Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)
Факультет комп'ютерно-інформаційних систем і програмної інженерії
(назва факультету)
Комп’ютерні науки
(повна назва кафедри)

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломної роботи

магістр
(освітній рівень)

на тему: **Архітектура системи автентифікації користувачів інформаційної системи на основі віддаленого виділеного сервера**

Виконав: студент 6 курсу, групи СТм-61
спеціальності 126 «Інформаційні системи та технолгії»
(шифр і назва спеціальності)

		Сметанка Ю.А.
	(підпис)	(прізвище та ініціали)
Керівник		Харченко О.Г.
	(підпис)	(прізвище та ініціали)
Нормоконтроль		Мацюк О.В.
	(підпис)	(прізвище та ініціали)
Рецензент		
	(підпис)	(прізвище та ініціали)

м. Тернопіль – 2019

РЕФЕРАТ

"Архітектура системи автентифікації користувачів інформаційної системи на основі віддаленого виділеного сервера" // Сметанка Юрій Андрійович // Тернопільський національний технічний університет ім. І.Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра кібербезпеки, група СТм-61 // Тернопіль, 2019 // с. – , рис. – , табл. – , джерел – .

Ключові слова: АВТЕНТИФІКАЦІЯ, ШИФРУВАННЯ, ДОВІРЕНИЙ КОРИСТУВАЧ, KERBEROS, СЕРВЕР, ПРОТОКОЛ.

У магістерській роботі виконано дослідження способів забезпечення необхідного рівня захищеності комп'ютерних мереж на основі авторизації користувачів з використанням віддаленого сервера Kerberos. Здійснено огляд принципів авторизації та аутентифікації.

В дипломній роботі показано актуальність оцінювання рівня захищеності комп'ютерних систем з на основі ОС Windows з використанням служби автентифікації Kerberos. Проаналізовано основні механізми авторизації та принципи роботи такої системи з виділеним сервером Kerberos.

ANNOTATION

"System architecture of information system users' authentification based on remote dedicated server" // Diploma paper of Master degree level // Smetanka Yurii Andriyovych // Ternopil Ivan Puluj National Technical University, Faculty of Computer Information Systems and Software Engineering, Computer Science Department // Ternopil, 2019 // p. – , Fig. – , Tables – , Refence. – .

Key words: AUTHENTIFICATION, ENCRYPTION, TRUSTED USER, KERBEROS, SERVER, PROTOCOL.

The master's thesis investigates how to provide the necessary level of security of computer networks based on user authorization using a remote Kerberos server. Authorization and authentication principles are reviewed.

The diploma thesis shows the relevance of assessing the security level of computer systems based on Windows using Kerberos authentication service. The basic authorization mechanisms and principles of operation of such a system with a dedicated Kerberos server are analyzed.

ЗМІСТ

ВСТУП

Актуальність теми. Сучасні комп'ютерні розподілені системи характеризуються високим рівнем інтегрованості функціональних можливостей, підтримкою взаємодії декількох апаратних та програмних платформ, часто з використанням принципів розподіленості та паралельної роботи користувачів. Цей факт обумовлює високу складність проектованих систем та необхідність керування доступом до спільних ресурсів. Керуаання доступом – це механізм авторизації, який базується на автентифікації користувачів розподіленої системи.

Отже, автентифікація користувачів – важлива частина забезпечення інформаційної безпеки. Одним з методів автентифікації є використання протоколу Kerberos, з доступом до відповідного сервера. Тому тема роботи є актуальною.

Мета роботи. Метою роботи є аналіз методів і засобів автентифікації користувачів комп'ютерної розподіленої системи на основі сервера Kerberos.

Для досягнення вказаної мети в рамках дипломної роботи було сформульовано та розв’язано наступні задачі:

– дослідити сучасний стан технологій автентифікації користувачів;

– розробити модель бізнес процесу автентифікації з використанням віддаленого сервера;

– проаналізувати потенційні ризики методу автентифікації з використанням віддаленого сервера;

– запропонувати типове рішення для розподіленої комп'ютерної системи для авторизації її користувачів на основі віддаленого сервера.

Об’єкт дослідження: процеси забезпечення, контролю та управління безпекою у комп'ютерних системах.

Предмет дослідження: протокол авторизації на основі виділеного сервера Kerberos.

Методи дослідження. Для досягнення мети дипломної роботи використовувались:

– методи узагальнення та аналізу – при проведенні огляду стану механізмів автентифікації;

– формалізації та математичного моделювання – при аналізі методу стійкості шифрування паролів.

Наукова новизна отриманих результатів. Наукова новизна полягає у вирішенні задачі забезпечення захищеності особистих даних користувача розподіленої комп'ютерної системи. При цьому було отримано такі результати:

– систематизовано моделі автентифікації користувачів;

– запропоновано практичний приклад налаштування процесу автентифікації користувачів розподіленої системи.

Практичне значення отриманих результатів. Всі проаналізовані методи та засоби автентифікації можуть використовуватись практично при вивченні відповідних дисциплін, що стосуються адміністрування розподілених комп'ютерних систем, а також при побудові розподілених систем, для котрих критичним параметром є процес надійної автентифікації користувачів з використанням виділеного сервера.

Апробація результатів та особистий внесок здобувача. Основні положення роботи доповідались, розглядались та обговорювались на науковій конференції Тернопільського національного технічного університету. Результати дипломної роботи опубліковані у тезах студентської наукової конференції, яка проводились у ТНТУ.

РОЗДІЛ 1
ПРОБЛЕМИ БЕЗПЕКИ В РОЗПОДІЛЕНИХ СИСТЕМАХ

Термін Розподілені системи (Distributed systems) ще не набув повністю усталеного значення та змісту. У цій роботі визначаємо, що це така система, у якій кілька автономних процесорів і сховищ даних, підтримуючих процеси та / або бази даних, взаємодіють для досягнення загальної цілі. Процеси координують свою оперування та обмінюються інформацією за допомогою комунікаційної мережі.

В якості ілютсрації розглянемо розподілену систему для гіпотетичного бізнесу. У економічному відділі є власна мережа робочих станцій із засобами зберігання даних та друку звітів. Він пов’язаний з виробничими підрозділами, деякі з яких можуть знаходитись на віддалених майданчиках, які мають здійснювати спеціалізований контроль виробництва. Існують системи складів для ведення товарних запасів, відділи продажу, які генерують рахунки-фактури тощо. Усі ці окремі частини організації перебувають у взаємодії з корпоративним центром. Їх інформація сприяє палнуванню корпоративного бізнесу, а їх дії реагують на корпоративну політику. Це ілютсрація системи, яка виконує життєво важливу роль у функціонуванні та управлінні сучасним бізнесом. Тому її потрібно забезпечити.

1.1 Ризики розподілених систем, пов'язані з безпекою

У розподілених системах існують спеціальні фактори ризику. Існуючі розподілені системи пропонують значні можливості для впровадження небезпечного чи шкідливого програмного забезпечення. Вони також дозволяють злом і перегляд. Навіть ті розподілені системи, які призначені для підтримки бізнесу з низьким або середнім рівнем ризику, все ж повинні бути обережними сьогодні, щоб не залишати себе повністю незахищеними.

Потрібне пильне відстежування усіх спроб щодо атак, що призводять до відмови у наданні послуги. Навіть якщо ці напади не ставлять під загрозу цілісність даних , вони можуть бути як незручними, так і дорогими. Досвід людей, постраждалих від «Інтернет-хробака» (Spafford, 1988), це ілютсрує це. Навмисно створена програма розповсюджувалася в кількох мережах, особливо в США. Хоча вона сама по собі не заподіює ніякої шкоди, вона постійно відтворюється до тих пір, поки не поглинає всі ресурси комп'ютерів, на яких вона вторглась, і не зупинила їх. Вартість відновлення оцінювалася в мільйони доларів. Інші ризики такого роду описані в [1].

Подібні ефекти можуть бути викликані випадково. Зокрема, неправильне поводження з повідомленнями про помилки в системах електронної пошти може спричинити "поштові бурі", які загрожують мережею. Це можливо буде спричинено, якщо меседж, що містить помилки, транслюється на кілька сайтів. Якщо кожен приймальний сайт повідомляє про помилку джерела помилки і окремо викликає повторне повторення всієї трансляції, кількість меседжів зростає експоненціально, поки мережа не зупиниться.

Інший ризик полягає в тому, що незахищені системи можуть використовуватися як точка входу до інших недостатньо захищених, але чутливих систем. Цей випадок ілютсрував випадок німецьких хакерів, які отримали доступ до багатьох чутливих систем [2]. Вони використовували незахищені системи як майданчики, з яких систематично досліджували недоліки безпеки в інших більш чутливих системах, і робили це з надзвичайно високим рівнем успіху. Це призвело не лише до розкриття конфіденційної інформації, але й до додаткових витрат на кілька сайтів, які виявили лише те, що в них потрапили, коли їхні рахунки за комунікацію стали несподівано високими.

Існує прямий ризик виявлення конфіденційної інформації при неконтрольованому, незахищеному використанні публічних мереж між

вузлами системи для передачі інформації. Є багато можливостей для персоналу мережі отримати доступ до переданої інформації, але, крім того, будь-яка супутникова радіозв'язок або точка-точка можливо буде перехоплена відповідним обладнанням. Якщо потрібна захищена мережа, потрібне забезпечення зашифровування та контролю доступу.

Поширення не лише вводить додаткові ризики в комп'ютерні системи, але й ускладнює поводження з ризиками. Наприклад:

– комунікація може спричинити значні часові затримки в системі стосовно інформації, що стосується безпеки; це може ускладнити систему управління безпекою співвіднесення інформації, яка, разом узяте, вказувала б на порушення безпеки;

– розбиття системи на різні географічні, політичні, технічні чи адміністративні сфери ускладнює встановлення та управління узгодженою політикою безпеки; це також додає труднощів відслідковувати порушення безпеки, які ініціюються з іншого домену.

1.2 Переваги розподілених систем з точки зору безпеки

Однак, на додаток до зниження ризиків, у розподілених системах є компенсуючі фактори, які можна використовувати для підвищення безпеки системи.

Несанкціонований доступ до корпоративних даних може надати зловмиснику цінну стратегічну інформацію. Перевага розподілу в цьому випадку полягає в тому, що він дозволяє поширювати чутливі дані по всій системі. Таким чином, лише знаючи спосіб розповсюдження та доступ до нього у всіх місцях, зловмисник може отримати повну інформацію.

Збиток, який може призвести при порушенні можливостей обробки системи, можливо буде дуже високим, особливо коли висока премія надається здатності обробляти інформацію. Поширення може забезпечити альтернативні

місця, з яких можна придбати ресурси для обробки. Випадкові збої, як правило, трапляються на одному сайті одночасно, і навмисні спроби зірвати надання послуги вимагатимуть втручання на декілька сайтів одночасно.

У розподіленій системі можуть бути різні вимоги безпеки. Однією з переваг розповсюдження є те, що воно не обмежує всі компоненти системи приймати один і той же режим безпеки. Якщо оточення розділено на окремі домени безпеки, кожен домен може відображати різний аспект політики організації щодо безпеки. Загальний контроль отримують або шляхом узгодженої політики взаємодії між менеджерами доменів, або шляхом ієрархічної структуризації доменів, при цьому один менеджер бере на себе відповідальність за координацію взаємодії всіх.

Цілі безпеки в розподілених системах можна визначити на кількох різних рівнях, від цілі високого рівня, таких як "захист активів організації", до низького рівня, наприклад "гарантувати, що слова словника не використовуються як паролі", з ієрархією цілей між ними. Кожен рівень допомагає досягти цілей більш високого рівня. Ці цілі можуть бути досягнуті механізмами на декількох різних архітектурних рівнях в межах розподіленої системи. Прикладом цього є захист даних при передачі. Це можливо буде досягнуто шляхом захисту ліній зв'язку, захистом від кінця до кінця або на проміжному рівні. Поєднання цілей безпеки та архітектурних рівнів, на яких вони можуть підтримуватися, утворюють рамку, в якій можна описати безпеку.

Міжнародна організація стандартів (ISO) Відкриті системи взаємодії відкритих систем (OSI) Архітектура безпеки [3] визначає набір служб безпеки на основі загально узгоджених цілей та встановлює варіанти архітектурних рівнів, на яких вони можуть надаватися. Цілі більш детально описані в Огляді систем безпеки OSI [4].

1.3 Цілі забезпечення безпеки розподілених систем

Корисно розрізняти первинні та вторинні цілі безпеки. Основні цілі відповідають таким загрозам, як розголошення, корупція, втрата, відмова в наданні послуг, себе за себе, відмову. Вторинні цілі призводять до конкретизації послуг для підтримки первинних.

Є три основні цілі безпеки, які стосуються як збережених даних, так і меседжів, які перебувають у дорозі. Вони є:

Конфіденційність – збереження конфіденційності інформації, що зберігається в системах або передається між ними. Зазвичай це означає запобігання несанкціонованого доступу до збережених файлів даних та запобігання підслуховування меседжів у передачі. Однак у додатках підвищеної безпеки також можливо буде вимога щодо захисту найвищої інформації, яка може виводитися виключно з того, що дані передаються, а не з їх вмісту. Цю інформацію можна отримати з аналізу трафіку, аналізу джерела, місця призначення та обсягу зв'язку.

Цілісність – підтримка цілісності даних, що зберігаються в системах або комунікаційних між ними системах. Це запобігає втраті чи зміні інформації через, наприклад, несанкціонований доступ, збої компонентів або помилки зв'язку. У передачі даних можливо буде важливим також запобігання повторення меседж. Наприклад, меседж в системі електронних переказів коштів, що санкціонує переказ коштів з одного рахунку на інший, не повинно надсилатися та діяти двічі. Захист від цього ризику відомий як запобігання відтворення. Цілісність можливо буде досягнута двома різними способами : або взагалі запобігаючи виникненню збоїв, або виявляючи виникнення і відновлюючись після нього. Профілактика можливо буде досягнута рядом засобів; фізичним захистом, контролем доступу від несанкціонованих дій та процедурними заходами для запобігання помилок. Виявлення та відновлення

потребують своєчасного виявлення в поєднанні з резервними засобами, які дозволяють знову почати ситуацію з відомою цілісністю.

Доступність – підтримка доступності інформації, що зберігається в системах або передається між системами, гарантуючи, що служби, що надають доступ до даних, є доступними та дані не втрачаються. Загрози наявності можуть існувати на кількох рівнях. Файл даних недоступний для юзера, якщо комп'ютер, який надає послугу, апаратно знищений вогнем, або якщо файл було безповоротно видалено, або якщо зв’язок між юзерем та комп'ютером не відбувся. Як і щодо цілісності, доступні два різні способи захисту: запобігання; виявлення та відновлення за допомогою засобів резервного копіювання.

Дві інші основні цілі безпеки застосовуються спеціально для спілкування між юзерами та / або програмами:

Автентифікація – автентифікація особи, яка спілкується з партнерами, та автентичність походження та цілісності даних, що передаються між ними. Це важливо для кількох цілей. Ідентифікація особи, яка створює меседж, надає в електронних поштових системах конфіденційність того , що меседж є справжніми. Він також забезпечує основу для аудиту та бухгалтерського обліку. Це вимога до систем контролю доступу, заснованих на особі юзерів системи. Автентифікація вмісту меседж дозволяє знайти збої цілісності в повідомленнях.

Неприйняття – це запобігання юзереві неправильно заперечувати надсилання або отримання меседж. Перший з них відомий як доказ походження, а другий як доказ доставки. Неприйняття чинності є важливим у будь-якій ситуації, коли інтереси сторін, що надсилають та приймають, можуть конфліктувати. Наприклад, у системі передачі акцій було б у економічних інтересах відправника відмовитись від замовлення на продаж, якщо вартість запасу внаслідок цього зростає, а в інтересах отримувача – відмовити його у випадку невдачі. Це є ключовим питанням для контрактних

систем, заснованих на EDI (Електронний обмін даними), наприклад, системи придбання та постачання.

Вторинні цілі безпеки, визначені Архітектурою безпеки, є такими:

Управління доступом – надання контролю доступу до служб або їх компонентів для забезпечення доступу юзерів лише до служб та доступу до даних, на які вони мають право. Управління доступом – це один із засобів, який використовується для досягнення конфіденційності, цілісності та доступності. Це може забезпечуватися фізичними та / або логічними механізмами.

Несанкціонованому доступу до персонального комп'ютера можна запобігти дезактивацією клавіатури. Доступ до спільної системи може контролюватися логічною системою контролю доступу, використовуючи правила доступу, засновані на автентифікованій особі юзерів.

Аудиторський слід – надання аудиторського сліду оперування в системі для забезпечення підзвітності юзерів. Аудиторський слід містить свідчення того, хто що робив і коли. Важливий особливий випадок аудиту систем контролю доступу обговорюється в розділі VB

Сигналізація безпеки – виявлення подій, що вказують на фактичну або потенційну помилку безпеки, повинно викликати тривогу і призвести до того, що система працюватиме в режимі безвідмовної роботи. Деякі збої в безпеці не виявляються в той час і про них не можливо буде повідомлено, як, наприклад, невдача системи контролю доступу знайти несанкціонований доступ через власну слабкість. Інші види оперування можуть свідчити про можливі збої в безпеці та потребують розслідування; наприклад, змінений шаблон доступу юзерем. Завданням у цій ситуації є одночасно звести до мінімуму ризик втрати, якщо дійсно є збій безпеки та незручності для юзера , якщо помилкова тривога.

Викладені вище цілі безпеки є взаємозалежними і не повинні прийматися окремо. Автентифікація є основою для досягнення багатьох інших

цілей. Автентифіковані унікальності юзерів потрібні для контролю доступу, заснованого на унікальності, нерозподілу та аудиту, але для автентифікації на основі пароля потрібен як управління доступом, щоб захистити файл пароля, так і конфіденційність на основі зашифровування для подальшого захисту, якщо управління доступом не виконаний. Управління доступом, крім того, що вимагає та підтримує автоматичну перевіряння, є основою для конфіденційності, цілісності та доступності. Аудиторські шляхи та сигналізація про безпеку залежать від інших цілей та підтримують їх.

1.4 Архітектурні рівні служб безпеки

Архітектура безпеки ISO визначає можливі рівні протоколу зв'язку базової моделі взаємозв’язку відкритих систем, на якій могла б покластися кожна служба безпеки. Служба безпеки, така як конфіденційність, може застосовуватися для спілкування на різних рівнях моделі, але застосовувати службу на всіх рівнях недоцільно. Наприклад, юзереві, який отримує конфіденційність від кінця до кінця за допомогою зашифровування на шарі презентації, також не потрібно зашифровування даних для передачі даних. Подальша робота зі стандартів дозволить визначити відповідні профілі служб безпеки для конкретних програм.

Для досягнення цілей безпеки використовується низка різних механізмів. Вони включають:

- фізична та електронна безпека компонентів системи;
- механізми автентифікації;
- механізми контролю доступу;
- механізми захисту зв'язку.

Вони коротко описані тут. Зацікавлені читачі перейдуть до подальшого читання для більш детальної інформації.

1.5 Механізми апаратної безпеки

Механізми апаратної безпеки використовуються для захисту обладнання та для контролю доступу поза межами логічного контролю доступу або зашифровування. Вони потрібні для захисту від ризиків, таких як пожежа, буря, терористичні атаки та випадкові або зловмисні збитки з боку юзерів та технічних працівників. Фізична безпека вимагає різноманітних механізмів:

– Превентивна безпека – міцна конструкція, замки на дверях, вогнестійкість та гідроізоляція;

– Виявлення та стримування – сповіщувачі руху та дверні вимикачі, пов'язані з сигналізаціями, освітленням безпеки та телевізором із замкнутим контуром;

– Відновлення – надання резервного веб-сайту з альтернативними схемами обчислень та комунікацій.

Базовий рівень апаратної безпеки завжди потрібний навіть за умови логічного контролю доступу та зашифровування. У деяких ситуаціях апаратний захист можливо буде простішим та безпечнішим, ніж логічне рішення; наприклад, керуючи фізичним доступом до терміналів та персональних комп'ютерів та їх даних та зберігаючи конфіденційні дані на знімних носіях.

Фіг.3 ілютсрує ситуацію, коли зашифровування потрібно доповнити фізичним захистом лінії, якщо потрібно повністю захистити від кінця до кінця . Це потрібно, оскільки блок зашифровування не є невід'ємною частиною захищеного терміналу.

1.6 Механізми електронної безпеки

Можуть знадобитися електронні механізми захисту для захисту від перешкод від статичної електрики та радіочастотних перешкод, що може

призвести до несправності комп'ютерного та комунікаційного обладнання. Вони також потрібні для радіаційної безпеки, щоб уникнути пасивного підслуховування електромагнітного випромінювання від візуальних дисплеїв, принтерів та процесорних систем . Модульовані сигнали можуть бути виявлені сусідніми радіоприймачами та проаналізовані для виявлення даних, що відображаються, друкуються або обробляються. Профілактичні пристрої є у продажу, а також існують військові стандарти захисту (так звана "захист від шкідників").

1.7 Автентифікація

Метою особистої автентифікації в комп'ютерних системах є перевіряння заявленої особи людини. Для цього існує ряд різних механізмів, які базуються на одному або декількох принципах наступних крил:

– особиста характеристика юзера (відбиток пальців, геометрія руки, підпис тощо), яка є унікальною для людини;

– володіння юзера, наприклад, магнітно або електронно кодована картка, яка є унікальною для цієї особи;

– інформаційний іон, відомий лише юзереві, наприклад, секретний пароль або ключ зашифровування.

Секретні особисті паролі – це найпростіший і найдешевший спосіб втілення, і вони забезпечують належний рівень захисту для додатків середнього та низького рівня безпеки. Їм потрібен ряд підтримуючих заходів, якщо їх не підірвати. Заходи включають: регулярну зміну юзерем, одностороннє зашифроване сховище, мінімальну довжину та керований формат (наприклад, слова без словника), обмежена кількість дозволених спроб, а також перевіряння та перевіряння всіх збоїв. Їх можна посилити, обмеживши юзерів входити в конкретні апаратно захищені термінали;

наприклад, службовці з оплати праці можуть входити в цю службу лише за допомогою одного з терміналів, розміщених в конкретному офісі.

Застосування паролів через відкриті канали зв'язку в розподілених системах є особливою проблемою, оскільки пароль можна знайти, підслухавши канал, а потім використати його для себе. Одним з варіантів цього є застосування одноразових паролів, створених смарт-картами (див. Нижче).

Картки з магнітним кодом мають деякі переваги перед паролями – їх неможливо скопіювати так легко і їх менш легко забути. Однак вони також страждають від можливої експозиції свого вмісту у відкритих каналах комунікації.

Смарт-карти пропонують підвищену безпеку, оскільки їх можна запрограмувати для надання змінної інформації. Існує кілька режимів, в яких їх можна використовувати для особистої автентифікації. Дві з них:

– Генератори одноразових паролів, які генерують інший paSword кожен раз, коли вони використовуються. Один комерційний продукт щомісяця змінює пароль. У всіх випадках обчислювальна служба повинна синхронізуватися з генератором паролів.

– Пристрої реагування на виклики Хост надсилає номер виклику та смарт-карту h для обчислення правильної відповіді, включаючи вхід від юзера.

Смарт-карти стають дешевшими та простішими у використанні, і вони обіцяють забезпечити задовільний спосіб подолання проблем особистої автентифікації в розподілених системах. Однак система автентифікації повинна вирішувати проблему захисту захищеної інформації, на якій вона ґрунтується. Це аспект управління безпекою (див. Розділ VA).

Метою автентифікації меседжів у комп'ютерних та комунікаційних системах є перевіряння того, що меседж надходить від заявленого джерела і що воно не було змінено при передачі. Це особливо потрібно для EFT (Електронний переказ коштів). Механізм "p rotection" – це генерування коду

автентифікації меседж (MAC), приєднаного до меседж, який можливо буде перерахований отримувачем і знайде будь-які зміни на шляху. Див. Рисунок 4. Один уніфікований спосіб описаний у (ANSI, X9.9). Механізми автентифікації Messa можуть також використовуватися для досягнення неповернення меседжів.

1.8 Логічне управління доступом

Логічний управління доступом повинен використовуватися, коли апаратний управління доступом неможливий, як це має місце в системах з багатьма юзерами. Модель управління логічним доступом надається Довідковим монітором, який перехоплює всі спроби доступу та дозволяє їх лише у випадку дозволу доступу. В іншому випадку доступ заблокований, юзереві повертається меседж про помилку та вживаються відповідні реєстраційні та тривожні дії .

Існує дві основні форми логічного контролю доступу: обов'язковий управління доступом, заснований на фіксованих правилах; та дискреційний управління доступом, який дозволяє юзерам ділитися та контролювати доступ (див. розділ VI.A). Рекомендований дискреційний підхід до контролю доступу

Ідентифікація / авторизація. Система забезпечує автентифікацію ідентифікацій юзерів під час входу в систему, а контрольний монітор приймає рішення на основі правил доступу, що стосуються юзера, сутності, до якої звертаються, та операції, яку юзер намагається здійснити.

Є дві основні реалізації правил доступу:

– До цільових об'єктів додається Список контролю доступу (ACL), який визначає юзерів, які мають право доступу до них та операції, які вони можуть виконувати;

– Юзері отримують автентифіковані можливості, які діють як квитки, що дозволяють їм отримувати доступ до визначених ресурсів.

Багато персональних обчислювальних систем забезпечують лише управління доступом на основі паролів файлів. Вони забезпечують мінімальний, простий у використанні рівень захисту, який є адекватним для систем із низьким рівнем безпеки.

1.8 Механізми безпеки комунікації

Існує два основних механізми забезпечення безпеки зв'язку, крім апаратного захисту ліній та обладнання: зашифровування; і прокладка руху.

Зашифровування – один з найважливіших прийомів захисту комп'ютера та зв'язку. Криптографія означає буквально «таємне написання». Зашифровування перетворює (шифрує) звичайний текст у шифротекст, який неможливо прочитати, а дезашифровування знову перетворює його на читабельний звичайний текст (розшифровує його). Криптографією займаються тисячі років, але поява алгоритмів зашифровування на основі комп’ютера змінило її з важкої та ненадійної на просту та потужну. Алгоритми, такі як Стандарт зашифровування Da ta, описаний нижче, легко доступні, прості у використанні та забезпечують високий ступінь захисту від загроз конфіденційності та цілісності комунікацій.

Тут буде розглянуто лише короткий пробіг. Його ціль – приховати існування меседжів на лінії зв'язку, вставляючи фіктивні меседж на лінію, щоб забезпечити рівномірний рівень трафіку в усі часи. В основному це цікавить військовий рівень безпеки.

Зашифровування можна використовувати для декількох цілей: запобігання підслуховуванню; виявлення зміни меседжів; і, у поєднанні з застосуванням унікальних ідентифікацій меседжів, виявлення видалення та повторного відтворення меседж.

Зашифровування може використовуватися на окремих повідомленнях або в кінці. Зашифровування меседжів охоплює лише комунікаційні зв'язки.

На відміну від цього, кінцеве зашифровування проводиться безпосередньо між ініціюючою та цільовою системами. Проміжний рівень – це мережеве зашифровування, де зашифровування охоплює всю мережу, але не шлюзи між мережами. У всіх випадках, коли зашифровування здійснюється окремим апаратним блоком, зв'язок між терміналом і блоком зашифровування не охоплюється, а додатково потрібний апаратний захист. Дивіться рисунок 3.

Існує два основних типи зашифровування:

Зашифровування з секретним ключем, яке використовує єдиний секретний ключ, що ділиться між відправником та отримувачем.

Зашифровування з відкритого ключа, яке використовує пов'язану пару ключів. Один ключ є загальнодоступним і може використовуватися для зашифровування меседжів, тоді як інший ключ є секретним, відомим лише отримувачу, і він можливо буде використаний для розшифровки меседжів.

1.9 Політика безпеки

Політика – це плани організації щодо досягнення її цілей. У контексті безпеки політика безпеки визначає загальні цілі організації щодо ризиків для безпеки та плани поводження з ризиками відповідно до цих цілей. Політика зазвичай є ієрархічною; плани політики високого рівня – це цілі, які повинні вирішувати політики нижчого рівня.

Усі організації повинні проводити політику безпеки на високому рівні, визначаючи загальні цілі безпеки організації та встановлюючи рамки планів для досягнення цілей. Ці цілі на високому рівні істотно відрізняються від організації до організації. Військові організації приділяють велику цінність таємниці на відміну від академічних установ, які пропонують відкритість інформації. Економічні установи переймаються насамперед збереженням цілісності даних та меседжів, які представляють гроші. За замовчуванням для простих соціальних організацій взагалі немає політики безпеки.

Політики безпеки не завжди точно сформульовані або записані, але ефективна політика комп'ютерної безпеки вимагає відповіді на наступні питання:

- Які активи слід захищати та яка їхня вартість?
- Які загрози цим активам?
- Яку загрозу слід усунути та якими засобами?

Політика безпеки для розподіленої системи повинна відображати очікування вищих керівників щодо цілей безпеки організації. Часто так само, як цілі безпеки організації можуть бути не чітко визначені, ті, що знаходяться в розподіленому середовищі обробки, залишаються незмінними, і їх слід витягувати з інших документів або ідентифікувати та узгоджувати шляхом переконання та обговорення з працівниками відповідної організації.

Політика безпеки на високому рівні може зробити загальну заяву про цілі організації, але для її ефективності потрібно провести аналіз ризиків, щоб зрозуміти вразливість організації та наслідки порушень безпеки. Управління ризиками, про які йдеться нижче у розділі VC, потрібне через можливі компроміси між передбачуваною вартістю загроз та фактичними витратами на заходи безпеки. Заходи безпеки, що вживаються для протидії загрозі, повинні бути співмірними із самою загрозою. Th Результати е аналізу ризику може допомогти переглянути або зосередитися політики на високому рівні, а також визначити політику нижнього рівня для управління системою в безпечному режимі.

Вибір служб безпеки повинен узгодити низку конфліктуючих цілей, які включають наступне.

Політика безпеки часто визначається централізовано, але застосовується локально або в кожній програмі, а також у багатьох точках втручання в комунікацію між кожним юзерем. Тому існують практичні труднощі у забезпеченні дотримання політики безпеки.

Дизайн існуючих уніфікованих комунікаційних продуктів та багатьох операційних систем дозволяє уникнути чи нехтувати міркуваннями безпеки. Тому вимоги безпеки повинні узгоджуватись окремо з постачальниками системи, що вимагає великих затрат на реалізацію закупівель, ніж якщо б вони були визначені як частина уніфікованої або невід'ємної частини операційної системи.

Можна створити розподілену систему, в якій всі аспекти безпеки централізовано управляються загальним стандартом. Оскільки розподілена система, швидше за все, розвивається федерацією ряду існуючих різних (і неоднорідних) систем, можливо, раніше вони застосовували різні політики безпеки. Завжди можливо, що вони можуть бути несумісними або тому, що політики систем відрізняються рівнем безпеки, яку вони надають, або терплять, або тому, що існує технічна несумісність, наприклад, тому, що були обрані різні алгоритми зашифровування.

ISO визнали цю проблему і ввели концепцію політики взаємодії з безпекою як частину рамок безпеки. Це політика, яка приймається всіма учасниками взаємодії. Це має бути домовлено між ними, перш ніж вони зможуть спілкуватися. У питаннях , які повинні бути вирішені між ними як рівень безпеки і технічної сумісність їх механізмів безпеки. Що стосується рівня безпеки, це не обмежується параметрами безпеки, що стосуються їх зв'язку. Політика безпеки однієї організації може наполягати на тому, що сумісні стандарти безпеки діють на комп'ютерних підприємствах іншої організації до того, як буде дозволена взаємодія.

Міжорганізаційна політика взаємодії з безпекою, узгоджена та дотримана всіма сторонами, можливо буде складною для ведення переговорів через потрібність більш широкої сумісності, ніж просто стандартів безпеки зв'язку . Наприклад, можуть бути несумісності в рівнях безпеки їх операційних систем. Якщо не можна погодити загальну політику безпеки, можливо, буде прийнято рішення відмовитись від спілкування або через те, що ризик є

неприйнятним, або недійсним накладенням неприйнятних чи неконфесійних практик безпеки. Наприклад, більшість організацій, які працюють за своїми установками відповідно до державних стандартів військової безпеки, не дозволяють електронній пошті працювати на будь-якому з їх комп'ютерних мереж через відомі експозиції, пов’язані з цим. Вони повинні використовувати спеціальну вільну електронну пошту, відключену або завантажену від інших систем.

Політика взаемодії з безпекою в розподіленій системі повинна призвести до створення загальноприйнятого розкладу потрібних служб безпеки та їх механізмів підтримки.

Для ілютсрації практичного застосування політик безпеки інформаційних технологій (ІТ) в середовищі розподіленої обробки описуються деякі політики, які можуть застосовуватися до типової компанії. Вони поділяються на такі напрямки:

- політика управління безпекою;
- рівні безпеки;
- безпека зв'язку;
- системний управління доступом;
- управління доступом до даних;
- палнування катастроф;
- аудиторність системи;
- правова та регуляторна політика, що стосується безпеки.

Зверніть увагу насамперед на масштаби та межі цих політик. Вони включають сфери, потрібні для забезпечення конфіденційності, цілісності та доступності інформації, з двома помітними винятками. По-перше, вони не охоплюють процедури резервного копіювання та відновлення, які є частиною звичайних ІТ-операцій на сьогодні. Передбачається, що крім Політики безпеки ІТ, організація має Політику роботи з комп'ютером ІТ, яка охоплює щоденні

процедури керування комп'ютером, включаючи відновлення після інцидентів, таких як збої системи та збої дисків; а також Політика управління інформаційними комунікаціями, що охоплює такі процедури, як чергування у випадку відмов лінії. Виключення цих областей із політики безпеки є певною мірою довільним, але є загальним для багатьох організацій, які вважають за краще їх розглядати як аспекти системних та комунікаційних операцій. Звичайно, важливо забезпечити, щоб ці теми були охоплені тим чи іншим політичним документом.

По-друге, ці політики безпеки також виключають контроль змін системи. Це теж може розглядатися як довільне, але знову ж таки виправданням є те, що воно повинно бути охоплено в іншій політиці. Ці аспекти контролю змін, що стосуються реконфігурації системи шляхом зміни апаратних компонентів, таких як системи обробки та мережі, повинні охоплюватися інформаційними політиками комп'ютерних операцій та управління комунікаціями. Контроль за зміною програмного забезпечення повинен охоплюватися правилами з управління та розвитку комп'ютерних технологій.

У типовій комерційній організації безпека не зробить незначного або взагалі ніякого прямого внеску у досягнення своїх найближчих цілей (поки щось не піде не так) , витрачаючи при цьому значну кількість зусиль і коштів. Тому, якщо політика безпеки має бути ефективною, їх потрібно затвердити на найвищому рівні управління, як правило, на рівні Ради та включати в цілі, які встановлюються для кожного відділу. Тільки тоді менеджери підприємств розглядають їх як невід'ємну частину своїх цілей.

Далі, вони повинні ефективно повідомлятися. Багато організацій мають неофіційну політику безпеки, яку можна вивести з інших політичних документів та з управлінських рішень, які можуть бути поховані у внутрішніх меморандумах та їх важко знайти. Ефективні політики безпеки повинні бути окремо і чітко задокументовані, бажано як документ Політики безпеки або як

розділ у документі Політики IT. Тоді можливість IT та персоналу юзерів легко дізнатись, що таке політика; немає можливості ефективного здійснення політики, про яку ніхто не знає.

Отже, перша рекомендація компанії полягає в тому, щоб директор, відповідальний за IT, отримував схвалення Правління на створення та впровадження політики безпеки IT, як це викладено в документі Політики безпеки IT.

Основою політики безпеки організації буде ефективної організаційної структури. Хоча обробка інформації централізована, достатньо поставити когось, відповідального за забезпечення безпеки IT у всій організації. Однак , як тільки він розповсюджується, потрібно мати дворівневу організаційну структуру: центральний координатор безпеки; та Адміністратори безпеки, що охоплюють кожен відділ організації. Для дотримання самостійності розподілених систем відповідальний за безпеку в кожній системі повинен нести її адміністратор безпеки, але для того, щоб рівень безпеки був узгодженим у всій організації, Координатор безпеки повинен мати мету забезпечити, щоб кожна s ystem працює над сумісними стандартами та процедурами. Зазвичай Координатор безпеки має два завдання: забезпечення того, щоб кожен адміністратор безпеки знав про стандарти та процедури; і допомагати керівникам відомчих підприємств у забезпеченні їх дотримання.

Основна проблема компанії, окрім створення своєї організації IT-безпеки, повинна полягати у забезпеченні поширення мережі досить широко; кожна незалежна система, яка ухиляється від контролю зв'язку або управління системами, є потенційним ризиком для безпеки. Персональні комп’ютери, які стали «незалежними», можуть бути або не працювати відповідно до стандартів безпеки компанії. Кожен повинен бути внесений під парасолькою відповідного адміністратора безпеки.

1.10 Рівень безпеки

Політики управління повинні здійснюватися щодо груп об'єктів, а не окремих осіб. Більшість організацій повинні також застосувати цю концепцію до заходів безпеки. Якщо можна визначити лише кілька рівнів безпеки та встановити пакет заходів безпеки для кожного рівня, то детальних рішень щодо окремих об'єктів можна уникнути.

Комерційні організації, ймовірно, визначають два типи рівня безпеки: для даних та для юзерів. Більшість прагнуть забезпечити однаковий рівень захисту всіх своїх даних, щоб був лише один рівень безпеки для даних. Це набагато простіше в управлінні, ніж на декількох рівнях, і відповідає звичайній вимозі, що обмін та мобільність даних повинні бути включені, але контролюватися. Однак може виникнути вимога обробляти такі дані мені особливо безпечно, або тому, що її конфіденційність має вирішальне значення для успіху бізнесу, або, звичайно, через умови урядового контракту.

З іншого боку, досить часто хочеться робити розрізнення між категоріями юзерів, особливо коли стороннім особам надається обмежений доступ до системи для спеціальних цілей або коли дозволений безпечний набір номера дозволений. Таким чином, може існувати політика, що стосується трьох категорій юзерів по-різному: звичайних юзерів, які мають найменший доступ до розширеного відпочинку ; тих самих юзерів, коли вони набирають номер, яким повинен бути дозволений доступ до конкретно заздалегідь визначених даних; та сторонніх людей із подібним обмеженням. Зауважимо, що це поняття рівнів безпеки має схожість із рівнями безпеки військового типу для m та обов'язкової безпеки. Але формально визначене значно менше.

Фізична безпека є основою всієї безпеки системи. Повинна бути політика, яка вимагає відповідного рівня апаратного захисту всіх фізичних активів, що знаходяться під її контролем.

Політика безпеки зв'язку зазвичай визначає потрібні рівні конфіденційності, цілісності та доступності. Вони поділяються на дві категорії: безпека корпоративних мереж; а також критерії для додатків для забезпечення вищого рівня безпеки на завершення. Виходячи з сучасних технологій, політика, ймовірно, вимагає впевненої цілісності та визначеного відсотка доступності в мережі, але не наполягати на забезпеченій конфіденційності мережі. Протягом кількох років із збільшенням доступності дешевих продуктів зашифровування політика, ймовірно, буде вдосконалена, щоб також наполягати на конфіденційності мережі.

Припускаючи відносно слабку політику безпеки мережі, потрібна додаткова політика для того, щоб додатки, що потребують більш високого рівня безпеки, забезпечувались нею за допомогою заходів безпеки на рівні додатків. Зауважте, що політики автентифікації юзерів розглядаються нижче.

Часто найбільш важливі заходи безпеки повинні бути прийняті в організаціях з відкритими розподіленими системами робити з контролем доступу юзерів до системи. Тому потрібно розробити політику щодо контролю доступу до системи з двома основними частинами. У першій частині зазначено вимогу до унікальних ідентифікаторів юзерів та потрібність їх поваги. Друга визначає силу автентифікації, яку потрібно застосувати до юзерів, які намагаються увійти в систему. Зазвичай є два рівні автентифікації.

Звичайні юзері входять із терміналів та робочих станцій у приміщеннях ABC. У політиці будуть вказані стандарти для паролів, наприклад мінімальна довжина та потрібний формат, а також частота змін.

Юзері, незалежно від персоналу компанії чи зовнішні, які здійснюють реєстрацію за межами приміщень ABC. Буде набагато вищий рівень автентифікації, ймовірно, використовуючи смарт-карти чи інші генератори одноразових паролів.

Політика контролю доступу до даних містить такі елементи. Належить визначити право власності на всі елементи даних організації, при цьому власник несе відповідальність за рішення щодо застосування даних.

Усі дані повинні бути захищені, а доступ повинен бути дозволений лише за дозволу власника.

У всіх системах повинні бути системи управління процесом, які захищають дані до визначеного стандарту.

1.11 Управління безпекою

Управління безпекою – це активність управління функціями та механізмами, які використовуються різними службами в розподіленій системі для реалізації політики безпеки. Основні функції управління безпекою включають:

Керування ключами зашифровування (ANSI, X9.17) та іншими артефактами, таким як паролі. Це включає генерацію ключів у міру потрібності та розподіл ключів до відповідних компонентів системи, зберігання ключів та ключів архівації. Клавіші повинні мати обмежений термін експлуатації, і тому їх слід регенерувати в регулярні періоди.

Керування реєстрацією юзерів та інформацією, що використовується для перевірки їхньої унікальності (загальнодоступний ключ або пароль зашифровування).

Керування інформацією про управління доступом, що стосується юзерів та сервісів. Сюди входять списки контролю доступу, можливості, привілеї та багаторівневі мітки безпеки.

Забезпечення слідів аудиту безпеки. Вони фіксують усі виняткові події (спроби несанкціонованого доступу тощо) та вибрані звичайні події, такі як вхід у систему та доступ до файлів. Їх ціль – дозволити розслідування порушень безпеки та аудит дій адміністратора безпеки.

Практично неможливо гарантувати, що самовільний доступ чи перегляд чутливих матеріалів ніколи не відбуватиметься. У кращому випадку можливо обмежити потенційний розмір дозволеної групи доступу та розмістити її членів під юридичними чи договірними обмеженнями щодо застосування чи розкриття інформації. Якщо застосування законодавства серйозно розглядається як варіант, важливо підтримувати ефективний аудит доступу. Це само по собі означає потрібність високого рівня безпеки для механізму контролю доступу та пов’язаного з ним аудиту. Їх слід підтримувати за рахунок продуктивності та у випадку декількох відмов, якщо від них потрібно отримати якусь реальну цінність. Однак забезпечення запису є недостатнім; також повинен бути доступний швидкий та ефективний механізм аудиту.

1.12 Управління ризиками

Термін ризик часто застосовується стосовно інформаційних систем. Для розподілених систем потрібно з’ясувати як важливість ризику, так і актуальність управління ризиками . Потрібно розрізняти дві досить різні форми ризику; ризик безпеки та бізнес- ризик. Ризик безпеки пов'язаний з майбутніми подіями, в яких подія призводить лише до втрат. Прикладами є ризик втрати через шахрайство, порушення впевненості в собі та несправність обладнання. Цей тип ризику розглядається тут. Діловий ризик може спричинити за собою збиток або виграш і виникає в результаті звичайних управлінських рішень бізнесу. Ця стаття не висвітлена

Оцінка та управління ризиками – це оперування, яка раціонально враховує активи організації та ризики, з якими вони стикаються, а потім приймає рішення про захист, який їм слід надавати. Важливим елементом рішень є те, що витрати на захист повинні відповідати очікуваним витратам, що виникають внаслідок втрат безпеки. Існує кілька методологій управління

ризиками безпеки, пов'язаних з комп'ютером [5]. Всі вони мають спільні завдання:

– ідентифікація та оцінка активів;

– проти нитку до сценаріїв загроз безпеки;

– оцінка ймовірності сценаріїв та втрат, які можуть бути наслідком;

– визначення та вартість можливих заходів безпеки для кожного сценарію;

– вибір портфоліо заходів безпеки.

Аналіз ризиків є найбільш ефективним, коли він проводиться під час специфікації та проектування розробленої розподіленої системи. На цих етапах можуть бути встановлені ризикові наслідки проектних рішень (визначення місця, де проект показує найменш надійні характеристики) та вимоги безпеки та встановлення їх наслідків. Однак ця ідеальна політика рідко практична для розподілених систем, оскільки однією з їх характеристик є те, що вони часто виникають шляхом еволюційного процесу.

1.13 Стандарти безпеки

Існує три основні категорії стандартів безпеки, що стосуються розподілених систем. Перші – це стандарти, що стосуються безпеки окремих комп'ютерів, які існують певний час і є досить зрілими. Другі – це стандарти захисту передачі зв'язку та віддаленої автентифікації. Вони теж досить зрілі. По-третє, в процесі розробки – це ті, які інтегрують стандарти комп'ютерної та комунікаційної безпеки для забезпечення стандартів безпеки розподілених систем.

Стандарти, як правило, не передбачають фізичних чи процедурних механізмів, а також не встановлюють процедур управління ризиками чи оцінки ризиків або вимог щодо їх застосування. Це всі потрібні елементи, які слід врахувати та вирішити для ресурсів, що містять всю розподілену систему.

Тому, хоча стандарти безпеки є важливою підтримкою політики безпеки розподіленої системи, їх слід розглядати в контексті загальної політики безпеки, яка також використовує інші заходи.

Критерії оцінювання довірених комп'ютерних систем Міністерства оборони США (TCSEC – помаранчева книга) (Міністерство оборони (США), 1985), займається контролем доступу до окремих систем. Він визначає низку можливих рівнів, які можуть бути розміщені в системі, починаючи від сертифікованого високого рівня безпеки рівня A1 до низького рівня неофіційно визначеного рівня безпеки на рівні C1. Він зазвичай використовується як засіб індикації рівня безпеки, потрібного або забезпеченого в комп'ютерних системах.

Управління доступом поділяється цим стандартом на дві категорії: обов'язковий та дискреційний управління доступом. Попередні правила застосування, які вбудовані в дизайн системи і не можуть бути змінені, крім встановлення нової версії системи. Прикладом може слугувати політика, згідно з якою дані в багатошарових системах безпеки не можуть бути прочитані юзерем із нижчою класифікацією безпеки, ніж призначена для даних. Дискреційні механізми контролю доступу визначаються як такі, які дозволяють юзерам визначати та контролювати обмін ресурсами з іншими юзерами. Наприклад, дискреційна політика контролю доступу на рівні C2 визначається як вимагає механізмів, які забезпечують захист інформації та ресурсів від несанкціонованого доступу, а дозвіл на доступ призначається лише авторизованим юзерам.

Трактована мережева інтерпретація критеріїв оцінювання довіреної комп'ютерної системи (Міністерство оборони (США), 1987) (Червона книга) розширює критерії Червоної книги до мереж. Це головне питання щодо критеріїв безпеки, яким слід дотримуватися під час доступу до віддалених хостів.

Червона книга зараз досить стара, і вона завжди була більш орієнтована на безпеку військового типу, ніж на комерційну безпеку. Зараз впроваджуються уніфіковані критерії – критерії оцінювання інформаційної безпеки інформаційних технологій (ITSEC) (CEC, 1991). Це спільне зобов'язання урядів Великобританії, Нідерландів, Франції та Німеччини. Її метою є врахування потреб комерційних юзерів та вдосконалення Червоної книги шляхом відокремлення узгоджень щодо рівнів безпеки від способу оцінювання безпеки. У Великобританії Департамент торгівлі та промисловості та Група безпеки та зв'язку-електроніки створили Британську схему оцінювання та сертифікації безпеки IT (CESG, 1991), яка оцінює та сертифікує продукцію за критеріями ITSEC. Подібні зусилля ведуться і в інших країнах.

Стандарти безпеки передачі в основному стосуються методів зашифровування. Зокрема, один із алгоритмів став предметом уніфікованих зусиль: алгоритм DES для зашифровування секретного ключа, який є американським, але не міжнародним, стандартом. З іншого боку, алгоритм RSA для зашифровування відкритих ключів є предметом патентів США. Це стало фактичним стандартом криптографії з відкритим ключем, але через його запатентований статус наразі не визначається як національний або міжнародний стандарт. Ці два алгоритми були коротко описані у розділі II.E.2, вище.

Зашифровування залежить від його міцності від безпеки захищеного обладнання, яке використовується, і (BSI, 86/67937) описує стандарти апаратної безпеки криптографічного обладнання.

Основним стандартом для DES є (NBS, 46), доповнений (ANSI , X3.92). Існують додаткові стандарти, що описують його режими роботи (NBS, 81) та Керівні принципи встановлення та застосування (NBS, 74). Управління ключами в банківських додатках описано в (ANSI, X9.17), але цей стандарт досить виражений узагальнено і застосовуватиметься також і до інших

програм. Детальне обговорення стандартів DES знаходиться в (Davies & Price, 1989).

Розроблено низку стандартів для банківських програм для однорангового спілкування та автентифікації меседжів. Вони теж досить загальні за форматом і можуть бути використані для інших цілей. Вони включають (ANSI, X9.19) для автентифікації меседж та (ANSI, X3.118) для особистої автентифікації, використовуючи персональний ідентифікаційний номер (PIN).

Безпека мережі не викликала першочергових проблем, коли вперше почалися зусилля з відкритого системного взаємозв'язку (OSI) наприкінці 1970-х. Однак зараз розробляється низка стандартів ISO, які спрямовані на підвищення безпеки в ОС І. Стандарти визначають служби безпеки, з якими партнери в спілкуванні могли б погодитись, і протоколи, які будуть використовуватися для встановлення безпечної взаемодії.

Служби безпеки, які можуть знадобитися для засобів зв'язку, були визначені в ISO 7498-2 Архітектура безпеки (ISO, 7498-2). Протоколи їх надання досі знаходяться на стадії розробки та ще не доступні в продуктах OSI.

Багато інших заходів із стандартизації мають наслідки для безпеки, а отже, мають стандарти, пов'язані з безпекою, наприклад, в областях Каталог OSI, Управління системою OSI та Електронний обмін даними (EDI). Проводиться кілька подібних та пов’язаних із цим зусиль, включаючи профілі для опису характеристик безпеки вибраних програм.

1.14 Сервіс віддаленої автентифікації Kerberos як частина розподілених систем

Kerberos був спочатку розроблений для розподіленого обчислювального оточення, яке MIT розгорнуло у 1980-х роках як Project Athena. Понад два

десятиліття Kerberos, згідно з Інтернет-стандартами, є старшою технологією. Щоб поставити речі в перспективу, Kerberos і DNS з'явилися приблизно в один і той же час. Що стосується безпеки, то зрілість є надзвичайно бажаною: відкриті, широко використовувані та широко вивчені технології є найбільш передбачуваними безпечними і найменш схильні до нових подвигів.

Як і DNS, Kerberos вийшов із чіткої уваги на чітко визначене завдання. У таких оточеннях, як Ath ena, що є звичним сьогодні, але новим часом, від окремого юзера можна очікувати застосування багатьох різних робочих станцій (і багатьох різних сервісів, таких як доступ до файлів та друк, розміщених на багатьох різних сервісах).

Оригінальним питанням, що постало перед розробниками Kerberos-a, було те, як забезпечити єдиний вхід : дозволити юзерам отримувати доступ до різноманітних систем і служб, не потребуючи введення свого ідентифікатора юзера та пароля неодноразово (або, що ще гірше, без потрібності запам’ятовувати та надавати різні ідентифікатори юзерів та паролі для кожної з різних систем та сервісів, які вони використовують).

Хоча оригінальний Kerberos був розроблений для єдиного оточення, він базувався на чіткій архітектурній моделі, застосовній до інших середовищ. Через цю міцну архітектурну основу, Kerberos, як і DNS, міг зростати та підтримувати масштаб та широту функцій, які мало можна уявити під час свого первісного створення. Сьогодні Kerberos забезпечує не тільки єдиний вхід, він також забезпечує надійну загальну основу f або безпечну автентифікацію у відкритих розподілених системах.

Криптографічно безпечний, архітектурно звуковий і легко інтегрується як компонент в інші системи, Kerberos широко сприймається як спосіб надання основного набору служб безпеки для багатьох розроблених системних проектів і розробок, і сьогодні є невід'ємною частиною багатьох обчислювальних середовищ. Майже у всіх популярних операційних системах (ОС) вбудований Kerberos, як і у багатьох важливих програм, і він широко

використовується постачальником мережевого обладнання. Як і DNS, Kerberos – це сервіс, який більшість юзерів навіть не розуміють, що вони користуються, але він дозволяє багато взаємодій, що відбуваються в сучасних корпоративних мережах.

РОЗДІЛ 2
ТРИСТОРОННІЙ ПРОТОКОЛ АУТЕНТИФІКАЦІЇ

Аутентифікація – це процес ідентифікації себе в мережі і є основоположним для безпеки комп'ютерних систем. Не знаючи, хто вимагає операції, важко вирішити, чи слід дозволити операцію. Слабкі системи аутентифікації є автентифікацією за твердженням і припускають, що сервіси та машини не можуть бути порушені або підроблені, і що мережевий трафік не може контролюватися. Сильні системи аутентифікації, які не розкривають секрети в мережі та використовують шифрування, стають все більш популярними та важливими.

Усі веб-сайти інформатики використовували слабку автентифікацію, де паролі для входу та програми (такі як поштові інструменти) проходили в чистому тексті від клієнта до сервера по всій мережі. Цей вид слабкої аутентифікації є дуже поширеним явищем і застосовується протягом багатьох років у більшості установок UNIX. Однак він абсолютно непридатний для автентифікації користувачів у ненадійних середовищах, що тут створюється все більше використання портативних та самокерованих машин.

Наша вимога об'єднати старі простори користувачів з уже існуючих адміністративних доменів в єдиний простір користувачів для всієї інформатики означає, що нові процедури управління обліковими записами повинні були розробити з оглядом моделі безпеки. Обмін послугами веб-сайтами через мережі, якими ми безпосередньо не керуємо, та підтримка більш періодично підключених та самокерованих машин означає, що є ще більше причин відійти від машинного та мережевого довіри; і ми вже не можемо реально потурати подальшому використанню слабкої аутентифікації. Тому необхідно шукати альтернативні технології та інфраструктуру.

Сервер Kerberos має ряд переваг.

Сильна взаємна автентифікація. Секретні дані не передаються через мережу. Дані критичної аутентифікації шифруються. Клієнт (як правило, користувач) автентифікований на сервері, а сервер – на клієнті. Ідентифікація клієнта використовується для авторизації служб на сервері. Ідентифікація сервера запобігає підробці та захопленню служб.

Одноразова реєстрація. Зручність користувача, що означає одне ідентифікатор та пароль, може використовуватися для багатьох (в основному всіх, якщо ядерних) шкільних служб (а також, можливо, університету з підтримкою міжрегіональної сфери) лише з однією послідовністю входу.

Не існує реальної альтернативи Kerberos для сильної аутентифікації, за винятком використання інфраструктури відкритого ключа (PKI). Однак PKI є відносно новою технологією, і мало того, що є достатньо зрілим, щоб йому можна було довіряти, не кажучи вже про розгортання та розповсюдження як підтримувану виробничу систему. Проводиться робота по додаванню підтримки відкритого ключа до стандарту Kerberos. Одноразові паролі є занадто незручними для користувача, щоб бути реальною внутрішньою альтернативою. Наявність центральної служби Kerberos, яка також може автентифікувати клієнтів Windows, була б дуже бажаною.

2.1 Основні бізнес-вимоги до віддаленого сервера автентифікації

Для ефективної побудови системи, в тому числі і розподіленої комп'ютерної, завжди потрібно чітко сформулювати і довести до всіх зацікавлених сторін набір вимог до цієї системи. Це традиційно функціональні вимоги, котрі описують що саме повинна робити система, так і нефункціональні, котрі містять опис того, як повинні реалізовуватись функціональні вимоги та середовище, в котрому працює система. Зрозуміло, що правильно сформульовані вимоги є визначальним етапом для успішної реалізації проекту у тому вигляді, в якому його прийме користувач.

Прикладами невдало опрацьованих вимог в області розподілених комп'ютерних систем є NFS, WEP та взаємодія венб браузера з сервером.

Коли говорити про Керберос, то при створенні цього продукту архітектори відповідально підійшли до формування вимог, дотримались усіх необхідних процедур при цьому для вирішення проблем при роботі з цими вимогами, опрацювавши кожну з них. Хоча на момент створення цього сервісу деякі з тих вимог могли здатись надлишковими та неактуальними. На сьогодні такий підхід також себе виправдовує при проектуванні програмних продуктів в тому числі для локальних та розподілених мереж.

У цьому розділі подано визначення об'єктів і термінів, знання яких є важливими для подальшого опису протоколу Kerberos. Оскільки багато визначень базуються на інших, коли це можливо, я намагався привести їх у порядок, щоб значення терміна не було задано перед його визначенням. Однак, можливо, буде потрібно прочитати цей розділ двічі, щоб повністю зрозуміти всі умови.

Царство (Realm).

Термін realm позначає адміністративний домен автентифікації. Її метою є встановлення меж, у межах яких сервер автентифікації має повноваження щодо автентифікації користувача, хоста чи послуги. Це не означає, що автентифікація між користувачем і службою, що вони повинні належати одній царині: якщо два об'єкти є частиною різних областей і між ними існує довірчі відносини, тоді може відбутися аутентифікація. Ця характеристика, відома як перехресна автентифікація, буде описана нижче.

В основному, користувач / послуга належить до сфери, якщо і тільки якщо він / вона ділиться секретом (паролем / ключем) з сервером аутентифікації цього царства.

Ім’я області є чутливим до регістру, тобто є різниця між великими і малими літерами, але зазвичай області завжди відображаються в великих літерах. Також в організації добре застосовувати ім’я області так само, як і

домен DNS (хоча великими літерами). Дотримуючись цих порад під час вибору імені царства, значно спрощується конфігурація клієнтів Kerberos, перш за все, коли потрібно встановити довірчі відносини з субдоменами. В якості прикладу, якщо організація належить до домену DNS example.com, доречно, щоб пов'язана сфера Kerberos була EXAMPLE.COM.

Ролі (principal).

Як principal в сервісі Kerberos є ім'я, яке використовується для позначення записів у базі даних сервера аутентифікації. Principal пов'язаний з кожним користувачем, хостом або сервісом даної сфери. Основний Principal у Kerberos 5 має такий тип:

component1/component2/.../componentN@REALM

Однак на практиці використовується максимум два компоненти. Для запису, що стосується користувача, основним є наступний тип:

Name[/Instance]@REALM

Екземпляр не є обов'язковим і зазвичай використовується для кращого визначення типу користувача. Для прикладу користувачі адміністратора зазвичай мають екземпляр адміністратора. Нижче наведено приклади принципів, які стосуються користувачів:

pippo@EXAMPLE.COM admin/admin@EXAMPLE.COM pluto/admin@EXAMPLE.COM

Якщо натомість записи стосуються послуг, принципали приймають таку форму:

Service/Hostname@REALM

Перший компонент – це назва служби, наприклад imap, AFS, ftp. Часто саме слово хост використовується для позначення загального доступу до машини (telnet, rsh, ssh). Другий компонент – це повне ім'я хоста (FQDN) машини, що надає запитувану послугу. Важливо, щоб цей компонент точно відповідав (малі літери) зворотній роздільній здатності DNS IP-адреси сервера додатків. Нижче наведено дійсні приклади принципів, що стосуються сервісів:

imap/mbox.example.com@EXAMPLE.COM
host/server.example.com@EXAMPLE.COM
afs/example.com@EXAMPLE.COM

Слід зазначити, що останній випадок є винятком, оскільки другий компонент – це не ім’я хоста, а назва клітини AFS, на яку посилається головний. Нарешті, є принципи, які не посилаються на користувачів чи сервіси, але відіграють певну роль в роботі системи аутентифікації. Загальний приклад – krbtgt / REALM @ REALM з його пов’язаним ключем використовується для шифрування квитка, що надає квиток (ми розглянемо це пізніше).

У Kerberos 4 ніколи не може бути більше двох компонентів, і вони розділені символом "." Замість "/", тоді як ім'я хоста в головних директорах, що стосуються послуг, є коротким, тобто не FQDN. Нижче наведені реальні приклади:

pippo@EXAMPLE.COM pluto.admin@EXAMPLE.COM imap.mbox@EXAMPLE.COM

Квиток.

Квиток – це те, що клієнт подає серверу додатків, щоб продемонструвати справжність своєї особи. Квитки видаються сервером аутентифікації та шифруються за допомогою секретного ключа послуги, для якої вони

призначені. Оскільки цей ключ є секретом, який ділиться лише між сервером аутентифікації та сервером, що надає послугу, навіть клієнт, який запитав квиток, не може його знати або змінити його вміст. Основна інформація, що міститься в квитку, включає:

Основний запитуючий користувач (як правило, ім'я користувача);

Основний сервіс, для якого він призначений;

IP-адреса клієнтської машини, з якої можна використовувати квиток. У Kerberos 5 це поле є необов’язковим і може також бути багаторазовим, щоб мати змогу запускати клієнтів під NAT або мультихомедом.

Дата та час (у форматі часових позначок), коли розпочинається термін дії квитків;

Максимальний термін експлуатації квитка.

Ключ сеансу (це має принципову роль, описану нижче);

Кожен квиток має термін дії (як правило, 10 годин). Це важливо, оскільки сервер автентифікації більше не контролює вже виданий квиток. Незважаючи на те, що адміністратор царства може в будь-який час запобігти видачі нових квитків певному користувачеві, він не може перешкодити користувачам використовувати ті квитки, які вони вже мають. Це причина обмеження терміну експлуатації квитків, щоб обмежити зловживання з часом.

Квитки містять багато іншої інформації та прапорів, що характеризують їхню поведінку, але ми тут не будемо вникати в це. Ми знову обговоримо квитки та прапори, побачивши, як працює система аутентифікації.

Шифрування.

Як видно з описаного вище Kerberos часто потребує шифрування та розшифрування повідомлень (квитків та автентифікаторів), що передаються між різними учасниками аутентифікації. Важливо зазначити, що Керберос використовує лише симетричне шифрування ключа (іншими словами, той самий ключ використовується для шифрування та дешифрування). Деякі проекти (наприклад, pkinit) активні для впровадження системи відкритих

ключів, щоб отримати початкову автентифікацію користувача шляхом представлення приватного ключа, що відповідає сертифікованому відкритому ключу, але оскільки не існує стандарту, ми зараз пропустимо це обговорення.

Тип шифрування.

Kerberos 4 реалізує єдиний тип шифрування, який становить DES у 56 біт. Слабкість цього шифрування та інші вразливості протоколу зробили Kerberos 4 застарілим. Версія 5 Kerberos, однак, не визначає кількість або тип підтримуваних методологій шифрування. Завданням кожної конкретної реалізації є підтримка та найкраще узгодження різних типів шифрування. Однак ця гнучкість та розширюваність протоколу підкреслила проблеми інтероперабельності між різними реалізаціями Kerberos 5. Для того, щоб клієнти та сервери додатків та аутентифікації використовували різні реалізації для взаємодії, вони повинні мати принаймні один тип шифрування спільного. Класичність, пов'язана з взаємодією між реалізаціями Unix Kerberos 5 та наявними в Active Directory Windows, є класичним прикладом цього. Дійсно, Windows Active Directory підтримує обмежену кількість шифрів і має лише DES у 56 біт спільного з Unix. Це вимагало збереження останнього увімкненим, незважаючи на загальновідомі ризики, якщо потрібно було забезпечити сумісність. Згодом проблема була вирішена з версією 1.3 MIT Kerberos 5. Ця версія представила підтримку RC4-HMAC, яка також присутня в Windows і більш безпечна, ніж DES. Серед підтримуваних шифрів (але не від Windows) варто відзначити потрійний DES (3DES) та новіші AES128 та AES256.

Ключ шифрування.

Як було сказано вище, однією з цілей протоколу Kerberos є запобігання збереженню пароля користувача у незашифрованому вигляді навіть у базі даних сервера аутентифікації. Зважаючи на те, що кожен алгоритм шифрування використовує власну довжину ключа, зрозуміло, що якщо користувач не буде змушений використовувати інший пароль фіксованого

розміру для кожного підтримуваного методу шифрування, ключі шифрування не можуть бути паролями. З цих причин була введена функція string2key, яка перетворює незашифрований пароль в ключ шифрування, відповідний типу використовуваного шифрування. Ця функція викликається кожного разу, коли користувач змінює пароль або вводить його для автентифікації. Ключ string2 називається хеш-функцією, це означає, що вона незворотна: враховуючи, що ключ шифрування не може визначити пароль, який її створив (якщо тільки з грубої сили). Відомі алгоритми хешування – MD5 та CRC32.

Сіль.

У Kerberos 5, на відміну від версії 4, було введено поняття парольної солі. Це рядок, який слід з'єднати з незашифрованим паролем перед застосуванням функції string2key для отримання ключа. Kerberos 5 використовує той самий головний користувач, що і сіль:

$$K_{pippo} = string2key\ (P_{pippo} + "pippo@EXAMPLE.COM")$$

Kpippo – ключ шифрування користувача pippo, а Ppippo – це незашифрований пароль користувача.

Цей вид солі має такі переваги.

Два директори, що належать до однієї сфери і мають однаковий незашифрований пароль, все ще мають різні ключі. Наприклад, уявіть собі адміністратора, який має головного керівника для повсякденної роботи (pippo@EXAMPLE.COM) та адміністративного (pippo /admin@EXAMPLE.COM). Цілком імовірно, що цей користувач встановив однаковий пароль для обох директорів з міркувань зручності. Наявність солі гарантує, що пов'язані ключі відрізняються.

Якщо користувач має два облікові записи в різних царинах, досить часто, що незашифрований пароль є однаковим для обох областей: завдяки наявності

солі можливий компроміс облікового запису в одній царині не призведе автоматично до іншого піддаються компрометації.

Нульова сіль може бути налаштована на сумісність з Kerberos 4. Навпаки, для сумісності з AFS можна налаштувати сіль, яка не є повною назвою головного, а просто ім'ям комірки.

Обговоривши концепції типу шифрування, string2key та солі, можна перевірити точність наступного спостереження: для того, щоб була взаємодійна між різними реалізаціями Kerberos, недостатньо домовитись про загальний тип шифрування, але також потрібно використовувати однакові типи string2key та сіль.

Важливо також зазначити, що, пояснюючи поняття string2key і сіль, посилання робилися лише на головних користувачів, а ніколи на сервери. Причина зрозуміла: сервіс, навіть якщо він містить секрет із сервером аутентифікації, – це не незашифрований пароль (хто його вводить?), А ключ, який після створення адміністратором на сервері Kerberos запам'ятовується. як на сервері, що надає послугу.

Номер ключа версії (kvno).

Коли користувач змінює пароль або адміністратор оновлює секретний ключ для сервера додатків, ця зміна реєструється шляхом просування лічильника. Поточне значення лічильника, що ідентифікує ключову версію, відоме як номер версії ключа або, коротше, kvno.

Центр розподілу ключів (KDC).

Ми загалом говорили про сервер автентифікації. Оскільки це основний об'єкт, що бере участь в аутентифікації користувачів та сервісів, ми зараз розглянемо його більш глибоко, не вдаючись до всіх деталей його роботи, які натомість є темою розділу, присвяченого роботі протоколу.

Сервер аутентифікації в середовищі Kerberos, заснований на його функції розподілу квитків для доступу до сервісів, називається Центром розподілу ключів або коротше KDC. Оскільки він повністю розташований на

одному фізичному сервері (він часто збігається з одним процесом), його логічно можна розділити на три частини: база даних, сервер автентифікації (AS) та сервер надання квитків (TGS). Давайте коротко розглянемо їх.

Примітка: сервер можна зробити надмірним у межах сфери Master / Slave (MIT і Heimdal) або Multimaster (Windows Active Directory). Як отримати надмірність, протокол не описується, але залежить від використовуваної реалізації і тут не піде мова.

База даних.

База даних є контейнером для записів, пов'язаних з користувачами та службами. Ми посилаємось на запис, використовуючи головний (тобто назву запису), навіть якщо часто термін "головний" використовується як синонім запису. Кожен запис містить таку інформацію:

- основна особа, з якою пов'язаний запис;
- ключ шифрування та пов'язане з ним квно ;
- максимальна тривалість дії квитка, пов’язаного з довірителем;
- максимальний час, коли квиток, пов’язаний з довірителем, може бути поновлений (лише Керберос 5);
- атрибути або прапори, що характеризують поведінку квитків;
- дата закінчення терміну дії пароля.

Дата закінчення терміну дії довірителя, після чого квитки не видаватимуться.

З метою ускладнення викрадення ключів, що знаходяться в базі даних, реалізації шифрують базу даних за допомогою головного ключа, який асоціюється з головним K/M@REALM. Навіть будь-які скиди баз даних, що використовуються як резервні копії або для поширення від ведучого KDC до веденого, шифруються за допомогою цього ключа, який необхідно знати, щоб перезавантажити їх.

Сервер аутентифікації (AS).

Сервер аутентифікації – це частина KDC, яка відповідає на початковий запит аутентифікації від клієнта, коли користувач, ще не підтверджений автентифікацією, повинен ввести пароль. У відповідь на запит на аутентифікацію, AS видає спеціальний квиток, відомий як Білет, що надає квиток, або коротше TGT, основним асоційованим з яким є krbtgt / REALM @ REALM. Якщо користувачі насправді є тими, ким вони кажуть, що вони є (і ми побачимо згодом, як вони це демонструють), вони можуть використовувати TGT для отримання інших службових квитків, не потребуючи повторного введення пароля.

Сервер видачі квитків (TGS).

Сервер надання квитків – це компонент KDC, який розповсюджує службові квитки клієнтам з дійсним TGT, гарантуючи справжність особи для отримання запитуваного ресурсу на серверах додатків. TGS можна розглядати в якості сервера додатків (за умови, що доступ до нього, що необхідно представити TGT), який забезпечує видачу службових квитків в якості послуги. Важливо не плутати абревіатури TGT і TGS: перший вказує квиток, а другий – послугу.

Ключ сесії.

Як ми бачили, користувачі та сервіси діляться секретом із KDC. Для користувачів цей секрет – ключ, отриманий з їхнього пароля, тоді як для служб – це їх секретний ключ (встановлений адміністратором). Ці клавіші називаються довгостроковими, оскільки вони не змінюються при зміні робочого сеансу.

Однак необхідно, щоб користувач також ділився секретом із сервісом, принаймні на час, коли клієнт відкрив робочий сеанс на сервері: цей ключ, згенерований KDC при видачі квитка, називається Ключ сесії. Копія, призначена для послуги, огортається KDC у квитку (у будь-якому випадку їх сервер додатків знає довгостроковий ключ і може розшифрувати його та витягнути сесійний ключ), тоді як призначена для користувача копія

інкапсульована у зашифрованому пакеті за допомогою довгострокового ключа користувача. Ключ сеансу відіграє принципову роль у демонстрації справжності користувача, яку ми побачимо у наступному параграфі.

Аутентифікатор.

Навіть якщо головний користувач присутній у квитку і лише сервер додатків може витягти та, можливо, керувати такою інформацією (оскільки квиток зашифрований секретним ключем служби), цього недостатньо, щоб гарантувати справжність клієнта. Самозванець міг захопити (пам'ятайте гіпотезу про відкриту та незахищену мережу) квиток, коли він легітимним клієнтом він надсилає на сервер додатків, і у відповідний час надіслати його, щоб отримати незаконну послугу. З іншого боку, в тому числі IP-адреси машини, з якої це можливо використовувати, не дуже корисно: відомо, що у відкритій і незахищеній мережі адреси легко підробляються. Щоб вирішити проблему, потрібно використати той факт, що клієнт і сервер, принаймні під час сеансу, мають спільний ключ сеансу, який вони знають тільки (також KDC знає його, оскільки він його створив, але це довіряється за визначенням! !!). Таким чином, застосовується наступна стратегія: поряд із запитом, що містить квиток, клієнт додає ще один пакет (автентифікатор), куди додаються головна та часова марка користувача (його на той момент) і шифрується за допомогою сеансового ключа; сервер, який повинен запропонувати послугу, отримавши цей запит, розпаковує перший квиток, витягує сеансовий ключ і, якщо користувач насправді є тим, кого він / вона каже, сервер може розшифрувати автентифікатор, витягуючи позначку часу. Якщо останній відрізняється від часу сервера менш ніж на 2 хвилини (але допуск можна налаштувати), то аутентифікація проходить успішно. Це підкреслює критичність синхронізації між машинами, що належать до однієї сфери.

Кеш відтворення.

Існує можливість для самозванця одночасно викрасти і квиток, і автентифікатор, і використовувати їх протягом двох хвилин, коли

автентифікатор дійсний. Це дуже важко, але не неможливо. Для вирішення цієї проблеми з Kerberos 5 було введено кеш Replay. На серверах прикладних програм (але також і в TGS) існує здатність запам’ятовувати автентифікатори, які надійшли протягом останніх 2 хвилин, і відхиляти їх, якщо вони є репліками. З цим проблема вирішується до тих пір, поки самозванець не буде достатньо розумним, щоб скопіювати квиток та автентифікатор і змусити їх прибути на сервер додатків до того, як надійде законний запит. Це дійсно було б прихисткою, оскільки справжній користувач буде відхилений, тоді як самозванець отримає доступ до послуги.

Кеш довіри.

Клієнт ніколи не зберігає пароль користувача, а також не запам'ятовує секретний ключ, отриманий шляхом застосування string2key: вони використовуються для розшифрування відповідей від KDC та негайно відкидаються. Однак, з іншого боку, щоб реалізувати характеристику єдиного входу (SSO), коли користувачеві пропонується ввести пароль лише один раз на робочому сеансі, необхідно запам'ятати квитки та пов'язаний з ним сеансовий ключ. Місце, де зберігаються ці дані, називається "кеш-пам'ять". Де потрібно розмістити цей кеш, не залежить від протоколу, але варіюється від однієї реалізації до іншої. Часто для цілей портативності вони розташовані у файловій системі (MIT та Heimdal). В інших реалізаціях (AFS та Active Directory), щоб підвищити безпеку у випадку вразливих клієнтів, кеш-код облікових даних розміщується в області пам’яті, доступній лише ядрам і не змінюється на диску.

2.2 Модель для аутентифікації та авторизації Kerberos

Керберос вирішує кожну з основних проблем, виявлених традиційною моделлю аутентифікації, хоча за рахунок того, що вона є значно складнішою, ніж традиційна.

На огляді модель аутентифікації Kerberos використовує один або кілька надійних серверів аутентифікації (KDC або "центри розподілу ключів") для надання сторонніх послуг аутентифікації для співпрацюючих систем та додатків. У моделі Kerberos клієнтські машини набувають облікові дані автентифікації (називаються квитки) від надійного сервера (-ів) аутентифікації, який вони можуть згодом представити системам і програмам як доказ автентифікації і які, завдяки їх сильному зашифрованому, можуть бути надійно передані незахищена мережа.

Типовий сеанс Kerberos починається, коли користувач запускає програмне забезпечення на своєму локальному клієнтському апараті для придбання вихідного квитка аутентифікації (який називається квитком, що надає квиток) Пізніше клієнт подає користувачеві квиток на надання квитка сервісу надання квитків Kerberos для придбання сервісного квитка для конкретної системи або програми, яку користувач бажає використовувати. Потім цей сервісний квиток подається до потрібної послуги замість пари [ім'я користувача, passwd] як підтвердження автентичності.

Деталі розмов між клієнтом Kerberos, KDC і різними сервісами Kerberized, які використовуються клієнтом, досить складні. На малюнку III нижче графічно зображено взаємодію між системами співпраці, що беруть участь у моделі Kerberos.

Повністю архітектуру сервера та деталі протоколу буде описано далі у ці роботі. Зараз же підкреслимо основні моменти і спосіб функціонування, які показані на рис. 2.1.

Аврхітектура сервера побудована на принципі обміну повідомленнями, котрими обмінюються три сутності:

1. Користувачі сервісів (клієнти).

2. Сервери я к провайдери цих послуг. Нагадаємо, що і клієнти і сервери називаються в даній предметній області Principal.

3. Сервер(и), котрий (-і) реалізують сам протокол автентифікації. Нагадаємо, що їх ще називають також "KDC" (центри розподілу ключів) і вони складаються з декількох компонентів, кожен з котрих надає свій тип послуги у процесі автентифікації.

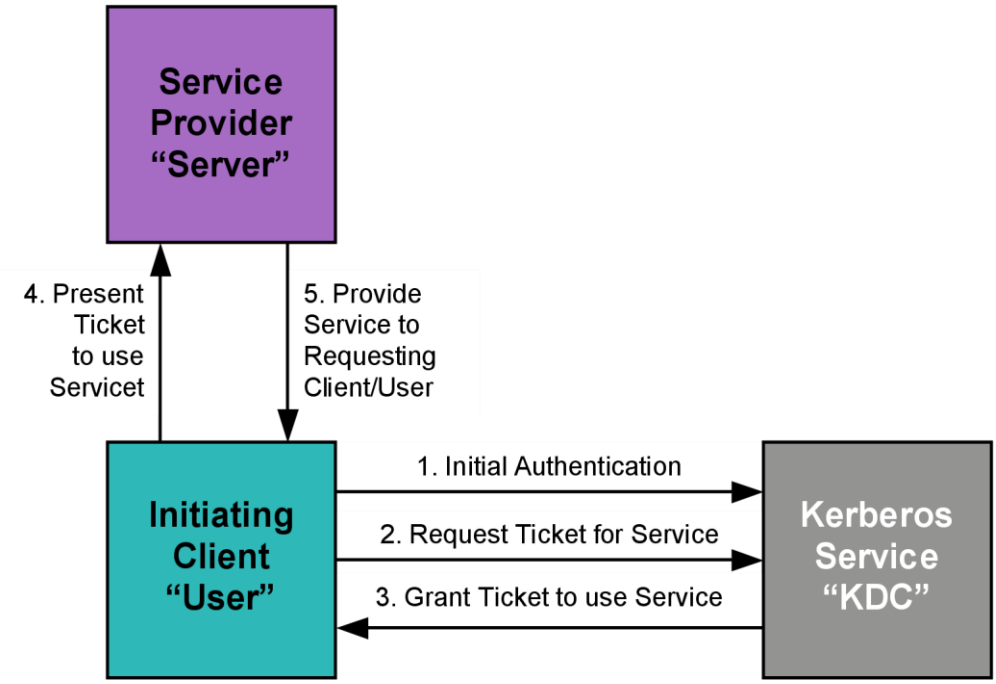


Рисунок 2.1 – Спрощена схема роботи Kerberos

Клієнти та сервери виктонують автентифікацію через виконання протоколу, який передбачає обмін так званими тікетами (квитками): це криптографічно захищені структури даних, котрі містять часові мітки з даними про учасників процесу автентифікації.

Крок 1:

Користувачі входять у систему та запитують послуги на хості. Таким чином, запит користувача на надання квитків-послугу.

Крок 2:

Сервер аутентифікації перевіряє право доступу користувача за допомогою бази даних, а потім надає ключ про надання квитків та ключ сесії. Результати шифруються за допомогою пароля користувача.

Крок 3:

Розшифровка повідомлення здійснюється за допомогою пароля, після чого надсилайте квиток на сервер надання квитків. Білет містить автентифікатори, такі як ім’я користувача та мережева адреса.

Крок 4:

Сервер видачі квитків розшифровує надісланий Користувачем квиток, а автентифікатор підтверджує запит, а потім створює квиток для запиту послуг від Сервера.

Крок 5:

Користувач надсилає Квиток та Аутентифікатор на Сервер.

Крок-6:

Сервер перевіряє Квиток, а автентифікатори потім генерують доступ до послуги. Після цього Користувач може отримати доступ до послуг.

2.3 Стандарти розробки сервера Kerberos

Очевидний факт того, що протокол автентифікації Kerberos широко використовується на сьогодні, доказує, що підходи, реалізовані у протоколі були правильними і ефективними. Важливим залишається також те, що цей протокол має реалізації на різних програмних платформах і, таким чином, може використовуватись для інтеграції служб корпорації, які працюють в різних операційних системах.

Базова специфікація Kerberos (RFC 4120) визначає основні обміни повідомлень Kerberos. Усі програми Kerberos покладаються на обмін повідомленнями KDC, визначений в RFC 4120. Крім того, повідомлення автентифікації додатків, зазначені в цьому RFC, використовуються в певній формі майже всіма ліцензіями додатків Kerberos .

RFC 4120 також визначає два повідомлення про цілісність та конфіденційність даних користувачів. Повідомлення KRB_SAFE забезпечує

захищену цілісністю обгортку навколо даних користувача. Повідомлення KRB_PRIV шифрується та цілісність захищає дані користувача. Однак за межами деяких спеціальних додатків ці повідомлення рідко використовуються – найкраща практика полягає у використанні альтернатив на основі GSS-API, а не повідомлень KRB_PRIV або KRB_SAFE.

Повідомлення KRB_SAFE є особливо проблематичним. Помилка у ранніх версіях MIT Kerberos видала неправильно відформатоване повідомлення KRB_SAFE. Деякі реалізації очікують версії повідомлення, виробленого MIT Kerberos; інші очікують версії в стандарті. Як результат, оперативна сумісність KRB_SAFE обмежена. Оскільки повідомлення настільки рідко використовується у нас , воно не було широко перевірене.

Крім того, не існує стандартних API для доступу до цих повідомлень. Станом на Windows Vista, Microsoft не надає механізму, щоб користувацькі програми створювали KRB_PRIV або KRB_SAFE повідомлення. Solaris до оновлення версії 10 не надав доступ до необмеженої програми Kerberos, необхідної для використання цих повідомлень.

Сирі повідомлення Kerberos дійсно забезпечують деяку гнучкість, якої немає в інших інтерфейсах. Програми, яким потрібно безпосередньо отримати доступ до клавіш Kerberos, або додати поля, які не знайдені в інших інтерфейсах, можливо, повинні використовувати необроблені повідомлення.

Механізми GSS-API: GSS-API і SSPI.

GSS-API (RFC 2743) – незалежний від механізму механізм, що дозволяє програмам запитувати послуги безпеки, такі як аутентифікація, цілісність та конфіденційність. Специфікації GSS-API орієнтовані на API, який можуть використовувати програми.

Однак є також ряд мережевих протоколів, пов'язаних з GSS-API. Кожен механізм має пов'язаний мережевий протокол. Kerberos GSS-API мені chanism (RFC 4121) описує повідомлення , які реалізують служби безпеки GSS-API з інфраструктурою Kerberos.

Механізм GSS-API використовує обмін аутентифікацією додатків RFC 4120 з деяким додатковим обрамленням, щоб вказати, що це повідомлення Kerberos і для надання деяких специфічних параметрів GSS-API. Механізм Kerberos GSS-API надає послуги, схожі на KRB_SAFE та KRB_PRIV, але формати повідомлень по мережі абсолютно інші.

GSS-API доступний на більшості платформ сьогодні. Однак , навіть якщо GSS-API недоступний, механізми GSS-API все ще можуть бути використані. Формати повідомлень з механізму Kerberos можуть бути вбудовані в невелику вбудовану систему без складності та кодового простору повного GSS-API.

Існує кілька переваг використання механізму GSS-API Kerberos:

– Хороша доступність для платформ, включаючи Windows, Mac OS, Java та більшість варіантів Unix.

– Хороша сумісність між реалізаціями.

– Підтримка майбутньої розширюваності в межах Kerberos та інших технологій безпеки.

GSS-API підтримує механізми захисту, які беруть довільну кількість зворотних поїздок для аутентифікації клієнта до послуги. Це дозволяє підтримувати переговори щодо того, яку службу безпеки h використовувати, а також майбутню розширюваність, оскільки варіанти додаються до механізмів. В результаті програми GSS-API демонструють характерний цикл для аутентифікації додатків. Клієнт генерує перше повідомлення, викликаючи GSS_Init_sec _context. Клієнт відправляє це повідомлення на сервер, який викликає GSS _Accept _sec _context. і GSS _Init _sec _context, і GSS _Accept _sec _context надають повернене значення у вигляді основного стану. Цей головний статус вказує додатку, що робити tx t. Якщо функція повертає GSS_S_CONTINUE_NEEDED, очікується, що інша сторона з'єднання поверне ще одне повідомлення, і цикл повинен продовжуватися. В іншому випадку петля закінчується; залежно від того, успішний статус чи ні, тобто

автентифікаційний іон успішний чи невдалий. Навіть якщо функція аутентифікації не повертається, "продовжувати потрібно", вона все одно може виводити повідомлення, яке надсилається іншій стороні.

Деякі протоколи додатків залежать від того, скільки знає кількість аутентифікації в обидва кінці та безпека, яка потребує etup. GSS-API все ще слід використовувати в цих додатках, але додаток має обмежувати набір підтримуваних механізмів. Механізм Кербероса завжди здійснює одну поїздку з можливістю взаємної автентифікації. Якщо в майбутньому розширення на механізм Kerberos збільшить кількість туди-поїздок, то або ідентифікатор об'єкта механізму (OID) зміниться (створивши новий механізм), або додаткові кругові рейси будуть використовуватися лише тоді, коли включена нова необов'язкова функція.

Після успішної початкової аутентифікації програми GSS-API можуть використовувати GSS_Wrap для запиту цілісності або конфіденційності даних користувачів. Функція GSS_GetMIC запитує код цілісності повідомлення, який можна перенести для забезпечення цілісності даних, що надсилаються в обрамленні програми.

Як вже згадувалося , механізми GSS-API можуть використовуватися без GSS-API. Вікна

SSPI є широко розгорнутим прикладом цього. Windows SSPI надає особливості Microsoft

API для служб безпеки. Однак механізм Kerberos SSPI такий же, як і механізм Kerberos SSPI

Kerberos GSS-API механізм. Програми, які використовуватимуть як SSPI, так і GSS-API, повинні уникати залежно від проміжку GSS-API або помилок дублікату маркера, якщо пропущене або повторюване повідомлення не вважатиметься фатальною помилкою.

Взаємодія SPNEGO та GSS-API.

Більшість реалізацій GSS-API забезпечують механізм «Просте і захищене переговори» (SPNEGO) (RFC 4178), щоб дозволити програмам вибирати механізм захисту, який вони бажають використовувати. З метою використання механізму SPNEGO програма передає ідентифікатор об'єкта, іменуючи механізм SPNEGO, в GSS_Init_sec_context замість ідентифікатора об'єкта для Kerberos або механізму за замовчуванням.

Реалізація GSS-API буде містити перелік доступних механізмів у маркері, який програма надсилає до сервісу. Цей перший маркер також включає повідомлення з механізму, який клієнт вважає за краще використовувати. Якщо цей механізм також буде обраний службою, переговори не запроваджують жодних туди і назад. Якщо служба не вибирає бажаний механізм клієнта, то клієнт і служба обмінюються кодом цілісності повідомлення після успішної аутентифікації. Цей захист захищає обидві сторони проти зловмисника, який намагається змусити їх вибрати інший механізм захисту, який легше атакувати. SPNEGO доступний на платформі Windows через SSPI. Абоненти повинні використовувати пакет безпеки "домовитись", щоб використовувати SPNEGO.

SASL.

SASL (RFC 4422) надає засоби для узгодження механізмів безпеки, аутентифікації та необов'язкової цілісності та конфіденційності. SASL працює для орієнтованих програм, які надсилають лише один потік байтів у кожному напрямку. Наприклад, SASL добре працює для програм, які надсилають потік байтів через одне TCP-з'єднання. Це не добре працює для додатків, яким потрібно боротися із доставкою поза замовлення або втраченими даними. Багато інтернет-додатків використовують SASL.

SASL та GSS-API – це додаткові технології, які працюють разом: підтримка Kerberos SASL використовує GSS-API. Насправді існує декілька сімейств механізмів SASL для підтримки GSS-API. Найдавніший і найчастіше використовується (RFC 4752) широко реалізований. Більш нова сім'я (GS2)

підтримує зв'язування каналів (див. Kerberos та інші механізми аутентифікації), але широко не застосовується. Програми, які потребують прив'язки каналів для забезпечення адекватної взаємної автентифікації, повинні планувати оновлення до GS2, коли воно стане більш доступним .

Порівняння SASL та GSS-API.

Оскільки SASL використовує GSS-API для забезпечення підтримки Kerberos вищого рівня, SASL є переважним для механізмів GSS-API як мережевого протоколу, але лише тоді, коли він відповідає потребам програми. Основна перевага SASL полягає в тому, що він забезпечує більшу структуру, ніж GSS-API і залишає менше деталей для програми. Наприклад, SASL представляє абстракцію потоку, а не абстракцію повідомлення, а специфікації SASL надають більш детальну інформацію про те, що потрібно зробити програмі для правильного використання SASL. Однак API SASL не так добре стандартизовані, як GSS-API, і деякі розробники додатків зіткнулися з проблемами із загальною реалізацією рамок SASL. Добре правило: якщо використання GSS-API має більше сенсу використовувати SASL з міркувань реалізації, то використовуйте GSS-API. У наступній таблиці подано стислий перелік факторів, які слід враховувати при вирішенні того, який механізм API надає найбільш відповідний протокол для даної програми.

SASL і TLS.

SASL часто використовується уздовж бічних TLS (або SSLv3). Багато механізмів SASL, такі як прості паролі ("звичайний" механізм), не забезпечують належної безпеки лише. Інші механізми SASL забезпечують аутентифікацію, але не надають послуги конфіденційності та цілісності .

У цій схемі використання програма робить SASL та TLS доступними на початку з'єднання. За межами TLS доступні лише механізми SASL, які забезпечують належну безпеку для дотримання політики контролю доступу. Якщо вибрано TLS, SASL також буде доступний як опція після завершення

встановлення сеансу TLS. Зазвичай на даний момент доступний більший набір механізмів.

Механізми GSS-API, такі як Kerberos, які забезпечують конфіденційність та цілісність, повинні бути запропоновані перед початком сеансу TLS. Програми повинні підтримувати, використовуючи захисний рівень SASL (цілісність або конфіденційність, наданий SASL) одночасно з TLS.

Таблиця 2.1 – Фактори для врахування при виборі між SASL та GSS-API

Фактор	SASL	GSS-API Механізми
Мережевий профіль	Програми з одним потоком даних, наприклад, одним TCP-з'єднанням	Надає послуги за повідомленнями, що дозволяють додаткам мати декілька потоків або використовувати протоколи дейтаграм
Анотація інтерфейсу	Програма читає і записує дані, схожі на TCP або TLS	Програма вимагає захисту повідомлень та обробляє деталі надсилання даних по мережі
Підтримка механізму, який не є Kerberos	Прості паролі, виклик / відповідь, одноразовий пароль	Обмежена доступність механізмів, що не належать до Kerberos, на широкорозроблених платформах
API Стандартизація	Для деяких додатків доступні та використовуються крос-платформні API, хоча стандартизовані API все ще розвиваються	Широко доступний стандартизований API

Розробникам додатків було важко підтримувати як захисні рівні TLS, так і SASL в одному і тому ж додатку. Спосіб запису даних у мережу залежить від

того, які служби безпеки використовуються. Іноді дані проходять через TLS, іноді через SASL, а іноді і те й інше. Кілька додатків не підтримують шари безпеки SASL. Якщо вони використовуються з Kerberos та іншими сильними механізмами, ці програми не забезпечують взаємної аутентифікації. З цієї причини найкращою практикою сьогодні є підтримка рівнів безпеки SASL, хоча є додаткова складність. По мірі того, як прив'язка до каналу стає все більш розширеною, найкраща практика, ймовірно, розвиватиметься для використання каналу прив'язки та не для використання шарів безпеки SASL у додатках, які також підтримують TLS.

2.4 Масштабованість сервера

Особливості, додані в поточній версії Kerberos, розроблені для того, щоб дозволити міжмережеву автентифікацію (в термінології Kerberos, яку називають "перехресною автентифікацією"). Останні пропозиції включали використання криптографії з відкритим ключем як для початкової аутентифікації клієнтів (TGT), так і для міжреальної аутентифікації. Такі зміни дозволять Керберосу зробити більш можливим масштабування на більші набори мереж, але питання далеко не вирішене.

Версія 5 додала підтримку для переадресуючих, поновлюваних та поштових квитків. Вони містять тривалі процеси та процеси, які потребують автоматичного запуску в майбутньому, крім того, що дозволяють користувачам використовувати свої облікові дані на машині, відмінній від тієї, на якій вони увійшли.

Квитки на Kerberos тепер можуть містити кілька IP-адрес та адреси для різних типів мережевих протоколів. Це дозволяє використовувати багатодомні машини

Кеші відтворення відслідковують нещодавно видані квитки та не дозволяють використовувати один і той же квиток двічі поспіль. Це зменшує здатність зловмисників викрадати кешовані квитки до їх закінчення.

Зараз існує підтримка транзитивної перехресної автентифікації, яка знімає вимогу, щоб кожна пара царств, які бажають дозволити автентифікацію, повинна мати таємну інформацію. У великих мережах, що складаються з багатьох областей, кількість секретів може стати досить великою і не масштабується. Натомість транзитивна перехресна автентифікація дозволяє визначити шлях між областями спільного використання секретних даних, так що облікові дані від потрібної сфери можна отримати, слідуючи цим шляхом.

Масштабованість сервера Kerberos сприяє можливості використовувати єдину систему автентифікації для різних служб IT-інфраструктури компанії та також для різних її підрозділів..

Початково модель протоколу створювалась для університетського кампусу і була перенесена практично без змін на велику кількість установ та організацій. Однак з вибухоподібним ростом мережі Інтернет перед протоколом автентифікації постали задачі реалізовувати своє призначення не лише в локальнизх мережах, але також і між мережами віддалених підрозділів компанії, в підмережах. Тобто модель автентифікації Kerberos вимагала нових розширень для забезпечення достатнього рівня масштабованості.

З цієї причини в систему понять сервера ввели термін "Царство" для позначення автономної мережі, якою керує один сервер автентифікації. Але одночасно можна налагодити взаємну автентифікацію для використання мережевих служб всієї компанії між різними її підмережами (царствами в термінах Kerberos).

Тут доречним буде згадати організацію мережі під керування операційної системи Microsoft Windows, побудованої на основі доменів. В цьому випадку кожен домен може бути вдображений як окреме царство

Kerberos. Такі підходи можуть дати значні переваги при організації та адмініструванні корпоративних служб каталогів. Тоді сервер Kerberos службу автентифікації з метою адміністрування згаданої служби каталогів.

Обговоримо проблеми масштабування сервісу автентифікації у мережі WIndows, які можуть створити проблеми продуктивності. Такі проблеми можуть призвести до перебоїв, коли ви досягнете певного порогу активності. Проблема зводиться до різних характеристик продуктивності NTLM та Kerberos. Причина віддати перевагу автентифікації Kerberos в Windows Active Directory (AD) і пізніше це не тільки притаманна розширена безпека; це також більш ефективний метод аутентифікації. Кожна аутентифікація на основі NTLM є унікальною – навіть якщо це повторна автентифікація на одному ресурсі з однаковою ідентичністю. Kerberos, з іншого боку, надає службовий квиток, що надає багаторазовий доступ до ресурсу, для цього ресурсу, і це повторне використання не потребує взаємодії з сервером автентифікації або контролером домену (ДК/DC). NTLM – це більш дорогий протокол аутентифікації, а також менш безпечний.

Важливо знати, коли ви досягли точки збою аутентифікації через вузькі місця ресурсів та надмірний обсяг аутентифікації NTLM. Вузьке місце ресурсу може виникнути, коли комп'ютеру Windows необхідно виконати автентифікацію NTLM для певного користувача. Для тих, хто знайомий з архітектурою Windows, ви пам’ятаєте, що за обробку запитів аутентифікації відповідає система підсистеми місцевого управління безпеки (lsass.exe). Це справедливо для всіх версій та ролей Windows. У Lsass.exe є потоки, і ви можете вважати їх працівниками, які виконують завдання з виконання коду. Для аутентифікації NTLM існує максимальна кількість ниткових працівників, які можуть працювати в будь-який час для виконання завдання. За замовчуванням для цього встановлено можливість створення єдиного потоку, якщо комп'ютер є членом домену, та двох потоків, якщо це постійний струм. Цей потік NTLM використовується на доменному комп'ютері для надсилання

запиту до постійного струму, а аналогічна нитка на постійному струмі використовується для створення відповіді. Так, у типовій транзакції є щонайменше два комп’ютери, які можуть бачити це вузьке місце. Під час тієї єдиної транзакції аутентифікації члена домену DC в цьому домені клієнтська нитка перебуває в стані очікування, поки DC не відповість.

Якщо користувач вимагає автентифікації від довіреного домену, у вас є додатковий контакт постійного струму, щоб закінчити цю трансакцію аутентифікації. Цей стан очікування, про який я згадував раніше, тепер матиме оригінальний клієнт і постійний постійний комп'ютер, під час очікування відповіді надійного постійного струму.

Звичайно, потік, що виконує транзакцію NTLM, відбувається швидше, ніж моргання ока. Швидкість не викликає занепокоєння, поки у вас не буде великої кількості одночасних запитів на аутентифікацію NTLM або якщо багато з цих транзакцій переходять через довірені межі DC в інших областях. Додайте зайнятий сервер – генеруючи безліч запитів на автентифікацію NTLM для своїх користувачів – то виникли проблеми.

Назва обмеження для потоків аутентифікації NTLM – MaxConcurrentApi. MaxConcurrentApi (типу даних REG_DWORD) може бути налаштований у реєстрі під HKEY_LOCAL_MACHINE \ SYSTEM \ Параметри параметрів CurrentControlSet \ Services \ Netlogon \, а для введення в дію потрібна лише служба Netlogon.

MaxConcurrentApi – це код Windows, який визначає створення додаткових потоків для обробки нових запитів аутентифікації NTLM. Без потоку для обробки запиту на автентифікацію клієнти-запитувачі (які можуть бути віддаленими комп'ютерами) можуть вимкнути час, стати невідповідними або повернути користувачеві помилки з відхиленням доступу. Ця неоднозначність є причиною, чому з’ясувати першопричину може бути дуже важко.

Для всіх версій Windows встановлене налаштування за замовчуванням для MaxConcurrentApi становить лише 1 для сервера-члена та 2, якщо комп'ютер контролер домену. У Windows Server 2003/2008 то можна змінити налаштування MaxConcurrentApi до 10. Якщо у вас є сервер 2008 R2, максимум – 150, хоча значення за замовчуванням однакові. Якщо у вас є оригінальна установка Server 2008 (а не R2), ви можете встановити виправлення (описане в статті Microsoft "Ви час від часу запитуєте на отримання облікових даних або час очікування, коли ви підключаєтесь до аутентифікованих служб" на support.microsoft.com / kb / 975363), що також дозволить вам збільшити максимум до 150. Це пояснює механіку вузького місця. Тепер поговоримо про ідентифікацію.

Пошук та усунення проблеми.

Найскладнішим аспектом ідентифікації вузького місця аутентифікації є те, що на жодному комп'ютері не зафіксовано жодної події. Натомість всі помилки трапляються в програмі, яка вимагала аутентифікацію. Залежно від обробки помилок програми, може бути недостатньо деталей, щоб точно визначити вузькі місця в NTLM.

Оскільки у вас немає події і може не бути корисної помилки, вам потрібно шукати інші симптоми. Майте на увазі, що це може статися з будь-якою програмою, що використовує NTLM. Найпоширенішими винуватцями є старі програми для бізнесу (LOB), які використовують NTLM, оскільки це був найнижчий загальний знаменник на той час.

Найкращий спосіб визначити, чи досягаєте ви вузьких місць аутентифікації NTLM, це визначити, чи є ці збої результатом обсягу. Якщо збої, як правило, трапляються в часи з високим рівнем використання (наприклад, вранці в понеділок, коли користувачі прибувають і починають свій робочий день), це показник, але не обов'язково остаточний.

Використовуйте об’єкт продуктивності Netlogon Performance Monitor для моніторингу відповідного сервера протягом часу, коли цей сервер

навантажений. Зверніть увагу, що ви повинні зробити це на ресурсному сервері, у якого користувачі мають проблеми з доступом, а також на постійних серверах; ви не хочете пропустити потенційне вузьке місце. У файлі журналу продуктивності (.blg) зверніть увагу на наступне.

Власники семафору, що дорівнюють поточному значенню налаштування значення реєстру MaxConcurrentApi.

Часи очікування семафору з будь-яким числом більше 0.

Семафори, будь-яка кількість яких перевищує 0.

Якщо у вас є очікування або офіціанти, у вас є вузьке місце аутентифікації NTLM.

Виявлення наявності вузького місця є лише першим кроком. Далі потрібно вирішити проблеми з ефективністю, які заважають користувачам отримувати доступ до необхідних послуг, щоб виконати свою роботу. Найпростіший спосіб вирішити – збільшити налаштування MaxConcurrentApi на всіх задіяних серверах до кількості, яке може справити навантаження. Оскільки максимальне число дорівнює 10, найкраще підвищити його до 10, якщо є Windows 2003 або Server 2008, або до більшої кількості, якщо встановлений сервер 2008 R2 (або встановлено оновлення). Потім перезапустіть службу Netlogon на цих серверах.

Коли просто збільшення параметра MaxConcurrentApi не усуває відключення, вам доведеться викопати трохи глибше, щоб з’ясувати, на які комп’ютери та облікові записи користувачів надсилаються запити автентифікації. Ці відповіді має журнал налагодження служби Netlogon. (Див. Статтю Microsoft "Увімкнення журналу налагодження для сервісу Net Logon" на веб-сайті support.microsoft.com/kb/109626 для отримання додаткової інформації.) Цей журнал за замовчуванням не включений, але його легко запустити, його не буде поповніть свій привід, і він індексується часом для довідки.

Що потрібно шукати, як у журналі налагодження служби Netlogon, так і в інших місцях – у порядку, найбільш поширеному для найменш поширених – такі.

NlpUserValidateHigher: Неможливо виділити слот API клієнта – Цей запис тексту в журналі журналу Netlogon вказує на те, що на комп'ютері є очікування на аутентифікацію NTLM, але вже на максимальній кількості потоків. Записи, що передують цьому, повідомлять вам ім'я користувача та комп'ютер, з якого надходить запит.

Вичерпано час вичерпання NlAllocateClientApi – Цей запис тексту в журналі Netlogon вказує на те, що один з клієнтів, який чекав на автентифікацію, відмовився, чекаючи 45 секунд. Поява цього запису означає, що користувач десь отримав запит на вхідні дані, код помилки або невизначене очікування.

(null) \ – Нульові записи в журналі Netlogon вказують на те, що застарілий клієнт у вашій мережі подає запити на автентифікацію NTLM для користувача домену, але пропускає домен користувача, тому замість домену \ user ви бачите (null) \ user. У Windows 2003 це може призвести до додаткового використання цих ресурсів аутентифікації, що означає посилення потенційного вузького місця у справжньому. Щоб вирішити цю проблему, відключіть поведінку ping, використовуючи параметр Neverping, оскільки стаття Microsoft "Процес Lsass.exe може перестати реагувати, якщо у вас є багато зовнішніх довірок на контролері домену Active Directory" (support.microsoft.com/kb/ 923241) описує. Зауважте, що це не стосується сервера 2008 та новіших версій.

Повторні правопорушники. Часті, неодноразові спроби аутентифікації (тобто записи починаються з SamLogon) від одного і того ж користувача та комп'ютера, що з'являються в журналі журналу Netlogon, можуть вказувати на шкідливий або неефективний додаток.

Перевірка PAC Kerberos – як не дивно, ця функція безпеки Kerberos реалізована в Netlogon і використовує ті самі потоки, які є вузьким місцем для аутентифікації NTLM. У цій поведінці є подія, яка з'явиться в системному журналі подій – подія 7 із полем джерела Kerberos. Якщо ви бачите великий обсяг цих подій, а також спостерігаєте періодичні відключення автентифікації для своїх користувачів, спробуйте відключити цю додаткову функцію безпеки тимчасово, поки ви не зможете додати більше серверів для обробки навантаження.

Постійно відключати цю функцію не рекомендується, і це спірне питання, якщо це сервіс пулу додатків Exchange Server або IIS, оскільки їх неможливо відключити. В іншому випадку стаття Майкрософт "У вас виникає затримка в процесі аутентифікації користувача, коли ви запускаєте серверну програму з великим обсягом на члені домену в Windows 2000 або Windows Server 2003 (support.microsoft.com/kb/906736) описує, як зробити Зроби це. Якщо ви підтвердите, що бачите вузькі місця в NTLM, найкращим рішенням буде замість цього використовувати Kerberos. Старіші програми мають меншу ймовірність підтримувати Kerberos, тому це може бути не варіантом.

2.5 Вибір механізмів аутентифікації

Kerberos є ключовою складовою забезпечення контролю доступу в багатьох сучасних компаніях. На початку дня користувачі входять у Kerberos, отримуючи облікові дані один раз, а потім користуючись програмами протягом дня. Kerberos надає відповідні обміни безпеки та гарантії, щоб ці програми не вимагали імені користувача або пароля. Користувач отримує зручність того, що його не вимагають постійно входити в кожну програму. Кожна програма отримує діючий спосіб іменування користувачів. Організація отримує єдиний пункт, за яким слід застосовувати політику безпеки. Інші компоненти стратегії управління доступом організації зазвичай включають

каталог для зберігання інформації про авторизацію та атрибути щодо використаннярs, а також інфраструктура управління. Однак Kerberos надає посилання, що дозволяє програмам брати участь у стратегії контролю доступу підприємства. Хоча ця модель поширена на великих підприємствах, вона розширюється до робочих груп та однорангових програм, оскільки Kerberos стає все більш інтегрованим у платформи операційної системи.

Як обговорюється в статті "Роль Кербероса в сучасних інформаційних системах", існує багато елементів стратегії управління доступом до підприємства. Kerberos дозволяє додаткам використовувати елементи управління доступом організації або платформи, а не вимагати від кожної програми надання повної стратегії контролю доступу. Навіть якби додатки забезпечували повні можливості контролю доступу, інтеграцією з іншими послугами аутентифікації та авторизації підприємств було б бажано уникнути витрат на управління декількома елементами контролю доступу. Як приклад такої інтеграції, якщо обліковий запис вимкнено, цей обліковий запис не зможе використовувати жодних додатків, інтегрованих із службами управління обліковими записами та аутентифікацією. Програми також можуть скористатися будь-якою інформацією про авторизацію, доступною в каталозі. Можливо, автоматично буде надано користувачам можливість користуватися програмами, коли обліковий запис користувача створено в організації.

Єдиний вхід.

Найбільш помітною перевагою для Kerberos для кінцевих користувачів є одинарний вхід. Користувачеві не потрібно входити в кожну програму, а натомість можна один раз увійти на свій комп’ютер. Керберос здійснює єдину реєстрацію, зберігаючи облікові дані, які зазвичай тривають приблизно один робочий день. Коли користувач підписується на комп'ютер, локальна реалізація Kerberos зв'язується з Центром розподілу ключів (KDC) для автентифікації користувача на KDC. Коли аутентифікація проходить успішно, KDC видає квиток. Квиток є обмеженим часом повідомленням від KDC для

себе, що свідчить про успішну автентифікацію. Цей квиток разом із ключем сеансу, відомим лише місцевому комп'ютеру, і KDC утворює обліковий запис, який можна використовувати для входу в програми. Коли Керберос бажає зв’язатися із заявкою, він пред'являє квиток разом із підтвердженням того, що KDC відомий клієнтом сеансовий ключ та отримує новий службовий квиток для програми, з якою зв’язується. Цей квиток разом із специфічним сеансовим ключем використовується для безпечного зв’язку з програмою.

KDC слугує центральним пунктом для виконання політики автентифікації організації та застосування загальної політики щодо управління обліковими записами.

Служби безпеки.

Ефективні політики контролю доступу залежить від цілого ряду служб безпеки. У цьому розділі обговорюються послуги, які надає Kerberos, і як вони вписуються в контроль доступу до додатків. Надання цих послуг є важливою метою інтеграції Kerberos у додаток. Таким чином, план інтеграції Kerberos у додаток повинен ретельно враховувати, як надаються ці послуги. Перевірка того, що ці послуги надаються, має важливе значення для підтвердження інтеграції Kerberos у додаток.

Аутентифікація.

Автентифікація – це процес перевірки в достатній мірі довіри тверджень про сторону чи повідомлення. Зазвичай мережевий додаток повинен знати деякі атрибути, наприклад, ім'я, про сторону, що надсилає йому повідомлення.

Kerberos відокремлює автентифікацію на дві фази. Початкова аутентифікація відбувається між клієнтом Kerberos та KDC. Використовувані механізми визначатимуться політикою сайту; типові приклади включають загальнодоступні секрети (паролі) або смарт-карти. Пізніше клієнт підтверджує автентифікацію програми. Як побічний ефект цього обміну, клієнт і додаток діляться ключем сеансу, який може використовуватися в наступних криптографічно захищених комунікаціях.

Сьогоднішні мережеві програми вимагають аутентифікації обох сторін з'єднання, щоб запобігти фішингу та інших зловмисних атак. Настільки ж важливо, щоб сервер пройшов автентифікацію для своїх клієнтів, щоб їх управління доступом можна було підтримувати, як і клієнтам, які мають автентифікацію на сервері. На щастя, Kerberos робить взаємну автентифікацію легкою. Керберос симетричний; будь-які дві сторони, які можуть автентифікуватись одним способом, також можуть аутентифікуватись в іншому напрямку.

Конфіденційність та цілісність.

Автентифікація сторін мережевого обміну є недостатньою для забезпечення розумних цілей контролю доступу. Сам вміст повідомлення потрібно захистити. Якщо повідомлення не захищені, зловмисник може змінювати повідомлення, перемагаючи політику контролю доступу. Kerberos надає засоби, щоб переконатися, що повідомлення не змінюються під час подорожі по мережі. Повідомлення необов'язково можуть бути зашифровані, тому лише ті учасники, які знають ключ сеансу, можуть перевіряти свої суперечки.

Деякі програми вже мають рівень протоколу, відповідальний за безпеку повідомлень. Наприклад, програми можуть використовувати безпеку транспортного шару (TLS, спадкоємця SSL) або IPsec для забезпечення захисту повідомлень. Техніка, що називається прив'язка каналу, може бути використана для прив'язки автентифікації Kerberos до рівня захисту повідомлень, щоб програма мала впевненість у тому, що повідомлення передаються між тими ж сторонами, які аутентифіковані Kerberos.

Авторизація.

Після того, як Kerberos успішно зареєстрував клієнта на послугу і після того, як будуть створені відповідні служби безпеки, клієнту та службі все одно потрібно приймати рішення про авторизацію для досягнення цілей контролю доступу. До яких об’єктів повинен бути дозволений клієнт? Які дозволи

повинні мати клієнт? Чи є послуга такою, з якою клієнт розраховував поговорити? Наскільки клієнту довіряють послугу? Усі ці питання є частиною дозволу, який потрібно виконувати клієнту та службі.

Kerber os автентифікує клієнта та послугу. У деяких середовищах Kerberos також надає інформацію про групову приналежність клієнта. Решта авторизації – це питання платформи та програми. У статті "Роль Kerberos" описано, як Kerberos вписується у функцію авторизації платформи. У цьому документі розглядаються деякі конкретні проблеми, де авторизація стосується того, як Kerberos інтегрується в додаток. У розділі Імена Федерації та Клієнтів обговорюються проблеми, пов’язані з узгодженням назв Ker beros із назвами програм. У розділі про SID та PAC обговорюється, як Active Directory Microsoft передає групове членство в службу.

Протокольні засоби інтеграції Kerberos.

Є дві важливі осі, які описують підходи до інтеграції Кербероса. Перший – це програма протоколу або мережеві повідомлення, які використовуються. Друга – реалізація Kerberos, яка використовується. Не всі комбінації можливі: не всі реалізації забезпечують доступ до всіх засобів протоколу. Цей розділ описує першу вісь, а наступний розділ описує вісь реалізації.

Kerberos надає додаткам гнучкість для визначення того, які повідомлення на рівні мережі потрібно прийняти. Існує дуже широка послідовність у взаємодії програм із KDC та початковими повідомленнями, які клієнт використовує для автентифікації до сервісу. Однак навіть повідомлення, яке клієнт використовує для автентифікації до послуги, потрібно перенести у певному обрамленні для програми. Крім того, програми можуть застосовувати різні підходи до цілісності та конфіденційності даних користувачів. У цьому розділі описано три найбільш поширені підходи.

Сировинні повідомлення Kerberos описані для встановлення контексту. Зазвичай вони не повинні використовуватися в нових програмах. Механізм GSS-API Kerberos є кращим способом інтеграції Kerberos у додатки. SASL

також описаний, оскільки він пропонує простий спосіб інтеграції механізмів GSS-API в додаток, припускаючи, що додаток вписується в обмеження, накладені SASL.

2.6 Дійові особи, які беруть участь в виконанні протоколу

Перерахуємо основних акторів (дійових осіб), які потрібні для того, щоби кожен вузол мережі успішно пройшов процес автентифікації.

Царство (царина, Realm): множина мережевих інтерфейсів, тобто хостів, котрі є робочими станціями. Віртуальнми машинами, серверати та іншими активними пристроями), котрі керуються одним сервером, тобто записи про котрі зберігаються в одній базі даних.

Елемент (Principal): певна унікальна сутність, котра може володіти квитком автентифікації, виданим сервером автентифікації. Це може бути будь-який хост у мережі, котрий проходить процес автентифікації для отримання доступу до певного сервісу. Форма ідентифікатора такого квитка була описана вище.

Сервер видачі тікетів (TGS): один з модулів і одночасно служба KDC, що відподіає за генерацію та присвоєння унікального квитка автентифікації протоколу. Тобто ця служба видає квитки клієнтам для використання конкретних сервісів у мережі.

2.7 Припущення стосовно конфігурації, необхідні для практичної реалізації системи в роботі

На основі специфікації Kerberos v5 зробимо наступні припущення.

По перше, захист від атак типу DDoS не може бути реалізований за допомогою сервера автентифікації. Це можна зробити за допомогою розширень та доповнень.

Кожен вузол мережі повинен зберігати свої секретні ключі безпечно. В іншому випадку вся мережа, контрольована сервером автентифікації, може виявитись скомпрометованою.

Сервер Kerberos не надає захисту від атак типу грубої сили (брут-форс), направлених на підбір пароля. Це, знову ж таки, випрішується тільки за допомогою додаткових розширень.

Кожен вузол у мережі, контрольованій одним сервером автентифікації, повинен мати доступ до служби часу для генерації часових міток тікетів (квитків). На сьогодні ця вимога практично реалізована майже у будь-якій операційній системі.

Політика створення імен (ідентифікаторів) для вузлів мережі (Principals) повинна передбачати уникнення повторного використання ідентифікаторів в рамках одного царства (realm).

2.8 Взаємна автентифікація сервісів (перехресна)

Kerberos v5 реалізує підтримку взаємної автентифікації між різними стуностями в довірчих царствах (realm) Kerberos. Для спроцення представлення вважатимемо, що всі суб'єкти процесу знаходяться в одному царстві. Коротко проаналізуємо складові протоколу автентифікації.

Кожен з них реалізований у вигляді підпротоколу. Їх є три і до них належать такі:

- сервіс автентифікації.
- сервіс обміну квитками (Token).
- служба взаємодії клієнта і сервера пісдя автентифікації.

РОЗІДЛ 3
ПРАКТИЧНА РЕАЛІЗАЦІЯ

3.1 NTP та синхронізація часу

Оскільки безпека автентифікації Kerberos частково базується на часових мітках квитків, важливо точно встановити годинник на серверах Kerberos. Як ми згадували у вступі до Кербероса, короткий термін експлуатації квитків використовується для запобігання зловмисникам виконувати успішні напади грубої сили або повторювати атаки.

Якщо ви дозволите дрейфувати вашим годинникам, ви зробите свою мережу вразливою до таких атак. Оскільки синхронізація годинника настільки важлива для безпеки протоколу Kerberos, якщо годинник не синхронізований у розумному вікні, Kerberos повідомить про фатальні помилки та відмовиться функціонувати. Клієнти, які намагаються здійснити автентифікацію з машини з неточним годинником, не зможуть KDC у спробах аутентифікації через різницю в часі з годинником KDC.

3.2 Попередня аутентифікація

Попередня автентифікація Kerberos визначена в RFC 6113 та реєстрі IANA. Це є функція безпеки, яка забезпечує захист від атак нападів пароля. Запит AS ідентифікує клієнта до KDC в Plaintext. Якщо включена попередня автентифікація Kerberos, часова мітка буде зашифрована, використовуючи хеш пароля користувача як ключ шифрування. Якщо KDC зчитує дійсний час, коли використовує хеш пароля користувача, який доступний в каталозі Microsoft Active Directory, для розшифрування часової позначки, KDC знає, що запит не є повтором попереднього запиту.

Без попередньої перевірки Kerberos зловмисник може безпосередньо надіслати фіктивний запит на аутентифікацію. KDC поверне зашифрований TGT і зловмисник може жорстоко примусити його в автономному режимі. Після перевірки журналів KDC нічого не буде видно, окрім одного запиту на TGT. Коли часова марка Kerberos позначається попередньою автентифікацією, зловмисник не може безпосередньо просити KDC за зашифрованим матеріалом Brute-Force в автономному режимі.

3.3 Квитки аутентифікації протоколу

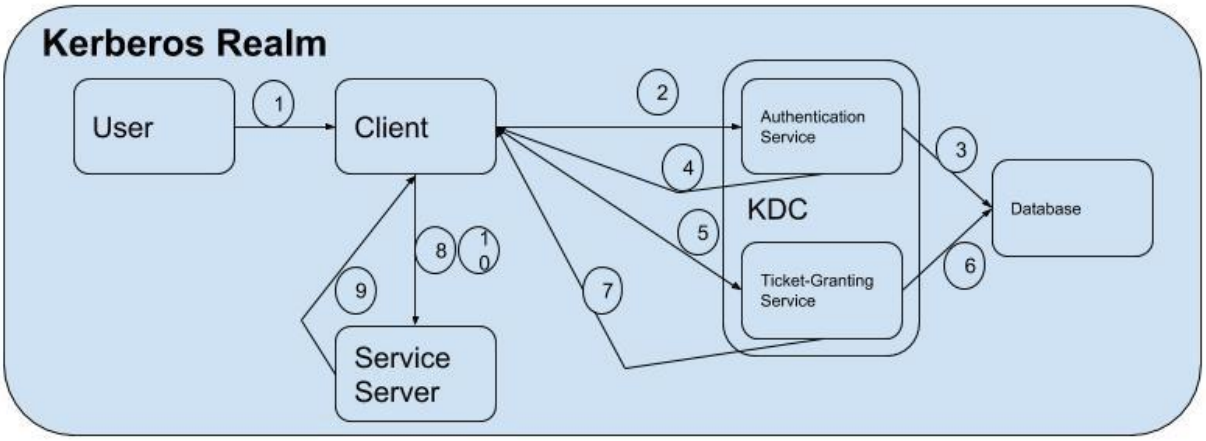
RFC 4120 описує процес Ticket-granting ticket (TGT), Квиток на видачу квитків (TGT) - це перший квиток, отриманий в системі kerberos. Це спеціальний квиток, який дозволяє клієнту отримати додаткові квитки на Kerberos в межах тієї ж області Kerberos.

У системі Kerberos клієнт (як правило, користувач, або послуга) надсилає запит на отримання квитка в Центр розподілу ключів (KDC). KDC створює для клієнта квиток (TGT), шифрує його, використовуючи пароль клієнта як ключ, і відсилає зашифрований TGT назад клієнту. Потім клієнт намагається розшифрувати TGT, використовуючи свій пароль. Якщо клієнт успішно розшифровує TGT (тобто, якщо клієнт вказав правильний пароль), він зберігає розшифровану TGT, що вказує на підтвердження ідентичності клієнта.

TGT, термін дії якого закінчується у визначений час, дозволяє клієнту отримати додаткові квитки, які дають дозвіл на конкретні послуги. Запит та надання цих додаткових квитків є зрозумілими для користувачів. Тримання дійсного TGT дозволяє довірителю запитати сервісний квиток.

Ця концепція реалізована шляхом видачі квитка, який не посилається на будь-яку мережеву адресу, звідки дозволено надходити запит.

На наступній діаграмі (рисунок 3.1) показано високоефективну взаємодію акторів Kerberosa в процесі видачі квитків та автентифікації клієнта.



1. User enters credentials (username + password).
2. Send KRB_AS_REQ.
3. Lookup user (and password) in database.
4. Send KRB_AS_RSP.
5. Send KRB_TGS_REQ.
6. Lookup service (and password) in database.
7. Send KRB_TGS_RSP.
8. Send KRB_AP_REQ.
9. Send KRB_AP_RSP.
10. Send service request to Service Server.

Рисунок 3.1 – Взаємодія ролей в протоколі аутентифікації

Схема на наступному рисункові (рисунок 3.2) містить опис обміну повідомленнями в деталях між учасниками процесу автентифікації.

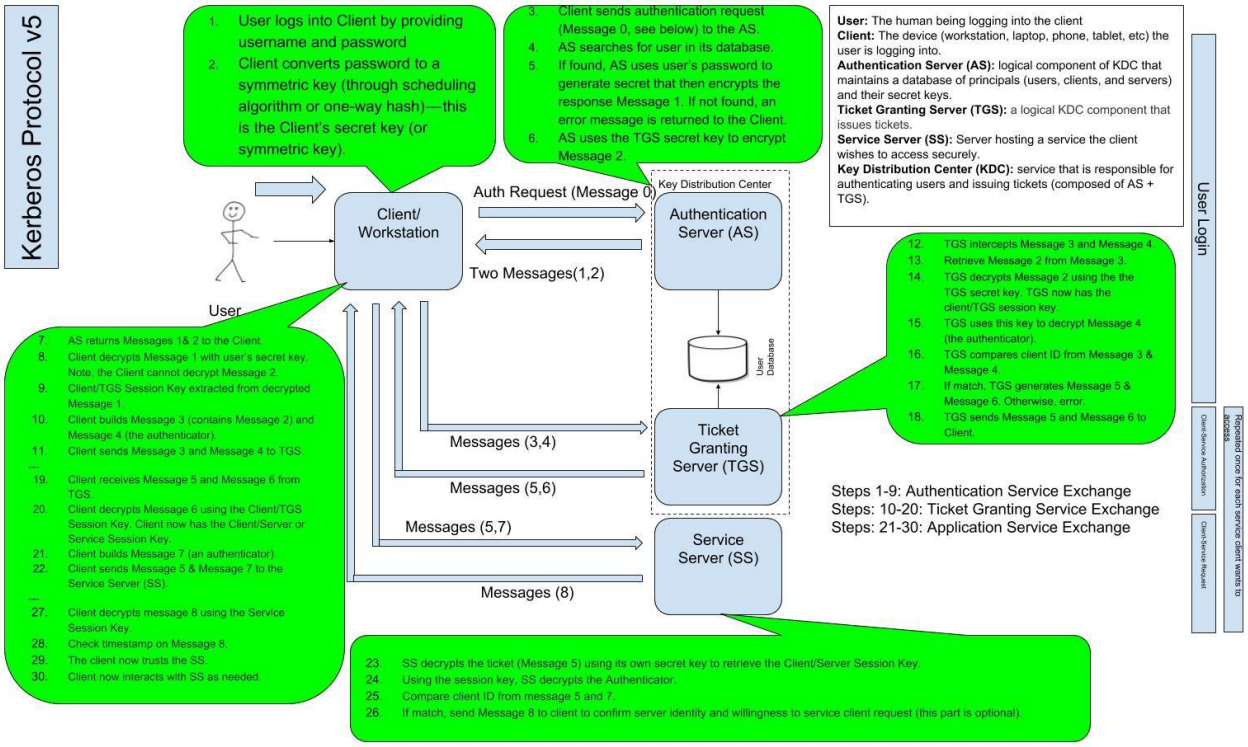


Рисунок 3.2 – Детальне представлення протоколу автентифікації

3.4 Служба аутентифікації

Кроки автентифікації, перераховані та коротко описані далі, ілюструють процес автентифікації клієнта.

1. На початку користувач проходить процес авторизації на клієнтській машині через свій логін та пароль.

2. На цій локальній машині отриманий пароль перетворюється у симетричний ключ за допомогою хеш-функції. Цей ключ є секретним ключем клієнта.

3. Після цього від клієнта до сервера увтентифікації Authentication Service (AS) надсилається запит аутентифікації (повідомлення 0 на рис. 3.3).

4. AS виконує пошук клієнта по своїй БД.

5. Коли запис про клієнта знайдено, то AS на основі пароля користувача шифрує відповідь. Цьому відповідає зображення Повідомлення 1 на рис. 3.3. Коли клієнт у БД не знайдений, він отримає повідомлення про помилку.

6. AS шифрує повідомлення 3 з використанням секретного ключа TGS.

7. Після цього від AS до Клієнта надходять повідомлення 1 разом 3.

8. Клієнт розшифровує перше повідомлення з використанням свого секретного ключа. При цьому клієнт повідомлення 3 розшифрувати фізично не може через відсутність ключа.

Деякі зауваження:

– Після проходження цих кроків креденціали користувача вважаються підтвердженими і він вважається автентифікованим.

– Згідно RFC 4120, "Такий обмін повідомленнями як правило застосовується при початку сеансу підключення дослужби видачі квитків авторизації, які потім використовуватимуться для отримання профілів на інших сервісах без застосування секретного ключа клієнта;

– Згідно RFC 4120, “Коли не виконувати попередньої автентифікації, то серверу невідомо нічого про клієнта, тобто чи він справжній чи ні. Сервер всього лиш надсилає відповідь не турбуючись про це питання. Але це є прийнятним, оскільки відповідь може використати лмше довіритель, який вказаний у запиті авторизації.

– Інформація передається у тестовому форматі, коли не вказно інше.

– Секретний ключ клієнтського комп'ютера не кешується з міркувань безпеки.

3.5 Обмін повідомленнями сервісів TGS

1. Клієнтська машина генерує повідомлення 3, яке інкапсулює повідомлення 2 та 4.
2. Далі від клієнта до TGS відправляється Повідомлення 3 та 4.
3. TGS отримує адресовані йому повідомлення 3 та 4.
4. З повідомлення 3 здійснюється екстракція другого поідомлення.
5. Служба TGS розшифровує екстраговане повідомлення з використанням секретного ключа. Тепер в служби наявний ключ сеансу зв'язку між клієнтом і службою TGS.
6. Тепер служба TGS може розшифрувати повідомлення 4.
7. TGS надсилає відповідь клієнтові, ідентифікатор котрого отриманий з третього та четвертого повідомлень.
8. При відсутності помилок (ідентифікатори однакові), TGS створює Повідомлення 5 та 6. Інакше клієнт отримає повідомлення про помилку.
9. Служба TGS відправляє клієнту повідомлення 5 та 6.
10. Клієнт генерує відповіді 3 та 4 та надсилає їх на TGS.
11. Від TGS клієнт отримує відповіді 5 і 6.
12. Повідомлення 6 розшифровується на клієнтській машині з використанням ключа сесії зв'язку.

3.6 Обмін повідомленнями між клієнтом та цільовою службою

1. Клієнт генерує Повідомлення 7 для автентифікації на сервісі.
2. Клієнт відправляє на цільовий сервер Service Server (SS) повідомлення 5 та 7.
3. SS розшифровує отриманий квиток (повідомлення 5).
4. Ідентифікатори клієнти з меседжів 5 та 7 порівнюються.

5. При позитивному результаті клієнтові надсилається меседж 8 для підтвердження того, що сервер автентифікований. Тепер сервер буде готовий виконувати запити користувача.

6. На клієнтській стороні повідомлення 8 розшифровується та виконується перевірка його часової мітки.

7. Тепер між клієнтом на SS встановлені довірчі відносини.

8. Клієнт взаємодіє з SS.

Всі описані кроки взаємодії у вигляді обміну повідомленнями показані на рисунку 3.3.

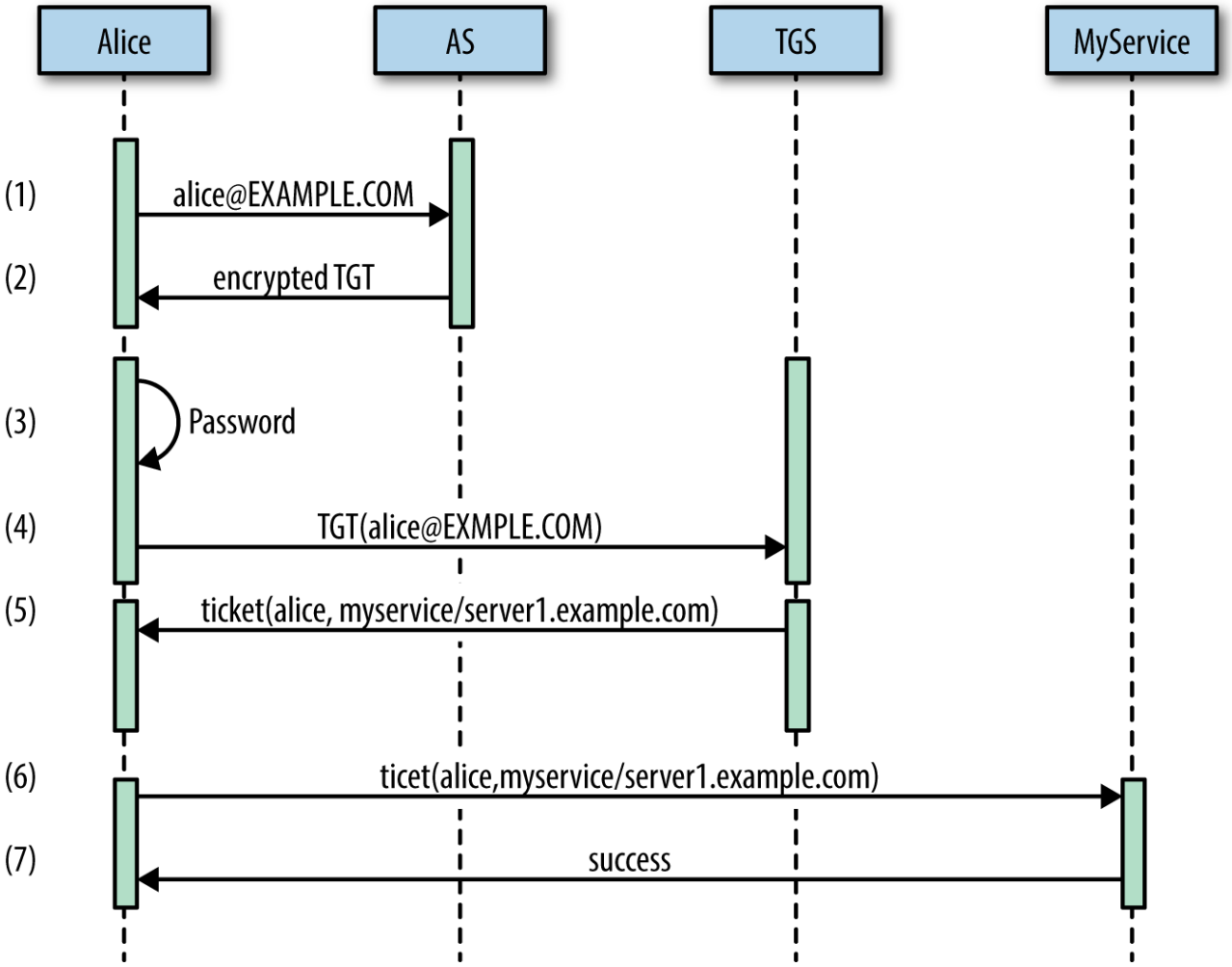


Рисунок 3.3 – Діаграма обміну повідомленнями при виконання автентифікації

3.7 Архітектура програмної системи автентифікації в ОС Windows на основі Kerberos

Рзглянемо використання Kerberos в Microsoft Windows (Microsoft Kerberos). Усі поточні версії Microsoft Windows містять підтримку Kerberos. Windows підтримує механізм Kerberos GSS-API, але не використовує GSS-API як інтерфейс програми. Натомість програми Windows можуть отримати доступ до Kerber os через інтерфейс постачальника підтримки безпеки (SSPI). SSPI надає аналогічні функції GSS-API, хоча конкретні виклики API, які потрібно робити додатку, відрізняються. У термінології Microsoft Kerberos є однією з декількох систем безпеки

Постачальники послуг (SSP), які можна використовувати через інтерфейс SSPI. Є кілька незначних відмінностей між механізмом Microsoft та іншими механізмами GSS-API Kerberos:

• Windows не підтримує функцію дублікату токена або маркер розриву GSS-API. Якщо виявлення послідовності чи повторного відтворення ввімкнено у Windows, тоді не відкритий простий текст, якщо повідомлення, що не перебувають у послідовності чи повторне відтворення, розгортаються.

• Windows підтримує режим стилю DCE, який незначно змінює формати токенів.

• Windows підтримує можливість передачі декількох буферів при генерації токена; інші реалізації GSS-API цього не роблять.

Kerberos від Microsoft забезпечує єдиний вхід у домен Windows. Доменні дані домену доступні і можуть використовуватися для підключення до мережевих служб. Windows також дозволяє програмам отримувати облікові дані Kerberos на основі пароля, який передається в SSPI. Однак Windows не надає користувальницький інтерфейс для отримання облікових даних за замовчуванням для ідентичності, відмінної від ідентичності домену користувача, а це означає, що користувач не може встановити альтернативні

облікові дані за замовчуванням, які можна використовувати під час звернення до мережевих служб. Windows надає сховище з паролем (Credential Manager), що дозволяє користувачам зберігати пароль, який буде використовуватися під час звернення до певних служб; якщо сервіс підтримує Kerberos, тоді Windows спробує використати цей пароль, щоб отримати квитки на Kerberos.

SID та PAC.

Active Directory використовує поле в квитку Kerberos під назвою "дані авторизації" для зберігання Microsoft PAC (сертифікат атрибутів Privilege). Крім усього іншого, PAC включає набір ідентифікаторів безпеки (SID) для клієнта. Ці ідентифікатори у жодному разі ідентифікують обліковий запис та будь-які групи, до яких належить обліковий запис у каталозі. Служби можуть довіряти PAC, оскільки він захищений цілісністю KDC. Програми можуть приймати рішення про авторизацію на основі групового членства користувача. Без PAC службам потрібно буде запитувати каталог, щоб з’ясувати, до яких груп належить обліковий запис.

SID вирішують проблему перейменування облікових записів. Для інших версій Kerberos основне ім’я використовується як ідентифікатор. Якщо основне ім’я зміниться, додатки більше не зможуть відстежувати інформацію про авторизацію цього облікового запису. Однак SID не змінюються, коли змінюється ім’я облікового запису користувача.

Порівнюючи KFW та Windows Kerberos.

Як обговорювалося вище, для платформи Windows обидва доступні Kerberos для Windows (версія M IT Kerberos для платформи Windows, скорочена як KFW) та Windows Kerberos (рідна підтримка Kerberos, що є частиною операційної системи Windows). Якщо рідної Windows Kerberos достатньо для заданої копії, використовувати її найпростіше. Щоб використовувати KFW, KFW потрібно встановити та налаштувати. Однак у деяких середовищах KFW має важливі переваги:

• KFW дозволяє машинам, які не входять до домену, легко використовувати Kerberos; це добре працює для індивідуальних машин, яким все ще потрібно отримати доступ до захищених мережевими ресурсами Kerberos.

• KFW підтримує необроблений API Kerberos та GSS-API. Це може дозволити розробникам створювати додатки, які легко переносяться як на Windows, так і на інші платформи.

• KFW підтримує AES та інше сильне шифрування у всіх версіях Windows. Windows Vista - перша версія Windows, яка підтримує AES. У майбутньому, KFW може дозволити використовувати нові функції Kerberos у існуючих версіях Windows.

• Починаючи з Windows Vista та KFW 3.2, KFW може встановити ідентифікацію за замовчуванням, яку використовуватиме клієнт Windows під час звернення до мережевих служб. Це може значно покращити досвід Kerberos під час доступу до служб за межами контексту домену Windows.

Інтеграція KFW в додаток.

Розглянемо спочатку, як відбувається вхід в локальну машину. На рисунку 3.4 показано цей процес та загальний вигляд побудови архітектури безпеки для ОС Windows. Припускаємо, що в якості основи обміну буде вибрано NTLM.

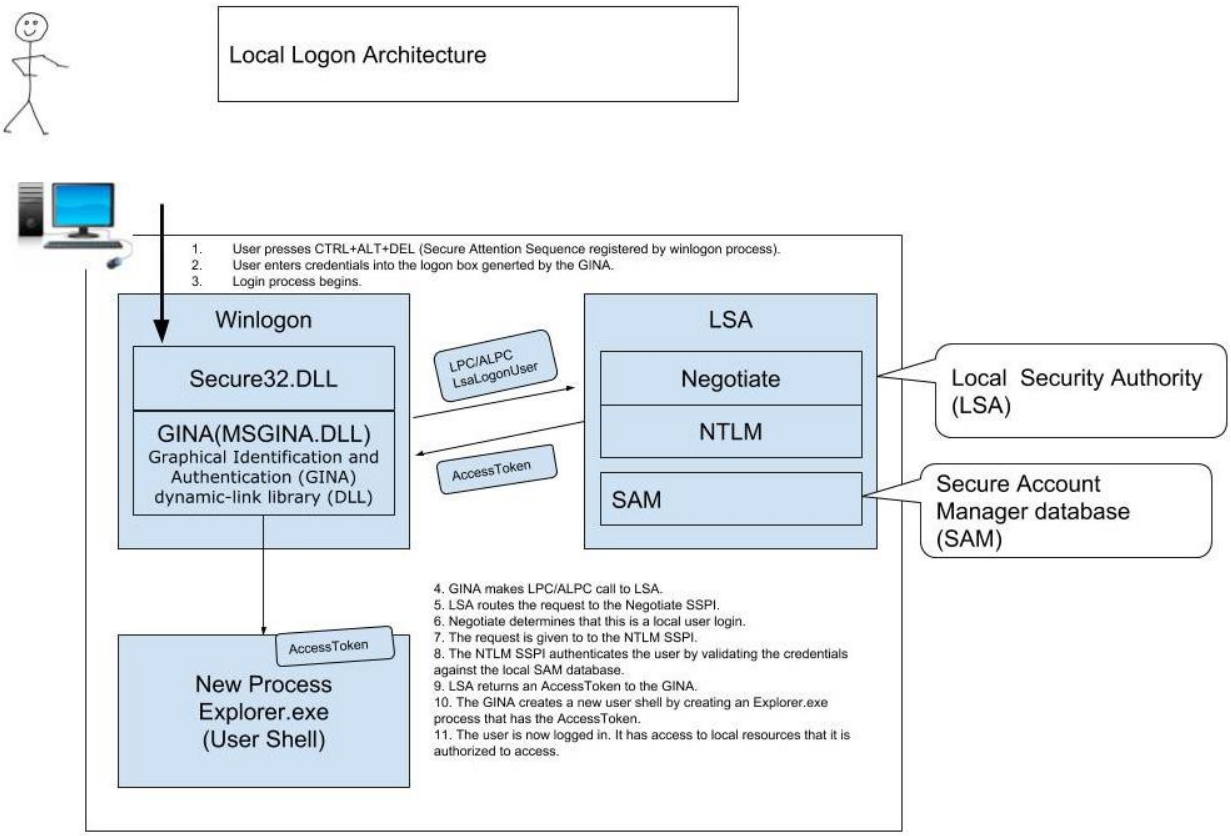


Рисунок 3.4 – Процес входу локальних профілів

На схемі з рисунка 3.5 зображено вхід користувача в домен Windows. І саме тут вже для автентифікації взаємної між клієнтською машиною та контролером домену використовується Kerberos, а не який-небудь інший варіант.

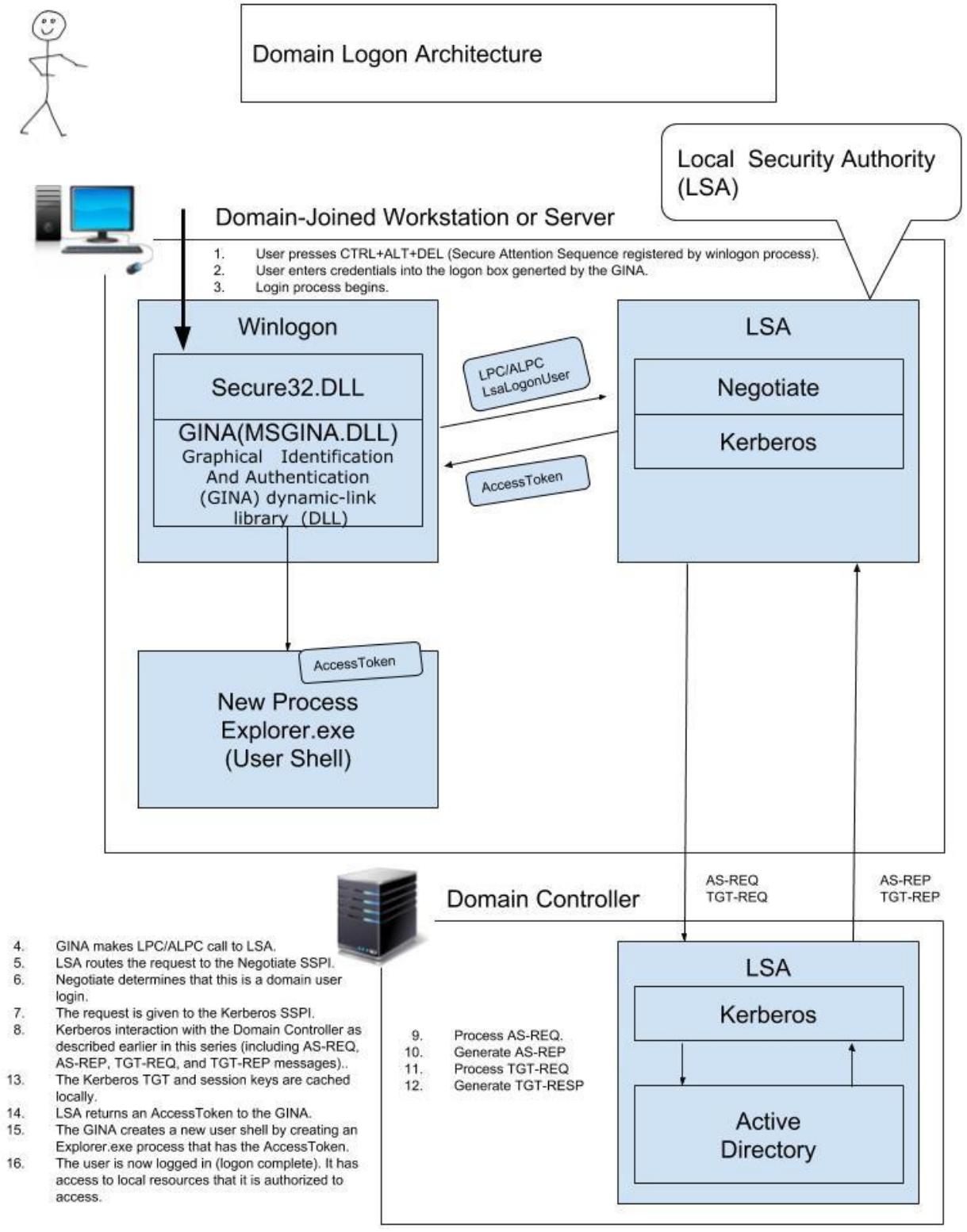


Рисунок 3.5 – Авторизація користувачів домену

Архітектура Windows SSPI як реалізація Microsoft GSS-API може бути зображена, як на рис. 3.6.

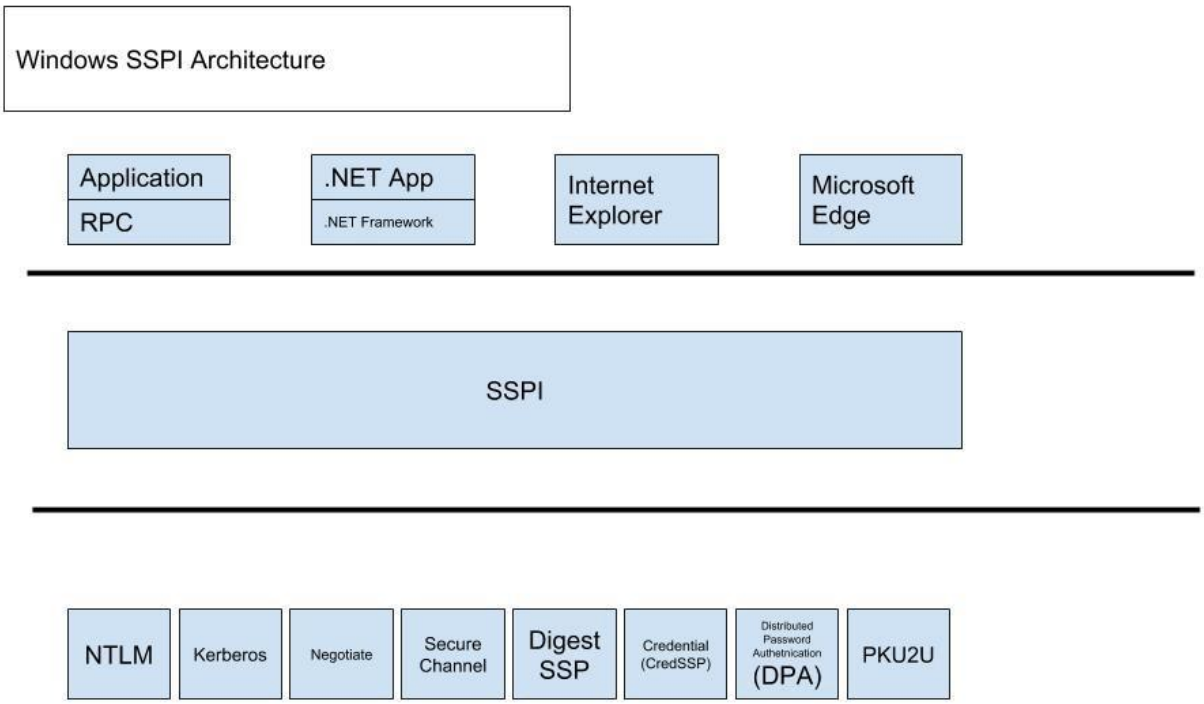


Рисунок 3.6 – Windows SSPI

Microsoft не застосовує RFC1964 (GSS-API) для Kerberos; їх SSPI – Security Service Provider Interface , використовує власний набір функцій GSS-API, та вони мають свої особливості реалізації під ОС Windows. Microsoft Windows володіє власним стеком служб автентифікації для Security Service Provider (SSP):

– Kerberos.

– NTLM (NT Manager).

– Безпечний канал (SChannel).

– Розподілена парольна автентифікація (DPA).

– Шифрування з відкритим ключем між користувачами мережі (PKU2U).

Після успішного виконання всіх кроків протоколу авторизації Kerberos володіє даними для привілейованих атрибутів (PAC), які є складовою частиною меседжів TGT_REP і служать для генерації маркера доступу.

Будь-який процес, створений в рамках сесії користувача (типово це Explor.exe) чи який-небудь підпроцес унаслідує значення маркера доступу.

Саме маркер доступу є основним ідентифікатором користувача чи клієнта, бо він однозначно співставляється з процесом чи потоком.

3.9 Приклад розгортання системи з сервером віддаленого доступу для домену Windows

Оскільки протокол Kerberos використовує в основному незашифровані повідомлення, то можна використовувати програмне забезпечення, котре доволить відслідкувати пакети повідормлень і проілюструвати виконання налаштування системи автентифікації на основі віддаленого сервера Для цього для прикладу можна застосувати програму Wireshark. Крім того, в барузері можуть використовуватись вбудовані засоби відладки та моніторингу за протоколом HTTP (S). Будемо фіксувати мережевий трафік на контролері домену Windows, на котрому в свою чергу запущена служ\ба автентифікації Kerberos.

Розглянемо квиток автентифікації Kerberos, який містить наступні дані. У його незашифрованій частині можна побачити такі поля:

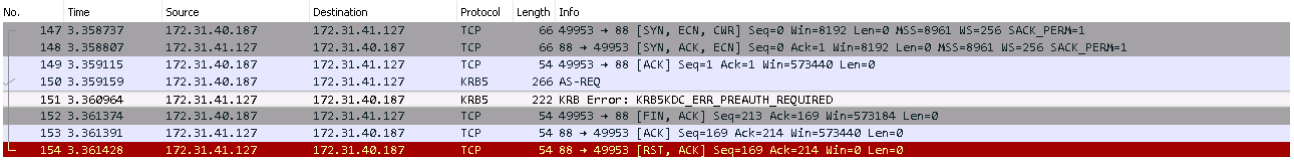
- номер версії протоколу.
- ідентифікатор царини.

У зашифрованій частині знаходяться такі дані, які мають значення для автентифікації і ці поля описуються інформацію про клієнта, цільовий сервер, часові мітки та ряд прапорів, котрі керують виконанням команд протоколу.

Аутентифікатори протоколу ніяким чином не можуть бути використані більше одного разу. В іншому випадку служба автентифікації має відхилити такі квитки. Розглянемо приклад автентифікатора: який містить меседж TGT-REQ.

На наступному рисунку, який містить знімок екрану, показано ініціалізацію процесу підключення через TCP-з'єднання служби Windowsв домені і контролером домену. Ім'я користувача – RCBJ. Таке ж і ім'я домену.

Початкові три пакети типово відображають протокол рукостискання SYN-SYNACK-ACK для TCP-сеансу, що видно з рисунка 3.7.



No.	Time	Source	Destination	Protocol	Length	Info
147	3.358737	172.31.40.187	172.31.41.127	TCP	66	49953 → 88 [SYN, ECN, CWR] Seq=0 Win=8192 Len=0 MSS=8961 WS=256 SACK_PERM=1
148	3.358807	172.31.41.127	172.31.40.187	TCP	66	88 → 49953 [SYN, ACK, ECN] Seq=0 Ack=1 Win=8192 Len=0 MSS=8961 WS=256 SACK_PERM=1
149	3.359115	172.31.40.187	172.31.41.127	TCP	54	49953 → 88 [ACK] Seq=1 Ack=1 Win=573440 Len=0
150	3.359159	172.31.40.187	172.31.41.127	KRB5	266	AS-REQ
151	3.360964	172.31.41.127	172.31.40.187	KRB5	222	KRB Error: KRB5KDC_ERR_PREAUTH_REQUIRED
152	3.361374	172.31.40.187	172.31.41.127	TCP	54	49953 → 88 [FIN, ACK] Seq=213 Ack=169 Win=573184 Len=0
153	3.361391	172.31.41.127	172.31.40.187	TCP	54	88 → 49953 [ACK] Seq=169 Ack=214 Win=573440 Len=0
154	3.361428	172.31.41.127	172.31.40.187	TCP	54	88 → 49953 [RST, ACK] Seq=169 Ack=214 Win=0 Len=0

Рисунок 3.7 – Початок сеансу в домені Windows

Наступний, четвертий пакет даних, що іде від клієнта на KDC , є меседж AS-REQ. Його показано на наступному рисунку (див. рис. 3.8).

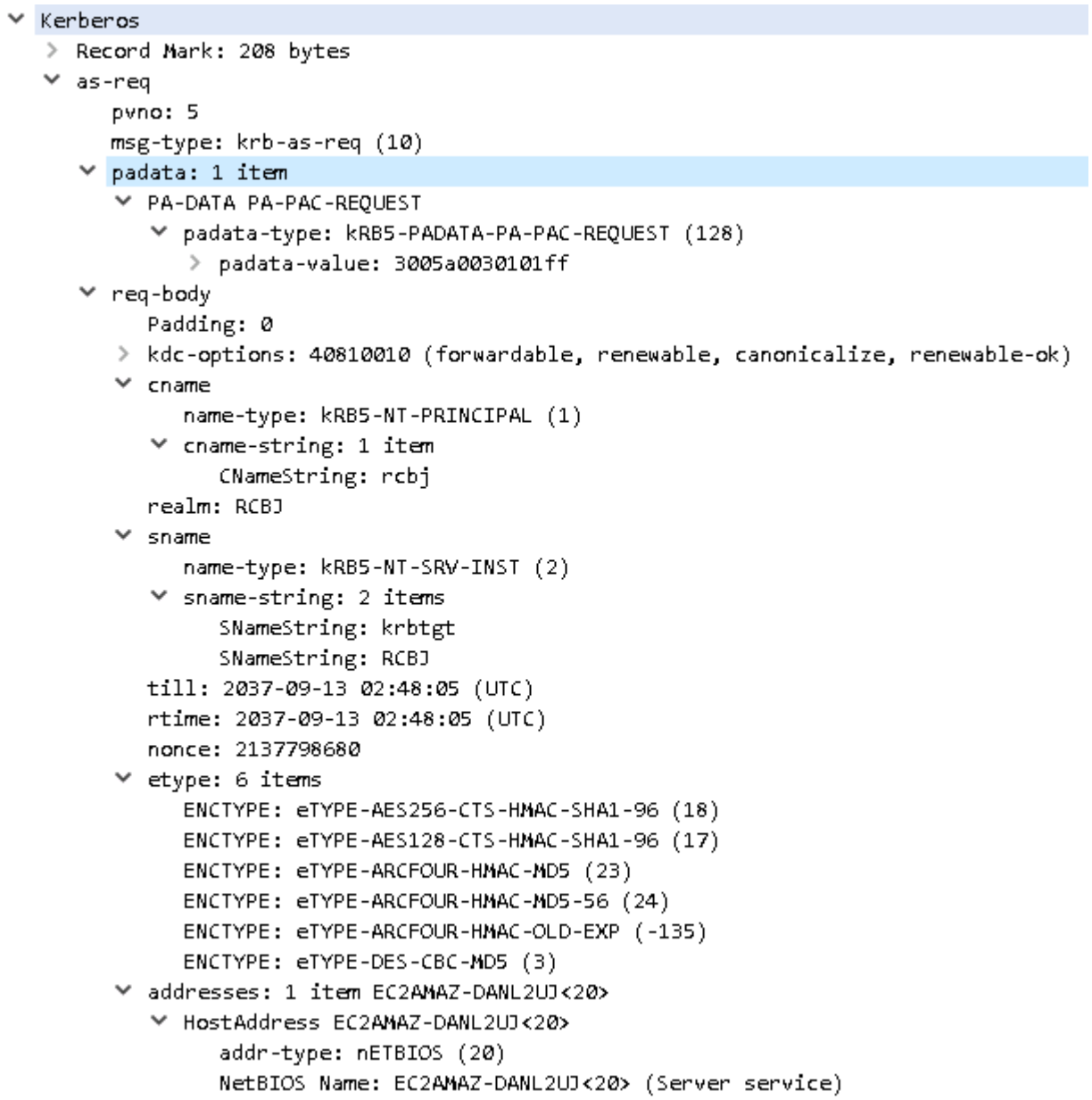


Рисунок 3.8 – Пакет AS-REQ Kerberos

Контролер домену з працюючою службою автентифікації виконує надання попередньої аутентифікації (рис. 3.9).

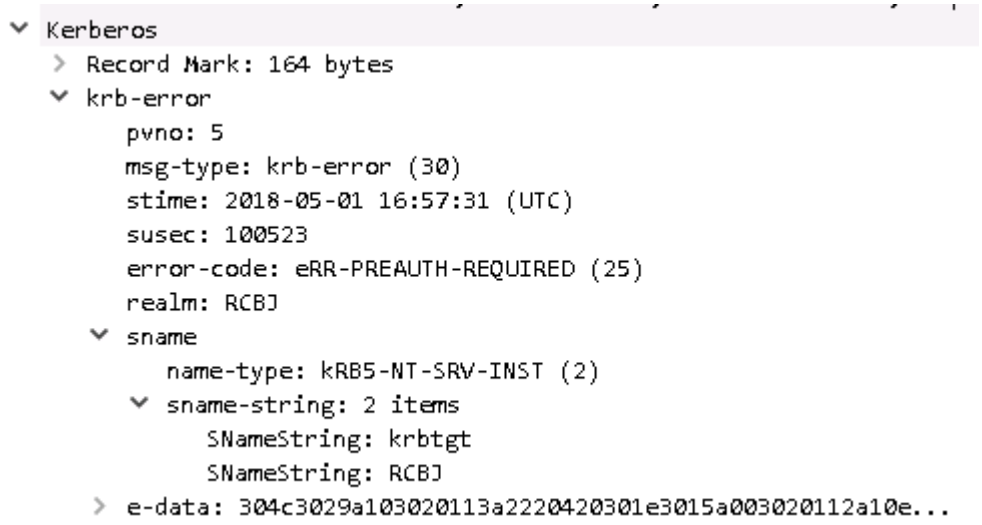


Рисунок 3.9 – Пакет пре-аутентифікації

Відповіді служби уатентифікації з кодами помилок Kerberos описані в RFC 4120.

При підключенні клієнта на цей час з'єднання по протоколу TCP ще закрите. Клієнт шифрує значення актуальної часової мітки і вона надсилається до служби аутентифікації KDC. По суті це перший месседж. Різниця в тому, що вже в ньому наявні поля після преаутентифікації. Служба автентифікації поверне клієнтові відповідь KRB_AS_REP з інформацію про розшифрування зашифрованої частини AS-REP.

Приклад MIT GSS.

Консорціум Kerberos надає простий приклад GSS-API, який ілюструє, як GSS-API може використовуватися в додатку. Клієнт встановлює з'єднання, підтверджує автентифікацію на сервер і отримує зашифрований маркер, який забезпечує повідомлення про помилку або вказівку на успіх. Сервер налаштовує розетку прослуховування, автентифікує клієнта, визначає, чи є клієнт авторизованим і записує зашифрований маркер назад клієнту, вказуючи на успіх або помилку авторизації.

Сторона сервера циклу аутентифікації GSS знаходиться у функції Authenticate на сервіс. Цикл аутентифікації клієнта знайдений у функції Authenticate is client

Цей приклад коду специфічний для Kerberos двома способами. По-перше, сервер переходить у GSS_C_NO_CREDENTIAL у GSS_Accept_sec_context. Перевага цього полягає в тому, що серверу не потрібно знати, яке ім'я використовував клієнт для його звернення. Це добре працює для серверів, які є багатодомними машинами, є частиною пулу балансування навантаження DNS або зареєстровані в декількох сферах. Однак він вводить залежність від Kerberos, оскільки сервіс повинен підтвердити, що ім'я дійсно стосується його. Наприклад, якщо і каталог LDAP, і сервер SMTP працювали на одній машині і використовували цей підхід, каталог LDAP повинен був переконатися, що клієнт не намагався пройти автентифікацію на сервері SMTP на одній машині, і навпаки. Друга залежність від Kerberos вводиться в код сервера через авторизацію клієнта, тобто сервіс повинен розуміти імена Kerberos, щоб авторизувати клієнта.

Додаток, який хотів би бути незалежним від будь-якого конкретного механізму GSS-API, повинен по- різному вирішувати проблему вибору сервісного доступу та авторизації клієнта. Для обробки вибору облікових даних серверів механічно залежним чином програма може або мати лише одне ім'я, за яким відома служба, або могла придбати облікові дані для кількох назв та спробувати GSS_Accept_sec_context по кожному з цих імен. Функція GSS_Export_name забезпечує незалежне від механізму рішення для авторизації клієнта. Ця функція забезпечує двійкове представлення імені, придатного для зберігання в ACL, які можна порівнювати за допомогою двійкового порівняння. Хоча ці підходи добре працюють для незалежних від механізму додатків, підходи, наведені в прикладі, як правило, є більш придатними для конкретних додатків Kerberos.

Приклади Gsstest для крос-платформного GSS-API.

З моменту початкового випуску MIT Kerberos, MIT зробив доступним gss-sample у src / appl / gss-sample у дереві вихідного дерева MIT Kerberos. Спочатку цей код мав бути зразком програми GSS-API, що демонструє, як слід використовувати GSS-API. Однак зразок зростав і розвивався. На сьогодні головним призначенням коду є тестування особливостей GSS-API як частини автоматизованих тестів регресії та подій взаємодії. Код дещо складніший, ніж бажаний, для першого прикладу програми GSS-API. Крім того, код досить старий; він не використовує прототипи ANSI C.

Оскільки код використовувався в тестових сценаріях, він був перенесений на кілька різних платформ і, таким чином, є корисним прикладом для вивчення при написанні кросплатформенної програми GSSAPI. Незважаючи на еволюцію коду, MIT зберігає назву gss-sample для його розповсюдження. Ця назва дуже близька до муфти GSSExa , простий приклад програми, також розповсюджений MIT, про який йде мова в попередньому розділі.

Порт Windows.

Існує порт клієнтського зразка gss до KFW, розташований у src / windows / gss у дереві джерела MIT Kerberos. Цей порт функціонально ідентичний оригінальній версії, але використовує DLL-файли KFW та надає користувальницький інтерфейс для маніпулювання параметрами.

Порт SSPI.

Консорціум Kerberos поширює порт gss-зразка до SSPI. Цей порт використовує той же мережевий протокол, що і версія GSS-API, але використовує SSPI Microsoft для доступу до рідного механізму GSS-API Kerberos Windows Kerberos. Порівняння цього порту з оригіналом демонструє, як схожа база коду може використовуватися як для SSPI, так і для GSS-API.

SASL для Jabber у Pidgin.

Клієнт обміну миттєвими повідомленнями Pidgin (http://www.pidgin.im/) включає підтримку Kerberos у своєму плагіні протоколу Jabber. Pidgin використовує бібліотеку Cyrus SASL для підтримки механізму GSS-API SASL і, таким чином, Kerberos. Інтеграцію SASL можна знайти в libpurple / Protocol / jabber / auth.c у підручниках Pidgin .

Pidgin - приклад програми, яка підтримує захисні рівні TLS та SASL. Інтеграція Pidgin SASL намагається повернутися до інших механізмів аутентифікації, якщо переважні механізми, такі як GSS-API, виходять з ладу. За замовчуванням Cyrus SASL не вдасться пройти автентифікацію, якщо її бажаний механізм не працює. Як було обговорено в розділі Імена федерації та клієнтів, навіть якщо і клієнт, і сервер підтримують Kerberos, автентифікація між ними може не працювати. У таких ситуаціях поведінка за замовчуванням Cyrus SASL може порушити автентифікацію.

SASL в Thunderbird.

Thunderbird Mail Reader (http://www.mozilla.org/) підтримує аутентифікацію GSS-API SASL для пошуку та подання пошти. Інтеграція Thunderbird має значно інший підхід, ніж інтеграція Pidgin. Thunderbird зосереджується на забезпеченні постійного досвіду на всіх платформах. Як такий, існує необхідний мінімум, щоб мінімізувати зовнішні залежності, і тому Cyrus SASL не був відповідним рішенням. Натомість Thunderbird створює власну оболонку SASL навколо викликів GSS-API.

Thunderbird розробники також хотіли уникнути залежності часу зв'язку від конкретної реалізації Kerberos, оскільки в цільовій системі може бути інша реалізація, ніж у системі збірки. Отже, Thunderbird динамічно завантажує бібліотеку GSSAPI і отримує з неї символи.

Інтеграція Thunderbird Kerberos забезпечує вичерпну ілюстрацію того, як справді крос-платформенні програми можна створювати та впроваджувати на практиці.

4 СПЕЦІАЛЬНА ЧАСТИНА
РОБОТА З ПРОГРАОЮ ЗАХВАТУ TCP-ПАКЕТІВ WIRESHARK

4.1 Загальні відомості про програму захоплення пакетів

Навіть поверхневе знання програми Wireshark і її фільтрів на порядок заощадить час при усуненні проблем мережевого або прикладного рівня. Wireshark корисний для багатьох завдань в роботі мережевого інженера, фахівця з безпеки або системного адміністратора. Ось кілька прикладів використання:

Усунення неполадок мережного підключення:

- Візуальне відображення втрати пакетів
- Аналіз ретрансляції TCP
- Графік по пакетам з великою затримкою відповіді

Дослідження сесій прикладного рівня (навіть при шифруванні з допомогою SSL / TLS, см. Нижче)

- Повний перегляд HTTP- сесій, включаючи всі заголовки і дані для запитів і відповідей
- Перегляд сеансів Telnet, перегляд паролів, введених команд і відповідей
- Перегляд трафіку SMTP і POP3, читання листів

Усунення неполадок DHCP з даними на рівні пакетів

- Вивчення трансляцій широковещательного DHCP
- Другий крок обміну DHCP (DHCP Offer) з адресою та параметрами
- Клієнтський запит по запропонованим адресою
- Ack від сервера, що підтверджує запит

Витяг файлів з сесій HTTP

– Експорт об'єктів з HTTP, таких як JavaScript, зображення або навіть виконувані файли

Витяг файлів з сесій SMB

– Аналогічно опції експорту HTTP, але витяг файлів, переданих по SMB, протоколу загального доступу до файлів в Windows

Виявлення і перевірка шкідливих програм

– Виявлення аномального поведінки, яке може вказувати на шкідливе ПЗ

– Пошук незвичайних доменів або кінцевих IP

– Графіки введення-виведення для виявлення постійних з'єднань (маячків) з керуючими серверами

– Отфільтровка « нормальних » даних і виявлення незвичайних

– Витяг великих DNS- відповідей і інших аномалій, які можуть вказувати на шкідливе ПЗ

Перевірка сканування портів та інших типів сканування на уразливості

– Розуміння, який мережевий трафік надходить від сканерів

– Аналіз процедур по перевірці вразливостей, щоб розрізняти хибнопозитивні і помилково негативні спрацьовування

Ці приклади - лише вершина айсберга. У керівництві ми розповімо, як використовувати настільки потужний інструмент.

4.2 Установка Wireshark

Wireshark працює на різних операційних системах і його нескладно встановити. Згадаємо лише Ubuntu Linux, Centos і Windows.

Установка на Ubuntu або Debian

Apt-get update

Apt-get install wireshark tshark

Установка на Fedora або CentOS

Yum install wireshark-gnome

Установка на Windows

На сторінці завантаження лежить виконуваний файл для установки. Досить просто ставиться і драйвер захоплення пакетів, з допомогою якого мережева карта переходить в «нерозбірливий» режим (promiscuous mode дозволяє приймати всі пакети незалежно від того, кому вони адресовані).

4.3 Робота з програмою

З першим перехопленням ви побачите в інтерфейсі Wireshark стандартний шаблон і подробиці про пакет. Як тільки захопили сесію HTTP, зупиніть запис і пограйте з основними фільтрами і настройками Analyze | Follow | HTTP Stream. Назви фільтрів говорять самі за себе. Просто вводите відповідні вирази в рядок фільтра (або в командну рядок, якщо використовуєте tshark). Основне перевага фільтрів - в видаленні шуму (трафік, який нам не Інтерест). Можна фільтрувати трафік по MAC-адресу, IP-адресою, підмережі або протоколу. Самий простий фільтр - ввести http, так що буде відображатися тільки трафік HTTP (порт tcp 80).

Приклад фільтру по IP-адресах

```
ip.addr == 192.168.0.5
! (ip.addr == 192.168.0.0/24)
```

Приклад фільтру по протоколу

```
tcp
udp
tcp.port == 80 || udp.port == 80
http
not arp and not (udp.port == 53)
```

Спробуйте зробити комбінацію фільтрів, яка показує весь вихідний трафік, крім HTTP і HTTPS, який прямує за межі локальної мережі. Це хороший спосіб виявити програмне забезпечення (навіть шкідливе), яке взаємодіє з інтернетом по незвичайним протоколам.

Як тільки ви захопили кілька HTTP- пакетів, можна застосувати на одному з них пункт меню Analyze | Follow | HTTP Stream. Він покаже цілком сесію HTTP. У цьому новому вікні ви побачите HTTP- запит від браузера і HTTP- відповідь від сервера.

За замовчуванням Wireshark налаштований перетворювати мережеві адреси в консолі. Це можна змінити в настройках. Edit | Preferences | Name Resolution | Enable Network Name Resolution Як і в разі tcpdump, процедура резолвінг сповільнить відображення пакетів. Також важливо розуміти, що при оперативному захопленні пакетів DNS- запити з вашого хоста стануть додатковим трафіком, який можуть перехопити.

Якщо Wireshark скомпільовано з підтримкою GeoIP і у вас є безкоштовні бази Maxmind, то програма може визначати місце розташування комп'ютерів по їх IP-адресами. Перевірте в About | Wireshark, що програма скомпільована з тієї версією, яка у вас в наявності. Якщо GeoIP присутній в списку, то перевірте наявність на диску баз GeoLite City, Country і ASNum. Вкажіть розташування баз в меню Edit | Preferences | Name Resolution. Перевірте систему на дампі трафіку, вибравши опцію Statistics | Endpoints | IPv4. В колонках праворуч повинна з'явитися інформація про місцезнаходження та ASN для IP-адреси. Інша функція GeoIP - фільтрація трафіку по місцю розташування з допомогою фільтра ip.geoip. Наприклад, так можна виключити трафік з конкретної ASN. Нижчезазначених команда виключає пакети від мережевого блоку ASN 63949 (Linode).

Один з способів розшифровки сесій SSL / TLS - використовувати закритий ключ з сервера, до якого підключений клієнт. Звичайно, у вас не завжди є доступ до приватного ключу. Але є інший варіант простого перегляду

трафіку SSL / TLS на локальній системі. Якщо Firefox або Chrome завантажуються з допомогою спеціальної змінної середовища, то симетричні ключі окремих сеансів SSL / TLS записані в файл, який Wireshark може прочитати. З допомогою цих ключів Wireshark покаже повністю розшифровану сесію.

5 ОБҐРУНТУВАННЯ ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ

Метою цього розділу дипломної роботи є здійснення економічних розрахунків, спрямованих на визначення економічної ефективності від розробки, а також прийняття рішення щодо подальшого розвитку і впровадження або ж недоцільність впровадження відповідної розробки.

Передбачається, що описаний в роботі метод аналізу алгоритмів шифрування може бути імплементовано у вигляді спеціального програмного продукту. Розробка такого продукту вимагатиме певних затрат. Тому розрахуємо ці затрати.

Для здійснення оцінки потрібно зробити розрахунки трудомісткості кожної операції.

5.1 Визначення стадій технологічного процесу та загальної тривалості проведення НДР

Витрати часу по окремих операціях технологічного процесу відображені в таблиці 5.1.

Таблиця 5.1 – Операції технологічного процесу та час їх виконання

№	Назва операції (стадії)	Виконавець	Середній час виконання операції, год.
1.	Витрати праці на підготовку опису задачі	інженер	12
2.	Витрати праці на розробку проекту	інженер	20
3.	Витрати праці на розробку структури системи	інженер	15
4.	Витрати праці на створення системи по вибраному проекту та структурі	інженер	77
5.	Витрати праці на підготовку документації	інженер	15
6.	Витрати праці на відлагодження роботи зпроектованої системи при комплексній відладці	інженер	46
Разом			185

Загальні затрати на дипломний проект становить 185 годин.

5.2 Визначення витрат на оплату праці та відрахувань на соціальні заходи

Відповідно до Закону України “Про оплату праці” заробітна плата – це “винагорода, обчислена, як правило, у грошовому виразі, яку власник або уповноважений ним орган виплачує працівникові за виконану ним роботу”.

Розмір заробітної плати залежить від складності та умов виконуваної роботи, професійно-ділових якостей працівника, результатів його праці та господарської діяльності підприємства. Заробітна плата складається з основної та додаткової оплати праці.

Основна заробітна плата нараховується на виконану роботу за тарифними ставками, відрядними розцінками чи посадовими окладами і не залежить від результатів господарської діяльності підприємства.

Додаткова заробітна плата – це складова заробітної плати працівників, до якої включають витрати на оплату праці, не пов’язані з виплатами за фактично відпрацьований час. Нараховують додаткову заробітну плату залежно від досягнутих і запланованих показників, умов виробництва, кваліфікації виконавців. Джерелом додаткової оплати праці є фонд матеріального стимулювання, який створюється за рахунок прибутку.

При розрахунку заробітної плати кількість робочих днів у місяці слід в середньому приймати – 24,5 дні/міс., або ж 196 год./міс. (тривалість робочого дня – 8 год.).

Місячний оклад кожного працівника слід враховувати згідно існуючих на даний час тарифних окладів. Згідно закону України «Про Державний бюджет України на 2018 рік», зокрема Статтею восьмою мінімальна заробітна плата у погодинному розмірі встановлена у розмірі 22,41 грн. Рекомендовані

тарифні ставки: керівник дипломної роботи – 30,00...50,00 грн./год., інженер – 22,41...30,00 грн./год., консультант – 22,41...30,00 грн./год., технік – 22,41...30,00 грн./год., лаборант – 22,41...25,00 грн./год.

Основна заробітна плата розраховується за формулою:

$$З_{осн.} = Т_с \cdot К_г \,, \qquad (5.1)$$

де $Т_с$ – тарифна ставка, грн.;

$К_г$ – кількість відпрацьованих годин.

Оскільки всі види робіт в даному проекті виконує інженер, то основна заробітна плата буде розраховуватись тільки за однією формулою

$$З_{осн.} = 30 \cdot 175 = 5550 \text{ грн.}$$

Додаткова заробітна плата становить 10–15 % від суми основної заробітної плати.

$$З_{дод.} = З_{осн.} \cdot К_{допл.}, \qquad (5.2)$$

де $К_{допл.}$ – коефіцієнт додаткових виплат працівникам, 0,1–0,15 (візьмемо його рівним 0,15).

$$З_{дод.} = 5550 \cdot 0{,}15 = 832{,}50 \text{ грн.}$$

Звідси загальні витрати на оплату праці ($В_{о.п.}$) визначаються за формулою:

$$В_{о.п.} = З_{осн.} + З_{дод.}. \qquad (5.3)$$

$B_{о.п.}$=5550+832,50 = 6382,50 грн.

Крім того, слід визначити відрахування на соціальні заходи:

1) ЄСВ + ПДФО 22 %;
2) військовий збір – 1,5 %.

У сумі зазначені відрахування становлять 23,5 %.

Отже, сума відрахувань на соціальні заходи буде становити:

$$B_{с.з.} = Ф_{оп} \cdot 0{,}235, \qquad (5.4)$$

де $Ф_{ОП}$ – фонд оплати праці, грн.

$$B_{с.з.} = 6382{,}50 \cdot 0{,}235 = 1499{,}89 \text{ грн.}$$

Проведені розрахунки витрат на оплату праці зведемо у таблицю 5.2.

Таблиця 5.2 – Зведені розрахунки витрат на оплату праці

№ п/п	Категорія працівник-ків	Основна заробітна плата, грн.			Додаткова заробітна плата, грн.	Нарахув. на $Ф_{ОП}$, грн.	Всього витрати на оплату праці, грн. *6=3+4+5*
		Тарифна ставка, грн.	К-сть відпра-цьов. год.	Фактично нарах. з/пл., грн.			
А	Б	1	2	3	4	5	6
1	інженер	30	185	5550	832,50	1499,89	7882,39

Загальні витрати на оплату праці становить 7882,39 грн.

5.3 Розрахунок матеріальних витрат

Матеріальні витрати визначаються як добуток кількості витрачених матеріалів та їх ціни:

$$M_{B\,i} = q_i \cdot p_i, \tag{5.5}$$

де: q_i – кількість витраченого матеріалу i-го виду;

p_i – ціна матеріалу i-го виду.

Звідси, загальні матеріальні витрати можна визначити:

$$З_{м.в.} = \sum M_{B\,i}. \tag{5.6}$$

Проведені розрахунки занесемо у таблицю 5.3. Для розробки ПЗ передбачається покупка Visual Studio Team Foundation Server CAL SNGL LicSAPk OLP NL UsrCAL 2017, вартість якого на сьогодні становить 19400 грн.

Таблиця 5.3 – Зведені розрахунки матеріальних витрат

Найменування матеріальних ресурсів	Одиниця виміру	Норма витрат	Ціна за одиницю, грн	Затрати матеріалів, грн	Транс-портно-заготівель-ні витрати, грн	Загальна сума витрат на матеріа-ли, грн
1. Основні матеріали						
Програмне забезпечення	комп.	1	19400,00	19400,00	–	19400,00
2. Допоміжні матеріали						
Папір формату А4	шт.	200	0,18	36	–	36
Разом:						19436,00

Загальні матеріальні затрати становлять 19436,00 гривень.

5.4 Розрахунок витрат на електроенергію

Затрати на електроенергію 1-ці обладнання визначаються за формулою:

$$З_е = W \cdot T \cdot S \ , \qquad (5.7)$$

де W – необхідна потужність, кВт;

Т – кількість годин роботи обладнання;

S – вартість кіловат-години електроенергії.

Вартість кіловат-години електроенергії слід приймати згідно існуючих на даний час тарифів. Отже, 1 кВт з ПДВ коштує 2,42 грн.

Потужність комп'ютера для створення проекту – 550 Вт, кількість годин роботи обладнання згідно таблиці 5.1 – 169 годин.

Тоді, $З_е = 0{,}55 \cdot 185 \cdot 2{,}42 = 246{,}24$ грн.

5.5 Розрахунок суми амортизаційних відрахувань

Характерною особливістю застосування основних фондів у процесі виробництва є їх відновлення. Для відновлення засобів праці у натуральному виразі необхідне їх відшкодування у вартісній формі, яке здійснюється шляхом амортизації.

Амортизація – це процес перенесення вартості основних фондів на вартість новоствореної продукції з метою їх повного відновлення.

Комп'ютери та оргтехніка належать до четвертої групи основних фондів. Для цієї групи річна норма амортизації дорівнює 60 % (квартальна – 15 %).

Для визначення амортизаційних відрахувань застосовуємо формулу:

$$A = \frac{Б_В \cdot Н_А}{100\%}, \qquad (5.8)$$

де A – амортизаційні відрахування за звітний період, грн.;

$Б_В$ – балансова вартість групи основних фондів на початок звітного періоду, грн.;

$Н_А$ – норма амортизації, %.

Для даного проекту засобом розробки є комп'ютер. Його сума становить 20000 грн. Отже, амортизаційні відрахування будуть рівні:

$$А = 20000 \cdot 5\% / 100\% = 1000 \text{ грн.}$$

Оскільки робота виконувалась 175 годин, то амортизаційні відрахування будуть становити:

$$А = 1000 \cdot 175 / 150 = 1166{,}67 \text{ грн.}$$

5.6 Обчислення накладних витрат

Накладні витрати пов'язані з обслуговуванням виробництва, утриманням апарату управління спілкою та створення необхідних умов праці.

В залежності від організаційно-правової форми діяльності господарюючого суб'єкта, накладні витрати можуть становити 20 – 60 % від суми основної та додаткової заробітної плати працівників.

$$Н_в = В_{о.п.} \cdot 0{,}2 \ldots 0{,}6, \qquad (5.9)$$

де $Н_В$ – накладні витрати.

Отже, накладні витрати:

$Н_В = 6382{,}50 \cdot 0{,}2 = 1276{,}50$ грн.

5.7 Складання кошторису витрат та визначення собівартості НДР

Результати проведених вище розрахунків зведемо у таблицю 5.4.

Таблиця 5.4 – Кошторис витрат на НДР

Зміст витрат	Сума, грн.	В % до загальної суми
Витрати на оплату праці (основну і додаткову заробітну плату)	6382,50	21,3%
Відрахування на соціальні заходи	1499,89	5,0%
Матеріальні витрати	19436,00	64,8%
Витрати на електроенергію	246,24	0,8%
Амортизаційні відрахування	1166,67	3,9%
Накладні витрати	1276,50	4,3%
Собівартість	30007,79	100,0%

Собівартість (C_B) проекту розрахуємо за формулою:

$$C_в = B_{о.п.} + B_{с.з.} + З_{м.в.} + З_в + А + Н_в. \quad (5.10)$$

Отже, собівартість проекту дорівнює:

$C_в$ = 6382,50 + 1499,89 + 19436 + 246,24 +1166,67 + 1276,50 = 30007,79 грн.

5.8 Розрахунок ціни проекту

Ціну НДР можна визначити за формулою:

$$Ц = \frac{C_B \cdot (1 + P_{рен}) + К \cdot B_{н.і.}}{К} \cdot (1 + ПДВ), \quad (5.11)$$

де $P_{рен.}$ – рівень рентабельності, 30 %;

К – кількість замовлень, од. (встановлюється лише при розробці програмного продукту та мікропроцесорних систем);

$В_{н.і.}$ – вартість носія інформації, грн. (встановлюється лише при розробці програмного продукту);

ПДВ – ставка податку на додану вартість, (20 %).

Оскільки розробка є прикладною, і використовуватиметься тільки для одного підприємства, то для розрахунку ціни не потрібно вказувати коефіцієнти *К* та $В_{і.н}$, оскільки їх в даному випадку не потрібно.

Тоді, формула для обчислення ціни розробки буде мати вигляд:

$$Ц = С_В \cdot (1 + Р_{рен}) \cdot (1 + ПДВ). \qquad (5.12)$$

Звідси ціна на проект складе:

Ц = $С_В$ · (1+0,3)(1+0,2) = 46812,16 грн.

5.9 Визначення економічної ефективності і терміну окупності капітальних вкладень

Ефективність виробництва – це узагальнене і повне відображення кінцевих результатів використання робочої сили, засобів та предметів праці на підприємстві за певний проміжок часу.

Економічна ефективність ($Е_р$) полягає у відношенні результату виробництва до затрачених ресурсів:

$$Е_р = П / С_В, \qquad (5.13)$$

де *П* – прибуток;

$С_В$ – собівартість.

Плановий прибуток ($П_{пл}$) знаходимо за формулою:

$$П_{пл} = Ц - С_{В}. \quad (5.14)$$

Розраховуємо плановий прибуток:

$П_{пл}$ = 46812,16 – 30007,79 = 16804,36 грн.

Отже, формула для визначення економічної ефективності набуде вигляду:

$$E_{р} = \frac{П_{пл}}{Св}. \quad (5.15)$$

Тоді, $E_р$ = 16804,36 / 30007,79 = 0,56

Поряд із економічною ефективністю розраховують термін окупності капітальних вкладень ($T_р$):

$$T_р = 1/ E_р , \quad (5.16)$$

Термін окупності дорівнює:

$T_р$ = 1/0,56 = 1,8 роки.

В цьому розділі дипломної роботи було розраховано основні техніко-економічні показники проекту (див. таблицю 5.5).

Розраховане значення економічної ефективності становить 0,56 що є високим значенням.

Так само нормальним є термін окупності. Для даного продукту він становить 1,8 роки.

Таблиця 5.5 – Техніко-економічні показники НДР

№ п/п	Показник	Значення
1.	Собівартість, грн.	30007,79
2.	Плановий прибуток, грн.	16804,36
3.	Ціна, грн.	46812,16
4.	Економічна ефективність	0,56
5.	Термін окупності, рік	1,8

Отже, даний проект може бути впроваджений та мати подальший розвиток, оскільки він є економічно вигідним за всіма основними техніко-економічними показниками.

6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

6.1 Оцінка стійкості роботи об’єкту економіки до впливу вражаючих факторів при надзвичайних ситуаціях

В наш час стрімкого розвитку техніки ростуть і загрози різноманітних техногенних аварій, які можуть завдати значної шкоди як економічному потенціалу окремого регіону чи цілої країни. Актуальною, особливо в останній час, є також проблема тероризму.

Здатність підприємств протидіяти таким шкідливим впливам та аваріям характеризується стійкістю роботи об’єкту в надзвичайних ситуаціях.

Зрозуміло, що збереження обладнання підприємства та його будівель має значення тільки тоді, коли в надзвичайних ситуаціях буде збережено також персонал, який зможе працювати на цьому обладнанні. Тому інженерний захист населення та персоналу займає важливе місце серед заходів по забезпеченню стійкості підприємства в надзвичайних ситуаціях. Підтвердження цій думці знаходимо і в літературі з цивільної оборони.

Принципами стійкості роботи промислових підприємств в надзвичайних ситуаціях є єдина нормативна директивна база, яка включає:

– Конституцію України;

– Закон про цивільну оборону України;

– Закон України про захист населення і території від надзвичайних ситуацій техногенного та природного характеру;

– Положення по цивільній обороні;

– нормативні документи про стійкість роботи об’єктів;

– директиви начальника штабу цивільної оборони України.

Під стійкістю роботи промислових підприємств розуміють їх можливість в умовах надзвичайних ситуацій мирного і воєнного часу

виробляти продукцію в запланованому обсязі і номенклатурі, а при слабких пошкодженнях відновлювати виробництво в мінімальні терміни.

Стійкість роботи промислового підприємства складається з:

– стійкості інженерно-технічного комплексу (будівель, споруд, систем енерго-, газо-, водозабезпечення, технічного обладнання і т.п.) до дії зовнішніх факторів при аваріях, катастрофах, стихійному лиху, а також при застосуванні щодо них сучасної зброї;

– стійкості виробничої діяльності (захист виробничого персоналу, надійність систем управління, постачання, поновлення роботи в найкоротші терміни).

Під стійкістю роботи об’єктів, які не виробляють матеріальних цінностей, розуміють їх можливість виконувати свої функції в умовах надзвичайних ситуацій.

Фактори, від яких залежить стійкість роботи об’єктів в надзвичайних ситуаціях мирного і воєнного часу:

– надійність захисту робітників і службовців;

– безпечність розташування об’єкту відносно зон можливих руйнувань;

– можливість інженерно-технічного комплексу протистояти ударній хвилі будь-якого вибуху і уражаючим діям ядерної зброї;

– безперервність постачання електроенергією, паливом, сировиною, газом і всім необхідним для впуску продукції;

– надійність керування виробництвом, силами і засобами цивільної оборони;

– підготовленість підприємства до поновлення виробництва та проведення РіІНР.

Із перерахованих факторів випливають такі шляхи і засоби підвищення стійкості роботи промислових підприємств і галузей господарства України:

– нагромадження фондів захисних споруд і засобів індивідуального захисту;

– будівництво важливих підприємств за межами зон можливих руйнувань;

– будівництво підприємств-дублерів;

– розширення шляхів сполучення і розвиток всіх видів транспорту;

– підсилення і дублювання енергетичних потужностей;

– розширення зв'язків між галузями промисловості і підприємствами;

– утворення матеріально-технічних резервів;

– підтримування сил цивільної оборони в постійній готовності.

6.2 Організація робіт і заходів для дослідження стійкості об'єкту економіки

Основою для проведення заходів по підвищенню стійкості роботи промислових підприємств надзвичайних ситуаціях є вимоги норм інженерно-технічних заходів цивільної оборони.

З метою підвищення стійкості роботи підприємств в надзвичайних ситуаціях мирного і воєнного часу проводяться дослідження по оцінці стійкості. Внаслідок дослідження повинні бути вивчені наступні питання:

– захист виробничого персоналу;

– захист засобів виробництва;

– стійкість виробничої діяльності при стихійному лиху, аваріях, катастрофах, а також при застосуванні противником сучасної зброї;

– готовність до відновлення порушеного виробництва.

Головна мета дослідження втому, щоб на основі вивчення всіх умов, які визначають виробничу діяльність підприємства в надзвичайних ситуаціях, виробити заходи, які сприяли б підвищенню стійкості його роботи.

Оцінка стійкості підприємства має на меті:

– визначення стійкості його роботи до уражаючих дій сучасної зброї, стихійного лиха, аварій, катастроф;

– визначення можливості виникнення вторинних уражаючих факторів і оцінка характеру ураження від цих факторів;

– аналіз надійності систем управління, постачання і промислових зв’язків.

6.2.1 Проведення дослідження стійкості роботи об’єкту економіки

Дослідження стійкості роботи в надзвичайних ситуаціях здійснюється штабом цивільної оборони підприємства і розрахунково-дослідними групами, які створюються з інженерно-технічного персоналу підприємства. Для роботи в цих групах можуть залучатись працівники науково-дослідних і проектно-конструкторських організацій.

На промисловому підприємстві можна утворювати такі розрахунково-дослідні групи:

– головного технолога;

– головного механіка;

– головного енергетика;

– відділу капітального будівництва;

– заступника директора з постачання і збуту і інші.

Кожна група проводить дослідження підвідомчого їй господарства (елементів об’єкту), оцінює їх стійкість і планує інженерно-технічні заходи в умовах надзвичайних ситуацій мирного і воєнного часу.

Для загального керівництва і координації роботи розрахунково-дослідних груп утворюється група керівництва на чолі з головним інженером.

Робота по дослідженню стійкості планується за трьома етапами:

– підготовчий – 10–15 днів;

– період досліджень – 1–2 місяці;

– заключний – 7–10 днів.

За висновками досліджень і пропозиціями керівництво цивільної оборони підприємства складає звіт з висновками, рекомендаціями і планами заходів по підвищенню стійкості роботи в надзвичайних ситуаціях мирного і воєнного часу, пересилає в штаб цивільної оборони для узгодження і затвердження.

6.2.2 Параметри об’єктів економіки, котрі враховуються при визначенні оцінки інженерного захисту робітників і службовців

Забезпечення надійного захисту робітників і службовців об’єкту в надзвичайних ситуаціях – один із основних шляхів підвищення стійкості роботи. В комплексі заходів по реалізації цього шляху важливе місце займають заходи по інженерному захисту робітників і службовців.

Інженерний захист робітників і службовців – це захист з використанням інженерних споруд: сховищ, ПРУ, простіших укриттів. Він досягається шляхом завчасного проведення інженерних заходів по будівництву і обладнанню захисних споруд з врахуванням умов розташування об’єкту і вимог будівельних норм і правил.

Успішне виконання завдань інженерного захисту можливе при дотриманні таких умов:

– загальна місткість захисних споруд дозволяє сховати найбільшу працюючу зміну;

– захисні властивості споруд відповідають потребам (забезпечують захист людей від надлишкового тиску ударної хвилі і радіоактивного випромінювання, які очікуються);

– фільтровентиляційне обладнання захисних споруд забезпечує життєдіяльність людей протягом встановленого терміну безперервного перебування їх в захисних спорудах;

– розміщення захисних споруд відносно місця роботи.

З перерахованого випливає, що оцінка інженерного захисту робітників і службовців підприємства полягає у визначенні показників, які характеризують можливість інженерних споруд забезпечити ці умови.

На основі висновків оцінки інженерного захисту робітників і службовців підприємства визначають заходи по підвищенню надійності захисту, а, відповідно, і по підвищенню стійкості роботи підприємства в надзвичайних ситуаціях.

6.3 Розрахунок штучної вентиляції

Загальнообмінна вентиляція застосовується для видалення надлишкового тепла при відсутності токсичних виділень, а також у випадках, коли характер технологічного процесу та особливості виробничого устаткування виключають можливість використання місцевої витяжної вентиляції.

В умовах промислового виробництва найбільш розповсюджена припливно-витяжна система вентиляції із загальним припливом в робочу зону та місцевою витяжкою шкідливих речовин безпосередньо з місць їх утворення.

Місцева витяжна вентиляція здійснюється за допомогою місцевих витяжних зонтів, всмоктуючих панелей, витяжних шаф, бортових відсмоктувачів.

Основне завдання розрахунку загальнообмінних систем штучної вентиляції – визначити кількість повітря, що необхідно подати і вилучити з приміщення.

Для приміщень, де немає шкідливих виділень (або кількість їх незначна) приплив (витяжку) повітря можна визначити за кратністю повітрообміну (k) – відношенням об'єму вентиляційного повітря L ($м^3$/год) до об'єму приміщення V ($м^3$):

$$k = \frac{L}{V_п}. \qquad (6.1)$$

Для нашого випадку при розмірах приміщення 3м · 6м · 3 м маємо об’єм приміщення $V_п$=54 м^3. Об’єм повітря на одного працівника при трьох працюючих становить 18 м^3.

Для приміщень без шкідливих виділень та надлишкового тепла можна використати формулу:

$$L = l \times n, \qquad (6.2)$$

де *l* – мінімальне подання повітря на одного працівника відповідно до санітарних норм (при об'ємі приміщення, що припадає на одного працівника, до 20 м^3 – 30 м^3/год, а при об'ємі більше 20 м^3 – 20 м^3/год);

n– кількість працівників в приміщенні.

Отже, для трьох працюючих в приміщенні згідно (6.2) L=30·3 = 90м^3.

Тепер кратність повітрообміну згідно (6.1) становить 90/54=1,67.

Використовуючи спеціальну довідникову літературу [ ], можна встановити для даного значення кратності повітрообміну об’єм припливного повітря. Схему вентиляції зображено на рисунку 6.1.

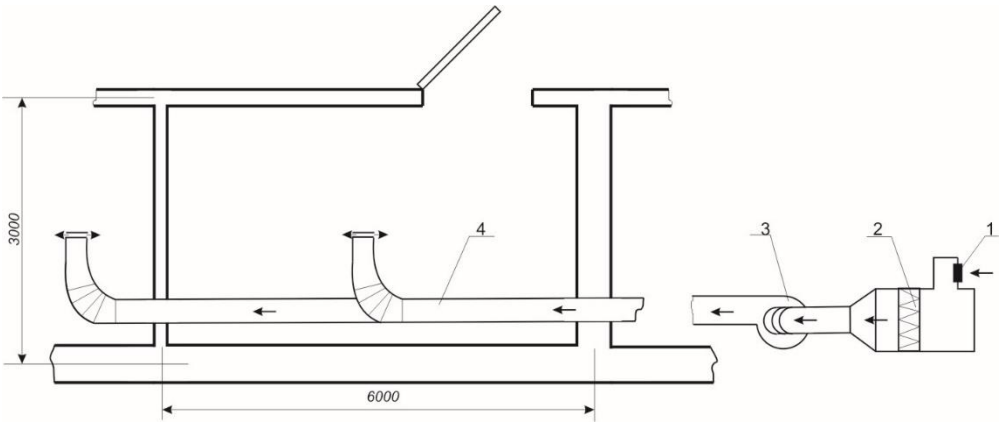


Рисунок 6.1 – Схема вентиляції приміщення:
1 – повітрозабірний пристрій; 2 – очисний фільтр; 3 – вентилятор;
4 – система повітропроводів та припливних патрубків

6.4 Електробезпека

Негативні дії електричного струму відбуваються під час дії електричного струму (включення людини в електричну мережу). Існує чотири особливості:

- перша – відсутність зовнішніх ознак загрозливої небезпеки ураження електричним струмом. Людина не може побачити, почути, відчути чи якось інакше завчасно виявити можливість ураження;
- друга – тяжкість електротравм. Втрата працездатності від електротравм, як правило, буває довгою, з можливим летальним наслідком;
- третя особливість полягає в тому, що струми промислової частоти величиною 10...25 мА можуть викликати інтенсивні судороги м'язів, внаслідок чого відбувається так зване «приковування» до струмопровідних частин. Людина в цьому випадку не може самостійно звільнитися від дії електричного струму;
- четверта особливість визначається можливістю подальшого механічного травмування. Наприклад, людина працювала на висоті, була уражена електрострумом, знепритомніла і впала.

Дія електричного струму на живу тканину має своєрідний та різносторонній характер. Проходячи через тіло людини, електричний струм чинить термічну, електролітичну і механічну (динамічну) та біологічну дію.

Термічна дія струму виявляється в опіках окремих ділянок тіла, нагріванні до високої температури кровоносних судин, нервів, серця, мозку та інших органів, які перебувають на шляху струму, що спричиняє в них суттєві розлади.

Електролітична дія струму виявляється в розкладанні органічної рідини, у тому числі крові, що супроводжується значними змінами її складу, а також тканини в цілому.

Механічна (динамічна) дія струму виявляється в розшаруванні, розриві та інших подібних пошкодженнях різних тканин організму, в тому числі м'язової тканини, стінок кровоносних судин та судин легеневої тканини тощо внаслідок електродинамічного ефекту, а також миттєвого вибухоподібного утворення пари від перегрітої струмом тканинної рідини та крові.

Біологічна дія струму є специфічним процесом, що проявляється в подразненні та збудженні живих тканин організму, а також в порушенні внутрішніх біоелектричних процесів, які протікають в нормально діючому організмі та найтісніше пов'язані з його життєвими функціями.

Вказана багатогранність дій електричного струму на організм людини нерідко призводить до різних електротравм, які умовно можна звести до двох видів: місцевих електротравм, коли виникає місцеве пошкодження організму, та до загальних електротравм, так званих електричних ударів, коли уражається (або створюється загроза ураження) весь організм внаслідок порушення нормальної діяльності життєво важливих органів та систем.

Електричний струм викликає загальну рефлекторну реакцію нервової і серцево-судинної системи. Результат електротравми залежить як від умов зовнішнього середовища так і від параметрів організму людини. Ступінь поразки людини залежить від роду і величини напруги і струму, частоти

електричного струму, шляху струму через людину, тривалості дії й умов зовнішнього середовища.

Для запобігання ураження людини електричним струмом застосовують захисне заземлення. Захисним заземленням називається навмисне електричне з'єднання з землею чи її еквівалентом металевих не струмоведучих частин, що можуть виявитися під напругою.

Пристрій, що заземлює, складається з одного чи декількох заземлюючих, металевих елементів, занурених на визначену глибину в ґрунт і провідників, що заземлюють, з’єднуюче устаткування, що заземлюється, із заземлювачем. Принцип дії захисного заземлення заснований на зниженні напруги щодо землі до припустимих рівнів напруги дотику.

Згідно правил будови електроустановок при напрузі до 1000 В опір пристрою, що заземлює, повинне бути не більш 4 Ом, при напрузі понад 1000 В опір повинний бути не більш 10 Ом.

7 ЕКОЛОГІЯ

7.1 Електромагнітне забруднення довкілля, його вплив на людину. Шляхи його зменшення

Несприятливий вплив на організм людини мають і електромагнітні випромінювання промислової частоти (50 Гц) та частот радіохвильового діапазону. У помешканнях електромагнітні поля створюють: радіоапаратура, телевізори, холодильники тощо, що має певну небезпеку. Справа в тому, що кожен внутрішній орган працює на певній частоті, наприклад, серце – біля 700 Гц (коливань в секунду), мозок у стані сну – 10 Гц, бадьорості – 50 Гц ін. Якщо поруч знаходиться постійне джерело електромагнітного випромінювання, яке працює на аналогічній (чи є кратною) частоті, що може призвести до збільшення або зменшення нормальної частоти роботи органу. Наслідком цього може бути головний біль, порушення сну, перевтома, навіть загроза виникнення стенокардії. Найбільш небезпечне випромінювання, коли людина (а особливо дитина) спить.

Безперечно, обійтися без електропобутових приладів неможливо, та й не потрібно. Головне – дотримуватись певних правил:

– у спальні не варто встановлювати комп’ютер, “базу” для радіотелефону, а також вмикати на ніч пристрої для підзарядки батарейок та акумуляторів;

– телевізор, музичний центр, відеомагнітофон на ніч треба вимикати з електромережі;

– електронний будильник не повинен стояти в узголів’ї;

– потужність мікрохвильових печей може змінюватись, тому час від часу треба звертатися до майстра, щоб контролювати рівень випромінювання.

7.2 Аналіз сучасних програмних продуктів опрацювання великих масивів екологічної інформації

Оперативна, якісна і точна обробка великих масивів статистичної інформації може бути виконана лише з використанням сучасних засобів обчислювальної техніки. Наявність потужних, надійних і разом з тим простих в експлуатації програмних продуктів статистичного аналізу звільняє дослідника від рутинних операцій, розширює сферу застосування статистичних методів в різних галузях людської діяльності, сприяє появі якісно нових можливостей статистичного аналізу і моделювання даних. Використання пакетів прикладних програм – це єдиний реальний практичний інструмент розв'язування задач багатофакторного кореляційно-регресійного та аналізу в багатовимірному просторі.

Програмне забезпечення статистичних досліджень досить розвинуте. Сучасний ринок програмних продуктів пропонує різноманітні пакети програм для статистичної обробки даних. Всесвітньо відомі статистичні пакети для комплексної обробки даних: BMDP, SPSS, SAS, Systat, Minitab, S-Plus, Statgraphics Statistica та інші.

Використання згаданих пакетів програм дає змогу автоматизувати процес статистичного дослідження в таких напрямах: створення файлів даних і таблиць; групування даних; графічний аналіз даних; розрахунок варіаційних характеристик вибіркових сукупностей; побудова рядів розподілу; аналіз рядів динаміки і прогнозування їх майбутніх рівнів; кореляційно-регресійний аналіз; багатомірний аналіз.

Багатофункціональна, графічно орієнтована на обробку масових даних система Statistica відповідає основним стандартам Windows (динамічний обмін даними з іншими додатками, підтримка основних операцій з буфером обміну, робота в мережевому середовищі та інші).

Передусім це стандарти користувацького інтерфейсу – MDI, використання буфера-обміну, механізму динамічного зв'язку (DDE) з іншими додатками; система підтримує всі операції, реалізовані за допомогою методу Drag-and-Drop – «Перетягти та опустити», включаючи автозаповнення, інші.

Складніші процедури обробки даних у системі Stratgraphics виконує спеціалізований модуль Data Management – «Управління даними», а для обробки великих масивів даних або даних з довгими текстовими значеннями застосовують процедури Megafile Manager Data – «Менеджера мегафайлів».

Система Stratgraphics працює з чотирма типами документів. Це: електронна таблиця Spreadsheet, призначена для введення і перетворення первинних даних; електронна таблиця Scrollsheet – для виведення результатів аналізу; графік – для візуалізації результатів обробки та аналізу даних; звіт – файл у формі RTF (розширений текстовий формат), в якому зберігається текстова, числова і графічна інформація.

Усі статистичні процедури системи розбито на окремі модулі, кожен з яких об'єднує групу логічно зв'язаних між собою статистичних методів і в рамках конкретної моделі забезпечує повний і всебічний аналіз закономірностей.

У системі Statistica реалізовано принцип постійного логічного підказування. Якщо користувач не може визначитися щодо наступного кроку діалогу, через команду Enter система сама спрямує до відповідного діалогового вікна. Якщо виникають складнощі з вибором параметрів обчислювальної процедури, вони задаються системою «за умовчуванням».

Важливою характеристикою системи с наявність засобів всебічної графічної підтримки процесу обробки даних і візуалізації результатів аналізу. Графічні можливості й засоби системи унікальні. Вона включає сотні різних типів користувацьких і спеціальних статистичних графіків, доступних у будь якому модулі й на будь-якому етапі статистичної обробки даних. Інструменти

компонування складної графічної інформації з текстовою і числовою інформацією розглядаються у кожному модулі.

Використання сучасних комп'ютерних технологій обробки даних, інтерактивний спосіб взаємодії з системою перетворюють статистичний аналіз, моделювання та прогнозування в захоплююче дослідження закономірностей навколишнього світу. Завдяки різноманітним формам організації діалогу, максимально простій із звичними для статистики термінами мові спілкування, наявності контекстно залежної довідкової системи, мові програмування STATISTICA BASIC пакет є ефективним інструментом проведення статистичного дослідження як для користувача-початківця, так і для професіонала.

7.3 Проблема екологічності інформаційних і телекомунікаційних технологій

Екологія та технологія протягом півтора століть – від виникнення поняття екології в сучасному розумінні – залишались антагоністами. Вони не просто протистояли одне одному – існування одного виключало можливість існування другого. Але якщо перший етап розвитку обох явищ був боротьбою протилежностей, то після виходу і екології, і технології на якісно новий рівень відбулося єднання цих протилежностей.

Коли технології стали інформаційними (та телекомунікаційними), вони змогли сприйняти цілі й цінності екології. А екологія не просто отримала потужного союзника у перетворенні людського суспільства – вона набула реального, практичного змісту, перестала бути сухою філософською теорією, стала більш чи менш усвідомлюваним фоном повсякденного життя для більшості людства.

Інформаційні та телекомунікаційні технології, включивши в себе екологію в якості гуманних підвалин розвитку, перетворились на ідею

Інформаційного суспільства, стали способом життя людства, запорукою нового циклу розвитку цивілізації та планети.

Екологія ж віднайшла спосіб втілення і розв'язання тисячолітнього конфлікту "людське проти природного". В новому судженні зв'язок між двома елементами змінив характер: "або" перетворено Інформаційним суспільством на "та".

На сьогодні основними практичними проблемами є екологічність інформаційних і телекомунікаційних технологій та технологічність (перш за все "інформаційність") екологічних потреб. Якою мірою враховано інтереси людини, природи та планети у новітніх технологічних розробках? Якою мірою екологічні вимоги можуть бути практично втілені за допомогою цих технологій (але перш за все – доведені до відома мешканців Землі)?

Зупинимось на першому аспекті. Останні дані дозволяють нам зробити висновок якщо не про абсолютну екологічну ефективність інформаційно-телекомунікаційних технологій, то про їх чітку екологічну спрямованість. Так найпотужніші комп'ютери світу працюють на екологічні програми: на сьогоднішній день найбільш потужним у світі суперкомп'ютером визнано IBM ASCI White. Його встановлено в американській урядовій дослідницькій лабораторії Lawrence Livermore National Laboratory й використовувано для створення повноцінної тривимірної моделі термоядерної реакції.

Це означає, що більше не треба здійснювати вибухи або запускати ненадійні – бо експериментальні – реакторні установки. Зникає ще одна небезпека для людини і природи: не забруднюватиметься атмосфера і ґрунт при видобуванні радіоактивних елементів та захороненні відпрацьованих – а отже менше хворітимуть мешканці відповідних районів та працівники, задіяні в цьому.

Другим за потужністю визнано комп'ютер, встановлений у дослідницькому центрі National Energy Research Scientific Computing Center (NERSC). Суперкомп'ютер центру NERSC окрім того визнано першим за

потужністю серед систем, відкритих для загального користування. Ним користуються 2 тисячі різних дослідників, що займаються розробками у галузі створення екологічно чистих і більш економічних видів палива, вивченням глобальних змін клімату планети та іншими проблемами.

Використання інформаційних технологій для моніторингу екологічних систем та моделювання їх розвитку уможливило й інші серйозні міжнародні проекти. Так Організація об'єднаних націй має намір провести вивчення екологічного стану Землі. Згідно з офіційною заявою, зробленою представником Генерального секретаря ООН Фредом Екхартом, програма розрахована на чотири роки. До реалізації настільки масштабного проекту буде залучено більш як півтори тисячі вчених.

В результаті фахівці-екологи повинні дати оцінку теперішньому станові лісів, лук та полів Землі, а також прісних та солоних водойм. Передбачається, що "обнародувані до 2005 року експертами висновки й рекомендації буде використано урядами різних країн для ухвалення більш компетентних та обґрунтованих рішень", заявив Ф.Екхарт.

Загалом, концепція глобальних та локальних електронних інформаційних мереж є екологічною за своєю суттю. Це твердження вірне з точки зору як екології планети, так і екології людини, людської спільноти.

Інтернет оберігає планету від надмірного антропогенного втручання, бо він став продовженням людини. Тож порух миші тепер замінює безліч фізичних операцій, одним з неминучих наслідків яких є вплив на довкілля, і вплив, найчастіше, негативний.

Глобальна мережа перетворила планету і людство на щось дуже мале, обмежене у часі і просторі. Вони тепер вмістяться у долонях дитини, тож потребують захисту і турботи людини розумної.

Проблема, що виникла в Австралії, Антарктиді чи Африці, більше не сприймається як щось безмірно далеке, неспроможне бодай якось вплинути на буденне життя "тут і тепер", у цивілізованіших регіонах планети.

Глобальні мережі поширили ареал “тут і тепер” на цілу планету. Тож коли щось негаразд на дні океану, на гірській вершині, в тропосфері, на полярній шапці чи в заштатному містечку N-ську – мережеве суспільство дійсно тим переймається, дійсно займається вирішенням проблеми, дійсно непокоїться за своє майбутнє і майбутнє планети. Адже це майбутнє тепер так просто побачити – інформаційні технології і мережеве знання миттєво збудують модель того, що з нами буде, коли ми не оберігатимем себе і планету.

Інформаційні технології сьогодні є екологічнішими за більшість інших видів активної людської діяльності, проте їх ще не можна назвати справді екологічними. Скажімо, ефективність інформаційних мереж напряму залежить від кількості користувачів, тобто, від кількості комп’ютерів, включених до мережі. Але, як зазначає член Європарламенту від Зеленої партії Керолайн Лукас, для виготовлення одного звичайного персонального комп’ютера потрібно від 15 до 19 тонн матеріалів. Це порівнювано з 25 тонами, потрібними для виготовлення автомобіля.

На кожен функціонуючий комп’ютер (використовуваний в середньому протягом 4 років) припадає 1,5 комп’ютери вироблених. А близько третини комп’ютерів ніколи не буває продано взагалі – через швидкість, з якою вони втрачають технологічну актуальність. Це означає, що затрачувані ресурси справді наближаються до рівня автомобіля.

Є потреба нової концепції розвитку інформаційних технологій – основаної на екоефективності, включаючи спільне використання машин, повторне застосування та ремонт.

Але це не єдиний шлях підвищення екологічної ефективності інформаційно-телекомунікаційних технологій. Нещодавно крупний виробник мобільних телефонів – компанія “Nokia” – повідомила про наміри протягом кількох років розробити мобільні телефони з біорозкладуваними компонентами.

В компанії вже розпочато випробування біорозкладуваних корпусів для мобільних телефонів, але поки що серед полімерних матеріалів не вдалося знайти таких, що були б при цьому стійкими до дії гострих предметів (тобто, матеріалів, на яких не залишається подряпин). Дослідники "Nokia" не сумніваються, що з часом рішення буде знайдено, однак це потребуватиме не менше двох років роботи.

Проблема утилізації використаних мобільних телефонів (як, проте, і комп'ютерів, периферійного обладнання, пейджерів тощо) стає гострішою з кожним роком. Обсяги виробництва продуктів інформаційно-телекомунікаційних технологій та частота їх заміни на нові моделі примушують компанії замислюватись над проблемою біодеградації.

Переходячи до другого аспекту проблеми – реалізації цілей екології через інформаційно-телекомунікаційні технології, ми знов зауважимо, що передумовою перетворення цих технологій на дієвий інструмент екології є масове їх поширення. Вони мають змінити спосіб життя достатньої кількості людей, родин, підприємств для того, аби ці зміни відбилися на суспільстві в цілому.

Тож основною перевагою такого значущого на сьогодні фактору людської діяльності як інформаційні мережі є не стільки їх "інформаційність", скільки "електронність", себто доступність, простота, зручність та швидкість задоволення потреб користувача. За інакших обставин поява інформаційних мереж практично не позначилася б на способі життя людей, бо не здобула б їхньої прихильності.

Це означає, що тільки широке розповсюдження інформаційно-телекомунікаційних технологій забезпечить досягнення помітного екологічного ефекту. Екологія – явище, можливе лише в масштабах планети та людства.

ВИСНОВОК

У магістерській роботі виконано дослідження способів забезпечення автентифікації користувачів розподіленої комп'ютерної системи на основі протоколу Kerberos.

Основні наукові та практичні результати полягають в наступному.

1. Проведено аналіз наукових публікацій, протоколів та практичних рішень в області реалізації методів автентифікації користувачів розподіленохї комп'ютерної системи.

2. Проаналізовано можливості протоколу на основі віддаленого сервера автентифікації.

3. Здійснено аналіз ризиків при використанні методу автентифікації з віддаленим сервером.

4. Запропоновано практичну реалізацію методу для операційної системи Windows.

ПЕРЕЛІК ПОСИЛАНЬ

1. Denning, P. J. (Ed.). (1990). Computers Under Attack: Intruders, Worms and Viruses. Addison-Wesley Paper, 592 pp.

2. Stoll, C. (1989). The cuckoo's egg: tracking a spy through a maze of computer espionage. New York: Doubleday.

3. ISO 7498-2 (1988). Open Systems Interconnection - Basic Reference Model - Part 2: Security Architecture. International Standards Organisation.

4. ISO 10181-1 (1991). Open Systems Interconnection - Security Frameworks - Part 1: Overview. International Standards Organization.

5. Gilbert, I. E. (1989). Guide for Selecting Automated Risk Analysis Tools (NIST Special Publication 500-174).

6. Методичні вказівки по виконанню організаційно-економічної частини дипломних проектів науково-дослідницького характеру для студентів спеціальності 7.080401 “Інформаційні управляючі системи та технології” / Кирич Н.Б., Зяйлик М.Ф., Брощак І.І., Шевчук Я.М – Тернопіль, ТНТУ, – 2009. –11 с.

7. Основы охраны труда: учебник / А. С. Касьян, А. И. Касьян, С. П. Дмитрюк. – Дн-ськ : Журфонд, 2007. – 494 с.

8. Безпека життєдіяльності: Навч. посібник./ За ред. В.Г. Цапка. 4–те вид., перероб. і доп. – К.: Знання, 2006. – 397 с.

ДОДАТКИ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ПУЛЮЯ

М А Т Е Р І А Л И

VII НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

«ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ »

11–12 грудня 2019 року

ТЕРНОПІЛЬ
2019

УДК 004.056
Р. Леськів, Ю. Сметанка
Тернопільський національний технічний університет імені Івана Пулюя

РОЛЬ АУТЕНТИФІКАЦІЇ У РОЗПОДІЛЕНИХ КОМП'ЮТЕРНИХ СИСТЕМАХ

UDC 004.056
R. Leskiv, Yu. Smetanka
(Ternopil Ivan Pului National Technical University, Ukraine)

AUTENTIFICATION ROLE IN DISTRIBUTED COMPUTER SYSTEMS

Питання авторизованого доступу до обчислювальних ресурсів є актуальним як для локальних комп'ютерів, так і для мереж та хмарних сервісів. Дати відповідь на запитання, а чи є наш партнер в поточній сесії зв'язку в розподіленій системі саме тим, за кого він себе видає, ми можемо лише шляхом узгодження з ним спільної політики безпеки.

Щоб рішення на основі довіри працювало, ми повинні бути повністю впевнені, що надіслані дані можуть бути перевірені нашими партнерами як такі, що справді надходять від нас, а також ми повинні бути впевнені, що дані, які приходять до нас, дійсно були створені нашими партнерами. Це задача автентифікації. Зазвичай для аутентифікації використовується пароль та/або приватний ключ.

Паролі, як правило, корисні, якщо існує велика кількість сторін, яким потрібно автентифікувати себе конкретній іншій стороні. Публічні ключі, як правило, корисні, якщо є одна сторона, якій потрібно автентифікувати себе величезній кількості партій.

За допомогою пароля аутентифікація надає докази того, що хтось знає пароль. Якщо потрібно точно знати, хто це (що зазвичай важливо), тільки аутентифікація з залученням третьої сторони може надати таку інформацію. З відкритим ключем багато сторін можуть знати ключ, але тільки одна сторона, яка знає відповідний приватний ключ, може підтвердити автентифікацію самої себе. Тому ми схильні використовувати обидва механізми, але для різних випадків. Коли веб-сайт автентифікує у себе користувачеві, це робиться за допомогою криптографії. Поширюючи один відкритий ключ (для величезної кількості користувачів), веб-сайт може бути автентифікований усіма його користувачами.

Як практично ми використовуємо кожен з цих механізмів аутентифікації в розподіленій системі? Потрібно буде зашифрувати транспортування пароля через мережу. Шифрування пароля вимагатиме від нас мати або спільний симетричний ключ, або відкритий ключ нашого партнера.

Надання паролів з використанням третьої сторони реалізує сервер автентифікації Kerberos. На сьогодні Kerberos є одним з найстаріших протоколів автентифікації, що використовуються на сьогоднішній день [1]. До цього протоколу є багато розширень та доповнень як загального характеру, так і спеціальних. Цей протокол не базується на HTTP на відміну від багатьох інших протоколів автентифікації. Завдяки цьому дані, що передаються мережею, зовсім непридатні для читання людиною без застосування додаткових інструментів. Протокол з моменту створення зазнав немало доповнень та модифікацій, але з середини 80-х років минулого століття змін не зазнавав [2].

Література

1. The Kerberos Network Authentication Service (V5). [Електронний ресурс] . – Режим доступу: https://tools.ietf.org/html/rfc4120
2. Kerberos Protocol Tutorial. [Електронний ресурс] – Режим доступу: http://www.kerberos.org/software/tutorial.html