З перерахованих вище протоколів та стандартів саме протокол SNMP дозволяє розробити систему моніторингу доступності ресурсів мережі, що забезпечує мінімальне навантаження на мережу та вчасне інформування адміністраторів про втрату зв'язку з критичними вузлами.

Перелік посилань

1. Фейт С. TCP / IP. Архитектура. Протоколы. Реализация / Сидни Фейт. – Издательство Лори, 2016. – 424 с.

2. Эделман Д. Автоматизация программируемых сетей/ Джейсон Эделман, Мэтт Осуолт, Скотт С. Лоу. − Издательство : ДМК, 2019. – 616 с.

**Контроль цілісності інформації за допомогою хешування**

Акатов О.В. Капустян М.В., Огнєвий О.В.
Хмельницький національний університет

Більш надійними, ніж методи «парності», «контрольних сум», «циклічного контрольного коду» і «турбо-коду», можуть бути методи, побудовані на використанні односпрямованих криптографічних функцій хешування. Аналіз цілісності окремого об'єкта (тексту, файлу) може бути заснований на обчисленні хешу цього об'єкта за узгодженим алгоритмом і на наступному порівнянні його з початковим хешем об'єкта. Подібний аналіз використовують при синхронізації даних, при архівації, при резервуванні при здійсненні цифрового підпису, а також при інших процедурах.

Однак цей метод вразливий, тому що при навмисному порушенні цілісності інформації, особливо якщо порушення проводиться особою з санкціонованим доступом, може бути замінений і її контрольний хеш.

Окремим сформованим напрямком контролю цілісності даних є реєстрація часу надходження даних, що використовує засоби для виявлення порушення їх цілісності заднім числом – TSP (Time-Stamp Protocol). Цьому напрямку приділено увагу в багатьох роботах: від ранніх до однієї з останніх. Принцип реєстрації даних в цих роботах заснований на формуванні ланцюжка об'єднаних хешів (hash-chain based protocols for time-stamping and secure logging) за схемою, наведеною на рисунку 3, і на закріпленні реєстрації шляхом публікацій, що дозволяє виявляти порушення цілісності даних, вироблених заднім числом [1].

Метод полягає в тому, що реєструючу інформацію піддають хешуванню (отримують її хеш H), потім обчислюють об'єднуючий хеш H, враховуючий h і значення попереднього об'єднаного хеша. Кожен *N*-ий об'єднаний хеш публікують.

Засобам закріплення реєстрації засобів зберігання і обробки у зовнішньому інформаційному середовищі присвячений ряд робіт, в якій запропоновано проводити реєстрацію з використанням n-серверів, з яких в кожному акті реєстрації бере участь до обраних випадковим чином серверів, результати роботи яких аналізуються за заданим алгоритмом. Такий спосіб підвищує стійкість, наприклад, до DDoS-атак [2].

Для реєстрації даних оператору TSP – відповідно до схеми на рисунку 2 – направляють хеш цих даних і отримують у відповідь завірену електронними засобами посилку, що включає реєстрований хеш (хеш реєстрованих даних), час його реєстрації і криптографічу мітку, наприклад відповідний об'єднаний хеш (*H* на рисунку 1).

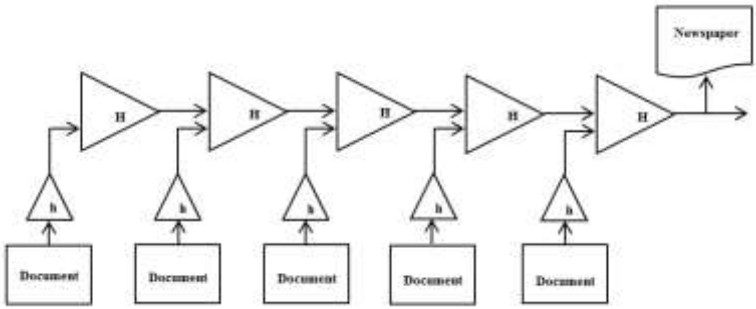


Рисунок 1- Лінійна схема об'єднаних хешів

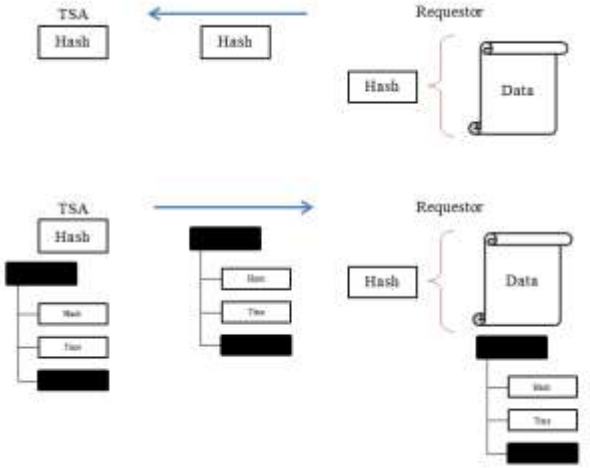


Рисунок 2 - Порядок реєстрації даних

Описаний вище спосіб TSP володіє значним потенціалом, частково використаним в даній роботі. Недоліком способу TSP є те, що контроль цілісності зареєстрованої в спеціальній базі даних інформації, може здійснювати тільки власник цієї БД по зберігаючій у нього системі хешів, а публікуючі окремі хеші (кожен *N*-ий, наприклад) не дають користувачеві можливостей для самостійного контролю цілісності інформації – ними безпосередньо може скористатися лише власник БД, якщо бажає. Якщо ж не бажає, то контроль цілісності виявиться недоступний для користувача – тобто контроль цілісності інформації залежить від волі обмеженого кола осіб, тобто необ'єктивний і вразливий [3].

Функції хешування повинні забезпечувати стиснення даних (отримання хеша), повинні просто обчислюватися і можуть бути безключовими (залежними тільки від повідомлення) або з секретним ключем (залежні від повідомлення і від секретного ключа). До безключових хеш-функцій відносяться коди виявлення змін (modification detection codes, MDC-коди), до яких пред'являються вимоги незворотності (обчислювальна неможливість відновлення даних по їх хешу, односпрямованість), стійкості до колізій першого роду (обчислювальна неможливість знаходження другого повідомлення з хешем, як у даного), стійкість до колізій другого роду (обчислювальна неможливість знаходження пари повідомлень з співпадаючими хешами). Такі хеш-функції називаються криптографічними. Вони оцінюються по відсутності кореляцій між вхідними та початковими бітами, по стійкості до близьких колізій (обчислювальна неможливість колізій, що відрізняються малою кількістю бітів), по стійкості до часткової односпрямованості (обчислювальна неможливість відновити частину початкового повідомлення), по можливості розтягування (можливість хешування коротких повідомлень) і ін. При цьому *n*-бітна хеш-функція вважається крипостійкою, якщо обчислювальна складність знаходження колізій для неї близька до 2, тобто до середнього числа атак «днів народження» для хеша довжиною в n розрядів. Атаки, які не залежать від алгоритму: атака "грубою силою", атака методом "дня народження", повний перебір ключів. При таких атаках вразливі всі алгоритми, єдина можливість протистояти їм - збільшити довжину хеш значення [4].

Список поширених алгоритмів хешування включає: Adler-32, CRC, SHA-1, SHA-2 (SHA-224, SHA-256, SHA-384, SHA-512), HAVAL, MD2, MD4, MD5, N-Hash, RIPEMD -160, Snefru, Tiger (TTH), Whirlpool, IP Internet Checksum (RFC 1071).

Схема алгоритму обчислення значення хеш-функції приведена на рисунку 3, де $H_i$ – *i*-те наближення хеш-функції у вигляді рядка довжиною 256 біт ($H_i$ - довільне); $m_i$ – *i*-ий блок довжиною 256 біт, на які розбитий хешуючий рядок (доповнюється при необхідності нулями); $f$ – крокова

функція хешування, яка відображає два блоки довжиною 256 біт в один блок довжиною 256 біт;

Після застосування функції $f$ до всіх блоків $m_i$ і відповідним проміжним значенням $H_i$ її застосовують до довжини вхідного повідомлення по модулю $2^{256}$ з $H_{n+1}$ і до контрольної суми $m_1 + m_2 + ... + m_n$ з $H_{n+2}$. В результаті отримують хеш вхідного повідомлення [5].

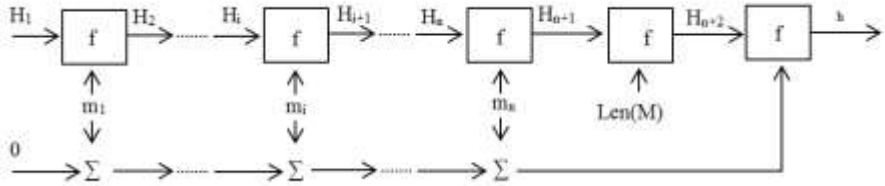


Рисунок 3 - Схема алгоритму хешування

Схема алгоритму хешування є варіантом ітераційного ланцюжка Меркле-Дамгарда, зображеного на рисунку 4.

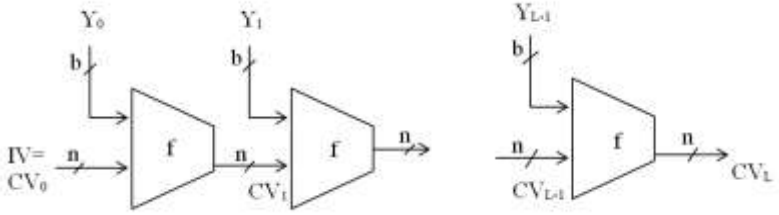


Рисунок 4 - Схема ітераційного ланцюжка Меркле-Дамгарда

Іншим варіантом розвитку цього алгоритму є алгоритм хешування BSA з двома ітераційними ланцюжками, на одного з яких подаються непарні за номером блоки $Y_{2n+1}$, на які розбитий хешуючий рядок, а на другий – парні $Y_{2n}$, а хеш отримують як конкатенацію результатів, отриманих за цими двома ланцюжках.

Перелік посилань

1.Петров, А. А. Компьютерная безопасность / А. А. Петров. — Криптографические методы защиты. - М.: ДМК. — 2000. — 448 с: ил.

2.Доля, А. Внутренняя ИТ-безопасность [Электронный ресурс] /А. Доля // Компьютер Пресс № 4'2005 НТМ. Режим доступа: http://www.compress.ru/article.aspx?id=10495&iid=430. - 2005.

3.Леваков, А. Анатомия информационной безопасности США / А. Леваков // Jet Info Информационный бюллетень, 6 (109). - 2002. - С. 29.

4.Суслопаров, А. В. Информационные преступления: диссертация / А.В. Суслопаров. - 2008.

5. Systems and methods for integrity certification and verification of contentconsumption environments: pat. US6931545. / Thanh Ta, Xin Wang. US. - 2000.

## Багаторівнева архітектура комп'ютерної мережі

Войцехівський Б.І.
Науковий керівник – к.т.н., доц. Кльоц Ю.П.
Хмельницький національний університет

Комп'ютерна мережа − це сукупність комп'ютерів, які можуть здійснювати інформаційну взаємодію один з одним за допомогою комунікаційного устаткування і програмного забезпечення.

Головною метою об'єднання комп'ютерів в мережу є надання користувачам можливості доступу до різних інформаційних ресурсів і їх спільного використання.

Основною метою застосування багаторівневої архітектури при побудові мережі є забезпечення високої надійності, масштабованості (можливості розширення або перебудови мережі з мінімальними витратами), високої продуктивності.

У загальному випадку в мережах виділяємо такі рівні:

– ядро мережі;

– рівень агрегації;

– рівень доступу.

Приклад трирівневої ієрархічної моделі мережі показано на рисунку 1.

Завдання ядра мережі − високошвидкісна комутація трафіку. Пристрої, що входять до складу ядра мережі, виконують функції:

– високошвидкісну маршрутизацію/комутацію трафіку мережі;

– резервування на рівні апаратури і каналів;

– розділення навантаження по паралельних каналах;

– швидкого перемикання між основним і резервним каналами;

– ефективного використання смуги пропускання з'єднань.

Ядро мережі будується з модулів, утворених одним високопродуктивним пристроєм, із забезпеченням апаратного резервування. Побудова ядра мережі на базі спеціально підібраних комутаторів скорочує час простою мережі, як в разі відмови апаратного (за рахунок гнучких схем резервування), так і в разі програмних помилок або помилок оператора (за рахунок різноманітних механізмів пошуку несправностей).