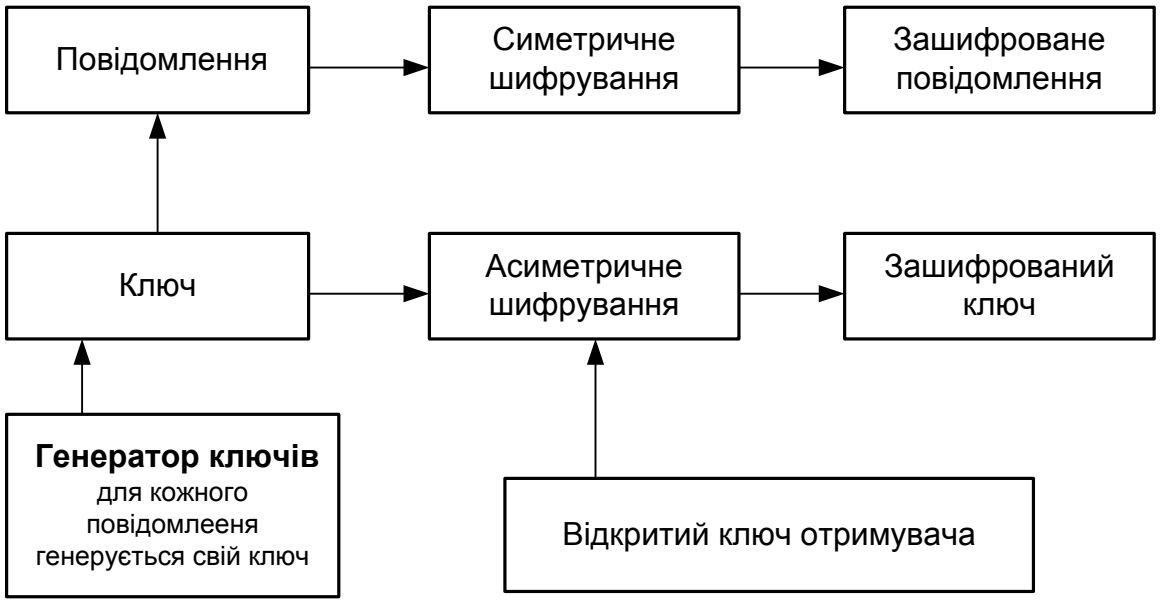


Рис. 9.3. Ефективне шифрування повідомлення

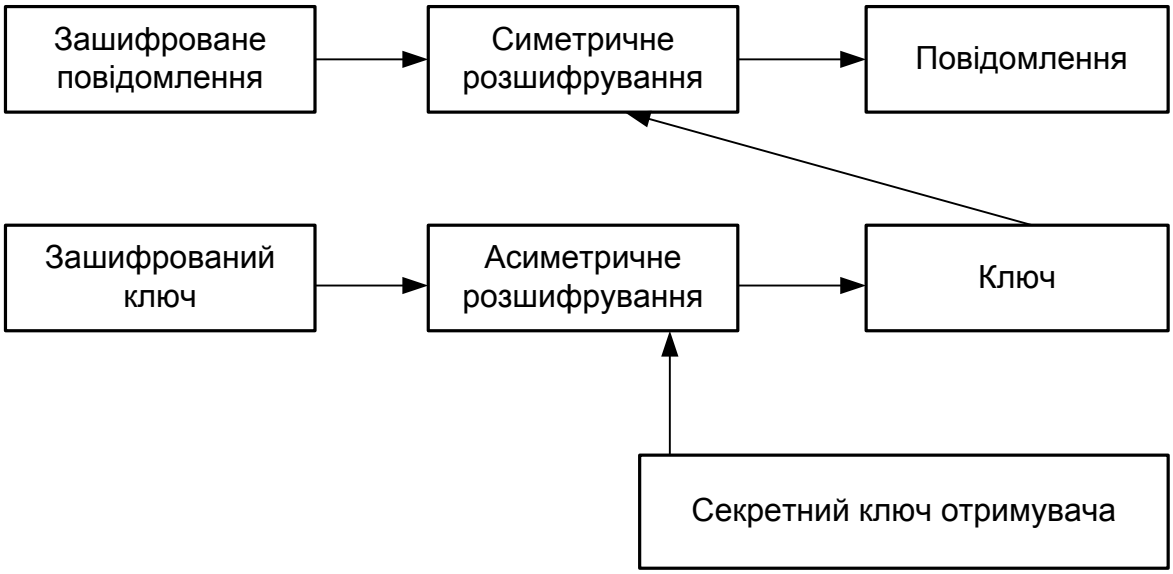


Рис. 9.4. Розшуфрування ефективно зашифрованого повідомлення

Дещо розповсюдився різновид симетричного шифрування, заснований на використанні складених ключів. Ідея полягає в тому, що секретний ключ ділиться на дві частини, що зберігаються окремо. Кожна частина сама по собі не дозволяє виконати розшифрування. Якщо в правоохоронних органів з'являються підозри щодо особи, що використовує деякий ключ, вони можуть у встановленому порядку одержати половинки ключа й далі діяти звичайним для симетричного розшифрування чином.

Порядок роботи зі складеними ключами - вдалий приклад проходження принципу поділу обов'язків. Він дозволяє сполучати права на різного роду таємниці (персональну, комерційну) з можливістю ефективно стежити за порушниками закону, хоча, звичайно, тут дуже багато тонкостей і технічного, і юридичного плану.

Багато криптографічних алгоритмів у якості одного з параметрів вимагають псевдовипадкового значення, у випадку передбачення якого в алгоритмі з'являється уразливість (подібне уразливе місце було виявлено в деяких варіантах Web-навігаторів). Генерація псевдовипадкових послідовностей - важливий аспект криптографії, на якому ми, однак, зупинятися не будемо.

Більш докладну інформацію про комп'ютерну криптографію можна дізнатися зі статті Г. Семенова "Не тільки шифрування, або Огляд криптотехнологій" (Jet Info, 2001, 3).

9.5. Контроль цілісності

Криптографічні методи дозволяють надійно контролювати цілісність як окремих порцій даних, так і їхніх наборів (таких як потік повідомлень); визначати дійсність джерела даних; гарантувати неможливість відмовитися від зроблених дій ("безвідмовність").

В основі криптографічного контролю цілісності лежать два поняття:

* хеш-функція;

* електронний цифровий підпис (ЕЦП).

Хеш-функція - це складнозворстнє перетворення даних (однобічна функція), реалізована, як правило, засобами симетричного шифрування зі зв'язуванням блоків. Результат шифрування останнього блоку (що залежить від усіх попередніх) і слугує результатом хеш-функції. Нехай є дані, цілісність яких потрібно перевірити, хеш-функція й раніше обчислений результат її застосування до вихідних даних (так званий дайджест). Позначимо хеш-функцію через Ь, вихідні дані - через Т, перевіряючі дані - через Т. Контроль цілісності даних зводиться до перевірки рівності h(T') = h(T). Якщо воно виконано, уважається, що Т = Т. Збіг дайджестів для різних даних називається колізією. У принципі, колізії, звичайно, можливі, оскільки потужність сукупності дайджестів є меншою, ніж потужність безлічі хешованих даних, однак те, що п є функція однобічна, означає, що за прийнятний час спеціально організувати колізію неможливо.

Розглянемо тепер застосування асиметричного шифрування для вироблення й перевірки електронного цифрового підпису. Нехай E(T) позначає результат шифрування тексту Т за допомогою відкритого ключа, а $D(T)$ -результат розшифрування тексту Т (як правило шифрованого) за допомогою секретного ключа. Щоб асиметричний метод міг застосовуватися для реалізації ЕІДП, необхідно виконання тотожності

E(D(T)) = D(E(T)) = T

На рис. 9.5 показана процедура вироблення електронного цифрового підпису, що склдає в шифруванні перетворенням D дайджесту h(T).

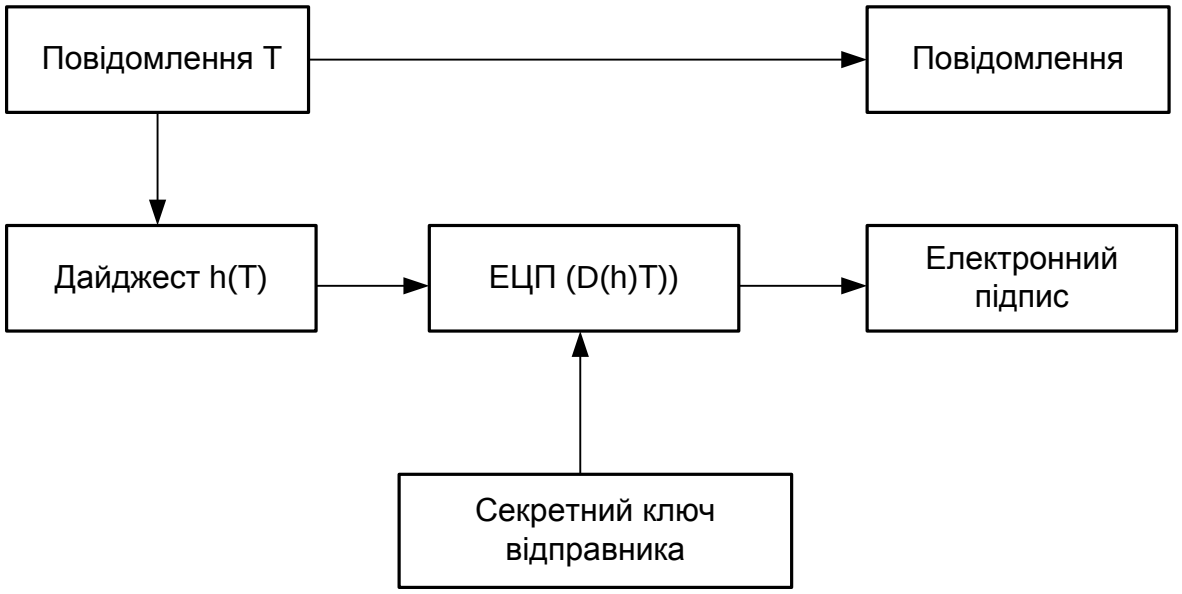


Рис. 9.5. Вироблення електронного цифрового підпису

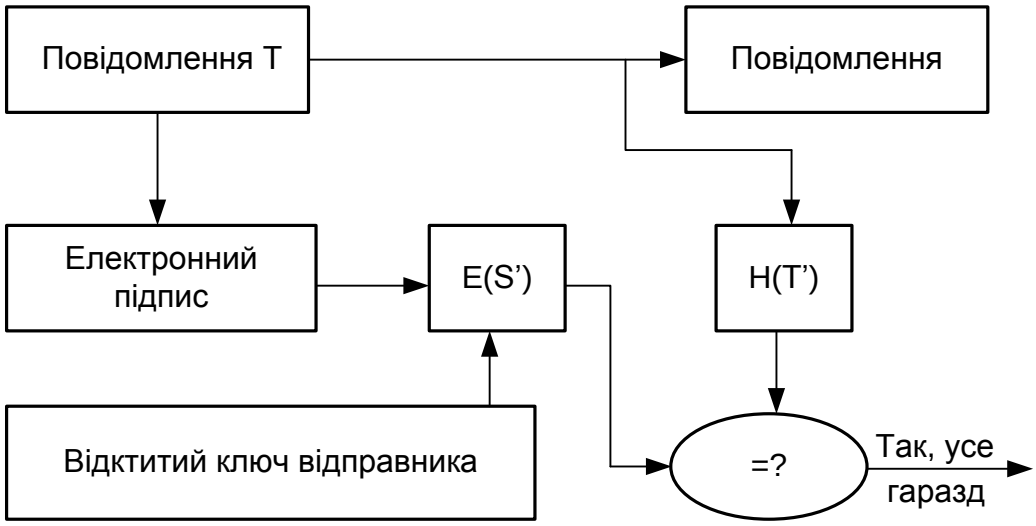


Рис. 9.6. Перевірка електронного цифрового

Із рівності

E(S')=h(T')

витікає, що S' =D(h(T)) (для доказу досить застосувати до обох частин перетворення Б і викреслити в лівій частині тотожне перетворення D(E).

Таким чином, електронний цифровий підпис захищає цілісність повідомлення й засвідчує особистість відправника, тобто захищає цілісність джерела даних та є основою безвідмовності

9.6. Цифрові сертифікати

При використанні асиметричних методів шифрування (і, зокрема, електронного цифрового підпису) необхідно мати гарантію дійсності пари (ім'я користувача, відкритий ключ користувача). Для вирішення цього завдання в специфікаціях X.509 уводяться поняття цифрового сертифіката й центру, що засвідчує.

Центр, що засвідчує, - це компонент глобальної служби каталогів, відповідальний за керування криптографічними ключами користувачів. Відкриті ключі й інша інформація про користувачів зберігається центрами, що засвідчують, у вигляді цифрових сертифікатів, що мають наступну структуру:

* порядковий номер сертифіката;
* ідентифікатор алгоритму електронного підпису;
* ім'я центру, що засвідчує;
* строк придатності;
* ім'я власника сертифіката (ім'я користувача, якому належить сертифікат);
* відкриті ключі власника сертифіката (ключів може бути декілька);
* ідентифікатори алгоритмів, асоційованих з відкритими ключами власника сертифіката;
* електронний підпис, згенерований з використанням секретного ключа центру, що засвідчує (підписується результат хешування всієї інформації, що зберігається в сертифікаті).

Цифрові сертифікати мають наступні властивості:

* будь-який користувач, що знає відкритий ключ центру, що засвідчує, може довідатися відкриті ключі інших клієнтів центру й перевірити цілісність сертифіката;
* ніхто, крім центру, що засвідчує, не може модифікувати інформацію про користувача без порушення цілісності сертифіката.

У специфікаціях X.509 не описується конкретна процедура генерації криптографічних ключів і керування ними, однак даються деякі загальні

рекомендації. Зокрема, обумовлюється, що пари ключів можуть породжуватися кожним з наступних способів.

* ключі може генерувати сам користувач. У такому випадку секретний ключ не потрапляє в руки третіх осіб, однак потрібно вирішувати завдання безпечного зв'язку із центром, що засвідчує;
* ключі генерує довірена особа. У такому випадку доводиться вирішувати завдання безпечної доставки секретного ключа власникові й надання довірених даних для створення сертифіката;
* ключі генеруються центром, що засвідчує. У такому випадку залишається тільки завдання безпечної передачі ключів власникові.

Цифрові сертифікати у форматі X.509 версії 3 стали не тільки форматним, але й фактичним стандартом, підтримуваним численними центрами, що засвідчують.

Розділ 10. Екранування, аналіз захищеності

10.1. Екранування. Основні поняття

Формальна постановка завдання екранування полягає в наступному. Нехай є дві сукупності інформаційних систем. Екран - це засіб розмежування доступу клієнтів з однієї сукупності до серверів з іншої сукупності. Екран здійснює свої функції, контролюючи всі інформаційні потоки між двома сукупностями систем (рис. 10.1). Контроль потоків полягає в їхній фільтрації, можливо, з виконанням деяких перетворень.

На наступному рівні деталізації екран (напівпроникну мембрану) зручно представляти як послідовність фільтрів. Кожен з фільтрів, проаналізувавши дані, може затримати (не пропустити) їх, а може й відразу "перекинути" за екран. Крім того, допускається перетворення даних, передача порції даних на наступний фільтр для продовження аналізу або обробка даних від імені адресата й повернення результату відправникові (рис. 10.2)

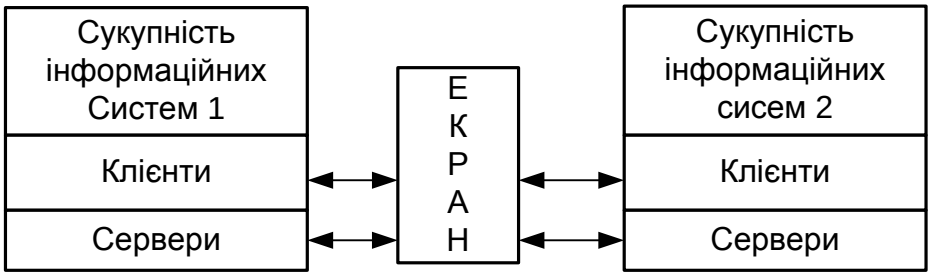
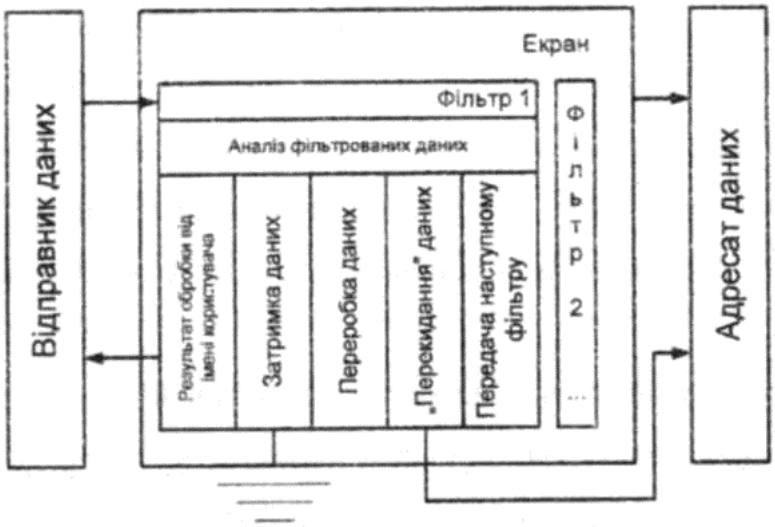


Рис. 10.1. Екран як засіб розмежування доступу



Мал. 10.2. Екран як послідовність фільтрів

Крім функцій розмежування доступу, екрани здійснюють протоколювання обміну інформацією.

Звичайно, екран не є симетричним, для нього визначені поняття "усередині" й "зовні". При цьому завдання екранування формулюється як захист внутрішньої області від потенційно ворожої зовнішньої. Так, міжмережеві екрани (МЕ) (запропонований автором переклад англійського терміна firewall) найчастіше встановлюють для захисту корпоративної мережі організації, що має вихід в Internet (див. наступний розділ).

Екранування допомагає підтримувати доступність сервісів внутрішньої області, зменшуючи або взагалі ліквідуючи навантаження, викликане зовнішньою

активністю. Зменшується уразливість внутрішніх сервісів безпеки, оскільки спочатку зловмисник повинен перебороти екран, де захисні механізми сконфігуровані особливо ретельно. Крім того, екрануюча система, на відміну від універсальної, може будуватися більш просто й, отже, більш безпечним чином. Екранування дає можливість контролювати також інформаційні потоки, спрямовані в зовнішню область, що сприяє підтримці режиму конфіденційності в **ІС** організації. Підкреслимо, що екранування може використовуватися як сервіс безпеки не тільки в мережевому, але й у будь-якому іншому середовищі, де відбувається обмін повідомленнями. Найважливіший приклад подібного середовища - об'єктно-орієнтозані програмні системи, коли для активізації методів об'єктів виконується (принаймні, у концептуальному плані) передача повідомлень. Ймовірно, що в майбутніх об'єктно-орієнтованих середовищах екранування стане одним з найважливіших інструментів розмежування доступу до об'єктів.

Екранування може бути частковим, захищаючи певні інформаційні сервіси. Екранування електронної пошти описано в статті "Контроль над корпоративною електронною поштою: система "Дозор-Джет"" (Jet Info, 2002, 5).

Обмежуючий інтерфейс також можна розглядати як різновид екранування. На невидимий об'єкт важко нападати, особливо за допомогою фіксованого набору засобів. У цьому змісті Web-інтерфейс має природний захист, особливо в тому випадку, коли гіпертекстові документи формуються динамічно. Кожен користувач бачить лише те, що йому належить бачити. Можна провести аналогію між динамічно формованими гіпертекстовими документами й поданнями в реляційних базах даних, з тим істотним застереженням, що у випадку Web можливості істотно ширші.

Екрануюча роль Web-сервісу наочно проявляється й тоді, коли цей сервіс здійснює посередницькі (точніше, інтегруючі) функції при доступі до інших ресурсів, наприклад до таблиць бази даних. Тут не тільки контролюються потоки запитів, але й ховається реальна організація даних.

10.2. Архітектурні аспекти

Боротися з загрозами, притаманними мережевому середовищу, засобами універсальних операційних систем не надається можливим. Універсальна ОС -це величезна програма, що напевно містить, крім явних помилок, деякі особливості, які можуть бути використані для нелегального одержання привілеїв. Сучасна технологія програмування не дозволяє зробити надто великі програми безпечними. Крім того, адміністратор, що має справу зі складною системою, далеко не завжди в змозі врахувати всі наслідки вироблених змін.

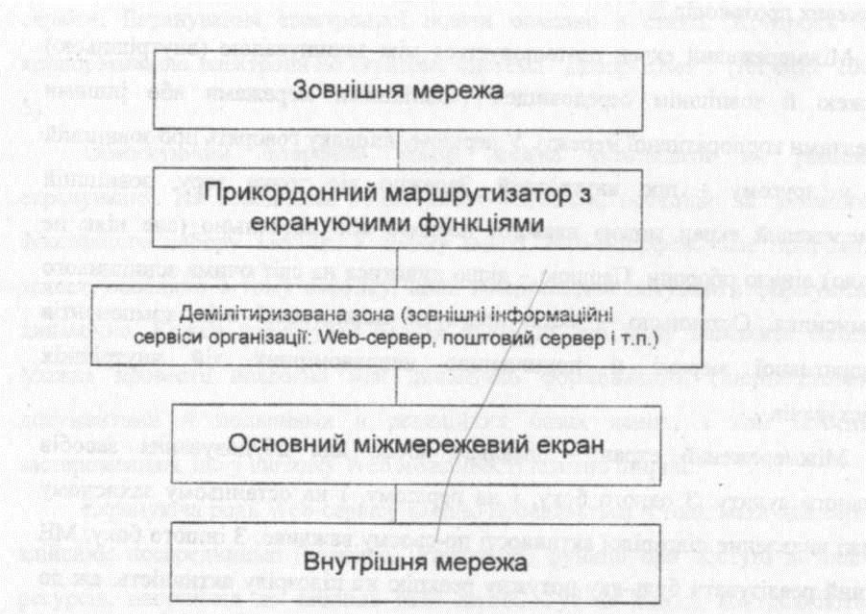
Нарешті, в універсальної багатокористувацької системі проломи у безпеці постійно створюються самими користувачами (слабкі й/або рідко змінювані паролі, невдало встановлені права доступу, залишений без догляду термінал і т.п.). Єдиний перспективний шлях пов'язаний з розробкою спеціалізованих сервісів безпеки, які в силу своєї простоти допускають формальну або неформальну верифікацію. Міжмережевий екран саме і є таким засобом, що допускає подальшу декомпозицію, пов'язану з обслуговуванням різних мережевих протоколів.

Міжмережевий екран розташовується між захищуваною (внутрішньою) мережею й зовнішнім середовищем (зовнішніми мережами або іншими сегментами корпоративної мережі). У першому випадку говорять про зовнішній МЕ, у другому - про внутрішній. Залежно від точки зору, зовнішній міжмережевий екран можна вважати першою або останньою (але ніяк не єдиною) лінією оборони. Першою - якщо дивитися на світ очима зовнішнього зловмисника. Останньою - якщо прагнути захищеності всіх компонентів корпоративної мережі й припиненню неправомірних дій внутрішніх користувачів.

Міжмережевий екран - ідеальне місце для вбудовування засобів активного аудиту. З одного боку, і на першому, і на останньому захисному рубежі виявлення підозрілої активності гіо-своєму важливо. З іншого боку, МЕ здатний реалізувати будь-яку потужну реакцію на підозрілу активність, аж до розриву зв'язку із зовнішнім середовищем. Щоправда, потрібно усвідомлювати те, що з'єднання двох сервісів безпеки в принципі може створити пролом, що сприяє атакам на доступність.

На між мережевий екран доцільно покласти ідентифікацію/аутентифікацію зовнішніх користувачів, що мають потребу у доступі до корпоративних ресурсів (з підтримкою концепції єдиного входу в мережу).

У силу принципів ешелонованості оборони для захисту зовнішніх підключень звичайно використовується двокомпонентне екранування (див. рис. 10.3). Первинна фільтрація (наприклад, блокування пакетів керуючого протоколу SNMP, небезпечного атаками на доступність, або пакетів з певними IP-адресами, включеними в "чорний список") здійснюється граничним маршрутизатором (див. також наступний розділ), за яким розташовується так звана демілітаризована зона (мережа з помірною довірою безпеки, куди виносяться зовнішні інформаційні сервіси організації - Web, електронна пошта й т.п.) і основний МЕ, що захищає внутрішню частину корпоративної мережі.



Мал. 10.3. Двокомпонентне екранування з демілітаризованою зоною

Теоретично міжмережевий екран (особливо внутрішній) повинен бути багатопротокольним, однак на практиці домінування сімейства протоколів TCP/IP настільки велике, що підтримка інших протоколів уявляється надмірністю, шкідливою для безпеки (чим сервіс складніший, тим він більш уразливий).Загалом кажучи, і зовнішній, і внутрішній міжмережеві екрани можуть стати вузьким

місцем, оскільки обсяг мережевого трафіка має тенденцію швидкого зростання. Один з підходів до вирішення цієї проблеми припускає розбивку МЕ на кілька апаратних частин й організацію спеціалізованих серверів-посередників. Основний міжмережевий екран може проводити грубу класифікацію вхідного трафіка за видами і передоручати фільтрацію відповідним посередникам (наприклад, посередникові, що аналізує HTTP-трафік). Вихідний трафік спочатку обробляється серзером-посередником, що може виконувати й функціонально корисні дії, такі як кешування сторінок зовнішніх Web-серверів, що знижує навантаження на мережу взагалі й основний МЕ зокрема.

Ситуації, коли корпоративна мережа містить лише один зовнішній канат, є скоріше виключенням, ніж правилом. З іншого боку, типова ситуація, при якій корпоративна мережа складається з декількох територіально рознесених сегментів, кожний з яких підключений до Internet. У цьому випадку кожне підключення повинне захищатися своїм екраном. Можна вважати, що корпоративний зовнішній міжмережевий екран є складовим, і потрібно вирішувати завдання узгодженого адміністрування (керування й аудиту) всіх компонентів.

Протилежністю складовим корпоративних МЕ (або їхнім компонентам) є персональні міжмережеві екрани й персональні екрануючі пристрої. Перші є програмними продуктами, які встановлюються на персональні комп'ютери й захищають тільки їх. Другі реалізуються на окремих пристроях і захищають невелику локальну мережу, таку як мережа домашнього офісу.

При розгортанні міжмережевих екранів варто дотримуватися розглянутих нами раніше принципів архітектурної безпеки, у першу чергу подбавши про простоту й керованість, про ешелонованість оборони, а також про неможливість переходу в небезпечний стан. Крім того, варто брати до уваги не тільки зовнішні, але й внутрішні загрози.

10.3. Класифікація міжмережевих екранів

При розгляді будь-якого питання, що стосується мережевих технологій, основою слугує еемирівнева еталонна модель ІБО/ОБІ. Міжмережеві екрани також доцільно класифікувати за рівнем фільтрації - канальному, мережевому,

транспортному або прикладному. Відповідно, можна говорити про екрануючі концентратори (мости, комутатори) (рівень 2), маршрутизатори (рівень 3), про транспортне екранування (рівень 4) і про прикладні екрани (рівень 7). Існують також комплексні екрани, що аналізують інформацію на декількох рівнях.

Фільтрація інформаційних потоків здійснюється міжмережевими екранами на основі набору правил, що є вираженням мережевих аспектів політики безпеки організації. У цих правилах, крім Інформації, яка зберігається у фільтрованих потоках, можуть фігурувати дані, отримані з оточення, наприклад, поточний час, кількість активних з'єднань, порт, через який надійшов мережевий запит і т.д. Таким чином, у міжмережевих екранах використовується дуже потужний логічний підхід до розмежування доступу.

Можливості міжмережєвого екрана безпосередньо визначаються тим, яка інформація може використовуватися в правилах фільтрації і якою може бути потужність наборів правил. Загалом кажучи, чим вищим є рівень моделі 180/081, на якому функціонує МЕ, тим більш змістовна інформація йому доступна й, отже, тим тонше й надійніше він може бути сконфігурованим.

Екрануючі маршрутизатори (і концентратори) мають справу з окремими пакетами даних, тому іноді їх називають пакетними фільтрами. Рішення про те, пропустити або затримати дані, приймаються для кожного пакета незалежно, на підставі аналізу адрес й інших полів заголовків мережевого (канального) і, можливо, транспортного рівнів. Ще один важливий компонент аналізованої інформації - порт, через який надійшов пакет.

Екрануючі концентратори є засобом не стільки розмежування доступу, скільки оптимізації роботи локальної мережі за рахунок організації так званих віртуальних локальних мереж. Останні можна вважати важливим результатом застосування внутрішнього міжмережєвого екранування.

Сучасні маршрутизатори дозволяють зв'язувати з кожним портом кілька десятків правил і фільтрувати пакети як на вході, так і на виході. У принципі, як пакетний фільтр може використовуватися й універсальний комп'ютер, обладнаний декількома мережевими картами.

Основні переваги екрануючих маршрутизаторів, - доступна ціна (на межі мереж маршрутизатор потрібний практично завжди, питання лише в тому, як задіяти його екрануючі можливості) і прозорість для більш високих рівнів моделі OSI. Основний недолік - обмеженість аналізованої інформації та, як наслідок, відносна слабкість забезпечуваного захисту.

Транспортне екранування дозволяє контролювати процес установлення віртуальних з'єднань і передачу інформації з них. З погляду реалізації екрануючий транспорт, являє собою досить просту, а виходить, надійну програму.

У порівнянні з пакетним,, фільтрами, транспортне екранування володіє більшою кількістю інформації, тому відповідний МЕ може здійснювати більш тонкий контроль за віртуальними з'єднаннями (наприклад, він здатний відслідковувати кількість переданої інформації й розривати з'єднання після перевищення певного порогу, перешкоджаючи тим самим несанкціонованому експорту інформації). Аналогічно, можливе нагромадження більш змістовної реєстраційної інформації. Головний недолік - звуження області застосування, оскільки поза контролем залишаються датаграмні протоколи. Звичайно, транспортне екранування застосовують у сполученні з іншими підходами, як важливий додатковий елемент.

Міжмережевий екран, що функціонує на прикладному рівні, здатний забезпечити найбільш надійний захист. *Як* правило, подібний МЕ являє собою універсальний комп'ютер, на якому функціонують екрануючі агенти, інтерпретуючи протоколи прикладного рівня (HTTP, FTP, SMTP, telnet і т.д.) у тому ступені, який необхідний для забезпечення безпеки.

При використанні прикладних МЕ, крім фільтрації, реалізується ще один найважливіший аспект екранування. Суб'єкти із зовнішньої мережі бачать тільки шлюзовий комп ютер; відповідно, їм доступна тільки та інформація про внутрішню мережу, як)' він уважає за потрібне експортувати. Прикладний МЕ насправді екранує, тобто закриває, внутрішню мережу від зовнішнього світу. У той же час, суб'єктам внутрішньої мережі здається, що вони напряму спілкуються з об'єктами зовнішнього світу. Недолік прикладних МЕ -відсутність повної

прозорості, що вимагає спеціальних дій для підтримки кожного прикладного протоколу.

Якщо організація має у своєму розпорядженні вихідні тексти прикладного МЕ й у стані ці тексти модифікувати, перед нею відкриваються надзвичайно широкі можливості по налаштуванню екрана з урахуванням власних потреб. Справа в тому, що при розробці систем клієнт/сервер у багатоланковій архітектурі з'являються специфічні прикладні протоколи, які потребують захисту не менше стандартних. Підхід, заснований на використанні екрануючих агентів, дозволяє побудувати такий захист, не знижуючи безпеку й ефективності інших додатків і не ускладнюючи структуру зв'язків у міжмережевому екрані.

Комплексні міжмережеві екрани, що охоплюють рівні від мережевого до прикладного, поєднують у собі кращі властивості "однорівневих" МЕ різних видів. Захисні функції виконуються комплексними МЕ прозорим для додатків чином, не вимагаючи внесення будь-яких змін ні в існуюче програмне забезпечення, ні в дії, що стали для користувачів звичними.

Комплексність МЕ може досягатися різними способами: "знизу догори", від мережевого рівня через нагромадження контексту до прикладного рівня, або "зверху вниз", за допомогою доповнення прикладного МЕ механізмами транспортного й мережевого рівнів.

Крім виразних можливостей і припустимої кількості правил, якість міжмережєвого екрана визначається ще двома дуже важливими характеристиками - простотою використання й власною захищеністю. У плані простоти використання першорядне значення мають наочний інтерфейс при визначенні правил фільтрації й можливість централізованого адміністрування складених конфігурацій. У свою чергу, в останньому аспекті хотілося б виділити засоби централізованого завантаження правил фільтрації й перевірки набору правил на несуперечність. Важливий і централізований збір і аналіз реєстраційної інформації, а також одержання сигначів про спроби виконання дій, заборонених політикою безпеки.

Власна захищеність міжмережєвого екрана забезпечується тими ж засобами, що й захищеність універсальних систем. Мається на увазі фізичний захист,

ідентифікація й аутентифікація, розмежування доступу, контроль цілісності, протоколювання й аудит. При виконанні централізованого адміністрування варто також подбати про захист інформації від пасивного й активного прослуховування мережі, тобто забезпечити її (інформації) цілісність і конфіденційність. Надто важливо оперативне накладення латок, що ліквідують виявлені уразливі місця ME. Хотілося б підкреслити, що природа екранування як сервісу безпеки дуже глибока. Крім блокування потоків даних, що порушують політику безпеки, міжмережевий екран може приховувати інформацію про захищувані мережі, що, тим самим перешкоджають діям потенційних зловмисників. Потужним методом приховування інформації є трансляція "внутрішніх" мережевих адрес, шо попутно вирішує проблему розширення адресного простору, виділеного організації.

Відзначимо також наступні додаткові можливості міжмережевих екранів:

* контроль інформаційного наповнення (антивірусний контроль "на ходу", верифікація Iaya-апплетів, виявлення ключових слів в електронних повідомленнях і т.п.);
* виконання функцій ПЗ проміжного шару. Особливо важливим уявляється останній з перерахованих аспектів. ПЗ

проміжного шару, як і традиційні міжмережеві екрани прикладного рівня, приховує інформацію про надавані послуги. За рахунок цього воно може виконувати такі функції, як маршрутизація запитів і балансування навантаження. Уявляється цілком природним, щоб ці можливості були реалізовані в межах міжмережевого екрана. Це істотно спрощує дії по забезпеченню високої доступності експортованих сервісів і дозволяє здійснювати перемикання на резервні потужності прозорим для зовнішніх користувачів чином. У результаті до послуг, які традиційно надаються міжмережевими екранами, додається підтримка високої доступності мережевих сервісів.

10.4.Аналіз захищеності

Сервіс аналізу захищеності призначений для виявлення уразливих місць із метою їхньої оперативної ліквідації. Сам по собі цей сервіс ні від чого не захищає, але допомагає виявити (і усунути) пробіли в захисті раніше, ніж їх зможе

використати зловмисник. У першу чергу, маються на увазі не архітектурні (їх ліквідувати складно), а "оперативні" проломи, що з'явилися в результаті помилок адміністрування або через неуважність до відновлення версій програмного забезпечення.

Системи аналізу захищеності (названі також сканерами захищеності), як і розглянуті вище засоби активного аудиту, засновані на нагромадженні й використанні знань. У цьому випадку маються на увазі знання про прогалини в захисті: про те, як їх шукати, наскільки вони серйозні і як їх усувати.

Відповідно, ядром таких систем є база уразливих місць, що визначає доступний діапазон можливостей і вимагає практично постійної актуалізації.

У принципі, можуть виявлятися проломи найрізноманітнішої природи: наявність шкідливого ПЗ (зокрема, вірусів), слабкі паролі користувачів, невдало сконфігуровані операційні системи, небезпечні мережеві сервіси, невстановлені латки, уразливості в додатках і т.д. Однак найбільш ефективними є мережеві сканери (очевидно, у силу домінування сімейства протоколів TCPЯP), а також антивірусні засоби. Антивірусний захист ми зараховуємо до засобів аналізу захищеності, не вважаючи її окремим сервісом безпеки.

Сканери можуть виявляти уразливі місця як шляхом пасивного аналізу, тобто вивчення конфігураційних файлів, задіяних портів і т.п., так і шляхом імітації дій атакуючого. Деякі знайдені уразливі місця можуть усуватися автоматично (наприклад, лікування заражених файлів), про інші повідомляється адміністраторові.

Системи аналізу захищеності містять у собі традиційний "технологічний цукор": автовиявленням компонентів аналізованої ІС і графічним інтерфейсом, який допомагає ефективно працювати із протоколом сканування.

Контроль, забезпечуваний системами аналізу захищеності, носить реактивний, запізнілий характер, він не захищає від нових атак, однак варто пам'ятати, що оборона повинна бути ешелонованою, і в якості одного з рубежів, контроль захищеності цілком адекватний. Відзначимо також, що переважна більшість атак носить рутинний характер; вони можливі тільки тому, що відомі проломи в захисті роками залишаються неусунутими.

Розділ 11. Забезпечення високої доступності

11.1. Доступність. Основні поняття

Інформаційна система надає своїм користувачам певний набір послуг (сервісів). Говорять, що забезпечено потрібний рівень доступності цих сервісів, якщо наступні показники перебувають у заданих межах:

Ефективність послуг. Ефективність послуги визначається в термінах максимального часу обслуговування запиту, кількості підтримуваних користувачів і т.п. Потрібно, щоб ефективність не знижувалася нижче заздалегідь установленої межі.

Час недоступності. Якщо ефективність інформаційної послуги не задовольняє накладеним обмеженням, послуга вважається недоступною. Потрібно, щоб максимальна тривалість періоду недоступності й сумарний час недоступності за деякий період (місяць, рік) не перевищували заздалегідь заданих меж.

По суті, потрібно, щоб інформаційна система майже завжди працювала з потрібною ефективністю. Для деяких критично важливих систем (наприклад, систем керування) час недоступності повинен бути нульовим, без усяких "майже". У такому випадку говорять про ймовірності виникнення ситуації недоступності й вимагають, щоб ця ймовірність не перевищувала заданої величини. Для рішення даного завдання створювалися й створюються спеціальні відмовостійкі системи, вартість яких, як правило, досить висока.

До переважної більшості комерційних систем пред'являються менш суворі вимоги, однак сучасне ділове життя й тут накладає досить жорсткі обмеження, коли кількість користувачів, що обслуговують, може вимірюватися тисячами, час відповіді не повинен перевищувати декількох секунд, а час недоступності - декількох годин на рік.

Завдання забезпечення високої доступності необхідно вирішувати для сучасних конфігурацій, побудованих у технології клієнт/сервер. Це означає, що в захисті страждає весь ланцюжок - від користувачів (можливо, вилучених) до критично важливих серверів (у тому числі серверів безпеки).

Основні загрози доступності були розглянуті нами раніше.

Тому, під відмовою розуміється подія, що полягає в порушенні працездатності виробу. У контексті цього виріб - це інформаційна система або її компонент. У найпростішому випадку можна вважати, що відмова будь-якого компонента складеного виробу призводить до загальної відмови, а розподіл відмов у часі являє собою простий пуассонівський потік подій. У такому випадку вводять поняття інтенсивності відмов і середнього часу опрацювання на відмову, які зв'язані між собою співвідношенням

$$T_i = \frac{1}{\lambda_i}$$

де і - номер компонента, **λ** - інтенсивність відмов, Т, - середній час опрацювання на відмову.

Інтенсивності відмов незалежних компонентів додаються:

$$\lambda = \lambda_1 + ... + \lambda_n$$

а середній час опрацювання на відмову для складеного виробу задається співвідношенням

$$T = \frac{1}{\lambda}$$

Уже ці найпростіші вирази показують, що якщо існує компонент, інтенсивність відмов якого є набагато більше, ніж в інших, таким чином він визначає середній час опрацювання на відмову всієї інформаційної системи. Це є теоретичним обгрунтуванням принципу першочергового зміцнення найслабшої ланки.

Пуассонівська модель дозволяє обґрунтувати ще одне важливе положення, яке полягає в тому, що емпіричний підхід до побудови систем високої доступності не може бути реалізований за прийнятний час. При традиційному циклі тестування/налагодження програмної системи за оптимістичними оцінками кожне виправлення помилки призводить до експонентного зниження (приблизно на половину десяткового порядку) інтенсивності відмов. Отже, щоб на досвіді переконатися в досягненні необхідного рівня доступності, незалежно від застосованої технології тестування й налагодження, доведеться витратити час, який

практично дорівнює середньому часу опрацювання на відмову. Наприклад, для досягнення середнього часу опрацювання на відмову 10^5 годин буде потрібно більше 10^{45} годин, що складає більше трьох років. Виходить, потрібні інші методи побудови систем високої доступності, методи, ефективність яких доведена аналітично або практично за більш ніж п'ятдесят років розвитку обчислювальної техніки й програмування.

Пуассонівська модель застосовується в тих випадках, коли інформаційна система містить одиничні крапки відмови, тобто компоненти, виведення яких з ладу призводить до відмови всієї системи. Для дослідження систем з резервуванням застосовується інший формалізм.

Відповідно до постановки завдання будемо вважати, що існує кількісна міра ефективності надаваних виробом інформаційних послуг. У такому випадку вводяться поняття показників ефективності окремих елементів та ефективності функціонування всієї складної системи.

Як міру доступності можна прийняти ймовірність прийнятності ефективності послуг, надаваних інформаційною системою, протягом усього розглянутого проміжку часу. Чим більшим запасом ефективності володіє система, тим вищою є її доступність.

При наявності надмірності в конфігурації системи ймовірність того, що в розглянутий проміжок часу ефективність інформаційних сервісів не знизиться нижче допустимої межі, залежить не тільки від імовірності відмови компонентів, але й від часу, протягом якого вони залишаються непрацездатними, оскільки при цьому сумарна ефективність знижується, і кожна наступна відмова може стати фатальною. Щоб максимально збільшити доступність системи, необхідно мінімізувати час непрацездатності кожного компонента. Крім того, варто враховувати, що ремонтні роботи можуть призвести до зниження ефективності або навіть тимчасового відключення працездатних компонентів; такого типу вплив також необхідно мінімізувати.

Декілька термінологічних зауважень. Звичайно в літературі по теорії надійності замість доступності говорять про готовність (у тому числі про високу готовність).

Ми віддали перевагу терміну "доступність", щоб підкреслити, що інформаційний сервіс повинен бути не просто „працездатним", але й бути доступним для своїх користувачів в умовах, коли ситуації недоступності можуть викликатися причинами, які на перший погляд не мають прямого відношення до сервісу (приклад - відсутність консультаційного обслуговування).

Далі, замість часу недоступності зазвичай говорять про коефіцієнт готовності. Нам хотілося б звернути увагу на два показники - тривалість одиничного простою й сумарну тривалість простоїв, тому ми віддали перевагу терміну "час недоступності" як більш ємному.

11.2. Основи заходів забезпечення високої доступності

Основою звходів підвищення доступності є застосування структурованого підходу, що знайшов втілення в об'єктно-орієнтованій методології. Структуризація необхідна по відношенню до всіх І аспектів і складових частин інформаційної системи - від архітектури до адміністративних баз даних, на всіх етапах її життєвого циклу - від ініціації до виведення з експлуатації. Структуризація, важлива сама по собі, є одночасно необхідною умовою практичної реалізації інших заходів підвищення доступності. Тільки маленькі системи можна будувати й експлуатувати як завгодно. У більших системах свої закони, які, як ми вже вказували, програмісти вперше усвідомили більше ЗО років тому.

При розробці заходів забезпечення високої доступності інформаційних сервісів пропонується керуватися наступними архітектурними принципами, що розглядалися раніше:

* апробованість усіх процесів і складових частин інформаційної системи;
* уніфікація процесів і складових частин;
* керованість процесів, контроль стану частин;
* автоматизація процесів;
* модульність архітектури;
* орієнтація на простоту рішень.

Доступність системи в загальному випадку досягається за рахунок застосування трьох груп заходів, спрямованих на підвищення:

* безвідмовності (під пим розуміється мінімізація ймовірності виникнення якої-небудь відмови; це елемент пасивної безпеки, що далі розглядатися не буде);
* відмовостійкості (здатності до нейтралізації відмов, "живучості", тобто здатності зберігати необхідну ефективність, незважаючи на відмови окремих компонентів);
* обслуговування (під обслуговуванням розуміється мінімізація часу простою компонентів, що відмовили, а також негативного впливу ремонтних робіт на ефективність інформаційних сервісів, тобто швидке й безпечне відновлення після відмов).

Головне при розробці й реалізації заходів забезпечення високої доступності - повнота й систематичність. У цьому зв'язку уявляється доцільним скласти (і підтримувати в актуальному стані) карту інформаційної системи організації (на що ми вже звертали увагу), у якій фігурували б усі об'єкти ІС, їхній стан, зв'язки між ними, процеси, асоційовані з об'єктами й зв'язками. За допомогою подібної карти зручно формулювати намічені заходи, контролювати їхнє виконання, аналізувати стан ІС.

11.3. Відмовостійкість і зона ризику

Інформаційну систему можна уявити у вигляді графа сервісів, ребра якого відповідають відношенню "сервіс А безпосередньо використовує сервіс В".

Нехай у результаті здійснення деякої атаки (джерелом якої може бути як людина, так і природне явище) виводиться з ладу підмножина сервісів В| (тобто ці сервіси в результаті нанесених ушкоджень стають непрацездатними). Назвемо Б і зоною поразки.

У зону ризику Б ми будемо включати всі сервіси, ефективність яких при здійсненні атаки знижується нижче припустимої межі. Очевидно, Б і підмножина S. S суворо включає S_b коли є сервіси, безпосередньо не порушені атакою, але критично залежні від уражених, тобто нездатні перемкнутися на використання еквівалентних послуг або в силу відсутності таких, або в силу неможливості доступу до них. Наприклад, зона поразки може зводитися де одного порту концентратора,

що обслуговує критичний сервер, а зона ризику охоплює всі робочі місця користувачів сервера.

Щоб система не містила одиночних крапок відмови, тобто залишалася "живучою" при реалізації кожної з розглянутих загроз, жодна зона ризику не повинна містити в собі надавані послуги. Нейтралізацію відмов потрібно виконувати усередині системи, непомітно для користувачів, за рахунок розміщення достатньої кількості надлишкових ресурсів. З іншого боку, природно порівнювати зусилля по забезпеченню "живучості" з розглянутими загрозами. Коли розглядається набір загроз, що відповідають ним, зони поразки можуть виявитися вкладеними, так що "живучість" стосовно більш серйозної загрози автоматично спричиняє й "живучість" у більш простих випадках. Варто враховувати, однак, що звичайно вартість перемикання на резервні ресурси зростає разом зі збільшенням обсягу цих ресурсів. Виходить, що для найбільш ймовірних загроз доцільно мінімізувати зону ризику, навіть якщо передбачено нейтралізацію всеохоплюючої загрози. Немає сенсу перемикатися на резервний обчислювальний центр тільки тому, що в одного із серверів вийшов з ладу блок живлення.

Зону ризику можна трактувати не лише як сукупність ресурсів, але і як частину простору, яка використовується під час реалізації загрози. У такому випадку, як правило, чим більшою відстань дублюючого ресурсу від межі зони ризику, тим вищою вартість його підтримки, оскільки збільшується довжина ліній зв'язку, час переїзду персоналу й т.п. Це ще одне твердження на користь адекватної протидії загрозам, яке варто брати до уваги при розміщенні надлишкових ресурсів й, зокрема, при організації резервних центрів.

Уведемо ще одне поняття. Назвемо зоною нейтралізації загрози сукупність ресурсів, залучених у нейтралізацію відмови, що з'явилися внаслідок реалізації загрози. Маються на увазі ресурси, режим роботи яких у випадку відмови змінюється. Очевидно, зона ризику є підмножиною зони нейтралізації. Чим менше різниця між ними, тим економічним є даний механізм нейтралізації.
Усе, що перебуває поза зоною нейтралізації, відмови "не відчуває" і може трактувати внутрішність цієї зони як безвідмовну. Таким чином, в ієрархічно

організованій системі межа між "живучістю" й обслуговуванням, з одного боку, і безвідмовністю, з іншого боку, відносна. Доцільно конструювати цілісну інформаційну систему з компонентів, які на верхньому рівні можна вважати безвідмовними, а питання "живучості" й обслуговування вирішувати в межах кожного компонента.

11.4. Забезпечення відмовостійкості

Основним засобом підвищення "живучості" є внесення надмірності в конфігурацію апаратних і програмних засобів, підтримуючої інфраструктури й персоналу, резервування технічних засобів і тиражування інформаційних ресурсів (програм і даних). Заходи щодо забезпечення відмовостійкості можна розділити на локальні й розподілені. Локальні заходи спрямовані на досягнення "живучості" окремих комп'ютерних систем або їх апаратних і програмних компонентів (у першу чергу з метою нейтралізації внутрішніх відмов ІС). Типові приклади подібних заходів - використання кластернйх конфігурацій як платформа критичних серверів або "гаряче" резервування активного мережевого устаткування з автоматичним перемиканням на резерв.

Якщо в сукупність розглянутих ризиків входять серйозні аварії підтримуючої інфраструктури, що призводять до виходу з ладу виробничого майданчика організації, варто передбачити розподілені заходи забезпечення живучості, такі як створення або оренда резервного обчислювального центру. При цьому, крім дублювання й/або тиражування ресурсів, необхідно передбачити засоби автоматичного або швидкого ручного переконфігурування компонентів ІС, щоб забезпечити перемикання з основної майданчика на резервну.

Апаратура - відносно статична складова, однак було б помилкою повністю відмовляти їй у динамічності. У більшості організацій інформаційні системи перебувають у постійному розвитку, тому протягом усього життєвого циклу ІС варто співвідносити всі зміни з необхідністю забезпечення "живучості", не забувати "тиражувати" нові й модифіковані компоненти.

Програми й дані більш динамічні, ніж апаратура, і резервуватися вони можуть постійно, при кожній зміні, після завершення деякої логічно замкнутої групи змін або після закінчення певного часу.

Резервування програм і даних може виконуватися багатьма способами -за рахунок дзеркалювання дисків, резервного копіювання й відновлення, реплікації баз даних і т.п. Будемо використовувати для всіх перерахованих способів термін "тиражування". Виділимо наступні класи тиражування:

Симетричне/асиметричне. Тиражування називається симетричним, якщо всі сервери, що надають даний сервіс, можуть змінювати приналежну ним інформацію й передавати зміни іншим серверам. У протилежному випадку тиражування називається асиметричним.

Синхронне/асинхронне. Тиражування називається синхронним, якщо зміна передається всім екземплярам сервісу в межах однієї розподіленої транзакції. У протилежному випадку тиражування називається асинхронним.

Здійснюване засобами сервісу, що зберігає інформацію/зовнішніми засобами.

Розглянемо, яким способам тиражування варто надавати перевагу.

Безумовно, слід віддати перевагу стандартним засобам тиражування, які вбудовані у сервіс.

Асиметричне тиражування теоретично простіше симетричного, тому доцільно вибрати асиметрію.

Складніше вибрати між синхронним й асинхронним тиражуванням. Синхронне ідейно простіше, але його реалізація може бути великоваговою й складною, хоча це внутрішня складність сервісу, невидима для додатків.

Асинхронне тиражування стійкіше до відмов у мережі, воно менше впливає на роботу основного сервісу.

Чим надійніший зв'язок між серверами, залученими в процес тиражування, тим меншим є час, що відводиться на перемикання з основного сервера на резервний, чим жорсткіші вимоги до актуальності інформації, тим більш кращим виявляється синхронне тиражування.

З іншого боку, недоліки асинхронного тиражування можуть компенсуватися процедурними й програмними заходами, спрямованими на контроль цілісності інформації в розподіленій 1С. Сервіси, що входять до складу 1С, у стані забезпечити ведення й зберігання журналів транзакцій, за допомогою яких можна виявляти операції, загублені при перемиканні на резервний сервер. Навіть в умовах нестійкого зв'язку з вилученими філіями організації подібна перевірка у фоновому режимі займе не більше декількох годин, тому асинхронне тиражування може використовуватися практично в будь-яких 1С.

Асинхронне тиражування може розповсюджуватися на сервер, що працює в режимі "гарячого" резерву, можливо, навіть обслуговувати частин}' користувацьких запитів, або на сервер, що працює в режимі "теплого" резерву, коли зміни періодично "накочупичуються", але сам резервний сервер запитів не обслуговує.

Достоїнство "теплого" резервування в тому, що його можна реалізувати, впливаючи на основний сервер. Цей вплив взагалі може бути зведений до нуля, якщо асинхронне тиражування здійснюється шляхом передачі інкрементальних копій з основного сервера (резервне копіювання необхідно виконувати в кожному випадку).

Основний недолік "теплого" резерву полягає в тривалому часі включення, що може бути неприйнятним для "важких" серверів, таких як кластерна конфігурація сервера СУБД. Тут необхідно проводити вимір в умовах, близьких до реальних.

Другий недолік "теплого" резерву випливає з небезпеки малих змін. Може виявитися, що в найпотрібніший момент терміновий переклад резерву в штатний режим неможливий.

З огляду на наведені міркування, потрібно в першу чергу розглядати можливість "гарячого" резервування, або ретельно контролювати використання "теплого" резерву й регулярно (не рідше одного разу на тиждень) проводити тестові перемикання резерву в "гарячий" режим.

11.5. Програмне забезпечення проміжного шару

За допомогою програмного забезпечення проміжного шару (ПЗ ПШ) можна для довільних прикладних сервісів домогтися високої "живучості" з повністю прозорим для користувачів перемиканням на резервні потужності.

Перерахуємо основні достоїнства ПЗ ПШ, істотні для забезпечення високої доступності.

* ПЗ ПШ зменшує складність створення розподілених систем. Подібне ПЗ покладає на себе частину функцій, які в локальному випадку виконують операційні системи;
* ПЗ ПШ покладає на себе маршрутизацію запитів, дозволяючи тим самим забезпечити "живучість" прозорим для користувачів чином;
* ПЗ ПШ здійснює балансування завантаження обчислювальних потужностей, що також сприяє підвищенню доступності даних;
* ПЗ ПШ у стані здійснювати тиражування будь-якої інформації, а не тільки вмісту баз даних. Отже, будь-який додаток можна зробити стійким до відмов серверів;
* ПЗ ПШ у стані відслідковувати стан додатків і при необхідності тиражувати й перезапускати програми, що гарантує "живучість" програмних систем;
* ПЗ ПШ дає можливість прозорим для користувачів чином виконувати переконфігурування (і, зокрема, нарощування) серверних компонентів, що дозволяє масштабувати систему, зберігаючи інвестиції в прикладних системах. Стабільність прикладних систем - важливий фактор підвищення доступності даних.

Раніше ми згадували про достоїнства використання ПЗ ТИП у межах міжмережевих екранів, які в такому випадку стають елементом забезпечення відмовостійкості надаваних інформаційних сервісів.

11.6. Забезпечення обслуговування

Заходи щодо забезпечення обслуговування спрямовані на зниження строків діагностування й усунення відмов та їхніх наслідків.

Для забезпечення обслуговування рекомендується дотримуватися наступних архітектурних принципів:

* орієнтація на побудову інформаційної системи з уніфікованих компонентів з метою спрощення заміни частин, що відмовили;
* орієнтація на рішення модульної структури з можливістю автоматичного виявлення відмов, динамічного переконфігурування апаратних і програмних засобів і заміни компонентів, що відмовили, в "гарячому" режимі.

Динамічне переконфігурування переслідує дві основні цілі:

* ізоляція компонентів, що відмовили;
* збереження працездатності сервісів.

Ізольовані компоненти утворять зону поразки реалізованої загрози. Чим менше відповідна зона ризику, тим вищим є обслуговування сервісів. Так. при відмовах блоків живлення, вентиляторів й/або дисків у сучасних серверах зона ризику обмежується компонентом, що відмовив; при відмовах процесорних модулів весь сервер може зажадати перезавантаження (що здатне викликати подальше розширення зони ризику). Очевидно, в ідеальному випадку зони поразки й ризику збігаються, і сучасні сервери й активне мережеве устаткування, а також програмне забезпечення провідних виробників досить близькі до цього ідеалу.

Можливість програмування реакції на відмову також підвищує обслуговування систем. Кожна організація може вибрати свою стратегію реагування на відмови тих або інших апаратних і програмних компонентів й автоматизувати цю реакцію. Так, у найпростішому випадку можливе відправлення повідомлення системному адміністратору, щоб прискорити початок ремонтних робіт; у більш складному випадку може бути реалізована процедура "м'якого" вимикання (перемикання) сервісу, щоб спростити обслуговування.

Можливість вилученого виконання адміністративних дій - важливий напрямок підвищення обслуговування, оскільки при цьому прискорюється початок відбудовних заходів, а в ідеалі всі роботи (зазвичай пов'язані з обслуговуванням програмних компонентів) виконуються у вилученому режимі, без переміщення кваліфікованого персоналу, тобто з високою якістю та в найкоротший термін. Для

сучасних систем можливість вилученого адміністрування - стандартна властивість, але важливо подбати про його практичну реалізацію в умовах різнорідності конфігурацій (у першу чергу клієнтських). Централізоване поширення й конфігурування програмного забезпечення, керування компонентами інформаційної системи й діагностування - надійний фундамент технічних заходів підвищення обслуговування.

Істотний аспект підвищення обслуговування - організація консультаційної служби для користувачів (обслуговування користувачів), впровадження програмних систем для роботи цієї служби, забезпечення достатньої пропускної здатності каналів зв'язку з користувачами, у тому числі в режимі пікових навантажень.

Розділ 12. Тунелювання й керування

12.1. Тунелювання

На каш погляд, тунелювання варто розглядати як самостійний сервіс безпеки. Його суть полягає в тому, щоб "упакувати" передану порцію даних, разом зі службовими полями, у новий конверт". Як синоніми терміна "тунелювання" можуть використовуватися "конвертування" й "обгортання".

Тунелювання може застосовуватися для декількох цілей:

* передача через мережу пакетів, що належать протоколу, який у даній мережі не підтримується (наприклад, передача пакетів IPv6 через старі мережі, що підтримують тільки IPv4);
* забезпечення слабкої форми конфіденційності (у першу чергу конфіденційності графіка) за рахунок приховування істиних адрес та іншої службової інформації;
* забезпечення конфіденційності й цілісності переданих даних при використанні разом із криптографічними сервісами.

Тунелювання може застосовуватися як на мережевому, так і на прикладному рівнях. Наприклад, стандартизоване тунелювання для IP і подвійне конвертування для пошти X.400.

На рис. 12.1 Показаний приклад обгортання пакетів IPv6 у формат IPv4.

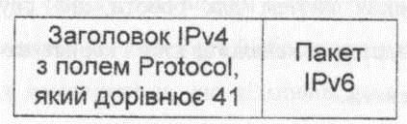


Рис. 12.1. Обгортання пакетів IPv6 у формат IPv4 з метою їх тунелювання через мережі IPv4.

Комбінація тунелювання й шифрування (поряд з необхідною криптографічною інфраструктурою) на виділених шлюзах й екранування на маршрутизаторах постачальників мережевих послуг (для поділу просторів "своїх" та "чужих" мережевих адрес у дусі віртуальних локальних мереж) дозволяє реалізувати такий важливий в сучасних умовах захисний засіб, як віртуальні приватні мережі. Подібні мережі, накладені звичайно поверх Internet, є істотно дешевші й набагато безпечніші, ніж власні мережі організації, побудовані по виділених каналах. Комунікації на всьому їхньому шляху фізично захистити неможливо, тому краще споконвічно виходити із припущення про їхню уразливість і відповідно забезпечувати захист. Сучасні протоколи, спрямовані на підтримку класів обслуговування, допоможуть гарантувати для віртуальних приватних мереж задану пропускну здатність, величину затримок і т.п., ліквідуючи тим самим єдину на сьогодні реальну перевагу власних мереж.

IP-пакет з істинними адресами та відкритим змістом		Міжмере-жевий екран **1** (1-й кінець тунеля)		Обгорнутий зашифро-ваний IP-пакет з адресами кінців тунеля		Міжмере-жевий екран **2** (2-й кінець тунеля)		IP-пакет з істинними адресами та відкритим змістом
Внутрішня мережа 1				Internet				Внутрішня мережа 2

Рис. 12.2. Міжмережеві екрани як крапки реалізації сервісу віртуальних приватних мереж

Кінцями тунелів, що реалізують віртуальні частки мережі, доцільно зробити міжмережеві екрани, що обслуговують підключення організацій до зовнішніх мереж (див. рис. 14.2). У такому випадку тунелювання й шифрування стануть додатковими

перетвореннями, виконуваними в процесі фільтрації мережевого трафіка поряд із трансляцією адрес.

Кінцями тунелів, крім корпоративних міжмережевих екранів, можуть бути мобільні комп'ютери співробітників (точніше, їх персональні МЕ).

12.2. Керування. Основні поняття

Керування можна віднести до числа інфраструктурних сервісів, що забезпечують нормальну роботу компонентів і засобів безпеки. Складність сучасних систем полягає в тому, що без правильно організованого керування вони поступово деградують як у плані ефективності, так й у плані захищеності. Можливий й інший погляд на керування - як на інтегруючу оболонку інформаційних сервісів і сервісів безпеки (у тому числі засобів забезпечення високої доступності), що забезпечує їх нормальне, погоджене функціонування під контролем адміністратора ІС.

Відповідно до стандарту X.700, керування розподіляється на:

* моніторинг компонентів;
* контроль (тобто видачу й реалізацію керуючих впливів):
* координацію роботи компонентів системи. Системи керування повинні:
* дозволяти адміністраторам планувати, організовувати, контролювати й ураховувати використання інформаційних сервісів;
* давати можливість відповідати на зміну вимог;
* забезпечувати передбачуване поводження інформаційних сервісів;
* забезпечувати захист інформації.

Іншими словами, керування мусить бути функціональним, бути результативним, гнучким й інформаціно безпечним.

У X.700 виділяється п'ять функціональних областей керування:

* керування конфігурацією (установка параметрів для нормального функціонування, запуск і зупинка компонентів, збір інформації про поточний стан системи, прийом повідомлень про істотні зміни в умовах функціонування, зміна конфігурації системи);

* керування відмовами (виявлення відмов, їхня ізоляція й відновлення працездатності системи);
* керування продуктивністю (збір і аналіз статистичної інформації, визначення продуктивності системи в штатних і позаштатних умовах, зміна режиму роботи системи);
* керування безпекою (реалізація політики безпеки шляхом створення, видалення й зміни сервісів і механізмів безпеки, поширення відповідної інформації й реагування на інциденти);
* керування обліковою інформацією (тобто стягнення плати за користування ресурсами).

У стандартах сімейства X.700 описується модель керування, здатна забезпечити досягнення поставлених цілей. Уводиться поняття керованого об'єкта як сукупності характеристик компонента системи, важливих з погляду керування. До таких характеристик відносяться:

* атрибути об'єкта;
* припустимі операції;
* повідомлення, які об'єкт може генерувати;
* зв'язки з іншими керованими об'єктами.

Відповідно до рекомендацій X.701, системи керування розподіленими ІС будуються в архітектурі менеджер/агент. Агент (як програмна модель керованого об'єкта) виконує керуючі дії й породжує (при виникненні певних подій) повідомлення від його імені. У свою чергу, менеджер дає агентам команди на керуючі впливи й одержує повідомлення.

Ієрархія взаємодіючих менеджерів та агентів може мати кілька рівнів. При цьому елементи проміжних рівнів відіграють двояку роль: стосовно розміщених вище елементів вони є агентами, а до розміщених нижче -менеджерами. Багаторівнева архітектура менеджер/агент - ключ до розподіленого, масштабованого керування більшими системами.

Логічно пов'язаною з багаторівневою архітектурою є концепція довіреного (або делегованого) керування. При довіреному керуванні менеджер проміжного рівня

може управляти об'єктами, що використовують власні протоколи, у той час як "нагорі" керуються винятково стандартними засобами.

Обов'язковим елементом при будь-якій кількості архітектурних рівнів є керуюча консоль.

З погляду вивчення можливостей систем керування варто враховувати поділ, уведений у X.701. Керування підрозділяється на наступні аспекти:

* інформаційний (атрибути, операції й повідомлення керованих об'єктів);
* функціональний (керуюча дія й необхідна для неї інформація);
* комунікаційний (обмін керуючою інформацією);
* організаційний (розбивка на області керування).

Ключову роль відіграє модель керуючої інформації. Вона описується рекомендаціями X.720. Модель є об'єктно-орієнтованою з підтримкою інкапсуляції й успадкування. Додатково вводиться поняття пакета як сукупності атрибутів, операцій, повідомлень і відповідного поводження.

Клас об'єктів визначається позицією в дереві успадкування, набором включених пакетів і зовнішнім інтерфейсом, тобто видимими зовні атрибутами, операціями, повідомленнями й відповідною повоедінкою.

До числа концептуально важливих можна віднести поняття "проактивного", тобто попереджувального керування. Попереджувальне керування засноване на передбаченні поводження системи на основі поточних даних і раніше накопиченої інформації. Найпростіший приклад подібного керування - подача сигналу про можливі проблеми з диском після серії програмно-нейтралізованих помилок читання/запису. У більш складному випадку певний характер робочого навантаження й дій користувачів може передувати різкому вповільненню роботи системи; адекватним керуючим впливом могло б стати зниження пріоритетів деяких завдань і повідомлення адміністратора про наближення кризи.

12.3.Можливості типових систем

Розвинені системи керування мають, якщо можна так виразитися, двомірну налаштовуваність - на потреби конкретних організацій і на зміни в інформаційних технологіях. Системи керування живуть (принаймні, повинні жити) довго. За цей

час у різних предметних областях адміністрування (наприклад, в області резервного копіювання) напевно з'являться рішення, що перевершать початково закладені в керуючий комплект. Останній повинен уміти еволюціонувати, причому різні його компоненти можуть робити це з різною швидкістю. Ніяка тверда, монолітна система такого не витримає.

Єдиний вихід - наявність каркаса, з якого можна знімати старе й "навішувати" нове, не втрачаючи в ефективності керування.

Каркас як самостійний продукт необхідний для досягнення принаймні наступних цілей:

* згладжування різнорідності керованих інформаційних систем, надання уніфікованих програмних інтерфейсів для швидкої розробки керуючих додатків;
* створення інфраструктури керування, що забезпечує наявність таких властивостей: підтримку розподілених конфігурацій, масштабованість, інформаційна безпека й т.д.;
* надання функціонально корисних універсальних сервісів, таких як планування завдань, генерація звітів і т.п.

Питання про те, що, крім каркаса, повинно входити в систему керування, є досить складним. По-перше, багато систем керування мають мейнфреймове минуле й просто успадкували деяку функціональність, що перестала бути необхідною, По-друге, для багатьох функціональних завдань з'явилися окремі, високоякісні рішення, переважаючі аналогічні по призначенню "штатні" кохмпоненти. Як бачимо, з розвитком об'єктного підходу, багатоплатформність найважливіших сервісів та їхньої взаємної сумісності, системи керування дійсно перетворяться в каркас. Поки ж у них залишається досить важливих областей, а саме:

* керування безпекою;
* керування завантаженням:
* керування подіями;
* керування зберіганням даних;
* керування проблемними ситуаціями;
* генерація звітів.

На рівні інфраструктури присутні рішення ще одного найважливішого функціонального завдання - забезпечення автоматичного виявлення керованих об'єктів, виявлення їхніх характеристик і зв'язків між ними.

Відзначимо, що керування безпекою в сукупності з відповідним програмним інтерфейсом дозволяє реалізувати платформно незалежне розмежування доступу до об'єктів довільної природи й (що дуже важливо) винести функції безпеки із прикладних систем. Щоб з'ясувати, чи дозволений доступ поточною політикою, додатку досить звернутися до менеджера безпеки системи керування.

Менеджер безпеки здійснює ідентифікацію/аутентифікацію користувачів, контроль доступу до ресурсів і протоколювання невдалих спроб доступу. Можна вважати, що менеджер безпеки вбудовується в ядро операційних систем контрольованих елементів ІС, перехоплює відповідні обіги й здійснює свої перевірки перед перевірками, виконуваними ОС, так що він створює ще один захисний рубіж, не скасовуючи, а доповнюючи захист, реалізований засобами ОС.

Розвинені системи керування володіють централізованою базою, у якій зберігається інформація про контрольовану ІС й, зокрема, деяке подання про політику безпеки. Можна вважати, що при кожній спробі доступу виконується перегляд збережених у базі правил, у результаті якого з'ясовується наявність у користувача необхідних прав. Тим самим для ведення єдиної політики безпеки в межах корпоративної інформаційної системи створюється міцний технологічний фундамент.

Зберігання параметрів безпеки в базі даних дає адміністраторам ще одну важливу перевагу - можливість виконання різноманітних запитів. Можна одержати список ресурсів, доступних даному користувачеві, список користувачів, що мають доступ до даного ресурсу й т.п.

Одним з елементів забезпечення високої доступності даних є підсистема автоматичного керування зберіганням даних, що виконує резервне копіювання даних, а також автоматичне відстеження їхнього переміщення між основними й резервними носіями.

Для забезпечення високої доступності інформаційних сервісів використовується керування завантаженням, яке можна підрозділити на керування проходженням завдань і контроль продуктивності. Контроль продуктивності - поняття багатогранне. Сюди входять і оцінка швидкодії комп'ютерів, і аналіз пропускної здатності імереж, і відстеження кількості одночасно підтримуваних користувачів, і час реакції, і нагромадження й аналіз статистики використання ресурсів. Звичайно в розподіленій системі відповідні дані доступні "у принципі", вони поставляються крапковими засобами керування, але проблема отримання цілісної картини, як поточної, так і перспективної, залишається досить складною. Вирішити її здатна тільки система керування корпоративного рівня.

Засоби контролю продуктивності доцільно розбити на дві категорії:

* виявлення випадків неадекватного функціонування компонентів інформаційної системи й автоматичне реагування на ці події;
* аналіз тенденцій зміни продуктивності системи й довгострокове планування.

Для функціонування обох категорій засобів необхідно вибрати відслідковувані параметри і припустимі межі, вихід за які означає "неадекватність функціонування". Після цього, завдання зводиться до виявлення нетипової поведінки компонентів J для чого можуть застосовуватися статистичні методи.

Керування подіями (точніше, повідомленнями про події) - це базовий механізм, що дозволяє контролювати стан інформаційних систем у реальному часі. Системи керування дозволяють класифікувати події й призначати для деяких з них спеціальні процедури обробки. Тим самим реалізується важливий принцип автоматичного реагування.

Очевидно, що завдання контролю продуктивності й керування подіями, так само як і методи їхнього рішення в системах керування, близькі до аналогічних аспектів систем активного аудиту. У наявності ще одне свідчення концептуальної єдності області знань під назвою "інформаційна безпека" і необхідності реалізації цієї єдності на практиці.

Післямова

Розповідати щодо інформаційної безпеки, як і про будь-яку справу, в якій необхідно використовувати поєднання знань та творчого підходу, можна дуже довго.

Тому автори, відносно глибини висвітлення питань з інформаційної безпеки, керувалися таким принципом: гарантованим створенням необхідного мінімуму знань щодо питань та використанням ситуацій та засобів, які найбільш часто зустрічаються на практиці ситуаціями та засобами.

Формування і забезпечення функціонування ефективно діючої системи інформаційної безпеки в державі - складний і багатогранний процес, який потребує значних зусиль усіх гілок влади, вітчизняної науки, керівників усіх рівнів. Вирішення деяких проблем, очевидно, потребує не один рік, але їх вирішення зумовлене необхідністю формування виваженої державної політики забезпечення інформаційної безпеки. Водночас інформаційна безпека, яка забезпечує охорону з боку держави, не повинна гальмувати процеси формування національного інформаційного простору, що відповідав би інформаційно-інтелектуальному потенціалові держави та не перешкоджав би входженню України у світовий інформаційний простір як суб'єкта рівноправних міжнародних відносин. Зважаючи на це, стратегічним завданням державної політики щодо інформаційної безпеки має стати формування систем на основі науково обгрунтованих політичних, соціальних, економічних критеріїв та світового досвіду правового регулювання та організації забезпечення її функціонування. Система інформаційної безпеки відомостей, які підлягають охороні з боку держави, повинна відповідати правовому режимові кожного виду відомостей, діяти безперервно на кожному етапі їх формування і поширення та бути адекватною загрозам, що діють в інформаційній сфері.

Література

1. Мастяниця Й.І., Соснін О.В., Шиманський Л.Є. Захист інформаційних ресурсів України: проблеми і шляхи їх розв'язання. - К.: Національний інститут стратегічних досліджень, 2000. - 98 с.

2. Василюк В.Я., Климчук СО. Інформаційна безпека держави : Курс лекцій. - К.: КНТ, Видавничий дім «Скіф», 2008. - 136с,

3. Баранов О.А. Інформаційне право України: Стан, проблеми, перспективи. -К.: Видавничий дім «СофтПрес», 2005. - 316с.

4. Богуш В.М., Юдін О.К. Інформаційна безпека держави. - К.: «МК-Прес», 2005.-432с.

5. Почерцов Г.Г. Информационные войны. Основы военно-коммуникативных исследований. - М.:. Рефл-бук, К.: Ваклер, 2000. - 576с.

6. Расторгуев СП. Информационная война. - М.: Радио и связь, 1999. -416с.

7. Расторгуев СП. Философия информационной войны. - М: Московский психолого-социальный институт, 2003. - 486с.

8. Соснін О.В., Шименський Л.Є Про правові основи удосконалення системи державного управління інформаційними ресурсами. Політологічний вісник. 36. наук, праць, №10. - К.: Т-во «Знання України», 2002. - с.212-219

9. Баранов А.А. Концептуальные вопросы информационной безопасности Украины // Безопасность информации. - 1995, №2. - с.4-10

10. Конеев И.Р., Беляев А.В. Информационная безопасность предприятия. - СпБ.: БХВ - Петербург, 2003. - 688с.

11. П. Браіловський М.М., Головень СМ. та інші. - Технічний захист інформації на об'єктах інформаційної діяльності/ За ред. Проф. В.О. Хорошка. -К.: ДУІКТ, 2007. - 178с.

12.Петренко СА., Курбатов В.А. Политики информационной безопасности. - М.: Компания Ай Ти, 2006. - 400с.

Додаток 1

Варіант 1

1. Засоби інформаційної безпеки спрямовані на захист від:

* завдання неприйнятних збитків
* нанесення будь-якого збитку
* підглядання в замкову щілину

2. Що з перерахованого не відноситься до числа основних аспектів інформаційної безпеки?

* доступність
* цілісність
* конфіденційність
* правдиве відображення дійсності

3. Витрати організацій на інформаційну безпеку:

- ростуть
- залишаються на одному рівні
- знижуються

4. У звіті "Комп'ютерна злочинність і безпека-2002" 25 респондентів заявили про випадки підробок. Середній збиток від підробки складає:

* близько 1 млн. доларів
* близько 4,5 млн. доларів
* близько 9,8 млн. доларів

Варіант 2

1. Що таке захист інформації?

* захист від несанкціонованого доступу до інформації
* випуск броньованих коробочок для дискет
* комплекс заходів, спрямованих на забезпечення інформаційної безпеки

2. Що з перерахованого не відноситься до числа основних аспектів інформаційної безпеки?

* доступність

* масштабованість
* цілісність
* конфіденційність

3. Комп'ютерна злочинність у світі:
* залишається на одному рівні
* знижується
* росте

4. У звіті "Комп'ютерна злочинність і безпека-2002" 26 респондентів заявили про випадки крадіжок. Середній збиток крадіжки становить:
* близько 1 млн. доларів
* близько 6,5 млн. доларів
* близько 12,8 млн. доларів

Варіант 3

1. Що розуміється під інформаційною безпекою?
* захист духовного здоров'я телеглядачів
* захист від завдання неприйнятних збитків суб'єктам інформаційних відносин
* забезпечення інформаційної незалежності України

2. Що з перерахованого не відноситься до числа основних аспектів інформаційної безпеки?
* доступність
* цілісність
* захист від копіювання
* конфіденційність

3. Середній збиток від комп'ютерного злочину в США становить приблизно:
* сотні тисяч доларів
* десятки доларів
* копійки

4. У звіті «Комп'ютерна злочинність і безпека-2002» використані результати опитування:

* близько 100 респондентів

* близько 500 респондентів

* близько 1000 респондентів

Додаток 2

Варіант 1

1. Об'єктно-орієнтований підхід допомагає справлятися з:

* складністю систем

* недостатньою реактивністю систем

* неякісним користувацьким інтерфейсом

2. Об'єктно-орієнтований підхід використовує:

* семантичну декомпозицію

* об'єктну декомпозицію

* алгоритмічну декомпозицію

3. Вимога безпеки повторного використання об'єктів суперечить:

* інкапсуляції

* успадкуванню

* поліморфізму

Варіант 2

1. Будь-який розумний метод боротьби зі складністю спирається принцип:

* не варто множити сутності надміру

* заперечення заперечення

* розділяй і пануй

2. У число основних понять об'єктного підходу не входить:

* інкапсуляція

* успадкування

* поліфонізм

3. Розподіл на активні й пасивні сутності суперечить:

* класичній технології програмування
* основам об'єктно-орієнтованого підходу
* стандарту на мову програмування Сі

4. Припустимо, що при розмежуванні доступу ураховується семантика програм. У такому випадку на переглядач файлів певного формату можуть бути накладені наступні обмеження:

* заборона на читання файлів, крім переглядаючих і конфігуруючих
* заборона на зміну файлів крім переглядаючих і конфігуруючих
* заборона на зміну яких-небудь файлів

Варіант 3

1. Структурний підхід спирається на:
* семантичну декомпозицію
* алгоритмічну декомпозицію
* декомпозицію структур даних
2. Контейнери в компонентних об'єктних середовищах надають:
* загальний контекст взаємодії з іншими компонентами й оточенням
* засоби для збереження компонентів
* механізми транспортування компонентів
3. Метод об'єкта реалізує волю:
* користувача, який його викликав
* власника інформаційної системи
* розробника об'єкта
4. Припустимо, що при розмежуванні доступу враховується семантика програм. У такому випадку на ігрову програму можуть бути накладені наступні обмеження:
* заборона на читання яких-небудь файлів, крім конфігураційних
* заборона на зміну яких-небудь файлів, крім конфігураційних
* заборона на встановлення мережних з'єднань

Додаток З

Варіант 1

1. Вікно небезпеки - це:

* проміжок часу
* частина простору
* погано закріплена деталь будівельної конструкції

2. Найнебезпечнішими загрозами є:

■ ненавмисні помилки штатних співробітників

■ вірусні інфекції

■ атаки хакерів

3. Дублювання повідомлень є загрозою:

■ доступності

■ конфіденційності

■ цілісності

4. Melissa - це:

■ бомба

■ вірус

■ хробак

Варіант 2

1. Вікно небезпеки з'являється, коли:

■ стає відомо про засоби використання уразливості

■ з'являється можливість використати уразливість

■ установлюється нове ПЗ

2. Найнебезпечнішими джерелами загроз є:

* внутрішні
* зовнішні
* прикордонні

3. Перехоплення даних є загрозою:

* доступності
* конфіденційності

* цілісності

4. Melissa - це:

* макровірус для файлів MS-Word
* макровірус для файлів PDF
* макровірус для файлів Postscript

Варіант 3

1. Вікно небезпеки перестає існувати, коли:

* адміністратор безпеки довідається про загрозу
* виробник ПЗ випускає латку
* латка встановлюється в захищувану ІС

2. Найнебезпечнішими джерелами внутрішніх загроз є:

* некомпетентні керівники
* скривджені співробітники
* цікаві адміністратори

3. Агресивне споживання ресурсів є загрозою:

* доступності
* конфіденційності
* цілісності

4. Melissa піддає атаці на доступність:

* системи електронної комерції;
* геоікформаційної системи;
* системи електронної пошти.

Додаток 4

Варіант 1

1. Головна мета заходів, що впроваджуються на адміністративному рівні:

* сформувати програму безпеки й забезпечити її виконання
* виконати положення чинного законодавства
* відзвітувати перед вищими інстанціями

2. У число цілей політики безпеки верхнього рівня входять:

* рішення сформувати або переглянути комплексну програму безпеки

* забезпечення бази для дотримання законів і правил
* забезпечення конфіденційності поштових повідомлень

3. У число етапів життєвого циклу інформаційного сервісу входять:
* закупівля
* продаж
* виведення з експлуатації Варіант 2

1. Політика безпеки:
* фіксує правила розмежування доступу
* відбиває підхід організації до захисту своїх інформаційних активів
* описує способи захисту керівництва організації

2. У число цілей політики безпеки верхнього рівня входять:
* формулювання адміністративних рішень з найважливіших аспектів реалізації програми безпеки
* вибір методів аутентифікації користувачів
* забезпечення бази для дотримання законів і правил

3. У число етапів життєвого циклу інформаційного сервісу входять:
* ініціація
* термінація
* установка Варіант 3

1. Політика безпеки будується на основі:
* загальних подань про ІС організації
* вивчення політик подібних організацій
* аналізі ризиків

2. У число цілей політики безпеки верхнього рівня входять:
* визначення правил розмежування доступу
* формулювання цілей, які переслідує організація в області інформаційної безпеки
* визначення загальних напрямків у досягненні цілей безпеки

3. У число етапів життєвого циклу інформаційного сервісу входять:

* експлуатація
* специфікація прав людини
* виведення з експлуатації

Додаток 5

Варіант 1

1. Ризик є функцією:
   * ймовірності реалізації загрози
   * розміру можливого збитку ?
   * числа уразливих місць у системі
2. У число можливих стратегій нейтралізації ризиків входять:
   * ліквідація ризику
   * ігнорування ризику
   * прийняття ризику
3. У число етапів керування ризиками входять:
   * ідентифікація активів
   * ліквідація пасивів
   * вибір аналізованих об'єктів
4. Перший крок в аналізі загроз - це:
   * ідентифікація загроз
   * аутентифікація загроз
   * ліквідація загроз Варіант 2

1. Ризик є функцією:
   * розміру можливого збитку
   * числа уразливих місць у системі
   * статутного капіталу організації
2. У число можливих стратегій нейтралізації ризиків входять:
   * зменшення ризику
   * приховування ризику
   * афішування ризику
3. У число етапів керування ризиками входять:

* оцінка ризиків

* вибір рівня деталізації аналізованих об'єктів

* покарання за створення уразливостей

4. Після ідентифікації загрози необхідно оцінити:

* ймовірність її здійснення

* збиток від її здійснення

*частоту її здійснення Варіант 3

1. Ризик є функцією:

* ймовірності реалізації загрози

* вартості захисних засобів

* числа уразливих місць у системі

2. У число можливих стратегій нейтралізації ризиків входять:

* переадресація ризику

* деномінація ризику

* декомпозиція ризику

3. У число етапів керування ризиками входять:

* аналіз загроз

* загрози проведення аналізу

* виявлення уразливих місць

4. При аналізі вартості захисних заходів не слід ураховувати:

* витрати на закупівлю устаткування

* витрати на закупівлю програм

* витрати на навчання персоналу

Додаток 6

Варіант 1

1. У число класів заходів процедурного рівня входять:

* логічний захист

* фізичний захист

планування відбудовних робіт

2. У число принципів керування персоналом входять:

* розділяй і пануй"
* поділ обов'язків
* інкапсуляція успадкування

3. У число етапів процесу планування відбудовних робіт входять:

* виявлення критично важливих функцій організації
* визначення переліку можливих аварій
* проведення тестових аварій

Варіант 2

1. У число класів заходів процедурного рівня входять:

* керування персоналом
* керування персоналками
* реагування на порушення режиму безпеки

2. У число принципів керування персоналом входять:

* мінімізація привілеїв
* мінімізація зарплати
* максимізація зарплати

3. У число етапів процесу планування відбудовних робіт входять:

* ідентифікація персоналу
* перевірка персоналу
* ідентифікація ресурсів Варіант 3

1. У число класів заходів процедурного рівня входять:

* підтримка працездатності
* підтримка фізичної форми
* фізичний захист

2. У число принципів фізичного захисту входять:

* нещадна відсіч
* безперервність захисту в просторі й часі
* мінімізація захисних засобів

3. У число етапів процесу планування відбудовних робіт входять:

* розробка стратегії відбудовних робіт
* сертифікація стратегії
* перевірка стратегії

Додаток 7

Варіант 1

1. Протоколювання й аудит можуть використовуватися для:

* попередження порушень ІБ
* виявлення порушень
* відновлення режиму ІБ

2. Укажіть найбільш істотні з погляду безпеки особливості сучасних українських ІС:

* домінування платформи Wintei
* наявність підключення до Internet
* наявність різнорідних сервісів

3. У число основних принципів архітектурної безпеки входять:

* застосування найбільш передових технічних рішень
* застосування простих апробованих рішень
* сполучення простих і складних захисних засобів Варіант 2

1. Екранування може використовуватися для:

* попередження порушень ІБ
* виявлення порушень
* локалізації наслідків порушень

2. Укажіть найбільш істотні з погляду безпеки особливості сучасних українських ІС:

* низька пропускна здатність більшості комунікаційних каналів
* складність адміністрування користувальницьких комп'ютерів

* відсутність достатнього набору криптографічних апаратно-програмних продуктів

3. У число основних принципів архітектурної безпеки входять:

* наслідування визнаним стандартам

застосування нестандартних рішень, не відомих зловмисникам

* розмаїтість захисних засобів

Варіант 3

1. Контроль цілісності може використовуватися для:

* попередження порушень ІБ
* виявлення порушень
* локалізації наслідків порушень

2. Укажіть найбільш істотні з погляду безпеки особливості сучасних українських ІС:

* використання ПЗ з активними агентами
* використання піратського ПЗ
* використання вільно розповсюджуваного ПЗ

3. У число основних принципів архітектурної безпеки входять:

* посилення найслабшої ланки
* зміцнення найбільш ймовірного об'єкта атаки
* ешелонованість оборони

Додаток 8

Варіант 1

1. У якості аутентифікатора в мережевому середовищі можуть використовуватися:

* координати суб'єкта
* прізвище суб'єкта
* секретний криптографічний ключ

2. Аутентифікація на основі пароля, переданого по мережі у відкритому вигляді, погана, тому що не забезпечує захисту від:

* перехоплення
* відтворення
* атак на доступність

3. У число основних понять рольового керування доступом входять:

* роль
* виконавець ролі
* користувач ролі

Варіант 2

1. У якості аутентифікатора в мережевому середовищі можуть використовуватися:

* мережева адреса суб'єкта
* пароль
* цифровий сертифікат суб'єкта

2. Аутентифікація на основі пароля, переданого по мережі в зашифрованому вигляді, погана, тому що не забезпечує захисту від:

* перехоплення
* відтворення
* атак на доступність

3. У число основних понять рольового керування доступом входять:

* об'єкт
* суб'єкт
* метод

Варіант 3

1. У якості аутентифікатора в мережевому середовищі можуть використовуватися :

* кардіограма суб'єкта
* номер картки пенсійного страхування

* результат роботи генератора одноразових паролів

2. Аутентифікація на основі пароля, переданого по мережі в зашифрованому вигляді й забезпеченого відкритою тимчасовою міткою, погана, тому що не забезпечує захист)- від:

* перехоплення
* відтворення
* атак на доступність

4. Рольове керування доступом використовує наступні засоби об'єктно орієнтованого підходу

* інкапсуляція
* успадкування
* поліморфізм

Додаток 9

Варіант 1

1. Протоколювання саме по собі не може забезпечити невідмовність, тому що:
   * реєстраційна інформація, як правило, має низькорівневий характер, невідмовність відноситься до дій прикладного рівня
   * реєстраційна інформація має специфічний формат, незрозумілий людині
   * реєстраційна інформація має занадто великий обсяг
2. Сигнатурний метод виявлення атак хороший тим, що він:
   * піднімає мало фіктивних тривог
   * здатний виявляти невідомі атаки
   * простий у налаштуванні й експлуатації
3. Цифровий сертифікат містить:
   * відкритий ключ користувача
   * секретний ключ користувача
   * ім'я користувача

Варіант 2

1. Протоколювання саме по собі не може забезпечити невідмовність, тому що:

* реєстраційна інформація може бути роззосереджена по різних сервісах різних компонентах розподіленої ІС
* цілісність реєстраційної інформації може бути порушена
* повинна дотримуватися конфіденційність реєстраційної інформації, перевірка невідмовності порушить конфіденційність

2. Статистичний метод виявлення атак хороший тим, що він:
* піднімає мало фіктивних тривог
* здатний виявляти невідомі атаки
простий у налаштуванні й експлуатації 3. Цифровий

сертифікат містить;
* відкритий ключ центра, що засвідчує
* секретний ключ центра, що засвідчує
* ім'я центру, що засвідчує

Варіант 3

1. Протоколювання саме по собі не може забезпечити невідмовність, тому що:
* реєстраційна інформація відноситься до різних рівнів стека мережевих протоколів
* обсяг реєстраційної інформації дуже швидко росте, її доводиться переміщати на вторинні носії, читання з яких поєднано з технічними проблемами
* реєстраційна інформація для різних компонентів розподіленої системи може виявитися неузгодженою

2. Граничний метод виявлення атак хороший тим, що він:
* піднімає мало фіктивних тривог
* здатний виявляти невідомі атаки
* простий у налаштуванні й експлуатації

3. Цифровий сертифікат містить:
* ЕЦП користувача
* ЕЦП довіреного центра
* ЕЦП генератора криптографічних ключів

Додаток 10

Варіант 1

1. Екран виконує функції:

* розмежування доступу
* полегшення доступу
* ускладнення доступу

2. На міжмережевий екран доцільно покласти функції:

* активного аудиту
* аналізу захищеності
* ідентифікації/аутентифікації вилучених користувачів 3.

Екранування на мережевоіму рівні може забезпечити:

* розмежування доступу по мережевих адресах
* вибіркове виконання команд прикладного протоколу
* контроль обсягу7 даних, переданих по TCP-з' єднанню
* **Варіант 2**

1. Екран виконує функції:

* прискорення обміну інформацією
* протоколювання обміну інформацією
* уповільнення обміну інформацією

2. Демілітаризована зона розташовується:

* перед зовнішнім міжмережевим екраном
* між міжмережевими екранами
• за внутрішнім міжмережевим екраном

3. Екранування на мережевому й транспортному рівнях може забезпечити:

* розмежування доступу по мережевих адресах
* вибіркове виконання команд прикладного протоколу
* контроль обсягу даних, переданих по TCP-з'єднанню

Варіант 3

1. Екран виконує функції:

* очищення деяких елементів переданих даних

* поповнення деяких елементів переданих даних
* перетворення деяких елементів переданих даних

2. До міжмережевих екранів доцільно застосовувати наступні принципи архітектурної безпеки:
* посилення найслабшої ланки
* ешелонованість оборони
* неможливість переходу в небезпечний стан

3. Комплексне екранування може забезпечити:
* розмежування доступу по мережевих адресах
* вибіркове виконання команд прикладного протоколу
* контроль обсягу даних, переданих по TCP-з'єднанню

Додаток 11

Варіант 1

1. Інформаційний сервіс уважається недоступним, якщо:
* його ефективність не задовольняє накладеним обмеженням
* підписка на нього коштує занадто дорого
* не вдається знайти підходящий сервіс

2. Середній час наробітку на відмову:
* пропорційний інтенсивності відмов
* обернено пропорційний інтенсивності відмов
* не залежить від інтенсивності відмов

3.Достоїнствами синхронного тиражування є:
* ідейна простота
* простота реалізації
* стійкість до відмов мережі

Варіант 2

1. Ефективність інформаційного сервісу може вимірятися як:
* рентабельність роботи сервісу
* максимальний час обслуговування запиту
* кількість користувачів, що обслуговують одночасно

2. Інтенсивності відмов незалежних компонентів:

* додаються
* множаться
* підносяться до квадрату і додаються

3. Достоїнствами асинхронного тиражування є:

* ідейна простота
* простота реалізації
* стійкість до відмов мережі

Варіант 3

1. Забезпечення високої доступності можна обмежити:

* критично важливими серверами
* мережевим устаткуванням
* всім ланцюжком від користувачів до серверів

2. У число основних загроз доступності входять:

* відмова користувачів
* підвищення цін на послуги зв'язку
* відмова підтримуючої інфраструктури

3. Основними функціями ПЗ проміжного шару, істотними для забезпечення високої доступності, є:

* маршрутизація запитів
* балансування завантаження
* доступність вільно розповсюджуваних реалізацій

Додаток 12

Варіант 1

1. Тунелювання може використовуватися на наступних рівнях еталонної семирівневої моделі:

* канальному
* транспортному
* сеансовому

2. Відповідно до стандарту X.700, у число функцій керування конфігурацією входять:

* запуск і зупинка компонентів
* вибір закуповуваної конфігурації
* зміна конфігурації системи

3. Каркас необхідний системі керування для додання:

* гнучкості
* твердості
* стійкості

Варіант 2

!. Тунелювання може використовуватися на наступних рівнях еталонної семирівневої моделі:

* мережевому
* сеансовому
* рівні подання

2. Відповідно до стандарту X.700, у число функцій керування відмовами входять:

* попередження відмов
* виявлення відмов
* усунення відмов

3. Виявлення неадекватного поводження виконується системами керування шляхом застосування методів, типових для:

* систем аналізу захищеності
* систем активного аудита
* систем ідентифікації Варіант 3

1. Тунелювання може використовуватися на наступних рівнях еталонної семирівневої моделі:

* канальному
* мережевому
* транспортному

2. Відповідно до стандарту X.700, у число функцій керування безпекою входять:

* створення інцидентів
* реагування на інциденти
* усунення інцидентів

3. Архітектурними елементами систем керування є:

* агенти
* клієнти
* менеджери

НАВЧАЛЬНЕ ВИДАННЯ

Хорошко Володимир Олексійович Чередниченко
Володимир Станіславович Шелест Михайло Євгенович

ОСНОВИ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

За редакцією професора Хорошка В.О. (українською мовою)

Редактор О.М. Антонова Технічний редактор Д.В. Чирков Коректор О.І. Стельмаховська Художник-дизайнер С.В.Корнієнко

Підписано до друку 31.03.2008 р. Формат 60х84 1/16. Друк офсет. Ум. др. арк. 13,.28 Наклад 350 прим. Зам. № 4/08

Видавець та виробник Державний університет інформаційно-комунікаційних технологій 03110, Київ, вул. Солом'янська, 7
Свідоцтво суб'єкта видавничої справи серія ДК2539 від 26.06.2006 р.