НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
НАВЧАЛЬНО-НАУКОВИЙ ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

КОМП'ЮТЕРНИЙ ПРАКТИКУМ № 3

«ДОСЛІДЖЕННЯ КРИПТОГРАФІЧНИХ ПРОТОКОЛІВ СИСТЕМ WEBMONEY, PAYPAL»

Виконали:

Студенти групи ФІ-22 мн

Ковальчук Ольга

Коломієць Андрій

Толмачов Євгеній

Київ – 2024

Мета та основні завдання роботи

Дослідження особливостей реалізації криптографічних механізмів платіжних систем.

Вимоги

Дослідити особливості реалізації криптографічних протоколів, а також особливості роботи з електронними гаманцями систем WebMoney та PayPal.

Звіт має містити детальний опис проведеного дослідження особливостей реалізації криптографічних механізмів протоколів систем WebMoney та PayPal.

Також звіт має містити загальні теоретичні відомості побудови платіжних систем та їх основні характеристики (специфікація SET), зокрема систем мікроплатежів, таких як Payword та Micromint, та протоколи електронних грошей. Для кожної наведеної системи або протоколу необхідно обгрунтувати його захищеність та вибір криптографічних примітивів.

Основні теоретичні відомості

1. Необхідні теоретичні відомості містяться в роботах: М.А. Деднев, Д.В. Дыльнов, М.А. Иванов Защита информации в банковском деле и электронном бизнесе. – М.: КУДИЦ-ОБРАЗ, 2004. – 512 с.

2. PayPal security guidelines and best practices [WWW Document], n.d. URL https://web.archive.org/web/20210619025314/https://developer.paypal.com/docs/api/info-security-guidelines/ (accessed 12.2.23).

Виконання комп’ютерного практикуму

WebMoney

Облікова система WebMoney Trasfer забезпечує проведення розрахунків у реальному часі за допомогою облікових одиниць - титульних знаків

WebMoney(WM). Управління рухом титульних знаків здійснюється користувачами за допомогою клієнтської програми WM Keeper.

Системою підтримується кілька типів титульних знаків, забезпечених різними активами та зберігаються на відповідних електронних гаманцях:

- WMR - еквівалент RUR - на R-гаманцях,
- WME - еквівалент EUR - на E-гаманцях,
- WMZ - еквівалент USD - на Z-кошельках,
- WMU- еквівалент UAH- на Z-кошельках,
- WM-C і WM-D - еквівалент WMZ для кредитних операцій - на C - і D - кошельках.

При переказі коштів використовуються однотипні гаманці, а обмін різних титульних знаків виробляється в обмінних сервісах (http://www.exchanger.ru).

Гарантом за WMR-операціями є АНО "ВМ-Центр" - автономна некомерційна організація, що представляє WebMoney Transfer на території росії. Гарант за WMZ-і WME-операціями виступає компанія Amstar Holdings Limited, S. A. Гарантом за WMU-операціями виступає компанія ТОВ “Українське гарантійне агентство".

Для того щоб стати учасником системи WebMoney Transfer, досить встановити на своєму комп'ютері клієнтську програму WM Keeper і зареєструватися в системі, отримавши при цьому WM-ідентифікатор і прийнявши угоди системи. Процес реєстрації також передбачає введення персональних даних і підтвердження їх достовірності засобом клієнтської програми WM Keeper.

Хоча система і володіє всіма споживчими властивостями електронних готівки, на диску комп'ютера користувача зберігаються не самі електронні гроші, а всього лише закодовані записи про наявність грошей у клієнта. Гроші, на відміну від стовідсоткових цифрових готівки, при втраті файлу-гаманця не пропадають.

WebMoney дає своїм клієнтам можливість вибрати, який статус WM буде для них найбільш прийнятний, і використовує форми взаємин, які не потребують спеціального дозволи або ліцензування. Однією із зручних форм роботи з WM є

їх статус як "зобов'язання, що дає права доступу в систему WMT"» Це зобов'язання володіє фіксованою цінністю і враховується бухгалтерією “за балансом". Однак за бажанням клієнта з ним може бути укладений договір, що надає WM будь-який інший законний статус, наприклад договір позики.

Гроші в систему з кредитних карт вводити не можна: шахрай, який отримав електронні гроші по фальшивій карті, встигне їх витратити, перш ніж шахрайство буде виначено , і-зворотний бік анонімності та конфіденційності розрахунків обчислити його, швидше за все, не вдасться. Можна перевести гроші з системи на свою банківську карту. Ця операція зводиться до перекладу вмісту гаманця на банківський рахунок. З WebMoney гроші можна перевести в будь-який банк світу.

У WebMoney мінімальна сума платежу становить 0,01 WM (тобто одну копійку або один цент), мінімальна величина комісійних також 0,01 WM. Це означає, що платіж в одну копійку обійдеться вам в 50% комісійних, платіж в 10 коп. - в 10%, а платіж в 50 коп. - у 2%.

Ідентифікація. При реєстрації уастнику WebMoney Transfer присвоюється 12-ний WM-ідентифікатор, необхідний для роботи в системі. WM-ідентифікатор служить унікальним позначенням учасника.

Власник WM-ідентифікатора самостійно призначає пароль і визначає місце для зберігання файлів з секретним ключем і гаманцями. Він може вносити в інформаційні поля програми дані про себе і при необхідності змінювати їх. Крім ідентифікації користувачів за допомогою WM-ідентифікатора в системі здійснюється аутентифікація на підставі секретного ключа і пароля до нього. Секретний ключ не повинен передаватися третім особам, оскільки є власністю власник WM-ідентифікатора.

Для посвідчення особи власника WM-ідентифікатора в системі діє WM-аттестація.

Користувачам системи надається можливість використання автоматизованих засобів для ідентифікації та аутентифікації учасників при побудові власнихних додатків.

Безпека. Технологія WebMoney Transfer розроблена з урахуванням сучасних вимог безпеки, що пред'являються до систем управління інформацією через

Інтернет. Для забезпечення високого рівня захисту даних і засобів користувачів в системі підтримується комплекс заходів, ефективність якого підтверджена практикою багаторічної безперебійної роботи системи. Розглянемо деякі заходи безпеки, застосовувані в системі.

Для входу в програму WM Keeper необхідно знання унікального 12-значного WM-ідентифікатора користувача, його особистого пароля (призначається користувачем), а також місця розміщення в пам'яті комп'ютера файлів з секретним ключем і гаманцями.

Архітектура системи виключає несанкціонований доступ до WM-гаманців користувачів і не дозволяє проводити розрахунки з використанням WM-гаманців, на яких нема коштів (цього захисту позбавлені, зокрема, системи оплати за допомогою кредитних карт). Всі операції в системі-зберігання WebMoney на гаманцях, виписка рахунків, розрахунки між учасниками, обмін повідомленнями-відбуваються в зашифрованому вигляді, з використанням алгоритму, подібного RSA, з довжиною ключа більше 1040 біт. Для кожного сеансу використовуються унікальні сеансові ключі, що забезпечує гарантовану конфіденціальність здійснення угод та обміну інформацією.

На системному рівні забезпечується стійкість по відношенню до обривів зв'язку. Якщо будь-яка операція не була успішно завершена, вона не враховується системою. При здійсненні транзакції кошти завжди знаходяться або на WM-гаманці відправника, або на WM-гаманці одержувача. Проміжного стану в системі не існує. Таким образом, принципово не може виникнути ситуації, коли WM-кошти будуть втрачені.

Конфіденційність. При бажанні за допомогою налаштувань програми WM Keeper можна закрити персональні відомості (ім'я, прізвище, E-mail, поштову адресу і т. п.) Для огляду іншими учасниками WebMoney Transfer. В цьому випадку при здійсненні угод друга сторона не зможе отримати зазначених відомостей.

Якщо в подальшому ваш торговий партнер зажадає вказівки деяких з вище перерахованих особистих відомостей і ви погодитеся з цією вимогою, то настройки програми WM Keeper дозволять зробити цю інформацію доступною.

За вашим WM-ідентифікатором неможливо визначити номери використовуваних вами WM-гаманців. При бажанні ви можете інсталювати на

своєму комп'ютері будь яке число версій WM Keeper і входити в систему під різними WM-ідентифікаторами.

У таблиці нижче дається характеристики розглянутих платіжних систем.

Параметр	WebMoney
Тип системи	Гнучка позабанкіе система
Сфера застосування	B2B, B2C, C2C
Транзакції між користувач	Реалізований

Параметр	WebMoney
Вывод денег из системы	Наличными (в 4 странах), банковским, почтовым, телеграфным переводом, через Сбербанк, Western Union
Криптографичес-кие методы защиты	SSL+ симметричный криптоалгоритм собственной разработки (1024 бит) + RSA (1048 бит)

Параметр	WebMoney
Правовий простір	Міжнародна система
Надійність	Гроші втратити неможливо
Мультивалютність	Є (в рамках системи - дві валюти). На вході-виході різні світові валюта
Мікроплатежі	Від 1 коп. (WMR) або від 1 цент (WMZ)
Комісійні, стягуються система	0,8% при транзакціях і виведення грошей з системи. Введення грошей в систему - безкоштовно. Мінімальний рівень - 0,01 WM
Прийом платежів за кредитками	Можливість заявлена
Стикування з бухгалтерія	+
Анонімність платіж	Можливий
Посвідчення учасників угоди	Тільки за бажаниям клієнтів
Кредитування	Підтримувати
Двофазний платежі з захистом угоди	Підтримуються на рівні система
Введення грошей в система	Готівкою (у 4 країнах), банківським, поштовим, телеграфним перекладом, через Сбербанк, Western Union, за передплаченими карта

PayPal

Протоколи SSL /TLS є основою безпечного зв’язку в Інтернеті. Вони також піддаються постійним атакам. Експерти з безпеки намагаються бути на крок попереду кіберзловмисників, вивчаючи протоколи SSL/TLS на наявність вразливостей. Результатами таких досліджень стали уразливості POODLE і Heartbleed . Щоб захистити вашу інтеграцію від поточних і майбутніх загроз безпеці, ми рекомендуємо вам дотримуватися найкращих практик, наведених нижче:

1. Припиніть використання кореневого сертифіката VeriSign G2
2. Оновлення до сертифікатів SSL SHA-256
3. Використовуйте TLS версії 1.2 або новішої
4. Нехай протокол погоджує найвищу версію
5. Не використовуйте жорсткий код певних шифрів
6. Дозволити ідеальну пряму секретність
7. Будьте пильні

Припиніть використання кореневого сертифіката VeriSign G2

Громадські центри сертифікації активно відмовляються від 1024-бітних кореневих сертифікатів на користь більш безпечних 2048-бітних кореневих сертифікатів. У результаті вам потрібно припинити використання з’єднань SSL, які покладаються на старіші 1024-розрядні сертифікати, такі як кореневий сертифікат VeriSign G2.

Оновлення до сертифікатів SSL SHA-256

SHA-1 — це 22-річний криптографічний алгоритм, якому загрожує збільшення обчислювальної потужності. Вам потрібно перейти від використання сертифікатів SSL, які використовують SHA-1, до більш надійного алгоритму підпису SHA-256.

Використовуйте TLS версії 1.2 або новішої

PayPal оновив свої служби, вимагаючи TLS 1.2 або новішої версії для всіх з'єднань HTTPS. TLS версій 1.0 і 1.1, а також SSL версій 1.0, 2.0 і 3.0 — це старіші протоколи з відомими вразливими місцями, які вже не підтримуються.

Крім того, PayPal також вимагає HTTP/1.1 для всіх з'єднань.

Нехай протокол погоджує найвищу версію

Оскільки протоколи Інтернету часто змінюються у відповідь на загрози, ми не рекомендуємо жорстко кодувати свою інтеграцію до певної версії. Натомість ми рекомендуємо вам дозволити протоколу автоматично узгоджувати найвищу версію.

Не використовуйте жорсткий код певних шифрів

Нижче наведено кілька причин, чому вам не слід жорстко кодувати певні шифри в інтеграціях:

- Такі шифри, як RC4 і DES, широко використовуються для TLS, але виявилися ненадійними та вразливими до атак.
- Досконаліші шифри, такі як AES і GCM , хоча й є одними з найнадійніших із доступних сьогодні, у майбутньому можуть виявитися вразливими.
- Експлойти безпеки можуть змусити PayPal вимкнути певні шифри в майбутньому.

Щоб мінімізувати вашу вразливість до поточних і майбутніх загроз, ми рекомендуємо не вказувати певні шифри у своїх інтеграціях.

Дозволити ідеальну пряму секретність

Технологія Perfect Forward Secrecy (PFS) призначена для запобігання компрометації довгострокового секретного ключа від впливу на конфіденційність минулих або майбутніх розмов. Ми рекомендуємо вам застосувати PFS у своїй інтеграції.

З реалізованим PFS будь-які захищені передачі, які ви записали в минулому, залишаються безпечними та не можуть бути скомпрометовані, навіть якщо поточний ключ скомпрометовано. Те саме стосується майбутніх передач. Без PFS, якщо скомпрометована одна передача, можуть бути скомпрометовані всі минулі та майбутні передачі.

Під час впровадження PFS вам потрібно дозволити протоколу узгоджувати найвищу версію TLS і ніколи не використовувати жорстко закодовані спеціальні шифри. Коли PFS увімкнено, узгодження протоколу TLS виконується на стороні PayPal. Обов'язково не обмежуйте шифри Діффі–Хеллмана (DHE) або еліптичної кривої Діффі–Хеллмана (ECDHE) у своїй інтеграції.

Візуальне представлення PFS можна знайти нижче: діаграма архітектури PFS .

Perfect Forward Secrecy (PFS) Overview

Difference between no PFS (with RSA) and PFS (with DH) in TLS/SSL/IPsec Connections

Key Exchange via RSA (no PFS)

Client

The client generates a session key, encrypts it via RSA, and sends it to the server.

Server

- Generates session key K_Sess
- (RSA Encryption) Encrypts K_Sess with the public long term key from the server K_Pub and sends it to server

Encrypted K_Sess

- (RSA Decryption) Decrypts K_Sess with its private key K_Priv

Communication encrypted with symmetric cipher using K_Sess

If the third party has access to K_Priv some day, it can subsequently decrypt all communication since it can reproduce all session keys.

Complete communication is stored by a third party

Evil Third Party

Key Agreement via DH (with PFS)

Client

Client and server deliver input to derive the session key. The session key itself is not transmitted through the network.

Server

- (Diffie-Hellman) Generates random value a and computes A
- Sends A to the server
- **Computes K_Sess from input of itself (a) and the server (B).**

A

B

- (Diffie-Hellman) Generates random value b and computes B
- Sends B to the client
- **Computes K_Sess from input of itself (b) and the client (A).**

Communication encrypted with symmetric cipher using K_Sess

Since K_Sess is freshly generated for each session, not transmitted on the net, and not encrypted with a long term key, a third party cannot decrypt the communication unless it breaks every single session key.

Complete communication is stored by a third party

Evil Third Party

by Johannes Weber | http://blog.webernetz.net | Last Change: 2014-07-18

Будьте пильні

Незважаючи на те, що ми б хотіли, щоб інтеграція була одноразовою діяльністю, яка розрахована на майбутнє, загроза кібератак вимагає постійної пильності. PayPal постійно працює над захистом наших продавців і випереджає тенденції безпеки в Інтернеті. Щоб зменшити свою вразливість, принаймні раз на рік перевіряйте свою інтеграцію на відповідність найкращим галузевим практикам. Виконання незначних кроків, описаних вище, може істотно змінити безпеку вашої інтеграції.

Безпечні програми

Щоб переконатися, що програми, які ви розробляєте, безпечні та оптимізовані для найкращої взаємодії з користувачем, дотримуйтесь найкращих практик, викладених у цьому розділі.

Не використовуйте WebView для відображення веб-сторінок PayPal у своїй програмі

Ваша програма не повинна використовувати WebView або подібний настроюваний механізм браузера для відображення веб-сторінок PayPal. Натомість використовуйте відповідний пакет SDK PayPal для керування роботою PayPal або запустіть веб-сторінку PayPal у системному браузері чи схваленому механізмі перегляду браузера, наприклад Safari View Controller на iOS або Chrome Custom Tabs на Android.

Інструкції з інформаційної безпеки для розробників

Платформа PayPal дозволяє розробникам створювати програми, які мають можливість здійснювати покупки від імені сторонніх клієнтів без перенаправлення клієнтів на paypal.com для завершення платіжних операцій. Такі ініційовані продавцем рахунки за одноразові платежі/платежі без передплати включають попереднє схвалення адаптивних платежів і контрольні транзакції Express Checkout.

Використання попередніх схвалень або довідкових транзакцій може забезпечити елегантний і безпроблемний досвід покупки для клієнта (вони ніколи не залишають додаток або веб-сайт, де вони роблять покупку). Однак

розробник такої програми повинен брати до уваги наведені нижче Інструкції з інформаційної безпеки, коли вони кодують програму, яка інтегрує функцію виставлення рахунків, ініційовану продавцем.

Щоб використовувати можливість попереднього схвалення, додатки повинні отримати явну згоду користувача PayPal на цей тип оплати «без входу». PayPal надає розробникам мову згоди під час розгляду заявки або раніше (за запитом). Щоб дізнатися більше, зверніться до Угоди розробника PayPal .

Інструкції з інформаційної безпеки стосуються можливостей попереднього затвердження та стосуються таких сфер:

- Аутентифікація
- Антифішинг
- Захист від підробки міжсайтового сценарію/міжсайтового запиту
- Процес безпеки сайту
- За запитом додавання певного вмісту або функцій для виявлення шахрайства

Вимоги до автентифікації

Автентифікація визначає, хто надсилає запит на доступ до системи чи програми. Наприклад, клієнт може ввести ідентифікатор користувача та пароль під час входу. Крім того, облікові дані автентифікації можуть бути отримані програмою з файлу cookie або FSO, наданого під час попереднього сеансу на веб-сайті.

Нижче наведено вимоги PayPal до автентифікації для програм, які використовують функцію попереднього схвалення без входу в систему.

- Паролі мають відповідати найкращим галузевим практикам щодо вмісту

Як правило, довжина пароля має бути не менше шести символів і містити принаймні один буквений і один цифровий символи, наприклад fr1gx9. Паролі-кандидати, які відповідають цим вимогам, також мають бути перевірені на відповідність словнику загальних паролів і списку правил. Наприклад, потенційний пароль blink182(назви музичної групи) слід відхилити як занадто поширений. Паролі, які збігаються з ідентифікатором користувача, також не проходять перевірку правил.

- Паролі для входу мають бути надійно зашифровані під час передачі та зберігатися в незворотній формі

Щоб гарантувати, що ніхто не зможе перехопити пароль під час входу, його потрібно надіслати через зашифрований канал, наприклад https. Це запобігає перехопленню пароля кимось, хто прослуховує мережу.

Крім того, пароль має зберігатися на сервері таким чином, щоб навіть внутрішні співробітники, які мають доступ до бази даних і ключів шифрування, не змогли отримати пароль у відкритому текстовому вигляді. Зазвичай це робиться хешуванням пароля за допомогою унікального ідентифікатора або солі, пов'язаної з окремим користувачем.

- Маркери сеансу, такі як файли cookie сеансу веб-переглядача, повинні бути захищені від крадіжки сеансу, передаючи їх лише в безпечних сеансах

Як зазначалося раніше, інформація, що зберігається в файлах cookie сеансу, зазвичай використовується для автентифікації кожного запиту до веб-сайту. Ці файли cookie мають бути захищені від злому будь-ким, хто може прослуховувати їх у тій самій мережі. Зазвичай це робиться шляхом позначення файлу cookie як безпечного, щоб він передавався лише під час підключення до сайту за допомогою https.

- Маркери сеансу мають бути згенеровані таким чином, щоб бути криптографічно надійними та високостійкими до передбачення значень cookie сеансу
- Токени сеансу, які будь-коли були неактивні більше 15 хвилин у будь-який час, повинні бути повторно автентифіковані за допомогою входу до обробки транзакцій PayPal

Щоб гарантувати, що комп'ютер не залишили на 15 хвилин після останнього виконання будь-якої дії, і зараз він активно використовується кимось іншим, потрібно, щоб перед виконанням транзакції PayPal було введено вхід для входу, якщо сеанс коли-небудь був неактивний протягом 15 хвилин.

- Повинен бути реалізований елемент керування, який запобігає атаці грубою силою облікових даних для входу

Поширеною атакою на веб-сайти є спроба входу за допомогою різних паролів, які часто використовуються для певного ідентифікатора входу. Повинен бути використаний якийсь метод, щоб гарантувати, що людина не зможе здійснити такий вид атаки. Загальним рішенням є блокування спроб входу в обліковий запис на певний період часу. Щоб гарантувати, що ці механізми не генерують засоби атак на облікові записи через відмову в обслуговуванні, ці блокування мають бути скасовані через певний період часу. (Кілька годин зазвичай.)

- Слід запровадити контроль, який запобігає вгаду паролів методом грубої сили, особливо якщо атака походить від ботнету

Як правило, це вимагатиме збору метаданих про входи в систему, реєстрації їх у центральному сховищі журналів, а потім виконання аналітики цих даних у реальному часі. У разі виявлення атаки грубою силою буде ввімкнено надійну CAPTCHA (стійку до машинних/авторських атак). Існують інші методи впровадження, але це найменш інвазивний з точки зору взаємодії з користувачем. Зауважте, що це концептуально та функціонально відрізняється від A6.

- Має бути можливість підтримувати користувачів, зареєстрованих за допомогою ключів безпеки для транзакцій PayPal

Необхідно звернути увагу на користувачів, які підписалися на вищий рівень автентифікації у своєму обліковому записі PayPal, і ці користувачі все ще повинні мати можливість належним чином входити в систему, якщо потрібно. Прикладом є ключ безпеки PayPal, який вимагає введення одноразового пароля.

- Уся передача конфіденційної інформації (наприклад, паролі та сеансові файли cookie) повинна бути захищена за допомогою TLS

Усі версії стеку SSL небезпечні, тому їх не слід використовувати, натомість рекомендуються новіші протоколи TLS.

- Необхідно вжити технічних заходів, щоб гарантувати, що частини сайту, яким не потрібно мати можливість читати або записувати файли cookie сеансу, не зможуть це робити

Це просто вимога через стандартні правила прив’язки файлів cookie/домену, які означають, що багато частин великої програми мають доступ до набору файлів cookie, що може спричинити загрози безпеці.

- Вимоги до антифішингу

PayPal витратив багато сил на боротьбу зі зловмисниками, які надсилають електронні листи клієнтам, які нічого не підозрюють, заявляючи, що вони з PayPal, із посиланнями на шкідливі сайти, схожі на PayPal. Ці сайти обманюють клієнта та захоплюють облікові дані для входу, а іноді викрадають фінансову та особисту інформацію.

- Облікові дані для входу потрібно збирати лише на сторінках, які використовують протокол https із сертифікатами розширеної перевірки (EV).

Щоб дозволити клієнтам перевірити, чи вони справді підключені до сайту партнера, і заохочувати загальні передові практики, облікові дані для входу потрібно збирати на сторінках із увімкненим протоколом https і з використанням сертифікатів розширеної перевірки (EV).

- Має бути процес проактивного моніторингу та агресивного вимкнення підроблених сайтів

PayPal використовує кілька методів для виявлення підроблених сайтів, які використовуються для підтримки різних форм фішингу. Після виявлення сайти завчасно припиняють роботу, щоб запобігти подальшому ризику для наших клієнтів. Щоб забезпечити повну безпеку моделі, партнер також повинен запровадити моніторинг і видалення сайту.

- Повинен існувати процес, за допомогою якого клієнти можуть повідомляти про підроблені сайти

Ці сайти слід перевірити вручну, чи вони справді підроблені; Після перевірки URL-адреси слід якомога швидше надіслати до галузевих списків заборони, таких як APWG, MarkMonitor тощо.

- Захист від підробки міжсайтового сценарію/міжсайтового запиту

Атака на веб-сайти може бути здійснена шляхом відображення інформації, що надходить із браузера користувача, назад на веб-сторінку, яка містить HTML або JavaScript, які можна використовувати для зміни зовнішнього вигляду веб-сторінки, водночас вказуючи за допомогою URL-адреси, що клієнт підключено до оригінального веб-сайту. Атака підробки міжсайтового запиту здійснюється шляхом представлення посилання на сайт, на якому клієнт, можливо, вже автентифікований. Посилання міститиме закодовану інформацію, яка змусить сайт виконати якийсь запит, який користувач не мав на увазі.

Основна веб-програма повинна:

- Містять відповідні компоненти рамки, які гарантують, що сайт не піддається атакам міжсайтового сценарію (XSS) .
- Містить відповідні компоненти рамки, які гарантують, що сайт не піддається атакам підробки міжсайтових запитів (XSRF) .
- Періодично проходьте тестування за допомогою комерційно доступного інструменту або комерційно доступної служби, щоб продемонструвати, що він не вразливий ні до атак XSS, ні до XSRF. У будь-якому випадку сайт слід тестувати в міру розгортання нового коду.

- Процес безпеки сайту

Повинен існувати процес управління вразливістю, за допомогою якого керується інфраструктура, на якій працює сайт.

Має бути канал інформації про вразливості з одного або кількох надійних джерел (наприклад, IT-ISAC, iDefense, Symantec тощо); дані з цього каналу слід переглянути, щоб визначити, які вразливості є актуальними. Уразливості слід

класифікувати за категоріями критичності та застосовувати відповідні виправлення на основі цього позначення критичності.

Подібним чином має існувати процес, за допомогою якого слід керувати вразливими місцями в бібліотеках додатків, а також встановлювати пріоритети процесу відновлення/випуску додатків.

Повинен існувати безпечний процес розробки, який описує стандарти, які допомагають зробити сайт безпечним, і весь відповідний персонал (не обов'язково всі розробники) повинен бути навчений методам безпечної розробки.

Має бути спосіб, за допомогою якого можна продемонструвати, що процес безпеки сайту працює.

Це може статися через використання галузевих стандартних перевірок, таких як PCI, перевірок, керованих аудиторами, таких як SAS/70, або комерційних перевірок, таких як сертифікація Cybertrust.

- Попередньо схвалені вдосконалення платіжного продукту

PayPal може попросити додати певний вміст або функції до певних веб-сторінок, щоб допомогти виявити шахрайство. Загалом, ці запити включатимуть технічні зміни, які будуть «легкими» для досягнення. Однак ми не можемо визначити, чи знадобляться такі зміни, доки продукт не буде впроваджено у виробництво та потенційно підданий атаці шахраїв.

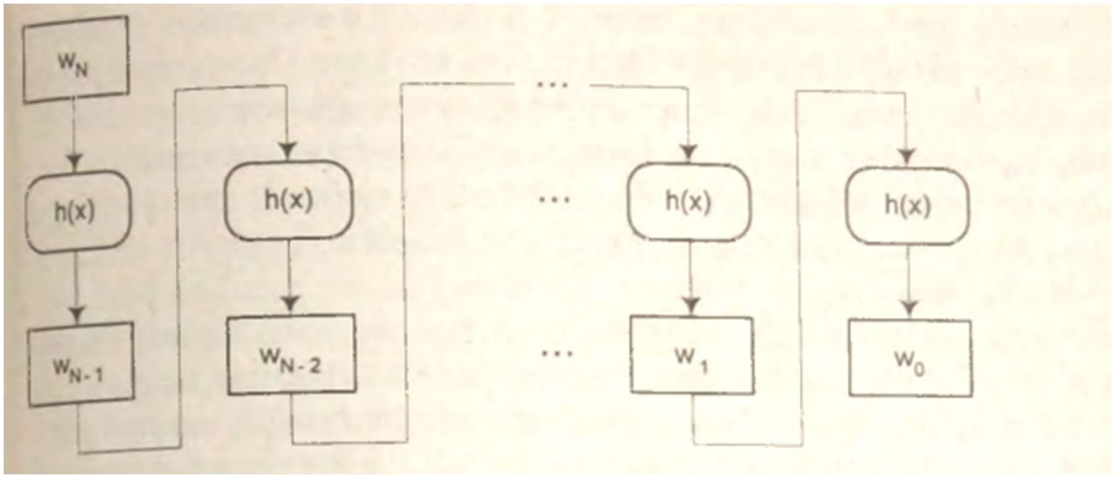
Оскільки PayPal продовжує вдосконалювати Попередньо схвалений платіжний продукт, партнер своєчасно запровадить розширену версію продукту. У майбутньому PayPal може розширити продукт попередньо схвалених платежів за допомогою перенаправлення на paypal.com для перевірки ризиків і безпеки; це перенаправлення зазвичай має бути прозорим для користувача, якщо не активуються засоби контролю ризиків PayPal.

PayWord та MicroMint

PayWord є однією з двох варіацій Millicent. Іншою варіацією є MicroMint. Обидві ці системи були запропоновані Рональдом Рівестом (Laboratory of Computer Science of MIT) і Аді Шаміром (Weizmann Institute of Science in Izrael). В обох випадках авторизація відбувається в автономному режимі.

Система розрахунків при використанні PayWord грунтується на використанні умовних одиниць, названих payword. Як і в системі Millicent, учасниками транзакції PayWord є продавець, покупець і брокер. Кожен брокер здійснює авторизацію покупців при їх реєстрації з метою придбання ними умовних одиниць payword, які використовуються для оплати товарів і послуг продавців в мережі брокера. У той же час брокер зобов'язується компенсувати продавцям кредит, зібраний ними в умовних одиницях payword.

Ланцюжок payword формується шляхом рекурсивного застосування хеш-функції до початкового значенням W_0. Кожен отриманий вираз є деяким еквівалентом кредита для продавця, пов'язаного з мережею брокера. Ілюстрація послідовності виразів приведена на рис. нижче.



Завдання верифікації для продавця полягає в перевірці цифрового підпису отриманого початкового значення з використанням алгоритму шифрування з відкритим ключем. Завдяки єдиної операції перевірки цифрового підпису продавець виробляє аутентифікацію всієї послідовності описаних вище виразів. Завдяки такому методу реалізації процедури верифікації всі перевірки виконуються локально і зв'язок з брокером при проведенні платежів не потрібно.

Проте локальні процедури провірки вимагають наявності списків анульованих платіжних одиниць і параметрів процедур верифікація.

Щодня брокер отримує список останніх використаних користувачами системи умовних одиниць, що дозволяє брокеру здійснювати щоденне оновлення списку дійсних payword. Крім того, брокер створює та оновлює список користувачів, займаються шахрайством, і тих користувачів, які заявили про втрату або крадіжку секретних ключів користувача SK_U.

Опис відносин між продавцями в мережі брокера і процесу проведення фінансових розрахунків між продавцями і брокером лежить за межами специфікацій PayWord.

Реєстрація та нарахування коштів. Користувач звертається до брокера з метою відкриття рахунку. При цьому користувач з користуванням захищеного каналу зв'язку передає брокеру свої банківські реквізити або номер банківської карти. Даний канал може бути простою поштовою пересилкою або каналом Internet, захищеним з використанням протоколу SSL. Після успішного завершення процесу реєстрації Користувач отримує карту передплатника наступного формату:

$$C_U = \{ B, U, A_U, PK_U, E, l_U \} SK_B.$$

Отримана карта підписана з використанням секретного ключа брокера SK_B. У своєму складі карта містить наступні поля: ім'я брокера B, ім'я користувача U, адреса користувача A11 (який може бути поштовою адресою, IP-адресою або бути адресою електронної пошти), відкритий ключ користувача PK_U, термін придатності карти E і додаткову інформацію I_U (наприклад, номер картки, обмеження кредиту для кожного з продавців, дані про брокера, правила і умови проведення угод і т.д.). Таким чином, термін "карта передплатника" є позначенням сертифіката, яке було введено з метою виключення конфліктів позначень з сертифікатами аутентифікації x. 509.

Карта передплатника дозволяє користувачеві зробити авторизацію з метою створення ланцюжки payword з допомогу транзакції з брокером. При отриманні карти передплатника продавець має можливість перевірки отриманої картки з використанням відкритого ключа брокера PK_B. Таким чином, продавець отримує гарантію проведення розрахунків по отриманим за допомогою картки платежам до закінчення терміну придатності картки. В якості міри обережності доставка замовлених товарів і послуг проводиться за адресою клієнта, вказаною в карті передплатника.

Незважаючи на популярність продавцю даних Покупця, система не встановлює ніяких явних зв'язків між фактом продажу окремого предмета і особистістю покупця. Це підвищує конфіденційність і ускладнює відстеження транзакцій покупців. Завданням передплатника є захист свого секретного ключа SKu. При цьому факт збереження секретного ключа клієнта на своєму комп'ютері є одним із слабких місць системи. Однак економія, що досягається при використанні системи PayWord, виправдовує певний рівень ризику при проведенні фінансових операцій.

Процес покупки. Процес покупки складається з двох фаз: передача ланцюжка payword і використання засоби.

Фаза 1. Передача ланцюжка payword. В ході даної фази встановлюються коротко-термінові відносини між продавцем і покупцем. Ці відносини аналогічні відносинами, що встановлюються при видачі боргового зобов'язання. При передачі ланцюжка payword покупець обіцяє здійснити оплату кредиту брокера відповідно до одиницями payword W_1,..., W_{N-1}, W_N, які продавець представить брокеру для проведення розрахунків до закінчення їх терміну придатності.

Кожен раз, коли Покупець бажає здійснити оплату з використанням системи PayWord, повинна бути створена нова ланцюжок payword {W_i,..., W_{N-1}, W_N). Процес створення зазначеної ланцюжка відбувається наступним чином:

1. Нехай N-число умовних одиниць, необхідних для оплати покупки. Виберемо випадкове число W_N, яке буде N-ю одиницею payword.

2. З використанням хеш-функції H() (наприклад, MD5 або SHA) отримуємо значення W_{N-1}. Таким чином, $W_{N-1} = H(W_N)$.

3. Для отримання W_{N-2} застосовуємо хеш-функцію ще раз. Отже,

$$W_{N-2} = H(W_{N-1}) = H(H(W_N)\}.$$

4. Таким чином, проводиться створення всього ланцюжка {W_o,..., W_{N-1}, W_N).

Для встановлення відповідності між продавцем і створеної ланцюжком покупець створює платіжне зобов'язання M наступної форми:

$$M = \{ V, C_U, W_g, D, I_M \} SK_u,$$

де поле V позначає продавця Cu, являє собою карту передплатника, Wo є коренем ланцюжка payword, поле D позначає термін придатності зобов'язання і Iм-додаткову інформацію (наприклад, довжину ланцюжка або значення однієї одиниці payword). Перед відправкою створеного зобов'язання продавцю проводиться вироблення його цифрового підпису з використанням в секретному Ключі покупця SKu. Цифровий підпис позначений як { } SKu.

Процес вироблення зобов'язання є найскладнішим обчислювальним процесом Користувача, так як він включає в себе формування цифрового підпису з використанням алгоритму RSA.

Завдяки отриманню Карти передплатника і платіжного зобов'язання M продавець отримує можливість перевірки автентичності покупця з використанням відкритих ключів покупця і брокера. Перевірка автентичності досягається за рахунок перевірки цифрового підпису отриманого продавцем зобов'язання M та цифрового підпису картки передплатник. Після проведення зазначених вище перевірок продавець може бути впевнений, що терміни придатності E і D, поряд зі значенням кореня ланцюжка payword Wo, були передані і коректно отримані.

Використання коштів. Користувач виробляє витрачання payword в висхідному порядку: W1 перед W2 і т. д. якщо ціна замовленого товару становить i одиниць payword, то платіж P користувача U продавцю V буде виглядати наступним чином:

$$P = (W_j, i).$$

Таким чином, платіж не володіє цифровим підписом.

Для перевірки отриманого платежу продавець повинен виконати i операцій хешуванням для отримання значення кореня ланцюжка payword Wo (отриманої в складі платіжного зобов'язання M). Однак кількість необхідних для процедури перевірки обчислень може бути скорочено при обліку способу побудови ланцюжка $\{W_1,..., W_{N-1}, W_N\}$. Припустимо, що одиниця payword W, вже була використана і при цьому j < i. в цьому випадку верифікація платежу P зажадає виконання (i - j) операцій хешування до досягнення. Значення Wj, замість виконання i операцій хешування до досягнення значення Wo. Таким чином,

продавець може зменшити складність процедури перевірки платежу шляхом зберігання та використання історії витрачання коштів користувача, а саме значення Wj, для кожного користувача окремо.

Зауваження:

- Тільки зобов'язання має бути підписано для забезпечення цілісності Wo. Платіж не повинен володіти підписом.
- Автентичність користувача перевіряється за допомогою картки абонента,
- яка в цьому випадку може бути названа сертифікатом користувача.
- Процеси аутентифікації та авторизації виконуються локально в автономному режимі. Таким чином, продавець не відчуває необхідності в забезпеченні сеансу звязку з брокером в процесі аутентифікації та авторизації.
- У платежі не вказується оплачуваний ним товар, що призводить до ускладнення відслідковування транзакцій. Проте це не забезпечує захисту від недбайливих або нечесних продавців.
- Брокер грає роль судді з дрібних справ, крім своєї основної ролі довіреної
- третьої сторони.
- Продавець має можливість збереження записів про всі отримані одиницях payword (в тому числі про використані і компенсованих брокером) до закінчення їх терміну придатності з метою захисту від їх повторного використання.

Фінансові розрахунки. Для перевірки автентичності карт передплатника Cu продавець не обов'язково повинен володіти встановленими відносинами з брокером. Для перевірки автентичності продавцю цілком достатньо знання відкритого ключа брокера РКв. Проте для отримання компенсації за накопиченими платежами продавцю потрібні встановлені відносини з брокером системи PayWord.

Продавець періодично (наприклад, щодня) відправляє брокеру запит на компенсацію коштів, накопичених в процесі оплати товарів і послуг передплатниками мережі брокера. Для групи платежів кожного передплатника формується окремий запит на компенса цію. Дане повідомлення містить в своєму складі платіжне зобов'язання M і останню отриману продавцем одиницю payword.

Брокер проводить перевірку кожного отриманого платіжного зобов'язання придопомоги відкритого ключа користувача PK_U, отриманого з карти передплатника. Крім того, при перевірці платіжних зобов'язань враховується термін придатності як самого платіжного зобов'язання, так і карти передплатника, з якої був отриманий відкритий ключ користувач. Далі, для перевірки правильності останньої одиниці payword, наприклад W_K, брокер виробляє K до операцій хешування для досягнення W_0.

Брокер виробляє угруповання невеликих обсягів платежів перед їх списанням зі рахунки банківської карти. Інтерфейс з рахунком банківської карти реалізується з урахуванням потреб до безпеки та протоколом функціонування банківської мережі. Таким чином, покупець виявляється пов'язаний договором з двома організаціями: банком, що випустив платіжну карту, і брокером PayWord.

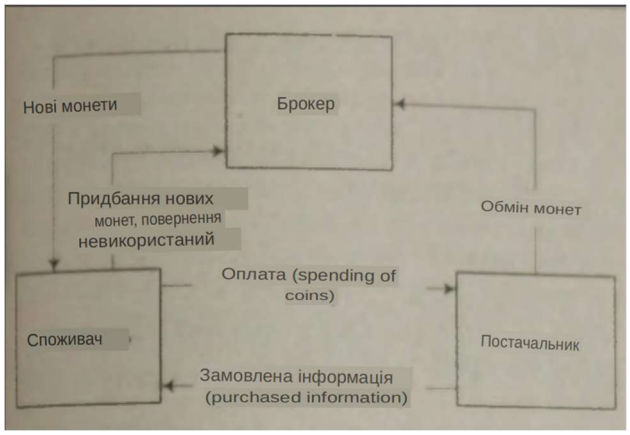
Безпосередньо фінансові розрахунки на основі дійсних грошових одиниць здійснюються з використанням внутрішніх банківських механізмів і лежать поза специфікацій системи PayWord.

Характеристика	PayWord
Верифікація	В автономному режимі
Використання передоплати	Не використовується
Представлення коштів оплата	Цифрова функція (хеш)
Безпека	Захист від повторного використання
Формат зберігання коштів оплата	Використовувати послідовний хеш-операції
Можливість використання різних валют	Бувши

MICROMINT. MicroMint є другою системою проведення мікроплатежів, розробленої Р. Рівестом і А. Шаміром.

Система MicroMint заснована на використанні жетонів, які називаються монетами MicroMint. Перевірка правильності бітових послідовностей, що відповідають жетонам, може бути виконана дуже просто, в той час як їх створення є дуже працею Кім процесом. Як і у випадку виробництва металевих

монет, у випадку системи Micro Mint вартість виробництва однієї одиниці оплати знижується при збільшенні обсягів виробництво. У той же час одиниці оплати з низькою вартістю економічно невигідні. На відміну від системи PayWord, система MicroMint не використовує алгоритми шифрування з відкритим ключем з метою зниження обчислювального навантаження на системи. Цикл використання монет системи MicroMint представлений на рис. нижче.



Продавці щоденно проводять обмін зібраних монет системи MicroMint на дійсні кошти оплата. Термін придатності нових монет становить близько одного місяця. Після закінчення цього періоду проводиться повернення монет з вичерпаним терміном придатності брокеру і їх обмін на нові монети або інші грошові кошти оплати.

Реєстрація та нарахування коштів. Купівля монет MicroMint у брокера проводиться з використанням класичних засобів оплати (банківські картки, чеки та ін.). Брокер повинен зберігати список всіх купленийних у нього монет кожним з передплатників його мережі. При цьому в системі MicroMint невиконується попередня реєстрація відносин учасників мережі одного брокера.

Якщо отримання монет MicroMint проводиться з використанням мережі Internet, то захись від крадіжки переданих по мережі монет може відбуватися з використанням симетричних алгоритмів шифрування. Дане рішення стає можливим завдяки тривалого характеру відносин між передплатником мережі брокера і брокером.

Процес покупки. В процесі покупки відбувається обмін придбаних раніше монет MicroMint на замовлені товари або сервіси. При цьому продавець повинен провести перевірку автентичності отриманих монет MicroMint. Однак

ця перевірка не включає перевірки повторного використання монет. Це є наслідком технології відкладеної перевірки повторного використання монет MicroMint, яка реалізується брокером. Така таким чином, брокер може легко виявити повторне використання одних і тих же монет MicroMint і повідомити про це продавцю.

Фінансові розрахунки. Кожні 24 години продавці надсилають брокеру дані про накопичені монети MicroMint. Після проведення процесу верифікації отриманих монет брокер виробляє компенсацію фінансових коштів продавцю відповідно до обсягу отриманих від продавця коштів, за винятком комісійних. При цьому брокер може вільно відмовити продавцю у прийомі монет, які раніше вже були компенсовані цьому або іншому продавцю.

Безпека. В основі системи забезпечення безпеки лежить припущення про недоцільність підробки монет MicroMint з метою отримання помітного обсягу коштів. Таким чином, механізми забезпечення безпеки орієнтовані на виявлення множинних фальсифікацій, таких, як множинна підміна, крадіжка монет або регулярне повторне використання монет. При створенні монети MicroMint використовується k колізій хеш-функції

$$H: x > y,$$

де x і y є векторами довжин n і m (m < n) відповідно. Колізією називають ситуацію, при якій для двох різних параметрів x_1 і x_2 хеш-функція дає однаковий результат:

$$H(x_1) = H(x_2) = y.$$

Інакше кажучи, для отримання k колізій потрібне дотримання наступних умов:

$$H(x_1) = H(x_2) = \ldots = H(x_{k-1}) = H(x_k) = y.$$

При цьому вектори x_1 ... x_k повинні бути різні. Таким чином, монета C має наступний формат:

$$C = \{ X_1, X_2, ..., X_{k-1}, X_k \}.$$

Процес верифікації монети полягає в перевірці збігу значень хеш-функцій для елементів, що складають монету. Наприклад, припустимо, що

$$n = 52 \text{ і } k = 4.$$

У разі використання стандартних функцій хешування, таких, як MD5 і SHA, довжина результуючих значень хеш-функцій становить 128 і 160 біт відповідно.

Таким чином, для отримання шуканих 52 біт, беруть Молодші 52 розряду результату. При цьому необхідно мати на увазі, що зазначений вище процес верифікації монет не дозволяє виявляти повторного використання монети або її підробки.

Захист від підробки. Виходячи їх малої популярності підробок невеликих обсягів монет MicroMint, всі захисні заходи орієнтовані на захист від великомасштабних підробок. Нижче наведено деякі з цих заходів.

- Щомісячне оновлення критеріїв перевірки правильності, наприклад:
  - ✓ відповідність старших розрядів результатів хеш-функції певній масці;
  - ✓ випущені монети належать до певної множини, чиє хеш-значення
  - ✓ має задану структуру, що дозволяє виділяти підроблені монети на основі аналізу отриманого хеш-значення;
  - ✓ різні x_i, повинні задовольняти деяким умовам.

- Брокер може збільшити складність обчислень, необхідних для створення монет MicroMint, за рахунок завчасного початку створення монет. Це призведе до додаткових складнощів при підробці монет.

- В якості крайнього заходу в разі злому сервера MicroMint може бути проведений відкликання всіх вироблених монет з подальшою їх заміною на нові монети.

Захист від крадіжки монет. Крадіжка монет в процесі їх збору у брокера може бути попереджена з використанням алгоритмів шифрування. При цьому в якості

останніх може використовуватися алгоритм шифрування з відкритим ключем. Це стає можливими завдяки тривалому характеру відносин між брокером і продавцем. Інший підходом, що дозволяє обійтися без шифрування даних в процесі обміну, може слугувати персоналізація монет для кожного з передплатників мережі брокера. В результаті, при умови втрати анонімності, з'являється можливість повного контролю користувача за використанням монет, а монети, в свою чергу, стають електронними аналогами чек. З іншого боку, якщо персоналізація монет була проведена щодо продавця, то їх крадіжка буде марна, так як продавець негайно помістить їх в чорний список і відмовить в їх обслуговуванні. В цьому випадку буде досягнута анонімність при використанні монет на шкоду їх універсальності.

Захист від повторного використання монет. Брокер може виявити повторне Використання монет завдяки зберіганню записів про монети, випущених для кожного користувача його мережі. При отриманні від продавців запитів на компенсацію накопичених монет брокер має можливість ідентифікації продавців. Далі, за сприяння продавців і використовуючи накопичену ними інформацію, з'являється можливість відстеження покупців, які повторно використовували монети Micromine. Однак в силу відмови від використання цифрових підписів в системі Micromine однозначно довести особистість фальсифікації практично неможливо. Проте надалі брокер може відмовити підозрюваному користувачеві у видачі монет. У будь-якому випадку виявлення повторного використання монет в системі MicroMint є складним процесом. Однак невеликий обсяг транзакцій робить можливим використання цієї системи при допустимій величині ризику.

Порівняльна таблиця систем Payword та MicroMint.

Характеристика	PayWord	MicroMint
Безпека	Алгоритм шифрування з відкритим ключем. Перевірка чорного списку	Шифрування не використовується. Відсутній простий механізм захисту від повторного Використання монет
Універсальність використання умовна одиниця	Прив'язка до продавця	Прив'язка відсутня, але можлива
Характер операцій з умовними одиниця	Кредит	Дебет
Номінальне значення умовної одиниця	1 payword = 1 цент	1 монета = 1 цент
Верифікація (у брокера)	В автономному режимі	В автономному режимі
Формат зберігання коштів	У вигляді послідовності хеш-значення	У вигляді послідовності хеш-значень
Підтримка різних національних валют	Бувши	Бувши
Анонімність	Ні	Ні
Відстежуваність	Ні	Бувши