ЖУКОВИЦЬКИЙ І.В., ОСТАПЕЦЬ Д.О.

ЗАХИСТ ІНФОРМАЦІЇ В КОМП’ЮТЕРНИХ МЕРЕЖАХ

Навчальний посібник

КЛАСИФІКАЦІЯ АТАК У КОМП'ЮТЕРНИХ МЕРЕЖАХ

ХАРАКТЕРИСТИКА ТИПОВИХ МЕРЕЖЕВИХ АТАК

МЕХАНІЗМИ РЕАЛІЗАЦІЇ ДЕЯКИХ МЕРЕЖЕВИХ АТАК

МІЖМЕРЕЖЕВІ ЕКРАНИ

ЗАХИЩЕНІ ВІРТУАЛЬНІ МЕРЕЖІ (VPN)

РОЗПОДІЛ КРИПТОГРАФІЧНИХ КЛЮЧІВ

УДУНТ, 2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

УКРАЇНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ НАУКИ І ТЕХНОЛОГІЙ

І. В. Жуковицький, Д. О. Остапець

Захист інформації в комп’ютерних мережах

НАВЧАЛЬНИЙ ПОСІБНИК

ДНІПРО
2024

УДК 004.056.5(075.8)
Ж 85

Авторський колектив:
Жуковицький І. В., Остапець Д. О.

Рекомендовано Радою якості освітньої діяльності УДУНТ
Протокол № 5 від «23» січня 2024 р.

Ж 85 **Жуковицький І. В.** Захист інформації в комп'ютерних мережах : навч. посіб. / І. В. Жуковицький, Д. О. Остапець ; за ред. д-ра техн. наук, проф. І. В. Жуковицького ; Укр. держ. ун-т науки і технологій. – Електрон. вид. – Дніпро: УДУНТ, 2024. – 146 с.

ISBN 978-617-8314-15-6 (PDF)

У навчальному посібнику викладено теоретичні основи захисту інформації в комп'ютерних мережах, перш за все в мережі Internet.

Призначений для опанування освітньої компоненти «Захист інформації в комп'ютерних мережах» та дипломного проєктування за спеціальністю 125 «Кібербезпека та захист інформації» для ОП «Кібербезпека та захист інформації» першого (бакалаврського) рівня.

Іл. 118, табл. 5, бібліогр. 8 назв.

УДК 004.056.5(075.8)

ISBN 978-617-8314-15-6 (PDF)
DOI 10.15802/978-617-8314-15-6

ЗМІСТ

ПЕРЕДМОВА

Важливою часткою інформаційних систем є комп'ютерні мережі і, безумовно, найбільша комп'ютерна мережа Inetrnet. До мережі Internet підключено багато сервісів, включаючи інформаційні сервіси, комерційні сервіси, банківські сервіси тощо. Між тим мережевий стек TCP/IP, який є основою Internet, побудований ще в 80 роках минулого століття, коли інформаційні загрози, які сьогодні широко розповсюджені, ще не відчувались. Тому питанням інформаційної безпеки в цьому стеку майже не приділялось уваги. Але сьогодні атаки на сервіси Internet завдають багатомільярдні збитки власникам і користувачам цих сервісів. Тому захист комп'ютерних мереж стає все більш актуальнішою задачею. Саме питанням інформаційної безпеки комп'ютерних мереж присвячений курс «Захист інформації в комп'ютерних мережах», що викладається студентам освітньої програми «Кібербезпека та захист інформації» спеціальності 125 «Кібербезпека та захист інформації» першого (бакалаврського) рівня. Метою представленого навчального посібника є допомогти студентам досягти компетентностей, які основані на зазначених в освітньо-професійній програмі (ОП) цієї спеціальності.

- Здатність до використання інформаційно- комунікаційних технологій, сучасних методів і моделей інформаційної безпеки та/або кібербезпеки.
- Здатність забезпечувати захист інформації, що обробляється в інформаційно-телекомунікаційних (автоматизованих) системах з метою реалізації встановленої політики інформаційної та/або кібербезпеки.
- Здатність застосовувати методи та засоби криптографічного та технічного захисту інформації на об'єктах інформаційної діяльності).

Також навчальний посібник спрямований на те, щоб довести до студентів основні положення курсу, допомогти досягнути наступних результатів навчання:

- Вирішувати завдання захисту програм та інформації, що обробляється в інформаційно-телекомунікаційних системах програмно-апаратними засобами та давати оцінку результативності якості прийнятих рішень.
- Забезпечувати процеси захисту та функціонування інформаційно-телекомунікаційних (автоматизованих) структурних (структурно-логічних) схем, топології мережі, сучасних архітектур та моделей захисту електронних інформаційних ресурсів з відображенням взаємозв'язків та інформаційних потоків, процесів для внутрішніх і віддалених компонент.
- Реалізовувати заходи з протидії отриманню несанкціонованого доступу до інформаційних ресурсів і процесів в інформаційних та інформаційно-телекомунікаційних (автоматизованих) системах
- Впроваджувати заходи та забезпечувати реалізацію процесів попередження отриманню несанкціонованого доступу і захисту інформаційних, інформаційно- телекомунікаційних (автоматизованих) систем на основі еталонної моделі взаємодії відкритих систем

- Вирішувати задачі захисту потоків даних в інформаційних, інформаційно-телекомунікаційних (автоматизованих) системах.
- Виконувати впровадження та підтримку систем виявлення вторгнень та використовувати компоненти криптографічного захисту для забезпечення необхідного рівня захищеності інформації в інформаційно-телекомунікаційних системах.

Для реалізації цієї мети в навчальному посібнику надано теоретичні положення захисту інформації в комп'ютерних мережах, приклади використання цих методів, опис стандартних протоколів захисту тощо. Крім того, кожний розділ навчального посібника закінчується контрольними питаннями, відповідь на які (або пошук цієї відповіді) сприяє засвоєнню студентами матеріалу розділу.

РОЗДІЛ 1 КЛАСИФІКАЦІЯ АТАК У КОМП'ЮТЕРНИХ МЕРЕЖАХ

1.1 Загальні принципи класифікації атак у комп'ютерних мережах

У літературі описано велику кількість атак у мережах TCP/IP. Запропоновано різні класифікації таких атак.

Всі мережеві загрози в залежності від об'єкта, що піддається впливу, можна розділити на наступні два підмножини:

• віддалені атаки на інфраструктуру (під інфраструктурою мережі ми розумітимемо сформовану систему організації відносин між об'єктами мережі та сервісні служби, що використовуються в мережі) і протоколи мережі (множина 1);

• віддалені атаки на телекомунікаційні служби чи сервери надання віддаленого сервісу (множина 2).

Загрози множини 1 використовують уразливості в мережевих протоколах та в інфраструктурі мережі, а загрози множини 2 – уразливості в телекомунікаційних службах («дірки», програмні закладки, програмні помилки).

Далі в курсі розглядаються мережеві атаки, засновані на загрозах множини 1. Набула поширення наступна класифікація таких атак у комп'ютерних мережах (рис.1.1).

1.2 Класифікації атак за характером впливу

1.2.1 Пасивний вплив

Пасивним впливом на об'єкт мережі (мережевий вузол, набір мережевих вузлів, ділянку мережі) назвемо вплив, що не безпосередньо впливає на роботу об'єкта, але може порушувати його політику безпеки.

Саме відсутність безпосереднього впливу на роботу розподіленої ПС призводить до того, що пасивний віддалений вплив практично неможливо виявити. Прикладом типового пасивного впливу комп'ютерної мережі служить прослуховування каналу зв'язку. При пасивному впливі, на відміну від активного, немає ніяких слідів. Однак, у деяких випадках у процесі пасивного впливу потенційний порушник може бути виявлений.

1.2.2 Активний вплив

Активним впливом на об'єкт назвемо вплив, що безпосередньо впливає на його роботу (зміна конфігурації, порушення працездатності і т. д.) і порушує прийняту в ньому політику безпеки. Більшість віддалених атак є активним впливом. Очевидною відмінністю активного впливу пасивного є важлива можливість виявлення, оскільки у його здійснення у системі відбуваються певні зміни.

Прикладом активного впливу в комп'ютерній мережі служить атака «Відмова в обслуговуванні».

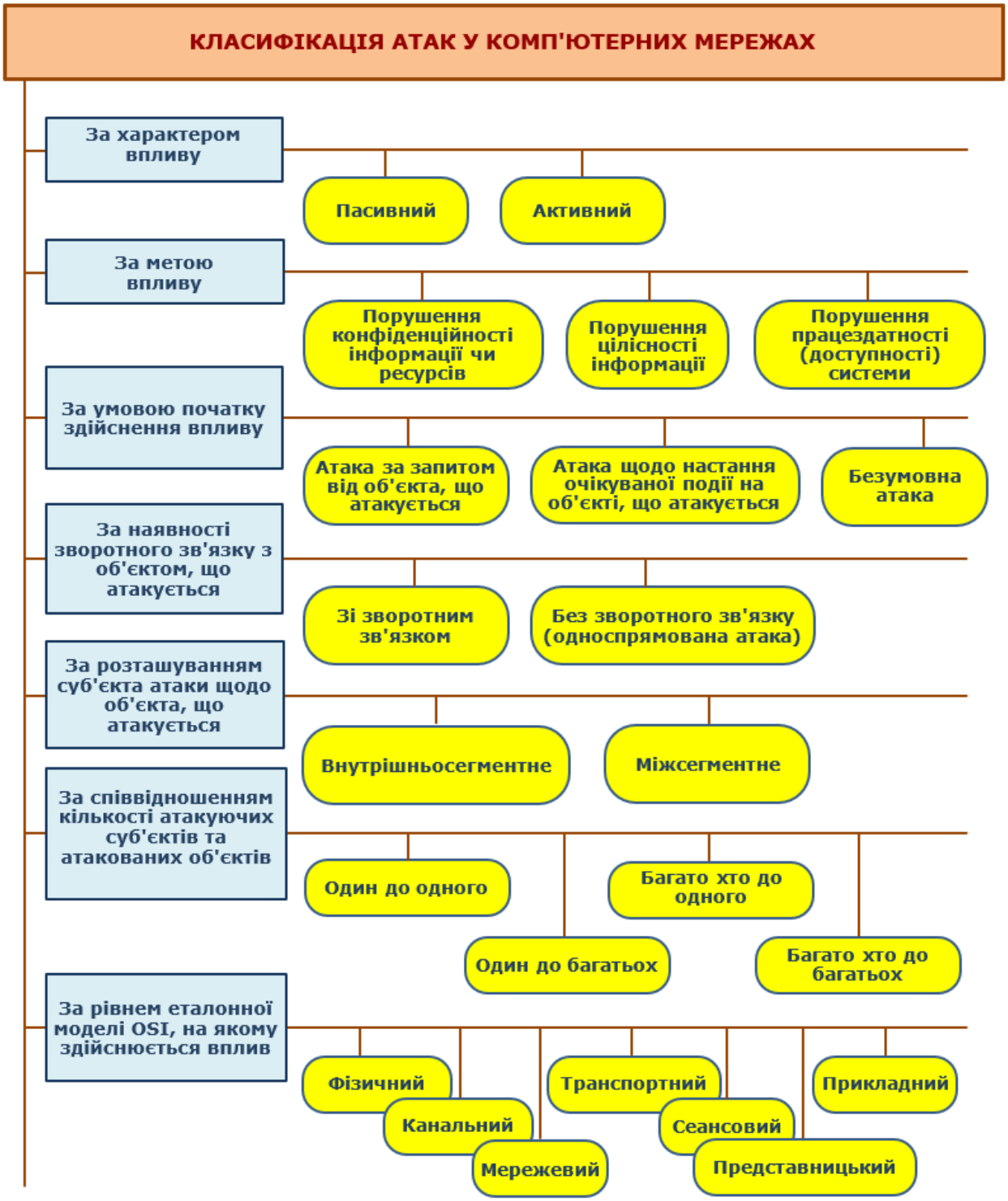


Рисунок 1.1. Класифікація атак у комп'ютерних мережах

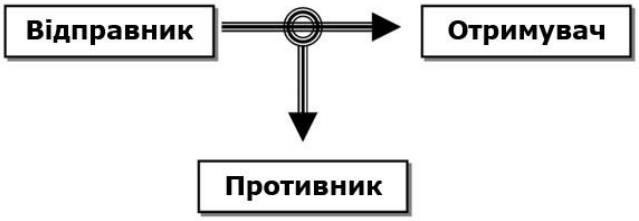


Рисунок 1.2. Модель пасивної дії

Рисунок 1.3. Модель активної дії

1.3 Класифікації атак за метою впливу

1.3.1 Модель інформаційної безпеки «Тріада КДЦ»

Мета будь-якої атаки в мережі – порушення якоїсь категорії інформаційної безпеки. Перелік цих категорій визначається моделями інформаційної безпеки. Одна з найпростіших (і перша з відомих) моделей інформаційної безпеки – це тріада КДЦ (рис. 1.4) – **К**онфіденційність, **Д**оступність, **Ц**ілісність. Їй відповідають такі цілі впливу.

Рисунок 1.4. Модель інформаційної безпеки «Тріада КДЦ»

Порушення конфіденційності інформації чи ресурсів системи. Перехоплення інформації може призвести до порушення її конфіденційності (якщо інформація незашифрована). Прикладом перехоплення інформації може бути прослуховування каналу в мережі. І тут реалізується несанкціонований доступом до інформації без можливості її спотворення.

Перехоплення службової інформації, протокольних повідомлень може дати відомості про використовувані мережеві протоколи, операційні системи, встановлені на хостах та ін.

Сканування IP-адрес або портів транспортного рівня може дати інформацію про ресурси системи.

Порушення цілісності інформації. Можливість спотворення інформації означає або повний контроль над інформаційним потоком між об'єктами системи (наприклад, атака «Помилковий ARP-сервер», описана в наступному розділі), або можливість надсилання повідомлень від імені іншого суб'єкта (атака «Підміна довіреного суб'єкта», описана нижче). Отже, зрозуміло, що спотворення чи фабрикація інформації веде порушення її цілісності.

Порушення працездатності (доступності) системи. І тут передбачається отримання атакуючим несанкціонованого доступу до інформації. Його основна мета – домогтися, щоб для всіх інших об'єктів системи доступ до ресурсів атакованого об'єкта був би неможливий. Прикладом такої атаки може бути атака DoS (Denial of Service – «відмова в обслуговуванні»).

1.3.2 Модель інформаційної безпеки «Гексада Паркера»

Ця модель (рис. 1.5) доповнила модель «Тріада КДЦ» ще трьома категоріями інформаційної безпеки, котрим відповідають наступні цілі впливу.

Порушення автентичності – зловмисник намагається видати себе за іншого, видати спотворений чи самостійно складений документ за документ, складений довіреним суб’єктом.

Порушення володіння – можливість переведення системи у стан, у якому фізичний контроль над пристроєм чи іншим середовищем передавання інформації надається не лише тим, хто має на це право, а й зловмиснику.

Порушення корисності – можливість переведення системи у стан, при якому не забезпечується зручність практичного використання як інформації, так і пов'язаних з її обробкою і передаванням процедур. У безпечній системі заходи, що вживаються для захисту системи, не повинні неприйнятно ускладнювати роботу співробітників, інакше вони сприйматимуть їх як перешкоду та намагатимуться за будь-якої можливості їх обійти.

1.3.3 Модель інформаційної безпеки STRIDE

Це модель виявлення загроз комп'ютерної безпеки, що розроблена Microsoft, надає мнемоніку загроз безпеці у шести категоріях (рис. 1.6).

Слід відмітити що «Зміна» – це аналог «Порушення цілісності», «Розголошення відомостей» – це аналог «Порушення конфіденційності», а «Відмова в обслуговуванні» – аналог «Порушення доступності» з тріади КДЦ. Додаткові загрози з цієї моделі наступні.

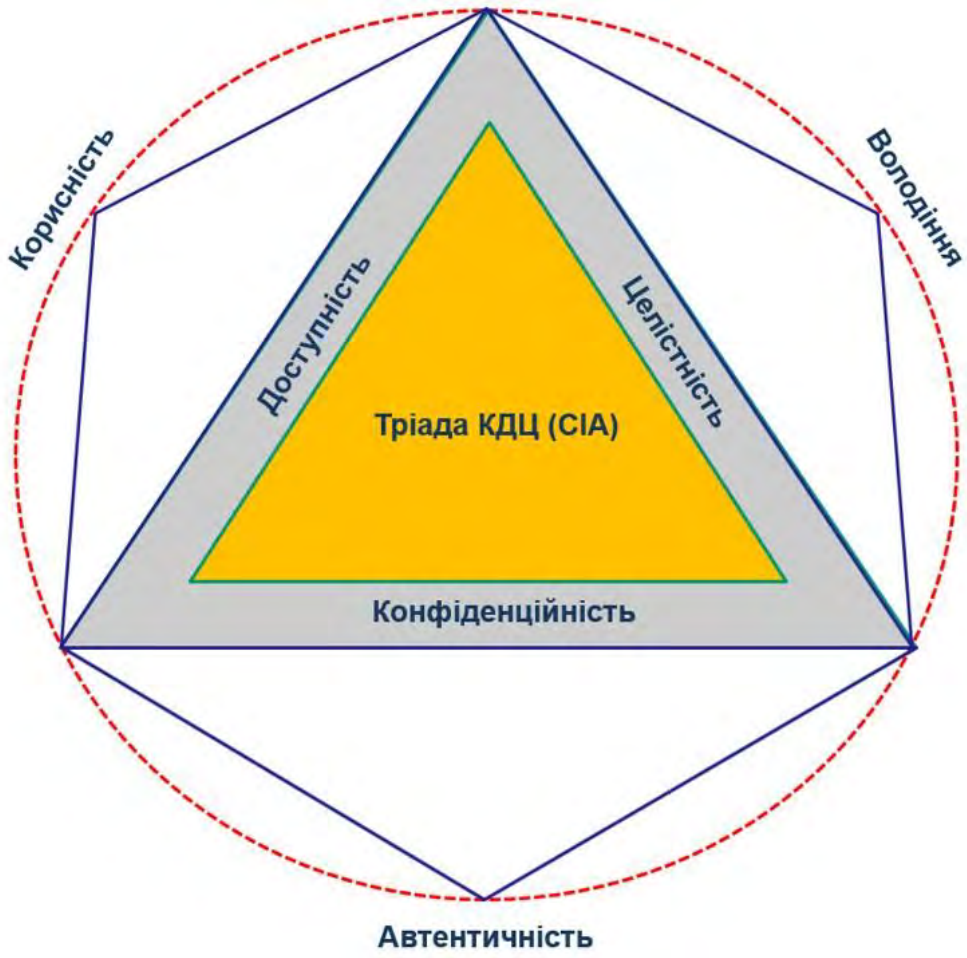


Рисунок 1.5. Модель інформаційної безпеки «Гексада Паркера»

Spoofing	Підміна даних
Tampering	Зміна
Repudiation	Відмова від відповідальності
Information disclosure	Розголошення відомостей
Denial of service	Відмова в обслуговуванні
Elevation of privilege	Захоплення привілеїв

Рисунок 1.6. Модель інформаційної безпеки STRIDE

Підміна даних – порушення, при якому користувач або інший суб'єкт ІС шляхом заміни даних (наприклад, IP-адреси відправника) видає себе за іншого. Бачимо, що це аналог «Порушення автентичності» моделі «Гексада Паркера».

Відмова від відповідальності – порушення, відоме, як порушення невідмовності – хибне твердження, що ви чогось не зробили чи не несете відповідальності.

Захоплення привілеїв – порушення, при якому зловмисник несанкціонованим чином підвищує свої повноваження в системі (наприклад, незаконне надання зловмиснику прав мережевого адміністратора)

1.4 За умовою початку здійснення дії

1.4.1 Атака за запитом від об'єкта, що атакується

У цьому випадку атакуючий очікує передачі від потенційної мети атаки запиту певного типу, який буде умовою початку здійснення впливу. Наприклад, при атаці «Помилковий ARP-сервер» атакуючий чекає на ARP-запит, на який дасть помилкову ARP-відповідь (атак розглянута в наступній лекції).

1.4.2 Атака щодо настання очікуваної події на об'єкті, що атакується

У цьому випадку атакуючий здійснює постійне спостереження за станом операційної системи віддаленої цілі атаки (в інтернет є велика кількість програм для віддаленого стеження за комп'ютером: AeroAdmin, NeoSpy та ін.) та при виникненні певної події в цій системі (наприклад, некоректного завершення роботи процесу) починає дію. Як і в попередньому випадку, ініціатором здійснення початку атаки виступає сам об'єкт, що атакується.

1.4.3 Безумовна атака

У цьому випадку початок здійснення атаки безумовний по відношенню до мети атаки, тобто атака здійснюється негайно і безвідносно до стану системи та об'єкта, що атакується. Прикладом такої атаки може бути прослуховування каналу зв'язку, багато версій DoS. Отже, у разі атакуючий є ініціатором початку здійснення атаки.

1.5 За наявності зворотного зв'язку з об'єктом, що атакується

1.5.1 Зі зворотним зв'язком

Віддалена атака, що здійснюється за наявності зворотного зв'язку з об'єктом, що атакується, характеризується тим, що на деякі запити, передані на атакований об'єкт, атакуючому потрібно отримати відповідь, і, отже, між атакуючим і ціллю атаки існує зворотний зв'язок, який дозволяє атакуючому адекватно реагувати на все зміни, що відбуваються на об'єкті, що атакується (рис. 1.7).

Рисунок 1.7. Атака за наявності зворотного зв'язку

Прикладом такої атаки може бути формування з'єднання протоколу TCP. Проведення такої атаки утруднено (для атакуючого) тим, що в більшості

випадків потрібно видати себе за довірений об'єкт, щоб отримати повідомлення у відповідь.

1.5.2 Без зворотного зв'язку (односпрямована атака)

Атаки даного виду зазвичай здійснюються передачею на об'єкт, що атакується, одиночних повідомлень, відповіді на які атакуючому не потрібні (рис. 1.8). Такі повідомлення надсилаються, наприклад, у вигляді ICMP-повідомлень або повідомлень, що передаються протоколом UDP. Подібну атаку можна називати односпрямованою віддаленою атакою.

Рисунок 1.8. Атака без зворотного зв'язку

1.6 За розташуванням суб'єкта атаки щодо об'єкта, що атакується

1.6.1 Сегментація мережі

Сегмент мережі є частиною комп'ютерної мережі. Характер і ступінь сегмента залежить від характеру мережі та пристрою чи пристроїв, що використовуються для з'єднання кінцевих станцій.

Для стандарту Ethernet фізичний сегмент мережі є електричне з'єднання між мережевими пристроями з використанням загального середовища. Фізичний сегмент цієї мережі є доменом колізій або сегментом рівня 1 Пристрої, що працюють на першому рівні моделі OSI (повторювачі, концентратори), не обмежують домен колізій. Саме на цьому рівні можливе пасивне прослуховування каналу.

За допомогою мостів, комутаторів сегменти першого рівня OSI (шар 1) можуть бути об'єднані в сегмент шару 2, тобто всі вузли можуть взаємодіяти один з одним за допомогою MAC-адресації. Сегмент шару 2 еквівалентний поняттю широкомовний домен другого рівня OSI.

Сегментом шару 3 в IP-мережі називають підмережу, утворену всіма вузлами, що розділяють один і той же префікс мережі, як визначено їх IP-адресою і маскою підмережі. Зв'язок між підмережами (шар 3 сегментів) реалізують маршрутизатори.

1.6.2 Внутрішньосегментна атака

Якщо суб'єкт (атакуюча програма чи оператор) який безпосередньо здійснює вплив на об'єкт атаки (хост, маршрутизатор) перебувають у одному фізичному сегменті (рівня 1), це дозволить атакуючому пасивно прослухати канал зв'язку (рис. 1.9).

Рисунок 1.9. Зловмисник в одному фізичному сегменті з об'єктами атаки (робочими станціями, маршрутизатором)

Розташування зловмисника і об'єкта атаки в одному сегменті другого рівня (але у різних фізичних сегментах – на різних портах комутатора, наприклад) не дозволяє реалізувати пасивне прослуховування (рис. 1.10). Однак, тут можливі атаки з використанням широкомовних повідомлень канального рівня, наприклад, «хибний ARP-сервер» (розглядається далі в курсі), що дозволяють реалізувати активне прослуховування.

Рисунок 1.10. Зловмисник і об'єкти атаки в одному сегменті другого рівня (але у різних фізичних сегментах – на різних портах комутатора

1.6.3 Міжсегментна атака

Суб'єкт та об'єкт атаки знаходяться в різних сегментах 2 або 3 шару.

Насправді міжсегментну атаку здійснити значно складніше, ніж внутрішньосегментну, оскільки в атакуючого немає доступу до каналу немає можливості прямого прослуховування. Більше того, немає можливості здійснити атаку з використанням широкомовних повідомлень канального рівня (такі повідомлення можливі лише всередині одного сегмента другого шару).

Міжсегментна віддалена атака становить набагато більшу небезпеку, ніж внутрішньосегментна. Це пов'язано з тим, що у випадку міжсегментної атаки об'єкт її і безпосередньо атакуючий можуть перебувати на відстані багатьох

тисяч кілометрів один від одного, що може суттєво перешкодити заходам відображення атаки.

1.7 За рівнем еталонної моделі ISO/OSI, на якому здійснюється вплив

1.7.1 Фізичний рівень

Зловмиснику необхідний доступ до фізичного каналу.
Отримавши такий доступ, зловмисник може:

• Обмежити доступ легальних користувачів до каналу:
  – для провідних мереж – пошкодити кабель;
  – для провідних (фізично незахищених) та бездротових мереж заглушити основний сигнал електромагнітними перешкодами (рис. 1.11).

Рисунок 1.11. Атакуючий глушить основний сигнал від точки доступу бездротової мережі

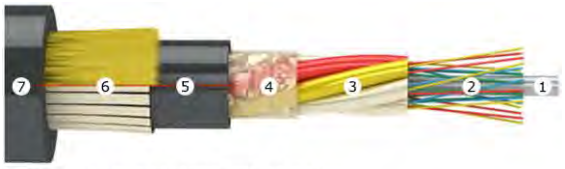
• Прослухати канал зв'язку:
  – для бездротових (радіо-) мереж можливе прослуховування радіохвиль, які забезпечують зв'язок між вузлами мережі (рис. 1.12);

Рисунок 1.12. Прослуховування зловмисником радіохвиль між точкою доступу та користувачами мережі

  – для металевих провідних мереж можливе знімання інформації шляхом прослуховування електромагнітних сигналів, які випромінюють металеві дроти (табл. 1.1).
  – оптичні провідні мережі ніякі сигнали не випромінюють, тому для їх прослуховування необхідне фізичне вклинювання в кабель, що потребує дорогого обладнання та може бути виявлене; цей вид зв'язку є одним з самих безпечних.

Таблиця 1.1.

Список кабельних з'єднань за зростанням складності їх прослуховування та глушення

Невита пара – сигнал може прослуховуватися на відстані кілька сантиметрів без безпосереднього контакту;	
Віта пара – сигнал дещо слабший, прослуховування без безпосереднього контакту в кабелі без екранування також можливе.	
Для кабелю з екрануванням можливе лише фізичне врізання Кабель категорії 6	
Кабель категорії 7	
Коаксіальний кабель – центральна жила надійно екранована обплетенням. Необхідний спеціальний контакт, що розсуває або ріже частину обплетення, і проникає до центральної жили.	
Оптичне волокно. Для прослуховування оптичних провідних мереж необхідно вклинювання в кабель та дороге обладнання, процес під'єднання до кабелю супроводжується перериванням зв'язку та може бути виявлений.	 1. ЦСЕ - склопластиковий стрижень 2. Оптичне волокно 3. Оптичны модулы заповнены гідрофобним гелем 4. Міжмодульний гідрофобний гель 5. Проміжна оболонка 6. Зміцнюючі елементи (армідні нитки/стеклонитки) 7. Оболонка

• Провести атаку «людина посередині»:
- для провідних мереж зловмиснику необхідно виконати фізичне врізання в кабель, при якій сигнал йтиме не безпосередньо до одержувача, а через зловмисника;
- для бездротової мережі імітувати сигнал від легального відправника.

1.7.2 Канальний рівень

Доступ до каналу дозволяє прослуховувати інформацію, що передається (кадри та їх зміст), формувати помилкові запити та відповіді в протоколах канального рівня (наприклад, в ARP-протоколі). Приклади атак на канальному рівні:

• переповнення таблиці комутації;

• прослуховування;
• несанкціонована зміна MAC-адреси;
• заміна записів у ARP-таблиці;
• заміна DHCP-сервера в мережі.

1.7.3 Мережевий рівень

На цьому рівні атака реалізується з використанням службових пакетів та протоколів мережного рівня. Наприклад, за допомогою ICMP-повідомлень або керуючих повідомлень маршрутизаторів. Приклади атак на мережевому рівні:

• підміна IP-адрес;
• фрагментація;
• інкапсуляція даних у ICMP пакети;
• модифікація маршрутної таблиці;
• використання широкомовних адрес для посилення атаки

1.7.4 Транспортний рівень

На цьому рівні проводиться сканування портів, формування неправдивих UDP або TCP-повідомлень у тих службових протоколах, для яких передбачені обміни UDP-дейтаграмами або TCP-сегментами (деякі протоколи маршрутизації, протокол DNS тощо). UDP або TCP-повідомлення використовуються в багатьох різновидах DoS-атак. Приклади атак на транспортному рівні:

• підміна порту відправника;
• маніпуляції або переповнення таблиці стану TCP з'єднань;
• використання нестандартних комбінацій прапорів та параметрів TCP;
• перехоплення TCP з'єднання;
• DoS атака.

1.7.5 Прикладний рівень

На цьому рівні відбувається втручання у роботу прикладних програм. Як приклад:

• отруєння кеша DNS;
• заміна ідентифікатора користувача;
• використання помилок ПЗ;
• підбір пароля;
• ін'єкції SQL.

1.8 За співвідношенням кількості атакованих об'єктів та атакуючих суб'єктів

Традиційна модель атаки будується за принципом «один до одного» (рис. 1.13) або «один до багатьох» (рис. 1.14), тобто атака виходить із одного джерела.

Рисунок. 1.13. Відношення «один до одного»

Рисунок 1.14. Відношення «один до багатьох»

Велику небезпеку становлять розподілені атаки. Ці атаки дозволяють одному або декільком зловмисникам проводити сотні і тисячі нападів, що здійснюються в один момент часу, на один або кілька вузлів. Модель розподіленої або скоординованої атаки заснована на відносинах «багато-до-одного» (рис. 1.15) і «багато хто-до-багатьох» (рис. 1.16).

Всі розподілені атаки засновані на «класичних» атаках типу «відмова в обслуговуванні», точніше – на їхньому підмножині – лавинних атаках. Сенс даних атак полягає у посилці великої кількості пакетів на заданий вузол мережі (мета атаки), що може призвести до виведення цього вузла з ладу, оскільки він «захлинеться» в потоці пакетів, що посилаються і не зможе обробляти запити авторизованих користувачів. У випадку ж розподіленої атаки атака відбувається вже не з однієї точки Internet, а відразу з декількох, що обумовлює різке зростання трафіку і вихід вузла, що атакується, з ладу.

Рис. 1.15. Відношення «багато хто до одного

Рис. 1.16. Відношення «багато хто до багатьох»

1.9 Запитання до розділу

1) Класифікація атак у комп'ютерних мережах за характером впливу? Приклади.

2) Класифікація атак у комп'ютерних мережах з мети впливу? Які моделі для цього використовуються. Приклади.

3) Класифікація атак у комп'ютерних мережах за умови початку здійснення дії? Приклади.

4) Класифікація атак у комп'ютерних мережах за наявності зворотного зв'язку з об'єктом, що атакується? Приклади.

5) Класифікація атак у комп'ютерних мережах щодо розташування суб'єкта щодо об'єкта, що атакується? Приклади.

6) Класифікація атак у комп'ютерних мережах за рівнем еталонної моделі ISO/OSI, на якому здійснюється вплив? Приклади.

РОЗДІЛ 2 ХАРАКТЕРИСТИКА ТА МЕХАНІЗМИ РЕАЛІЗАЦІЇ ТИПОВИХ МЕРЕЖЕВИХ АТАК

2.1 Поняття типової атаки

Незалежно від мережевих протоколів, топології, інфраструктури комп'ютерної мережі, механізми реалізації віддалених впливів на вузли мережі інваріантні по відношенню до особливостей конкретної системи. Це пояснюється тим, що комп'ютерні мережі проектуються з урахуванням тих самих принципів, і, отже, мають практично однакові проблеми безпеки.

Типова мережева атака — це віддалена інформаційна руйнівна дія, що програмно здійснюється по каналах зв'язку і характерна для будь-якої комп'ютерної мережі.

2.2 Аналіз мережевого трафіку – прослуховування каналу зв'язку

Вузли мережі (об'єкти) обмінюються один з одним набором даних. Зловмисник, підключившись до каналів зв'язку мережі, має можливість ці дані перехопити і проаналізувати.

Аналіз мережевого трафіку дозволяє: по-перше, вивчити логіку роботи мережі. Знання логіки роботи мережі дозволяє практично моделювати і здійснювати типові віддалені атаки.

По-друге, перехоплення потік даних, якими обмінюються об'єкти мережі, порушує конфіденційність інформації, якою обмінюються мережеві абоненти. Зазначимо, що пасивний перехоплення можливе лише всередині одного фізичного сегмента мережі (рівня 1) і при цьому відсутня можливість модифікації трафіку.

Здійснення цієї атаки без зворотного зв'язку веде до порушення конфіденційності інформації всередині одного фізичного сегмента мережі на канальному рівні OSI. При цьому початок здійснення атаки безумовно по відношенню до мети атаки.

Однак, крім пасивного прослуховування, може бути активне, розглянуте далі.

2.3 Підміна довіреного об'єкта чи суб'єкта у мережі

2.3.1 Загрози підміни

Загроза полягає у можливості порушника видавати себе за легітимного користувача та виконувати прийом/передачу даних від його імені. Цю загрозу можна охарактеризувати як «імітація дій клієнта чи сервера».

Ця загроза обумовлена слабкостями технологій мережевої взаємодії, які часто не дозволяють виконати перевірку справжності джерела чи отримувача інформації. Реалізація цієї загрози можлива за наявності у порушника підключення до обчислювальної мережі, а також відомостей про конфігурацію мережевих пристроїв, типу програмного забезпечення, що використовується тощо.

При цьому існують два різновиди даної типової віддаленої атаки:

2.3.2 Атака при встановленому з'єднанні

У процесі створення з'єднання (наприклад, TCP-з'єднання, рис. 2.1) об'єкти мережі обмінюються певною інформацією, що унікально ідентифікує дане з'єднання. Такий обмін зазвичай називається «рукостисканням» (handshake). Для TCP-з'єднання інформацією, що ідентифікує є IP-адреса хоста, початковий номер байта у власному потоці сегментів (ISN), підтвердження початкового номера байта партнера.

Рисунок 2.1. У процесі створення TCP-з'єднання сторони обмінюються рядом службових повідомлень

У разі встановленого віртуального з'єднання атака полягатиме у присвоєнні собі атакуючим прав довіреного суб'єкта взаємодії при підключенні до об'єкта атаки, що дозволить атакуючому вести сеанс роботи з об'єктом розподіленої системи від імені довіреного суб'єкта. Для здійснення атаки даного типу необхідно подолати систему ідентифікації та аутентифікації повідомлень.

Також можлива ситуація, коли зловмисник перехоплює запит клієнта до сервера (довіреного об'єкта) і відповідає клієнту від імені сервера.

2.3.3 Атака без з'єднання

Для службових повідомлень у мережі часто використовується передача одиночних повідомлень, які потребують підтвердження, тобто не потрібно створення з'єднання. Як приклад – ICMP-повідомлення, дейтаграми UDP.

Атака без встановленого віртуального з'єднання полягає у передачі службових повідомлень від імені мережевих пристроїв, наприклад, від імені адміністратора маршрутизатор (рис. 10.5). Здійснення помилкових керуючих повідомлень може призвести до серйозних порушень роботи розподіленої ПС (наприклад, зміни її конфігурації).

2.3.4 Захист від атак підміни

Використання надійних систем ідентифікації та аутентифікації. Це може бути сертифікована ЕЦП, фаєрволи, механізми VPN (захищеної віртуальної мережі), які будуть розглянуті у наступних розділах.

Рисунок 2.2. Хост порушника передає маршрутизатор керуюче повідомлення від імені адміністратора мережі

2.4 Хибний об'єкт у мережі

2.4.1 Поняття хибного об'єкта

Хибним об'єктом у мережі називають об'єкт, керований зловмисником, який перенаправляється трафік, призначений іншому, легальному об'єкту. Зазвичай перенаправлення відбувається на хост зловмисника, але це може бути й інший об'єкт «призначений» зловмисником. Як правило, хибний об'єкт виконує роль «людини посередині» (Man in the Middle), рис. 2.3.

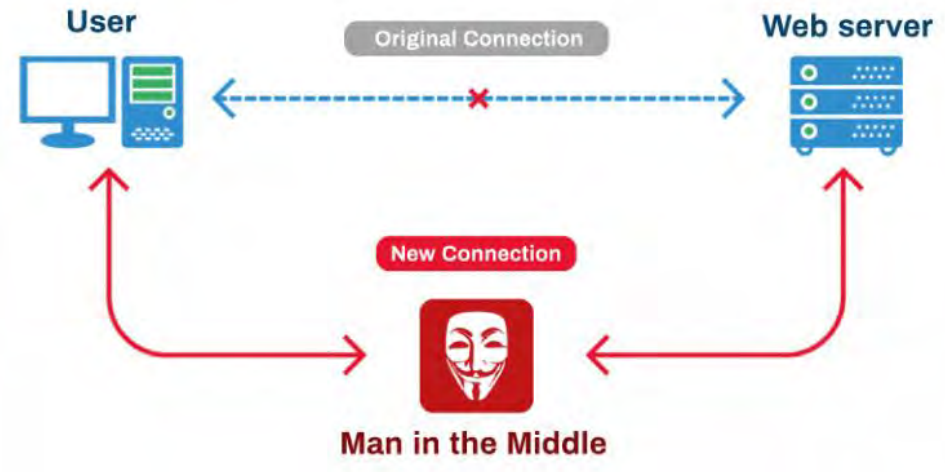


Рисунок 2.3. Атака «людини посередині» (Man in the Middle)

Є кілька способів впровадити в мережу хибний об'єкт.

2.4.2 Впровадження хибного об'єкта шляхом нав'язування хибного маршруту

Маршрутом називається послідовність вузлів мережі (маршрутизаторів), якою дані передаються від джерела до приймача. Кожен маршрутизатор має спеціальну таблицю, звану таблицею маршрутизації, у якій кожному за

адресата вказується подальший маршрут (адреса наступного маршрутизатора).

Основна мета атаки, пов'язаної з нав'язуванням хибного маршруту, полягає в тому, щоб змінити вихідну маршрутизацію на об'єкті мережі так, щоб новий маршрут проходив через хибний об'єкт – хост атакуючого.

Реалізація даної типової віддаленої атаки полягає у несанкціонованому використанні протоколів управління мережею шляхом зміни записів у вихідних таблицях маршрутизації.

Протоколи управління мережею дозволяють:

• обмінюватися інформацією між маршрутизаторами – протокольні повідомлення протоколів маршрутизації (RIP, OSPF тощо);

• повідомляти хости про новий маршрут (ICMP-повідомлення Redirect, Router Advertisement/Solicitation);

• дистанційно керувати маршрутизаторами (спеціальний протокол SNMP – Simple Network Management Protocol).

Для зміни маршрутизації атакуючому необхідно надіслати по мережі спеціальні службові повідомлення від імені мережевих пристроїв, що управляють (наприклад, ICMP-повідомлення від імені маршрутизатора рис. 2.4). В результаті успішної зміни маршруту атакуючий отримає повний контроль над потоком інформації, якою обмінюються два об'єкти мережі.

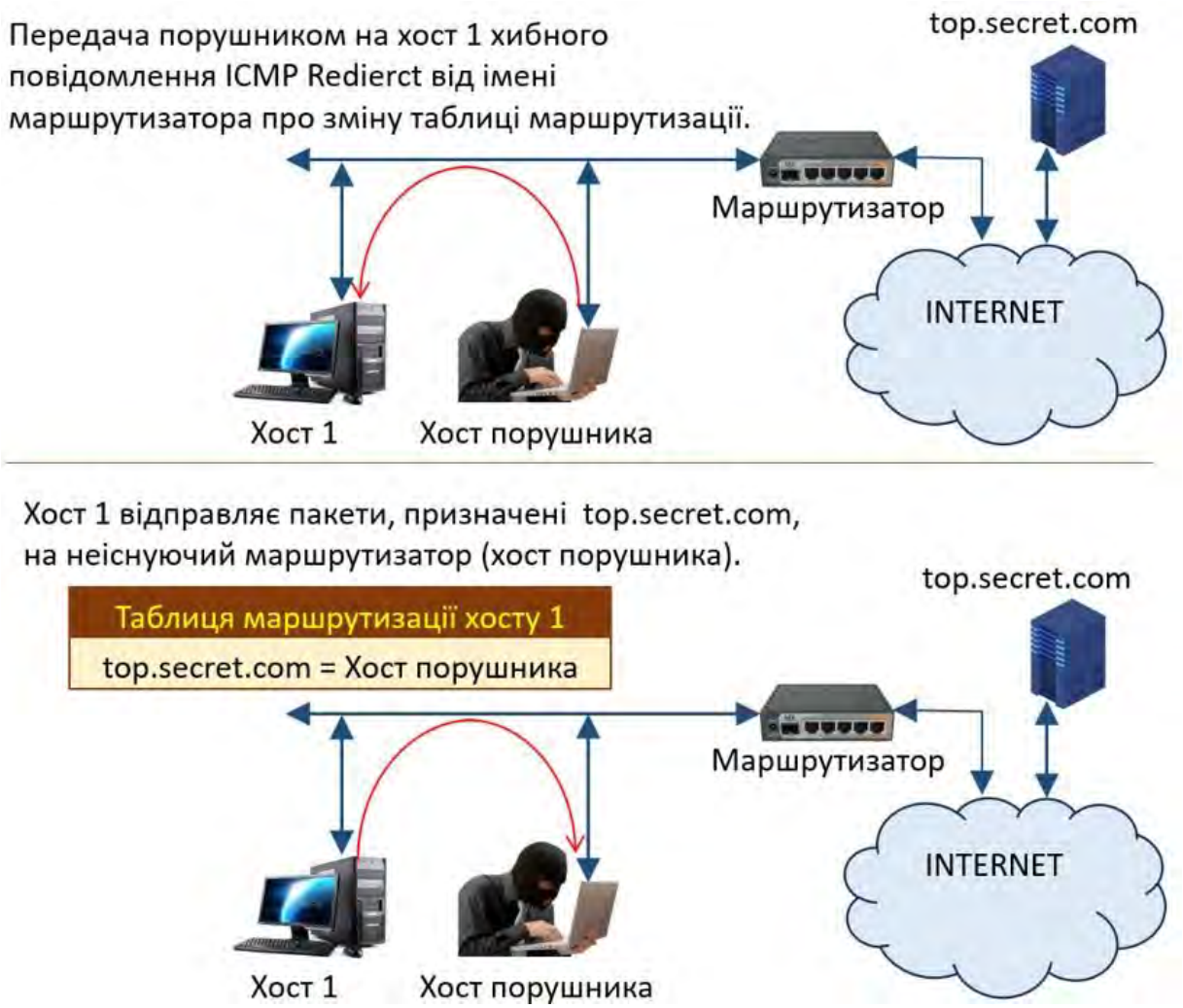


Рисунок 2.4. Нав'язування хосту неправдивого маршруту з використанням протоколу ICMP

Для захисту від цього типу атак рекомендується:

• блокувати (за допомогою фаєрвола) ICMP-повідомлення, що керують маршрутизацією);

• надійно автентифікувати джерела протокольних повідомлень.

2.4.3 Впровадження помилкового об'єкта шляхом використання недоліків алгоритмів віддаленого пошуку

У розподіленій ВС віддалені об'єкти часто не мають достатньо інформації, необхідної для адресації повідомлень (адреса мережевого адаптера, IP-адреса web-сервера і т. д.). Для отримання подібної інформації використовуються різні алгоритми віддаленого пошуку, що полягають у передачі через мережу спеціального виду пошукових запитів і в очікуванні відповідей на запит із шуканою інформацією. Зловмисник може передати від імені легального об'єкта відповідь, у якій вкаже адресу хибного об'єкта (приклад на рис. 2.5). Прикладом запитів, у яких базуються алгоритми віддаленого пошуку, можуть бути ARP- і DNS-запит у мережі Internet (атаки на протокол ARP докладніше у п.3.2).

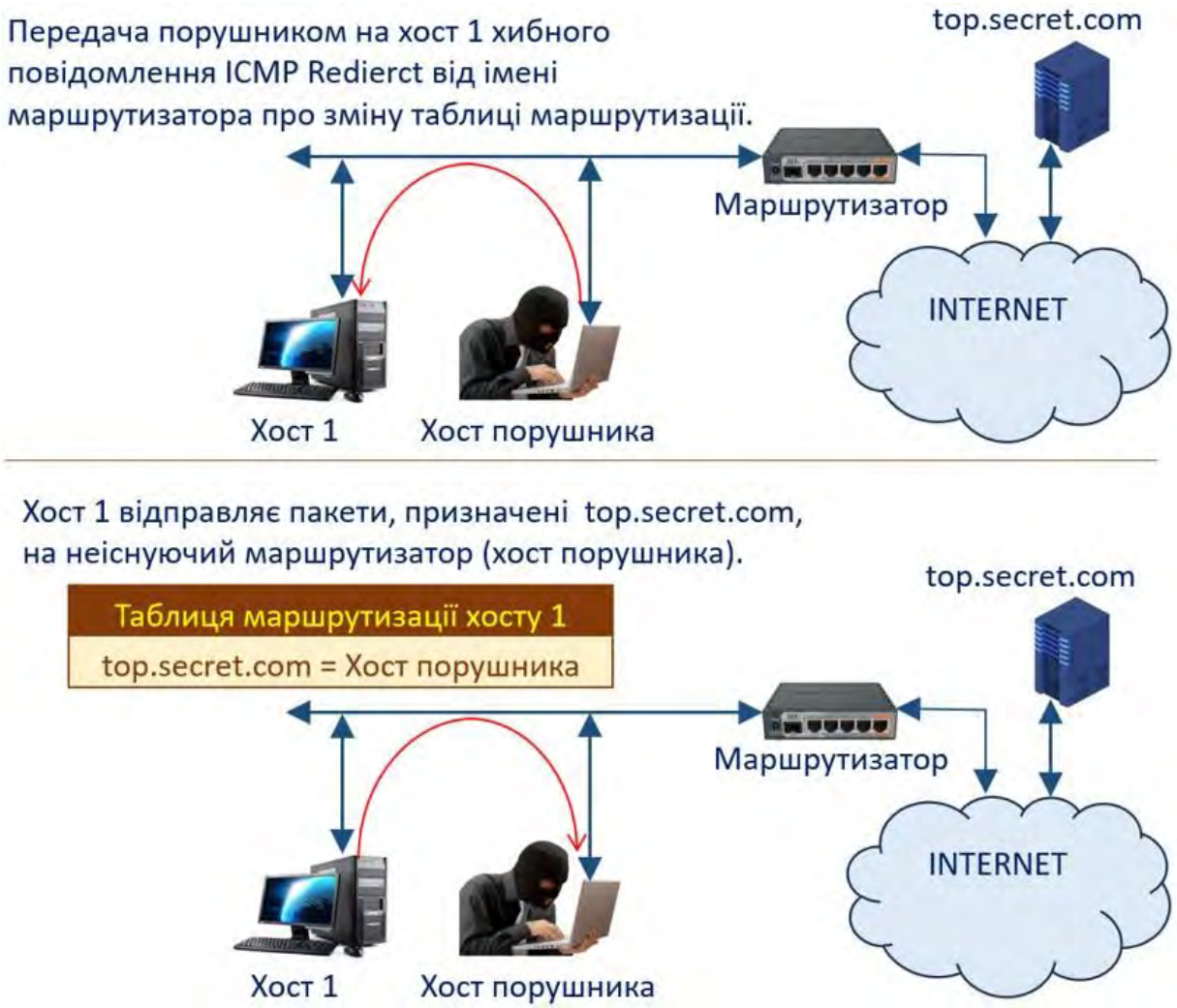


Рисунок 2.5. Атака «Помилковий DNS-сервер»

2.4.4 Використання хибного об'єкта для організації віддаленої атаки на об'єкти мережі

Селекція потоку інформації та збереження її на хибному об'єкті.

У пакетах обміну крім полів даних існують службові поля, які не становлять атакуючого безпосереднього інтересу. Для того, щоб отримати файл, що безпосередньо передається, необхідно проводити на помилковому об'єкті динамічний семантичний аналіз потоку інформації для його селекції.

Модифікація інформації

a) модифікація даних, що передаються;

b) модифікація коду, що передається.

- використання РПС (руйнівних програмних засобів);
- зміна логіки роботи файлу, що виконується.

Підміна інформації

Помилковий об'єкт дозволяє не лише модифікувати, а й підміняти перехоплену ним інформацію. Якщо модифікація інформації призводить до її часткового спотворення, то підміна – її повної зміни.

2.5 Відмова в обслуговуванні (DoS – Denial of Service)

2.5.1 Поняття атаки DoS

Результат застосування цієї віддаленої атаки – порушення на атакованому об'єкті працездатності відповідної служби надання віддаленого доступу, тобто неможливість отримання віддаленого доступу з інших хостів до цієї служби – відмова в обслуговуванні. Зазвичай об'єктом атаки є сервер великої компанії. Завдання сервера полягає в тому, щоб постійно чекати отримання запиту на підключення від віддаленого об'єкта. У разі отримання подібного запиту по можливості передати на об'єкт, що запитав, відповідь, в якій або дозволити підключення, або ні. Зазвичай взаємодія сервера та клієнта відбувається з використанням TCP-з'єднання. Кількість можливих з'єднань обмежена ресурсами сервера (обсяг оперативної пам'яті, швидкодія, пропускна здатність каналів тощо). Завдання атакуючого вичерпати ресурси сервера.

2.5.2 Деякі прийоми реалізації атаки

Якщо в системі не передбачені правила, що обмежують кількість запитів, що приймаються, з одного об'єкта (адреси) в одиницю часу, то атакуючий передає з однієї адреси таку кількості запитів на об'єкт, що атакує, яке дозволить трафік (направлений «шторм» запитів) - рис. 2.6.

Якщо на об'єкті, що атакується, не передбачено засобів автентифікації адреси відправника, то атакуючий передає на цей об'єкт нескінченну кількість анонімних запитів на підключення від імені інших об'єктів. Існують різновиди DoS-атак, спрямованих на заняття (насичення – flood) смуги пропускання каналів атакованого сервера всілякими запитами (ping-flood, icmp-flood, HTTP-flood, SYN-flood тощо).

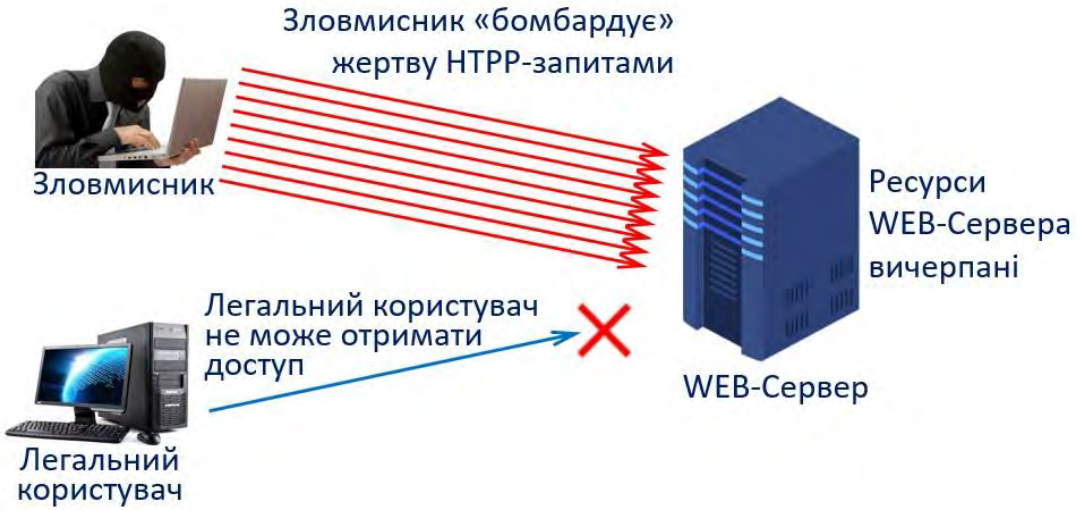


Рис. 2.6. Спрямований «шторм» запитів може вичерпати ресурси сервера

Також існує можливість для атакуючого передача на об'єкт, що атакується, некоректного, спеціально підібраного запиту. В цьому випадку у віддаленій

системі можливе зациклювання процедури обробки запиту, переповнення буфера з подальшим зависанням системи тощо.

2.5.3 Атака DDoS

Атака DoS, яка виконується одночасно з великої кількості комп'ютерів, називається DDoS (Distributed Denial of Service – розподілена атака типу «відмова в обслуговуванні»). Така атака проводиться у разі, якщо потрібно викликати відмову у обслуговуванні добре захищеної великої компанії чи урядової організації – рис. 2.7.

Насамперед зловмисник атакує ряд вузлів і отримує на них права адміністратора. На захоплені вузли встановлюються троянські програми. Такі комп'ютери називаються комп'ютерами-зомбі (або агентами), а мережі таких комп'ютерів – бот-мережами (ботнетами). Існують два типи агентів: «майстри» (master) та «демони» (daemon). Зловмисник керує невеликою кількістю «майстрів», які, у свою чергу, командують «демонами».

Далі зловмисник відправляє певні команди захопленим комп'ютерам і ті, у свою чергу, здійснюють потужну DoS-атаку на цільовий комп'ютер.

Демони встановлюються шляхом використання на скомпрометованих вузлах різних вразливостей, які дозволяють отримати права адміністратора на вузлі із встановленим демоном. Як тільки демон встановлено, він повідомляє про це «майстра» (зазвичай трьох чи чотирьох). Після отримання певних команд від зловмисника «майстер» програмує «демона» на виконання відповідних дій проти жертви. Ці команди містять адресу жертви, тип атаки, час та тривалість атаки.

Крім того, при атаці можлива заміна адреси відправника ворожих пакетів, що також негативно впливає на ефективність контрзаходів.

Існують програми для добровільної участі в DDoS-атаках.

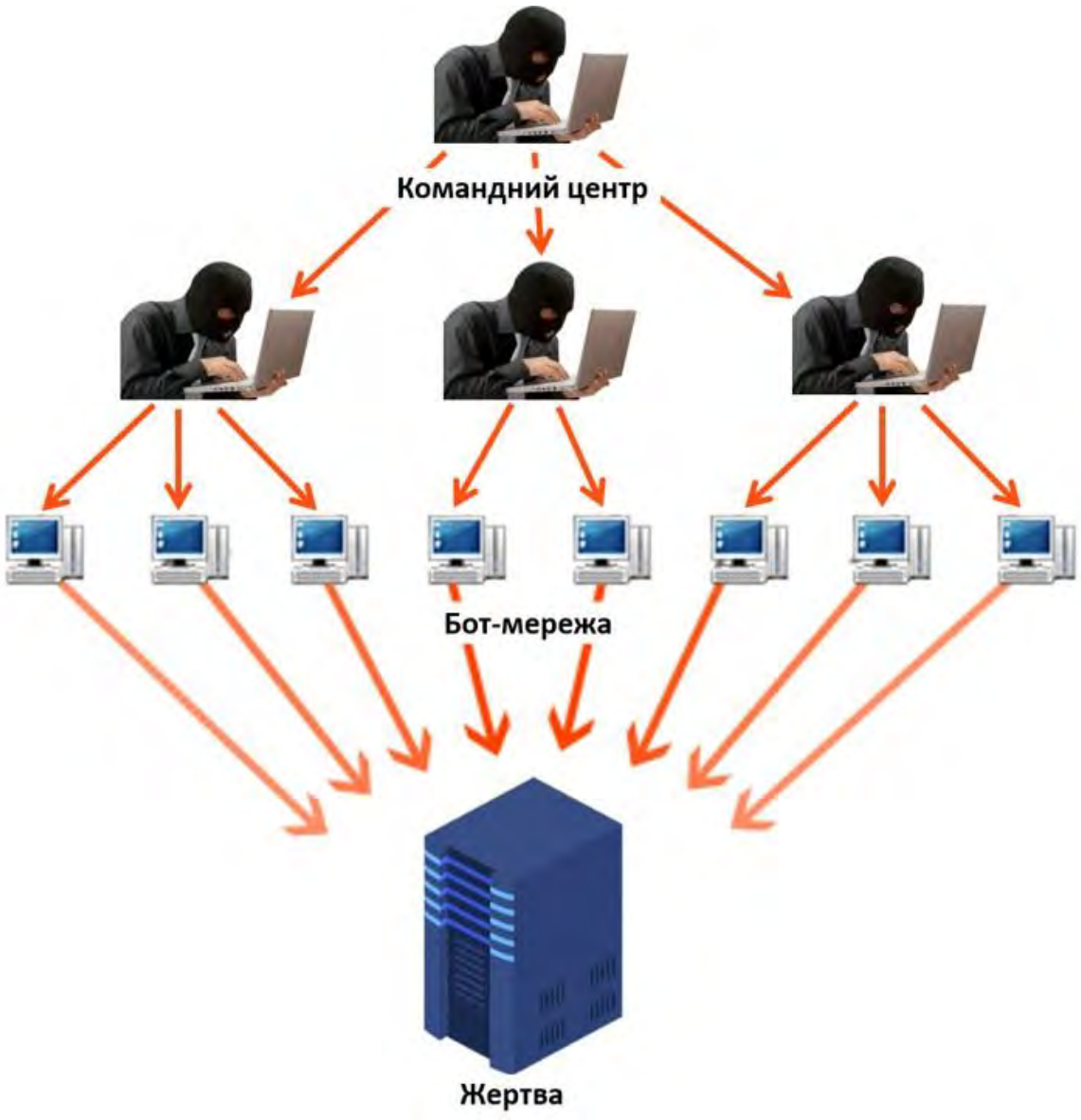


Рисунок 2.7. Архітектура DDoS-атаки

Складність блокування подібних атак полягає в тому, що виявлення та блокування одного або декількох «майстрів» або «демонів» не призводить до закінчення атаки, оскільки кожен «демон» діє незалежно від інших і, отримавши відповідні команди від «майстра», вже не потребує подальшої підтримки зв'язку з ним. Тобто «демон» працює автономно, що суттєво ускладнює виявлення та блокування всіх демонів, що беруть участь у розподіленій атаці.

2.5.4 Захист від DoS-атак

Для запобігання DOS- та DDoS-атакам зазвичай використовуються апаратні методи захисту периметра мережі – фаєрвол у поєднанні з системою виявлення вторгнень (Intrusion Detection Systems, IDS). Такі механізми захисту можуть діяти як мережеві фільтри таким чином, що послідовно аналізують прохідний трафік, виявляючи нестандартну мережну активність та помилки. До аналізованих шаблонів нестандартного трафіку входять всі

відомі на сьогоднішній день методи атак, у тому числі реалізовані і за допомогою розподілених бот-мереж

2.5.5 Запитання до розділу

1) Характеристика та механізм реалізації типової віддаленої атаки «Аналіз мережного трафіку».
2) Характеристика та механізм реалізації типової віддаленої атаки «Підміна довіреного об'єкта».
3) Характеристика та механізм реалізації типової віддаленої атаки «Впровадження хибного об'єкта шляхом нав'язування хибного маршруту».
4) Характеристика та механізм реалізації типової віддаленої атаки «Впровадження хибного об'єкта шляхом використання недоліків алгоритмів віддаленого пошуку».
5) Характеристика та механізм реалізації типової віддаленої атаки «Відмова в обслуговуванні».
6) Використання хибного об'єкта для організації віддаленої атаки у мережі.

РОЗДІЛ 3 МЕХАНІЗМИ РЕАЛІЗАЦІЇ ДЕЯКИХ МЕРЕЖОВИХ АТАК

3.1 Сніффінг (прослуховування)

3.1.1 Поняття сніффінгу

Атаки прослуховуванням називаються сніффінг. А програма, яка реалізує процес прослуховування, називається сніффером.

Сніффер бачить лише дані, що входять і виходять від машини, на якій він встановлений. Решта інформації, що протікає в мережі, йому недоступна. Сніффер може збирати трафік у тому сегменті мережі, в якому він встановлений, якщо переключити його мережну картку на потрібний режим роботи. Мережеву карту можна встановити в один із таких режимів:

• «розбірливий» – збір даних, що йдуть лише за MAC-адресою цієї картки;

• «нерозбірливий» (promiscuous mode) – збирання всього трафіку, який проходить через мережну карту комп'ютера, на якому запущено сніффер.

Приклади сніферів:

• tcpdump – безкоштовний сніффер для багатьох платформ UNIX;

• windump – безкоштовна версія tcpdump для Windows;

• sniffit – безкоштовний сніффер для багатьох платформ UNIX (рис. 3.1);

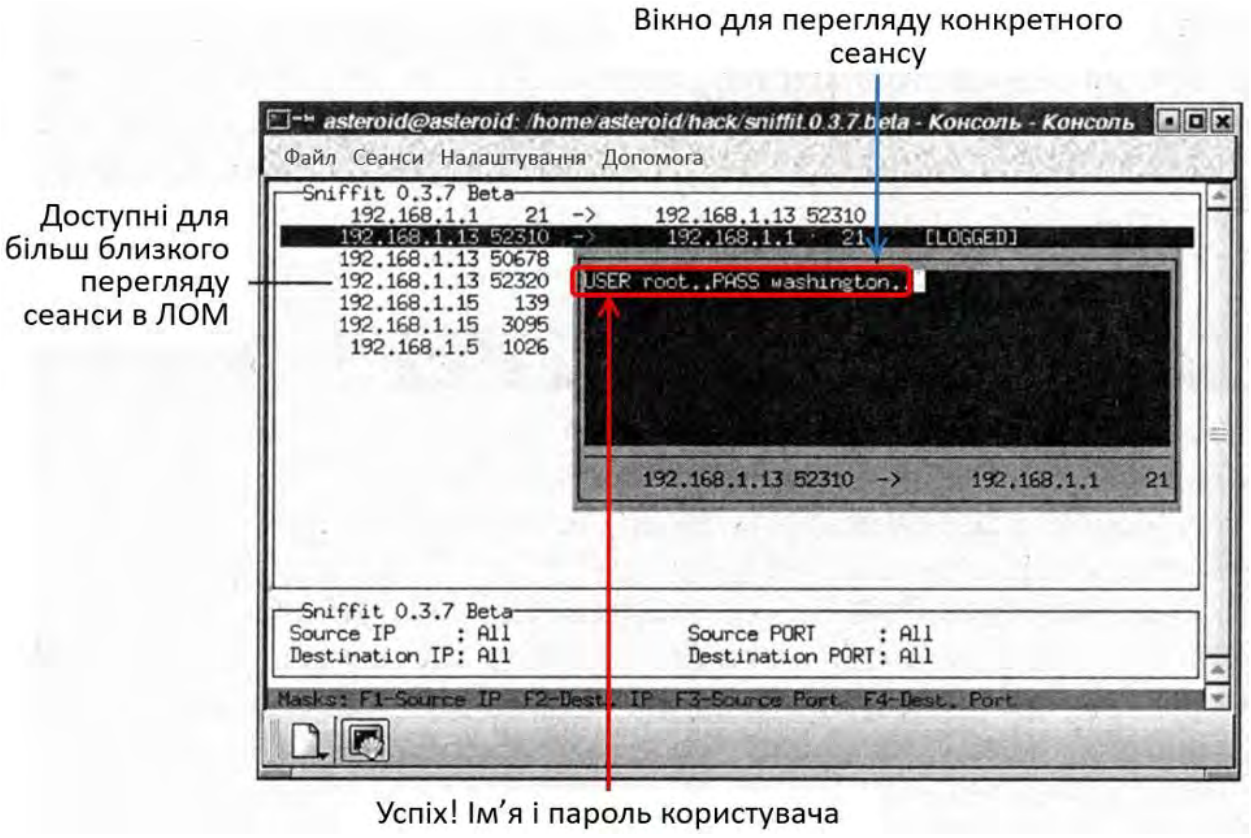


Рисунок 3.1. Використання Sniffit в інтерактивному режимі для перехоплення імен та паролів

• dsniff – безкоштовний комплект інструментальних засобів, створений на базі сніффера, що працює під UNIX

• snort – безкоштовний багатоплатформений сніффер.

3.1.2 Сніффінг через концентратори (пасивний сніффінг)

Концентратор працює у сегменті першого рівня, тобто. створює трансляційне середовище, доступне всім системам ЛВС (дивись рис. 1.8).

Захист від пасивного сніффінгу:

• Використання програм, які виявляють прослуховування (AntiSniff). Принцип роботи антисніффера полягає у вимірі часу реагування хостів на мережеві запити та визначенні, чи не доводиться хостам обробляти «зайвий» трафік.

• Шифрування трафіку.

3.1.3 Сніффінг через комутатори (активний сніффінг)

Комутатор переглядає MAC-адресу призначення кожного кадру, що проходить через нього, направляючи цей кадр тільки на порт, до якого приєднано хост із зазначеною адресою (дивись рис. 1.8). На хост зловмисника надходять лише кадри, направлені на його адресу (і на широкомовну адресу). Однак існують прийоми, які дозволяють зловмиснику прослуховувати трафік мережі, де використовуються комутатори, наприклад, створення перешкод комутаторам за допомогою перевантаження, хибний ARP-сервер (ця атака розглянута в п. 3.2).

Створення перешкод комутаторам за допомогою перевантаження. Якщо на вхід комутатора надходить кадр з MAC-адресою відправника, якого немає в пам'яті комутатора, то комутатор запам'ятовує цю адресу (і її відповідність порту, звідки ця адреса надійшла) у буферній пам'яті. В подальшому, якщо на який-небудь порт комутатора надійде кадр з цією MAC-адресою призначення, то комутатор, переглядаючи буферну пам'ять знайде відповідний MAC-адресу порт і направить кадр на цей порт. Зловмисник використовує метод, який ґрунтується на переповненні пам'яті комутатора помилковими MAC-адресами відправника (наприклад, за допомогою програми Macof з пакету Dsniff). При виснаженні ресурсів пам'яті деякі моделі комутаторів починають переадресовувати дані у всі ланки мережі, пов'язані з комутатором.

Для захисту від цього механізму сканування рекомендується не використовувати моделі комутаторів, схильні до вищезазначеного недоліку (інші моделі комутаторів при виснаженні ресурсу пам'яті перестають запам'ятовувати наступні MAC-адреси.

Інший прийом, що дозволяє реалізувати активний сніффінг через комутатори пов'язаний з атакою «хибний сервер ARP», яка розглянута далі.

3.2 Атака «Хибний ARP-сервер»

3.2.1 ARP-протокол

ARP (Address Resolution Protocol) – протокол канального рівня вирішує проблему перетворення відомої відправнику IP-адреси на фізичну адресу (MAC-адресу).

Підготовлений для відправлення IP-пакет повинен бути поміщений у кадр канального рівня і відправлений за MAC-адресою того хоста, якому відповідає IP-адреса призначення (якщо цей хост у тій же мережі, де і хост-відправник) або MAC-адресою маршрутизатора (якщо хост призначення в іншій мережі). Відповідність між IP-адресою призначення та відповідною MAC-адресою хост-відправник шукає у власній ARP-таблиці (рис. 3.2).

IP-адреса	MAC-адреса	Тип
172.16.10.253	00:1C:C5:34:B3:01	Динамічний
172.16.10.34	00:2A:25:1F:93:14	Статичний

Рисунок 3.2. Приклад ARP-таблиці мережного вузла

Якщо для необхідної IP-адреси в ARP-таблиці відсутня MAC-адреса, то всім машинам у мережі надсилається кадр з ARP-запитом (з широкомовною MAC-адресою в заголовку) – рис. 3.3.

Поле	Значення
Тип мережі	1
Тип протоколу	2048
Довжина локальної адреси	6
Довжина глобальної адреси	4
Операція	1
Локальна адреса відправника	00:2A:25:1F:93:14
Глобальна адреса відправника	172.16.10.34
Локальна адреса отримувача	00:00:00:00:00:00
Глобальна адреса отримувача	172.16.10.253

Рисунок 3.3. Формат повідомлення «ARP-запит»

Кожна машина мережі, що прийняла ARP-запит, порівнює власну IP-адресу з IP-адресою у запиті. Якщо IP-адреса збіглася, то за MAC-адресою відправника запиту надсилається відповідь, що містить як IP-адресу машини, що відповіла, так і її MAC-адресу (рис. 3.4). З отриманої відповіді

формується запис (тип – «Динамічний») в ARP-таблиці – рис. 3.2 Слід врахувати, що динамічні записи ARP-таблиці зберігаються від кількох хвилин до кількох десятків хвилин, після чого видаляються автоматично. Статичні записи в ARP-таблиці формуються спеціальною командою arp і також видаляються цією командою.

Поле	Значення
Тип мережі	1
Тип протоколу	2048
Довжина локальної адреси	6
Довжина глобальної адреси	4
Операція	2
Локальна адреса відправника	00:1C:C5:34:B3:01
Глобальна адреса відправника	172.16.10.253
Локальна адреса отримувача	00:2A:25:1F:93:14
Глобальна адреса отримувача	172.16.10.34

Рисунок 3.4. Формат повідомлення «ARP-відповідь»

3.2.2 Атака «Хибний ARP-сервер» з перехопленням ARP-запиту

Послідовність такої атаки показано на рис. 3.5.

1) Зловмисник, об'єкт якого знаходиться в сегменті мережі атакованого (шар 2 – широкомовний домен другого рівня OSI) чекає на ARP-запит (рис. 3.5 – 1 Фаза очікування ARP-запиту).

2) Зловмисник отримує надісланий широкомовний запит (рис. 3.5 – 2 Фаза реалізації загрози), який проходить і через концентратори, і через комутатори (але не через маршрутизатори!). У ARP-запиті вказується IP-адреса об'єкта, для якого потрібно знайти відповідну MAC-адресу (на рис. 3.3 поле «Глобальна адреса відправника»).

3) При отриманні такого запиту атакуючий передає по мережі на хост, який цей запит відправив, підроблену ARP-відповідь (рис. 3.5 – 3 Фаза прийому, аналізу ...), де вказується MAC-адреса мережевого адаптера атакуючої станції (хибного ARP-сервера) та MAC-адреса, на якій прийматиме пакети хибний ARP-сервер (на рис. 3.4 – Локальна адреса відправника).

Слід уточнити, що запит отримає не лише атакуючий, а й «легальний» об'єкт, який також надішле ARP-відповідь, але вже з «правильною» MAC-адресою. Успіх атаки визначиться тим, чия відповідь прийде раніше – атакуючого чи «легального» об'єкта.

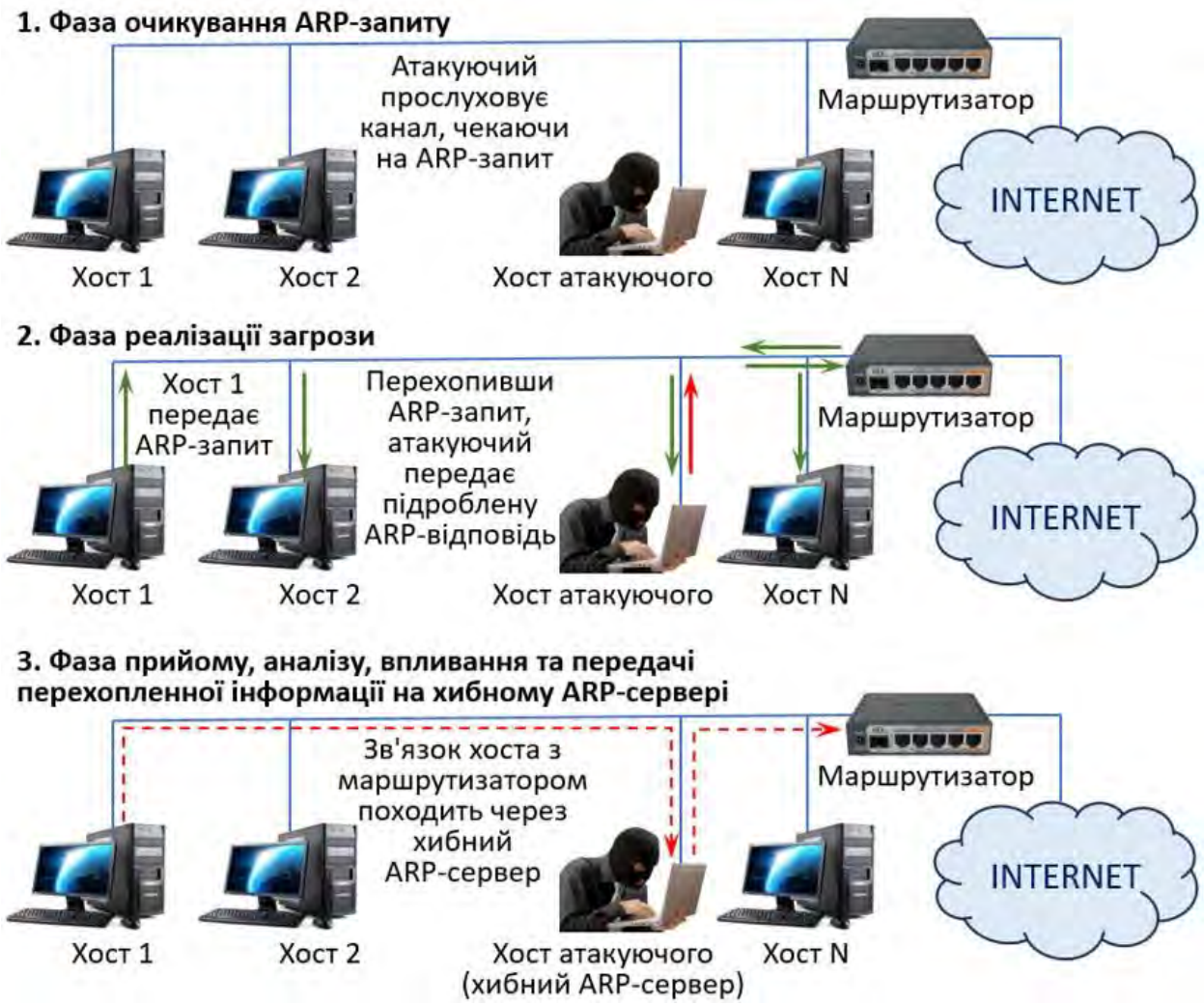


Рисунок 3.5. Атака «Помилковий ARP-сервер» з перехопленням ARP-запиту

Якщо атака буде вдалою, то надалі весь потік інформації між суб'єктом та об'єктом взаємодії проходитиме через хибний ARP-сервер (рис. 3.5 – 3 Фаза прийому, …).

3.2.3 Атака «Хибний ARP-сервер» зі штормом підроблених відповідей

У цьому випадку зловмисник періодично передає на об'єкт, що атакується, заздалегідь підготовлену підроблену ARP-відповідь без прийому пошукового запиту. При передачі атакованим об'єктом пошукового ARP-запиту хибна ARP-відповідь атакуючого негайно матиме успіх.

3.2.4 Використання режиму «Самовільний ARP» (gratuitous ARP)

Протокол ARP можна налаштувати на режим «самовільного ARP» (gratuitous ARP). У такому режимі хост сприймає вхідні повідомлення формату ARP-відповідь без попередньої посилки ARP-запиту, як посібник до дії. Зловмисник може скористатися цим, реалізувавши атаку під назвою ARP-spoofing (рис. 3.6).

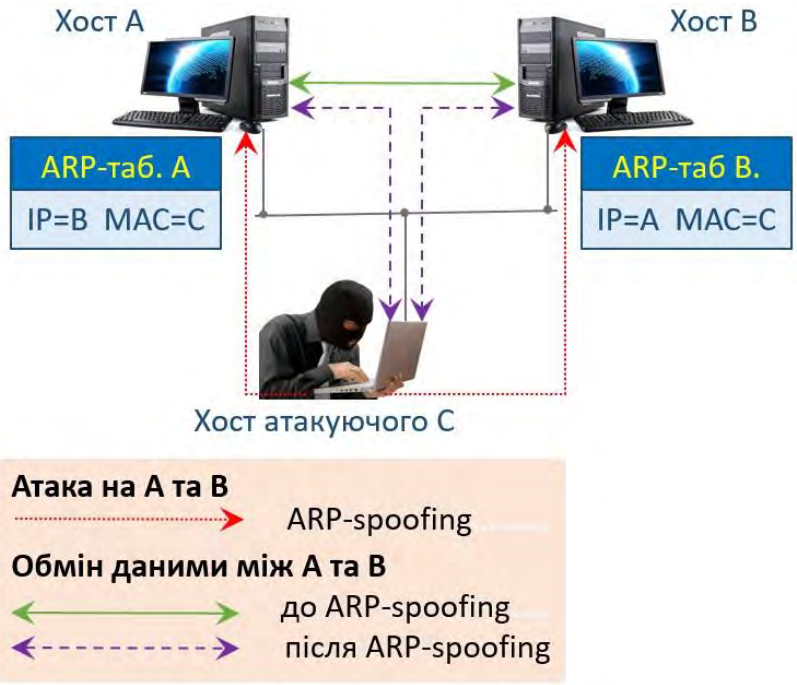


Рисунок 3.6. Схема атаки «ARP-spoofing»

До виконання ARP-spoofing в ARP-таблицях хостів A та B (ці таблиці на рис. 3.6 не показані) існують записи з IP- та MAC-адресами один одного. Обмін інформацією здійснюється безпосередньо між вузлами A і B (зелена стрілка).

В ході виконання ARP-spoofing'а хост C зловмисника, що виконує атаку, відправляє (червоні стрілки) ARP-відповіді (без отримання запитів):

• хосту A: з IP-адресою хосту B та MAC-адресою хосту C;

• хосту B: з IP-адресою хосту A та MAC-адресою хосту C.

У силу того, що вузли підтримують самовільну ARP (gratuitous ARP), вони модифікують власні ARP-таблиці і поміщають туди записи, де замість справжніх MAC-адрес хостів A і B стоїть MAC-адреса хосту зловмисника C (саме ці таблиці в спрощеному вигляді показані на рис. 3.5).

Після того як атака виконана, якщо, наприклад, хост A хоче передати пакет вузлу B, то A дивиться в свою ARP-таблицю, знаходить запис з IP-адресою вузла B, вибирає звідти MAC-адресу (а там вже MAC-адреса хосту C – комп'ютера зловмисника) і передає пакет за цією фізичною адресою хосту C. Хост C потім може ретранслювати пакет (проаналізувавши і, можливо, змінивши його вміст) тому, кому він дійсно адресований – тобто вузлу B (сині стрілки).

3.2.5 Захист від атаки «Хибний ARP-сервер»

• Використання таких програм, як arpwatch, BitComet, AntiARP. Ці програми відстежують активність ARP на заданих інтерфейсах і повідомляють адміністратора про помічені порушення. Можуть виявити атаку ARP-spoofing, але не можуть запобігти її. Для запобігання атаки потрібне втручання адміністратора мережі.

• Організація VLAN (Virtual Local Area Network, віртуальна локальна мережа). Якщо в локальній мережі є поділ на декілька VLAN, то атака ARP-spoofing може бути застосована лише до комп'ютерів, що знаходяться в одній VLAN. Ідеальною ситуацією, з погляду безпеки, є наявність лише одного комп'ютера та інтерфейсу маршрутизатора в одному сегменті VLAN. Атака ARP-spoofing для такого сегмента неможлива.

• Використання статичних ARP-таблиць. Можна уникнути атаки ARP-spoofing шляхом настроювання ARP-таблиці вручну (тип ARP-запису «Статичний» – рис. 3.2). Тоді зловмисник не зможе оновлювати ARP-таблиці шляхом посилки ARP-відповідей на інтерфейси комп'ютерів, оскільки не формуватимуться ARP-запити.

• Не налаштовувати протокол ARP на роботу в режимі «самовільний ARP».

3.3 Розвідка у мережі

3.3.1 Відображення мережі

На етапі розвідки атакуючий хоче більше дізнатися про мережу: оцінити мережу обраної компанії, визначити необхідні адреси та отримати певне уявлення про топологію мережі. Досвідчений атакуючий нарисує схему інфраструктури мережі, намагатиметься поставити себе на місце її проектувальника, щоб виявити вразливі хости (головні комп'ютери), маршрутизатори та брандмауери.

Незалежно від того, відображають і сканують периметр або систему внутрішньої мережі, використовуються одні й ті ж інструменти та одна методологія.

Пошук активних хостів. Для того щоб побудувати схему системи, до якої отримано доступ, атакуючий спробує визначити активні хости, відправляючи на всі можливі адреси мережі програму ping (Packet InterNet Groper) – інструмент, що використовує ICMP-повідомлення Echo Request. Атакуючий здатний надіслати цей запит на кожну адресу мережі. Якщо відповідь (ICMP-повідомлення Echo Replay) прийде, то за цією адресою знаходиться активна машина. В іншому випадку можна припустити, що адреса не прослуховується (або фільтрується). Звичайно, що атакуючим не потрібно зондувати всю мережу за допомогою ping вручну, вони задіяють автоматичний інструмент для охоплення всіх цільових адрес при пошуку активних хостів.

Трасування маршрутів. Як тільки атакуючий виявить активні хости, він захоче дізнатися про топологію вашої мережі та застосує техніку трасування маршрутів (traceroute), щоб визначити маршрутизатори та шлюзи, які становлять інфраструктуру мережі. Трасування маршрутів ґрунтується на значенні поля TTL (Time to Live – час життя пакету, що передається), розташованого в IP-заголовку. Поле TTL вказує, скільки транзитних вузлів має пройти пакет, перш ніж буде вилучений маршрутизатором. Коли маршрутизатор отримує будь-який IP-пакет, він спочатку зменшує значення в

полі TTL на одиницю. Якщо величина TTL дорівнює нулю, маршрутизатор відправляє творцю вхідного пакета ICMP-повідомлення Time Exceeded, яке означає таке: «Вибачте, але час життя даного пакета був недостатнім для того, щоб досягти адресата».

Атакуючі застосовують TTL для визначення шляху, пройденого пакетом по мережі. Як показано на рис. 3.7 від вихідної машини відправляється IP-пакет (с вкладеним ICMP-повідомленням) зі значенням TTL в заголовку рівним 1 Перший маршрутизатор, отримавши пакет, зменшить значення TTL до нуля і поверне ICMP-повідомлення Time Exceeded. Адреса джерела, яке надіслало таке повідомлення – це IP-адреса першого маршрутизатора на шляху до адресата. Потім від вихідної машини відправляється пакет зі значенням TTL рівним 2 Перший маршрутизатор зменшить TTL на одиницю і перешле пакет далі. Другий маршрутизатор зменшить TTL до нуля і поверне ICMP-повідомлення Time Exceeded, тим самим повідомивши адресу другого транзитного вузла. Далі все повторюється: від вихідної машини відправляються пакети з значенням TTL, що збільшується, до тих пір, поки пакет не потрапить до адресата. Тоді на вихідній машині будуть IP-адреси всіх маршрутизаторів, розташованих між вихідною машиною та комп'ютером адресата.

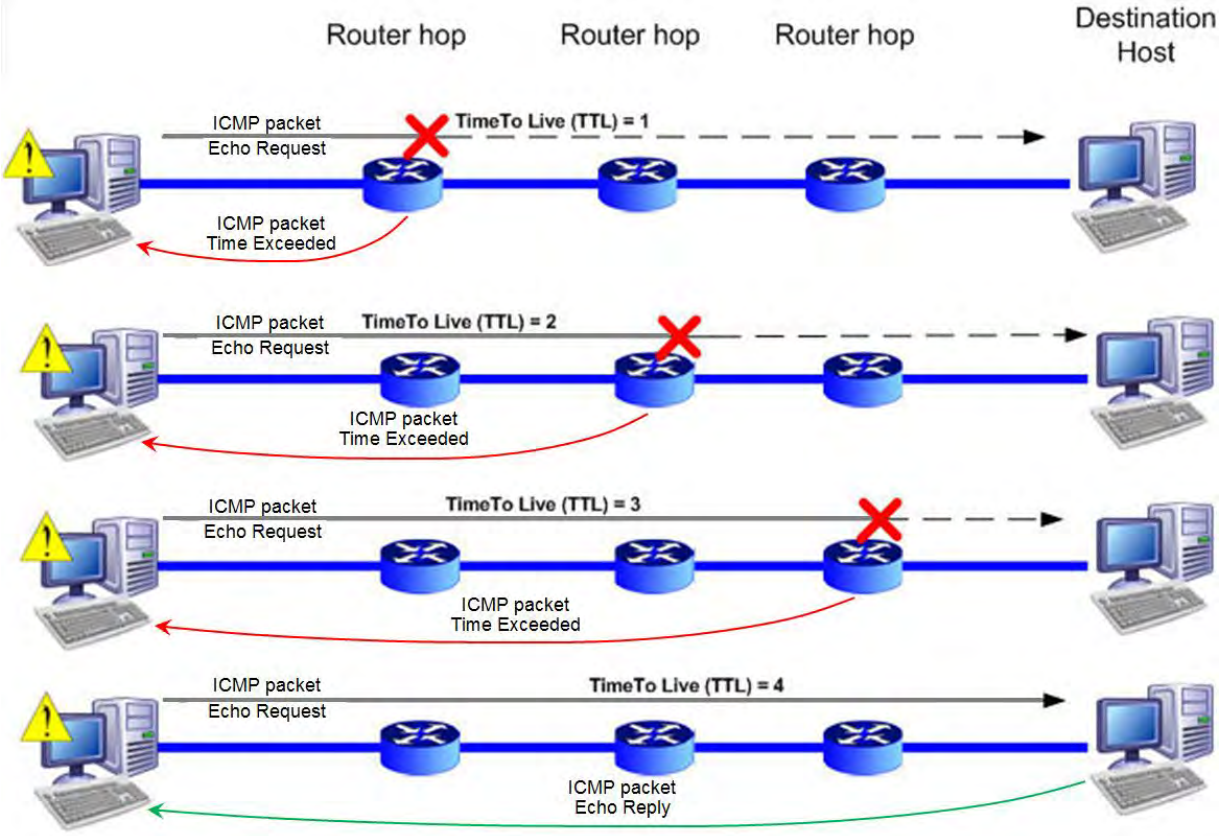


Рисунок 3.7. За допомогою трасування маршрутів нескладно визначити шлях від джерела до адресата

Слід врахувати, що при трасуванні маршруту можна використовувати не тільки ICMP-повідомлення Echo Request, як це показано на рис. 3.7, але й інші типи ICMP-повідомлень та IP-пакетів.

Для автоматизації описаного процесу багато систем UNIX містять версію програми traceroute, яка надсилає UDP-дейтаграми зі збільшенням значенням у полі TTL (заголовка IP-пакета, в який вкладена UDP-дейтаграма) і одночасно шукає відповідні ICMP-повідомлення Time Exceeded. До операційних систем сімейства Windows також включений подібний інструмент під ім'ям tracert. Утиліта tracert відправляє ICMP-повідомлення (а не UDP-дейтаграми) зі збільшенням значення TTL, очікуючи повернення ICMP-повідомлення Time Exceeded. На рис. 3.8 показано виведення програми tracert у Windows.

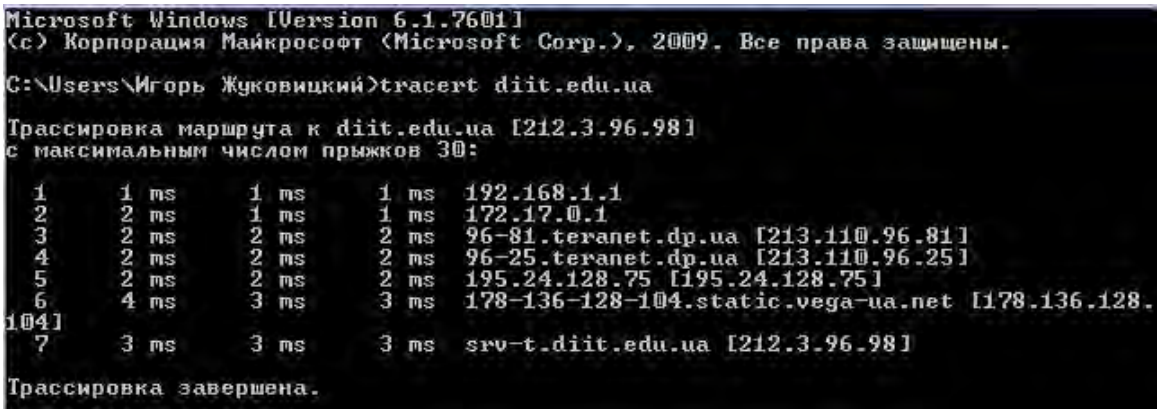


Рисунок 3.8. Виведення програми tracert у Windows

Атакуючий скористається трасуванням маршрутів визначення шляху до кожного хосту, виявленому у процесі атаки утилітою ping. Порівнюючи результати трасування маршрутів для кожної цілі і узгоджуючи між собою різні маршрутизатори і шлюзи, атакуючий зможе відтворити топологію вашої мережі.

3.3.2 Способи захисту проти відображення мережі

Для того, щоб завадити атакуючому визначити топологію вашої мережі за допомогою ping, traceroute та подібних інструментів, необхідно відфільтровувати повідомлення, за допомогою яких ці програми отримують необхідну інформацію. Зробити це можна за допомогою міжмережевих екранів та фільтрації пакетів маршрутизатором. Для цього фільтри маршрутизаторів необхідно налаштувати так, щоб вхідні ICMP-повідомлення типу «Echo Request» пропускалися лише в тому випадку, якщо були відправлені системою провайдера. Доцільно блокувати вихідні від маршрутизаторів ICMP-повідомлення типу «Time Exceeded», щоб перешкодити роботі функції traceroute.

3.3.3 Сканування портів. Стандартне звернення до портів

IP-пакети, що надходять на транспортний (TCP/UDP) рівень хоста, організуються операційною системою у вигляді багатьох черг до точок входу різних мережних додатків цього хоста. У термінах TCP/IP такі системні черги називають портами. На кожному хості є 2^{16-1} портів TCP та 2^{16-1} портів UDP (порт 0 – резервний). Якщо мережний додаток активний, то порт, пов'язаний з

цією програмою, вважається «відкритим», при закритті мережного додатка порт, пов'язаний з ним, переходить у стан «закритий».

Приклади стандартних номерів портів:

• TCP-порт 21 – протокол передачі файлів (FTP);

• TCP-порт 23 – telnet;

• TCP-порт 25 – простий протокол електронної пошти (SMTP);

• TCP-порт 80 – Всесвітнє павутиння (WWW, протокол HTTP

• TCP-порт 666 – Doom (комп'ютерна гра).

Мета сканування – визначити, які порти хосту відкриті, тобто, які програми запущені. Після того, як буде складено список активних (відкритих) портів, починається фаза активних дій.

Для автоматизації процесу сканування використовуються спеціальні програми – сканери портів. Один з відомих сканерів портів – програма Nmap, на прикладі якої будемо розглядати різноманітні прийоми сканування.

UDP-сканування. При скануванні UDP Nmap направляє UDP-пакет на кожен порт сканованої машини. Якщо у відповідь прийде ICMP-повідомлення типу «Порт недоступний», значить, порт закритий. В іншому випадку Nmap вважатиме порт відкритим. На нещастя (для атакуючого), такий метод вкрай ненадійний, оскільки досліджувана система або мережа вправі взагалі не надсилати ICMP-повідомлення типу «Порт недоступний». Тому деякі порти легко прийняти за відкриті, хоч це й не так.

Для додаткової перевірки порту, який не визначений як закритий, можна скористатися відповідним клієнтом, відправивши на службу, яка обслуговується даним портом, будь-який запит і за реакцією служби перевірити, чи справді цей порт прослуховується.

TCP-сканування базується на протоколі створення TCP-з'єднання, який отримав назву «триетапне квитування».

Протокол створення TCP-з'єднання крім адрес порту-джерела та порту-отримувача в заголовку TCP-сегменту використовує спеціальні контрольні біти заголовка – прапори:

• URG (urgent pointer) – використовувати покажчик терміновості, що має особливе значення в полі TCP-заголовка;

• ACK (acknowledgement) – біт підтвердження, що використовується для підтвердження прийому попередніх пакетів;

• PSH (push) – функція «проштовхування», застосовується для швидшого переміщення даних на TCP-рівні;

• RST (reset) – розрив внаслідок помилки;

• SYN (synchronize) – синхронізація номерів послідовності, що використовується при встановленні сеансу зв'язку;

• FIN – розрив з'єднання, якщо від відправника не надходить більше жодних даних.

Протокол триетапного квитування (його елементи зображені на рис. 2.1) реалізується обміном трьох службових сегментів TCP між клієнтом і

сервером. На рисунку 2.1 показані лише характерні прапори та номери байтів у потоці даних у TCP-заголовку.

Протокол дозволяє визначити початкові номери послідовності бітів ISN_A, ISN_B (у потоці даних, якими клієнт А та сервер В будуть обмінюватися в TCP-сеансі). При скануванні ці номери не аналізуються, аналізуються зазвичай лише прапори та можливі ICMP-відповіді.

«Ввічливе» сканування: TCP-connect. Очевидний метод, заснований на принципах створення TCP-з'єднання, що складається в послідовній передачі на різні порти об'єкта сканування TCP- SYN-запитів на створення з'єднання (рис. 3.9). Якщо порт відкритий, то на даний скануючий запит буде отримано відповідь TCP-SYN-ACK; якщо порт закритий – відповіддю буде TCP RST чи ICMP-повідомлення про відсутність порту. У разі відкритого порту атакуючий завершує триетапне квитування (TCP-ACK) та розриває з'єднання, відправляючи TCP-FIN.

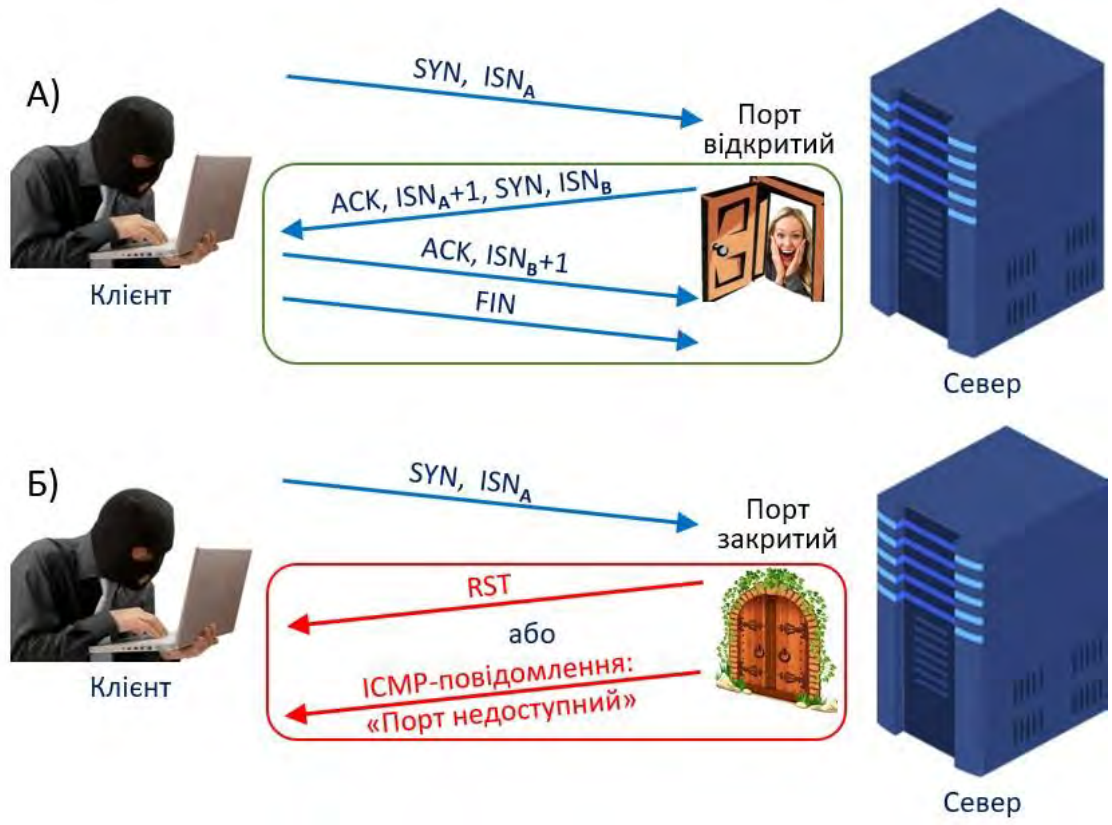


Рисунок 3.9. «Ввічливе» сканування:
А) випадок відкритого порту; Б) випадок закритого порту

Недолік:

• легко можна знайти на об'єкті сканування, тому що кожне TCP-з'єднання фіксується у системному журналі;

• досить великий час сканування.

«Напіввідкрите» сканування: TCP-SYN. На відміну від попереднього методу, атакуючий у разі відкритого порту (отримання у відповідь на TCP-SYN відповіді TCP-ACK) завершує сеанс (третій етап) передачею не

стандартного TCP-ACK, а передачею TCP-RST, перериваючи з'єднання до того, як його було встановлено (рис. 3.10).

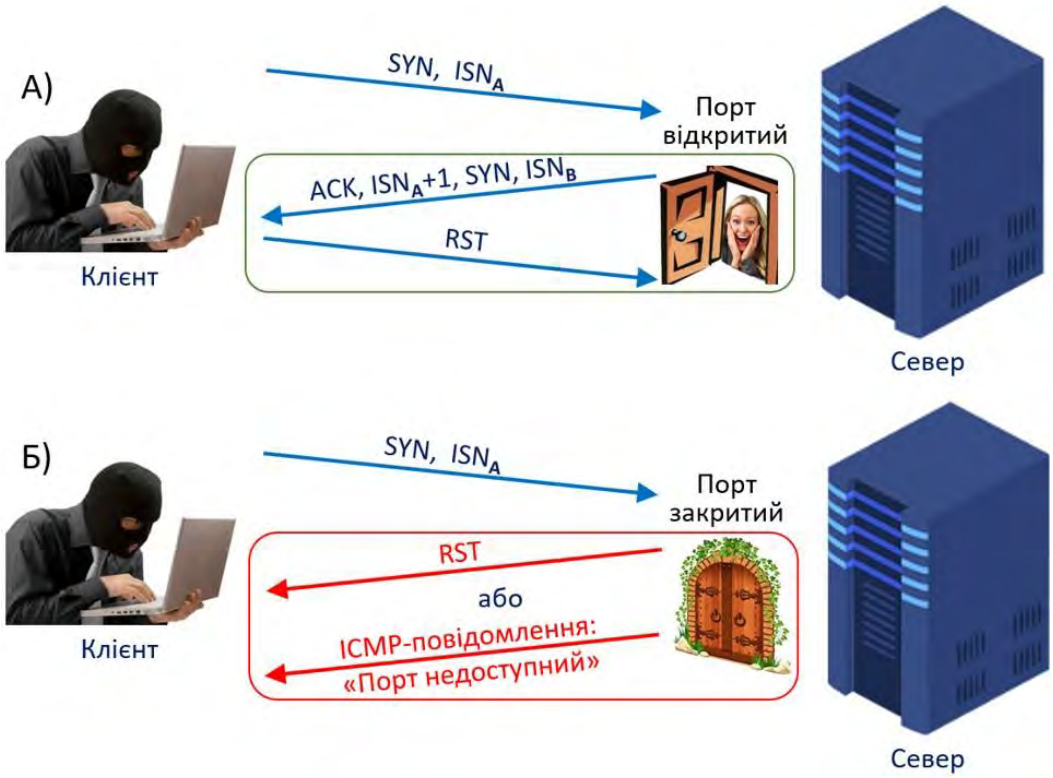


Рисунок 3.10. «Напіввідкрите» сканування:
А) випадок відкритого порту; Б) випадок закритого порту

Така операція у системному журналі не реєструється, так як третій етап протоколу відсутній та з'єднання не встановлюється. Але якщо в системі, в якій відбувається сканування, є брандмауер, то таке сканування також може бути зареєстроване.

Другою перевагою TCP-SYN сканування є його швидкість, оскільки з'єднання розривається до встановлення.

3.3.4 TCP-сканування з порушенням специфікації протоколу

Сканування TCP-FIN, Xmas Tree, TCP-Null:

- Null-сканування – не встановлюються жодні прапори у заголовку TCP;
- FIN-сканування – встановлюється лише прапор FIN;
- XmasTree-сканування – встановлюються прапори FIN, PSH та URG; незвичайна назва цього сканування пов'язана з тим, що встановлені біти в полі прапорів TCP-заголовка комусь видалися схожими на вогники новорічної ялинки (Christmas Tree).

Ці три типи сканування (рис. 3.11) використовують непомітну лазівку TCP RFC, щоб розділяти порти на відкриті і закриті. Коли сканується система, що відповідає вимогам RFC, будь-який пакет, що не містить встановленого біта SYN, RST або ACK, спричинить надсилання TCP-RST у відповідь у випадку,

якщо порт закритий, або не спричинить жодної відповіді, якщо порт відкритий.

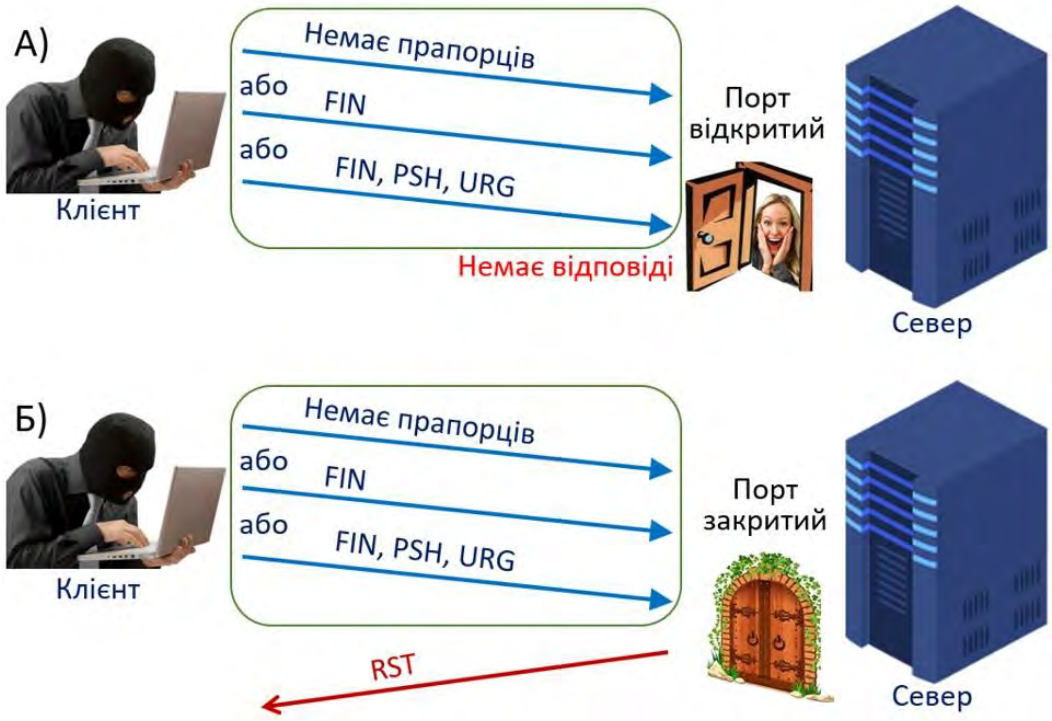


Рисунок 3.11. Сканування TCP-FIN, Xmas Tree, TCP-Null
А) випадок відкритого порту; Б) випадок закритого порту

Ключовою особливістю цих типів сканування є їхня здатність непомітно обійти деякі пакетні фільтри.

Цей метод не працює для систем Windows, які не дотримуються специфікацій RFC.

Сканування TCP-ACK. Цей тип сканування відрізняється від інших тим, що він не може визначити відкритий порт. Він використовуються для виявлення правил пакетних фільтрів, а також для визначення портів, що фільтруються ними.

Часто пакетні фільтри налаштовуються таким чином, щоб забороняти запити від зовнішніх джерел (такі запити починаються з TCP-SYN) до своїх мережних додатків. Тобто TCP-сегменти із прапором SYN пакетний фільтр заблокує. Але доступ внутрішнім клієнтам до зовнішніх серверів зазвичай дозволено. Для доступу до зовнішніх серверів внутрішній клієнт реалізує протокол триетапного квитування, посилаючи TCP-SYN і очікуючи у відповідь TCP-SYN-ACK. Оскільки таке підключення правилом фільтрації зазвичай дозволене, то така відповідь (з прапором ACK) мережевий фільтр пропускає (рис. 3.12).

Пакет запиту при TCP-ACK сканування містить лише ACK прапор. При скануванні нефільтрованих систем відкриті та закриті порти обидва повертатимуть у відповідь RST-пакет.

Порти, які не відповідають або надсилають у відповідь ICMP повідомлення про помилку, позначаються як фільтровані (рис. 3.13).

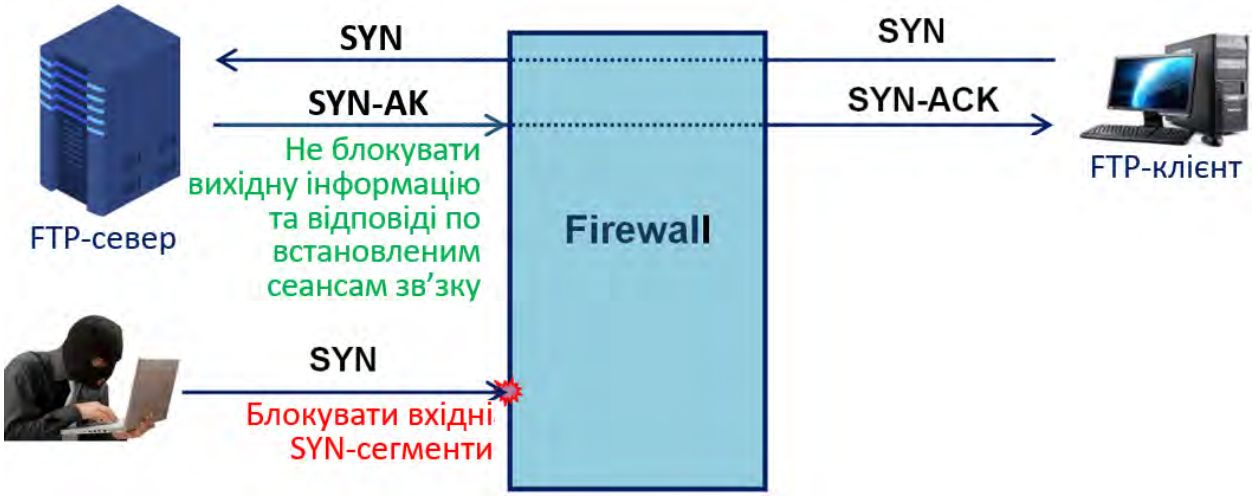


Рисунок 3.12. Вихідні сеанси зв'язку (і відповіді) дозволені, а запити на відкриття з'єднань блокуються.

Рисунок 3.13. ACK-сканування

3.3.5 Додаткові механізми сканування

Вказівка портів джерела для успішного сканування. При скануванні портів порт джерела також включається в заголовок пакета і номер цього порту може спричинити блокування повідомлення. Щоб збільшити ймовірність проходу пакета через маршрутизатори і брандмауери, що захищають мережу, атакуючий вибирає спеціальні номери TCP- або UDP-портів джерела для пакетів, що пересилаються. Найпопулярніший варіант – TCP-порт 80, щоб створювалося враження, що інформація походить від Web-сервера. Також часто застосовується TCP-порт 25 – це виглядає так, ніби дані відправлені з поштового сервера Internet, що працює за протоколом SMTP.

Атакуючий може використовувати TCP-порт джерела 20, зображуючи, що дані йдуть за встановленим FTP з'єднання. Як показано на рис. 3.14, при роботі FTP-протоколу є два з'єднання: контрольне та з'єднання для безпосередньої передачі файлів. Контрольне з'єднання FTP відкривається клієнтом, який передає серверові команди, такі як вхід до системи, запит списку файлів тощо. Після отримання запиту на завантаження файлу FTP

сервер встановлює зворотне з'єднання з клієнтом, використовуючи стандартний номер порту-відправника 20 (активний режим роботи FTP-протоколу).

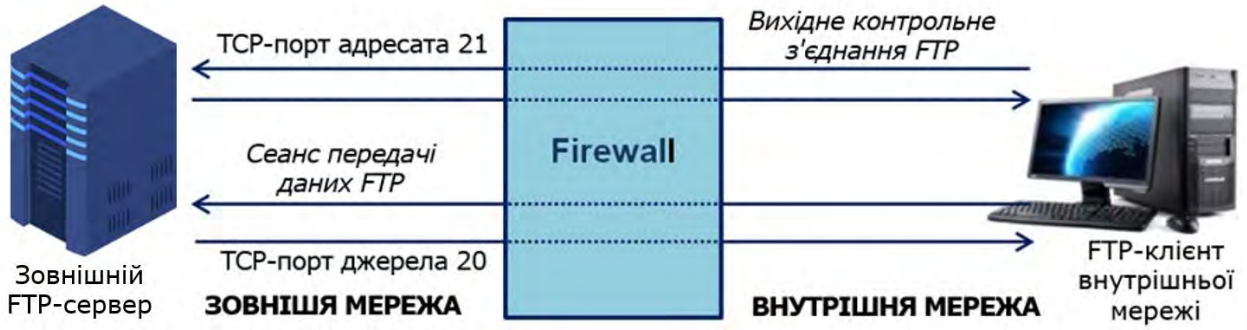


Рисунок 3.14. Стандартні з'єднання FTP
(контрольна та безпосередня передача даних)

Налаштування багатьох мережевих екранів дозволяють вхідні FTP-з'єднання, щоб клієнти внутрішньої мережі могли завантажувати файли з сервера FTP, розташованого в зовнішній мережі. Атакуючий скористається перевагою, що надається такими мережами, відправивши в процесі сканування пакет з TCP-портом джерела 20 (рис. 3.15).

Аналогічно при скануванні UDP-сервісів, якщо номер порту джерела дорівнює 53, здаватиметься, що це DNS-відповіді, і, швидше за все, такі пакети пройдуть на мережу на відміну від пакетів, у яких номер порту джерела буде іншим.

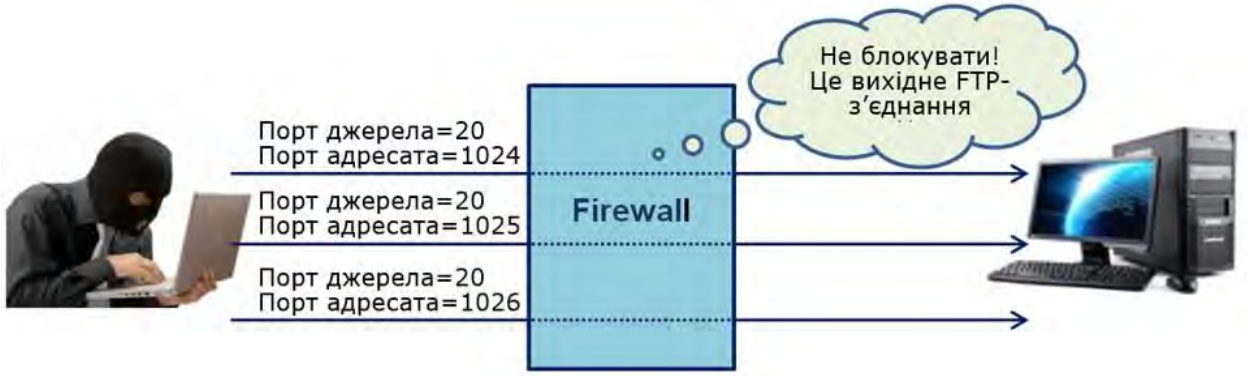


Рисунок 3.15. Сканування з використанням TCP-порту джерела 20
(імітація FTP-з'єднань)

Хибні адреси. Для атакуючого важливо приховати справжнє джерело сканування. Можна, звичайно, в заголовку IP-пакета вказати неправдиву адресу відправника, але при цьому атакуючий не отримає відповіді. Один із пакетів SYN-ACK, RESET або ICMP-повідомлення типу «порт недоступний» повинні повернутися до атакуючого, інакше неможливо буде правильно обробити результати. Єдиний спосіб отримання результатів – включення справжньої адреси джерела в один з пакетів, у той час як безліч інших аналогічних надіслати з помилковими адресами. Мережа, що сканується, не буде точно знати, звідки здійснюється сканування. Якщо атакуючий

використовує, наприклад, 30 помилкових адрес, «жертві» доведеться перевірити безліч різних джерел атаки.

Отже, хибні адреси заважатимуть дослідженню, що надасть атакуючому більше часу, перш ніж його вирахують.

3.3.6 Захист від сканування портів

- Кращий спосіб перешкодити атакуючому виявити відкриті порти на вашому комп'ютері закрити всі непотрібні порти.
- Правильне налаштування брандмауера (доцільно, наприклад, запам'ятовувати історію протокольних обмінів).
- Використання систем виявлення вторгнень (Intrusion Detection System, IDS).

3.4 Запитання до розділу

1) Яким чином можна визначити активні хости в IP-мережі?
2) Як працює програма trceroute при визначенні маршруту до заданого хосту?
3) Яким чином можливо захистися від атаки «Хибний ARP-сервер»?
4) При посилці UDP-дейтаграми на заданий порт (нефільтрований) у відповідь не надійшло жодне повідомлення. Що це означає?
5) При посиланні TCP SYN-запиту на заданий порт (нефільтрований) надійшла RST-відповідь. Що це означає?
6) Які Ви знаєте методи захисту від сканування портів.

РОЗДІЛ 4 МІЖМЕРЕЖЕВІ ЕКРАНИ

4.1 Загальні положення

Якщо в якості зовнішньої мережі використовується відкрита або будь-яка інша потенційно ворожа мережа, то виникають загрози порушення встановлених правил міжмережевої взаємодії, а саме:

- загрози неправомірного вторгнення у внутрішню мережу із зовнішньої;
- загрози несанкціонованого доступу до зовнішньої мережі з внутрішньої.

Неправомірне вторгнення у внутрішню мережу із зовнішньої може виконуватися як із метою несанкціонованого використання ресурсів внутрішньої мережі, наприклад розкрадання інформації, так і з порушення її працездатності.

Загрози несанкціонованого доступу до зовнішньої мережі з внутрішньої мережі актуальні у разі обмеження дозволеного доступу до зовнішньої мережі правилами, встановленими в організації. Таке обмеження може знадобитися, наприклад, у таких випадках:

- для запобігання витоку конфіденційних даних;
- при забороні доступу, наприклад, у навчальних закладах, до інформації нецензурної та небажаної спрямованості;
- у разі заборони службового доступу до розважальних комп'ютерних ресурсів у робочий час.

Проблема захисту від несанкціонованих дій при взаємодії із зовнішніми мережами успішно може бути вирішена тільки за допомогою спеціалізованих програмно-апаратних комплексів, що забезпечують цілісний захист комп'ютерної мережі від ворожого зовнішнього середовища. Такі комплекси називають міжмережевими екранами (міжмережевий екран), брандмауерами чи системами FireWall.

Міжмережевий екран повинен розташовуватися між мережею організації, що захищається (внутрішньою мережою), і потенційно ворожою зовнішньою мережею (рис. 4.1). При цьому всі взаємодії між цими мережами повинні здійснюватися тільки через міжмережевий екран. Організаційно екран входить до складу мережі, що захищається.

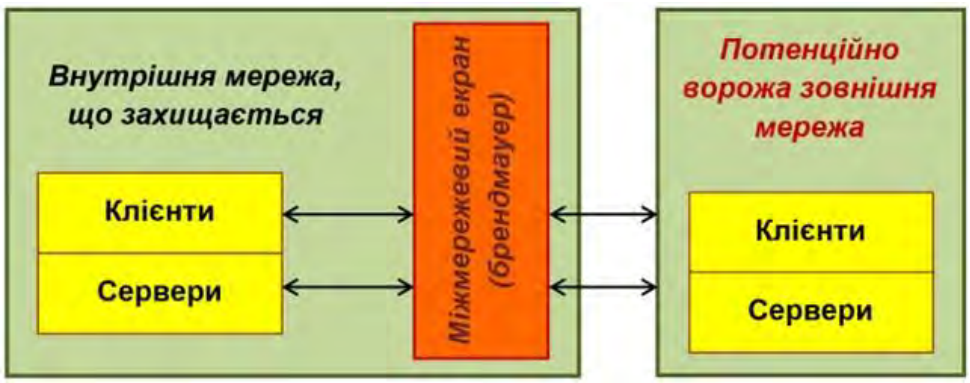


Рисунок 4.1. Схема підключення міжмережевого екрану

Міжмережевий екран не є симетричним. Для нього окремо задаються правила, що обмежують доступ із внутрішньої мережі до зовнішньої мережі та навпаки.

У загальному випадку робота міжмережевого екрану ґрунтується на динамічному виконанні двох груп функцій:

- фільтрації інформаційних потоків, що проходять через нього;
- посередництва при реалізації міжмережевих взаємодій.

Залежно від типу екрана, ці функції можуть виконуватися з різною повнотою. Прості міжмережеві екрани орієнтовані на виконання лише однієї з даних функцій. Комплексні екрани забезпечують спільне виконання цих функцій захисту.

Повнота і правильність управління вимагають, щоб комплексний брандмауер мав можливість аналізу та використання наступних елементів:

• Інформації про з'єднання – інформації від кількох рівнів OSI у пакеті (кадрі), який аналізує брандмауер. Тобто службової інформації у заголовках протокольних блоків даних і, власне, вмісту цих блоків.

• Історії з'єднань – інформації, отриманої від попередніх з'єднань. Наприклад, вихідна команда PORT сесії FTP має бути збережена для того, щоб надалі можна було перевірити вхідне з'єднання FTP-data. TCP-сегмент з прапором SYN необхідно запам'ятати, щоб правильно інтерпретувати сегмент у відповідь з прапором ACK при створенні TCP-з'єднання.

• Стану прикладного рівня – інформації про стан, отриманої з інших програм. Наприклад, автентифікованому раніше користувачу можна надати доступ через брандмауер тільки для авторизованих видів сервісу.

• Агрегуючих елементів – обчислень різноманітних виразів, заснованих на всіх перерахованих вище факторах.

Пристрій, подібний до міжмережевого екрану, може використовуватися і для захисту окремого комп'ютера. У цьому випадку екран, що вже не є міжмережевим, встановлюється на комп'ютер, що захищається. Такий екран називається брандмауером комп'ютера або брандмауером операційної системи.

4.2 Функції фільтрації

Фільтрація інформаційних потоків полягає у їх вибірковому пропусканні через екран, можливо, з виконанням деяких перетворень та повідомленням відправника про те, що його даним у пропуску відмовлено. Фільтрування здійснюється на основі набору правил, попередньо завантажених в екран і є виразом мережевих аспектів прийнятої безпекової політики. Тому міжмережевий екран зручно представляти як послідовність фільтрів (рис. 4.2), що обробляють інформаційний потік.

Кожен із фільтрів призначений для інтерпретації окремих правил фільтрації шляхом виконання наступних операцій:

1 Аналіз інформації за заданими в інтерпретованих правилах критеріями (умовами), наприклад адреса відправника або одержувача відповідає

небажаній адресі (або відповідає лише заданій адресі), або тип протоколу, для якого ця інформація призначена, не повинен оброблятися.

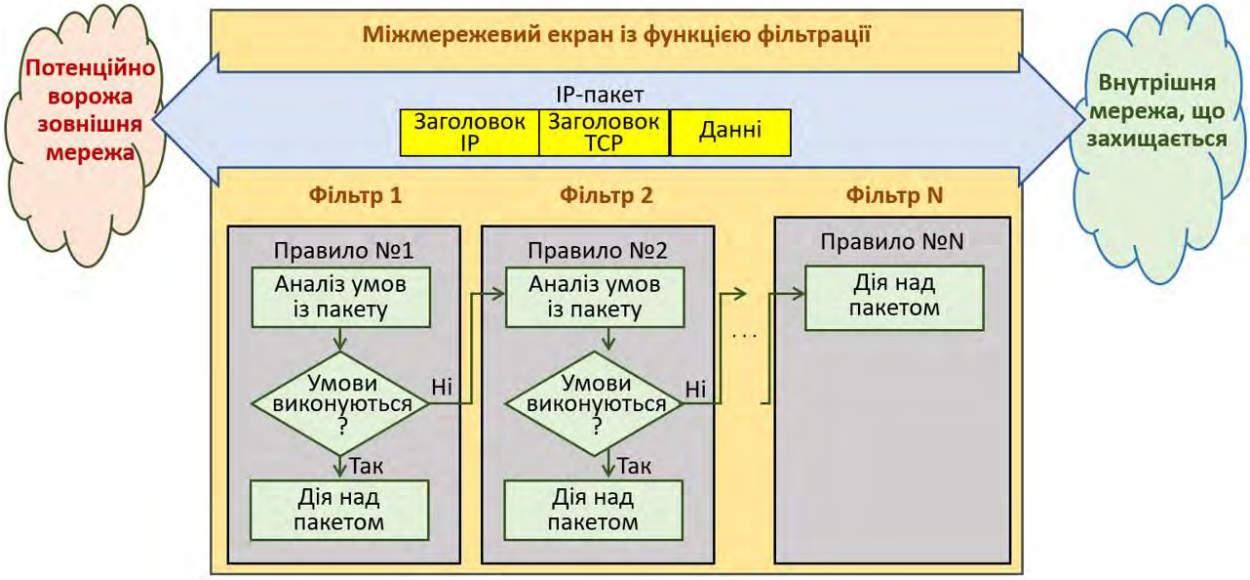


Рисунок 4.2. Структура міжмережевого екрану з функцією фільтрації

2 Прийняття з урахуванням інтерпретованих правил одного з наступних рішень:

• передати дані на наступний фільтр для продовження аналізу (якщо умови, задані у правилі, не виконуються); при цьому останній фільтр вказує на ту дію, яку потрібно виконати з пакетом, якщо не виконано жодної умови, зазначеної у попередніх фільтрах (правилах);

• не пропустити (знищити) дані;

• не пропустити (знищити) дані, та від імені одержувача повернути результат (зазвичай фіктивний) відправнику; наприклад, не пропустити дані на якийсь відкритий порт і повернути відправнику повідомлення, що порт закритий;

• пропустити дані у внутрішню мережу, ігноруючи наступні фільтри.

Правила фільтрації можуть задавати і додаткові дії, наприклад реєстрація подій тощо. Відповідно правила фільтрації визначають перелік умов, за якими з використанням зазначених критеріїв аналізу здійснюється:

• дозвіл або заборона подальшої передачі даних;

• виконання додаткових захисних функцій.

Як критерії аналізу інформаційного потоку можуть використовуватися такі параметри:

• службові поля пакетів повідомлень, що містять мережеві адреси, ідентифікатори, адреси інтерфейсів, номери портів та інші значущі дані;

• безпосередній вміст пакетів повідомлень, що перевіряється, наприклад, на наявність комп'ютерних вірусів;

• зовнішні характеристики потоку інформації, наприклад, часові, частотні характеристики, обсяг даних тощо.

Критерії аналізу, що використовуються, залежать від рівнів моделі OSI, на яких здійснюється фільтрація. У загальному випадку, чим вищий рівень моделі OSI, на якому міжмережевий екран фільтрує пакети, тим вищий рівень захисту, який він забезпечує.

4.3 Функції посередництва

Функції посередництва міжмережевий екран виконує за допомогою спеціальних програм, які називаються агентами, що екранують, або програмами-посередниками. Дані програми є резидентними і забороняють безпосередню передачу пакетів повідомлень між клієнтськими та серверними частинами розподілених мережних додатків, причому передбачається, що клієнти належать до внутрішньої мережі, а сервери – до зовнішньої (потенційно небезпечної) мережі (рис. 4.3).

Рисунок 4.3. Структура міжмережевого екрану з функцією посередництва

За потреби доступу з внутрішньої мережі у зовнішню мережу чи навпаки спочатку має бути встановлене логічне з'єднання з програмою-посередником, що функціонує на міжмережевому екрані. Програма-посередник перевіряє допустимість запитаної міжмережевої взаємодії і за його вирішенні сама встановлює окреме з'єднання з необхідним комп'ютером. Далі обмін інформацією між комп'ютерами внутрішньої та зовнішньої мережі здійснюється через програмного посередника, який виконує захисні функції.

Загалом екрануючі агенти, блокуючи прозору передачу потоку повідомлень, можуть виконувати такі функції:

- ***Ідентифікація та аутентифікація користувачів.*** Більшість програм-посередників розробляються таким чином, щоб користувач автентифікувався тільки на початку сеансу роботи з міжмережевим екраном. Після цього від нього не потрібна додаткова автентифікація протягом часу, який визначається адміністратором.
- ***Перевірка автентичності даних, що передаються.*** Перевірка автентичності повідомлень та програм полягає у контролі їх цифрових підписів.

• ***Розмежування доступу до ресурсів внутрішньої мережі.*** Способи розмежування до ресурсів внутрішньої мережі нічим не відрізняються від способів розмежування, що підтримуються на рівні операційної системи.

• ***Розмежування доступу до ресурсів зовнішньої мережі.*** При цьому найчастіше використовується один із наступних підходів:

- дозвіл доступу лише за заданими адресами у зовнішній мережі;
- фільтрація запитів на основі оновлюваних списків неприпустимих адрес та блокування пошуку інформаційних ресурсів за небажаними ключовими словами;
- накопичення та оновлення адміністратором санкціонованих інформаційних ресурсів зовнішньої мережі в дисковій пам'яті брандмауера та повна заборона доступу до зовнішньої мережі.

• ***Фільтрування та перетворення (наприклад, прозоре шифрування) потоку повідомлень.*** Тут слід розрізняти два види програм посередників:

- екрануючі агенти, орієнтовані на аналіз потоку повідомлень для певних видів сервісу, наприклад FTP, HTTP, Telnet;
- універсальні екрануючі агенти, що обробляють весь потік повідомлень, наприклад, агенти, орієнтовані на пошук та знешкодження комп'ютерних вірусів або прозоре шифрування даних.

• ***Трансляція внутрішніх мережних адрес для вихідних пакетів повідомлень.*** Ця функція реалізується по відношенню до всіх пакетів, що слідують із внутрішньої мережі до зовнішньої. Для цих пакетів посередник виконує автоматичне перетворення IP-адрес комп'ютерів-відправників в одну IP-адресу, що асоціюється з брандмауером, з якого передаються всі вихідні пакети. IP-адреса брандмауера стає єдиною активною IP-адресою, яка потрапляє у зовнішню мережу. При такому підході топологія внутрішньої мережі прихована для зовнішніх користувачів, що ускладнює завдання несанкціонованого доступу. Крім підвищення безпеки трансляція адрес дозволяє мати всередині мережі власну систему адресації, що ефективно вирішує проблему розширення адресного простору внутрішньої мережі та дефіциту адрес Інтернет.

• ***Реєстрація подій, реагування на події, що задаються, а також аналіз зареєстрованої інформації та генерація звітів.*** Як обов'язкову реакцію виявлення спроб виконання несанкціонованих дій має бути визначено повідомлення адміністратора, тобто видача попереджувальних сигналів. За рахунок використання спеціальних протоколів посередники можуть виконати віддалену сповіщення про певні події в режимі реального часу.

• ***Кешування даних, що запитуються із зовнішньої мережі.*** При доступі користувачів внутрішньої мережі до інформаційних ресурсів зовнішньої мережі, вся інформація накопичується на просторі жорсткого диска брандмауера, званого в цьому випадку rгоху-сервером. Тому, якщо при черговому запиті потрібна інформація опиниться на proxy-сервері, то посередник надає її без звернення до зовнішньої мережі, що суттєво

прискорює доступ. Функція кешування може успішно використовуватися для обмеження доступу до інформаційних ресурсів зовнішньої мережі. У цьому випадку на proxy-сервері накопичуються і оновлюються адміністратором лише санкціоновані інформаційні ресурси зовнішньої мережі. Користувачам внутрішньої мережі дозволяється доступ лише до інформаційних ресурсів proxy-сервера, а безпосередній доступ до ресурсів зовнішньої мережі забороняється.

Брандмауери з посередниками дозволяють також організовувати захищені віртуальні мережі (Virtual Private Network – VPN),

Екрануючі агенти набагато надійніші за звичайні фільтри і забезпечують більший ступінь захисту. Однак вони знижують продуктивність обміну даними між внутрішньою і зовнішньою мережами і не мають того ступеня прозорості для додатків і кінцевих користувачів, яка характерна для простих фільтрів.

4.4 Особливості міжмережевих екранів на різних рівнях OSI

4.4.1 Загальна схема відповідності мережевих екранів рівням OSI

Брандмауери підтримують безпеку міжмережевої взаємодії на різних рівнях моделі OSI. При цьому функції захисту, які виконуються на різних рівнях еталонної моделі, істотно відрізняються друг від друга. Тому комплексний міжмережевий екран зручно у вигляді сукупності неподільних екранів, кожен із яких орієнтований на окремий рівень моделі OSI. Найчастіше комплексний екран функціонує на мережевому, сеансовому та прикладному рівнях еталонної моделі. Відповідно розрізняють такі неподільні брандмауери (рис. 4.4), як містковий екран (екрануючий комутатор), екрануючий маршрутизатор, екрануючий транспорт (шлюз сеансового рівня), а також екрануючий шлюз (шлюз прикладного рівня).

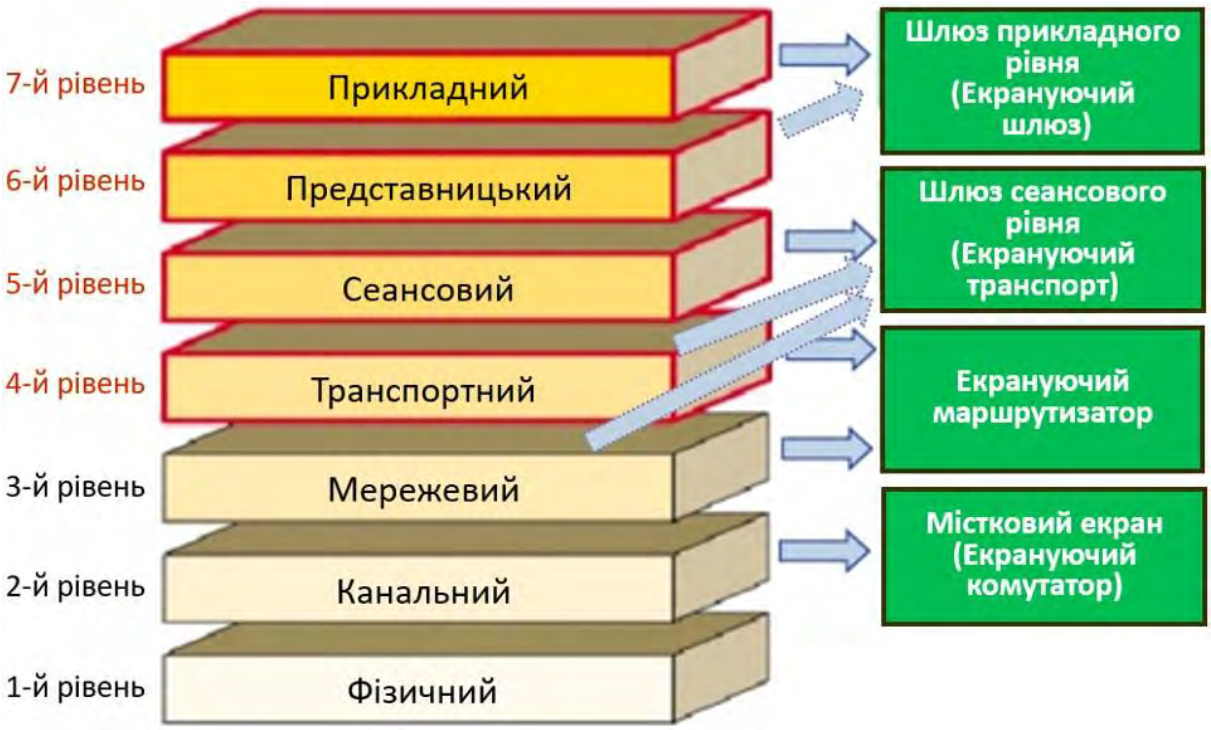


Рисунок 4.4. Типи міжмережевих екранів, що функціонують на окремих рівнях OSI

Враховуючи, що протоколи (TCP/IP, SPX/IPX), що використовуються в мережах, неоднозначно відповідають моделі OSI, то екрани перерахованих типів при виконанні своїх функцій можуть охоплювати і сусідні рівні еталонної моделі. Наприклад, прикладний екран може здійснювати автоматичне зашифровування повідомлень при їх передачі в зовнішню мережу, а також автоматичне розшифровування криптографічно закритих даних, що приймаються. У цьому випадку такий екран функціонує не тільки на прикладному рівні моделі OSI, але і на рівні представлення. Шлюз сеансового рівня при своєму функціонуванні охоплює сеансовий, транспортний та мережевий рівні моделі OSI. Екрануючий маршрутизатор під час аналізу пакетів повідомлень перевіряє їх заголовки як мережевого, так й транспортного рівня.

Міжмережеві екрани кожного з типів мають свої переваги та недоліки. Багато з використовуваних брандмауерів є або прикладними шлюзами, або екрануючими маршрутизаторами, не підтримуючи повну безпеку міжмережевої взаємодії. Надійний захист забезпечують тільки комплексні міжмережеві екрани, кожен з яких поєднує екрануючий маршрутизатор, шлюз сеансового рівня, а також прикладний шлюз.

4.4.2 Містковий екран

Містковий екран (екрануючий міст, керований комутатор), що функціонує на 2-му рівні моделі OSI, відомий також як прозорий (stealth), прихований, тіньовий міжмережевий екран. Фільтрація трафіку містковими екранами відповідає канальному рівні, тобто вони працюють із кадрами (frame, кадр).

До переваг подібних міжмережевий екран можна віднести:

• простота – не потрібно змінювати налаштування корпоративної мережі, не потрібно додаткового конфігурування мережних інтерфейсів міжмережевий екран;

• висока продуктивність – оскільки це прості пристрої, вони не вимагають великих витрат ресурсів;

• прозорість – оскільки цей екран функціонує на 2 рівні моделі OSI, то його мережевий інтерфейс не має IP-адреси і є невидимим для навколишнього світу, атакуючі навіть не знатимуть, що існує міжмережевий екран, який перевіряє кожен їхній кадр.

Недоліки місткових екранів.

• фільтрація кадрів за MAC-адресами не є ефективною, оскільки апаратно встановлена в мережній карті MAC-адреса легко змінюється програмним шляхом, причому значення, вказане через драйвер, має більш високий пріоритет, ніж зашите в плату;

• неможливість аналізу протоколів вищих рівнів.

4.4.3 Екрануючий маршрутизатор

Екрануючий маршрутизатор, званий ще пакетним фільтром, призначений для фільтрації IP-пакетів та забезпечує прозору взаємодію між внутрішньою та зовнішньою мережами. Він функціонує на мережному рівні моделі OSI, але при виконання своїх окремих функцій може охоплювати і транспортний рівень еталонної моделі. Рішення про те, пропустити або відбракувати дані, приймається для кожного пакета незалежно на основі вказаних правил фільтрації. Обробка цих правил відбувається так само, як і у будь-якого міжмережевого екрану, що виконує функції фільтрації (див. 4.2). Але для прийняття рішень тут аналізуються лише заголовки мережного та транспортного рівнів (рис. 4.5).

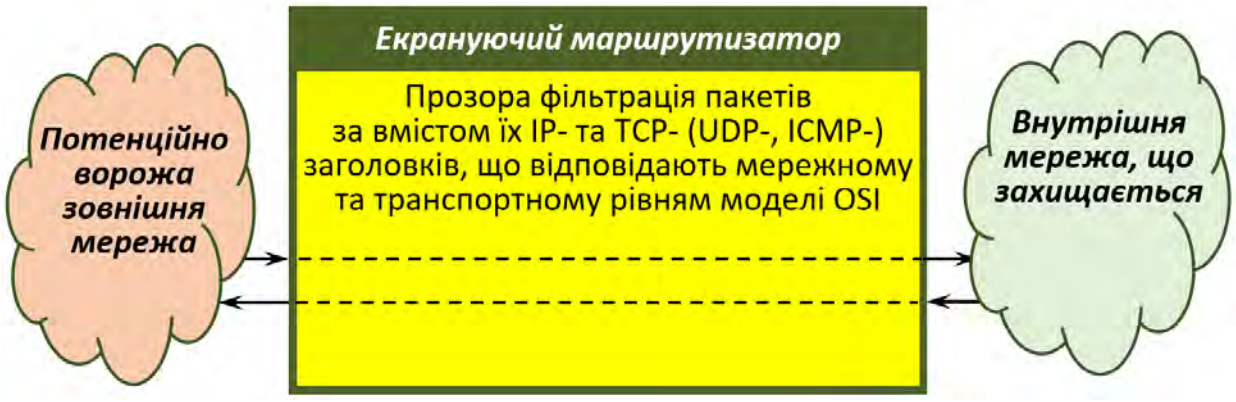


Рисунок 4.5. Схема функціонування пакетного фільтра

Зазвичай, як аналізовані поля IP- і TCP- (UDP)-заголовків кожного пакета виступають:

- адреса відправника;
- адреса одержувача;
- тип пакета;

• прапор фрагментації пакета
• номер порту джерела;
• номер порту одержувача.

Як пакетний фільтр може використовуватися як звичайний маршрутизатор, так і працююча на сервері програма, сконфігуровані таким чином, щоб фільтрувати вхідні та вихідні пакети. Сучасні маршрутизатори, наприклад маршрутизатори компаній Bay Networks і Cisco, дозволяють зв'язувати з кожним портом кілька десятків правил і фільтрувати пакети як на вході, так і на виході.

До переваг екрануючих маршрутизаторів відносяться:

• простота самого екрану, а також процедур його конфігурування та встановлення;

• прозорість для програмних програм та мінімальний вплив на продуктивність мережі;

• низька вартість, зумовлена тим, що будь-який маршрутизатор тією чи іншою мірою надає можливість фільтрації пакетів.

Недоліки екрануючих маршрутизаторів:

• не підтримують багато необхідних функцій захисту, наприклад, автентифікацію кінцевих вузлів, криптографічне закриття пакетів повідомлень, а також перевірку їхньої цілісності та автентичності;

• вразливі для таких поширених мережевих атак, як підробка вихідних адрес та несанкціонована зміна вмісту пакетів повідомлень.

4.4.4 Шлюз сеансового рівня

Шлюз сеансового рівня, званий ще екрануючим транспортом, призначений для контроля віртуальних з'єднань та трансляції IP-адрес при взаємодії із зовнішньою мережею. Він функціонує на сеансовому рівні моделі OSI, охоплюючи в процесі роботи також транспортний і мережевий рівні еталонної моделі. Захисні функції екрануючого транспорту належать до функцій посередництва.

Коли робоча станція (клієнт) запитує зв'язок із зовнішньою мережею, шлюз приймає цей запит, перевіряючи, чи він задовольняє встановленим критеріям. Потім, діючи від імені клієнта, шлюз встановлює з'єднання з комп'ютером зовнішньої мережі та стежить за виконанням процедури квитування зв'язку за протоколом TCP.

Контроль віртуальних TCP-з'єднань полягає у контролі квитування зв'язку, а також контролі передачі інформації за встановленими віртуальними TCP-каналами. Такий контроль ґрунтується на інформації, що міститься у заголовках пакетів сеансового рівня протоколу TCP. Однак якщо пакетний фільтр при аналізі TCP-заголовків перевіряє тільки номери портів джерела та одержувача, то транспортує аналізує інші поля, що відносяться до процесу квитування зв'язку.

Для контроля TCP-з'єднань у шлюзах сеансового рівня використовують спеціальні програми, які називають канальними посередниками (pipe proxies).

Ці посередники встановлюють між внутрішньою та зовнішньою мережами віртуальні TCP-канали, а потім контролюють передачу цими каналами пакетів, що генеруються додатками TCP/IP (рис. 4.6). Запитаний сеанс вважається допустимим тільки в тому випадку, якщо при виконанні процедури квитування прапори SYN і ACK, а також числа, що містяться в заголовках TCP-пакетів, логічно пов'язані між собою.

Рисунок 4.6. Схема функціонування шлюзу сеансового рівня

У процесі контролю передачі інформації з віртуальних каналів фільтрація даних TCP-сегментів екрануючим транспортом не здійснюється.

Насправді більшість шлюзів сеансового рівня не є самостійними продуктами, а поставляються у комплекті зі шлюзами прикладного рівня.

Шлюз сеансового рівня забезпечує також трансляцію внутрішніх адрес мережевого рівня (IP-адрес) при взаємодії із зовнішньою мережею.

Трансляція адрес викликана необхідністю посилення захисту шляхом приховування від зовнішніх користувачів структури внутрішньої мережі, що захищається. Канальний посередник приймає запит від робочої станції внутрішньої мережі та потім ініціює новий запит (з єдиною вихідною IP-адресою шлюза сеансового рівня) до комп'ютера зовнішньої мережі. Тому комп'ютер зовнішньої мережі сприймає запит як вихідний від посередника, а не від клієнта.

З погляду реалізації шлюз сеансового рівня є досить проста, отже, надійна програма. Він доповнює екрануючий маршрутизатор функціями контролю віртуальних TCP-з'єднань та трансляції внутрішніх IP-адрес.

Недоліки у шлюза сеансового рівня ті ж самі, що й у екрануючого маршрутизатора – не забезпечується контроль та захист вмісту пакетів повідомлень, не підтримуються автентифікація користувачів та кінцевих вузлів, а також інші функції захисту локальної мережі. Тому шлюз сеансового рівня застосовують як додаток до прикладного шлюзу.

4.4.5 Шлюз прикладного рівня

Прикладний шлюз, званий також екрануючим шлюзом, функціонує на прикладному рівні моделі OSI, охоплюючи також рівень представлення, і забезпечує найбільш надійний захист міжмережевих взаємодій. Захисні функції прикладного шлюзу, як і екрануючого транспорту, належать до функцій посередництва. Однак прикладний шлюз, на відміну від шлюзу

сеансового рівня, може виконувати значно більшу кількість функцій захисту, до яких належать практично всі функції посередництва, розглянуті в п. 4.3.

Прикладний шлюз є комп'ютером, на якому функціонують програмні посередники (екрануючі агенти) – по одному для кожного прикладного протоколу, що обслуговується (HTTP, FTP, SMTP, NNTP та ін.).

Посередник кожної служби TCP/IP орієнтований на обробку повідомлень та виконання функцій захисту, які стосуються саме цієї служби. Так само, як і шлюз сеансового рівня, прикладний шлюз функціонує як сервер-посередник і утворює два з'єднання: від робочої станції до брандмауера і від брандмауера до місця призначення, виключаючи прямі з'єднання між внутрішньою та зовнішньою мережею. Однак посередники, які використовуються прикладним шлюзом, мають важливі відмінності від канальних посередників рівня сеансового шлюза. По-перше, посередники прикладного шлюзу пов'язані з конкретними програмами (програмними серверами), а по-друге, вони можуть фільтрувати потік повідомлень на прикладному рівні моделі OSI (рис. 4.7).

На відміну від канальних посередників, посередники прикладного шлюзу пропускають лише пакети, згенеровані додатками, які їм доручено обслуговувати. Наприклад, програма-посередник служби HTTP може обробляти лише трафік, що генерується цією службою.

Якщо у мережі працює прикладний шлюз, то вхідні та вихідні пакети можуть передаватися лише для тих служб, котрим є відповідні посередники. Так, якщо прикладний шлюз використовує лише програми-посередники HTTP, FTP і Telnet, він буде обробляти лише пакети, які стосуються цих служб, блокуючи пакети всіх інших служб.

Фільтрування потоків повідомлень реалізується прикладними шлюзами на прикладному рівні моделі OSI. Відповідно посередники прикладного шлюзу, на відміну від канальних посередників, забезпечують перевірку вмісту оброблюваних пакетів. Наприклад, для служби FTP можливе динамічне знешкодження комп'ютерних вірусів у файлах, що копіюються із зовнішньої мережі.

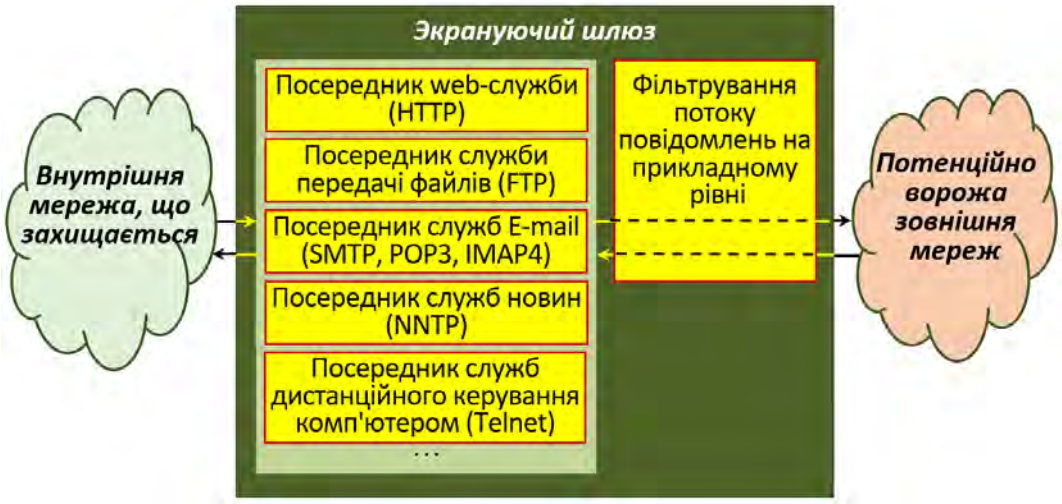


Рисунок 4.7. Схема функціонування прикладного шлюзу

Шлюз прикладного рівня має такі важливі переваги:

• за рахунок можливості виконання переважної більшості функцій посередництва забезпечує найвищий рівень захисту локальної мережі;

• захист на рівні додатків дозволяє здійснювати велику кількість додаткових перевірок, зменшуючи ймовірність проведення успішних атак, заснованих на недоліках програмного забезпечення;

• при порушенні працездатності прикладного шлюзу блокується наскрізне проходження пакетів між мережами, що розділяються, що не знижує безпеку мережі, що захищається, у разі виникнення відмов.

До недоліків прикладного шлюзу відносяться:

• досить велика складність самого брандмауера, а також процедур його встановлення та конфігурування;

• високі вимоги до продуктивності та ресурсомісткості комп'ютерної платформи;

• відсутність «прозорості» для користувачів та зниження пропускної спроможності при реалізації міжмережевих взаємодій;

• висока вартість.

4.5 Визначення та принципи розробка політики міжмережевої взаємодії

Політика міжмережевої взаємодії є частиною політики безпеки в організації, яка визначає вимоги до безпеки інформаційного обміну із зовнішнім світом. Дані вимоги обов'язково повинні відображати два аспекти:

- політику доступу до мережевих сервісів;
- політику роботи міжмережевого крану.

Політика доступу до мережевих сервісів визначає правила надання, а також використання всіх можливих сервісів комп'ютерної мережі, що захищається. Відповідно в рамках цієї політики повинні бути задані всі послуги, що надаються через мережевий екран, та допустимі адреси клієнтів для кожного сервісу. Крім того, повинні бути вказані правила для користувачів, що описують, коли та які користувачі яким сервісом і на якому комп'ютері можуть скористатися. Окремо визначаються правила автентифікації користувачів та комп'ютерів, а також умови роботи користувачів поза локальною мережею організації.

Політика роботи міжмережевого екрану задає базовий принцип керування міжмережевою взаємодією, покладений основою функціонування брандмауера. Може бути обраний один із двох таких принципів:

- заборонено все, що явно не дозволено;
- дозволено все, що явно не заборонено.

Залежно від вибору рішення може бути прийнято як на користь безпеки і на шкоду зручності використання мережевих сервісів, так і навпаки.

У першому випадку міжмережевий екран повинен бути сконфігурований таким чином, щоб блокувати будь-які явно не дозволені міжмережеві взаємодії. Враховуючи, що такий підхід дозволяє адекватно реалізувати принцип мінімізації привілеїв, він, з погляду безпеки, є найкращим.

При виборі принципу «дозволено все, що явно не заборонено», міжмережевий екран налаштовується таким чином, щоб блокувати тільки явно заборонені міжмережеві взаємодії. У цьому випадку підвищується зручність використання мережевих сервісів з боку користувачів, але знижується безпека міжмережевої взаємодії.

4.6 Визначення схеми підключення міжмережевого екрану

4.6.1 Підмережі в захищеній локальній мережі

Захищену локальну мережу розглядатимемо як сукупність закритої та відкритої підмереж. Тут під відкритою підмережою (часто називається демілітаризованою зоною) розуміється підмережа, доступ до якої з боку потенційно ворожої зовнішньої мережі може бути повністю або частково відкритий. У відкриту підмережу можуть, наприклад, входити загальнодоступні WWW-, FTP- та SMTP-сервери.

До закритої підмережі доступ із зовнішньої мережі повинен бути повністю закритий.

Серед безлічі можливих схем підключення брандмауерів типовими є такі:

- схема захисту мережі з використанням екрануючого маршрутизатора;
- схема єдиного захисту локальної мережі;
- схема з закритою і такою, що не захищається відкритою підмережами;
- схема з роздільним захистом закритої та відкритої підмереж.

4.6.2 Схема захисту мережі з використанням екрануючого маршрутизатора

Міжмережевий екран, заснований на фільтрації пакетів, є найпоширенішим і найпростішим у реалізації. Він складається з екрануючого маршрутизатора, розташованого між мережею, що захищається, і потенційно ворожою відкритою зовнішньою мережею (рис. 4.8).

Рисунок 4.8. Схема захисту мережі з використанням екрануючого маршрутизатора

Екрануючий маршрутизатор (пакетний фільтр) налаштований для блокування або фільтрації вхідних та вихідних пакетів на основі аналізу їх

адрес і портів. Комп'ютери, що знаходяться в мережі, що захищається, мають прямий доступ до мережі Інтернет, в той час як більша частина доступу до них з Інтернету блокується.

Недолік цієї схеми захисту обумовлений зазначеними вище (див. п. 4.6) недоліками екрануючих маршрутизаторів, головним з яких є відсутність багатьох функцій захисту, які притаманні мережевим екранам з функцією посередництва.

4.6.3 Схема єдиного захисту локальної мережі

У цій схемі (рис. 4.9) брандмауер повністю екранує локальну мережу від ворожої зовнішньої мережі. Зазвичай маршрутизатор налаштовується таким чином, що брандмауер є єдиною видимою зовні машиною. Відкриті сервери, що входять до локальної мережі, також будуть захищені міжмережевим екраном. Однак об'єднання серверів, доступних із зовнішньої мережі, разом з іншими ресурсами локальної мережі, що захищається, істотно знижує безпеку міжмережевих взаємодій. Тому цю схему підключення брандмауера можна використовувати лише за відсутності у локальній мережі відкритих серверів або коли наявні відкриті сервери робляться доступними із зовнішньої мережі лише обмеженої кількості користувачів, яким можна довіряти.

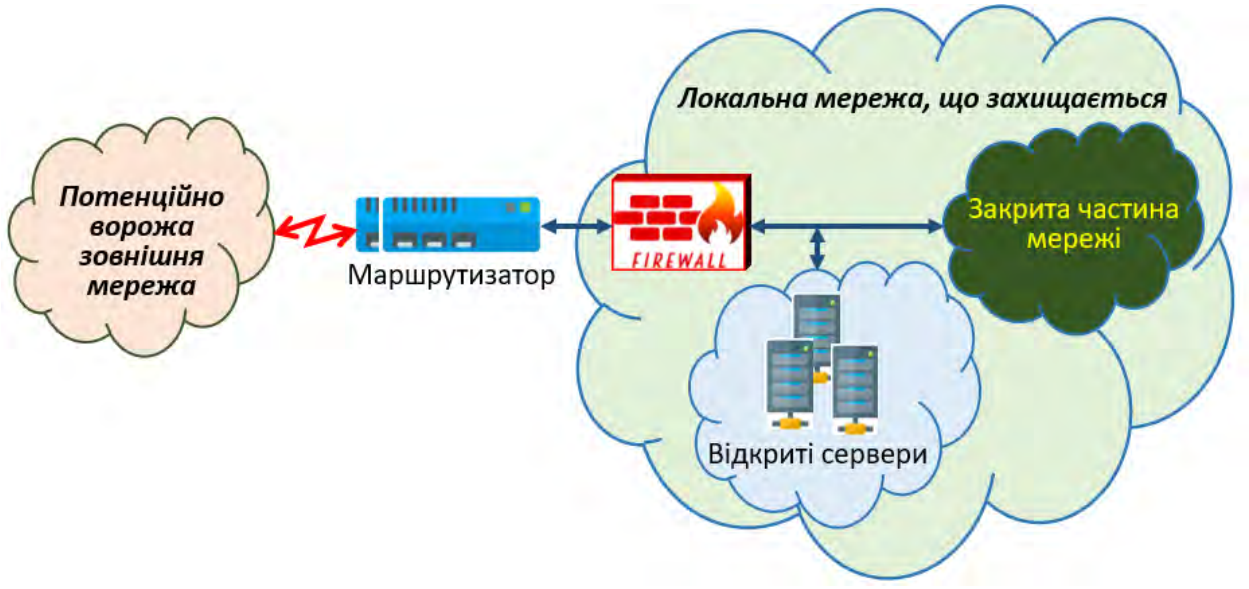


Рисунок 4.9. Схема єдиного захисту локальної мережі

4.6.4 Схема з закритою і відкритою, що не захищається підмережами

За наявності у складі локальної мережі загальнодоступних відкритих серверів їх доцільно винести як відкриту підмережу до міжмережевого екрану (рис. 4.10). Даний спосіб має більш високу захищеність закритої частини локальної мережі, але забезпечує знижену безпеку відкритих серверів, розташованих до міжмережевого екрану. Таку схему підключення

брандмауера з закритою підмережою, що захищається, і відкритою підмережою, що не захищається, доцільно використовувати лише при невисоких вимогах щодо безпеки до відкритої підмережі.

Рисунок 4.10. Схема з закритою і відкритою, що не захищається підмережами

4.6.5 Схема з роздільним захистом закритої та відкритої підмережами

У випадку, коли до безпеки відкритих серверів пред'являються підвищені вимоги, необхідно використовувати схему з роздільним захистом закритої та відкритої підмереж. Така схема може бути побудована на основі одного брандмауера із трьома мережевими інтерфейсами (рис. 4.11) або на основі двох брандмауерів із двома мережевими інтерфейсами (рис. 4.12). В обох випадках доступ до відкритої та закритої підмереж локальної мережі можливий лише через міжмережевий екран. При цьому доступ до відкритої підмережі не дозволяє здійснювати доступ до закритої підмережі.

Рисунок 4.11. Схема з роздільним захистом закритої та відкритої підмереж на основі одного брандмауера з трьома мережевими інтерфейсами

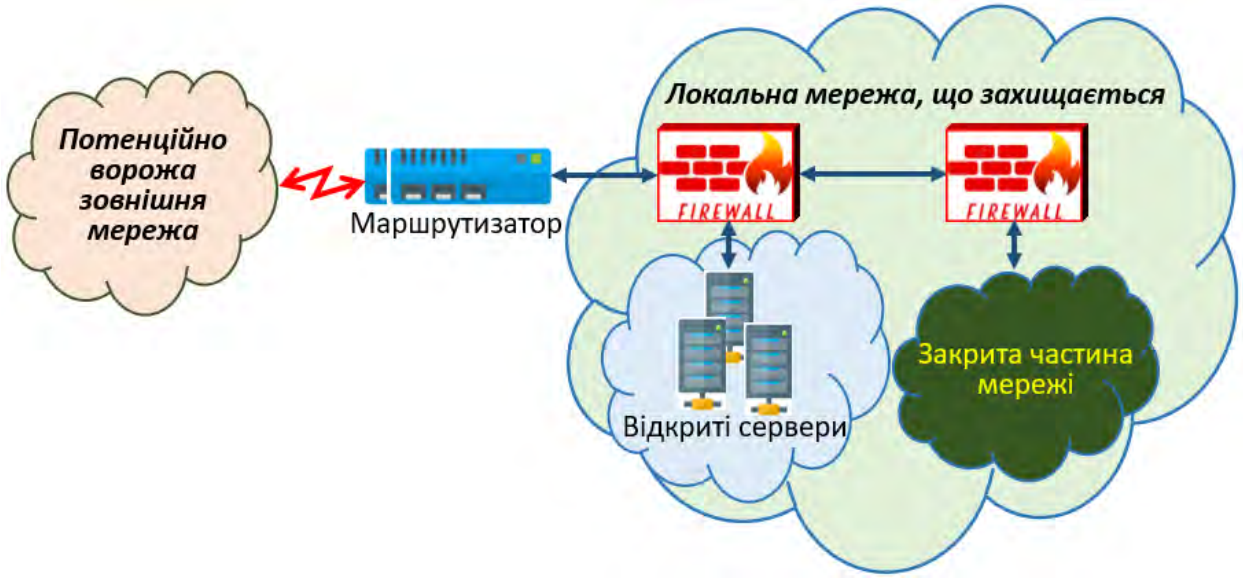


Рисунок 4.12. Схема з роздільним захистом закритої та відкритої підмереж на основі двох брандмауерів

4.7 Налаштування параметрів функціонування брандмауера

Міжмережевий екран є програмно-апаратним комплексом захисту, що складається з комп'ютера, а також функціонуючими на ньому операційної системи (ОС) і спеціального програмного забезпечення. Слід зазначити, що це спеціальне програмне забезпечення також називають брандмауером.

Комп'ютер брандмауера повинен бути досить потужним і фізично захищеним, наприклад, перебувати в спеціально відведеному та охоронюваному приміщенні. Крім того, він повинен мати засоби захисту від завантаження ОС із несанкціонованого носія.

Операційна система брандмауера також повинна задовольняти низку вимог:

- мати засоби розмежування доступу до ресурсів системи;
- блокувати доступ до комп'ютерних ресурсів в обхід програмного інтерфейсу;
- забороняти привілейований доступ до своїх ресурсів із локальної мережі;
- містити засоби моніторингу/аудиту будь-яких адміністративних дій.

Після встановлення на комп'ютер брандмауера вибраної операційної системи, її конфігурування, а також інсталяції спеціального програмного забезпечення можна розпочати налаштування параметрів функціонування всього міжмережевого екрану. Цей процес включає такі етапи:

- вироблення правил роботи міжмережевого екрану відповідно до розробленої політики міжмережевої взаємодії та опис правил в інтерфейсі брандмауера;
- перевірку заданих правил на несуперечність;
- перевірку відповідності параметрів налаштування брандмауера розробленої політики міжмережевої взаємодії.

База правил роботи міжмережевого екрану є формалізоване відображення розробленої політикою міжмережевої взаємодії. Компонентами правил є:

- об'єкти, що захищаються;
- користувачі;
- сервіси.

Об'єкти, що захищаються, можуть об'єднуватися в групи. Кожен об'єкт має набір атрибутів, таких як мережна адреса, маска підмережі тощо.

При описі правил роботи міжмережевого екрану користувачі наділяються вхідними іменами та об'єднуються у групи. Для користувачів вказуються допустимі вихідні та цільові мережеві адреси, діапазон дат та часу роботи, а також схеми та порядок автентифікації.

Визначення набору сервісів, що використовуються, виконується на основі вбудованої в дистрибутив брандмауера бази даних, що має значний набір TCP/IP сервісів.

Перевірка сформованих правил на несуперечність виконується автоматично. Виявлені неоднозначності слід усунути шляхом редагування суперечливих правил. Більшість брандмауерів після формування бази правил виконують процес остаточного налаштування автоматично.

Перевірка відповідності параметрів налаштування брандмауера розробленої політики міжмережевої взаємодії може виконуватись на основі аналізу протоколів роботи міжмережевого екрану. Однак найбільшої результативності такої перевірки буде досягнуто при використанні спеціалізованих систем аналізу захищеності мережі.

4.8 Системи виявлення вторгненням (IDS). Загальні відомості

Системи виявлення вторгнень (*англ. Intrusion Detection System – IDS*) – це системи, що збирають інформацію з різних точок комп'ютерної системи (обчислювальної мережі), що захищається, і аналізують цю інформацію для виявлення як спроб порушення, так і реальних порушень захисту (вторгнень).

На відміну від міжмережевих екранів, які блокують підозрілий трафік, IDS пропускає будь-який трафік, аналізуючи його та сигналізуючи при виявленні підозрілої активності.

Класифікувати IDS можна за кількома параметрами (рис. 4.13). Вибір тієї чи іншої типу IDS має робитися з аналізу завдань, які ставляться перед системою виявлення.

Рисунок 4.13. Класифікація систем виявлення вторгнень

За способом реагування системи виявлення вторгнень поділяються на динамічні (системи реального часу) та статичні (системи відкладеної обробки). Системи відкладеної обробки аналізують вміст журналів реєстрації подій або масив попередньо записаного трафіку, а системи реального часу – потік подій від програмних датчиків. Очевидно, що адекватне реагування на спробу реалізації атаки, включаючи її запобігання, можливе лише за умови використання систем реального часу. У той самий час це означає, що IDS реального часу краще, ніж системи відкладеної обробки. Основною метою використання систем виявлення атак реального часу є швидке реагування на спроби реалізації атак, у тому числі припинення цих спроб. У зв'язку з цим типовими процедурами для цих систем є аналіз та фільтрація трафіку на мережевому та транспортному рівнях моделі OSI. Щоб скоротити продуктивні витрати, часто розглядаються лише заголовки пакетів, вміст пакетів «відкидається». Це, очевидно, значно скорочує перелік атак, що виявляються.

За способом збору інформації розрізняють мережні та системні IDS. IDS, які встановлюються на хості і виявляють злонамірні дії на ньому, називаються хостовими або системними IDS (Host-based Intrusion Prevention, HIPS). Як правило, IDS системного рівня контролюють систему, події та журнали реєстрації подій безпеки (security log або syslog) у мережах. Коли якийсь із цих файлів змінюється, IDS порівнює нові записи з сигнатурами атак, щоб перевірити, чи є відповідність. Якщо таке відповідність знайдено, то система посилає адміністратору сигнал тривоги чи приводить у дію інші механізми реагування.

На противагу їм системи виявлення вторгнень, які орієнтовані на аналіз всього доступного мережного трафіку, називають мережевими (Network-based Intrusion Prevention, NIPS). Мережеві IDS збирають і аналізують всі доступні їм мережеві пакети щодо наявності «підозрілого» вмісту чи несанкціонованих потоків інформації від одного вузла мережі до іншого. У зв'язку з цим, точка підключення IDS повинна забезпечувати максимальне охоплення трафіку, що циркулює в сегменті мережі. Зазвичай такі системи підключаються до спеціального порту комутатора (рис. 4.14) або встановлюються безпосередньо на маршрутизаторі мережі.

Необхідно відзначити, що аналіз інтенсивного потоку даних вимагає суттєвих обчислювальних витрат, тому апаратні вимоги до вузла, на якому встановлюється IDS, можуть бути дуже високими.

У ряді джерел виділяють ще один клас IDS – IDS для бездротових мереж (Wireless Intrusion Prevention Systems, WIPS).

За методами аналізу IDS поділяють на три групи: IDS, які порівнюють інформацію з встановленою базою сигнатур атак, IDS, що контролюють частоту подій або виявлення статистичних аномалій (виявлення зловживань) та IDS, що виявляє вторгнення, використовуючи методи штучного інтелекту. Кожен із цих напрямків має свої переваги та недоліки, тому в більшості існуючих IDS застосовуються комбіновані рішення, засновані на синтезі відповідних методів.

Аналіз сигнатур був першим методом, застосованим для виявлення вторгнень. Він базується на простому понятті збігу послідовності із зразком. У пакеті проглядається байт за байтом і порівнюється з сигнатурою – характерним рядком програми, що вказує на характеристику шкідливого трафіку. Така сигнатура може містити ключову фразу чи команду, яка пов'язана із нападом. Якщо збіг знайдено – оголошується тривога. Основна перевага систем аналізу сигнатур полягає в тому, що вони зосереджуються на аналізі даних і зазвичай породжують дуже мало помилкових тривог.

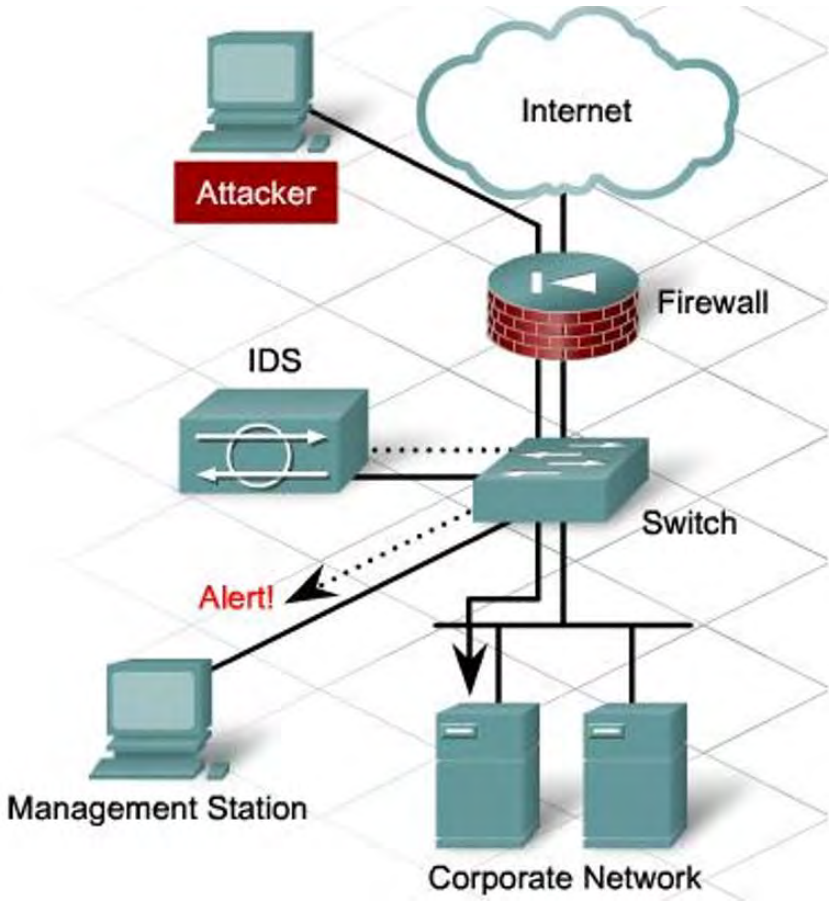


Рисунок 4.14. Схема підключення мережевої IDS

Головний недолік систем аналізу сигнатур пов'язаний з тим, що вони можуть визначати лише відомі атаки, для яких існують певна сигнатура. У міру виявлення нових атак розробники повинні будувати відповідні моделі, додаючи їх до бази сигнатур.

Ідея методів, що використовуються для виявлення аномалій (іноді цей метод також називають методом виявлення порушення профілю), полягає в тому, щоб розпізнати, чи процес, що викликав зміни в роботі системи, є діями зловмисника. Для цього автори IDS впровадили інструменти (детектори аномалій), що створюють так звані профілі, які описують нормальну поведінку користувача, хосту, мережі. Ці профілі створюють на підставі статистичних даних подій в мережі (хосту) під час нормальної поведінки. Якщо під час роботи мережі щось порушує ці стандартні профілі (таке порушення називається аномалією), то можлива зловмисність.

З іншого боку, атака характеризується певної послідовністю протокольних повідомлень. Мета IDS другого напряму (виявлення аномалій) – пошук послідовностей подій, визначених (адміністратором безпеки чи експертом під час навчання IDS) як етапи реалізації вторгнення.

Головна перевага систем виявлення аномалій полягає в тому, що вони можуть виявляти раніше невідомі атаки. Визначивши, що така «нормальна» поведінка, можна виявити будь-яке порушення, незалежно від того, передбачено воно моделлю потенційних загроз чи ні.

У реальних системах, однак, перевага виявлення раніше невідомих атак зводиться нанівець великою кількістю помилкових тривог.

Різновидом систем виявлення аномалій є експертні системи з інтелектуальним механізмом обробки трафіку, що проходить через систему. Зокрема, сучасні системи виявлення аномалій використовують методи штучного інтелекту, наприклад, нейронні мережі, які суттєво знижують ризик хибної тривоги.

4.9 Архітектура системи виявлення вторгнень

Структура системи виявлення вторгнень представлена на рис. 4.15.

У сучасних системах виявлення виділяють такі основні елементи: підсистему збору інформації, підсистему аналізу та модуль ухвалення рішення та представлення даних:

Рисунок 4.15. Структура системи виявлення вторгнень

• підсистема збору інформації – використовується для збору первинної інформації про роботу системи, що захищається;

• підсистема аналізу (виявлення) – здійснює пошук атак і вторгнень у систему, що захищається;

• підсистема ухвалення рішення та представлення даних (інтерфейс користувача) – дозволяє користувачам IDS стежити за станом системи, що захищається.

Крім того, до складу IDS зазвичай входить підсистема протоколювання та аудиту.

Підсистема збору інформації акумулює дані про роботу системи, що захищається. Для збирання інформації використовуються автономні модулі – датчики (сенсори). Кількість використовуваних датчиків різна і залежить від специфіки системи, що захищається. Датчики в IDS прийнято класифікувати

за характером інформації, що збирається. Відповідно до загальної структури інформаційних систем виділяють такі типи:

• датчики додатків – збір даних про роботу програмного забезпечення системи, що захищається;

• датчики хосту – збір даних про функціонування робочої станції системи, що захищається;

• датчики мережі – збирання даних для оцінки мережевого трафіку;

• міжмережеві датчики – збирають характеристики даних, що циркулюють між мережами.

Система виявлення вторгнення може включати будь-яку комбінацію наведених типів датчиків.

Підсистема аналізу структурно складається з одного чи більше модулів аналізу – аналізаторів. Наявність кількох аналізаторів потрібна для підвищення ефективності виявлення вторгнень. Кожен аналізатор виконує пошук атак чи вторгнень певного типу. Вхідними даними для аналізатора є інформація із підсистеми збору інформації або від іншого аналізатора. Результат роботи підсистеми – індикація про стан системи, що захищається. Якщо аналізатор повідомляє про виявлення несанкціонованих дій, на його виході може з'являтися деяка додаткова інформація. Зазвичай ця інформація містить висновки, що підтверджують наявність вторгнення або атаки.

Підсистема ухвалення рішення та подання даних необхідна для інформування зацікавлених осіб про стан системи, що захищається. У деяких системах передбачається наявність груп користувачів, кожна з яких контролює певні підсистеми системи, що захищається. Тож у таких IDS застосовується розмежування доступу, групові політики, повноваження тощо.

Підсистема протоколювання та аудиту є обов'язковим компонентом будь-якої АС. Протоколювання, або реєстрація, є механізмом підзвітності системи забезпечення інформаційної безпеки, що фіксує всі події, що стосуються питань безпеки. У свою чергу, аудит – це аналіз протокольної інформації з метою оперативного виявлення та запобігання порушенням режиму інформаційної безпеки. Системи виявлення вторгнень рівня хоста можна вважати, як системи активного аудиту.

Протокольовані дані поміщаються в реєстраційний журнал, який є хронологічно впорядкованою сукупністю записів результатів діяльності суб'єктів АС, достатню для відновлення, перегляду та аналізу послідовності дій з метою контролю кінцевого результату.

Типовий запис реєстраційного журналу представлено на рис. 4.16.

Рисунок 4.16. Типовий запис реєстраційного журналу

4.10 Системи запобігання вторгненням (IPS)

Системи запобігання вторгненням *(англ. Intrusion Prevention System, IPS)* – програмна або апаратна система мережевої та комп'ютерної безпеки, що виявляє вторгнення або порушення безпеки і автоматично захищає від них.

Технологія IPS доповнює технологію IDS тим, що може не тільки самостійно визначити загрозу, а й успішно заблокувати її. У цьому сценарії функціональність IPS набагато ширша, ніж у IDS:

• IPS блокує атаку (обрив сесії користувача, що порушує політику безпеки, блокування доступу до ресурсів, хостів, додатків);

• IPS змінює навколишнє середовище (зміна конфігурації мережевих пристроїв для запобігання атаки);

• IPS змінює зміст атаки (видаляє, наприклад, інфікований файл та відправляє його одержувачу вже очищеним).

Для реалізації такої можливості IPS інакше, в порівнянні з IDS підключається в мережі (рис. 4.17). Якщо IPS лише слідкує за трафіком (див. рис. 4.14), то IDS пропускає (або не пропускає, блокує) трафік крізь себе, за необхідністю корегуючи його.

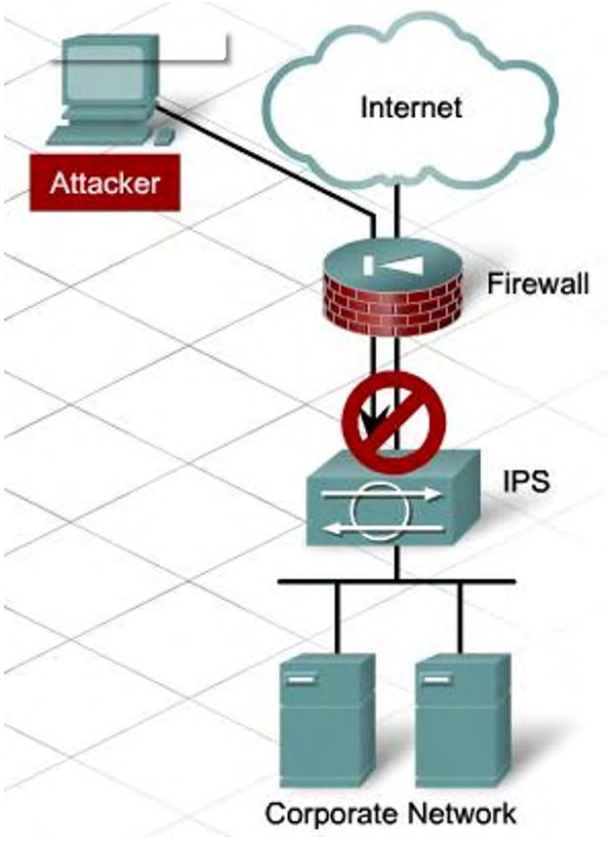


Рисунок 4.17. Підключення IPS в мережі

Але, крім очевидних плюсів, системи IPS мають своїм мінуси. Наприклад, IPS не завжди може точно визначити інцидент або помилково прийняти за інцидент нормальну поведінку трафіку або користувача.

4.11 Запитання до розділу

1) Що функціонально представляє собою міжмережевий екран, що виконує фільтрацію трафіку?

2) Які критерії аналізу інформаційного потоку можуть використовуватися при фільтрацію трафіку?

3) Які захисні функції можуть бути реалізовані в міжмережевому екрані з функцією посередництва?

4) Які недоліки та переваги екрануючих маршрутизаторів?

5) Які функції виконує шлюз сеансового рівня?

6) Яка поведінка шлюзу прикладного рівня, на який поступило повідомлення для деякої служби, а шлюз не має посередника для цієї служби?

7) Що визначає політика доступу до мережевих сервісів?

8) Назвіть відомі Вам схеми підключення фаєрволу до мережі, що захищається?

9) Чим системи запобігання вторгненням (IPS) відрізняється від системи виявлення вторгненням (IDS).

РОЗДІЛ 5 ВІРТУАЛЬНІ ЗАХИЩЕНІ МЕРЕЖІ (VPN)

5.1 Загальні принципи побудови захищених віртуальних мереж (VPN)

5.1.1 Поняття захищеної віртуальної мережі

Безпека інформаційної взаємодії локальних мереж та окремих комп'ютерів через відкриті мережі, наприклад через мережу Internet, потребує якісного вирішення двох базових завдань (рис. 5.1):

• захисту підключених до публічних каналів зв'язку локальних мереж та окремих комп'ютерів від несанкціонованих дій з боку зовнішньої середи;

• захисту інформації в процесі передачі відкритими каналами зв'язку.

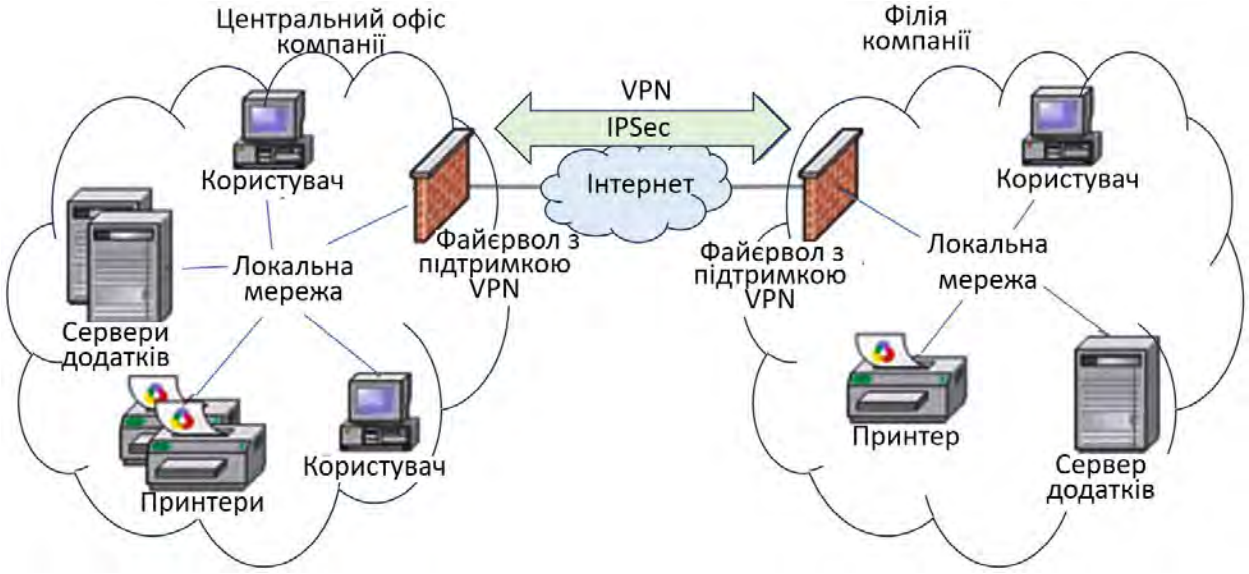


Рисунок 5.1. Приклад організації захисту локальних мереж компанії

Вирішення першої задачі засноване на використанні розглянутих вище міжмережевих екранів (брандмауерів), що підтримують безпеку інформаційної взаємодії шляхом фільтрації двостороннього потоку повідомлень, а також виконання функцій посередництва під час обміну інформацією.

Захист інформації в процесі передачі по відкритих каналах зв'язку ґрунтується на виконанні наступних функцій:

• автентифікації сторін, що взаємодіють;

• криптографічному закритті переданих даних;

• підтвердження справжності та цілісності доставленої інформації;

• захисту від повтору, затримки та видалення повідомлень;

• захисту від заперечення фактів надсилання та отримання повідомлень.

Об'єднання локальних мереж та окремих комп'ютерів через відкрите зовнішнє середовище передачі в єдину віртуальну мережу, що забезпечує безпеку циркулюючих даних, називають захищеною віртуальною мережею (VPN – Virtual Private Network).

Віртуальна мережа формується з урахуванням каналів зв'язку відкритої мережі. Сам термін «віртуальна» підкреслює, що канали зв'язку віртуальної мережі моделюються за допомогою реальних каналів зв'язку. Відкрита мережа може бути основою для одночасного співіснування безлічі віртуальних мереж, кількість яких визначається пропускною здатністю відкритих каналів зв'язку.

Мають місце й інші назви, альтернативні VPN: «захищений канал», «криптографічний тунель» та ін. Терміни «канал», «тунель» підкреслює той факт, що захист даних забезпечується між двома вузлами мережі (хостами або шлюзами) вздовж деякого віртуального шляху, прокладеного в мережі з комутацією пакетів.

Слід зазначити, що актуальним завданням є не тільки захист відкритих каналів за межами локальної мережі, але також захист всередині локальної мережі.

5.1.2 Способи організації захищених віртуальних мереж

Будь-який з двох вузлів віртуальної мережі, між якими формується захищений тунель, може належати кінцевій або проміжній точці потоку повідомлень, що захищається. Відповідно можливі різні способи утворення захищеного віртуального каналу (рис. 5.2).

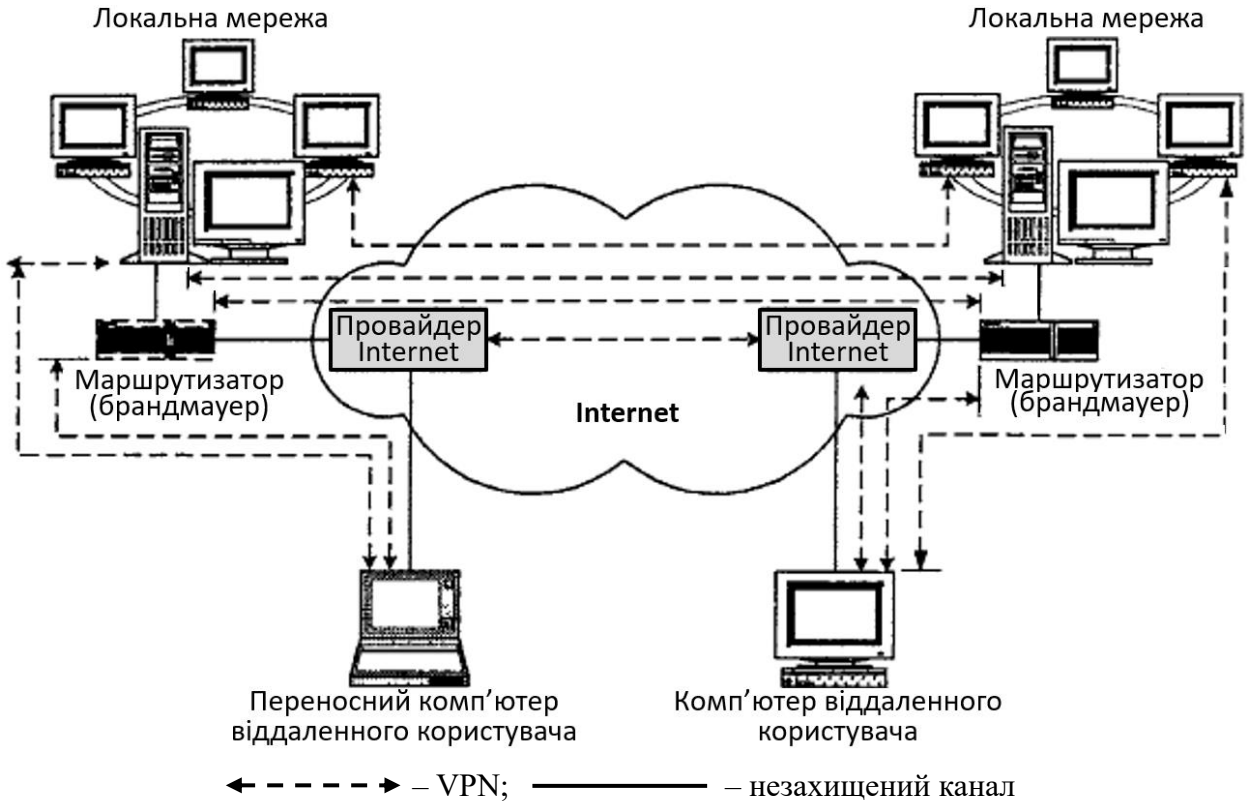


Рисунок 5.2. Різновид захищених віртуальних каналів

Варіант, коли кінцеві точки захищеного тунелю збігаються з кінцевими точками потоку повідомлень, що захищається, є з точки зору безпеки кращим. У цьому випадку забезпечується повна захищеність каналу вздовж

всього шляху прямування пакетів повідомлень. Однак такий варіант веде до децентралізації управління та надмірності ресурсних витрат. Потрібна установка засобів створення захищених тунелів на кожен клієнтський комп'ютер локальної мережі, що ускладнює централізоване керування доступом до комп'ютерних ресурсів та економічно не завжди є виправданим.

Тому якщо відсутня необхідність захисту трафіку всередині локальної мережі, що входить у віртуальну мережу, то як кінцева точка захищеного тунелю доцільно вибрати брандмауер або прикордонний маршрутизатор цієї локальної мережі.

У випадку ж, коли всередині локальної мережі потік повідомлень також повинен бути захищений, то як кінцева точка тунелю в цій мережі повинен виступати комп'ютер, що представляє одну зі сторін захищеної взаємодії.

Поширений також варіант, що характеризується нижчою безпекою, але вищою зручністю застосування. Згідно з цим варіантом при об'єднанні локальних мереж тунель формується тільки між прикордонними провайдерами Internet.

5.1.3 Ієрархія технологій VPN

VPN можна побудувати за допомогою системних засобів, реалізованих на різних рівнях моделі OSI (рис. 5.3).

<table>
<tr><th>Рівні OSI</th><th>Характеристика VPN</th><th>Приклади VPN</th></tr>
<tr><td>ПРИКЛАДНИЙ</td><td rowspan="3">Впливають на додатки, не залежать від мережної технології</td><td>S/MIME</td></tr>
<tr><td>ПРЕДСТАВНИЦЬКИЙ</td><td rowspan="2">SSL/TLS</td></tr>
<tr><td>СЕАНСОВИЙ</td></tr>
<tr><td>ТРАНСПОРТНИЙ</td><td></td><td></td></tr>
<tr><td>МЕРЕЖНИЙ</td><td rowspan="3">Прозори для додатків, залежать від мережної технології</td><td>IPSec</td></tr>
<tr><td>КАНАЛЬНИЙ</td><td rowspan="2">PPTP, L2TP</td></tr>
<tr><td>ФІЗИЧНИЙ</td></tr>
</table>

Рисунок 5.3. Протоколи, що формують захищений канал на різних рівнях OSI

Якщо захист даних здійснюється засобами верхніх рівнів (прикладного, представницького або сеансового), такий спосіб захисту не залежить від технологій транспортування даних (IP або IPX, Ethernet або ATM), що можна вважати безперечною перевагою. У той самий час додатки стають залежними від конкретного протоколу захищеного каналу, оскільки у цих додатках мають бути вбудовані явні виклики функцій цього протоколу.

Захищений канал, реалізований на найвищому (прикладному) рівні, захищає лише певну мережеву службу, наприклад файлову, гіпертекстову або поштову. Так, протокол S/MIME захищає лише повідомлення електронної

пошти. При такому підході кожної служби необхідно розробляти власну захищену версію протоколу.

Робота протоколу захищеного каналу (SSL – Secure Socket Layer – шар захищених сокетів) на рівнях представницького та сеансового робить його більш універсальним засобом, ніж протокол безпеки прикладного рівня. Однак для того, щоб додаток зміг скористатися протоколом прикладного рівня, до нього доводиться вносити виправлення, хоча й не такі суттєві, як у випадку протоколу прикладного рівня. Модифікація програми в даному випадку зводиться до вбудовування явних звернень до API відповідного протоколу безпеки.

Засоби захищеного каналу стають прозорими для додатків у тих випадках, коли безпека забезпечується на мережному та канальному рівнях. Однак тут ми стикаємося з іншою проблемою – залежністю сервісу захищеного каналу від протоколу нижнього рівня. Наприклад, протокол PPTP захищає кадри лише протоколу PPP канального рівня, упаковуючи їх в IP-пакети.

Протокол IPSec, що працює на мережному рівні, є компромісним варіантом. З одного боку, він прозорий для додатків, з іншого – може працювати практично у всіх мережах, оскільки заснований на поширеному протоколі IP і використовує будь-яку технологію канального рівня (PPP, Ethernet, ATM тощо).

5.2 Архітектура служби IPSec

5.2.1 Базовий набір стандартів служби IPSec

Хоча в ряді джерел використовується термін «протокол IPSec» (скорочення від IP Security), насправді IPSec – це набір протоколів, стандартів, алгоритмів для забезпечення захисту даних, що передаються міжмережевим протоколом IP (рис. 5.4).

Рисунок 5.4. Базовий набір стандартів служби IPSec

Ядро IPSec складають три протоколи:

• AH (Authentication Header – заголовок автентифікації) гарантує цілісність та автентичність даних;

• ESP (Encapsulating Security Payload – інкапсуляція зашифрованих даних) шифрує передані дані, забезпечуючи конфіденційність, може також підтримувати автентифікацію та цілісність даних;

• IKE (Internet Key Exchange – обмін ключами Інтернету) вирішує допоміжне завдання автоматичного надання кінцевим точкам захищеного каналу секретних ключів, необхідних для роботи протоколів автентифікації і шифрування даних.

Для об'єднання всіх компонентів архітектури використовується так звана DOI (Domain of Interpretaion – область інтерпретації). Це, зокрема, ідентифікатори алгоритмів шифрування та автентифікації і навіть деякі параметри, наприклад, тривалості життєвого циклу ключів.

Як видно з короткого опису функцій можливості протоколів AH і ESP частково перекриваються. У той час як AH відповідає тільки за забезпечення цілісності та автентифікації даних, ESP може шифрувати дані та, крім того, виконувати функції протоколу AH (хоча, як ми побачимо пізніше, автентифікація та цілісність забезпечуються ним у дещо урізаному вигляді). ESP може підтримувати або всю групу функцій (автентифікацію/цілісність), або лише шифрування.

Кожен із протоколів AH та ESP може використовуватися як самостійно, так і одночасно з іншим.

Поділ функцій захисту між протоколами AH та ESP викликаний застосовуваною у багатьох країнах практикою обмеження експорту та/або імпорту засобів, що забезпечують конфіденційність даних шляхом шифрування, так що в тих випадках, коли шифрування через чинні обмеження застосовувати не можна, систему можна поставляти тільки із протоколом AH.

5.2.2 Безпечна асоціація

Для того щоб протоколи AH і ESP могли виконувати свою роботу із захисту даних, що передаються, протокол IKE встановлює між двома кінцевими точками логічне з'єднання – тунель безпеки (рис. 5.5).

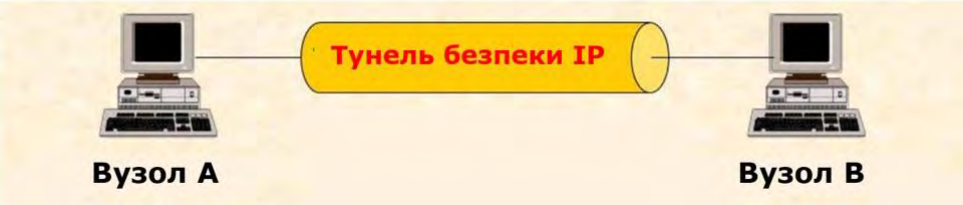


Рисунок 5.5. Тунель безпеки

Процес передачі даних через тунель безпеки передбачає задіяння правил безпеки, прийнятих у системах, між якими ці дані передаються. Ці правила безпеки включають адреси сторін, що з'єднуються, метод форматування (за допомогою якого інформація буде поміщатися в блоки даних іншого протоколу), алгоритми шифрування тощо.

Захищений тунель IP реалізується всіма цими правилами безпеки (включаючи протоколи, методи шифрування тощо). Цей набір правил безпеки (параметрів) тунелю безпеки IP у стандартах IPSec має назву «безпечна асоціація» або «асоціація безпеки» (Security Association, SA). SA і тунель безпеки не одне й те саме. Правильніше сказати, що тунель створюється з урахуванням SA.

Стандарти IPSec дозволяють кінцевим точкам захищеного каналу використовувати єдину безпечну асоціацію для передачі трафіку всіх хостів, що взаємодіють через цей канал, або створювати для цієї мети довільне число безпечних асоціацій, наприклад по одній на кожне TCP-з'єднання. Це дає можливість вибирати потрібний ступінь деталізації захисту кожного додатка.

Безпечна асоціація в протоколі IPSec є односпрямованою (симплексною) логічною сполукою, тому якщо потрібно забезпечити безпечний двосторонній обмін даними, необхідно встановити дві безпечні асоціації.

В IPSec асоціація безпеки включає низку даних, деякі з яких показані на рис. 5.6.

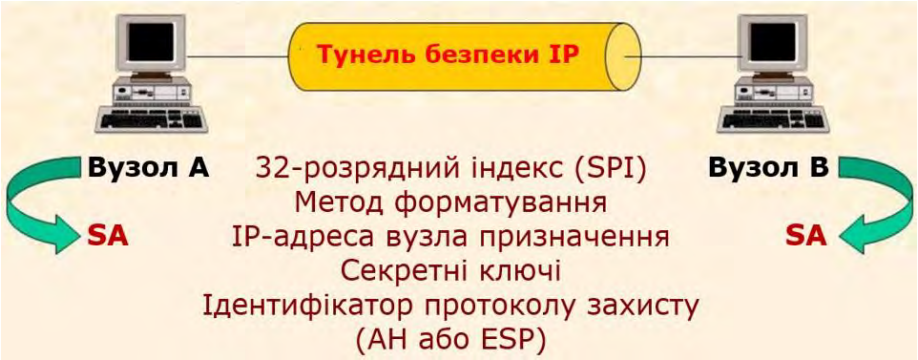


Рисунок 5.6. Асоціація безпеки (SA) та тунель безпеки IP

• IP-адреса одержувача.

• Протокол безпеки, який використовується при передачі даних.

• Секретні ключі, які застосовуються під час шифрування.

• Метод форматування, який визначає, яким чином створюються заголовки та те, яка частина цих заголовків та даних користувача буде захищена в процесі передачі даних.

• Індекс параметрів захисту (Security Parameter Index – SPI) – один із ідентифікаторів SA. Він дозволяє стороні визначити, за допомогою якої асоціації безпеки слід обробляти пакет, що прийшов.

Протокол IPSec допускає як автоматичне, так і ручне встановлення безпечної асоціації. При ручному способі адміністратор конфігурує кінцеві вузли так, щоб підтримували узгоджені параметри асоціації, включаючи

секретні ключі. При автоматичній процедурі встановлення SA протоколи IKE, що працюють з різних боків каналу, вибирають параметри під час переговорного процесу.

Протокол AH або ESP функціонує вже в рамках встановленого логічного з'єднання SA, за його допомогою і здійснюється необхідний захист даних, що передаються з використанням вибраних параметрів.

5.3 Режими роботи IPSec

Протоколи AH та ESP можуть захищати дані у двох режимах: транспортному та тунельному. У транспортному режимі передача IP-пакета через мережу виконується за допомогою оригінального заголовка цього пакета, а в тунельному режимі вихідний пакет поміщається в новий IP-пакет і передача даних через мережу виконується виходячи з заголовка нового IP-пакета.

5.3.1 Транспортний режим

Транспортний режим (рис. 5.7) забезпечує захист насамперед протоколів вищого рівня. Це означає, що захист транспортного режиму поширюється на корисний вантаж пакету IP. Прикладами можуть бути сегменти TCP, дейтаграми UDP або ICMP пакети. Зазвичай транспортний режим забезпечує наскрізний зв'язок двох вузлів (наприклад, клієнта та сервера або двох робочих станцій).

Рисунок 5.7. Транспортний режим IPSec

ESP у транспортному режимі шифрує і, якщо потрібно, автентифікує корисний вантаж IP, але не заголовок IP. AH у транспортному режимі передбачає автентифікацію корисного вантажу IP та деяких частин заголовка IP.

5.3.2 Тунельний режим

Тунельний режим (рис. 5.8) забезпечує захист всього пакета IP. Щоб виконати це завдання, після додавання до пакета IP полів AH чи ESP весь пакет разом із полями захисту розглядається як корисний вантаж деякого нового «зовнішнього» пакета IP із новим зовнішнім заголовком IP. Оскільки оригінальний пакет інкапсульований у новий, більший пакет, цей новий пакет може мати зовсім інші параметри джерела та адресата, що, очевидно, посилює захист.

Тунельний режим використовується тоді, коли один або обидва кінці захищеного зв'язку є шлюзами захисту, наприклад брандмауерами або маршрутизаторами, що використовують IPSec. При використанні тунельного режиму розміщені за брандмауерами системи можуть здійснювати захищений обмін даними без застосування IPSec.

Рисунок 5.8. Тунельний режим IPSec

ESP в тунельному режимі шифрує і, якщо потрібно, автентифікує весь внутрішній пакет IP, включаючи внутрішній заголовок IP. AH у тунельному режимі автентифікує весь внутрішній пакет IP та деякі частини зовнішнього заголовка IP.

5.3.3 Застосування режимів IPSec

Застосування того чи іншого режиму залежить від вимог, що пред'являються до захисту даних, а також від ролі, яку грає в мережі вузол, який завершає захищений канал. Такий вузол може бути хостом (кінцевим вузлом) чи шлюзом (проміжним вузлом). Відповідно, є три схеми застосування IPSec:

- «хост-хост»;
- «шлюз-шлюз»;
- «хост-шлюз».

У схемі «хост-хост» захищений канал встановлюється між двома кінцевими вузлами мережі (див. рис. 5.9). Тоді протокол IPSec працює на кінцевих вузлах і захищає дані, що передаються від хосту 1 до хосту 2 Для схеми «хост-хост» найчастіше використовується транспортний режим захисту.

Рисунок 5.9. Схема захисту «хост-хост»

Відповідно до схеми «шлюз-шлюз» захищений канал встановлюється між двома проміжними вузлами, так званими шлюзами безпеки (Security Gateway,

SG), на кожному з яких працює протокол IPSec (рис. 5.10). Захищений обмін даними може відбуватися між будь-якими двома кінцевими вузлами, підключеними до мереж, які розташовані за шлюзами безпеки. Від кінцевих вузлів підтримка протоколу IPSec не вимагається, вони передають свій трафік в незахищеному вигляді через внутрішні мережі підприємств, що заслуговують на довіру. Трафік, що направляється в загальнодоступну мережу, проходить через шлюз безпеки, який забезпечує його захист за допомогою протоколу IPSec. В цій схемі захисту доступний тільки тунельний режим роботи.

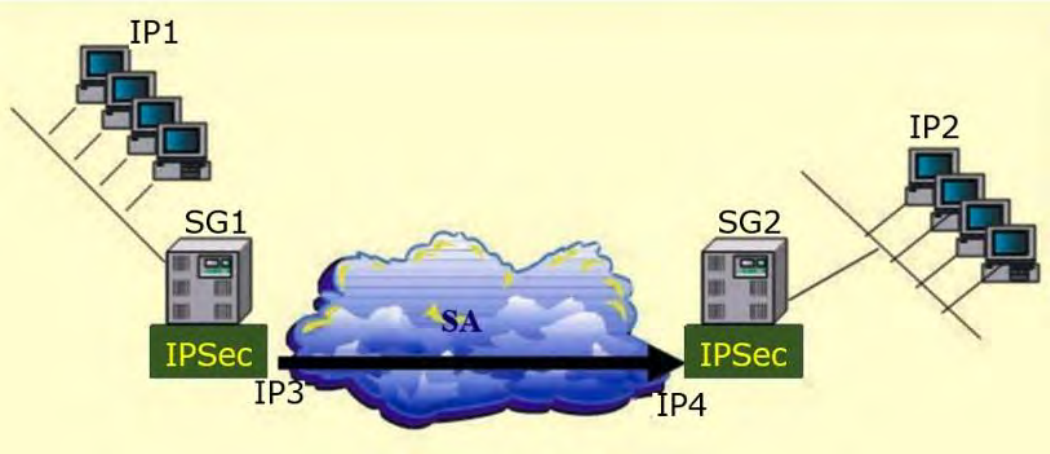


Рисунок 5.10. Схема захисту «шлюз-шлюз»

На рис. 5.10 користувач комп'ютера з адресою IP1 посилає пакет за адресою IP2, використовуючи тунельний режим протоколу IPSec. Шлюз SG1 зашифровує пакет повністю, разом із заголовком, і забезпечує його новим IP-заголовком, в якому як адресу відправника вказує свою адресу – IP3, а як адреса одержувача – адресу IP4 шлюзу SG2 Вся передача даних зовнішньої IP-мережі виконується виходячи з заголовка зовнішнього пакета, а внутрішній пакет стає у своїй мережі полем даних зовнішнього пакета.

Схема «хост-шлюз» (рис. 5.11) часто застосовується при віддаленому доступі. У цьому випадку захищений канал прокладається між віддаленим хостом, на якому працює протокол IPSec, та шлюзом, що захищає трафік для всіх хостів, що входять у внутрішню мережу підприємства.

Рисунок 5.11. Схема захисту «шлюз-хост»

5.4 Протокол AH

5.4.1 Призначення протоколу AH та його заголовок

Протокол AH дозволяє приймальній стороні переконатися в наступному:

- пакет був відправлений стороною, з якою встановлено цю асоціацію;
- вміст пакета не був спотворений у процесі передачі його через мережу;
- пакет не є дублікатом деякого пакета, отриманого раніше.

Дві перші функції обов'язкові для протоколу AH, а остання – факультативна – вибирається під час встановлення асоціації. Для цих функцій протокол AH використовує заголовок наступного виду (рис. 5.12).

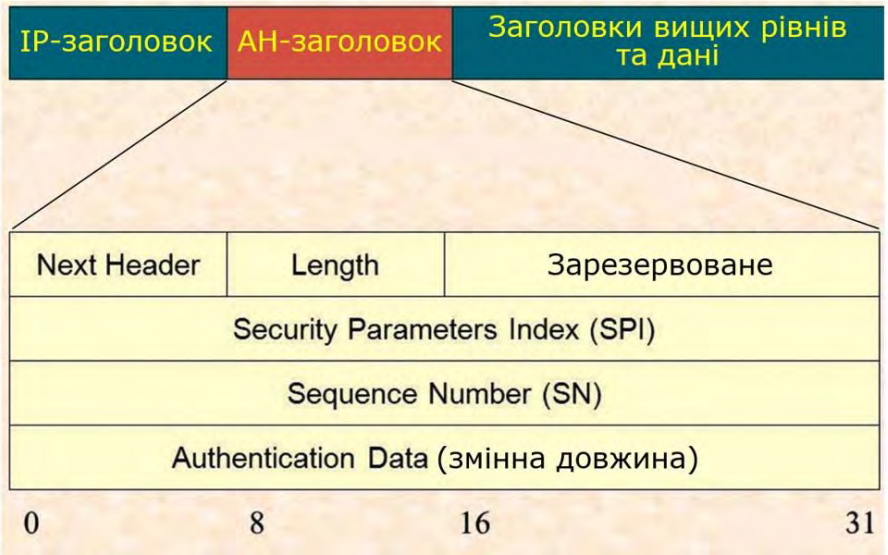


Рисунок 5.12. Заголовок AH (транспортний режим)

Заголовок AH складається з наступних полів:

• Next Header – наступний заголовок (8 біт). Вказується код типу протоколу вищого рівня, тобто протоколу, повідомлення якого розміщено на полі даних пакета IP. Швидше за все, їм буде один із протоколів

транспортного рівня (TCP або UDP) або протокол ICMP, але може зустрітися і протокол ESP, якщо він використовується в комбінації з AH.

• Payload Length – довжина корисного вантажу (8 біт). Довжина заголовка автентифікації в 32-бітових словах, зменшена на 2 Наприклад, встановлена за замовчуванням довжина поля даних автентифікації дорівнює 128 біт, або чотирьом 32-бітовим словам. Разом із заголовком фіксованої довжини три слова загальна довжина всього заголовка виявляється рівної семи словах, у полі довжини корисного вантажу у такому разі вказується значення 5.

• Security Parameters Index (SPI) – індекс параметрів безпеки (32 біта). Використовується для зв'язку пакета із передбаченою йому безпечною асоціацією.

• Sequence Number (SN) – номер у послідовності (32 біта). Вказує на послідовний номер пакета та застосовується для захисту від його помилкового відтворення, коли третя сторона намагається повторно використовувати захищені пакети, що перехоплені. Значення цього поля збільшує в кожному новому пакеті, що передається в рамках даної асоціації, так що прихід дубліката буде помічений стороною, що приймає.

• Authentication Data – дані автентифікації (змінна довжина). Містить так зване значення контролю цілісності (Integrity Check Value, ICV) зване також дайджестом. Обчислення дайджесту в транспортному (рис. 5.13) або тунельному (рис. 5.14) режимах реалізується з використанням односторонньої функції MAC, як аргумент якої виступає вміст пакета, а як параметр – симетричний секретний ключ, який був заданий для даної асоціації вручну або автоматично за допомогою протоколу IKE. Оскільки довжина дайджесту залежить від обраної функції, це поле має у загальному випадку змінний розмір.

Протокол AH намагається охопити при обчисленні дайджесту якнайбільше полів вихідного IP-пакета, але деякі з них (наприклад, значення поля часу життя – Time To Live, TTL – і, відповідно, контрольної суми) змінюються непередбачуваним чином у процесі передачі пакета по мережі і тому не можуть бути включені до автентифікованої частини пакета. При обчисленні дайджесту в якості вхідного параметра використовується весь AH-пакет, але поля зовнішнього IP-заголовка, що змінюються, і поле «дані автентифікації» AH-заголовка замінюються нулями.

Розташування заголовка AH у пакеті залежить від того, в якому режимі – транспортному або тунельному – налаштований захищений канал.

5.4.2 Протокол AH у транспортному режимі

Результуючий пакет у транспортному режимі виглядає так, як показано на рис. 5.13.

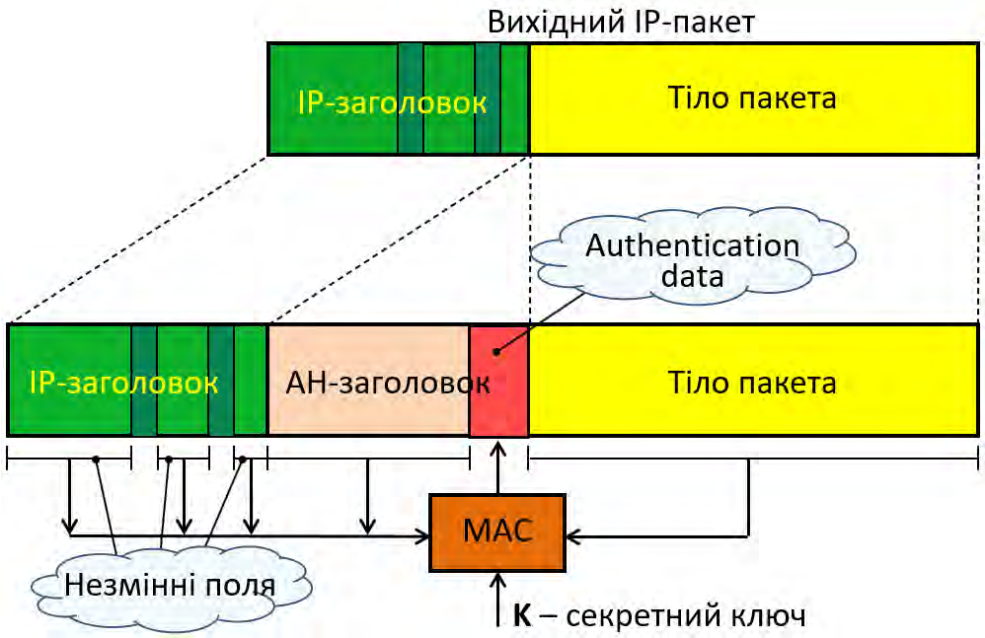


Рисунок 5.13. Формування поля Authentication Data (транспортний режим)

5.4.3 Протокол AH у тунельному режимі

При використанні тунельного режиму, коли шлюз IPSec приймає вихідний пакет, що проходить через нього транзитом і створює для нього зовнішній IP-пакет, протокол AH захищає всі поля вихідного пакета, а також незмінні поля нового заголовка зовнішнього пакета (рис. 5.14).

У новому заголовку шлюз вказує як адресу джерела IP-адресу свого інтерфейсу із загальнодоступною мережею. Як адреса призначення зовнішнього пакета вказується IP-адреса приймаючого шлюзу. Вся передача даних по складовій мережі IP виконується на підставі заголовка зовнішнього пакета, а внутрішній пакет стає полем даних для зовнішнього.

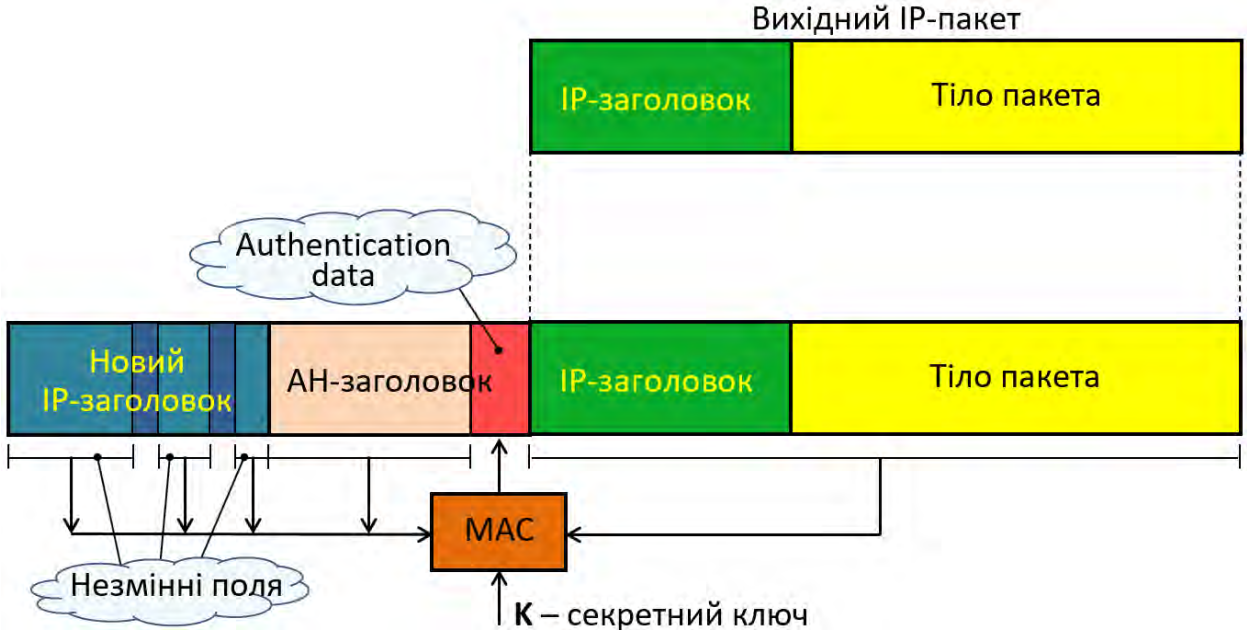


Рисунок 5.14. Формування поля Authentication Data (тунельний режим)

5.5 Протокол ESP

5.5.1 Призначення та формат службової інформації протоколу ESP

Протокол ESP вирішує дві групи завдань. До першої з них відносяться функції, аналогічні функцій протоколу AH, – це забезпечення автентифікації та цілісності даних на основі дайджесту (ця функція опціональна), а до другої – захист даних від несанкціонованого перегляду шляхом шифрування даних, що передаються.

Для вирішення своїх завдань протокол ESP використовує службову інформацію, як це показано на рис. 5.15.

На рис. 5.15 позначено:

• Security Parameters Index (SPI) – індекс параметрів захисту (32 біт). Ідентифікує захищений зв'язок.

• Sequens Number (SN) – порядковий номер (32 біт). Значення лічильника, що використовується для захисту від атак відтворення, як і під час використання протоколу AH.

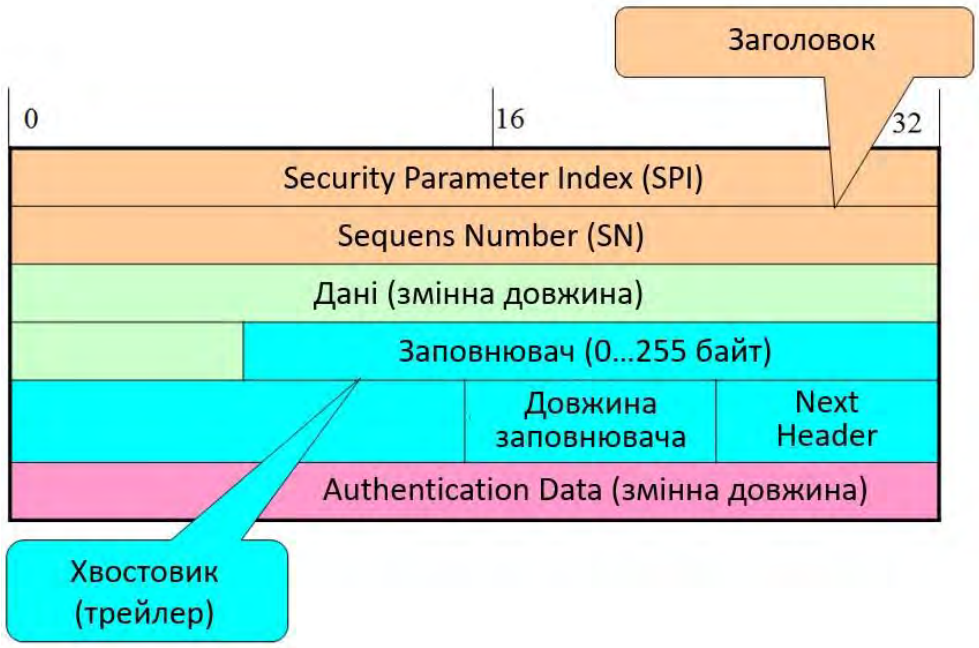


Рисунок 5.15. Розташування службових полів у ESP-пакеті

• Дані (змінної довжини). Це сегмент транспортного рівня (у транспортному режимі) або пакет IP (у тунельному режимі), що захищається шифруванням.

• Заповнювач (0-255 байт). Призначення цього поля пояснюється нижче.

• Довжина заповнювача (8 біт). Вказує кількість байтів заповнювача, що передує цьому полю.

• Next Header – наступне заголовок (8 біт). Ідентифікує тип даних, що містяться в полі даних корисного вантажу, вказуючи код першого заголовку корисного вантажу (заголовок розширення IPv6 або заголовок протоколу верхнього рівня, наприклад TCP).

• Дані автентифікації (змінної довжини). Поле змінної довжини (яка повинна являти собою ціле число 32-бітових слів), що містить код ICV (Integrity Check Value – код контролю цілісності), що обчислюється для всього пакету ESP без поля даних автентифікації.

Деякі службові поля аналогічні полям заголовка AH: Next Header, SPI, SN, Authentication Data. Але є і два додаткові поля – заповнювач (Padding) і довжина заповнювача (Pad Length). Заповнювач може знадобитися у трьох випадках. По-перше, для нормальної роботи деяких алгоритмів шифрування необхідно, щоб текст, що шифрується, містив кратне число блоків певного розміру. По-друге, формат заголовка ESP вимагає, щоб поле даних закінчувалося на межі чотирьох байтів. І, нарешті, заповнювач можна використовуватиме для приховування дійсного розміру пакету з метою забезпечення часткової конфіденційності трафіку.

Як видно з рис. 5.15, службова інформація поділяється на дві частини, що поділяються полем даних. Перша частина, яка далі позначатиметься як заголовок ESP, утворюється двома полями – SPI і SN і розміщується перед полем даних. Безпосередньо за полем даних слідує так званий трейлер, або хвіст, до якого входять заповнювач (Padding), довжина заповнювача (Pad Length), а також покажчик на протокол наступного рівня (Next Header). Завершує пакет поле контролю цілісності (Authentication Data). У тому випадку, коли при встановленні безпечної асоціації прийнято рішення не використовувати можливості ESP щодо забезпечення цілісності, це поле відсутнє.

5.5.2 Протокол ESP у транспортному режимі

Транспортний режим ESP служить для шифрування і, якщо потрібно, автентифікації даних, що пересилаються в пакеті IP (наприклад, сегмента TCP). У цьому режимі формування ESP-пакета відбувається наступним чином (рис. 5.16).

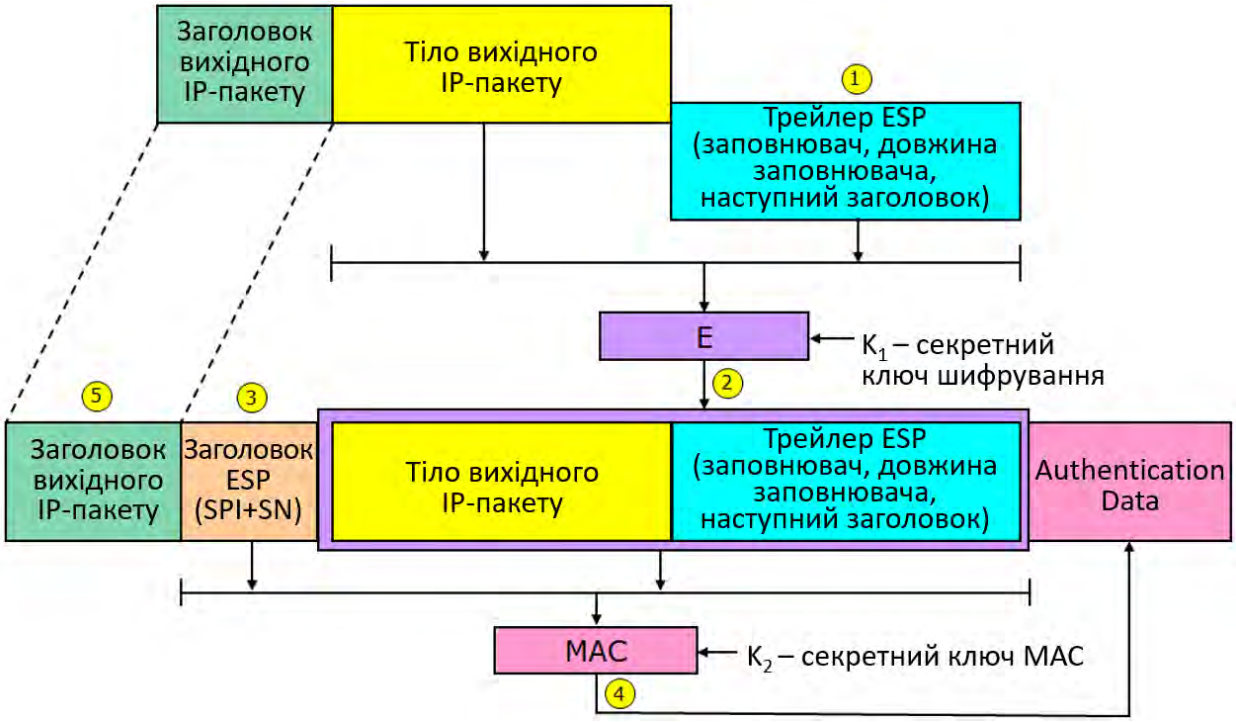


Рисунок 5.16. Формування ESP-пакету у транспортному режимі

1 Формування трейлера ESP.

2 Шифрування тіла вихідного IP-пакета разом із приєднаним до нього трейлером. Використовується виключно симетричне шифрування.

3 Формується заголовок ESP. ESP не шифрує поля SPI і SN цього заголовка, які повинні передаватися у відкритому вигляді для того, щоб пакет, що прибув, можна було віднести до певної асоціації і захиститися від помилкового відтворення пакета.

4 Якщо використовується функція автентифікації, з використанням MAC формується поле даних автентифікації ESP (Authentication Data), яке додається після кінцевика ESP. Як аргумент MAC тут виступає зашифрований вміст поля вихідного IP-пакета та хвостовика, а також незашифрований вміст заголовка ESP.

5 На початок ESP-пакета додається IP-заголовок вихідного IP-пакета. У цьому режимі ESP не шифрує заголовок IP-пакета, інакше маршрутизатор не зможе прочитати поля заголовка та коректно здійснити просування пакета між мережами.

Транспортний режим забезпечує конфіденційність для будь-якого програми, що використовує цей режим, що дозволяє уникнути необхідності реалізації функцій захисту в кожному окремому додатку. Цей режим досить ефективний, а обсяг доданих до пакету IP даних при цьому невеликий. Недоліком цього режиму є те, що при його використанні не виключається можливість аналізу трафіку пакетів, що пересилаються, за вмістом їх заголовків.

Як видно із рис. 5.16, на відміну від протоколу AH, контроль цілісності та автентичності даних у протоколі ESP не поширюється на заголовок вихідного

пакета, і з цієї причини має сенс застосовувати обидва протоколи спільно – ESP для шифрування, а AH для контролю цілісності.

5.5.3 Протокол ESP у тунельному режимі

Тунельний режим ESP передбачає шифрування всього пакета IP (рис. 5.17). У цьому режимі весь вихідний IP-пакет разом із його заголовком та разом із кінцевиком ESP шифруються (п.2). До зашифрованих даних додається, як префікс, заголовок ESP (п.3), дані автентифікації, якщо вони є (п. 4), і весь цей блок міститься у зовнішній пакет IP з новим заголовком (п.5), який міститиме достатньо інформації для маршрутизації, але не для аналізу трафіку. Цей режим доцільно використовувати, коли потрібно виключити можливість проведення атак, побудованих на аналізі трафіку.

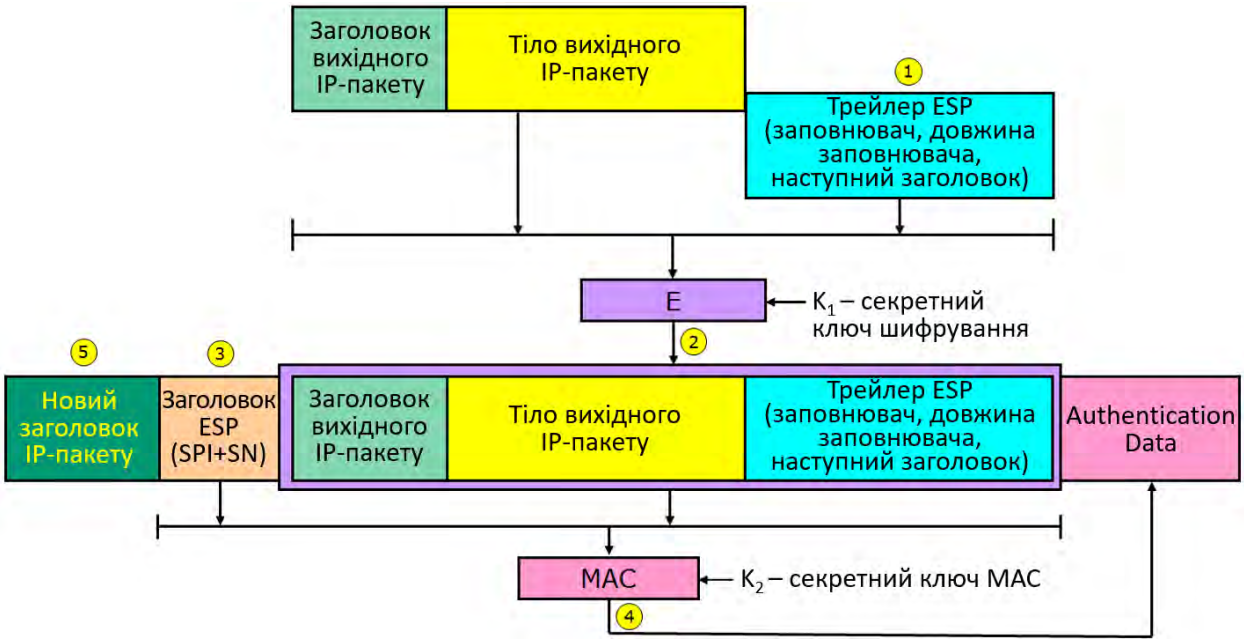


Рисунок 5.17. Формування ESP-пакету в тунельному режимі

У той час як транспортний режим підходить для захисту з'єднань між вузлами, які підтримують сервіс ESP, тунельний режим виявляється корисним у конфігурації, яка передбачає наявність брандмауера або іншого шлюзу захисту внутрішньої мережі від зовнішніх мереж. У тунельному режимі шифрування використовується для обміну лише між зовнішнім вузлом та шлюзом захисту або між двома шлюзами захисту. Це розвантажує вузли внутрішньої мережі, позбавляючи їх необхідності шифрування даних, і спрощує процедуру розподілу ключів, зменшуючи загальну кількість необхідних ключів.

5.6 Бази даних SAD та SPD

Технологія IPSec пропонує різноманітні методи захисту трафіку. Для вибору способу захисту, який повинен застосувати до трафіку, у кожному вузлі, який підтримує IPSec, формуються два типи баз даних:

- база даних безпечних асоціацій (Security Associations Database, SAD);
- база даних політики безпеки (Security Policy Database, SPD).

При встановленні безпечної асоціації (SA) дві сторони беруть низку угод, що регламентують процес передачі потоку даних між ними. Набори поточних параметрів, що визначають всі активні асоціації, зберігаються на обох вузлах захищеного каналу у вигляді баз даних безпечних асоціацій (SAD). Кожен вузол IPSec підтримує дві бази SAD – одну вихідних асоціацій, іншу для вхідних.

Інший тип бази даних – база даних політики безпеки (SPD) – визначає відповідність між IP-пакетами та встановленими для них правилами обробки. Записи SPD складаються з полів двох типів – полів селектора пакета та полів політики захисту для пакета з цим значенням селектора (рис. 5.18).

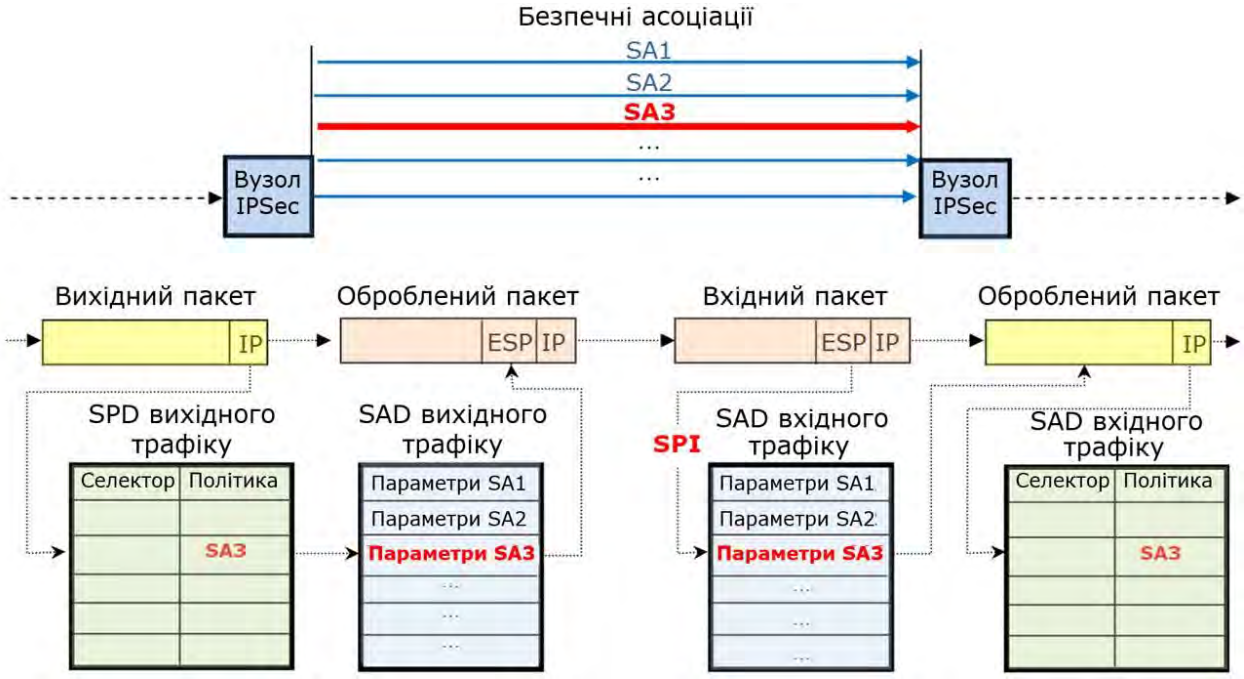


Рисунок 5.18. Використання баз даних SPD та SAD

Селектор в SPD включає наступний набір ознак, на підставі яких можна з великим ступенем деталізації виділити потік, що захищається:

• IP-адреси джерела та приймача;
• порти джерела та приймача (тобто TCP- або UDP-порти);
• тип протоколу транспортного рівня (TCP, UDP);
• ім'я користувача у форматі DNS або X.500;
• ім'я системи (хосту, шлюзу безпеки тощо) у форматі DNS або X.500.

Бази даних політики безпеки створюються та адмініструються або користувачем (цей варіант більше підходить для хосту), або системним адміністратором (варіант для шлюзу), або автоматично (протоколом IKE).

Для кожного IP-пакета, що надходить до захищеного каналу, IPSec переглядає всі записи в базі SPD і порівнює значення селекторів цих записів з відповідними полями IP-пакету. Якщо значення полів збігається з будь-яким

селектором, то над пакетом виконуються дії, визначені у полі політики безпеки цього запису. Політика передбачає або передачу пакета без зміни, або відкидання пакету, або обробку пакета засобами IPSec.

В останньому випадку поле політики захисту має містити посилання на запис у базі даних SAD, в яку розміщено набір параметрів безпечної асоціації для цього пакета (на рис. 5.18 для вихідного пакета визначено асоціацію SA3). На підставі заданих параметрів безпечної асоціації до пакету застосовується відповідний протокол (на рис. 5.18 – ESP), алгоритм шифрування та секретні ключі.

Коли пакет приходить у кінцевий вузол захищеного каналу, з його зовнішнього заголовка ESP або AH (на рис. 5.18 – із заголовка ESP) витягується значення SPI (воно не зашифроване) і подальша обробка пакета виконується з урахуванням всіх параметрів асоціації, заданої цим покажчиком.

Після розшифрування пакета приймальний вузол IPSec перевіряє його ознаки (стали доступними) на предмет збігу з селектором запису SPD для вхідного трафіку, щоб переконатися, що помилки не відбулося і обробка пакета, що виконується, відповідає політиці захисту, заданої адміністратором.

Відповідне налаштування бази SDP дозволяє вибирати потрібний ступінь деталізації захисту – від застосування однієї загальної асоціації для трафіку великої кількості кінцевих вузлів до захисту кожної окремої програми за допомогою індивідуально налаштованої асоціації.

5.7 Розташування IPSec

Конкретне розташування IPSec в апаратних засобах або програмному забезпеченні не визначено у вимогах до специфікації Інтернету, а лише в рекомендаціях. Є три найбільш ймовірні сценарії. Першим сценарієм буде розміщення IPSec прямо у вихідний код IP, написаний для хосту або шлюзу безпеки. Цей підхід називається «запхнути в код» (bump-in-the-code – BITC). Він вимагає спеціально написаного мережного драйвера, який може бути розширеним IP-драйвером, що включає в себе функції IPSec (рис. 5.19).

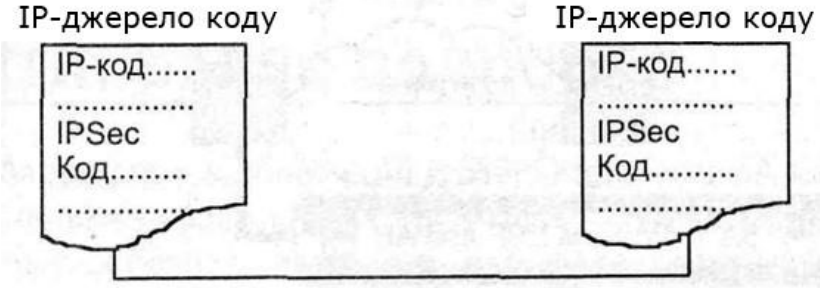


Рисунок 5.19. Сценарій BITC

Другий сценарій – IPSec поміщається в стек протоколів під IP, а це означає, що він буде діяти поверх драйвера IP (рис. 5.20). Цей підхід міг би підійти хостам, коли IPSec виконується поверх управління доступу до носія (media access control – MAC). Цей підхід називається «запхнути в стек»

(bump-in-the-stack – BITS). Він вимагає додаткового, IPSec драйвера, який перехоплюватиме звернення IP-драйвера до драйвера L_2, обробляти (якщо вимагає політика SPD) пакет, який IP-драйвер передає на нижній рівень і вже оброблений пакет передавати драйверу L_2.

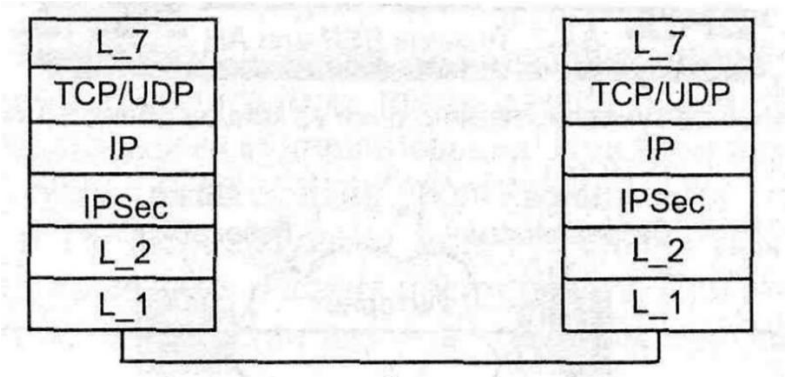


Рисунок 5.20. Сценарій BITS

Третім сценарієм є використання окремого обладнання, яке приєднується до хосту (рис. 5.21) чи шлюзу. Наприклад, такий пристрій може бути спеціальним процесором шифрування. Цей підхід називається «запхнути в залізо» (bump-in-the-wire – BITW). До такого роду пристроїв зазвичай звертаються за IP-адресою, і, якщо воно приєднано до хосту, то воно виглядає як BITS. Якщо ж воно використовується з маршрутизатором або брандмауером, воно буде виглядати як шлюз безпеки і має бути налаштовано як доповнення захисних функцій брандмауера.

Рисунок 5.21. Сценарій BITW

5.8 Запитання до розділу

1) Які протоколи входять до складу системи IPSec?
2) Що таке безпечна асоціація (в контексті стандартів IPSec)?
3) Чим відрізняються тунельний і транспортний режими в системі IPSec?
4) Які функції захисту реалізує протокол AH?
5) Які функції захисту реалізує протокол ESP?
6) Чим відрізняється функція автентифікації ESP відрізняється від функції автентифікації AH?
7) Яке призначення базі даних політики безпеки (SPD) в системі IPSec?
8) Яке призначення бази даних безпечних асоціацій (SAD) в системі IPSec?

РОЗДІЛ 6 ЗАХИЩЕНІ ВІРТУАЛЬНІ МЕРЕЖІ (VPN) СЕАНСОВОГО РІВНЯ. ПРОТОКОЛ SSL/TLS

6.1 Загальні відомості

Протокол SSL (Secure Sockets Layer – шар захищених сокетів), орієнтований на захист інформаційного обміну між клієнтом і сервером комп'ютерної мережі, є протоколом сеансового рівня моделі OSI, що використовує для забезпечення безпеки інформаційного обміну криптографічні методи захисту інформації. Конфіденційність переданих даних забезпечується за рахунок їх криптографічного закриття симетричними шифрами, а автентифікація взаємодіючих сторін – за рахунок формування та перевірки цифрового підпису, справжність і цілісність інформації, що передається за рахунок використання MAC.

Ядром протоколу SSL є технологія комплексного використання асиметричних та симетричних криптосистем. Як алгоритми асиметричного шифрування використовуються такі алгоритми, як RSA, а також алгоритм Діффі-Хеллмана. Для формування геш-функцій можуть застосовуватись стандарти MD5 та SHA-1 (для останніх версій – SHA-2). Набір криптографічних алгоритмів симетричного шифрування розширюється. Для автентифікації застосовуються цифрові сертифікати відкритих ключів користувачів (клієнта та сервера), засвідчені цифровим підписом спеціальних сертифікаційних центрів. Підтримуються цифрові сертифікати, які відповідають загальноприйнятому стандарту X.509.

SSL був розроблений компанією Netscape Communications для протоколу HTTPS веб-браузера Netscape Navigator (1995, v.1.0). Далі були розробки наступних версій SSL. Згодом на підставі протоколу SSL v3.0 було розроблено та прийнято (1999 р., v.1.0) стандарт RFC, який отримав ім'я TLS (Transport Layer Security – протокол захисту транспортного рівня). TLS 1.0 можна назвати версією SSL 3.1 Принципи роботи протоколів SSL та TLS близькі між собою; ключовою відмінністю TLS від SSL є наявність підтримки низки розширень, що дозволяють реалізувати сучасні методи захисту інформації.

Найсучасніша версія TLS – TLS 1.3, що вийшла в 2018 році і дуже істотно відрізняється від усіх попередніх версій TLS. Зокрема, радикально змінено логіку встановлення з'єднання. Версія TLS 1.3 не тільки несумісна з попередніми, а й побудована на іншій інженерній ідеології. Відмінність обумовлена переосмисленням моделі загроз: TLS спробували зробити і швидшим, і захищенішим, і більш потайливим.

Через свої позитивні якості SSL/TLS практично витіснив конкуруючі високорівневі протоколи захисту інформаційного обміну і став загальновизнаним неофіційним стандартом захисту в Internet- і intranet-мережах.

Клієнтська частина SSL реалізована у всіх популярних веб-браузерах, а серверна – в більшості як комерційних веб-серверів, так і тих, що розповсюджуються на некомерційних умовах.

Відповідно до протоколу SSL/TLS криптозахищені тунелі створюються між кінцевими точками віртуальної мережі (рис. 6.1).

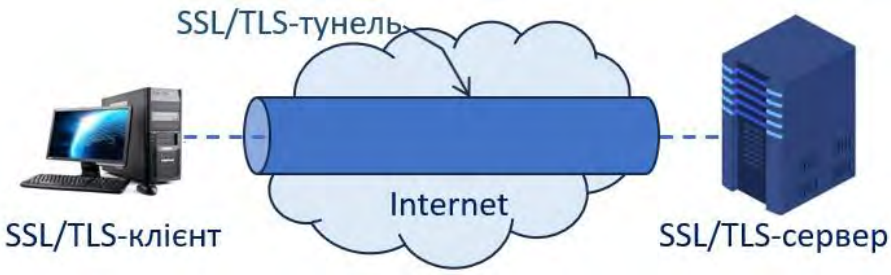


Рисунок 6.1. Криптозахищений тунель, сформований на основі протоколу SSL/TLS

Протокол SSL забезпечує наступні функції захисту інформаційного обміну між кінцевими точками криптотунелю:

- конфіденційність за рахунок шифрування;
- автентифікація сторін (за рахунок цифрового підпису);
- автентифікація (цілісність) інформаційного потоку за рахунок використання HMAC або спеціальних механізмів шифрування.

Протокол SSL/TLS підтримує три режими автентифікації:

- взаємну автентифікацію сторін;
- односторонню автентифікацію сервера без автентифікації клієнта;
- повну анонімність.

При використанні останнього варіанту забезпечується захист інформаційного обміну без будь-яких гарантій щодо справжності сторін. У цьому випадку взаємодіючі сторони не захищені від атак, пов'язаних із заміною учасників взаємодії.

6.2 Сеанси та з'єднання протоколу SSL/TLS

Криптографічні тунелі SSL/TLS формуються на основі сеансу і з'єднань (рис. 6.2).

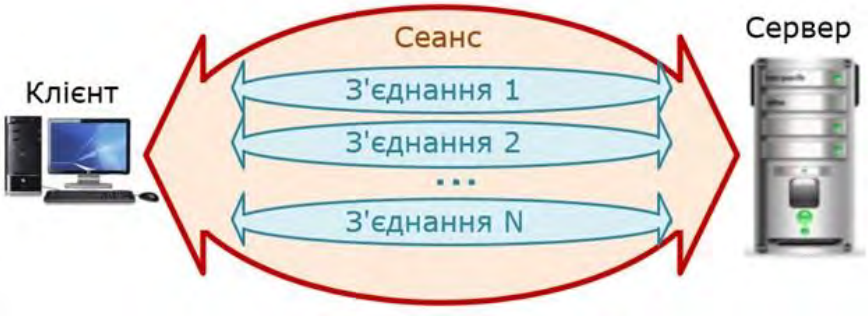


Рисунок 6.2. Сеанс та з'єднання SSL/TLS

- ***З'єднання*** **(*connection*)**. З'єднанням називається транспорт (у термінах моделі OSI), що забезпечує обслуговування деякого відповідного типу. У SSL

такі з'єднання є рівноправними відносинами між вузлами. З'єднання є тимчасовими. Кожне з'єднання асоціюється лише з одним сеансом.

• ***Сеанс* (*session*)**. Сеанс SSL – це зв'язок між клієнтом та сервером. Сеанс визначає набір параметрів криптографічного захисту, які можна використовувати кількома з'єднаннями. Сеанси дозволяють уникнути необхідності вести переговори щодо встановлення параметрів захисту для кожного нового з'єднання.

Між будь-якою парою сторін (наприклад, між HTTP-додатками клієнта і сервера) можна встановити багато захищених з'єднань. Теоретично між сторонами можна встановити кілька одночасно існуючих сеансів, але практично така можливість не використовується.

Сеанси та з'єднання створюються протоколом квітування SSL (SSL Handshake Protocol).

Кожен сеанс характеризується тим чи іншим станом. Наприклад, під час роботи протоколу квитування SSL створюється стан очікування (pending state) читання та запису. Після успішного завершення роботи протоколу квітування SSL стан очікування перетворюється на робочий стан (operating state) для читання і записи (тобто отримання та відправлення інформації).

Протокол SSL передбачає наступні етапи взаємодії клієнта та сервера:

• формування захищеного сеансу та з'єднання (або нового з'єднання в рамках раніше створеного сеансу); в результаті формування сеансу та з'єднання (сеанс не може бути створений без з'єднання) утворюється SSL-сесія;

• захищена взаємодія в рамках SSL-сесії.

У процесі встановлення SSL-сесії вирішуються такі задачі:

• автентифікація сторін (якщо це передбачено);

• узгодження криптографічних алгоритмів та алгоритмів стиснення, які будуть використовуватись при захищеному інформаційному обміні;

• формування спільного секретного майстер-ключа;

• генерація на основі сформованого майстер-ключа загальних секретних сеансових ключів для криптозахисту інформаційного обміну.

Процедура встановлення SSL-сесії, звана також процедурою рукостискання, відпрацьовується перед безпосереднім захистом інформаційного обміну і виконується за протоколом квитування (Handshake Protocol), що входить до складу протоколу SSL.

В рамках сеансу визначаються та формуються глобальні параметри SSL-сесії (табл. 6.1), такі, наприклад, як сертифікат вузла (за його наявності), головний ключ (master-secret) – 48-байтовий секретний ключ, що використовується клієнтом та сервером.

На основі параметрів сеансу формуються локальні параметри з'єднання (табл. 6.1), такі, як секретні сеансові ключі шифрування або автентифікації (HMAC).

При встановленні повторних з'єднань між клієнтом і сервером сторони можуть, за взаємною угодою, формувати нові сеансові ключі з урахуванням

«старого» головного ключа (дана процедура називається «продовженням SSL-сесії»), тобто. створювати нове з'єднання всередині цього сеансу.

Таблиця 6.1

Параметри сеансу та з'єднання

Параметри сеансу	**Параметри з'єднання**
ID сеансу	Випадковий ID
Сертифікат вузла	Ключ запису MAC сервера
Метод стиснення	Ключ запису MAC клієнта
Параметри шифрування (Cipher Spec)	Ключ запису сервера
Головний ключ (48 байт)	Ключ запису клієнта
Прапор відновлення	Вектори ініціалізації (для режимів зчеплення блокових шифрів)
	Порядкові номери повідомлень, що надсилаються в рамках з'єднання

6.3 Архітектура протоколу SSL/TLS

6.3.1 Склад протоколів SSL/TLS

Протокол SSL покликаний забезпечити можливість надійного захисту передачі даних з використанням протоколу TCP. Строго кажучи, SSL є не один протокол, а два рівні протоколів, як показано на рис. 6.3.

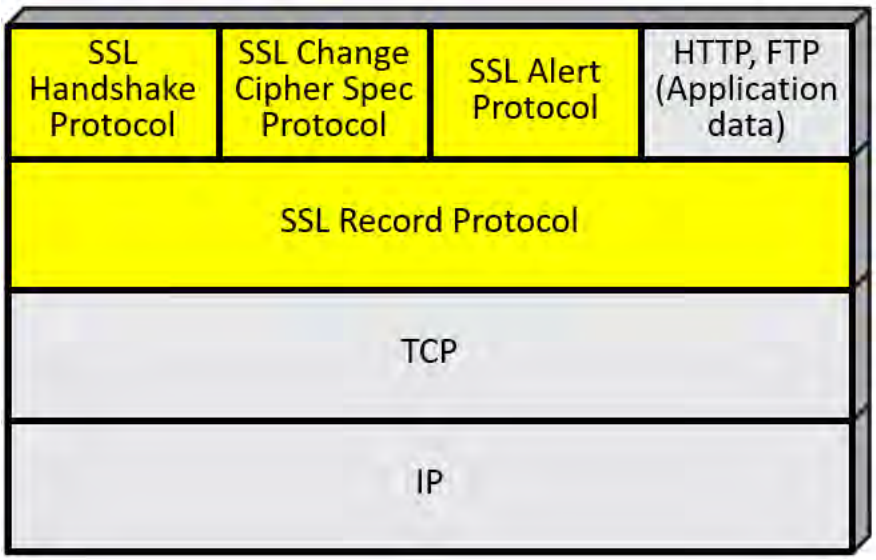


Рисунок 6.3. Архітектура протоколу SSL/TLS

Протокол запису SSL (SSL Record Protocol) розташований на сеансовому рівні мережевої моделі OSI та забезпечує базовий набір засобів захисту (конфіденційність та цілісність) для протоколів вищих рівнів (Application data). Цей протокол працює після того, як буде сформовано SSL-сесію. Після цього він може захищати TCP-трафік прикладних протоколів.

Частиною SSL вважаються і три протоколи вищого рівня:

• ***Протокол квитування (Handshake Protocol)*** – цей протокол використовується для формування, закриття та зміни параметрів SSL-сесії.

• ***Протокол сповіщення (Alert Protocol)*** – використовується для передачі іншій стороні службових повідомлень (зазвичай, про помилках).

• ***Протокол зміни параметрів шифрування (Change Cipher Spec Protocol)*** – допоміжний протокол для формування та зміни параметрів SSL-сесії.

6.3.2 Протокол запису SSL

Протокол запису SSL (SSL Record Protocol) забезпечує підтримку наступних двох сервісів для з'єднань SSL.

• ***Конфіденційність.*** Забезпечується симетричним шифрування даних, що передаються протоколом SSL. Загальний для клієнта та сервера секретний ключ, який використовуватиметься алгоритмом традиційної схеми шифрування, формує протокол квитування SSL.

• ***Цілісність повідомлень***. Забезпечується обчисленням значення MAC (Message Authentication Code – код автентичності повідомлення), яке передається протоколом запису SSL. Загальний секретний ключ для обчислення значень MAC також формує протокол квитування SSL.

На рис. 6.4 показано загальну схему роботи протоколу запису SSL.

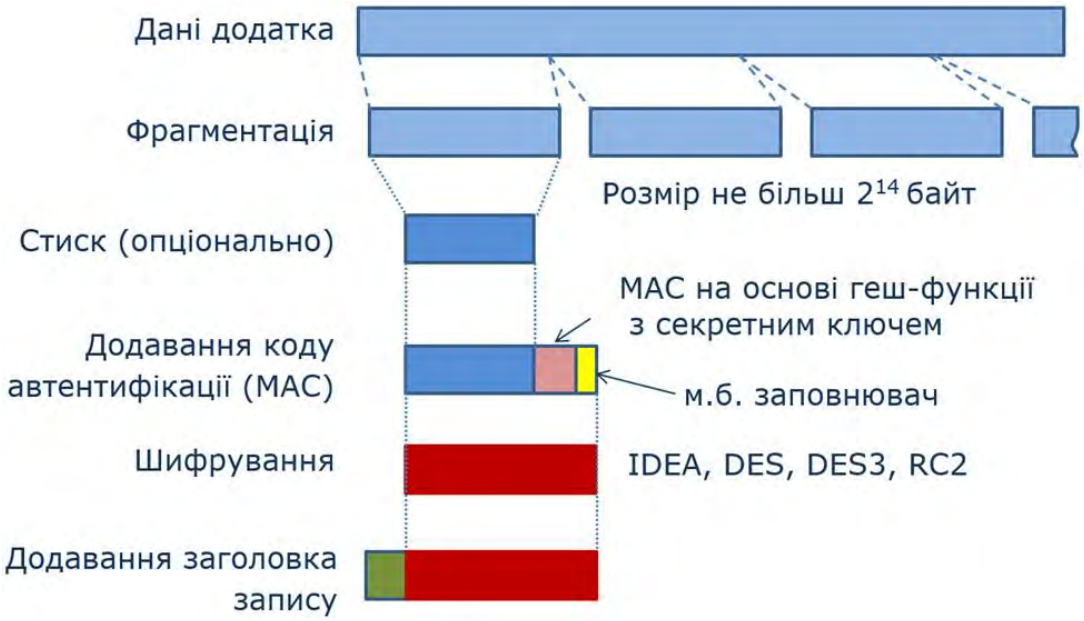


Рисунок 6.4. Схема роботи протоколу запису SSL

Першим кроком є фрагментація. Повідомлення, отримане від програми вищого рівня, поділяється на блоки розміром трохи більше 214 байт (16384 байт). Потім як необов'язкова можливість застосовується стискування. У специфікаціях SSLv3 (а також у версіях TLS) алгоритми стиснення не застосовується.

Наступним кроком є обчислення коду автентичності повідомлення (значення MAC) для блоку даних (можливо стисненого). Для цього є

загальний секретний ключ. Алгоритм обчислення MAC використовує геш-функцію повідомлення, що передається, і дуже схожий на алгоритм HMAC.

Потім повідомлення разом із доданим до нього значенням MAC шифрується з використанням симетричної схеми шифрування. У разі використання алгоритмів блочного шифрування після значення MAC може знадобитися додати заповнювач. При цьому безпосередньо за байтами заповнювача слідує 1-байтове значення, що вказує загальну довжину заповнювача. Для загальної довжини заповнювача вибирається найменше значення, при якому довжина блоку даних, що підлягають шифруванню (відкритий текст + MAC + заповнювач + 1), виявляється кратної довжини блоку шифру.

Завершальним кроком у роботі протоколу запису SSL є створення заголовка (рис. 6.5), що складається з полів що показані на рис. 6.5.

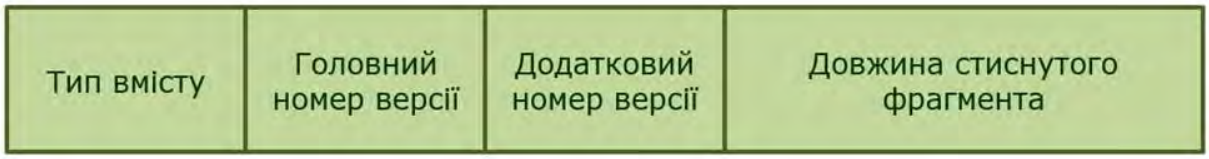


Рисунок 6.5. Формат заголовка запису, який формується протоколом запису SSL

• Тип вмісту (8 біт). Визначає протокол, що лежить вище рівня, за допомогою якого повинен оброблятися даний фрагмент. Для типу вмісту передбачено використання значень change_cipher_spec, alert, handshake та application_data. Перші три значення позначають протоколи стека SSL. Для інших протоколів прикладного рівня (HTTP, FTP, SMTP тощо) використовується значення application_data.

• Головний номер версії (8 біт). Вказує головний номер версії протоколу SSL. Для SSLv3 це поле містить 3.

• Додатковий номер версії (8 біт). Вказує додатковий номер версії протоколу SSL. Для SSLv3 це поле містить 0.

• Довжина стисненого фрагмента (16 біт). Довжина в байтах даного фрагмента відкритого тексту (або стисненого фрагмента, якщо використовується стиснення).

Використання протоколу запису для перенесення повідомлень протоколів SSL можливе лише за наявності створеної SSL-сесії. Якщо сесія ще не створена, всі повідомлення цих протоколів передаються незахищеним каналом зв'язку без використання протоколу запису.

6.3.3 Протокол зміни параметрів шифрування

Протокол зміни параметрів шифрування (Change Cipher Spec Protocol) є найпростішим із трьох протоколів вищого рівня у стеку протоколів SSL. Протокол зміни параметрів шифрування генерує однобайтове повідомлення, що містить значення 1 (рис. 6.6). Єдиною метою цього повідомлення є вказівка копіювати параметри стану очікування в поточний стан, у результаті оновлюється комплект шифрів, які використовуються у даному з'єднанні.

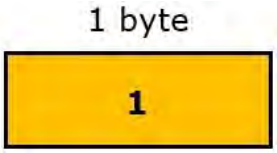


Рисунок 6.6. Формат повідомлення протоколу зміни параметрів шифрування

6.3.4 Протокол сповіщення

Протокол сповіщення (Alert Protocol) призначений для передачі сповіщень, що стосуються роботи SSL, іншій стороні, що бере участь в обміні даними.

Будь-яке повідомлення, яке генерується в рамках даного протоколу, складається з двох байтів (рис. 6.7).

Рисунок 6.7. Формат повідомлення протоколу сповіщення

Перший байт містить значення, що позначає відповідно рівень попередження (1) або рівень непереборної помилки (2). Якщо вказано рівень непереборної помилки, протокол SSL негайно розриває з'єднання. Інші з'єднання цього сеансу можуть продовжувати існувати, але встановити нове з'єднання для цього сеансу вже неможливо. Другий байт містить код, що означає конкретне значення повідомлення.

6.4 Протокол квитування

6.4.1 Призначення та загальна схема роботи протоколу квитування

Цей протокол дозволяє серверу та клієнту виконати взаємну автентифікацію, а також узгодити алгоритми шифрування, обчислення MAC та криптографічні ключі, які будуть застосовуватись для захисту даних, що пересилаються у записі SSL. Протокол квитування має використовуватися на початок пересилання даних прикладних програм.

У межах протоколу квитування генерується кілька повідомлень, якими обмінюються клієнт і сервер. Усі вони мають формат, показаний на рис. 6.8.

Будь-яке таке повідомлення містить три наступні поля.

• Тип (1 байт). Вказує один із 10 допустимих типів повідомлення. Допустимі типи повідомлень наведені в табл. 6.2.

• Довжина (3 байти). Довжина повідомлення в байтах.

• Вміст (≥ 0 байт). Параметри, пов'язані з повідомленням цього типу наведені в табл. 6.2.

На рис. 6.9 показана схема обміну повідомленнями під час встановлення логічного з'єднання між клієнтом та сервером.

Процес обміну складається з чотирьох основних етапів.

Рисунок 6.8. Формат повідомлення протоколу квитування

Таблиця 6.2

Типи повідомлень протоколу квитування

Тип повідомлення	Параметри
Hello_request	Ні
client_hello	Версія, випадкові значення, ідентифікатор сеансу, комплект шифрів, метод стиснення
server_he11o	Версія, випадкові значення, ідентифікатор сеансу, комплект шифрів, метод стиснення
Certificate	Ланцюжок сертифікатів X.509v3
server_key_exchange	Параметри, підпис
certificate_exchange	Тип сертифіката, назви уповноважених об'єктів
server_done	Ні
certificate_verify	Підпис
client_key_exchange	Параметри, підпис
finished	Геш-код

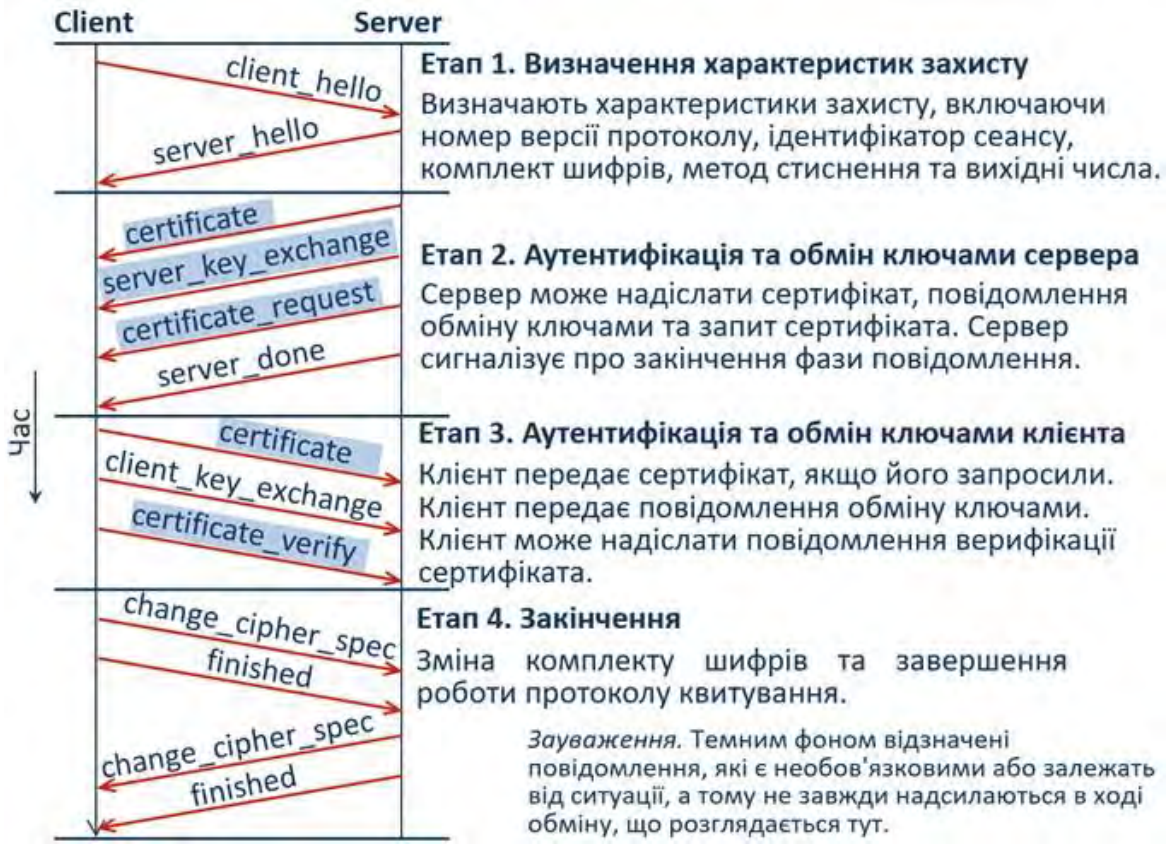


Рисунок 6.9. Схема роботи протоколу квитування

6.4.2 Етап 1 Визначення характеристик захисту

На цьому етапі наводиться ініціалізація логічного з'єднання і визначають пов'язані з ним характеристики захисту. Процес ініціюється клієнтом, який відправляє серверу повідомлення client_hello з запропонованими параметрами захисту та набори можливих (для клієнта) алгоритмів шифрування, геш-функцій, що використовуються для MAC, методів формування попереднього майстер-секрету тощо.

Після надсилання повідомлення client_hello (вітання клієнта) клієнт очікує від сервера повідомлення server_hello (вітання сервера), яке містить ті самі параметри, як і повідомлення client_hello, а з запропонованих клієнтів наборів вибирає конкретний алгоритм шифрування, конкретну геш-функцію тощо.

6.4.3 Етапи 2,3 Автентифікація та обмін ключами сервера, автентифікація та обмін ключами клієнта

Слід зазначити, що результатами роботи протоколу квітування має бути можлива автентифікація сервера та клієнта, а також формування головного ключа (майстер-ключа) та симетричних секретних сеансових ключів для шифрування та MAC. Автентифікація можлива за допомогою пред'явлення сертифікатів відкритих ключів. Для формування симетричних секретних ключів протокол квитування формує попередній майстер-секрет (секретний ключ) – pre_master_secret. Вищеназвані завдання етапів 2,3 можуть бути розв'язані різними методами, про які домовляються клієнт та сервер на етапі 1 Стандарт протоколу передбачає наступні методи автентифікації та обміну ключами.

• **RSA.** Секретне значення pre_master_secret генерується відправником (клієнтом) шифрується за допомогою відкритого ключа RSA одержувача (сервера). і передається серверу на 3 етапі в повідомленні klient_key_exchange. Для цього відправнику повинен бути доступний відкритий ключ одержувача. Цей ключ сервер посилає клієнту на 2 етапі або в повідомленні certificate, або в повідомленні server_key_exchange разом з підписами для параметрів ключа. В другому випадку в повідомленні cetificate сервер посилає клієнту сертифікат відкритого ключа для перевірки підпису.

• **Метод Діффі-Хеллмана з фіксованими параметрами (Fixed Diffie-Hellman).** В цьому методі на 2 етапі сервер передає клієнту параметри відкритого ключа алгоритму Діффі-Хеллмана в повідомленні certificate (параметри ключа фіксовані на період дійсності сертифікату). Клієнт повідомляє свої параметри відкритого ключа алгоритму Діффі-Хеллмана на 3 етапі або в сертифікаті (повідомлення certificate), якщо потрібна автентифікація клієнта (на 2 етапі сервер надав запит certificate_request), або в повідомленні обміну ключами client_key_exchange (в цьому випадку клієнт не автентифікується). Після обміну відкритими ключами кожна сторона виконує певні обчислення за методом Діффі-Хеллмана, в результаті яких виходить значення pre_master_secret, що використовується спільно.

• **Метод Діффі-Хеллмана з одноразовими параметрами (Ephemeral Diffie-Hellman).** Цей метод застосовується для створення одноразових відкритих ключів Діффі-Хеллмана. У цьому випадку сторони обмінюються відкритими ключами Діффі-Хеллмана, створеними лише для цього сеансу (в повідомленнях server_key_exchange та client_key_exchange). Автентифікація сервера в цьому випадку реалізується підписом свого відкритого ключа Діффі-Хеллмана і відправкою сертифіката відкритого ключа підпису в повідомленні certificate. Автентифікація клієнта, за необхідністю (сервер надіслав в 2 етапі повідомлення certificate_request), виконується підписом особового посилання, куди, зокрема, входить і вже сформований попередній майстер-секрет (повідомлення certificate_verify) і відповідного повідомлення клієнта certificate з сертифікатом відкритого ключа для перевірки цього підпису. Цей метод є найбезпечнішим із трьох зазначених тут варіантів методу Діффі-Хеллмана, оскільки відкриті параметри Діффі-Хеллмана використовуються лише один раз.

• **Анонімний метод Діффі-Хеллмана (Anonymous Diffie-Hellman).** Метод передбачає використання базового алгоритму Діффі-Хеллмана, але автентифікація не виконується. Іншими словами, кожна зі сторін надсилає свої відкриті параметри для алгоритму Діффі-Хеллмана іншій стороні в повідомленнях server_key_exchange та client_key_exchange без автентифікації. Цей підхід виявляється вразливим до атак «впровадження посередника», коли з обома сторонами за анонімним методом Діффі-Хеллмана обмін ключами проводить супротивник.

Наприкінці 2 етапу будь-якого з перерахованих вище методів сервер надсилає клієнту повідомлення server_done, яке означає кінець серії повідомлень сервера на цьому етапі і початок 3 етапу.

Після третього етапу відбувається формування криптографічних ключів (рис. 6.10).

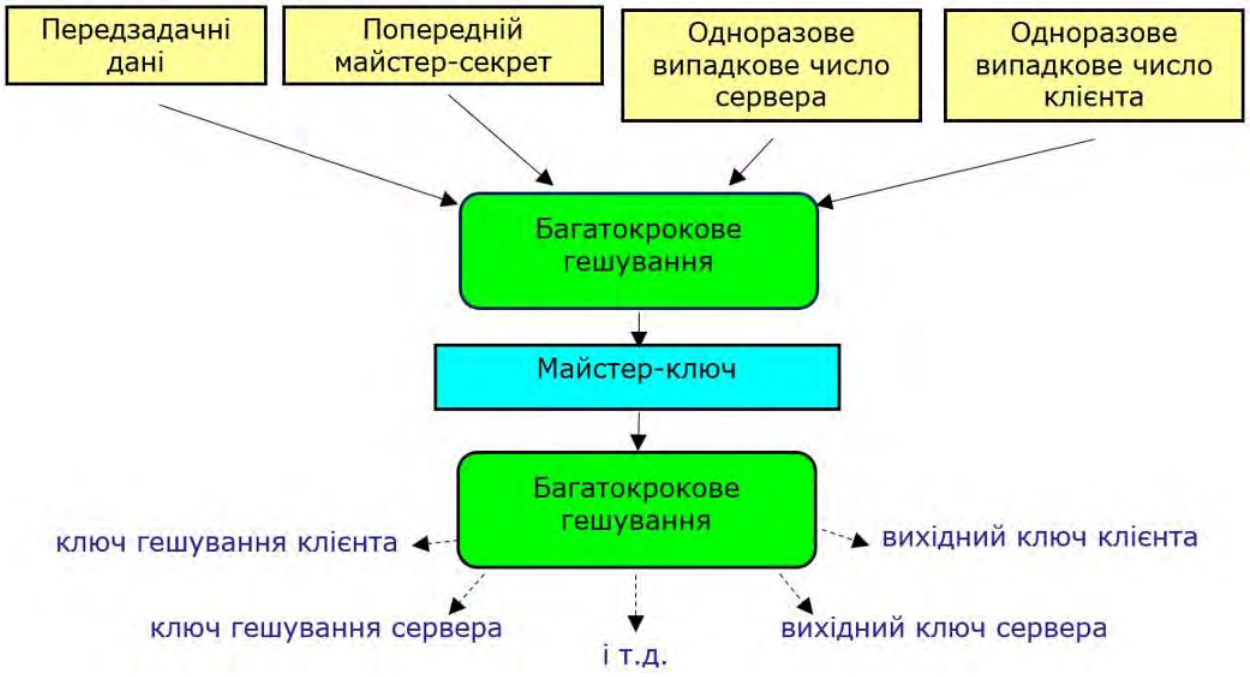


Рисунок 6.10. Процедура формування майстер-ключа та сеансових ключів

Спочатку за допомогою багатокрокового гешування формується головний секретний ключ (майстер-ключ). Крім попереднього майстер-секрету для формування головного майстер-ключа використовуються передзадачні дані і одноразові випадкові числа, якими клієнт і сервер обмінялися на першому етапі.

Потім формуються секретні сеансові ключі, які потрібні для роботи протоколу запису SSL: MAC-ключ клієнта для запису, MAC-ключ сервера для запису, ключ шифрування клієнта для запису (мається на увазі ключ, яким клієнт зашифровує дані, а сервер розшифровує), ключ шифрування сервера для запису, вектор ініціалізації клієнта для запису та вектор ініціалізації сервера для запису. Всі ці параметри генеруються з головного ключа за допомогою застосування функції гешування головного ключа для отримання захищеної послідовності байтів достатньої довжини.

Усі сформовані значення кожна сторона містить у полі CipherSpec стану очікування.

6.4.4 Етап 4 Завершення

Цей етап завершує створення захищеного з'єднання. Клієнт відправляє повідомлення change_cipher_spec (зміна параметрів шифрування) та копіює сформовані параметри (секретні ключі та вектора ініціалізації) з поля CipherSpec стану очікування у полі CipherSpec поточного стану. Зверніть увагу на те, що повідомлення change_cipher_spec не вважається частиною протоколу квитування, а надсилається в рамках протоколу зміни параметрів шифрування (Change Cipher Spec Protocol). Потім клієнт відправляє повідомлення finished, яке представляє собою геш код, сформований з усіх повідомлень квитування (за винятком цього повідомлення), шифроване новим алгоритмом з новими ключами. Повідомлення finished підтверджує, що процеси обміну ключами та автентифікації завершилися успішно.

У відповідь на ці два повідомлення сервер посилає своє повідомлення change_cipher_spec, переводить параметри CipherSpec стану очікування в поточний стан, розшифровує новим ключем та перевіряє правильність отриманого повідомлення finished (тотожність попередніх повідомлень квитування клієнта і сервера) і посилає своє повідомлення finished. Якщо перевірка клієнтом цього повідомлення буде успішна, то на цьому процес квитування завершується і тепер клієнт та сервер можуть розпочати обмін даними на прикладному рівні крізь нове SSL-з'єднання (за допомогою протоколу запису).

6.5 Особливості версії TLS 1.3

6.5.1 Загальні відомості щодо версії TLS 1.3

Версія TLS 1.3 не тільки несумісна з попередніми, а й побудована на іншій інженерній ідеології. Відмінність обумовлена переосмисленням моделі

загроз, у рамках якої проектується новий протокол: TLS спробували зробити і швидшим (принаймні потенційно), і більш захищеним, і більш потайливим. Що стосується останнього аспекту, то йдеться про зовсім новий для TLS напрямок: зведення до мінімуму витоків так званої метаінформації. Під метаінформацією розуміється сукупність таких відомостей про TLS-з'єднання, які дозволяють побічно судити про дані, що передаються в захищеному режимі. Наприклад, до метаінформації відносяться відомості про відкриті криптографічні ключі вузлів, час з'єднання, адреси, імена вузлів і так далі.

6.5.2 Початкове з'єднання (Handshake) версії TLS 1.3

У TLS 1.3 від попередніх версій успадковано лише базові принципи встановлення з'єднання: збереглися ролі вузлів (з'єднання ініціює клієнт), не змінилася послідовність ClientHello – ServerHello, є повідомлення-сигнал Finished. Однак на цьому схожість закінчується. У TLS 1.3 скорочено і загальну кількість повідомлень Handshake, і кількість повідомлень, що передаються у відкритому вигляді: вузли практично відразу переходять на зашифрований обмін. Тому зі схеми встановлення з'єднання видалено повідомлення ChangeCipherSpec.

Схема роботи Handshake TLS 1.3 представлена на рис. 6.11.

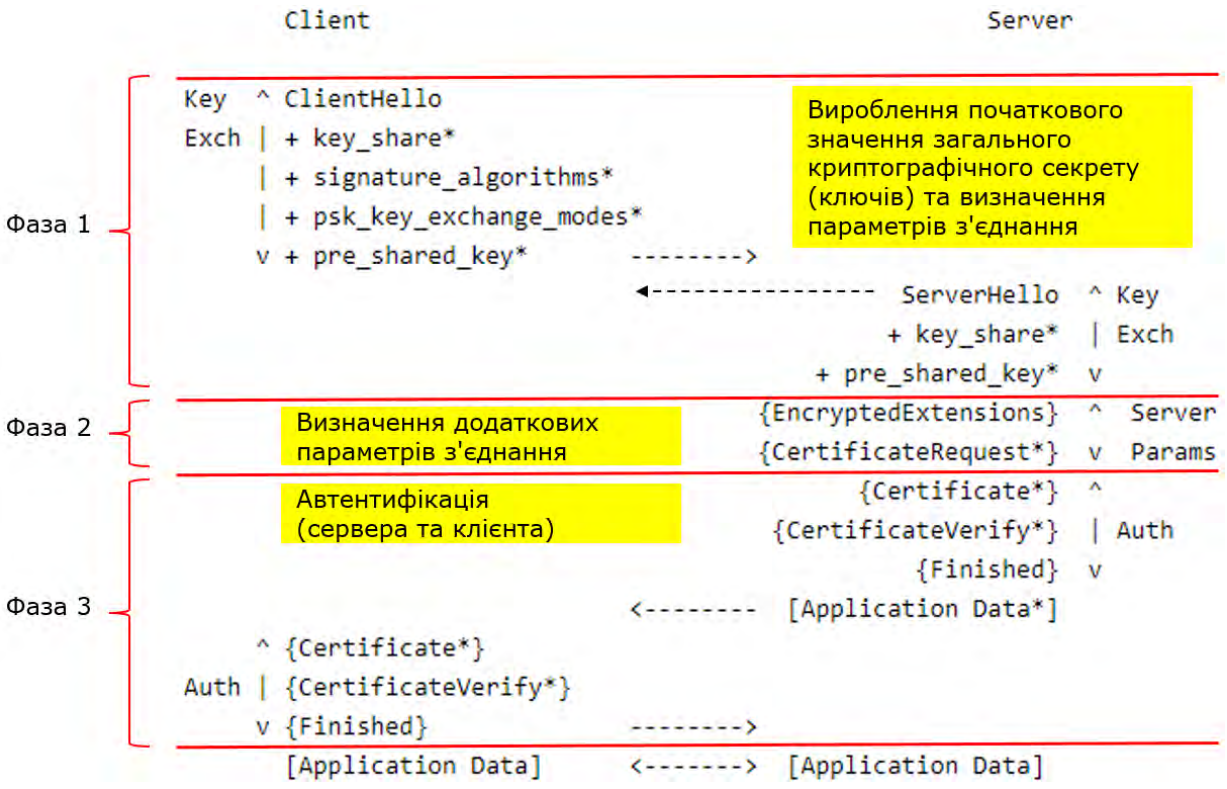


Рисунок 6.11. Схема роботи протоколу Handshake у TLS 1.3

Друга фаза включає серверні повідомлення EncryptedExtensions (нове повідомлення – додано до TLS 1.3, яке включає додаткові параметри для вироблення ключів шифрування) та, при необхідності автентифікації клієнта, CertificateRequest. Зверніть увагу: вже перша відповідь сервера містить

зашифровані дані, тоді як у TLS попередніх версій практично всі суттєві Handshake-повідомлення передаються у відкритому вигляді. Для шифрування тут використовуються секретні ключі, які сформовані на цей час, наприклад за алгоритмом Діффі-Хеллмана (з використанням еліптичних кривих), відкриті ключі якого передалися в розширеннях key_share.

У третій фазі, представленій групами повідомлень Certificate, CertificateVerify та Finished, відбувається автентифікація сервера та клієнта. Як і в попередніх версіях TLS, автентифікація може бути повністю виключена (анонімний режим), проте типовий сценарій використання передбачає принаймні автентифікацію сервера клієнтом. При цьому в TLS 1.3 сервер може перейти до відправлення даних корисного навантаження (Application Data на схемі) безпосередньо після серверного Finished. Це можливо тому, що до цього моменту сервер вже отримав перший набір симетричних ключів для шифрування повідомлень. Механізм, що дозволяє надсилати дані сервера та клієнта всередині фаз роботи протоколу handshake заощаджує час, необхідний для доставки пакетів від клієнта до сервера та назад.

При проектуванні TLS 1.3 велика увага приділялася зниженню втрат часу у роботі протоколу. Основний внесок у затримку за часом робить встановлення з'єднання. Тому передбачена скорочена схема, яка має умовну назву 0-RTT (Zero Round-Trip Time – нульова затримка прийому-передачі).

6.5.3 Керування сеансовими ключами у TLS 1.3

У TLS 1.3 використовується багатоетапний підхід до керування симетричними ключами. Його логіка будується на наступних реченнях:

- кожна фаза протоколу має використовувати свій набір ключів;
- значення ключів повинні залежати не тільки від загального секрету та параметрів Random, а й від інших властивостей конкретної сесії;
- кожен наступний набір ключів залежить від попередньої та додаткової інформації.

Для секретних ключів шифрування даних на першій та другій фазі протоколу (аналогічно попереднім версіям SSL/TLS) формується так званий майстер-секрет. Основу для майстер-секрету становить значення, отримане в рамках обміну алгоритму Діффі-Хеллмана, також (на додаток до секретів алгоритму Діффі-Хеллмана) може використовуватися секретне значення, яке вузли узгодили якимось способом раніше.

Сам алгоритм обчислення симетричних ключів та векторів ініціалізації на основі майстер-секрету у TLS 1.3 відрізняється від попередніх версій. Для отримання потрібної «порції бітів» секретних ключів використовується функція, яка ґрунтується на механізмі HKDF (HMAC Key Derivation Function).

6.5.4 Сумісність TLS 1.3 із ранніми версіями протоколів SSL/TLS

У TLS 1.3 з'явився новий механізм розпізнавання використовуваної версії протоколу. Поля, що містили номер версії в попередніх версіях, збережені у

статусі «історичних», їх значення зафіксовано. Версія ж 1.3 (її номер: 0x0304) передається у спеціальному розширенні першої фази. Дане розширення називається supported_versions (не показано на рис. 9.11), наявність якого є однією з ознак того, що клієнт (або сервер) будуть використовувати версію TLS 1.3 «Старі» поля повідомлень з номерами версій залишилися на своїх місцях, але містять значення, які не стосуються TLS 1.3 Розширення supported_versions у повідомленні ClientHello (якщо це версія TLS 1.3) включає список версій протоколу, які готовий підтримувати клієнт. Сервер передає у цьому розширенні номер вибраної версії. Цей механізм дозволяє зберегти можливість використання попередніх версій протоколу.

6.6 Запитання до розділу

1) Які функції захисту здатний забезпечити протокол SSL/TLS?
2) Яким чином протокол SSL/TLS забезпечує цілісність інформаційного потоку?
3) Які протоколи входять до складу SSL/TLS?
4) Які функції захисту забезпечує протокол запису SSL/TLS?
5) Що таке SSL-сеанс, SSL-з’єднання?
6) Які функції виконує протокол квитування SSL?
7) За рахунок чого реалізовано підвищення рівня захисту протоколу TLS 1.3 порівняно з попередніми версіями цього протоколу?

РОЗДІЛ 7 ЗАХИЩЕНІ ВІРТУАЛЬНІ МЕРЕЖІ (VPN) КАНАЛЬНОГО РІВНЯ

7.1 Основні принципи побудови VPN на канальному рівні

Протоколи VPN канального рівня використовують для своєї роботи принцип тунелювання.

Тунелювання – це нестандартний (що відрізняється від прийнятого в моделі OSI порядку) спосіб інкапсуляції пакетів деякого протоколу двох мереж, що з'єднуються, або вузлів в пакети протоколу транзитної мережі на її кордоні і передача пакетів мереж, що з'єднуються, через транзитну мережу. Тунелювання застосовується в тих випадках, коли транзитна мережа або не підтримує протокол мереж, що з'єднуються, або прагне ізолювати транзитну мережу від мереж, що з'єднуються.

За допомогою методики тунелювання пакети даних транслюються через загальнодоступну мережу як за звичайним двоточковим з'єднанням. Між кожною парою «відправник-одержувач даних» встановлюється своєрідний тунель – логічне з'єднання, що дозволяє інкапсулювати дані одного протоколу в пакети іншого. Крім того, тунелі дозволяють безпосередньо взаємодіяти між собою мережам із внутрішніми приватними адресами.

У процесі тунелювання формується віртуальне з'єднання poin-to-point.

Тунелювання використовує такі три типи протоколів:

• протокол «пасажир», який інкапсулюється в тунель (інкапсульований протокол), наприклад, IPv4, IPv6, IPX, ATM;

• протокол інкапсуляції – протокол, який виконує інкапсуляцію; найчастіше як протокол інкапсуляції використовується протокол GRE (Generic Routing Encapsulation – загальна інкапсуляція маршрутів);

• транспортний протокол (протокол «перевізник»), – протокол, який використовується для перенесення порції даних разом із заголовком інкапсульованого протоколу. Основний транспортний протокол – це IP, але як перевізник, можуть використовуватися і інші протоколи, як це показано в прикладі на рис. 7.1.

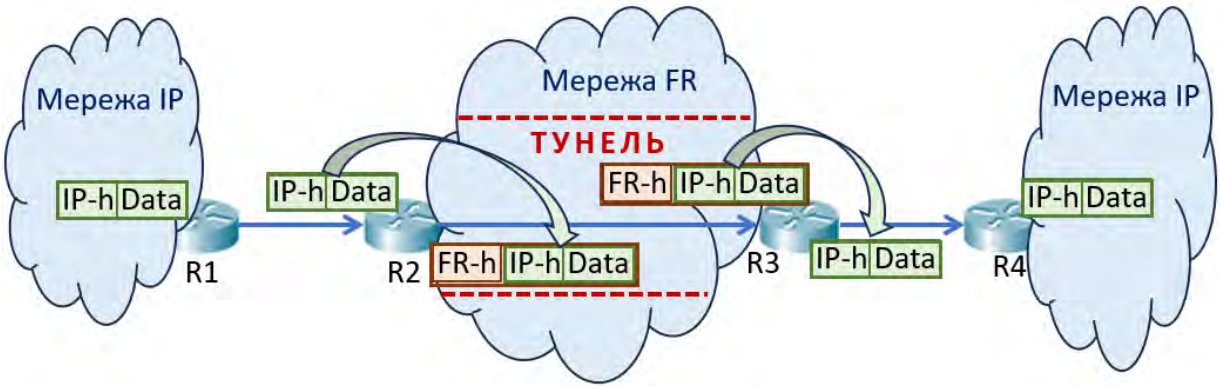


Рисунок 7.1. Тунелювання IP-трафіку через мережу Frame Relay (FR), яка не підтримує технологію IP

Тут протоколом «пасажиром» є протокол IP. Кадри цього протоколу надходять на маршрутизатор R2, програмне забезпечення якого включає протокол інкапсуляції. Цей протокол поміщає кадр IP, що надійшов, у поле даних кадра Frame relay (FR). Саме протокол FR і є у цьому прикладі транспортним протоколом. Кадр FR з «пасажиром» (пакетом IP у полі даних) передається на маршрутизатор R3 Протокол інкапсуляції – частина програмного забезпечення цього маршрутизатора – витягує пакет IP з поля даних FR-кадра і направляє цей пакет у мережу IP.

Зазвичай до VPN-протоколів канального рівня відносять протоколи PPTP (Point-to-Point Tunneling Protocol), L2F (Layer-2 Forwarding) та L2TP (Layer-2 Tunneling Protocol). Однак визначенню «VPN-протокол» точно відповідає тільки протокол PPTP, який забезпечує тунелювання і шифрування даних, що передаються. Протоколи L2F та L2TP підтримують лише функції тунелювання (і, опційно, функцію автентифікації). Для захисту даних, що тунелюються, в цих протоколах необхідно використовувати деякий додатковий протокол, зокрема IPSec.

Протокол PPTP розроблено компанією Microsoft, а протокол L2F – розробка компанії Cisco. Надалі ці компанії підписали угоду про створення на підставі PPTP та L2F нового протоколу – L2TP. В даний час реально використовується протоколи PPTP та L2TP, тому надалі розглядаються лише ці два протоколи.

Протоколи PPTP та L2TP ґрунтуються на протоколі канального рівня PPP (Point-to-Point Protocol) та є його розширеннями. Крім того, для своєї роботи протоколи PPTP та L2TP використовують протоколи транспортного рівня TCP (протокол PPTP) та UDP (протокол L2TP).

Схеми організації криптотунелю обох протоколів практично однакові. На рис. 7.2 показано можливу схему криптотунелю з використанням цих протоколів.

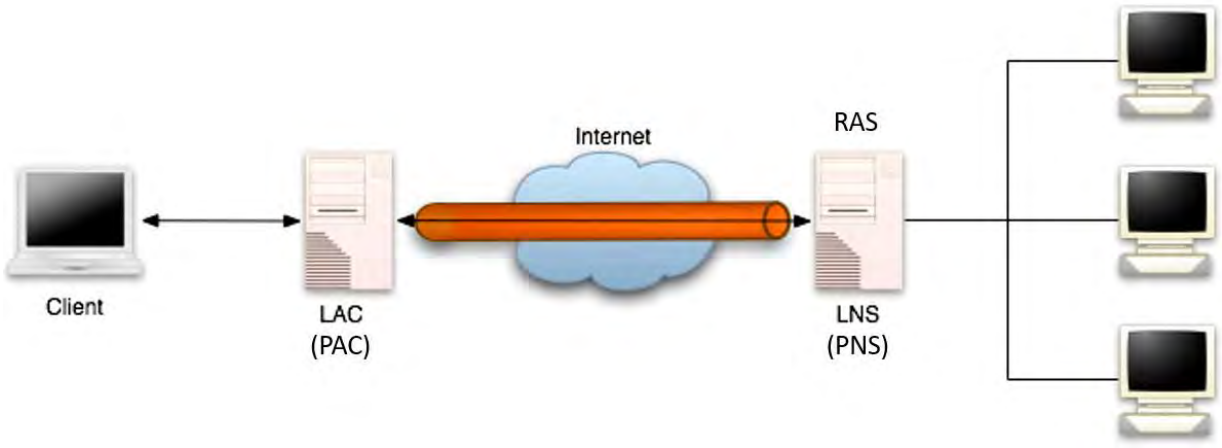


Рисунок 7.2. Можлива схема криптотунелю з використанням протоколів PPTP та L2TP

Криптотунель організується між двома вузлами. У документації до протоколів один із вузлів називається PPTP Access Concentrator (PAC) – концентратор доступу протоколу PPTP чи L2TP Access Concentrator (LAC)

концентратор доступу протоколу L2TP. Інший вузол називається PPTP Network Server (PNS) – мережевий сервер PPTP або L2TP Network Server (LNS) – мережевий сервер L2TP. В даний час і PNS і LNS реалізовані у вигляді віддаленого сервера доступу RAS (Remoute Access Server). Що стосується PAC та LAC, то найчастіше вони реалізовані у вигляді клієнтської частини RAS. І PAC і LAC можуть бути як окремим хостом, так і спеціалізованим пристроєм, до якого підключається група інших пристроїв (звідси назва – концентратор).

7.2 Протокол PPP – базовий протокол для побудови VPN на канальному рівні

7.2.1 Загальні відомості щодо протоколу PPP

PPP (протокол «точка-точка») є широко використовуваним методом транспортування IP-пакетів між користувачем та постачальником інтернет-послуг (ISP) по одній двоточковій лінії зв'язку шляхом використання інкапсуляції. Інкапсуляція в цьому випадку – це процес розміщення пакетів із протоколу мережевого рівня всередині кадрів PPP.

Крім того, двоточковий протокол PPP використовується в багатьох глобальних мережах, що базуються на послідовних каналах передачі даних. При цьому кадр PPP може ікапсулювати не тільки IP-пакети, але й PDU інших мережевих технологій (FR, ATM та ін.). Протокол PPP можна використовувати в різних фізичних середовищах передачі даних, включаючи кручені пари, оптоволоконні лінії та супутникові передачі, а також віртуальні з'єднання.

PPP використовує багаторівневу архітектуру. Для відповідності вимогам різних типів середовищ передачі двоточковий протокол встановлює між двома вузлами логічні з'єднання, які називаються сеансами. Сеанс PPP приховує фізичні середовища передачі даних від самого протоколу PPP і протоколів вищих рівнів. Також ці сеанси надають протоколу PPP метод для інкапсуляції кількох протоколів каналами типу «точка-точка». Кожен протокол, інкапсульований каналом, встановлює власний сеанс PPP.

Для передачі даних та формування сеансів PPP включає ряд протоколів:

• Протокол управління каналом високого рівня (HDLC) як основу для інкапсуляції дейтаграми в каналах «точка-точка».

• Протокол керування зв'язком (Link Control Protocol – LCP) використовується для встановлення, налаштування та тестування з'єднання з каналом передачі даних.

• Набір протоколів мережного керування (Network Control Protocol – NCP), які використовуються для встановлення та налаштування різних протоколів мережного рівня. Наприклад, протокол управління IP (IP Control Protocol – IPCP), що є різновидом NCP, застосовується для узгодження різних IP-параметрів, таких як IP-адреси, стиснення даних тощо.

Формат кадру PPP показано на рис. 7.3.

На цьому рисунку:

Рисунок 7.3. Формат кадру PPP

• **Прапор –** один байт, який вказує на початок або кінець кадру. Поле прапора складається із двійкової послідовності 011111110. Унікальність прапора гарантується використанням бітстаффінгу в синхронних з'єднаннях та байтстаффінгу в асинхронних. Бітстаффінг – вставка бітів, в кадрі PPP – біт 0 після п’яти поспіль бітів 1 Бітстаффінг (рис. 7.4) працює тільки під час передачі інформаційного поля (поля даних) кадру. Якщо передавач виявляє, що передано підряд п'ять одиниць, то він автоматично вставляє додатковий нуль у послідовність бітів, що передаються (навіть якщо після цих п'яти одиниць і так йде нуль). Тому послідовність 01111110 ніколи не з'явиться у полі даних кадру. Аналогічна схема працює у приймачі і виконує зворотну функцію. Коли після п'яти одиниць виявляється нуль, він автоматично видаляється із поля даних кадру.

Рисунок 7.4. Використання бітстаффінгу в кадрі PPP

• **Адреса –** один байт містить двійкову послідовність 11111111 – стандартну широкомовну адресу. PPP не призначає адреси окремих станцій.

• **Управління –** один байт, який має двійкову послідовність 00000011.

• **Протокол –** два байти, які розпізнають протокол, інкапсульований у поле даних кадру. Наприклад, код 0021 (шістнадцятковий) визначає в полі даних IP-пакет, код C021 – LCP-пакет, а код 8021 – NCP-пакет.

• **Дані –** містять дейтаграму (включаючи її службові поля) для протоколу, вказаного у полі «Протокол». Максимальна довжина становить 1500 байтів.

• **CRC (Cyclic redundancy check) –** Циклічний надлишковий код виявлення помилок. Зазвичай 16 біт (2 байти). Окремі реалізації PPP можуть використовувати 32-бітну (4 байти) CRC для кращого виявлення помилок.

7.2.2 Фази роботи протоколу PPP

У процесі конфігурування PPP-з'єднання, його підтримки та завершення це з'єднання проходить через кілька стадій, показаних на рис. 7.5.

Немає зв'язку (немає фізичної сполуки)

З'єднання починається і завершується цією фазою. Коли зовнішня подія (наприклад, виявлення несучої або зміна налаштувань мережі адміністратором) ініціює фізичне з'єднання, PPP переходить у фазу встановлення зв’язку.

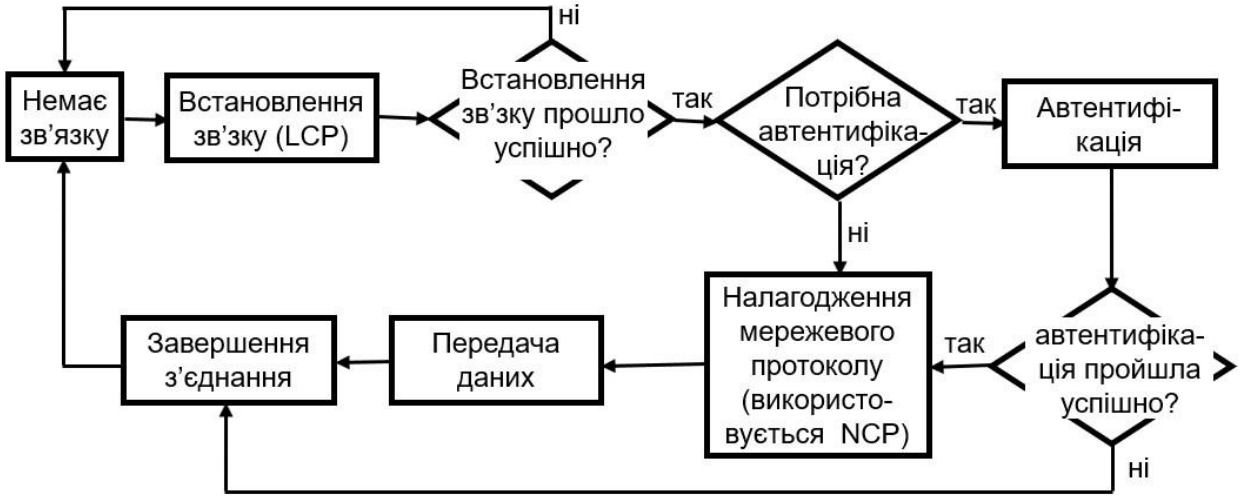


Рисунок 7.5. Спрощена фазова діаграма роботи протоколу PPP

Фаза встановлення зв'язку

Під час встановлення PPP-з'єднання спочатку використовується протокол керування з'єднанням (Link Control Protocol – LCP). За допомогою обміну конфігураційними пакетами в цьому протоколі визначаються функції налаштування та тестування з'єднання. З його допомогою узгоджуються такі, наприклад, процедури: чи виконуватиметься автентифікація і який із вбудованих протоколів (PAP, CHAP тощо) буде використовуватись для цієї мети. Після того як був здійснений обмін цими пакетами і був надісланий і прийнятий пакет підтвердження конфігурації, з'єднання вважається встановленим

Якщо налаштування було успішним, керування переходить у фазу аутентифікації або у фазу налаштування мережевого протоколу, залежно від того, чи потрібна аутентифікація.

Фаза автентифікації

Аутентифікація не є обов'язковою. Якщо в будь-якій реалізації необхідно, щоб сторони здійснювали перевірку за допомогою певного протоколу аутентифікації, такий протокол повинен бути запитаний у фазі встановлення з'єднання. Спосіб автентифікації залежить від реалізації. Надалі механізми аутентифікації цієї фази буде розглянуто докладніше.

Фаза протоколу мережевого рівня

Для узгодження певних варіантів налаштувань і параметрів PPP застосовує протокол управління мережею (Network Control Protocol – NCP), що використовується протоколом L_3 У цій фазі кожен протокол мережевого рівня (наприклад, IP, IPX або AppleTalk) налаштовується окремо за допомогою відповідного протоколу керування мережею з набору Network Control Protocol – NCP. Наприклад, протокол управління IP (IP Control

Protocol — IPCP), що є різновидом NCP, застосовується для узгодження різних IP-параметрів, таких як IP-адреси, стиснення даних тощо, включення модулів IP-протоколу на обох кінцях каналу PPP.

Фаза передачі даних

Після того, як NCP перейшов у стан встановленого (відкритого) з'єднання, PPP передаватиме відповідні пакети протоколу мережевого рівня. У цій фазі потік даних каналом зв'язку може складатися з пакетів протоколу мережевого рівня, а також пакетів як LCP, так і NCP.

Фаза завершення з'єднання

Для завершення зв'язку використовуються пакети LCP завершення зв'язку. Коли з'єднання закривається, PPP повідомляє про те протоколи мережевого рівня, щоб вони могли вжити відповідні дії. За винятком того випадку, коли зовнішня проблема викликала розрив зв'язку, з'єднання зазвичай завершується протоколом верхнього рівня або самим користувачем.

7.2.3 Приклад використання протоколу PPP

На рис. 7.6 наведено приклад того, як PPP може бути використаний для забезпечення операцій конфігурування мережі.

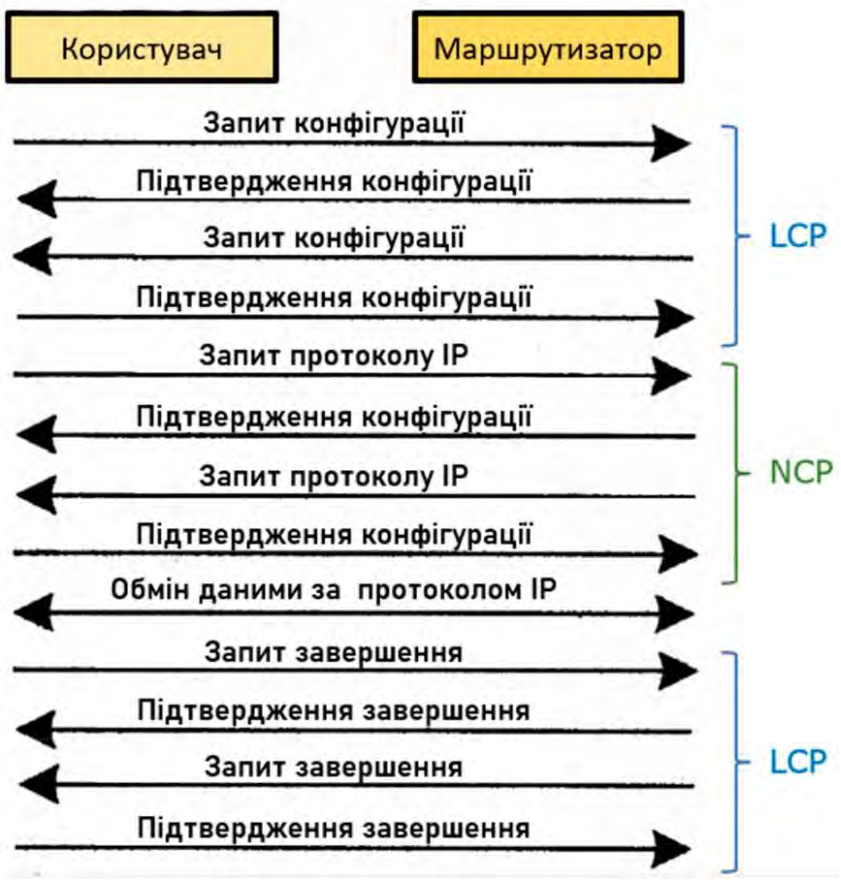


Рисунок 7.6. Приклад взаємодії протоколу PPP

Маршрутизатори, вузли тощо обмінюються PPP-фреймами визначення того, який протокол мережного рівня підтримується обома станціями. У

цьому прикладі два комп'ютери домовляються про використання IP. Спочатку використовуються методи протоколу LCP для встановлення та тестування з'єднання. Потім викликається один із протоколів NCP для узгодження параметрів мережевого протоколу (і деяких інших процедур, які будуть використовуватися між двома машинами). На рис. 7.6 показано роботу з протоколом IP, з набору NCP буде обрано протокол IPCP. Після завершення цього узгодження починається обмін даними. У будь-який час будь-який учасник з'єднання може розірвати канал зв'язку.

Протокол PPP має розширення, що дозволяють працювати поверх протоколів Ethernet (PPPoE) та ATM (PPPoA), шифрувати трафік.

7.3 Механізми автентифікації, що використовуються в протоколах PPP, PPTP та L2TP

7.3.1 Загальні відомості щодо протоколів автентифікації на канальному рівні. Протокол PAP (Password Authentication Protocol)

У початковій специфікації в PPP було визначено два протоколи автентифікації: Password Authentication Protocol (PAP) та Challenge Handshake Authentication Protocol (CHAP). Надалі цей список було розширено – MS CHAP різних версій, Extensible Authentication Protocol (EAP).

PAP – це проста процедура, що дозволяє одній стороні (зазвичай хосту чи маршрутизатору) засвідчити себе. Ця операція виконується після початкового етапу встановлення з'єднання. Як тільки фаза встановлення з'єднання завершена, пара ім'я/пароль багаторазово надсилається відповідальному за виконання авторизації вузлу, доки не підтверджується ідентифікація або розривається з'єднання.

PAP не розроблялася як надійна процедура автентифікації, тому всі паролі та імена надсилаються відкритим текстом. Вузли не мають жодних засобів проти моніторингу чи атак.

На рис. 7.7 показано формат пакета PAP у полі даних PPP.

Протокол	Інформація			
C023	Код	Ідентифікатор	Довжина	Дані

Рисунок 7.7. Формат пакета PAP у полі даних PPP

Поле протоколу PPP встановлюється рівним C023 Поле коду задає один із трьох видів пакетів: «Запит автентифікації», «Підтвердження автентифікації» та «Відмова автентифікації». Поле ідентифікатора (ID-пакета) використовується для зіставлення та координації повідомлень запитів з відповідними відповідями. Поле довжини задає довжину (в байтах) вмісту поля даних, виключаючи номер протоколу. Поле даних заповнюється ім'ям та паролем у пакеті «Запит автентифікації» та відповідним повідомленням у разі підтвердження чи відмови.

Послідовність обміну пакетами під час роботи протоколу PAP показано на рис. 7.8.

Рисунок 7.8. Послідовність обміну пакетами під час роботи протоколу PAP

У клієнт-серверному з'єднанні після успішного встановлення зв'язку клієнтська частина може вимагати введення логіну та пароля (якщо в процесі встановлення зв'язку партнери домовляться про автентифікацію з використанням проколу PAP). Після введення клієнтом логіну та паролю буде сформовано PAP-пакет «Запит автентифікації», у полі даних якого будуть розміщені введені клієнтом логін та пароль. Пакет «Запит автентифікації» надсилається багато разів, доки не буде отримано відповідь або (необов'язковий варіант) буде надіслано задану кількість пакетів.

Автентифікуюча сторона (сервер) очікує, що буде надіслано пакет «Запит автентифікації». Після отримання такого пакета повертається одна з можливих відповідей.

Якщо пара ім'я/пароль, що отримана в пакеті «Запит автентифікації» правильна, ідентифікуючий (сервер) посилає пакет «Підтвердження автентифікації». В іншому випадку він надсилає пакет «Відмова автентифікація». Тоді інша сторона вважається неавтентифікованою і обмін пакетами L_3 не може статися через те, що одержувач відмови повинен завершити зв'язок.

7.3.2 Протокол CHAP (Challenge Handshake Authentication Protocol)

Протокол автентифікації «Запит-підтвердження» (Challenge-Handshake Authentication Protocol – CHAP) є більш надійним, ніж PAP. Як і PAP, він

розроблений для роботи поверх PPP на лініях, що комутуються, між вузлом і, наприклад, маршрутизатором.

CHAP автентифікує вузол, що приєднується, за допомогою триетапного обміну повідомленнями в процесі початкової установки з'єднання. Крім того, пізніше він може бути викликаний у будь-який час. Для автентифікації протокол CHAP використовує одноразове випадкове число Random (у деяких джерелах його називають одноразовим відкритим ключем, оскільки воно пересилається у відкритому вигляді незахищеним каналом зв'язку) і загальний секрет S, який зберігається у кожної сторони і, звичайно ж, каналами зв'язку нікуди не передається.

У протоколі CHAP є чотири типи повідомлень: Виклик, Відповідь, Успіх та Невдача. Вони надсилаються у кадрах, форматів яких аналогічний формату пакета PAP, показаного на рис. 7.7 Повідомлення розміщені в полі «Дані» кадру CHAP. Номер протоколу CHAP дорівнює C223 Для CHAP ідентифікатор змінюється кожного разу після надсилання будь-якого повідомлення «Виклик» і поле ідентифікатора копіюється у відповідне повідомлення «Відповідь».

Після завершення фази установки каналу зв'язку PPP автентифікуючий (у прикладі – сервер) посилає іншій стороні (клієнту) повідомлення «Виклик» з кодом 1 (рис. 7.9). У цьому повідомленні в полі «Значення» розміщено випадкове одноразове значення Random, згенероване сервером.

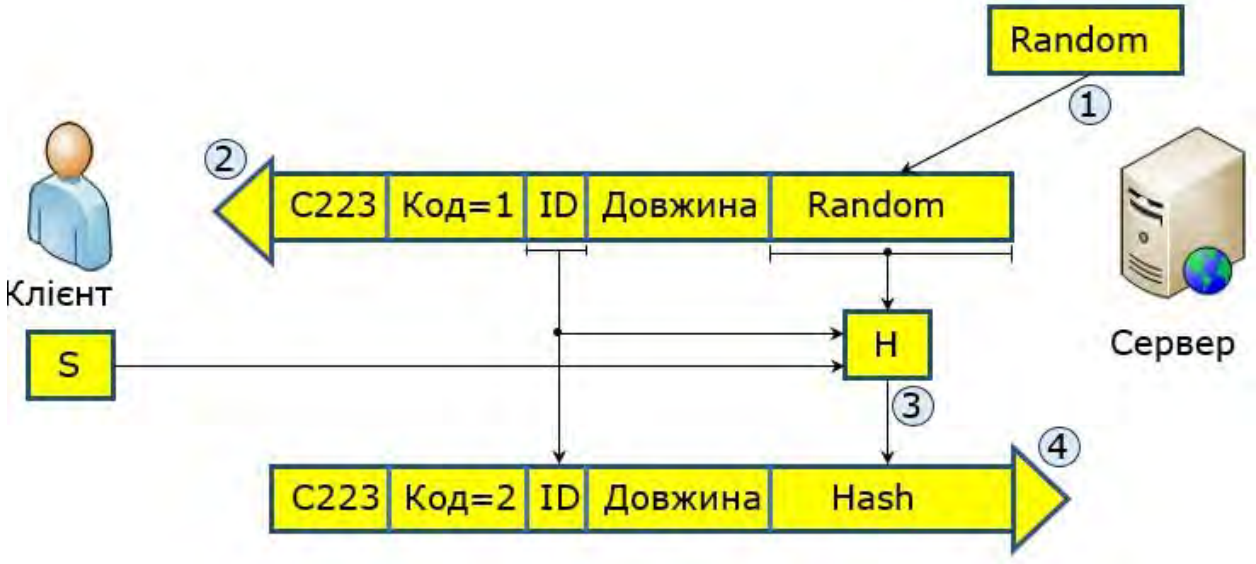


Рисунок 7.9. Клієнт, отримавши повідомлення «Виклик», обчислює хеш-функцію

Клієнт, отримавши повідомлення «Виклик» від сервера, повинен обчислити хеш-функцію. Аргументами хеш функції є:

• ідентифікатор (ID) повідомлення «Виклик» (значення ідентифікатора копіюється в полі ідентифікатора повідомлення у відповідь);

• одноразове випадкове значення Random із поля даних отриманого повідомлення;

• значення секрету S.

Результат хешування міститься в полі даних повідомлення у відповідь з кодом «Відповідь» (код = 2).

Ідентифікуючий (сервер) перевіряє отриману інформацію (рис. 7.10), провівши власні обчислення хешу. Збіг значень хеш-кодів доводить наявність секрету S у клієнта. У цьому випадку сервер підтверджує автентифікацію клієнта, надсилаючи клієнтові повідомлення «Успіх» (код = 3). В іншому випадку клієнту надсилається повідомлення «Невдача» (код = 4) і сервер ініціює розрив з'єднання.

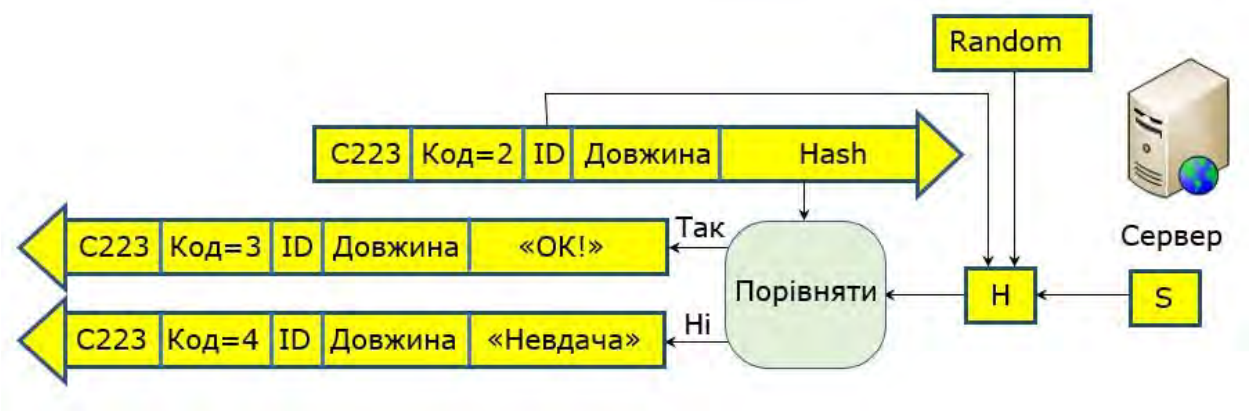


Рисунок 7.10. Дії сервера після отримання повідомлення «Відповідь» від клієнта

Автентифікація виконується в односторонньому режимі, але двостороннє поведінка PPP дозволяє CHAP діяти у двосторонньому режимі, тобто клієнт посилає серверу випадкове значення, сервер обчислює та пересилає клієнту хеш з використанням цього випадкового значення та загального секрету і вже клієнт перевіряє наявність загального секрету у сервера, порівнюючи значення надісланого та обчисленого ним значення хешу.

7.3.3 Протоколи MS CHAP та EAP

MS-CHAP являє собою механізм автентифікації, схожий на CHAP, але має важливу відмінність: у CHAP сервер повинен зберігати в зашифрованому вигляді пароль клієнта (секрет S), який розшифровується при кожній автентичності клієнта, а в MS-CHAP сервері для цього потрібно тільки хеш пароля. Зі знайдених проблем з безпекою MS-CHAP був перейменований на MS-CHAP v1 і потім замінений на MS-CHAP v2.

Розширений протокол автентифікації EAP (Extensible Authentication Protocol) – це платформа автентифікації, яка дозволяє використовувати різні методи автентифікації для технологій безпечного доступу до мережі. У протоколах PPTP та L2TP часто використовують наступні EAP.

EAP-Microsoft Challenge Handshake Authentication Protocol версії 2 (EAP-MSCHAP v2): визначений корпорацією Майкрософт метод EAP, який інкапсулює протокол автентифікації MSCHAP версії 2, який використовує ім'я користувача та пароль для автентифікації.

EAP-Transport Layer Security (EAP-TLS): метод EAP, який використовує TLS із сертифікатами для взаємної автентифікації. Методи EAP, які

використовують протокол EAP-TLS на основі сертифікатів, зазвичай забезпечують найвищий рівень безпеки.

7.4 Архітектура та принцип функціонування протоколу PPTP

7.4.1 Загальні відомості про протокол PPTP

PPTP, створений у 1999 році, є одним із найстаріших протоколів VPN і використовується досі. Розроблений Microsoft він використовувався у всіх версіях Windows з моменту створення Windows 95 Сьогодні майже всі пристрої, настільні та мобільні платформи підтримують PPTP.

PPTP найбільш популярний, тому що це найшвидший, найпоширеніший і найпростіший протокол VPN для налаштування.

Протокол PPTP дозволяє створювати захищені канали для обміну даними за різними мережевими протоколами – IP, IPX або NetBEUI. Дані цих протоколів інкапсулюються за допомогою протоколу PPTP у пакети протоколу IP, за допомогою якого переносяться у зашифрованому вигляді через мережу TCP/IP. Інкапсулюється початковий кадр PPP, тому протокол PPTP можна віднести до класу протоколів інкапсуляції канального рівня в мережевий.

Протокол PPTP включає дві працюючі паралельно компоненти (рис. 7.11) – керуюче з'єднання між кожною парою PAC – PNS, що працює за протоколом TCP, і тунель IP між тією ж парою пристроїв PAC – PNS, що служить для транспортування інкапсульованих за допомогою GRE пакетів PPP в сесіях користувача між парою пристроїв.

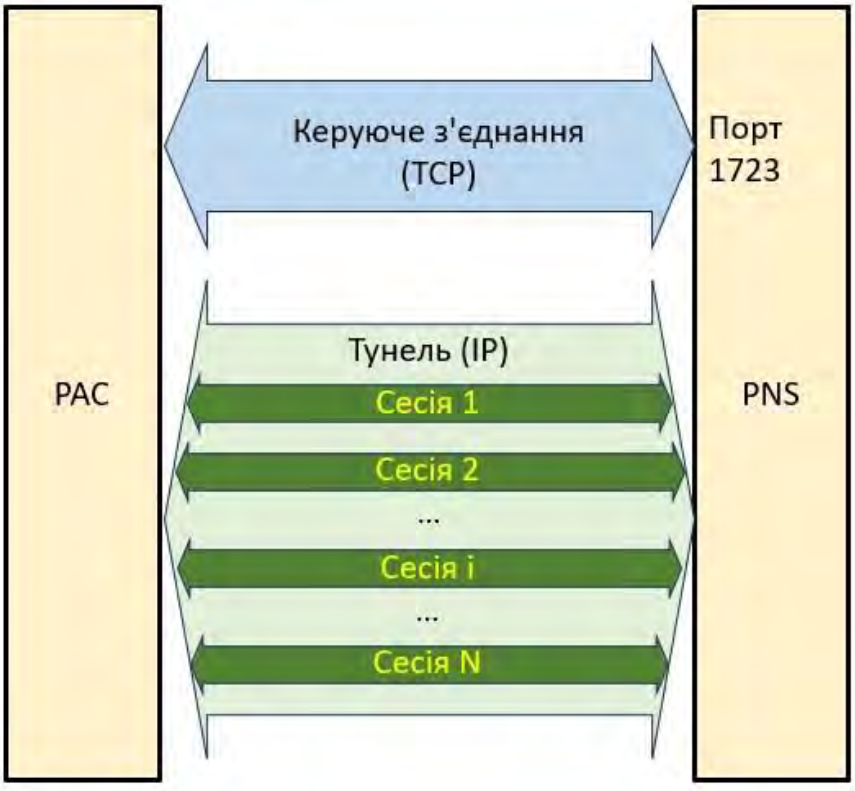


Рисунок 7.11. Компоненти протоколу PPTP, що працюють між PAC та PNS

7.4.2 Керуюче з'єднання протоколу PPTP

Для створення PPP-тунелю між PAC і PNS, потрібно спочатку організувати між цими пристроями керуюче з'єднання. Це з'єднання є звичайною сесією TCP, через яку передається інформація керування викликами та з'єднанням PPTP. Керуюче з'єднання відповідає за організацію, підтримку та завершення сесій, що організуються через тунель. Сесія TCP для керуючого з'єднання (рис. 7.11) організується ініціатором з'єднання TCP через порт 1723 Порт-джерело вибирається відправником зі складу вільних. Ініціатором з'єднання може бути як PAC, так і PNS.

PPTP визначає набір повідомлень, що передаються через керуюче з'єднання TCP між пристроями PNS та PAC (табл. 7.1).

Таблиця 7.1

Повідомлення, що надсилаються через керуюче з'єднання TCP між пристроями PNS та PAC

Керуюче повідомлення	Код
Підтримка керуючого з'єднання	
Start-Control-Connection-Request	1
Start-Control-Connection-Reply	2
Stop-Control-Connection- Request	3
Stop-Control-Connection- Reply	4
Echo-Request	5
Echo-Replay	6
Керування викликами	
Outgoing-Call-Request	7
Outgoing-Call-Replay	8
Incoming-Call-Request	9
Incoming-Call-Replay	10
Incoming-Call-Connected	11
Call-Clear-Request	12
Call-Disconnect-Notify	13
Повідомлення про помилки	
WAN-Error-Notify	14
Керування сесією PPP	
Set-Link-Info	15

Ініціатором організації керуючого з'єднання може бути PNS або PAC. Після організації необхідного для роботи з'єднання TCP пристроїв PNS і PAC організують між собою керуюче з'єднання, використовуючи повідомлення Start-Control-Connection-Request і Start-Control-Connection-Reply. Ці повідомлення також служать для обміну інформацією про базові можливості PAC і PNS.

Після організації керуючого з'єднання PAC або PNS може ініціювати сесії, запитуючи вихідні з'єднання або відповідаючи на вхідні (група повідомлень «Керування викликами»).

7.4.3 Організація тунелювання протоколом PPTP

PPTP вимагає організації тунелю кожної взаємодіючої пари PNS – PAC. Цей тунель служить для передачі кадрів PPP всіх сесій, що проходять через цю пару PNS – PAC. Номер сесії, до якої належить конкретний кадр PPP, вказується у відповідному полі в заголовку GRE. Такий спосіб забезпечує мультиплексування та демультиплексування пакетів PPP через тунель між парою пристроїв PNS – PAC.

Формування та структура кадрів, що передаються в рамках сесії PPTP, показано на рис. 7.12

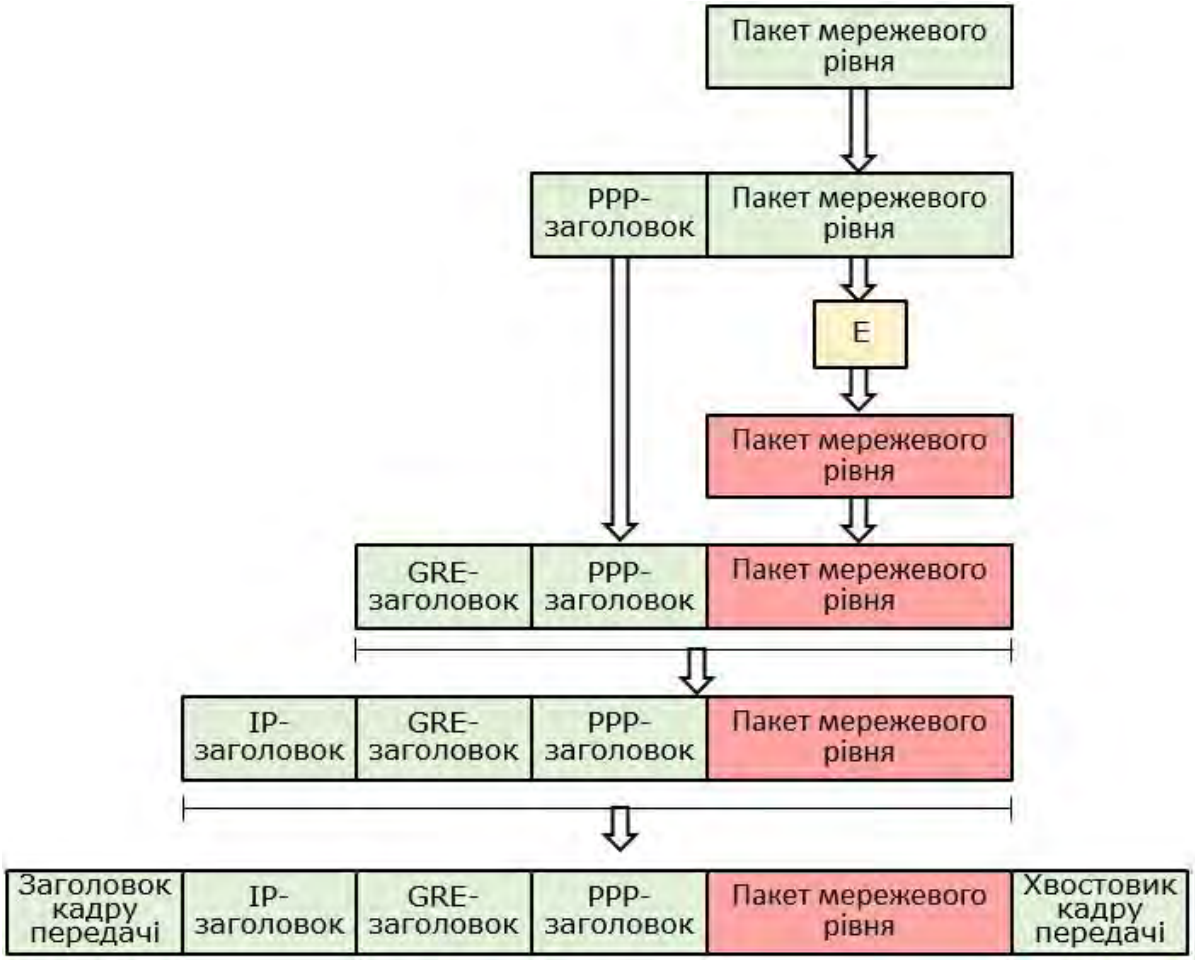


Рисунок 7.12. Формування та структура кадру для пересилання тунелем PPTP

Послідовність формування кадру для пересилання тунелем PPTP наступна. Пакет мережного рівня (протоколу IP, IPX тощо) міститься в PPP-кадр (зрозуміло, що попередньо PPP-з'єднання вже створено раніше за подобною ж схемою, але в PPP-кадр вкладається не пакет мережевого рівня, а повідомлення протоколів LCP, NCP, автентифікації – шифрування для цих PPP-кадрів не застосовується). Дані PPP-кадра (тобто пакет мережевого

рівня) шифруються, формується GRE-заголовок (заголовок загального методу інкапсуляції дня маршрутизації GRE – Generic Routing Encapsulation), в одному з полів якого вказується номер сесії PPTP. GRE-пакет міститься у полі даних новоствореного IP-пакета. У полі «протокол» IP-заголовка цього пакета вказується код 47 – ознака протоколу PPTP. Знову сформований IP-пакет міститься в полі даних кадру передачі, наприклад кадру Ethernet.

Вузол мережі, що приймає, витягує з пакетів IP кадри PPP, а потім витягує з кадру PPP зашифровані дані вихідного пакета мережевого рівня OSI, розшифровує їх і, якщо приймаючий вузол – сервер корпоративної мережі, відправляє ці дані по локальній мережі конкретному адресу. Багатопротокольність інкапсулюючих протоколів канального рівня, до яких відноситься протокол PPTP, є їх важливою перевагою перед протоколами захищеного каналу вищих рівнів. Наприклад, якщо в корпоративній мережі використовуються IPX або NetBEUI, застосування протоколів IPSec або SSL просто неможливо, оскільки вони орієнтовані лише на один протокол мережного рівня – IP. Такий спосіб інкапсуляції забезпечує незалежність від протоколів мережевого рівня моделі OSI та дозволяє здійснювати захищений віддалений доступ через відкриті IP-мережі до будь-яких локальних мереж.

7.4.4 Автентифікація та шифрування в протоколі PPTP

Згідно з протоколом PPTP при створенні захищеного віртуального каналу проводиться автентифікація віддаленого користувача і шифрування даних, що передаються. Для автентифікації віддаленого користувача можуть використовуватися різні протоколи, що застосовуються для PPP, такі, наприклад, як протокол розпізнавання за паролем PAP (Password Authentication Protocol), протокол розпізнавання при рукостисканні CHAP (Challenge-Handshaking Authentication Protocol) та MSCHAP (Microsoft Challenge -Handshaking Authentication Protocol) різних версій, протокол розпізнавання EAP-TLS (Extensible Authentication Protocol – Transport Layer Security). Для підвищення захищеності рекомендується використовувати сучасні протоколи автентифікації – MSCHAPv2 чи EAP-TLS.

Шифрування в PPTP гарантує, що ніхто не зможе отримати доступ до даних під час пересилання через Internet. Стандартний для PPTP протокол шифрування MPPE (Microsoft Point-to-Point Encryption) дозволяє автоматично вибирати довжину ключа шифрування за умови узгодження параметрів між клієнтом і сервером. MPPE використовує алгоритм потокового шифрування RC4, підтримує 40-, 56- та 128-бітові ключі, які змінюються протягом сесії, є можливість генерувати кожен пакет за новим ключем. Ключі шифрування генеруються в процесі автентифікації за протоколом MS-CHAP, MS-CHAPv2 або EAP-TLS.

7.4.5 Схеми застосування протоколу PPTP

Для протоколу PPTP визначено кілька схем застосування. Наприклад, у схемі тунелювання (рис. 7.13) віддалений користувач встановлює віддалене з'єднання з локальною мережею за допомогою клієнтської частини сервісу віддаленого доступу RAS (Remote Access Service), що входить до складу ОС Windows. Потім користувач звертається до сервера віддаленого доступу

локальної мережі, вказуючи його IP-адресу, і встановлює з ним зв'язок за протоколом PPTP.

Функції віддаленого сервера може виконувати прикордонний маршрутизатор локальної мережі або сервер провайдера цієї мережі. На комп'ютері видаленого користувача мають бути встановлені клієнтська частина сервісу RAS і драйвер PPTP, а на сервері віддаленого доступу — сервер RAS і серверна версія драйвера PPTP. Після створення криптозахищеного PPTP-тунелю починається інформаційний обмін між комп'ютером віддаленого доступу та хостами локальної мережі. Внутрішні хости локальної мережі можуть не підтримувати протокол PPTP, оскільки сервер віддаленого доступу витягує кадри PPP з пакеті IP, а вже з кадрів PPP витягує пакети мережевого рівня в необхідному форматі (див. рис. 7.12), які посилає в локальну мережу.

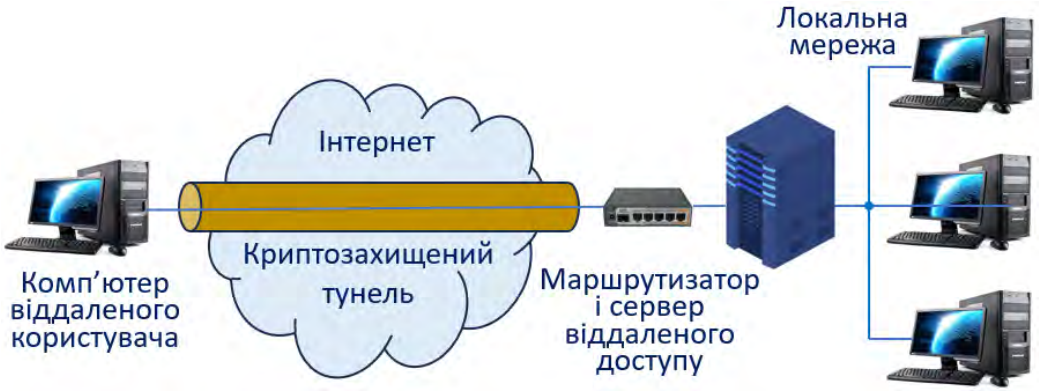


Рисунок 7.13. Схема тунелювання під час прямого підключення комп'ютера віддаленого користувача до Internet

7.4.6 Переваги протоколу PPTP

- PPTP підтримує більшість існуючих операційних систем (від комп'ютера до телефону). Це найважливіша перевага використання PPTP.
- PPTP — це найпростіший протокол VPN для налаштування навіть для тих, хто не має досвіду в налаштуванні VPN.
- PPTP – це найшвидший протокол VPN, який використовується на сьогоднішній день. Він відмінно підходить для завантаження, потокової передачі та загального використання.
- PPTP недорогий.

7.4.7 Недоліки протоколу PPTP

- Головна вразливість PPTP на сьогоднішній день полягає у слабкості алгоритмів парольної автентифікації (MSCHAP, MSCHAPv2), а також у тому, що при використанні цих алгоритмів сесійні ключі MPPE виходять з пароля користувача. Проте, якщо замість парольної автентифікації використовувати автентифікацію EAP-TLS, то стійкість автентифікації та ключів шифрування буде значно збільшено.

- Багато криптоаналітиків мають серйозні претензії до алгоритму шифрування MPPE.

7.5 Архітектура та принцип функціонування протоколу L2TP

7.5.1 Загальні відомості про протокол L2TP

Протокол L2TP (Layer-2 Tunneling Protocol) розроблений за підтримки компаній Microsoft і Cisco Systems на основі протоколів PPTP та L2F і в результаті увібрав у себе найкращі якості вихідних протоколів.

По суті, гібридний протокол L2TP є розширенням протоколу PPP функціями автентифікації віддалених користувачів, створення захищеного віртуального з'єднання та управління потоками даних.

У VPN, створеному L2TP (приклад однієї з достатньо поширених схем застосування наведений на рис. 7.14), мережеві компоненти включають наступні три частини:

Рисунок 7.14. Схема підключення віддаленого користувача та віддаленої філії до головної корпоративної мережі через провайдера засобами протоколу L2TP

- Віддалена система – це може бути віддалений користувач, віддалена філія, яка має бути підключена до головної мережі VPN, та набір мережевих пристроїв.
- LAC (L2TP Access Concentrator – концентратор доступу L2TP). Відповідно до специфікації протоколу L2TP, роль сервера віддаленого доступу провайдера повинен виконувати концентратор доступу LAC, який реалізує клієнтську частину протоколу L2TP і забезпечує віддаленому користувачеві мережевий доступ до його локальної мережі через Інтернет. LAC – це пристрій, підключений до мережі з кінцевою системою PPP та можливістю обробки протоколу L2TP. Зазвичай це NAS (Net Access Server – сервер доступу до мережі) локального інтернет-провайдера і переважно використовується для надання послуг доступу користувачам типу PPP.

- LNS (L2TP Network Server – сервер доступу до локальної мережі L2TP).

LNS є кінцевою системою PPP та серверною частиною протоколу L2TP і зазвичай використовується як прикордонний пристрій корпоративної мережі. LNS є одноранговим пристроєм LAC та кінцевою точкою логічного завершення сеансу PPP.

В такій схемі криптозахищений тунель (L2TP+IPSec) утворюється між сервером провайдера та сервером віддаленого доступу локальної мережі.

Як видно з рисунку 7.13 протокол L2TP, так само, як і протокол PPTP – двоточковий протокол, що має клієнтську та серверну частини. Віддалена система – необов'язковий компонент. Роль LAC може відігравати хост клієнта віддаленого доступу.

7.5.2 Повідомлення протоколу L2TP

Аналогічно протоколу PPTP у роботі протоколу L2TP при обміні між LAC та LNS використовуються два види повідомлень – інформаційні та керуючі (рис. 7.15).

<table>
<tr><td colspan="2">PPP-кадри</td></tr>
<tr><td>Інформаційні повідомлення L2TP</td><td>Керуючі повідомлення L2TP</td></tr>
<tr><td>Інформаційний канал L2TP
(ненадійний)</td><td>Керуючий канал L2TP
(надійний)</td></tr>
<tr><td colspan="2">Транспортування пакетів L2TP (UDP, FR, ATM тощо)</td></tr>
</table>

Рисунок 7.15. Структура обміну протоколу L2TP

Однак, на відміну від PPTP, протокол L2TP не прив'язаний до протоколу IP, тому він може бути використаний у будь яких мережах з комутацією пакетів, наприклад, у мережах ATM (Asynchronous Transfer Mode) або в мережах з ретрансляцією кадрів (Frame Relay). Для транспортування як керуючих, так і інформаційних повідомлень використовується той самий транспорт. Найчастіше (це варіант будемо використовувати в подальшому описі) як транспорт використовується протокол UDP (для роботи протоколу L2TP зареєстрований UDP порт 1701).

Керуючі повідомлення використовуються для встановлення, обслуговування та керування передачею тунелів та сеансових з'єднань. Передача керуючих повідомлень є надійною та підтримує контроль потоку та контроль перевантаження. Основні керуючі повідомлення протоколу L2TP наведено у табл. 7.2.

Таблиця 7.2

Основні керуючі повідомлення протоколу L2TP

Підтримка керуючих з'єднань
1 (SCCRQ) Start-Control-Connection-Request
2 (SCCRP) Start-Control-Connection-Reply

3 (SCCCN) Start-Control-Connection-Connected
4 (StopCCN) Stop-Control-Connection-Notification
Керування з'єднаннями (L2TP-сесіями)
7 (OCRQ) Outgoing-Call-Request
8 (OCRP) Outgoing-Call-Reply
9 (OCCN) Outgoing-Call-Connected
10 (ICRQ) Incoming-Call-Request
11 (ICRP) Incoming-Call-Reply
12 (ICCN) Incoming-Call-Connected
14 (CDN) Call-Disconnect-Notify

Інформаційні повідомлення (повідомлення даних) використовуються для інкапсуляції кадрів PPP та передачі їх по тунелю. Надсилання інформаційних повідомлень ненадійне. Якщо пакет даних втрачено, його не буде передано повторно.

Пакети L2TP для контрольного та інформаційного каналів використовують один і той самий формат заголовка показаний на рис. 7.16.

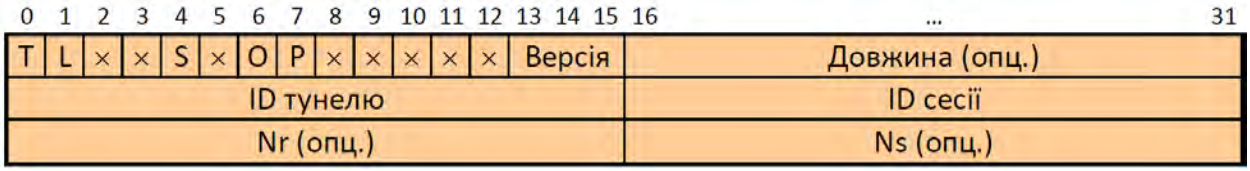


Рисунок 7.16. Формат заголовка L2TP

Найважливіші для розуміння роботи протоколу поля такі:

- Біт тип (T) – характеризує різновид пакета. Він встановлюється рівним 0 для інформаційних повідомлень та 1 – для керуючих.
- Довжина – вказує загальну довжину повідомлення у октетах.
- ID-тунелю – містить ідентифікатор керуючого з'єднання та обирається при формуванні тунелю.
- ID-сесії – визначає ідентифікатор для сесії даного тунелю та обирається при формуванні сесії.
- Ns – визначає порядковий номер інформаційного або керуючого повідомлення, починаючи з нуля та збільшуючись на 1 (за модулем 216) для кожного надісланого повідомлення.
- Nr – містить порядковий номер, який очікується для наступного повідомлення. Таким чином, Nr дорівнює Ns останнього по порядку отриманого повідомлення плюс один (за модулем 216). Порядкові номери необхідні у всіх керуючих повідомленнях і використовуються в каналі керування для забезпечення надійної доставки.

Зауважимо, що поля Довжина, Ns і Nr – опціонні для інформаційних пакетів, є обов'язковими для всіх керуючих повідомлень.

7.5.3 Організація взаємодії вузлів L2TP

L2TP-підключення включає два компоненти: тунель і сеанс (рис. 7.17). Тунель забезпечує надійне транспортування між двома кінцевими точками підключення L2TP управління та несе лише пакети управління. Сеанс логічно містяться в тунелі та переносить дані користувача. Один тунель може містити кілька сеансів.

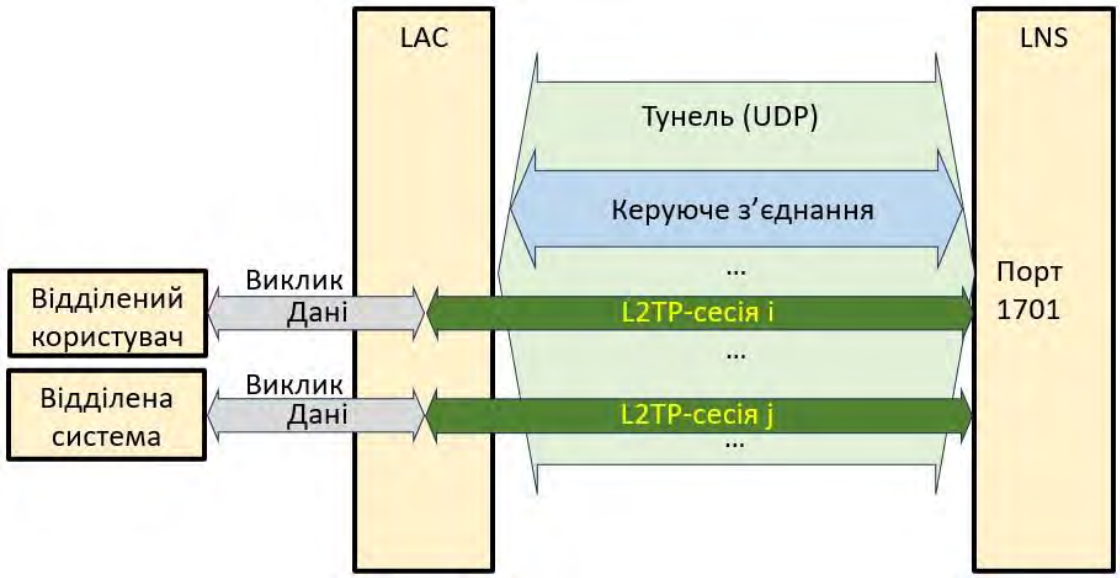


Рисунок 7.17. L2TP-тунелювання

Необхідна процедура встановлення PPP-сесії тунелювання L2TP включає два етапи:

1 Встановлення керуючого з'єднання L2TP-тунелю.

2 Формування L2TP-сесії відповідно до запиту вхідного або вихідного виклику. Керуюче з'єднання L2TP-тунелю є первинним, яке має бути реалізовано між LAC та LNS перш ніж запускати сесію. Встановлення керуючого з'єднання включає безпечну автентифікацію (опціону, CHAP-подібну) партнера, а також визначення версії L2TP, можливостей каналу, кадрового обміну тощо. Для встановлення керуючого з'єднання здійснюється обмін управляючими повідомленнями.

Типовим є наведений на рис. 7.18 обмін трьома повідомленнями:

У деяких випадках передається четверте повідомлення Zero-Length Body (ZLB) – повідомлення нульового розміру, яке служить для явного підтвердження пакетів на каналах із гарантованою доставкою.

Після успішного встановлення з'єднання, що управляє, можуть формуватися індивідуальні сесії (рис. 7.19). Кожна сесія відповідає одному інформаційному потоку PPP протоколу між LAC і LNS. На відміну від керуючого з'єднання організація сесії має напрями щодо LAC та LNS. Пристрій LAC запитує LNS сприйняття сесії для вхідних викликів, а LNS запитує LAC сприйняття сесії для вихідних викликів. На рис. 7.19 наведено приклад обміну керуючими повідомленнями для формування індивідуальної сесії з ініціативи LAC; наприклад, у разі надходження на LAC виклик від віддаленої системи, приєднаної до LAC (рис. 7.17).

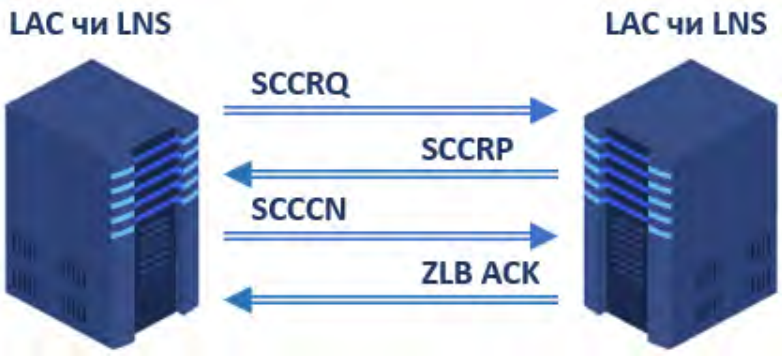


Рисунок 7.18. Обмін керуючими повідомленнями для встановлення керуючого з'єднання

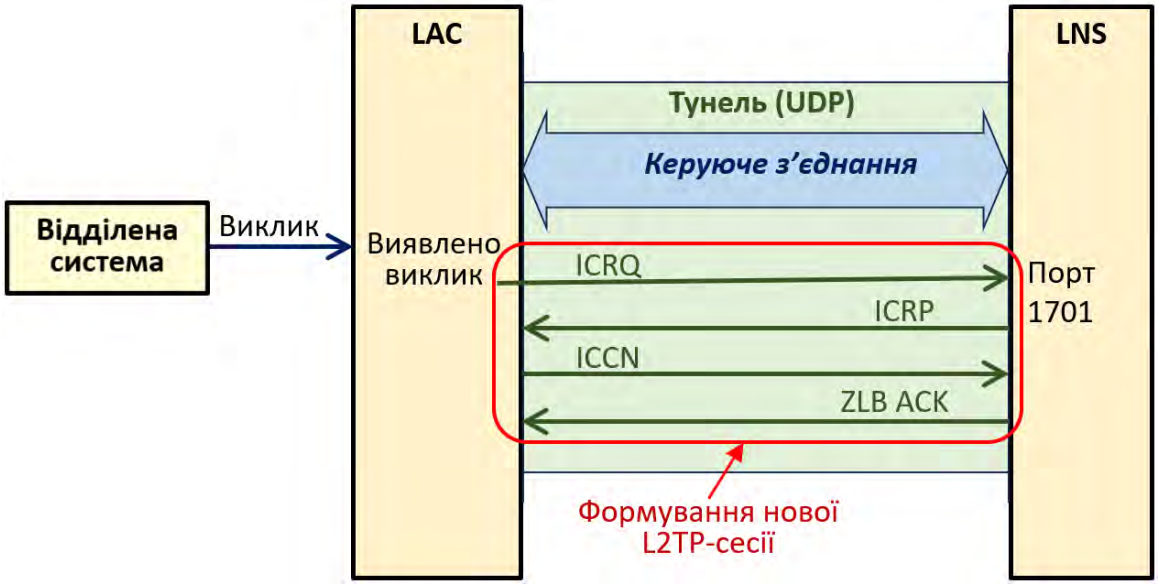


Рисунок 7.19. Обмін керуючими повідомленнями для формування індивідуальної L2TP-сесії за ініціативою LAC

Після обміну цим набором керуючих повідомлень між LAC та LNS сформується захищений тунель (L2TP-сесія). Слід врахувати, що між віддаленою системою та LAC захищений L2TP-тунель не створюється.

7.5.4 Формування та структура пакету L2TP

На рисунку 7.20 показано структуру інкапсуляції пакетів даних L2TP між LAC та LNS (передбачається, що використовується IP-мережа, зібраний IP-пакет міститься у кадрі канального рівня – Ethernet, ATM та ін., який вже передається на фізичний рівень).

У процесі передачі інформаційних повідомлень інкапсульовані кадри PPP можуть передаватися тунелем між концентратором LAC і мережевим сервером LNS в обох напрямках (по різних сесіях). При надходженні кадру PPP від віддаленого користувача концентратор LAC видаляє з нього байти обрамлення кадру, байти контрольної суми, потім інкапсулює його за допомогою L2TP протоколу в мережевий протокол і відправляє по тунелю мережевому серверу LNS.

Сервер LNS, використовуючи протокол L2TP, витягує з пакету кадр PPP і обробляє його стандартним чином.

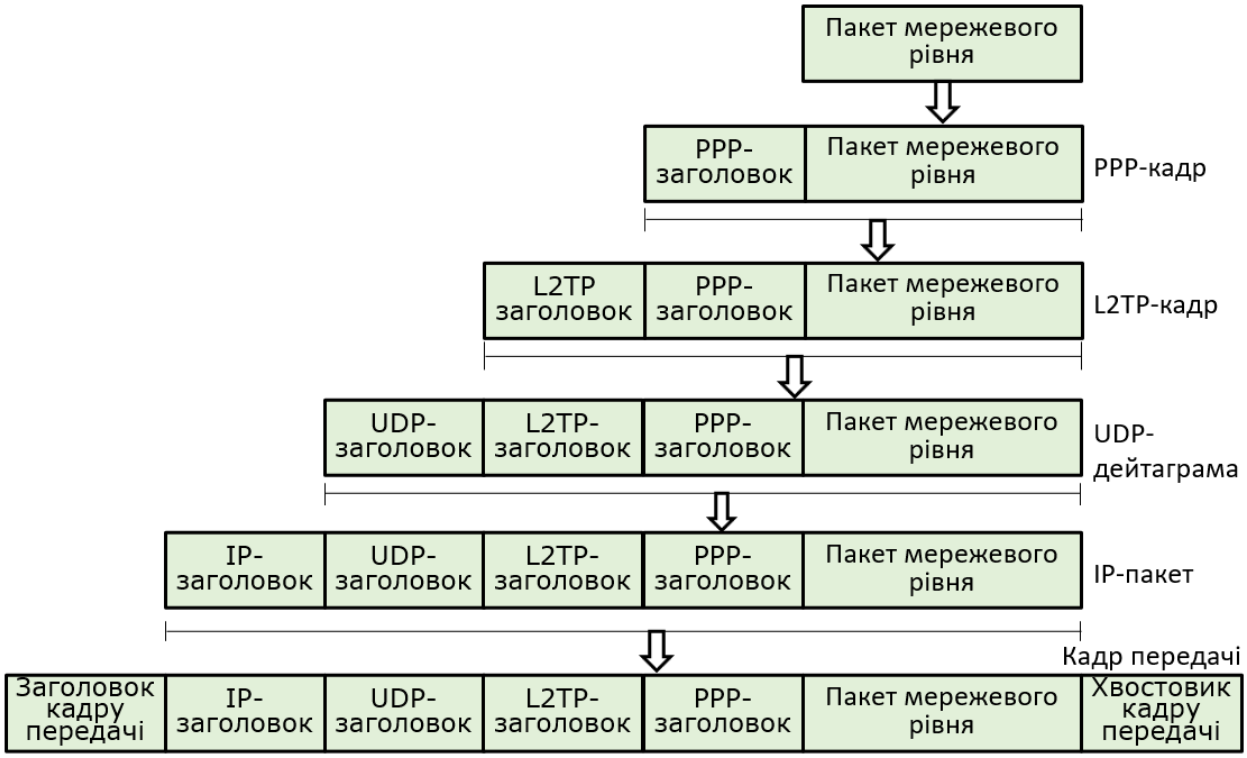


Рисунок 7.20. Структурна схема інкапсуляції при передачі даних за протоколом L2TP

7.5.5 Механізми захисту протоколу L2TP

Протокол L2TP може використовувати такі самі механізми аутентифікації, як і протокол PPTP (PAP, CHAP, MSCHAP, EAP). Що стосується шифрування, слід зазначити, що протокол L2TP не визначає конкретних методів шифрування і передбачає можливість застосування різних стандартів шифрування. Якщо захищений тунель планується сформувати в IP-мережах, тоді для реалізації криптозахисту, зазвичай, використовується протокол IPSec (рис. 7.21). На рисунку передбачається, що як дані мережевого рівня PPP-кадра є IP-пакет, а в системі IPSec застосований ESP-протокол в транспортному режимі.

7.5.6 Переваги протоколу L2TP

- На думку аналітиків протокол L2TP поверх IPSec забезпечує більш високий рівень захисту даних, ніж PPTP.
- На відміну від PPTP VPN протокол L2TP дуже надійний і не стикається з проблемами продуктивності при використанні в нестабільних з'єднаннях.

Рисунок 7.21. Приклад використання протоколу L2TP поверх IPSec

- Протокол L2TP не прив'язаний жорстко до протоколу IP і може як транспорт використовувати інші протоколи – Frame Relay, ATM тощо.
- На відміну від своїх попередників – протоколів PPTP і L2F, протокол L2TP надає можливість відкривати між кінцевими абонентами відразу кілька тунелів, кожен з яких може бути виділений для окремої програми. Ці особливості забезпечують гнучкість та безпеку тунелювання.

7.5.7 Недоліки протоколу L2TP

- Запропонована специфікація забезпечує стандартне шифрування лише в IP-мережах під час використання протоколу IPSec.
- Нижча порівняно з протоколом PPTP швидкість з'єднання.

7.6 Запитання до розділу

1) Протоколи яких типів застосовують для тунелювання?
2) Які складові протоколи) включає до себе протокол PPP?
3) Як працює протокол автентифікації PAP?
4) Як працює протокол автентифікації CHAP?
5) Які механізми захисту використовують протоколи PPTP, L2TP?
6) За яким протоколом передаються керуючи повідомлення протоколу PPTP?
7) Яка структура кадру формується протоколом PPTP для пересилання тунелем PPTP?
8) Яка схема інкапсуляції використовується при передачі даних за протоколом L2TP?

РОЗДІЛ 8 РОЗПОДІЛ КРИПТОГРАФІЧНИХ КЛЮЧІВ ТА УЗГОДЖЕННЯ ПАРАМЕТРІВ ЗАХИЩЕНИХ ТУНЕЛІВ

8.1 Загальні відомості щодо механізмів розподілу криптографічних ключів і узгодження параметрів захищених тунелів

Для роботи багатьох VPN потрібна підтримуюча інфраструктура, яка б забезпечувала розподіл криптографічних ключів і узгодження протоколів захисту між учасниками обміну. У ряді мережевих протоколів безпеки, наприклад IPSec, для створення криптозахищених тунелів між учасниками інформаційного обміну використовується таке поняття як асоціація безпеки (security_association – SA) – набір правил, процедур, параметрів, які застосовуються для забезпечення сервісу захисту транспортного потоку. До початку безпечної взаємодії сторони мають отримати узгоджений набір таких асоціацій, кожна з яких має власний ідентифікатор. У даному розділі розглядаються основні протоколи, які виконують ці завдання стосовно VPN IPSec, а також іншим протоколам захисту в стеку TCP/IP – протокол SKIP (Simple Key management for Internet Protocol – простий протокол управління ключами для Internet) та більш сучасний протокол IKE (Internet Key Exchange – обмін ключами в Інтернеті). Назва IKE прийшла в 1998 році на зміну ранішої – ISAKMP/Oakley – назву протоколів на базі яких побудований протокол IKE.

Роботу протоколу IKE можна визначити як роботу, переважно, двох інших протоколів.

• Протокол визначення ключів Oakley. Oakley являє собою протокол обміну ключами, заснований на алгоритмі Діффі-Хеллмана, але забезпечує додатковий захист. Протокол Oakley не диктує використання конкретних форматів повідомлень.

• Протокол захищених зв'язків та керування ключами в Internet (Internet Security Association and Key Management Protocol – ISAKMP). ISAKMP забезпечує підтримку необхідних форматів повідомлень для узгодження атрибутів захисту. При цьому ISAKMP не змушує використовувати якийсь конкретний алгоритм автентифікації або обміну ключами, а пропонує набір типів повідомлень, що дозволяють використовувати будь-який подібний алгоритм.

8.2 Організація роботи протоколу SKIP

8.2.1 Одноособиста робота протоколу SKIP

Протокол SKIP (Simple Key management for Internet Protocol – простий протокол управління ключами для Internet) розроблений компанією Sun Microsystems в 1994 році. У 1995 році протокол був удосконалений та дозволив працювати разом із протоколом IPSec. Цей протокол працює на мережевому рівні стека TCP/IP. Як IP-сумісний протокол, протокол SKIP забезпечує не тільки управління ключами шифрування, але й прозорий для додатків криптозахист IP-пакетів на мережевому рівні моделі OSI.

Протокол SKIP забезпечує захищену взаємодію між парою вузлів. При цьому кожен вузол має довгостроковий майстер-ключ. Цей майстер-ключ може бути сформований за допомогою алгоритму Діффі-Хеллмана. Нагадаємо, що при використанні даного алгоритму вузли вибирають велике просте число p – модуль, а також число $g < p$ (першорідний корінь p). Кожен абонент i генерує закритий ключ $x_i < p$, що є великим випадковим числом, та обчислює відкритий ключ y_i, що відповідає закритому ключу, відповідно до формули: $y_i = g^{x_i} \bmod p$. Якщо дві сторони А та В хочуть встановити захищене з'єднання, то сторони А та В обмінюються своїми відкритими ключами y_A та y_A й далі, кожна сторона використовуючи свій закритий ключ, обчислює загальний секретний ключ:

- сторона А

$$K_{AB} = (y_B)^{x_A} \bmod p = (g^{x_B})^{x_A} \bmod p = g^{x_B x_A} \bmod p,$$

– сторона В

$$K_{AB} = (y_A)^{x_b} \bmod p = (g^{x_A})^{x_B} \bmod p = g^{x_A x_B} \bmod p.$$

Загальний секретний ключ K_{AB} і є довгостроковим майстер-ключом – загальним (парним) для цих двох сторін. Надалі такий ключ будемо позначати $K_{\text{МКд}}$ – довгостроковий майстер-ключ.

Для того, щоб виключити заміну відкритого ключа партнера (атака «людина посередині»), відкриті ключі повинні мати сертифікат (як правило, за стандартом X.509).

Протокол SKIP допускає також ручний розподіл майстер-ключів, якщо відсутні сертифікати відкритих ключів партнерів.

Такий (по Діффи-Хеллману чи ручний) розподіл майстер-ключів є попередньою операцією, що виконується одноразово (або дуже рідко). Подальша взаємодія сторін відбувається в захищеному режимі. При цьому довгостроковий майстер-ключ, що є у сторін, не використовується безпосередньо для шифрування трафіку між партнерами. У протоколі SKIP використовується багаторівнева система ключів, як показано на рис. 8.1.

Послідовність захищеної взаємодії між сторонами відбувається за допомогою захищених SKIP-пакетів. Формування захищеного SKIP-пакета відбувається так (рис. 8.1).

1) Для кожного вихідного IP-пакету генерується (наприклад, з використанням генератора псевдовипадкових кодів – ГПК) одноразовий пакетний ключ $K_{\text{П}}$.

2) Вихідний IP-пакет шифрується (алгоритм шифрування визначається драйвером SKIP вузла-відправника) з використанням пакетного ключа $K_{\text{П}}$.

3) Зашифрований IP-пакет інкапсулюється в SKIP-пакет.

4) Для зашифрування пакетного ключа $K_{\text{П}}$ формується тимчасовий майстер-ключ $K_{\text{МКт}}$. Він формується з використанням хеш-функції h, аргументами якої є довгостроковий майстер ключ $K_{\text{МКд}}$ і деяке число n – значення лічильника, що постійно збільшується. Цей лічильник може

змінюватися на одиницю залежно від конкретної завдання щодня, щогодину, щохвилину тощо. Тобто $K_{МКт} = h(n,K_{МКд})$.

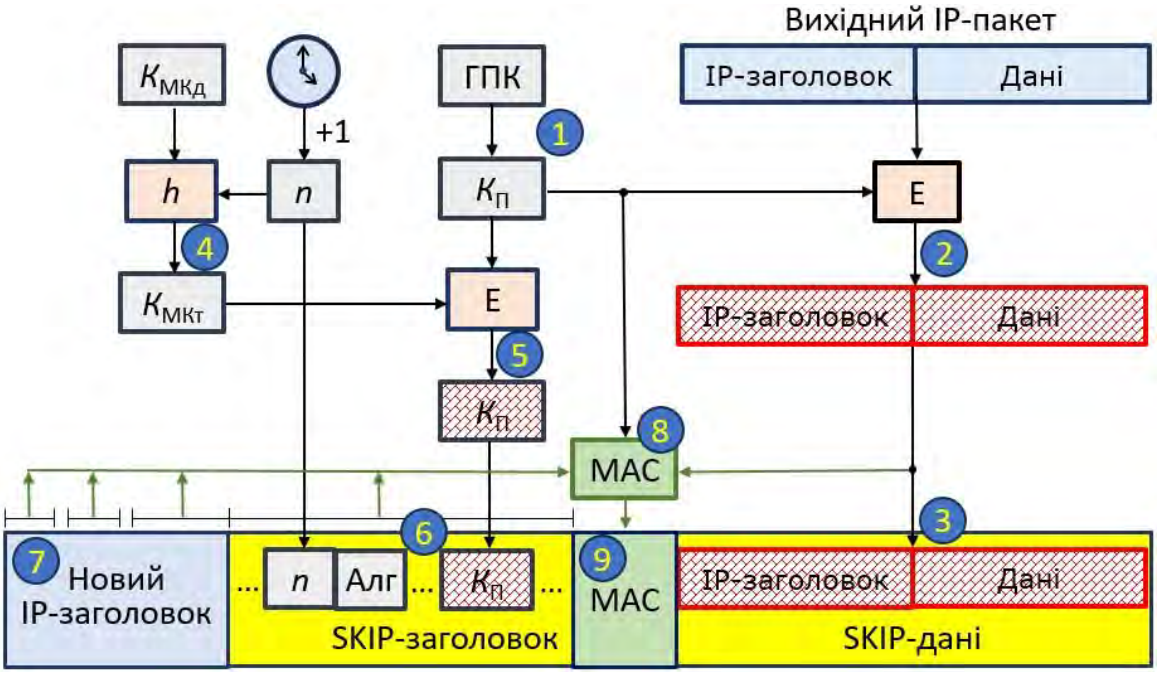


Рисунок 8.1. Схема формування захищеного SKIP-пакета

5) Пакетний ключ $K_П$ зашифровується (цей алгоритм шифрування також визначається драйвером SKIP вузла-відправника) з використанням тимчасового майстер-ключа $K_{МКт}$.

6) Формується заголовок SKIP-пакету. У цьому заголовку містяться: зашифроване значення пакетного ключа $K_П$; число n; ідентифікатори алгоритмів зашифрування IP-пакетів, пакетного ключа та алгоритму формування MAC (автентифікатора SKIP-пакету); деякі інші параметри SKIP пакета.

7) Формується нове значення IP-заголовка.

8) З використанням пакетного ключа $K_П$ формується значення MAC, яке захищає (автентифікує) значення зашифрованого IP-пакету, SKIP-заголовка та незмінних полів нового IP-заголовка.

9) Сформоване значення MAC міститься у SKIP-пакет.

При надходженні SKIP-пакета на вузол, що приймає, послідовність дій по обробці цього пакета показана на рис. 8.2.

1) Перевірка значення лічильника n заголовку SKIP-пакета. Правила для роботи з лічильником віднесені на розсуд розробника, але для забезпечення сумісності версій пропонується вважати, що n – час у годинах, відрахований від 00:00 01.01.95 Як правило, якщо значення лічильника n пакета, що прийшов, відрізняється більш ніж на 1 від лічильника приймаючого вузла, то пакет відкидається.

2) Обчислення значення тимчасового майстер ключа: $K_{МКт} = h(n,K_{МКд})$.

3) З використанням тимчасового майстер ключа розшифрування пакетного ключа $K_П$ зі SKIP-заголовка.

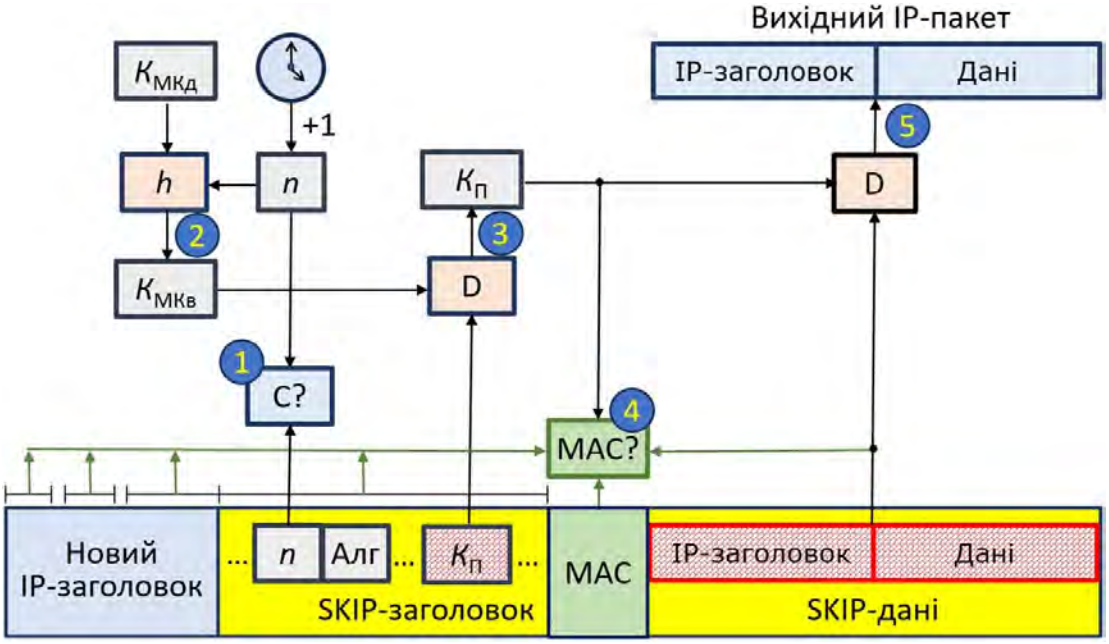


Рисунок 8.2. Послідовність дій приймаючого вузла з обробки SKIP-пакета

4) З використанням пакетного ключа автентифікація SKIP-пакету, тобто порівняння значення MAC, яке формується з незмінних полів IP-заголовка, SKIP-заголовка, SKIP-даних із значенням MAC в SKIP-пакеті, що прийшов.

5) З використанням пакетного ключа розшифрування вихідного IP-пакет з поля даних SKIP-пакету.

У деяких варіантах використання в полі даних SKIP-пакету міститься не зашифрований вихідний IP-пакет, а лише зашифроване поле даних вихідного IP-пакета. У такому випадку як заголовок SKIP-пакета виступає заголовок вихідного IP-пакета.

8.2.2 Робота протоколу SKIP спільно з IPSec

Під час роботи протокол SKIP разом із протоколами ESP чи AH захист IP-пакета реалізується засобами протоколів ESP чи AH. Завдання протоколу SKIP у цьому випадку полягає у розподілі секретних ключів для шифрування і Автентифікації.

Формування та шифрування пакетного ключа протоколу SKIP, генерація SKIP-заголовка проводиться стандартним чином, як зазначено вище. Приклади структури SKIP-AH та SKIP-ESP пакетів показані на рисунках 8.3 та 8.4.

Рисунок 8.3. Приклад структури SKIP-AH-пакету

Звернемо увагу, що поле «Prot» IP-заголовку має значення 47 – це номер протоколу SKIP, який присвоєний йому організацією IANA (Internet Assigned Numbers Authority).

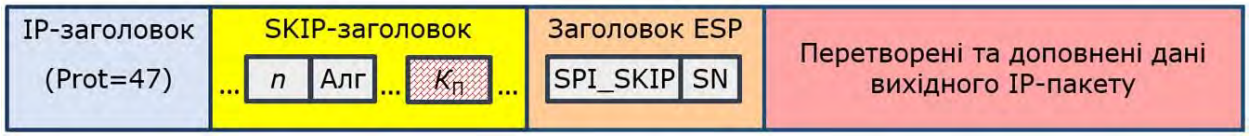


Рисунок 8.4. Приклад структури SKIP-ESP-пакету

У SKIP-заголовку міститься зашифроване значення пакетного ключа який буде використовуватися для шифрування та Автентифікації вихідного IP-пакета засобами ESP та/або AH відповідно до алгоритмів, коди яких також вказані у SKIP-заголовку.

У заголовках ESP і AH у полі SPI вказано значення SPI_SKIP рівне 1, яке говорить про те, що дані про ключ, алгоритми шифрування та автентифікації потрібно брати зі SKIP-заголовка.

Після приходу такого SKIP-пакета одержувачу протокол SKIP на стороні одержувача виконує дії показані на рисунку 8.2 (п.п. 1,2,3), формуючи значення пакетного ключа $K_П$, який передає протоколу IPSec. Далі протокол IPSec, використовуючи цей ключ, виконує необхідні перетворення пакета (розшифрування даних, автентифікацію).

Зазначимо, що алгоритми, зазначені у SKIP-заголовку пакета, попередньо не узгоджені з отримувачем. У отримувача може не бути засобів реалізації цих алгоритмів. Це один із недоліків протоколу SKIP.

8.3 Протокол Oaklay. Основні особливості та алгоритми

Oakley є варіант алгоритму обміну ключами, що виконується за вдосконаленою схемою Діффі-Хеллмана.

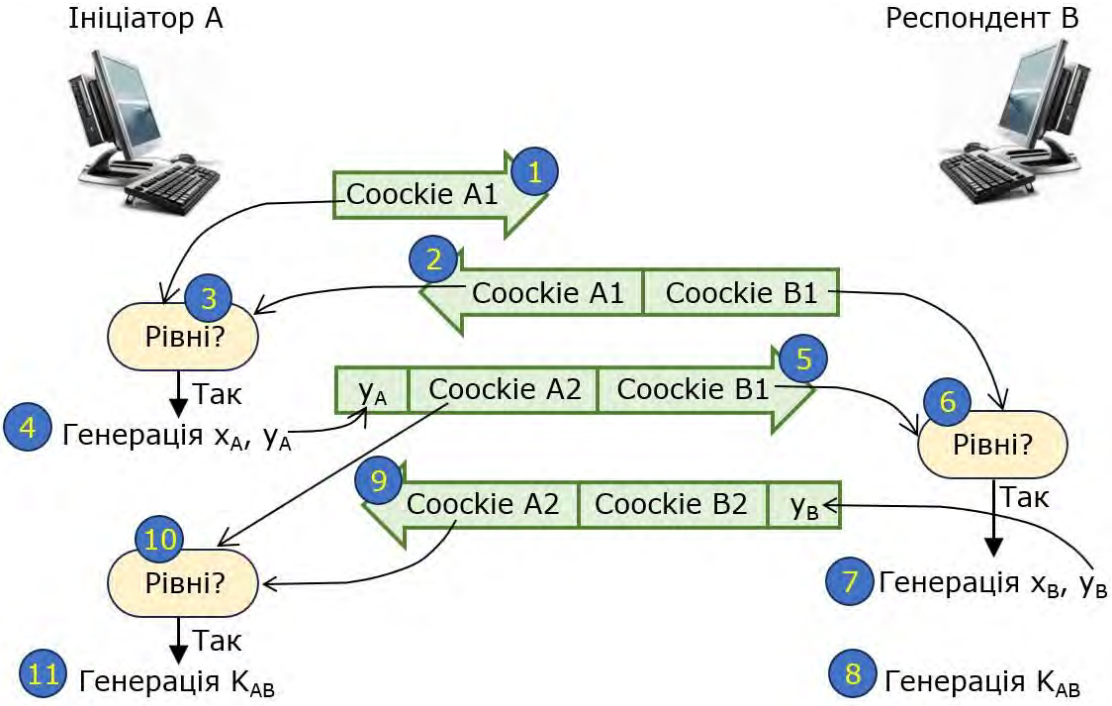
Алгоритм Oakley характеризується такими важливими особливостями.

1) Дозволяє двом сторонам домовитися про групу – глобальні параметри алгоритму обміну ключами Діффі-Хеллмана. Перші три групи представляють класичний алгоритм Діффі-Хеллмана, в якому відбувається зведення у ступінь в арифметиці класів відрахувань. Наступні дві групи використовують аналог алгоритму Діффі-Хеллмана для еліптичних кривих.

2) Здійснює обмін значеннями відкритих ключів Діффі-Хеллмана. За результатом формуються значення секретних ключів. Параметри обміну визначаються групою.

3) Використовує механізм так званих рецептів (cookies), щоб протистояти атакам засмічення. При цій атаці противник фальсифікує адресу законного джерела та посилає жертві відкритий ключ Діффі-Хеллмана, не очікуючи відповіді, яка буде відправлена законному джерелу. Жертва виконує модульне зведення у ступінь в арифметиці класів відрахувань (або композицію точок на еліптичній кривій), щоб обчислити секретний ключ. Багаторазово повторені повідомлення такого типу можуть засмічити систему сторони, що атакується,

марною роботою. Вимога обміну рецептами означає, що кожна зі сторін має надіслати у початковому повідомленні псевдовипадкове число, рецепт (cookie), який інша сторона має підтвердити (рис. 8.5). Подібні підтвердження повинні повторитись у наступних повідомленнях обміну ключами за Діффі-Хеллманом. Якщо адреса джерела була фальсифікована, противник відповіді не отримає і не зможе продовжити діалог. Таким чином, противник зможе змусити користувача генерувати підтвердження, а не виконувати обчислення з використанням алгоритму Діффі-Хеллмана.



x_A, x_B – секретні (особисті) ключі сторін A та B;
y_A, y_B – відкриті ключі сторін A та B;
K_{AB} – загальний (симетричний) секретний ключ сторін A і B, розрахований за секретним (особистим) ключем сторони та відкритим ключем партнера.

Рисунок 8.5. Класичний алгоритм Діффі-Хеллмана з використанням рецептів

Рецепт повинен залежати від параметрів і локальної секретної інформації сторони, що його генерує. Це не дасть можливості противнику отримати рецепт за допомогою реальних IP-адрес і порту UDP, а потім використовувати цей рецепт для того, щоб закидати жертву запитами з обраних випадковим чином IP-адрес або портів. Рекомендований метод створення рецептів полягає у швидкому обчисленні хеш-коду (наприклад MD5) для IP-адрес джерела та адресата, портів UDP джерела та адресата, локально генерованого секретного значення і поточного часу.

4) Використовує накази (nonce), щоб протистояти атакам відтворення повідомлень. При використанні класичного алгоритму Діффі-Хеллмана інформація від одного сеансу може бути записана без розшифровки та відтворена у майбутньому сеансі зловмисником. Для запобігання цій атаці до ряду повідомлень включаються оказії (nonce – «number that can only be used

once») Кожна оказія є локально породжене псевдовипадкове число. Оказії з'являються у відповідях та шифруються на певних стадіях обміну даними, щоб захистити їх від атаки відтворення повідомлень.

5 Виконує автентифікацію обміну Діффі-Хеллмана, щоб протистояти атакам посередника. З алгоритмом Oakley можуть застосовуватися три різні методи Автентифікації.

• Цифрові підписи. Автентифікація обміну даними здійснюється за допомогою підпису доступних обом сторонам будь-яких важливих параметрів, наприклад ідентифікаторів користувачів та оказій.

• Шифрування з відкритим ключем. Автентифікація обміну даними здійснюється за допомогою шифрування деяких параметрів обміну (наприклад, ідентифікаторів та оказій) з використанням особистого ключа відправника.

• Шифрування із симетричним ключем. Для автентифікації обміну даними може використовуватися шифрування параметрів обміну за симетричною схемою за допомогою деякого ключа, який отримується із застосуванням якогось додаткового механізму.

8.4 Протокол ISAKMP

8.4.1 Призначення та формат повідомлень

Протокол ISAKMP (Internet Security Association Key Management Protocol – протокол управління ключами та контекстами безпеки в Internet) використовується для узгодження параметрів та управління ключами при формуванні загального захищеного каналу та створення в його рамках окремих захищених з'єднань (SA). У порівнянні з протоколом SKIP протокол ISAKMP більш складний у реалізації, але забезпечує підвищену безпеку інформаційної взаємодії.

Усі комунікації IKE здійснюються у форматі пар повідомлень «запит-відгук», які надалі будуть називатися «обміном» (exchange) або «парою запит-відгук» (request/response pair).

Протокол ISAKMP визначає процедури та формати пакета, які використовуються для переговорів про створення, зміну або видалення захищених зв'язків (SA). Як частина процесу створення захищеного зв'язку, ISAKMP визначає корисний вантаж повідомлень обміну ключами та аутентифікації даних.

На рис. 8.6 показаний формат повідомлення ISAKMP. Видно, що повідомлення ISAKMP складається із заголовка та набору елементів корисного вантажу, кожен з яких, у свою чергу, має власний заголовок. Повідомлення ISAKMP зазвичай переносяться в дейтаграмах UDP (порти 500 та/або 4500).

8.4.2 Формат заголовка ISAKMP

Заголовок повідомлення ISAKMP (див. рис. 8.6) формується з наступних полів:

• Рецепт ініціатора (64 біти). Рецепт об'єкта, який ініціював процес створення, зміни чи видалення захищеного зв'язку.

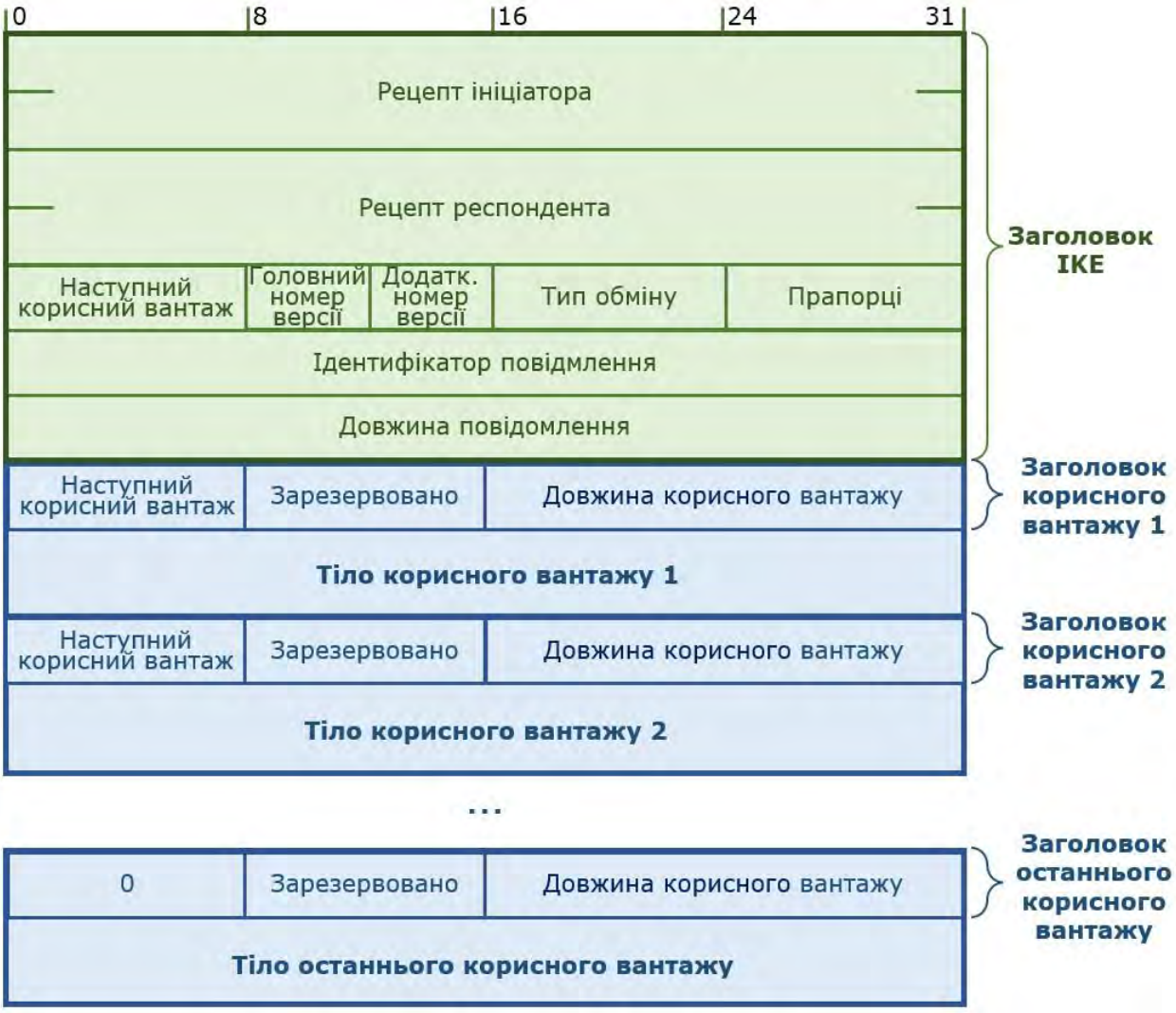


Рисунок 8.6. Формат повідомлення ISAKMP

• Рецепт респондента (64 біти). Рецепт об'єкта, що відповідає; у першому повідомленні ініціатора залишається порожнім.

• Наступний корисний вантаж (8 біт). Вказує тип першого корисного вантажу у повідомленні (типи корисного вантажу описані у наступному підрозділі).

• Головний номер версії (4 біти). Вказує головний номер версії ISAKMP, що використовується.

• Додатковий номер версії (4 біти). Вказує додатковий номер версії, що використовується.

• Тип обміну. Вказує тип обміну, який буде використовуватись. Це значення обмежує тип елементів даних, які передаються в кожному повідомленні обміну.

• Прапори (8 біт). Вказують параметри, встановлені для обміну ISAKMP. Поки що визначено два біти. Біт шифрування встановлюється тоді, коли всі

наявні корисні вантажі, що йдуть за заголовком, зашифровані з використанням алгоритму шифрування цього захищеного зв'язку. Біт фіксації призначений для того, щоб гарантувати, що шифрований матеріал не було отримано до створення захищеного зв'язку.

• Ідентифікатор повідомлення (32 біти). Унікальний ідентифікатор цього повідомлення.

• Довжина (32 біти). Довжина всього повідомлення (заголовка та всіх корисних вантажів) у байтах.

8.4.3 Типи корисного вантажу ISAKMP

Корисні вантажі у повідомленні ISAKMP переносять дані певного типу. Список типів корисного вантажу та характер даних, які переносить корисний вантаж певного типу, наведено у таблиці 8.1.

Таблиця 8.1

Список типів корисного вантажу у повідомленнях ISAKMP

Тип	Опис
Захищений зв'язок (SA)	Використовується для узгодження атрибутів захисту та вказівки області інтерпретації та ситуації, в рамках яких виконується таке узгодження
Пропозиція (P)	Використовується в ході узгодження параметрів захищеного зв'язку, що створюється, вказує застосовуваний протокол і число трансформацій
Трансформація (T)	Застосовується в ході узгодження параметрів захищеного зв'язку, вказує перетворення та відповідні атрибути захищеного зв'язку
Обмін ключами (KE)	Підтримує низку методів обміну ключами
Ідентифікація (ID)	Призначений для обміну інформацією ідентифікації
Сертифікат (CERT)	Служить для пересилання сертифікатів та іншої пов'язаної з сертифікатами інформації
Запит сертифіката (CR)	Використовується для запитів сертифікатів, вказує типи запитуваних сертифікатів і прийнятні центри сертифікації
Хешування (HASH)	Містить дані, що генеруються функцією хешування
Підпис (SIG)	Містить дані, що генеруються функцією цифрового підпису
Оказія (NONCE)	Містить оказію
Повідомлення (N)	Використовується для передачі даних сповіщення, наприклад ознаки виникнення помилки
Видалення (D)	Вказує захищений зв'язок, який більше не є дійсним

Усі корисні вантажі ISAKMP мають заголовки одного типу. Структура такого заголовка показано на рис. 8.6 Поле наступного корисного вантажу вказує на тип наступного корисного вантажу та має значення 0, якщо даний

корисний вантаж є в повідомленні останнім. Значення поля довжини корисного вантажу вказує довжину в байтах відповідного корисного вантажу, включаючи довжину його заголовка.

8.5 Протокол IKE. Фази роботи

8.5.1 Загальні відомості

Протоколи Oakley та ISAKMP – це інструменти, за допомогою яких протокол IKE (Internet Key Exchange) будує захищений тунель та захищені з'єднання всередені цього тунеля. Робота IKE може бути описана як дві фази з можливими попередніми діями (рис. 8.7). Перед виконанням першої фази можливо необхідно зробити певні попередні дії, наприклад розподілити між сторонами загальний секрет.

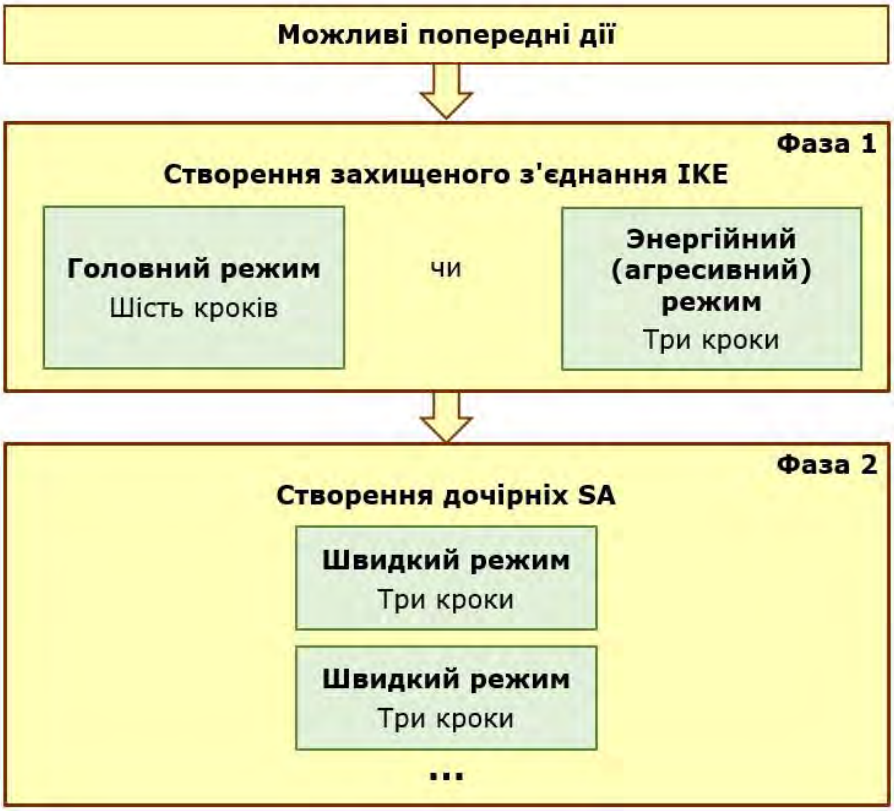


Рисунок 8.7. Фази роботи IKE

У першій фазі відбувається автентифікація учасників, сторони (мережні вузли) виконують узгодження деякого набору глобальних параметрів для формування загального захищеного тунелю (захищеного з'єднання IKE, так званого IKE SA). У термінології ISAKMP узгоджені глобальні параметри захищеного тунелю називають керуючим контекстом. До складу керуючого контексту входить головний ключ та набір сеансових ключів, сформованих на його основі. В даний час використовуються два режими для фази 1: головний режим та енергійний (агресивний) режим (див. рис. 8.7). Сформований захищений тунель із узгодженими глобальними параметрами є

двоспрямованим у тому сенсі, що кожна зі сторін має можливість ініціювати за допомогою цих параметрів окреме захищене з'єднання.

У фазі 2 після створення загального захищеного тунелю узгоджуються параметри окремих захищених з'єднань на основі сформованих глобальних параметрів каналу та утворюються контексти безпеки – SA (у стандарті IKEv2 вони називаються Child SA – дочірні SA). Кожне захищене з'єднання, створюване у межах загального захищеного тунелю, є односпрямованим з'єднанням, тобто з'єднанням від відправника пакета повідомлення до його одержувачу (рис. 8.8). Єдиний режим фази 2 – швидкий режим. За три кроки роботи в цьому режимі створюються два захищені з'єднання у різних напрямках, які дозволяють реалізувати симетричну взаємодію партнерів. У цій фазі може бути сформовано безліч пар захищених з'єднань (SA), які можуть бути використані протоколом IPSec та іншими протоколами захисту (див. рис. 8.8).

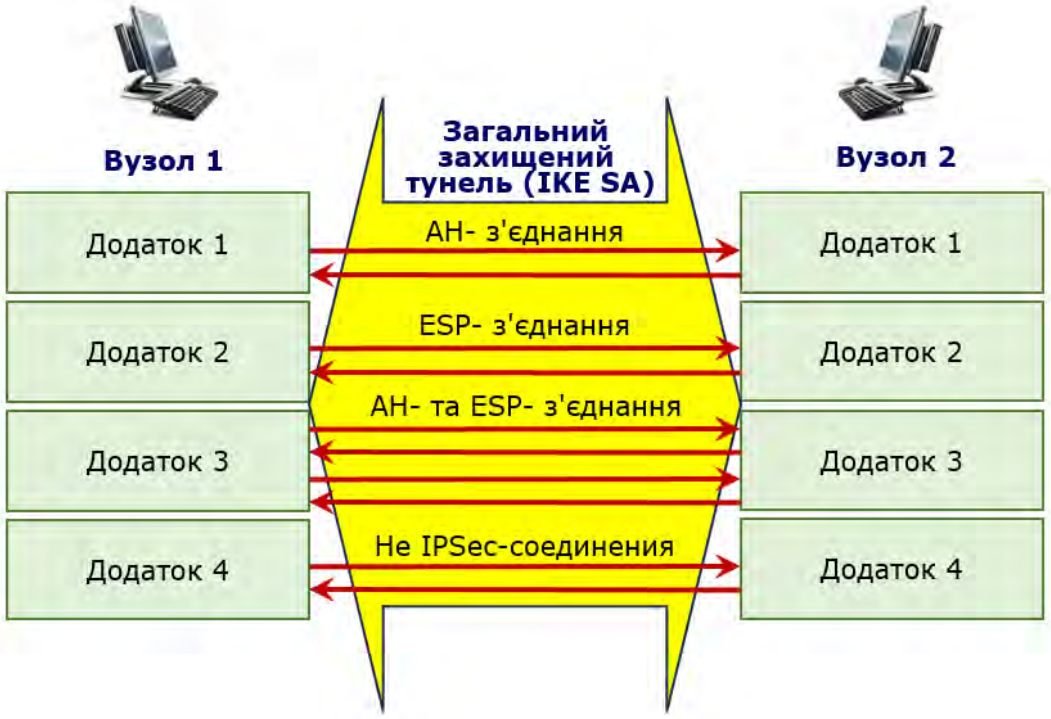


Рисунок 8.8. Формування захищених з'єднань протоколом IKE

При формуванні контексту безпеки узгоджується також термін існування захищеного з'єднання (один із параметрів SA), який залежить від необхідної криптостійкості. Час життя задається як для кожного захищеного з'єднання, так й для кожного захищеного каналу. Цей параметр може бути заданий максимальним інтервалом часу або максимальним обсягом переданих даних. Наприклад, термін існування захищеного з'єднання може бути визначений як 10 хвилин або 10 Гбайт, а термін існування захищеного каналу - як 40 хвилин або 40 Гбайт.

Контексти безпеки (SA) та їх ідентифікатори (SPI), які створюються для протоколу IPSec, зберігаються в базах даних SAD на кінцевих точках захищеного тунелю.

8.5.2 Фаза 1 – узгодження параметрів захищеного тунелю

Головний (основний) режим *(Main mode)* – входить у першу фазу і має на меті провести автентифікацію сторін і встановити захищене з'єднання (загальний захищений тунель IKE SA) для проведення другої фази.

Автентифікація сторін провадиться методами протоколу Oaklay (цифрові підписи, шифрування з відкритим ключем, шифрування з симетричним ключем). Розглянемо, наприклад, головний режим з використанням автентифікації методом цифрового підпису (рис. 8.9).

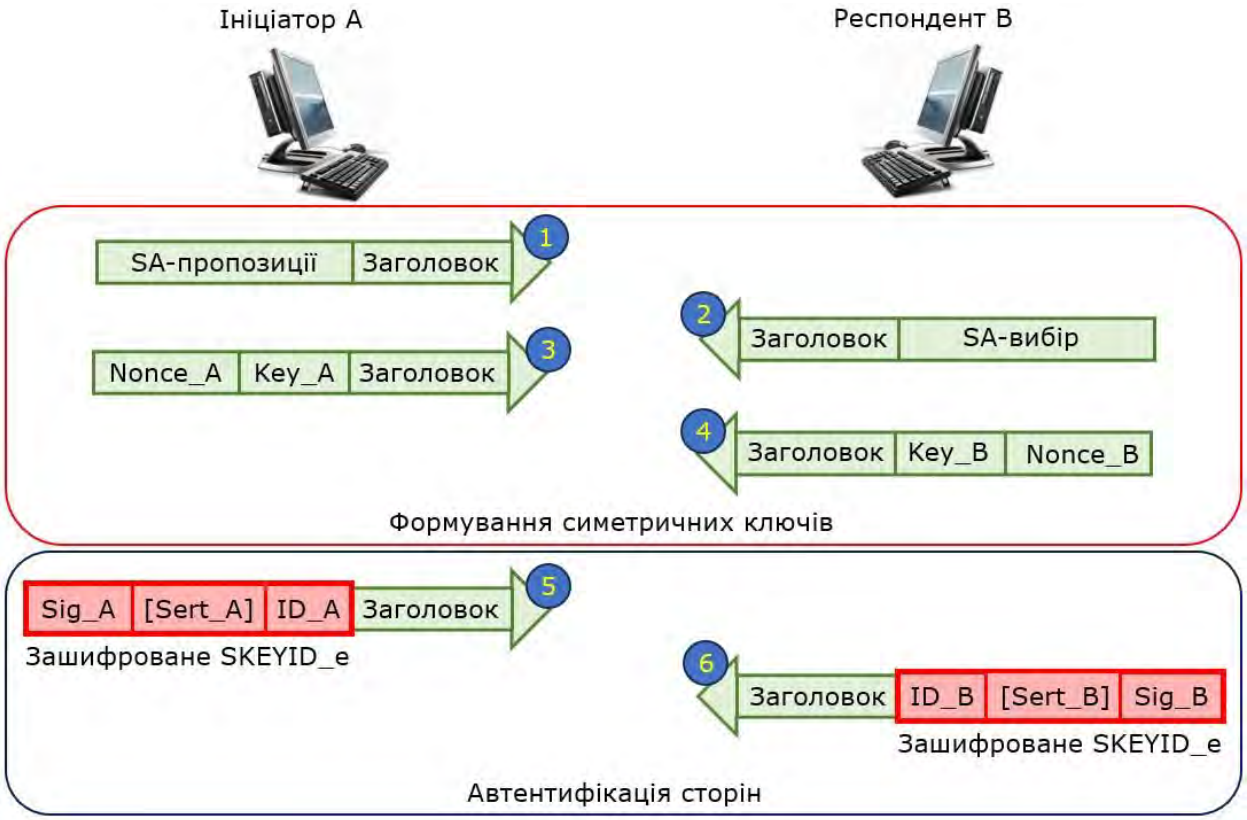


Рисунок 8.9. Основний режим фази I з використанням автентифікації методом цифрового підпису

Кожне із шести повідомлень, показаних на рисунку, має загальний заголовок IKE з рецепторами, які захищають обмін від атаки засмічення. Після заголовка слідує набір корисного вантажу певного типу.

Крок 1 Ініціатор захищеної взаємодії надсилає партнеру запит на формування захищеного каналу, що включає пропозиції щодо набору захисних алгоритмів та їх параметрів (корисний вантаж SA). У запиті ініціатора пропозиції щодо набору захисних алгоритмів та їх параметрів упорядковані за рівнем переваги.

Крок 2 Партнер у своєму повідомленні у відповідь інформує ініціатора про ті алгоритми захисту та параметри, які його влаштовують. Для кожної захисної функції (група ДХ, метод автентифікації, алгоритм шифрування) вибирається один алгоритм та його параметри з запропонованих.

Кроки 3 і 4 Ініціатор та партнер у корисному вантажі «Key» повідомлень надсилають один одному свої відкриті ключі ДХ, необхідні для вироблення спільного секретного ключа. Для забезпечення справжності відкритих ключів, що направляються один одному, доцільно відправляти цифрові сертифікати цих ключів (до складу яких входять і відкриті ключі) відповідно до стандарту X.509.

Відкриті ключі направляються взаємодіючими сторонами один одному разом із випадковими одноразовими числами Nonce, що служать для захисту від відтворення повідомлень.

Використовуючи відкриті ключі партнера та власні секретні ключі сторони виробляють спільний секретний ключ SKEYID за алгоритмом Діффі-Хеллмана (ДХ).

На основі загального секретного ключа сторони виробляється три види ключів:

• SKEYID_a – сеансові ключі, які використовуються для автентифікації сторін та узгоджуваних параметрів;

• SKEYID_e – сеансові ключі, які використовуються для шифрування узгоджуваних параметрів;

• SKEYID_d – ключовий матеріал, який використовується для створення тимчасових ключів для захищених з'єднань в другій фазі.

Кроки 5 та 6 Ініціатор та партнер обмінюються ідентифікаційною інформацією (власним ідентифікатором, який також знаходиться в заголовку повідомлення IKE), цифровим підписом хеша конкатенації всіх повідомлень п.п. 1-4 цього обміну (ключі для цифрового підпису створюються на попередньому етапі перед фазою 1) та сертифікатом відкритого ключа для цього підпису (опційно), зашифрованого ключами SKEYID_e.

Сторони, отримавши повідомлення, розшифровує їх ключем SKEYID_e, звіряють отриманий та власний хеші (вони мають бути однакові), перевіряють підписи. Наявність сертифіката у разі успішної перевірки підпису гарантує автентичність відправника.

Енергійний (агресивний) режим *(Aggressive mode)* призначений для тих же цілей, що й основний. Але він простіший у реалізації і водночас продуктивніший. Цей режим вимагає лише трьох кроків обміну, при цьому кількість пакетів, що передаються по мережі, вдається зменшити з шести до трьох (рис. 8.10).

Плата за ці переваги полягає в тому, що агресивний режим не забезпечує захист інформації, що служить для автентифікації сторін, оскільки така інформація передається через мережу в незашифрованому вигляді. З рисунка 8.10. видно, що секретні ключі можуть бути створені лише після другого

кроку, а автентифікаційна інформація починає передаватися саме на другому кроці до створення секретних ключів.

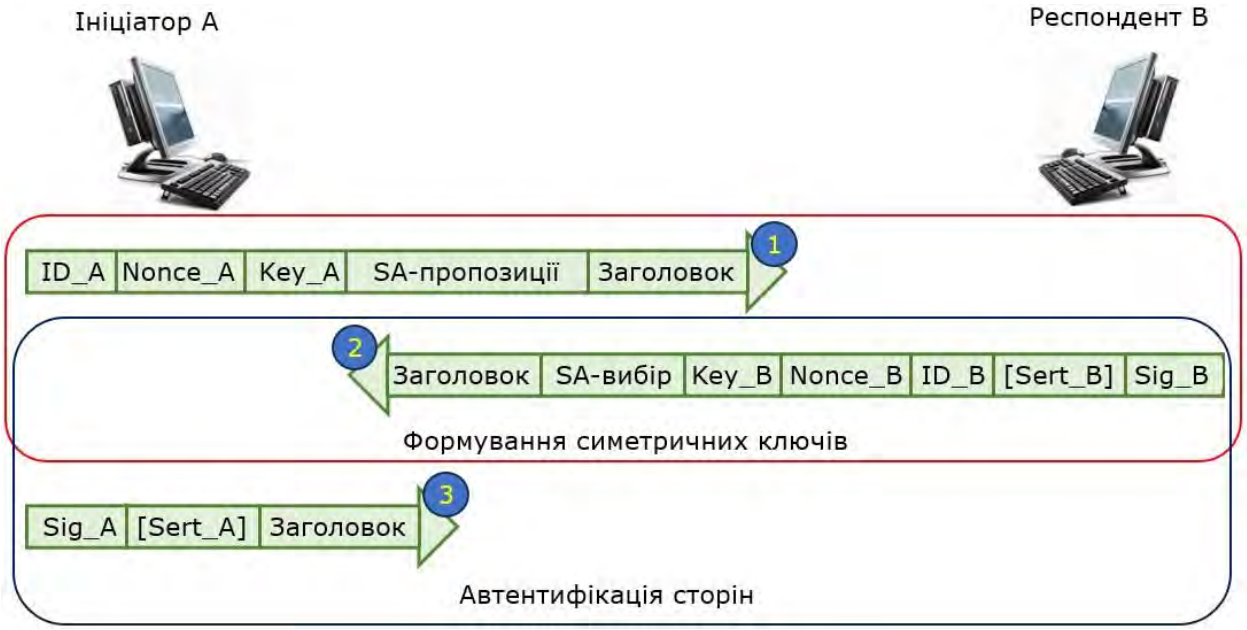


Рисунок 8.10.. Енергійний (агресивний) режим фази 1 з використанням автентифікації методом цифрового підпису

8.5.3 Фаза 2 – узгодження параметрів захищеного з'єднання

Після того, як у фазі 1 створено захищений тунель з його керуючим контекстом, може бути розпочата фаза 2 Для фази 2 в даний час існує лише один режим - швидкий режим.

Швидкий режим використовує захищений тунель, щоб створити безпечні асоціації (SA) для IPSec або будь-якого іншого протоколу. У цьому режимі узгоджуються параметри кожної безпечної асоціації SA і генерація для них нових ключів. Оскільки всі передачі здійснюються за захищеним тунельним з'єднанням, швидкий режим забезпечує надійний захист. Узгодження параметрів для двох захищених з'єднань (SA у прямому та зворотному напрямках) виконується за три кроки (рис. 8.11).

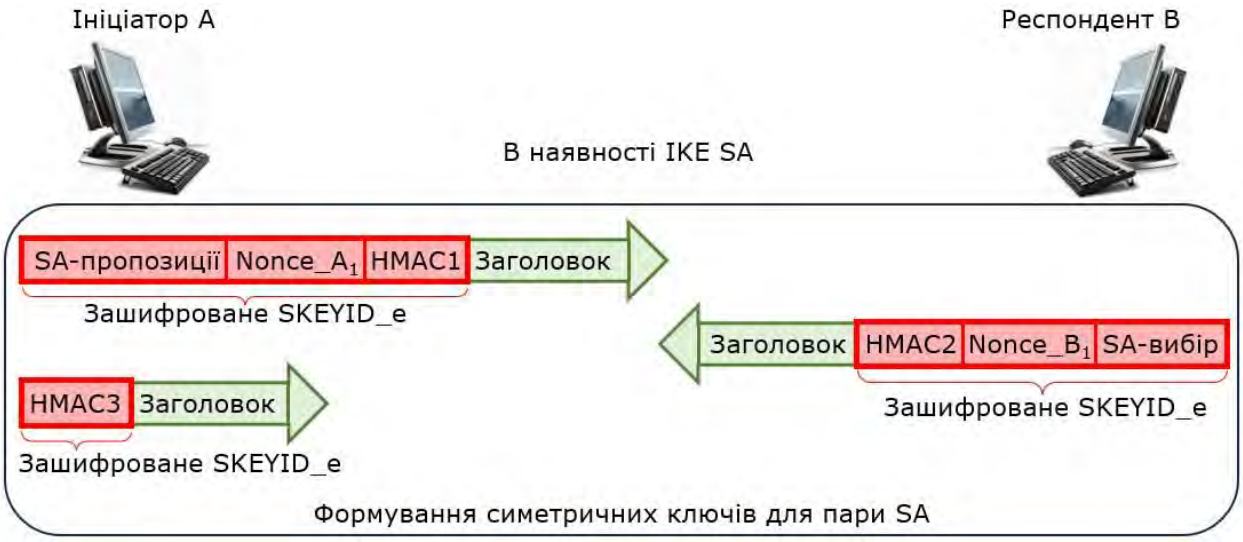


Рисунок 8.11. Швидкий режим фази 2

Крок 1 Ініціатор захищеного з'єднання направляє партнеру заявку на створення захищеного з'єднання (пропоновані алгоритми та їх параметри), а також свій одноразовий номер.

Крок 2 Партнер у своєму повідомленні у відповідь інформує про прийняті алгоритми захисту та параметри, а також повідомляє свій одноразовий номер.

Крок 3 Ініціатор надсилає партнеру свій одноразовий номер та одноразовий номер партнера.

Всі три повідомлення шифруються і автентифікуються за допомогою ключів SKEYID_e і SKEYID_a, сформованих у фазі 1 Для автентифікації використовується хеш-код, що обчислюється за допомогою хеш-функції, як аргументи якої виступають автентифіковане повідомлення і сеансовий ключ SKEYID_a.

8.5.4 Формування ключів для SA

Нагадаємо, що трикроковий режим фази 2 формує два захищених з'єднання (SA) – для прямого та зворотного напрямку (від ініціатора до партнера і назад). Кожне SA має мати свій індекс параметрів безпеки (SPI). Цей індекс створюється на основі рецептів із заголовків IKE.

Кожне захищене з'єднання (пряме та зворотне) повинно мати свій набір симетричних ключів для шифрування та автентифікації.

Симетричні ключі для захищеного з'єднання (SA) генеруються після завершення 3-го кроку шляхом застосування хеш-функції до ключа SKEYID_d з додатковими параметрами, до яких входять нонсенс ініціатора і партнера, а також SPI тієї SA, для якої ці ключі створюються.

Створений матеріал для ключів – односпрямований; кожна сторона створює свій різний матеріал для ключів, тому що матеріал, що використовується в кожному напрямку, різний.

8.6 Запитання до розділу

1) Які мережеві компоненти і якими механізмами захищає протокол SKIP?
2) На якому рівні моделі OSI працює протокол SKIP?
3) За допомогою якого протоколу формується довгостроковий майстер-ключ SKIP-вузла?
4) Для яких цілей використовується довгостроковий майстер ключ протоколу SKIP?
5) Назвіть важливі особливості протоколу Oakley?
6) Призначення протоколу ISAKMP?
7) Які фази роботи використовує протокол IKE?
8) Які режими можна використовувати в кожній фазі роботи протоколу IKE?

СПИСОК ЛІТЕРАТУРИ

1. Антонюк А. О. Основи захисту інформації в автоматизованих системах : навч. посіб. Київ : Видавничий дім "КМ Академія", 2003. 244 с.
2. Архипов О. Є., Луценко В. М., Худяков В. О. Захист інформації в телекомунікаційних мережах та системах зв'язку : навч.- метод. посіб. Київ : ІВЦ «Видавництво «Політехніка», 2003. 40 с.
3. Вертузаєв М. С., Юрченко О. М. Захист інформації в комп’ютерних системах від несанкціонованого доступу : навч. посіб. / за ред. С. Г. Лаптєва. Київ : Вид-во Європ. ун-ту, 2001. 321 с.
4. Грайворонський М. В., Новіков О. М. Безпека інформаційно-комунікаційних систем. Київ : Видавнича група BHV, 2009. 608 с.
5. Технології захисту інформації в інформаційно-телекомунікаційних системах : навч. посіб. / А. В. Жилін, О. М. Шаповал, О. А. Успенський ; ІСЗЗІ КПІ ім. Ігоря Сікорського. Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2021. 213 с.
6. Лагун А. Е. Криптографічні системи та протоколи : навч. посіб. Львів : Вид-во Львівської політехніки, 2013. 96 с.
7. Навчальні матеріали, розроблені в ході виконання міжнародного проекту SEREIN. *Веб-сайт проекту TEMPUS SEREIN.* URL: https://serein.eu.org/teaching-materials/ (дата звернення: 04.01.2024).
8. Skoudis E., Liston T. Counter hack reloaded : a step-by-step guide to computer attacks and effective defenses. 2nd ed. Hoboken : Prentice Hall, 2005. 784 p.

Навчальне видання

Жуковицький Ігор Володимирович,

Остапець Денис Олександрович

Захист інформації в комп’ютерних мережах

Навчальний посібник

Електронне видання

Експертний висновок склав д-р техн. наук, проф. А. Косолапов

Зареєстровано НМВ УДУНТ (№ 699 від 23.02.2024)

Формат 60х84 1/16. Ум. друк. арк. 8,49. Обл.-вид. арк. 5,88.
Зам. № 12

Видавець: Український державний університет науки і технологій.
вул. Лазаряна, 2, ауд. 2216, ауд. 263, м. Дніпро, 49010.
Свідоцтво суб’єкта видавничої справи ДК № 7709 від 14.12.2022