УДК 519.713

А. О. БОЙКО, канд. техн. наук, Ю. О. СЕРГІЙЧУК, А. В. ТРИПІЛКА

АНАЛІЗ УМОВ ІМПЛЕМЕНТАЦІЇ ЕЛЕКТРОННОЇ ІДЕНТИФІКАЦІЇ ТА ДОВІРЧИХ ПОСЛУГ ДЛЯ ЕЛЕКТРОННИХ ОПЕРАЦІЙ НА ВНУТРІШНЬОМУ РИНКУ

1. Вимоги проекту регламенту європейського парламенту і ради щодо електронної ідентифікації та довірчих послуг для електронних операцій на внутрішньому ринку

Послуга автентифікації веб-сайтів є необхідною для надання онлайн-послуг як державними органами (он-лайн декларування податків, запит свідоцтва про народження, участь в електронних процедурах державних закупівель і т. і.), так і приватними підприємствами (онлайн-торгівля, банківська діяльність, фінансові послуги тощо).

Послуга автентифікації веб-сайтів згідно з проектом Регламенту Європейського парламенту і Ради щодо електронної ідентифікації та довірчих послуг для електронних операцій на внутрішньому ринку [1] (далі – Регламент) має забезпечувати підтвердження справжності веб-сайту та особи, що ним володіє, чим ускладнити задачу фальсифікації веб-сайтів і тим самим знизити шахрайство на транскордонному рівні.

Стаття 37 Регламенту встановлює вимоги до кваліфікованих сертифікатів для аутентифікації веб-сайту, які можуть бути використані, щоб гарантувати справжність веб-сайту. Кваліфікований сертифікат для аутентифікації веб-сайту надає мінімальну кількість достовірної інформації про веб-сайт та про легальне існування свого власника.

Кваліфікований сертифікат для перевірки справжності веб-сайту – це атестат, що надає можливість перевірки справжності сайту та пов’язує сайт з особою, якій видано сертифікат, що видається кваліфікованим провайдером довірчих послуг і відповідає вимогам, викладеним в Додатку IV [1].

2. Правові заходи, необхідні для імплементації регламенту в Україні

2.1. Формат сертифікатів

Формат сертифікатів у системі ЕЦП України визначається стандартом ДСТУ ISO/IEC 9594-8:2006 (IDT) [2], значення окремих полів уточнюються Вимогами до формату посиленого сертифіката відкритого ключа, затвердженими спільним Наказом Міністерства юстиції України та Адміністрації Держспецзв’язку від 20.08.2012 № 1236/5/453 [3].

Однак на момент гармонізації стандарту в Україні відповідний міжнародний стандарт ISO/IEC 9594-8:2001 [2] вже був відкликаний, тому стандарт ДСТУ ISO/IEC 9594-8:2006 (IDT) [2] і досі не опублікований.

Ні стандарт ДСТУ ISO/IEC 9594-8:2006 (IDT) [2], ні вимоги до формату посиленого сертифіката відкритого ключа [3] не визначають поля сертифіката, у якому можливо було б вказати доменне ім’я сайту, чого вимагає проект Регламенту. Наявність такого поля передбачена у ISO/IEC 9594-8:2005 [4] (вже відкликаний), ISO/IEC 9594-8:2008 [5] (діючий) та у фінальній чернетці стандарту ISO/IEC FDIS 9594-8 [6] (планується введення в дію у 2013 - 2014 рр.)

Таким чином, з метою виконання Регламенту доцільно гармонізувати стандарт ISO/IEC 9594-8 у редакції ISO/IEC FDIS 9594-8 [6] одразу після введення його в дію на міжнародному рівні.

2.2. Стандарти криптографічних перетворень

Стандарти криптографічних перетворень, які використовуються у ЄС, відрізняються від тих, що використовуються в Україні. Тому, для досягнення мети проекту Регламенту [1] – безперешкодної транскордонної взаємодії – необхідно забезпечити сумісність криптографічних перетворень на рівні стандартів. Можливі два шляхи: гармонізація європейських стандартів та

просування вітчизняних стандартів у якості міжнародних стандартів (ISO/IEC, IETF тощо). Останній шлях було обрано Російською Федерацією. Було випущено RFC 4491 [7], що визначає представлення параметрів та ключів для криптографічних перетворень за стандартами ГОСТ Р 34.10-94, ГОСТ Р 34.10-2001, ГОСТ Р 34.11-94 у сертифікатах відкритих ключів, та RFC 4357 [8], що визначає порядок використання ГОСТ Р 34.10-94, ГОСТ Р 34.10-2001, ГОСТ Р 34.11-94, ГОСТ 28147-89 у протоколах SSL/TLS. Слід зауважити, що вказані стандарти криптографічних перетворень та RFC не застосовуються за межами Російської Федерації, зокрема їх реалізації відключені у стандартній поставці пакету OpenSSL. Користувачі пакету OpenSSL вимушені вмикати підтримку ГОСТ Р 34.10-94, ГОСТ Р 34.10-2001, ГОСТ Р 34.11-94, ГОСТ 28147-89 у протоколах SSL/TLS окремо.

Вибір оптимального шляху забезпечення транскордонної взаємодії в рамках проекту Регламенту є питанням для окремого дослідження, тим більше у проекті Регламенту наголошується на його технологічній нейтральності – незалежності від технічних рішень і стандартів.

3. Технічні заходи, необхідні для імплементації регламенту в Україні

3.1. Модель порушника

Для автентифікації веб-сайтів застосовується протокол Secure Socket Layer (SSL) [9], удосконаленням якого є протокол Transport Layer Security (TLS) версій 1.0, 1.1 та 1.2, закріплений у RFC 2246 [10], RFC 4346 [11] та RFC 5246 [12] відповідно. Далі по тексту, посилання на вказані протоколи здійснюватиметься як SSL/TLS, якщо не буде потреби явно вказати версію протоколу.

Діючі сторони у протоколі SSL/TLS:

- веб-сайт;
- клієнт;
- центр сертифікації ключів (ЦСК).

Ініціатором взаємодії є клієнт. Він, як правило, не має власного сертифіката відкритого ключа та відповідного йому особистого ключа, тому генерує особистий та відповідний йому відкритий ключ для кожного сеансу роботи з веб-сайтом.

У відповідь веб-сайт надсилає клієнту власний сертифікат відкритого ключа та запит на узгодження криптографічних алгоритмів і параметрів.

Клієнт перевіряє сертифікат і звертається до ЦСК для перевірки його чинності шляхом запиту списків відкликаних сертифікатів (CRL) або за протоколом Online Certificate Status Response (OCSP). Якщо виявиться, що сертифікат веб-сайту нечинний, то протокол завершується.

Якщо сертифікат веб-сайту чинний, то клієнт узгоджує з веб-сайтом криптографічні алгоритми і параметри, що будуть використовуватись, встановлює спільні ключі для симетричних алгоритмів шифрування та формування кодів автентифікації повідомлень.

Спільні ключі для симетричних алгоритмів шифрування та формування кодів автентифікації повідомлень використовуються для встановлення захищеного каналу зв’язку між клієнтом та веб-сайтом.

Метою порушника є нав’язування хибних повідомлень (веб-сторінок) клієнту від імені веб-сайта та перенаправлення повідомлень від клієнта на веб-сайт порушника. Додатковою метою порушника може бути перехоплення повідомлень від веб-сайта клієнту та нав’язування хибних повідомлень від імені клієнта. Таким чином метою порушника є порушення конфіденційності, цілісності та автентичності повідомлень, що передаються між клієнтом та веб-сайтом.

Схема взаємодії сторін при доступі до автентифікованого веб-сайта наведена на рис. 1.

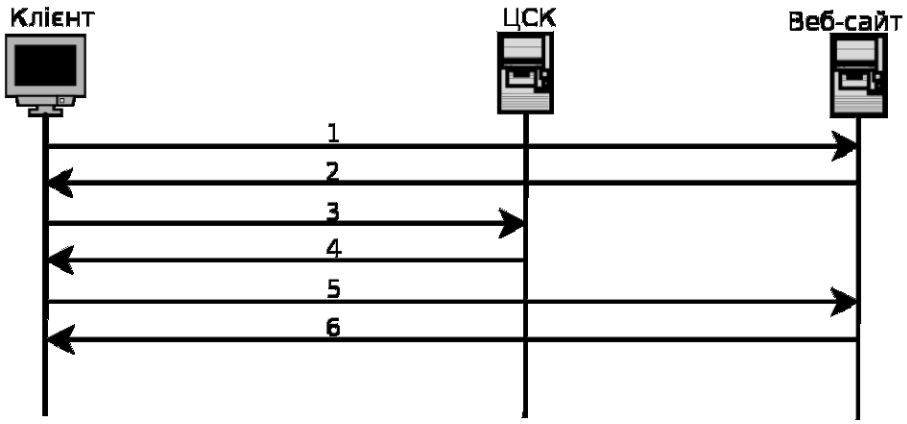

Рис. 1. Схема взаємодії сторін при доступі до автентифікованого веб-сайта

Пояснення до рисунку:

1 – клієнт звертається до веб-сайта за протоколом HTTPS;

2 – веб-сайт надсилає у відповідь власний сертифікат;

3 – клієнт звертається до ЦСК за протоколом OCSP для встановлення статуса сертифіката;

4 – ЦСК надсилає статус сертифіката;

5 – якщо сертифікат чинний, то клієнт ініціює встановлення захищеного з'єднання між клієнтом і веб-сайтом;

6 – веб-сайт встановлює захищене з'єднання з клієнтом.

Можливості порушника оцінюються як можливості спецслужби технологічно розвинутої держави. Порушник може:

- перехоплювати повідомлення, що передаються між клієнтом і веб-сайтом;
- модифікувати повідомлення, що передаються між клієнтом і веб-сайтом;
- створювати і направляти повідомлення у канал зв'язку між клієнтом і веб-сайтом;
- перехоплювати та модифікувати повідомлення, що передаються між клієнтом та іншими учасниками інформаційного обміну (ЦСК, DNS-серверами, іншими веб-сайтами тощо).

3.2. Атаки на протоколи автентифікації Web-сайтів і засоби протидії їм

3.2.1. Розповсюдження сертифікатів. У більшості криптографічних протоколів вважається, що всі учасники мають сертифікат відкритого ключа. Сертифікат відкритого ключа видається уповноваженою особою – центром сертифікації ключів при безпосередньому контакті ЦСК і учасника протоколу. Разом із сертифікатом відкритого ключа учасника протоколу ЦСК передає і автентифіковані копії усіх сертифікатів у ланцюжку аж до кореневого ЦСК.

Однак у випадку протоколів автентифікації веб-сайту клієнт, як правило, не має власного сертифіката, тому і не отримує ланцюжка довіреним чином. Без наявності довіреного ланцюжка сертифікатів клієнт не зможе перевірити справжність сертифіката веб-сайта, що дозволить порушнику нав'язати клієнту відповідь веб-сайта з сформованим порушником сертифікатом.

З метою попередження цієї атаки кореневі сертифікати розповсюджуються у складі операційних систем, веб-браузерів тощо. З метою забезпечення безперешкодної транскордонної взаємодії необхідно розробити механізми довіреного розповсюдження кореневого сертифіката Центрального засвідчувального органу України, зокрема, у складі операційних систем, веб-браузерів.

3.2.2. SSL Strip. SSL Strip – атака на SSL зєднання (протокол HTTPS), що базується на атаці «людина посередині» («Man in The Middle» – «MiTM») [13].

Атака можлива у випадку, коли користувач має можливість переходити із HTTP сторінки веб-ресурсу на HTTPS сторінку. У цьому разі порушник має можливість підмінити усі захищені посилання веб-сторінки на їх не захищені аналоги.

Сутність атаки полягає у тому, що порушник який знаходиться між клієнтом і веб-сайтом, підмінює усі запити за протоколом HTTPS на запити за протоколом HTTP, і, відповідно, отримує від веб-сайта сторінки, призначені для клієнта, за протоколом HTTPS, розшифровує їх, модифікує на власний розсуд і надсилає клієнту за протоколом HTTP. Таким чином, клієнт використовує для передачі чутливих даних незахищені запити за протоколом HTTP, за рахунок чого порушник має можливість читати та модифікувати чутливі дані, які передаються між клієнтом і веб-сайтом у незашифрованому вигляді без контролю цілісності та автентичності.

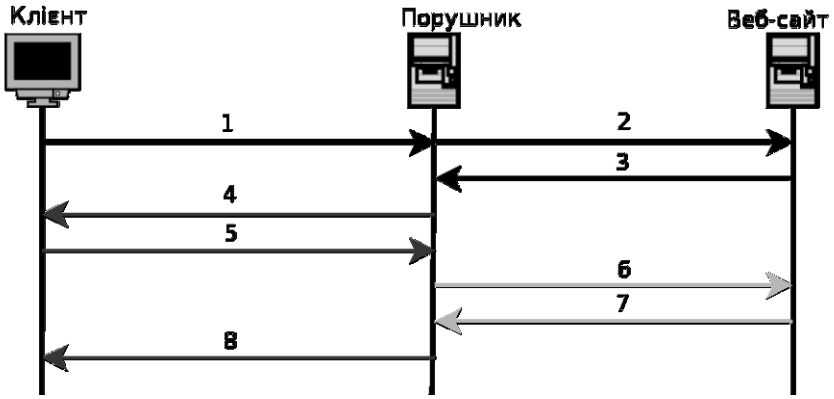

Рис. 2. Схема атаки SSL Strip

Пояснення до рисунку:

1 – клієнт надсилає запит до веб-сайту за протоколом HTTP;

2 – порушник пересилає запит до веб-сайту;

3 – веб-сайт відсилає веб-сторінку у відповідь за протоколом HTTP;

4 – порушник модифікує веб-сторінку, змінюючи посилання за протоколом HTTPS на посилання за протоколом HTTP;

5 – клієнт переходить по модифікованому посиланню і формує запит за протоколом HTTP;

6 – порушник перехоплює запит клієнта за протоколом HTTP, і надсилає запит з тим же змістом за протоколом HTTPS до веб-сайту;

7 – веб-сайт відсилає веб-сторінку у відповідь за протоколом HTTPS;

8 – порушник приймає веб-сторінку за протоколом HTTPS, модифікує її і надсилає клієнту від імені веб-сайта.

З метою попередження цієї атаки необхідно виключити можливість використання сторінок, доступних за протоколом HTTP для введення і передачі чутливих даних між клієнтом і веб-сайтом. Для попередження атак цього типу рекомендовано використовувати політику веб-сайтів «HSTS – HTTP Strict Transport Security» [14, 15] яка забезпечує:

- неможливість введення і передачі чутливих даних між клієнтом і веб-сайтом зі сторінок, доступних за протоколом HTTP шляхом автоматичного перенаправлення на сторінки, доступні лише за протоколом HTTPS;

- повідомлення клієнтів, що сайт використовує політику HSTS, щоб програмне забезпечення (веб-браузер) клієнта могло у майбутньому повідомити клієнта про спроби здійснення запитів певних сторінок за протоколом HTTP, що не відповідають політиці.

HSTS Bootstrap – вразливість, що базується на тому, що при першому переході на веб-сайт клієнт може використати посилання за протоколом HTTP замість HTTPS, або бути перенаправлений на веб-сайт із використанням такого посилання. При першому зверненні до веб-сайту клієнт ще не ознайомлений з політикою HSTS, що використовується сайтом, тому дозволяє з'єднання за протоколом HTTP без застережень, навіть якщо веб-сайт підтримує політику HSTS. У результаті перше посилання до веб-сайту буде зроблене з використанням незахищеного протоколу HTTP, що використовується порушником для перехоплення і модифікації даних.

Для попередження атак із використанням зазначеної вразливості використовується попередньо завантажений список веб-сайтів, які підтримують політику HSTS. Цей список – це

список із попередньо завантажених до веб-браузеру ланцюжків сертифікатів певних веб-сайтів. Попередньо завантажений список веб-сайтів, що підтримують політику HSTS, додається до веб-браузерів на етапі їх компіляції, що має виключити можливість неправомірної модифікації цього списку.

Таким чином навіть, якщо клієнт вперше відвідує веб-сайт, веб-браузер клієнта завчасно буде мати сертифікат відповідного веб-сайту, що дозволить відразу використовувати захищене з'єднання за протоколом HTTPS.

3.2.3. Атака повторного узгодження параметрів. Атака повторного узгодження параметрів [16] дозволяє порушнику включити свій запит перед запитом клієнта у момент звернення клієнта до веб-сайту. Порушник має можливість модифікувати запит клієнта до веб-сайту, але не має можливості переглядати інформацію відповіді веб-сайту клієнту (у загальному вигляді атаки). Схема атаки показана на рис. 3.

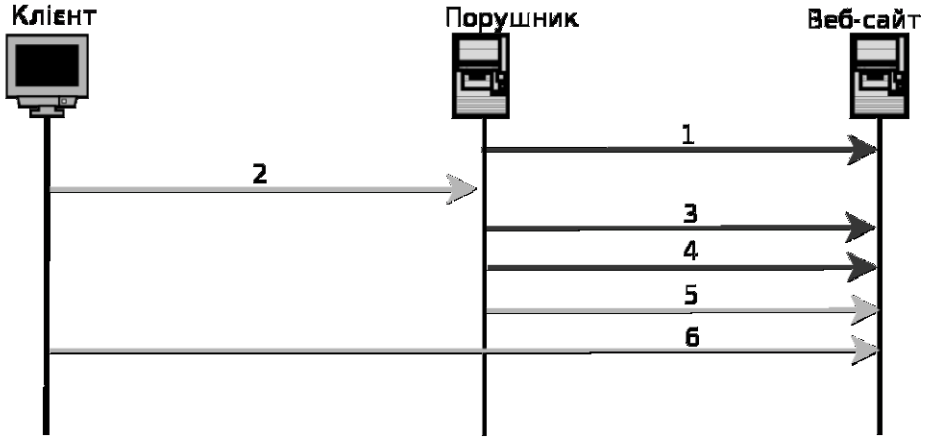

Рис. 3. Схема атаки повторного узгодження параметрів

Пояснення до рисунку:

1 – порушник ініціює з'єднання за протоколом SSL/TLS з веб-сайтом;

2 – клієнт надсилає запит на встановлення з'єднання за протоколом SSL/TLS з узгодженням криптографічних параметрів;

3 – порушник перехоплює і затримує запит клієнта і надсилає свій неповний запит HTTP, наприклад:

GET /pizza?toppings=pepperoni;address=attackersaddress

X-Ignore-This:

4 – порушник ініціює повторне узгодження параметрів;

5 – порушник пересилає до веб-сайту запит на встановлення з'єднання за протоколом SSL/TLS з узгодженням криптографічних параметрів, перехоплений на кроці 3;

6 – клієнт надсилає до веб-сайту свій запит, наприклад:

GET /pizza?toppings=sausage;address=victimssaddress

Cookie: victimscookie

Для веб-сайту два запити з'єднуються у один:

GET /pizza?toppings=pepperoni;address=attackersaddress

X-Ignore-This: GET /pizza?toppings=sausage;address=victimssaddress

Cookie: victimscookie

Таким чином, атака повторного узгодження дозволяє порушнику імперсоналізувати клієнта і нав'язувати веб-сайту запити від імені клієнта.

Атака може бути використана для нав'язування клієнту з'єднання за протоколом HTTP замість HTTPS. Для цього порушник на третьому кроці повинен сформувати запит, результатом якого буде перенаправлення на відкриту (незахищену) сторінку. Далі порушник реалізує атаку SSL Strip.

Для попередження атак типу «повторної» домовленності необхідно використовувати реалізації протоколу SSL/TLS з підтримкою поля Renegotiation Info, що несе в собі інформацію, створену із використанням криптографічних перетворень на основі певних даних про клієнта і веб-сайт, про останнє з'єднання між клієнтом і сервером [17]. Іншим засобом протидії є заборона повторного узгодження криптографічних параметрів за ініціативою клієнта. Як правило, у штатному режимі такої необхідності не виникає.

3.2.4. Атака пониження версій. За допомогою атаки пониження версій порушник має можливість нав'язати клієнту та веб-сайту найнижчу з підтримуємих версій SSL/TLS.

Сутність атаки полягає у тому, що порушник має можливість, знаходячись між клієнтом та веб-сайтом, змінювати деякі значення клієнтських «рукостискань» (client «handshake»), чим може запустити механізм зниження версії протоколу із рекомендованої до найнижчої із підтримуваних клієнтом та сервером.

Реалізація атаки можлива через необхідність забезпечення зворотної сумісності (підтримки попередніх версій) SSL/TLS, які мають вразливості що можуть бути використані порушником.

У загальному вигляді порушник реалізує атаку як підміну номеру версії протоколу, який має бути використаний, у «рукостисканні» клієнта або сервера.

Клієнтське рукостискання протоколу TLS 1.1 (код версії 0x0302) показане на рис. 4.

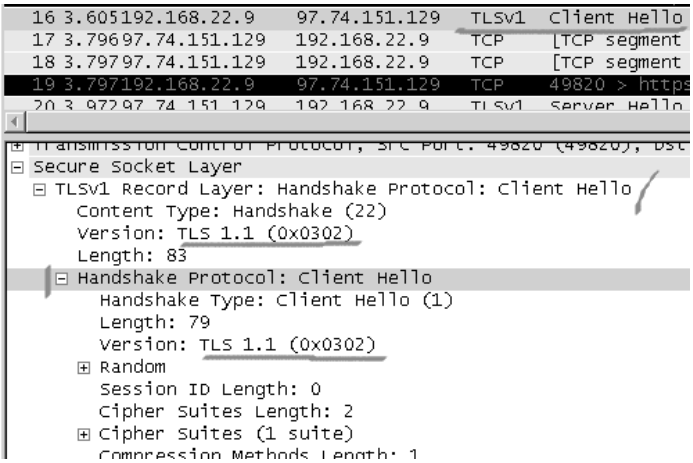

Рис. 4. Вміст повідомлення Client Hello

Відповідь сервера, який підтримує тільки TLS 1.0 показана на рис. 5.

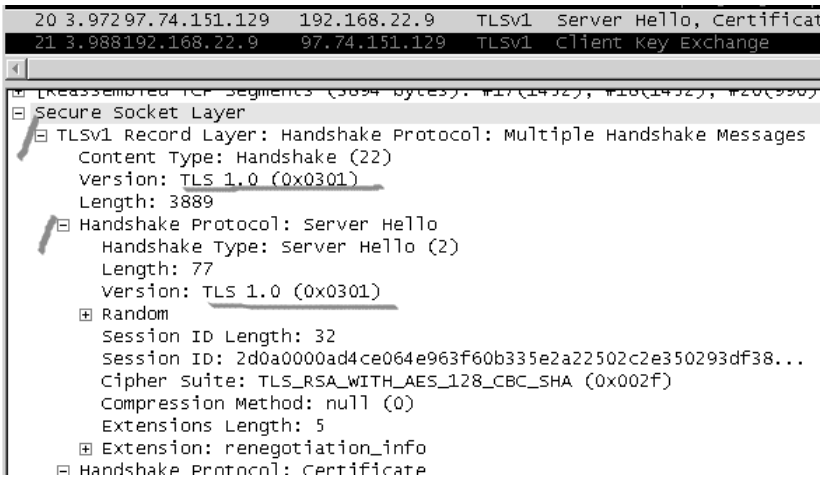

Рис. 5. Вміст повідомлення Server Hello

Відповідно до специфікації підтримки попередніх версій протоколу з'єднання має бути встановлене з використанням протоколу TLS версії 1.0. Порушник має можливість підробити запити клієнта/сервера для використання найбільш слабкого із протоколів, який підтримується і клієнтом і сервером.

Близькою до атаки пониження версій є атака нав'язування слабких криптографічних алгоритмів. У ході атаки порушник нав'язує клієнту та веб-сайту слабкі криптографічні алгоритми шляхом зміни поля Cipher Suite.

Результатом атаки є використання клієнтом та веб-сайтом слабких криптографічних алгоритмів, наприклад геш-функції MD5 або потокового шифру RC4 у експортному варіанті з ключем 40 біт.

Для попередження атак типу пониження версій рекомендовано обмежити використання тих версій протоколів, а також тих криптографічних алгоритмів, стійкість який визначена для системи як «слабка».

3.2.5. Атака BEAST. BEAST (Browser Exploit Against SSL/TLS) - це атака на SSL з'єднання (протокол https), що використовує передбачуваність значення вектора ініціалізації при шифруванні даних у режимі CBC.

Результатом атаки є порушення конфіденційності даних, що передаються.

Сутність атаки полягає у наступному. У протоколі TLS версії 1.0 і нижче у якості вектора ініціалізації (IV) для кожного нового зашифрованого повідомлення використовується значення останнього зашифрованого блока попереднього повідомлення. Отже порушнику відоме значення IV для наступного зашифрованого повідомлення до початку шифрування самого повідомлення.

У основі BEAST лежить атака з обраним відкритим тестом на шифрування у режимі CBC незалежно від блокового симетричного шифру, що використовується [18].

Нехай порушник спостерігав проходження наступного шифротексту від клієнта до веб-сайту: $C_1,...,C_l$. Порушник робить припущення, що блок відкритого тексту P_j значення має значення P^* . Для перевірки цього припущення порушник формує відкритий текст P', перший блок якого має значення $P'_1 = C_{j-1} \quad XOR \quad C_l \quad XOR \quad P^*$. Сформований таким чином відкритий текст нав'язується веб-браузеру клієнта для шифрування за формулою $C'_1 = E_k(IV \quad XOR \quad P'_1)$. Оскільки у якості IV використовується останній блок попереднього C_l , то при шифруванні P' веб-браузером клієнта буде використано $IV = C_l$.

У результаті

$$\begin{gathered} C'_1 = E_k(IV \quad XOR \quad P'_1) = \\ = E_k(C_l \quad XOR \quad P'_1) = \\ = E_k(C_l \quad XOR \quad C_{j-1} \quad XOR \quad C_l \quad XOR \quad P^*) = \\ = E_k(C_{j-1} \quad XOR \quad P^*) \end{gathered}$$

Нагадаємо, що $C_j = E_k(C_{j-1} \quad XOR \quad P_j)$. Отже, якщо $C_j = C'_1$, то $P^* = P_j$. Перебираючи значення P^* , порушник може встановити істинне значення P_j .

Практично атака була реалізована [19] з використанням зловмисного плагіна для веб-браузера - сформованої порушником бібліотеки, що завантажується у процес веб-браузера. Ціллю для атаки були обрані дані автентифікації (cookie), які передаються з кожним HTTP-запитом у складі відповідного заголовку. Схема взаємодії сторін при виконанні атаки наведена на рис. 6.

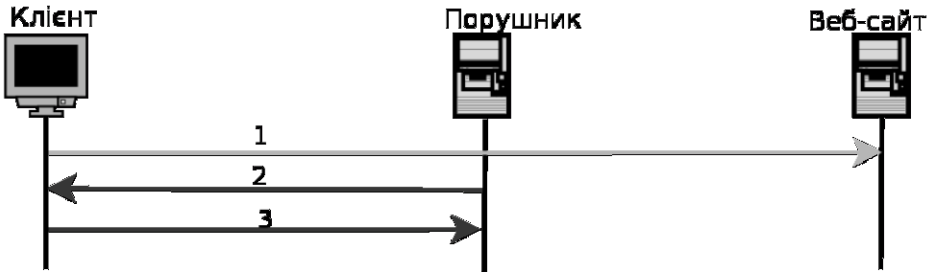

Рис.6. Схема атаки BEAST

Пояснення до рисунку:

1 – клієнт ініціює з'єднання за протоколом SSL/TLS з веб-сайтом і шифрує для нього деякі дані;

2 – порушник формує обраний відкритий текст і нав'язує його клієнту з використанням зловмисного плагіна у веб-браузері клієнта;

3 – порушник перехоплює і аналізує шифротекст сформований веб-браузером клієнта.

Етапи 2 і 3 повторюються до визначення правильного значення блока шифротексту порушником.

У протоколі TLS версії 1.1 і вище відповідна вразливість перекрита - для кожного наступного повідомлення формується непередбачуваний IV.

Для захисту від атаки BEAST необхідно використовувати протокол TLS версії 1.1 і вище.

3.2.6. Атаки CRIME та BREACH. BREACH (Browser Reconnaissance and Exfiltration via Adaptive Compression of Hypertext) – це атака на SSL-з'єднання (протокол HTTPS), що використовує побічні канали [20].

Атака «BREACH» використовує витік інформації про стиснуте і зашифроване повідомлення за рахунок зміни довжини повідомлення під час стискання на рівні HTTP [21]. CRIME використовує стискання на рівні TLS/SSL для тих же цілей [22].

Сутність атаки полягає у тому, що порушник змушує клієнта відправляти численні запити з незначними відмінностями, спостерігаючи за їх довжиною.

Схема атаки показана на рис. 7.

Рис. 7. Схема атак BREACH та CRIME

Пояснення до рисунку:

1 – клієнт ініціює з'єднання за протоколом SSL/TLS з веб-сайтом;

2 – порушник нав'язує клієнту веб-сторінку, що реалізує атаку Cross-site Request Forgery (CSRF)[23];

3 – веб-браузер клієнта надсилає від імені клієнта до веб-сайту численні запити, у складі яких передаються ідентифікатор сесії користувача і деякі змінні рядки, які контролюються порушником.

Аналізуючи довжину стиснутих і зашифрованих запитів, порушник може визначити наявність окремих символів. Якщо змінний рядок, який контролюється порушником, містить ту ж послідовність символів, що й ідентифікатор сесії користувача, то довжина стиснутого запиту

буде меншою, ніж у випадку, коли змінний рядок, який контролюється порушником, не має спільних послідовностей символів з ідентифікатором сесії клієнта.

Змінюючи рядки, порушник може визначити по черзі всі символи, наявні у ідентифікаторі клієнта. Замість ідентифікатора клієнта може бути атакований будь-який інший конфіденційний елемент, що передається у стисненому запиті.

Перед атаками BREACH та CRIME вразливі всі сайти , які використовують SSL/TLS шифрування з попередніми стисненням трафіку і при цьому дозволяють відправляти на сайт користувача запити довільного змісту (наприклад , пошукові запити).

Можливим способом захисту є відключення стиснення. У якості методу захисту на рівні web-додатків, пропонується випадковим чином змінювати представлення ідентифікаторів при кожному запиті, наприклад, через операцію XOR з випадковою маскою.

3.2.7. Атака на доповнення. Атака на доповнення – атака аналізу побічних каналів, яка використовує інформацію про реакцію веб-сайту на некоректно сформоване доповнення. Результатом атаки є порушення конфіденційності даних, що передаються між клієнтом і веб-сайтом.

Згідно зі специфікацією протоколу SSL/TLS отримане зашифроване повідомлення розшифровується. Після розшифрування веб-сайт перевіряє доповнення. Якщо доповнення коректне, то згідно з протоколом SSL/TLS веб-сайт обчислює MAC і перевіряє цілісність повідомлення. Якщо доповнення не коректне, то веб-сайту невідома справжня довжина повідомлення, тому веб-сайт обчислює MAC від тих блоків повідомлення, які гарантовано не належать до доповнення. Таким чином, час обчислення MAC у випадку коректного і некоректного доповнення відрізняється, що і дає порушнику інформацію про коректність доповнення.

Нехай C^* – блок шифротексту, відкритий текст P^* для якого бажає отримати. Нехай C' – попередній блок шифротексту перед C^*. Зв'язок між ними описується як $P^* = D_K\left(C^*\right)$ XOR C'. Для кожного блоку B (і відкритого тексту, і шифротексту) запишемо у вигляді $B = \left[B_{1,}...B_{b-1}\right]$, де b - кількість байтів у блоці, B_i – i-й байт у блоці, i – номер байта у блоці.

Нехай легітимний клієнт надіслав до веб-сайту криптограму $C_{client} = C_1 \,||\, C_2 \,||\, C' \,||\, C$. Порушник формує множину підробних криптограм $C_{att}\left(\Delta\right) = C_1 \,||\, C_2 \,||\, C' \quad XOR \quad \Delta \,||\, C$

Результатом розшифрування є відкритий текст $P = P_1 \,||\, P_2 \,||\, P_3 \,||\, P_4$, $P_4 = D_K\left(C^*\right) \; XOR \; \left(C' \; XOR \; \Delta\right) = P^* \; XOR \; \Delta$. Підбираючи значення Δ по байтам порушник модифікує останній блок. За часом відповіді порушник визначає чи успішно сформовано доповнення.

За специфікацією SSL/TLS доповнення формується як послідовність з $n+1$ байтів зі значенням n. Спочатку порушник підбирає останні 2 байти блоку Δ, щоб отримати доповнення $0x01 \,||\, 0x01$. На наступному кроці порушник підбирає третій з кінця байт блоку Δ, щоб отримати доповнення $0x02 \,||\, 0x02 \,||\, 0x02$ і так далі. Складність атаки оцінюється як 2^{23} сесій на 16-байтний блок. Детально атака і практичні результати описані у роботі [24]

Згідно з цією роботою для попередження атаки слід використовувати алгоритм AES-CGM або використовувати останні версії бібліотек, у яких внесені відповідні зміни.

3.2.8. Атаки, пов'язані з використанням RC4. Атаки на протокол SSL/TLS, пов'язані з використанням RC4, викликані тим, що RC4 має відомі відхилення від рівноімовірного розподілу імовірності появи певних байтів на початку шифруючої гами [25]. RC4 рекомендовано для використання з метою захисту від деяких інших атак на протокол SSL/TLS [26], однак цю рекомендацію не слід приймати: атаки з використанням побічних каналів потребують генерації порівняно великої кількості повідомлень, що потенційно може бути виявлено, тоді як атаки на RC4 потребують лише пасивного спостереження [25].

3.2.9. Атака обрізання. Атака обрізання використовує неправильну обробку подій завершення сеансу між клієнтом та веб-сайтом [27]. Сутність атаки полягає у тому, що порушник затримує і видаляє з потоку повідомлень штатний запит на завершення сеансу клієнта або надсилає пакет TCP FIN, який примусово завершує з'єднання. .Схема атаки показана на рис.8.

Рис. 8. Схема Truncation attack

Пояснення до рисунку:

1 – клієнт ініціює з'єднання за протоколом SSL/TLS з веб-сайтом;

2 – порушник надсилає TCP-пакет з встановленим флагом FIN, чим завершує сесію SSL/TLS, клієнт бачить у своєму веб-браузері коректне завершення власної сесії, однак у веб-браузері зберігаються дані автентифікації клієнта;

3 – порушник використовуючи фізичний доступ до веб-браузера клієнта порушник відновлює сесію клієнта і отримує доступ до веб-сайту з використанням даних автентифікації клієнта.

Авторами [27] у першій спробі атака була реалізована проти сервісів електронної пошти Gmail, Hotmail та ін. за модифікованою схемою. Сутність модифікованої схеми полягає у тому, що замість надсилання TCP-пакета з встановленим флагом FIN, порушник вилучає з потоку IP-пакетів пакети, які містять запит до веб-сайту на завершення сесії клієнта, але пропускає усі інші. Такі пакети мають визначений розмір.

У випадку Hotmail вилучалися пакети за правилом

iptables -A OUTPUT -m length --length 474:506 -j DROP

У випадку Gmail вилучалися пакети за правилом

iptables -A OUTPUT -m length --length 1165:1195 -p tcp -j REJECT --reject-with tcp-reset

Схема такої атаки показана на рисунку 9

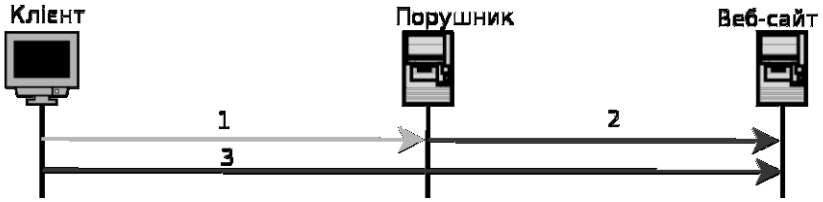

Рис. 9. Модифікована схема Truncation attack

Пояснення до рисунку:

1 – клієнт ініціює з'єднання за протоколом SSL/TLS з веб-сайтом;

2 – порушник фільтрує IP-пакети за розміром, не пропускаючи IP-пакет, що за розміром відповідає запиту на завершення сесії клієнта, клієнт у результаті спроби завершення сесії бачить у своєму веб-браузері коректне завершення власної сесії, однак у веб-браузері зберігаються дані автентифікації клієнта;

3 – порушник використовуючи фізичний доступ до веб-браузера клієнта порушник відновлює сесію клієнта і отримує доступ до веб-сайту з використанням даних аутентифікації клієнта.

Результатом атаки є те, що клієнт у веб-браузері бачить, що сесія завершена. У той же час, сесія насправді залишається чинною, дані автентифікації клієнта (ідентифікатор сесії HTTP)

залишаються у веб-браузері клієнта. Якщо порушник у цей час отримає фізичний доступ до веб-браузера клієнта, то він зможе отримати доступ до даних клієнта.

Для протидії цій атаці у TLS 1.2 впроваджено новий тип повідомлення close_notify, яким веб-сайт і клієнт повідомляють один одного про завершення сесії [12]. Веб-сайти мають коректно обробляти події і помилки завершення сесії і демонструвати сторінку з повідомленням про завершення сесії тільки після дійсного завершення сесії.

3.2.10. Неповна (неправильна) перевірка сертифіката. Атаки неповної перевірки сертифіката є атаками на імплементації протоколів SSL/TLS. Неповна (неправильна) перевірка сертифікатів може бути використана порушником для нав'язування клієнту неправильного сертифіката веб-сайту.

Неповна (неправильна) перевірка сертифікатів може бути реалізована завдяки:

- використанню геш-функції MD5 у сертифікаті [28];
- неправильній обробці ознаки можливості випуску дочірніх сертифікатів [XL29];
- неправильному розбору рядків у кодуванні ASN.1 [30].

Використання клієнтом слабких криптографічних перетворень, зокрема геш-функції MD5, може дозволити порушнику нав'язати клієнту підробний сертифікат веб-сайту. Атака можлива навіть якщо ЦСК вже не випускає сертифікати з використанням MD5 [31].

Неправильна обробка ознака того, що власник сертифіката є ЦСК чи кінцевим користувачем, дозволяє порушнику, який має легітимний сертифікат кінцевого користувача, виданий ЦСК, видати підробний сертифікат для веб-сайту, підписаний особистим ключем порушника. Якщо веб-браузер клієнта не перевіряє, чи підписаний сертифікат веб-сайту ЦСК чи кінцевим користувачем, то перевірка ланцюжка сертифікатів буде виконана успішно. Користуючись цим, порушник може отримати у ЦСК легітимний сертифікат, випустити сертифікат для веб-сайта, підписаний особистим ключем порушника, вказавши там усі облікові дані веб-сайта, і, при зверненні клієнта до веб-сайта, нав'язати клієнту випущений порушником сертифікат, якому відповідає особистий ключ, який веб-сайту не належить. Результатом стане атака, аналогічна SSLStrip, з тією відмінністю, що взаємодія між порушником і клієнтом також буде здійснюватися за протоколом HTTPS замість HTTP, отже клієнт жодним чином не зможе виявити підміну.

У кодуванні ASN.1 рядок представляється структурою, яка містить довжину рядка *n* і *n* символів (представлення Pascal). Таке представлення допускає наявність символу з ASCII кодом 0 у рядку. Однак програмне забезпечення, написане мовою C/C++, інтерпретує символ з ASCII кодом 0 як кінець рядка. Отже рядки example.com і example.com\0.ua будуть інтерпретовані як example.com. Порушник може сформувати і нав'язати клієнту підробний сертифікат, виданий на доменне ім'я example.com\0.ua. Якщо декодування рядків у веб-браузері клієнта з представлення ASN.1 не враховує вказаної особливості, то при перевірці такого підробного сертифіката для веб-сайту example.com сертифікат буде визнано таким, що відповідає доменному імені.

Щоб уникнути атак, пов'язаних з неповною (неправильною) перевіркою сертифіката, веб-браузер клієнта повинен повністю реалізовувати логіку перевірки сертифікатів. Будь-які відхилення від стандартів можуть стати причиною появи вразливостей.

3.2.11. Використання незахищеного контенту на захищеній сторінці. Використання незахищених елементів на захищеній веб-сторінці може бути використане для крадіжки даних, що вводяться клієнтом на захищеній сторінці, або для нав'язування вмісту веб-сторінки клієнту.

Веб-сторінка може містити посилання на такі елементи як програми мовою Javascript, списки стилів, інші веб-сторінки, зображення, які відображаються як частина захищеної веб-сторінки.

Порушник може перехоплювати і модифікувати елементи веб-сторінки, що передаються по незахищеному з'єднанню, впроваджувати у них власні функції, що реалізуються програмами мовою Javascript. Впроваджені порушником функції можуть бути використані для модифікації

даних на захищеній сторінці або крадіжки чутливих даних, що вводяться на захищеній веб-сторінці клієнтом.

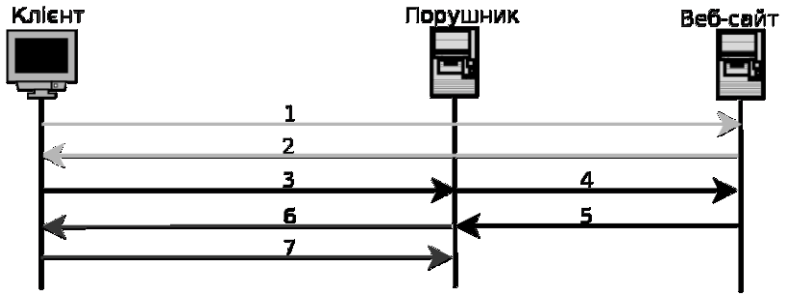

Рис. 10. Схема атаки з використанням незахищеного контенту на захищеній сторінці

Пояснення до рисунку:

1 – клієнт надсилає запит до веб-сайту за протоколом HTTPS;

2 – веб-сайт надсилає клієнту захищену веб-сторінку за протоколом HTTPS;

3 – веб-браузер клієнта аналізує сторінку, знаходить посилання на її елементи і надсилає до веб-сайту за протоколом HTTP;

4 – порушник перехоплює запит клієнта за протоколом HTTP і пересилає його до веб-сайту;

5 – веб-сайт надсилає у відповідь на запит елементи захищеної веб-сторінки за протоколом HTTP;

6 – порушник перехоплює елементи захищеної веб-сторінки і додає до них програми мовою Javascript, що модифікують вміст захищеної сторінки, змінюючи її вигляд;

7 – програми мовою Javascript, додані до захищеної веб-сторінки порушником, надсилають порушнику чутливі дані, що вводяться клієнтом.

Таким чином, щоб уникнути атаки з використанням незахищеного контенту на захищеній сторінці, усі елементи, що входять до складу захищеної веб-сторінки, повинні передаватись клієнту лише по захищеному з'єднанню, яке забезпечується протоколом HTTPS.

Висновки

За результатами проведеного аналізу зроблено наступні висновки.

1) Доцільно гармонізувати стандарт ISO/IEC 9594-8 у редакції ISO/IEC FDIS 9594-8 [6] одразу після введення його в дію на міжнародному рівні.

2) Доцільно провести окреме дослідження щодо шляхів забезпечення сумісності криптографічних перетворень при транскордонній взаємодії.

3) З метою забезпечення безперешкодної транскордонної взаємодії необхідно розробити механізми довіреного розповсюдження кореневого сертифіката Центрального засвідчувального органу України, зокрема у складі операційних систем, веб-браузерів.

4) З метою попередження атаки SSL Strip рекомендовано використовувати політику HSTS [15], яка забезпечує неможливість взаємодії клієнта і веб-сайту по незахищеному з'єднанню.

5) Для попередження атак повторного узгодження параметрів необхідно використовувати реалізації протоколу SSL/TLS з підтримкою поля Renegotiation Info, що несе в собі інформацію, створену із використанням криптографічних перетворень на основі певних даних про клієнта і веб-сайт, про останнє з'єднання між клієнтом і сервером. Іншим засобом протидії є заборона повторного узгодження криптографічних параметрів за ініціативою клієнта.

6) Для захисту від атаки пониження версії та атак нав'язування слабких криптографічних алгоритмів необхідно заборонити використання слабких криптографічних алгоритмів та попередніх версій протоколу SSL/TLS, які мають відомі вразливості.

7) Для захисту від атаки BEAST необхідно використовувати протокол TLS версії 1.1 і вище.

8) Для захисту від атак BREACH та CRIME необхідно заборонити використання стиснення даних. У якості методу захисту на рівні web-додатків, пропонується випадковим чином змінювати представлення ідентифікаторів при кожному запиті, наприклад, через операцію XOR з випадковою маскою.

9) Для попередження атаки на доповнення слід використовувати алгоритм AES-CGM або використовувати останні версії бібліотек, у яких внесені відповідні зміни.

10) Використання шифру RC4 є поганою практикою, у зв'язку з наявністю нерівно імовірного розподілу байтів на окремих позиціях у шифруючій гамі. Використання RC4 для захисту від padding attack є невдалою рекомендацією, оскільки створює більшу вразливість, ніж перекриває.

11) Причина атаки обрізання полягає у неправильній обробці веб-застосуваннями подій SSL/TLS-сесії. Для протидії цій атаці необхідно використовувати протокол TLS версії не нижче 1.2. Веб-сайти мають коректно обробляти події і помилки завершення сесії і демонструвати сторінку з повідомленням про завершення сесії тільки після дійсного завершення сесії.

12) Щоб уникнути атак, пов'язаних з неповною (неправильною) перевіркою сертифіката, веб-браузер клієнта повинен повністю реалізовувати логіку перевірки сертифікатів. Будь-які відхилення від стандартів можуть стати причиною появи вразливостей.

13) Щоб уникнути атаки з використанням незахищеного контенту на захищеній сторінці, усі елементи, що входять до складу захищеної веб-сторінки, повинні передаватись клієнту лише по захищеному з'єднанню, яке забезпечується протоколом HTTPS.

Список літератури: 1. *Brussels*, XXX. COM(2012) 238/2. Proposal for a REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on electronic identification and trust services for electronic transactions in the internal market (Text with EEA relevance) {SWD(2012) 135} {SWD(2012) 136}. 2. *Інформаційні* технології. Взаємозв'язок відкритих систем. Каталог. Частина 8. Основні положення щодо сертифікації відкритих ключів та атрибутів (ISO/IEC 9594-8:2001, IDT): ДСТУ ISO/IEC 9594-8:2006 – [Чинний від 2008-01-01]. – К. : Держспоживстандарт України, 2006. 3. *Міністерство* юстиції України. Адміністрація державної служби спеціального зв'язку та захисту інформації України. Наказ від 20.08.2012 № 1236/5/453 Про затвердження вимог до форматів, структури та протоколів, що реалізуються у надійних засобах електронного цифрового підпису: за станом на 1 груд. 2013 р.: Режим доступу: http://zakon1.rada.gov.ua/laws/show/z1398-12 4. *Веб-сторінка*, присвячена стандарту ISO/IEC 9594-8:2005 : за станом на 1 груд. 2013 р. : / International Organization for Standardization/ International Electrotechnical Commission. – Режим доступу: http://www.iso.org/iso/iso_ catalogue/catalogue_tc/catalogue_detail.htm?csnumber=43793. 5. *Веб-сторінка*, присвячена стандарту ISO/IEC 9594-8:2008 : за станом на 1 груд. 2013 р. / International Organization for Standardization/ International Electrotechnical Commission Режим доступу: http://www.iso.org/iso/home /store/catalogue_ics/catalogue_detail_ics.htm?csnumber=53372. 6. *Веб-сторінка*, присвячена проекту стандарта ISO/IEC FDIS 9594-8 : за станом на 1 груд. 2013 р. / International Organization for Standardization/ International Electrotechnical Commission. – Режим доступу: http://www.iso.org/iso/home/ store/catalogue_tc/catalogue_detail.htm?csnumber=64854. 7. *RFC* 4491 Using the GOST R 34.10-94, GOST R 34.10-2001, and GOST R 34.11-94 Algorithms with the Internet X.509 Public Key Infrastructure Certificate and CRL Profile. – Режим доступу: http://tools.ietf.org/html/rfc4491 8. *RFC* 4357 Additional Cryptographic Algorithms for Use with GOST 28147-89, GOST R 34.10-94, GOST R 34.10-2001, and GOST R 34.11-94 Algorithms. – Режим доступу: http://tools.ietf.org/html/rfc4357 9. RFC 6101 The Secure Sockets Layer (SSL) Protocol Version 3.0. – Режим доступу: http://tools.ietf.org/html/rfc6101 10. *RFC* 2246 The TLS Protocol Version 1.0. – Режим доступу: http://tools.ietf.org/html/rfc2246 11. RFC 4346 The Transport Layer Security (TLS) Protocol Version 1.1.- Режим доступу: http://tools.ietf.org/html/rfc4346 12. *RFC* 5246 The Transport Layer Security (TLS) Protocol Version 1. – Режим доступу: http://tools.ietf.org/html/rfc5246 13. Веб-сторінка SSL Strip . – Режим доступу: http://www.thoughtcrime.org/software/sslstrip/ 14. *Transport* Layer Protection Cheat Sheet . – Режим дос-

тупу: ://owasp.org/index.php/Transport_Layer_Protection_Cheat_Sheet 15. RFC 6797 HTTP Strict Transport Security (HSTS). – Режим доступу: http://tools.ietf.org/html/rfc6797 16. *Vulnerability* Note VU#120541. SSL and TLS protocols renegotiation vulnerability. – Режим доступу: http://www.kb.cert.org/vuls/id/120541 17. *RFC* 5746 Transport Layer Security (TLS) Renegotiation Indication Extension. – Режим доступу: http://tools.ietf.org/html/rfc5746 18. *Gregory V.* Bard. – Режим доступу: Vulnerability of SSL to Chosen-Plaintext Attack http://eprint.iacr.org/2004/111.pdf 19. *A. Doty, M. Jablonski, T. Bademian*. Cracking SSL/TLS Using BEAST. – Режим доступу: http://mason.gmu.edu/~msherif/isa564/fall11/projects/beast.pdf 20. *J. Kelsey*. Compression and Information Leakage of Plaintext. – Режим доступу:http://www.iacr.org/cryptodb/archive/2002/FSE/3091/3091.pdf 21. A. Prado, N. Harris, Y. Gluck BREACH: reviving the CRIME attack. – Режим доступу: http://breachattack.com/resources/BREACH%20-%20SSL,%20gone%20in%2030%20seconds.pdf 22. *T. Be'ery, A. Shulman* A Perfect CRIME? Only TIME Will Tell. – Режим доступу: https://media.blackhat.com/eu-13/briefings/Beery/bh-eu-13-a-perfect-crime-beery-wp.pdf 23. *Сторінка* проекту OWASP, присвячена вразливості CSRF. – Режим доступу: https://www.owasp.org/index.php/Cross-Site_Request_Forgery_%28CSRF%29 24. *N. J. AlFardan, K. G. Paterson* Lucky Thirteen: Breaking the TLS and DTLS Record Protocols. – Режим доступу:http://www.isg.rhul.ac.uk/tls/TLStiming.pdf 25. *S. Sarkar, S. Gupta, G. Paul, S.* Maitra Proving TLS-attack related open biases of RC4. – Режим доступу:https://eprint.iacr.org/2013/502.pdf. 26. *S. Vaudenay* CBC Padding: Security Flaws in SSL, IPSEC, WTLS. – Режим доступу: http://infoscience.epfl.ch/record/52417/files/IC_TECH_REPORT_200150.pdf 27. *B. Smyth, A. Pironti.* Truncating TLS Connections to Violate Beliefs in Web Applications . – Режим доступу: https://media.blackhat.com/us-13/US-13-Smyth-Truncating-TLS-Connections-to-Violate-Beliefs-in-Web-Applications-WP.pdf 28. *A. Lenstra, X. Wang, B. de Weger* Colliding X.509 Certificates[Електронний ресурс]: — Режим доступу: https://eprint.iacr.org/2005/067.pdf 29. *M. Rosenfeld*, Internet Explorer SSL Vulnerability. – Режим доступу: http://www.thoughtcrime.org/ie-ssl-chain.txt 30. *M. Marlinspike*. Null Prefix Attacks Against SSL/TLS Certificates. – Режим доступу: http://www.thoughtcrime.org/papers/null-prefix-attacks.pdf 31. *C. Meyer, J. Schwenk.* Lessons Learned From Previous SSL/TLS Attacks A Brief Chronology Of Attacks And Weaknesses. – Режим доступу: http://eprint.iacr.org/2013/049.pdf

Харківський національний
університет радіоелектроніки

Надійшла до редколегії 03.02.2014