

Винахід стосується сфери, пов'язаної з контролем цілісності та автентичності даних і, зокрема, із завантаженням програмних засобів.

Винахід може бути застосовано до всіх пристроїв, які містять принаймні один центральний блок, що в даний час використовуються в інформаційній технології, скажімо, процесор, щонайменше частина програми якого міститься всередині пам'яті з можливістю перезапису.

Добре відомо, що заміна або ушкодження даних залишає свої сліди в певних частинах інформації, яка піддавалася обробці і зберігається в пам'яті, чи то перед обробкою, чи після неї. Відомо також, що з метою визначення, чи зазнавали змін розглядувані дані, використовується такий простий математичний метод, як перевірка "контрольної суми", за рахунок створення еталонної контрольної суми.

Однак, імовірно, що система контролю також зазнала змін, і в подальшому вона не може перевіряти вміст своєї пам'яті. Тому під час проведення математичних операцій може мати місце поширення компенсуючих випадкових помилок, що дає результат, ідентичний очікуваному. Отже, в певних випадках верифікація відомими методами може виявитися неефективною.

Таким чином, існує проблема, яка не має задовільного розв'язку і яка полягає в необхідності покращити рівень надійності і захисту, досягнутий з допомогою відомих способів верифікації, особливо тоді, коли один і той же блок використовується для розрахунків своєї контрольної суми та її порівняння з еталонним значенням.

Добре відомо, що аби зробити всі зміни даних видимими, до даних застосовується однонаправлена операція, тобто операція, яку легко виконати в одному напрямі, але майже неможливо виконати в іншому

напрямі. Наприклад, операція X^y виконується легко, і в той же час операцію $\sqrt[y]{X}$ виконати набагато складніше.

Термін "вільна від конфліктів операція" означає операцію, згідно з якою будь-яка інша комбінація даних, що надходять, дає подібний результат.

В рамках даного винаходу ця однонаправлена операція є математичним застосуванням H з групи джерела до об'єктної групи, в якій кожному елементу x з групи джерела приписано символ $H(x)$. Ці функції особливо корисні, коли вони є функціями, відомими як хеш-функції згідно з їх визначенням на стор.27 видання RSA Laboratories "Запитання про сучасну криптографію, які часто виникають, т.4.0". Елемент x може мати будь-яку довжину, але $H(x)$ завжди складається з ряду символів фіксованої довжини (рядок фіксованого розміру). Таку функцію важко інвертувати, тобто знання $H(x)$ зовсім не означає, що ми зможемо знайти x . Кажуть, що вона більш вільна від конфліктів, якщо вона ін'єктивна, тобто коли $H(y)=H(x)$ приводить до $y=x$ або $H(y) \neq H(x)$ приводить до $y \neq x$.

Мета даного винаходу полягає в тому, аби забезпечити, щоб інформація, яка міститься в декодері приймача платного телебачення була, з одного боку, тією, яку передав центр управління, а з іншого боку, не була змінена.

Ця мета досягається шляхом застосування способу перевірки цілісності і автентичності набору даних (від $M1$ до Mn), що зберігаються в пам'яті блоку декодування приймача платного телебачення, до складу якого входять власне блок декодування і блок захисту, а також засоби зв'язку (NET, REC) з центром управління.

Спосіб полягає в:

передаванні даних (від $M1$ до Mn) до блоку захисту;

розрахунку контрольної інформації (Hx), що є відображенням результату застосування функції, відомої як однонаправлена і вільна від конфліктів, до всіх або лише до частини даних (від $M1$ до Mn);

шифруванні контрольної інформації (Hx) з допомогою першого шифроключа ($k1$); установленні відповідності контрольної інформації (Hx) шляхом спілкування з центром управління з допомогою одного із засобів зв'язку.

Таким чином, цілісність даних надалі не перевіряється виключно блоком декодування, в якому ці дані зберігаються, а гарантується зовнішнім пристроєм, який вважається непроникним, - блоком захисту.

Згідно з цим винаходом декодер може сам виконувати розрахунки і передавати результати до блоку захисту або передавати до блоку захисту дані від $M1$ до Mn , який потім виконає розрахунки хеш-інформації.

Шифрувальні ключі, що використовуються для шифрування інформації з центру управління, містяться виключно в блоці захисту. Декодер не має засобів, аби дешифрувати ці повідомлення, а відтак видозмінити дані, передані центром управління, коли ті ж самі повідомлення проходять через декодер.

Ці блоки захисту, як правило, виконуються у формі смарт-картки і містять пам'ять, мікропроцесор та засоби зв'язку.

Під засобами зв'язку ми розуміємо або двонаправлений зв'язок по кабелю, через модем або зв'язок в діапазоні радіохвиль. Цим терміном охоплюються основні засоби переносу даних та засоби, з допомогою яких передаються повідомлення, спрямовані до модуля захисту.

Операція верифікації відповідності контрольної інформації (Hx) може бути виконана кількома шляхами.

Модуль захисту пересилає зашифровану контрольну інформацію до центру управління, і цей останній відповідає за проведення верифікації. У відповідь центр управління може послати або простий результат порівняння ОК/НОК, або еталонне значення. Всі ці повідомлення шифруються з допомогою шифроключа модуля захисту.

Центр управління заносить результат в пам'ять з посиланням на кожен абонентський пристрій, як доказ правильності проведення операції завантаження, або, навпаки, як доказ зміни даних, наприклад, з огляду на повтор.

Згідно з одним із варіантів здійснення винаходу центр управління може першим послати еталонне значення безпосередньо до блоків захисту. Таким чином, відпадає необхідність робити запит до центру управління щодо верифікації відповідності розрахованої контрольної інформації, Hx .

Згідно з іншим способом проведення операції і у випадку, коли від блоку захисту надходить запит на верифікацію, центр управління пересилає до блоку захисту, як результат порівняння, еталонне значення (Hu) в зашифрованому вигляді $k2(Hu)$. Як тільки це відбувається, центр управління не лише інформує блок захисту про те, правильне воно, чи ні, але посилає це еталонне значення до блоку захисту. Це робитиметься

головним чином тоді, коли порівняння дало позитивний результат, аби блок захисту міг запам'ятати еталонне значення H_u .

Пересилання цієї інформації може здійснюватися допоміжними засобами зв'язку, наприклад, по модему або з допомогою основного шляху проходження даних.

У випадку, коли дані від M_1 до M_n вже супроводжуються засобами верифікації, такими, як циклічний надлишковий код, контрольна сума або хеш-функція, блок декодування може виконати початкове тестування з допомогою засобів, які в ньому знаходяться. Однак, надійність цього тестування повинна викликати сумнів, а саме, якщо дані будуть змінені третьою особою, напевно ця третя особа в той же спосіб змінить і засоби верифікації. Ось чому в способі, запропонованому даним винаходом, блок захисту може інформувати блок декодування, аби той не приймав результат тестування як гарантію автентичності даних, а ця автентичність визначається відповідно до способу, описаного далі.

Цей варіант важливий у випадку оновлення низки декодерів, частина з яких має операційну систему старої версії і потребує верифікації по контрольній сумі, або належить до тих інших, які вже обладнані для системи згідно із заявленим тут способом.

Коли завантажуються оновлені програмні засоби, цілком звичним є послати лише ту частину, що була змінена. Дані від M_1 до M_n не представляють усієї заново оновленої програми. Причина в тому, що з метою підтримки надійності засобів верифікації усієї програми, важливо мати у розпорядженні еталонне значення H_u , яке є відображенням хеш-функції у заново створеній програмі.

Це є першим способом, який полягає у встановленні початкової цілісності програми P_0 , тобто перед її оновленням. Щоб це здійснити, початкові результати H_0 хеш-функції в програмі P_0 або ініціалізуються при інсталяції програми, або визначаються відповідно до способу згідно з цим винаходом.

Коли автентичність даних оновленої редакції установлена і внесена в програму пам'яті, блок захисту може негайно дати команду, щоб хеш-функція була застосована до всієї нової програми P_1 , що дасть результат H_1 . Цей результат буде необхідним для наступних перевірок або подальших оновлень.

Один із варіантів цього способу полягає в одержанні від центру управління нового значення H_u , яке є відображенням дії хеш-функції на всю нову програму P_1 і яке тут показано послідовністю від M_0 до M_m .

Керуючі дані R , послані центром управління, можуть містити сервісний описувач D даних, який вказує блоку декодування (IRD), як використовувати ці дані. Описувач може бути виконаним у формі таблиці, в якій містяться всі адреси та одержувачі даних. Таким чином, ці дані не можна буде використати без описувача, причому останній буде повернутим назад до блоку декодування (IRD) лише тоді, коли порівняння дасть позитивний результат.

Згідно з варіантом здійснення винаходу центр управління включає до керуючих даних R підтвердження, з метою засвідчити передавача даних.

Ця функція верифікації пов'язана не лише із завантаженням нових даних в декодер, але дозволяє також проводити тестування достовірності і автентичності даних у будь-який момент. У цьому випадку операція полягає в розрахунку, періодично або за запитом, значень відображення H_x результату застосування так званої однонаправленої і вільної від конфліктів функції до всіх або лише до частини даних (від M_0 до M_m) в операційній пам'яті декодера і в передачі цієї інформації (H_x) до блоку захисту для порівняння з еталонним значенням (H_u).

Для виконання цієї операції існує перший спосіб, який полягає в тому, що розрахунки проводяться декодером, а їх результат передається до блоку захисту. Згідно з варіантом цього способу розрахунки проводяться блоком захисту, а від декодера до блоку захисту (SC) передаються дані (від M_0 до M_m).

Запит на виконання цих операцій верифікації може надійти від центру управління, від блоку захисту, від блоку тестування або від одного із засобів зв'язку, лише якщо вони перебувають під напругою.

Оскільки блок захисту порівнює розраховані значення H_x з еталонною величиною H_u , остання може бути представлена або значенням, що було розраховано декодером IRD, після підтвердження його достовірності центром управління, або еталонною величиною, наданою центром управління.

Один із шляхів, які зазвичай використовуються нечесними людьми в намаганні зрозуміти, як працює система платного телебачення, полягає у спостереженні за реакцією, що йде слідом за спробою видозмінити систему. Саме тому цей винахід однаковою мірою відкритий для методу передавання результату порівняння, виконаного в інший спосіб, наприклад, коли абонент вирішує прийняти інформацію про подію, і створене абонентом повідомлення відсилається до центру управління.

У цьому повідомленні корисно помістити інформацію про те, що дані від M_1 до M_n були змінені, інакше буде дуже важко пов'язати цю видозміну даних з блокуванням декодера, що може статися набагато пізніше.

Згідно з варіантом здійснення винаходу значення результату розрахунків H_x передається до центру управління. Для того, щоб це зробити і залишити це прихованим, згаданий результат ділиться на частини і розміщується, частина за частиною, всередині адміністративних повідомлень, що їх використовує система.

Центр управління знову формує величину H_x , частина за частиною, і коли її величина стає повною, визначає, чи видозмінювалися ці значення.

Одна з проблем, на яку наштовхуються під час оновлення великого числа декодерів, полягає у кількості запитів до центру управління, щоб одержати верифікацію.

Одним із запропонованих рішень в рамках цього винаходу є підрозділити у псевдовипадковий спосіб ці запити до центру управління щодо проведення верифікації.

Інше рішення, яке описано раніше, полягає у посиланні попередньої еталонної величини. Таким чином, якщо дані одержані правильно, що трапляється в більшості випадків, оновлення може набрати чинності без очікування запиту до центру управління. В будь-якому разі цей запит буде здійснено, аби підтвердити, що оновлення проведено правильно.

При особливому способі роботи розглядувана група містить вузол передавача, що знаходиться всередині центру управління, і приймач, який може бути виконано у вигляді достатньо великого числа периферійних блоків, котрі працюють схожим чином. Метою способу є гарантувати, що програмні засоби, послані вузлом

передавача, одержуються в автентичному вигляді і в повному обсязі кожним із периферійних блоків. У відповідності з термінологією, що використовується в платному телебаченні, яке є важливою, але не виключною, сферою застосування даного винаходу, в подальшій частині цього документу периферійні блоки будемо називати об'єднаним приймачем-декодером (IRD), до складу якого входять приймач, декодер для обробки прийнятого ним сигналу і центральний процесор, або CPU, який працює переважно з енергонезалежним запам'ятовуючим пристроєм, як це буває в різноманітних периферійних пристроях.

Енергонезалежний запам'ятовуючий пристрій є пам'яттю, вміст якої, навіть якщо виключене основне джерело живлення, підтримується неушкодженим, наприклад, з допомогою принаймні одного такого незалежного джерела енергії, як електричні батареї. Можуть бути використані й інші види енергонезалежних запам'ятовуючих пристроїв, наприклад, програмована постійна пам'ять з електричним стиранням (EEPROM) та флеш програмований постійний запам'ятовуючий пристрій (FEPROM). Саме ці енергонезалежні запам'ятовуючі пристрої зберігають дані неушкодженими у випадку переривання постачання струму, а це важливо для нормальної організації роботи процесора блоку IRD.

Інформація, що надходить з центру управління, одержується IRD у вигляді потоку даних, який приходить на приймач блоку IRD. У випадку закодованого телебачення, або більш загально інтерактивного телебачення, цей потік даних містить відеоінформацію, аудіоінформацію, інформацію у вигляді даних, виконавчі прикладні програми і, нарешті, різні види даних контрольної інформації.

У цьому випадку проблемою є гарантування того, щоб інформація була прийнята без помилок і була інтерпретована IRD перед тим, як бути записаною в оперативну пам'ять, особливо це стосується виконавчих даних, тобто програмних засобів.

Приймач блоку IRD передає їх до декодера, який потім запускає їх у кругообіг по IRD за допомогою шини. До шини приєднано спеціальний мультимедійний процесор, який у свою чергу зв'язаний з монітором та одним або більше гучномовцями, згаданий вище енергонезалежний запам'ятовуючий пристрій і один або більше необов'язкових підпорядкованих пристроїв. Саме процесор (CPU) організовує правильну роботу IRD і управляє ним, а також різними підпорядкованими пристроями, як, наприклад, інтерфейсом, допоміжним запам'ятовуючим пристроєм, іншими процесорами або модемом. Більш того, центр управління може приймати обмінну інформацію, наприклад, через модем, приєднаний до загальнодоступної мережі передачі даних.

Ці підпорядковані пристрої самі по собі можуть бути джерелом помилок, які потім необхідно виявляти і виправляти, особливо у випадку завантаження нової версії діючого програмного забезпечення IRD, і особливо його CPU, або певних виконавчих програм для IRD або його компонентів.

Програмні засоби і дані, для яких має бути гарантована автентичність і цілісність, можуть бути завантажені з допомогою різних засобів. Один із цих засобів, як уже говорилося, полягає в надсиланні згаданому вище приймачеві оновленої редакції пам'яті разом з потоком даних, який містить низку блоків даних M1, M2, ...Mn, подібних до заголовків, для того, щоб центральний блок міг легко розпізнати ці дані від M1 до Mn.

Альтернативно, або як доповнення, блоки даних можуть досягти IRD через один з його необов'язкових підпорядкованих пристроїв, наприклад, через модем.

В рамках цього винаходу блоки даних M1, M2, ...Mn без будь-яких перешкод можуть бути посланими в незашифрованому вигляді, тобто ще не будучи зашифрованими.

У даному вигляді спосіб згідно з цим винаходом полягає у першочерговому застосуванні, протягом етапу передавання, однонаправленої або хеш-функції до частини або до всіх блоків даних M1, M2, ...Mn, аби в результаті отримати відображення Hx групи від M1 до Mn. Блоки даних M1, M2, ...Mn можуть оброблятися окремо цілком схожим чином і в результаті давати H1x, яке відповідає M1, Hx2, яке відповідає M2 і Hxn, яке відповідає Mn. Цей результат або ці Hx заносяться в пам'ять центром управління для подальшої верифікації.

При перевірці автентичності даних особливу критичність викликають системи, з допомогою яких ці дані передаються по загальнодоступних каналах зв'язку, таких, як канали радіохвильового зв'язку, телефонні канали або інтернет. У даному випадку зловмисник може зайняти місце центру управління і послати дані для внесення змін у роботу системи, вибраної мішенню.

Добре відоме приєднання криптограми під час передавання даних для перевірки їх автентичності. Однак, ця криптограма відповідає лише потребам ідентифікації автора даних, але вона не впливає на декодер, який втратив критерії еталону.

Ефективність способу частково залежить від якості однонаправленої функції H і від затвердження цих сигнатур блоком захисту, який вважається непроникним. Таким чином, проста контрольна сума не дозволяє виявити перестановку двох блоків символів в даних тому, що додавання в математиці вважається комутативною і асоціативною дією. З іншого боку, результат застосування хеш-функції, Hx, є дуже реалістичним зображенням x, навіть якщо воно набагато довше, ніж Hx. Якщо перестановка символів проводиться в групі символів x, функція H(x) знайде її негайно, і після цього система не зможе далі функціонувати. Результатом буде відмова, викликана захистом.

Важливим аспектом винаходу є те, що він дозволяє в будь-який час проводити верифікацію достовірності даних в пам'яті периферійних блоків. Насправді, наявність цієї контрольної інформації в модулі захисту дозволяє декодеру виконувати автоверифікацію. Ця верифікація дає результат без його порівняння з контрольною сумою, яка зазвичай використовується в програмній пам'яті. Якщо ця верифікація дає результат, схожий на еталон, блок має різні засоби (модемний зв'язок, кабельний зв'язок) для інформування зовнішнього блоку, наприклад, центру управління, про невідповідність програми.

Якщо в цьому винаході перевага щодо створення і передавання контрольної інформації віддається центру управління, то винаходом передбачається також периферійний блок, в який попередньо завантажуються вся або частина програми разом з контрольною інформацією так, як це описано вище. Це може бути виконано під час виготовлення в момент ініціалізації перед здійсненням торгівельної операції з використанням процесора, або шляхом завантаження цієї контрольної інформації через один з периферійних пристроїв в момент стадії ініціалізації.

Винахід ілюструється блок-схемою об'єднаного приймача-декодера.

IRD або об'єднаний приймач-декодер, показаний на цій блок-схемі, є периферійною частиною системи, до якої застосовується спосіб згідно з винаходом в описаному нижче порядку. Цей IRD містить центральну шину DB, до якої приєднані всі інші модулі. Центральний модуль блоку IRD виконано у вигляді центрального процесора CPU, завданням якого є виконання різних процесів.

Приймач REC приймає потік даних, що містить відео- та аудіоінформацію, інформацію у вигляді даних та виконавчі прикладні програми, через різні канали обслуговування, такі, як кабель, диполь Герца, супутникова параболічна антена, Інтернет або інша відома техніка. Цей приймач REC зв'язаний з інтерфейсом DC каналу передачі даних, який також приєднаний до шини (DB).

До шини (DB) приєднані також наступні блоки:

мультимедійний процесор MP, призначений для обробки відео- та аудіоінформації, який пересилає її відповідно до монітору VD та гучномовців AD;

випробувальний канал TC, який може бути приєднано до тестера TEST для заводського налаштування та обслуговування;

енергонезалежний запам'ятовуючий пристрій NVM, який не залежить від основного джерела енергії і має власне джерело живлення;

інтерфейс INT для смарт-картки, який фізично сприймає смарт-картку;

допоміжний запам'ятовуючий пристрій або блок пам'яті TMEM;

модем MD, приєднаний до мережі загального користування NET, в якому використані широко відомі технічні та обслуговуючі засоби;

інші процесори OP, DP з різними функціями відповідно до потреб користувача, зокрема тих, що пов'язані з обробкою даних.

Саме CPU управляє оновленням програмних засобів, приклад чого буде описано. Він приймає їх або відкидає залежно від результатів тестування, проведеного з використанням способу, що є предметом цього винаходу.

Ці версії програмного забезпечення для CPU блоку IRD можуть надходити до IRD через приймач REC, через тестер TEST, через смарт-картку SC або через мережу NET. Далі буде описано приклад того, як потік відео- та аудіоінформації надходить до IRD через приймач REC.

Набір даних, які являють собою нову версію програмних засобів, що надходить до IRD, записується в тимчасову пам'ять TMEM блоку IRD разом із сервісною інформацією після їх перевірки на автентичність і цілісність. Це дозволяє центру управління завантажувати цю версію програмних засобів у велику кількість периферійних пристроїв та проводити безпомилкову інсталяцію через IRD.

Як тільки повідомлення було прийняте блоком IRD, дані розбиваються на частини, і ці різні елементи записуються в тимчасову пам'ять TMEM. IRD обробляє блоки від M1 до Mn в тому ж стані, в якому вони були передані, але у зворотному порядку. Зрозуміло, що у випадку, коли ці блоки прийняті в зашифрованій формі, першою операцією є дешифрування даних з допомогою відкритого ключа PuK, щоб мати дані в незашифрованому вигляді.

Наступний крок полягає в застосуванні однонаправленої функції H до блоків даних від M1 до Mn, щоб у результаті мати значення від Hu1 до Hup. У випадку, коли в блоки пам'яті M1, M2, ...Mn, протягом передачі повідомлення, вкралася помилка, ця помилка проявить себе на Hu, яке виявиться відмінним від Hx, котре міститься в блоці управління, і дані від M1 до Mn будуть відкинута.

Ці результати передаються до смарт-картки SC, яка відповідає за перевірку їх автентичності. Як описано вище, ця операція проводиться шляхом виходу на зв'язок з центром управління, чи то негайно, чи пізніше.

Прикладом функцій H є функції MD2, MD5 та SHA-1.

Згідно з іншим варіантом здійснення винаходу блок, що містить дані, не має каналу зв'язку з центром управління. Дані надходять до запам'ятовуючого пристрою разом з керуючою інформацією (R1), до якої входить результат застосування однонаправленої або вільної від конфліктів функції, що зветься хеш-функцією, до всіх або до частини даних (від M1 до Mn). Особливість цієї керуючої інформації (R1) полягає в тому, що, з одного боку, вона містить хеш-функцію для розглядуваного набору даних, а з іншого боку, ці дані записані у зашифрованому вигляді k2(Hu). Запам'ятовуючий пристрій не може їх ні розпізнати, ні видозмінити.

Протягом фази верифікації запам'ятовуючий пристрій передає контрольну інформацію в зашифрованому вигляді до блоку захисту. Блок захисту містить засоби для дешифрування цієї інформації, особливо для здобування результату від застосування хеш-функції (Hu).

Крім того, згідно з першим варіантом здійснення винаходу запам'ятовуючий пристрій застосовує хеш-функцію до даних від M1 до Mn, розраховує контрольну інформацію Hx і передає її до блоку захисту для порівняння. У відповідь блок захисту посилає до запам'ятовуючого пристрою зворотні дані (R2), включно з результатом порівняння.

Далі у випадку, якщо дані не автентичні, запам'ятовуючий пристрій забезпечує прийняття необхідних заходів.

Згідно з другим варіантом здійснення винаходу розрахунок контрольної інформації Hx проводиться блоком захисту, який у цьому випадку одержує від запам'ятовуючого пристрою дані від M1 до Mn.

Згідно з варіантом здійснення винаходу, який дає вищий рівень гарантії стосовно користування даними, до керуючих даних (R1) приєднують шифрувальний ключ k3 для шифрування даних від M1 до Mn.

Ці дані попередньо записуються у зашифрованому вигляді, і хеш-функція створюється в цих зашифрованих даних. Коли проводиться верифікація цілісності даних для блоку захисту і результат позитивний, блок захисту вносить в дані (R2) відповіді, яка надсилається до запам'ятовуючого пристрою, шифрувальний ключ k3, що дозволяє дешифрувати дані від M1 до Mn.

Згідно з варіантом способу, описаним вище, блок захисту не надсилає шифрувального ключа k3, але саме запам'ятовуючий пристрій відсилає зашифровані дані від M1 до Mn до блоку захисту SC для дешифрування.

Так само, як і в попередньому способі, цей контроль можна проводити в будь-який час протягом роботи

запам'ятовуючого пристрою.

Керуючі дані (R1) містять описувач даних D, який вказує запам'ятовуючому пристрою, як використовувати ці дані. Цей описувач може бути виконаний у формі таблиці, в якій містяться адреси та одержувачі даних. Таким чином, ці дані не можна буде використати без описувача, причому останній буде повернутим до запам'ятовуючого пристрою лише тоді, коли порівняння дасть позитивний результат.

Можна також передбачити, аби до керуючих даних (R1) було додано підтвердження, яке засвідчує передавача даних, з метою зберегти його слід у блоці захисту.

