УДК 004.62

К. Е. Петров,
І. В. Кобзев,
Ю. М. Онищенко

Політика безпеки Web-застосувань та серверів

Розглянуто питання вразливості використання Web-застосувань та серверів і яким чином можуть бути використані найбільш загальні способи захисту для запобігання програмним атакам на сервери та застосування. Функціонування Web-сервера неможливе, якщо не приділяти належну увагу питанням забезпечення його інформаційної безпеки. Ця проблема може бути вирішена шляхом використання комплексного підходу до захисту ресурсів сервера від можливих атак.

Ключові слова: Web-застосування, сервер, уразливість, програмне забезпечення, безпека, захист, атака.

Постановка проблеми. На сьогодні вразливості у Web-застосуваннях, як і раніше, залишаються одним з найбільш поширених недоліків забезпечення захисту інформації. Попри те, що уразливості Web-застосувань неодноразово описані в науково-популярній і спеціалізованій літературі, досить рідко зустрічається опис превентивних захисних механізмів, що знижують ризики використання цих вразливостей.

Недооцінка серйозності ризику реалізації загроз інформаційній безпеці з використанням Web-застосувань, доступних з боку мережі Internet, можливо, є одним з основних чинників низького рівня захищеності більшості з них. За даними статистики, у 2013 році на 63 % сайтів були виявлені критичні вразливості, за допомогою яких можна здійснити успішну атаку на Web-застосування або сервер. Найбільша кількість Web-застосувань, що містять вразливості високої міри ризику, була виявлена серед Web-застосувань, що належать засобам масової інформації, де у 80 % випадків рівень уразливості виявився критичним [1].

Протягом кількох останніх десятиліть головні постачальники комп'ютерної і програмної інфраструктури, включаючи IBM, Microsoft, Sun Microsystems, наполегливо працюють, удосконалюючи свою технологічну базу, щоб підтримати розвиток, розгортання, обслуговування і безпеку Web-серверів. Питання безпеки Web-серверів детально розглядаються в публікаціях [2–4]. Аналіз наукової літератури засвідчив, що при всій значущості питання захисту Web-серверів цьому напрямку в нашій країні приділяється недостатньо уваги та на сьогодні не існує чітко окресленої системи заходів протидії зловживанням в інформаційній сфері.

Більшість організацій, малих або великих, що приділяють багато уваги оформленню своїх сайтів, іноді нехтують основними принципами їх безпечного існування. Нещодавні атаки на Web-сайти показали,

що працездатність серверів може бути порушена навіть в результаті перевантаження одного або кількох сервісів. У цій статті ми ставимо за **мету** обговорити найбільш загальні способи захисту, які організації можуть взяти на озброєння для запобігання атакам на їх сервери або пом'якшення їх наслідків.

Виклад основного матеріалу. Для правильної організації захисту Web-сервера і написання безпечно працюючого Web-застосування необхідно розуміти, як виникають вразливості і яким чином вони можуть бути використані.

Одними з найбільш поширених і небезпечних Web-уразливостей, за версією OWASP, на даний момент є такі:

1) *cross-site scripting* (впровадження сценаріїв, XSS) – дозволяє тому, хто атакує, перехоплювати конфіденційну інформацію і виконувати шкідливий код у контексті Web-браузера;

2) *SQL injection* (упровадження SQL) – механізми атаки на Web-застосування, які використовують введені користувачем дані в SQL-запитах до бази даних без попередньої обробки, необхідної для видалення потенційно небезпечних символів і зарезервованих слів;

3) *cross-site request forgery* – один із способів використання вразливостей cross-site scripting, який полягає в підробці HTTP-запитів; з його допомогою той, хто атакує, дістає можливості виконувати запити від імені законного користувача;

4) *command injection* (впровадження команд) – виникає у тому випадку, коли дані, що передаються серверу в HTTP-запиті, використовуються для формування команд операційної системи без перевірки на наявність спеціальних символів [5].

Звичайно, найкращий захист – це відсутність вразливостей, тому ще на етапі первинного налаштування політики безпеки Web-сервера або написання Web-застосування необхідно враховувати, які дії може здійснити зловмисник для злому, і заздалегідь вживати заходи. Адже тільки сам розробник має точно знати, які дані повинні поступати і як ці дані належить обробляти, а які дані на сервер потрапити ні в якому разі не повинні. Тому перед розробкою Web-застосування рекомендується розробити модель порушника і модель загроз. Проте у будь-якому випадку, коли додаток вже написано, необхідно захищати його і Web-сервер, на якому додаток знаходиться.

Захист Web-сервера – це комплекс організаційних і технічних заходів. Тому для правильної організації захисту Web-сервера в першу чергу необхідно з'ясувати, які вразливості є присутніми у Web-застосуваннях і політиці безпеки Web-сервера. Проведення аудиту дозволяє виявити помилки й оцінити захищеність Web-застосування, а також отримати набір рекомендацій щодо посилення захисту.

Найбільш поширеними підходами до технологічного аудиту є тест на проникнення (показова демонстрація дій порушника) і «традиційний» аудит (аналіз параметрів конфігурації Web-сервера і застосування

сканера вразливостей). Останнім часом виділяється ще один тип аудиту – активний, який поєднує в собі можливості обох підходів і усуває деякі їх недоліки.

Мінімально необхідними є певні вимоги до методики аудиту, зокрема:

1) об'єктивність (достовірність) результату. Аудиторові необхідно подати докази того, що саме ці вразливості існують в інформаційній системі, і детально описати можливі наслідки їх реалізації порушником;

2) повнота аудиту. Необхідно довести, що в ході аудиту не були проігноровані які-небудь уразливості Web-застосування або сервера.

Аудит повинен проводитися регулярно, оскільки постійно виявляються нові вразливості операційних систем, на яких розміщуються Web-сервери; знаходяться нові методи атак на Web-застосування; допрацьовуються сканери вразливостей; відбуваються збої в політиці безпеки. Після закінчення аудиту надаються рекомендації щодо модернізації системи мережевого захисту, які дозволяють усунути небезпечні вразливості і таким чином підвищити рівень захищеності інформаційної системи від дій зовнішнього порушника. Природно, що найкращим методом боротьби з уразливостями є виправлення коду Web-застосування і грамотне налаштування політики безпеки. Однак зустрічаються ситуації, коли змінити само Web-застосування немає можливості або відсутні права на налаштування політики безпеки на Web-сервері. Тобто одними організаційними заходами проблему безпеки розв'язати неможливо. У таких випадках доводиться застосовувати технічні заходи для захисту, тобто встановлювати сторонні засоби для забезпечення захисту інформаційної системи.

Одним із засобів усунення вразливостей і одночасно профілактичним заходом є встановлення Web Application Firewall (WAF), що є програмним або апаратним міжмережевим екраном для Web-застосувань. Такого роду міжмережевий екран установлюється перед Web-застосуванням і надає ширші можливості, ніж звичайні програмні міжмережеві екрани, для системи. WAF у змозі контролювати всі об'єкти, які можуть бути доступні користувачам, – URL, що вводяться, а також параметри запитів GET і POST. Також міжмережевий екран не дозволяє користувачам запускати об'єкти, що не належать до Web-ресурсу [6].

Однак, як показала практика, часто WAF не справляється з поставленими завданнями. Причиною тому є два чинники: фундаментальні обмеження технології – нездатність повністю захистити Web-застосування від усіх можливих вразливостей, а також уразливості реалізації конкретного міжмережевого екрана. Приміром, опубліковано кілька помилок mod_security, що дозволяють обійти захист; виявлено варіанти рядків, що дозволяють реалізувати атаку SQL injection на Web-застосування зі встановленим phpids; уразливість в SecureSphere дозволяє видаленому користувачеві зробити XSS-напад.

Тому встановлення міжмережевих екранів не вирішує проблему повністю [7].

Можна виділити три рівні безпеки для сервера.

Рівень 1. Мінімальний рівень безпеки: модернізація наявного програмного забезпечення; єдині налаштування (політика) для усіх серверів.

Рівень 2. Опір вторгненню. Встановлення зовнішнього міжмережевого екрана: видалене адміністрування систем безпеки; обмеження на використання скриптів; захист Web-серверів з використанням фільтрації пакетів; навчання персоналу і розмежування прав доступу.

Рівень 3. Виявлення атак і послаблення їх дії: розподіл привілеїв; апаратні системи захисту; внутрішній міжмережевий екран; мережеві системи виявлення вторгнень; системи виявлення вторгнень, що розміщуються на серверах (хостах).

Можна виділити такі найбільш загальні способи захисту Web-серверів: видалення зайвого програмного забезпечення (додатків); виявлення спроб порушення захисту Web-серверів; виправлення вад у встановленому програмному забезпеченні; зменшення наслідків атак на мережу; захист іншої частини мережі у випадку, якщо Web-сервер був скомпрометований.

Вимога оновлення програмного забезпечення викликана тим, що будь-яке програмне забезпечення, встановлене на Web-сервері, може бути використане злочинцем для проникнення в систему. Це і операційні системи, і програмне забезпечення, що працює з мережевими пакетами, або те, що використовується адміністраторами мережі і системи безпеки.

Забезпечення безпеки інформації вимагає виділення окремого ресурсу (комп'ютера) під кожне завдання. Інакше помилка в системі безпеки може порушити роботу відразу кількох сервісів. Наприклад, не бажано розміщувати сервер електронної пошти, Web-сервер і сервер баз даних на одному й тому ж комп'ютері. Проте кожен новий сервер має бути оснащений системою захисту, інакше він може стати легкою мішенню для зловмисника.

Усе привілейоване програмне забезпечення, що не є обов'язковим для Web-сервера, має бути видалене. Під привілейованим програмним забезпеченням у цьому випадку слід розуміти таке, що працює з мережевими пакетами, або таке, що запускається з правами адміністратора. Деякі операційні системи запускають привілейовані програми за умовчанням, а адміністратори часто просто не знають про їх існування. Між тим, кожна така програма може бути використана хакером для атаки на Web-сервер. У ряді випадків для підвищення рівня безпеки адміністратори видаляють усе програмне забезпечення (а не тільки привілейоване), яке не використовується для забезпечення працездатності Web-сервера.

Установка міжмережевого екрана між корпоративною (внутрішньою) мережею і Web-серверами загального доступу дозволяє запобігти проникненню сторонніх пакетів у мережу організації: якщо зловмисник проникає на зовнішній Web-сервер, то потрапити в корпоративну мережу організації через firewall йому буде складно. Якщо ж Web-сервер знаходиться усередині корпоративної мережі, то хакер, проникнувши на нього, може, використовуючи захоплений ресурс як плацдарм, порушити працездатність усієї мережі й отримати повний контроль над нею.

Більшість сайтів містять скрипти, які запускаються при переході на особливу сторінку. Зловмисник може використовувати ці скрипти (за допомогою виявлених вад у коді) для проникнення на сайт. Для виявлення таких «дірок» йому зовсім не обов'язково знати початковий код, тому скрипти необхідно ретельно перевірити, перш ніж вони будуть викладені на сайт. Скрипти не повинні запускати випадкові команди або сторонні (небезпечні) програми, дозволяти користувачам виконання певних вузькоспеціалізованих завдань, а також обмежувати кількість параметрів потоку, що входить. Останнє потрібно для запобігання атакам на переповнювання буфера. При атаках такого роду зловмисник намагається змусити систему до запуску програми арбітражу з метою отримання додаткової інформації. Нарешті, скрипти не повинні мати прав адміністратора.

Маршрутизатори встановлюють для того, щоб відокремити Web-сервери від іншої частини мережі. Цей захід допомагає запобігти багатьом атакам, не допускаючи проникнення «чужих» (неправильних) пакетів. Зазвичай маршрутизатори видаляють усі пакети, які не йдуть на Web-сервер (наприклад на порт 80) або до портів, що використовуються при видаленому адмініструванні. Для підвищення рівня безпеки можна скласти перелік пакетів, що підлягають пропуску. Таким чином, хакеру залишається ще менше можливостей для проникнення в мережу. Маршрутизатор з функцією фільтрації пакетів буде ефективніший для запобігання атакам за умови видалення з сервера усього непотрібного програмного забезпечення (зловмисник не зможе запросити нестандартний сервіс). Проте слід мати на увазі, що застосування пакетної фільтрації знижує пропускну спроможність маршрутизатора і збільшує ризик втрати «правильних» пакетів.

Незалежно від серйозності заходів, спрямованих на забезпечення безпеки Web-сервера, вірогідність проникнення повністю виключити неможливо. Тому, якщо це все ж таки сталося, важливо мінімізувати наслідки атаки. Розподіл привілеїв являє собою ефективний спосіб для досягнення цієї мети: кожен користувач може запускати тільки певні програми. Тому хакер, що проник у мережу за скомпрометованими даними окремого користувача, зможе завдати системі лише обмеженої шкоди. Наприклад, у користувача на сайті є свої

сторінки, але інші сторінки йому не доступні. Отже, хакер, добувши дані цього користувача, не зможе вплинути на інші ресурси (сторінки). Так само буде і з програмним забезпеченням. З метою підвищення рівня безпеки для користувачів, які мають права запису, можна створити особисті піддиректорії.

Апаратура, в плані розподілу привілеїв, має вищий рівень безпеки, оскільки на відміну від програмного забезпечення не так легко модифікується. Але через «діри» в програмному забезпеченні хакер може отримати доступ і до апаратних засобів. Одним з найдоступніших способів захисту від цієї загрози є заборона режиму запису на зовнішні жорсткі диски, магнітооптичні диски тощо. Зазвичай для запобігання атакам Web-сервер конфігурують на режим «тільки читання».

Сучасні Web-сервери часто працюють з розподіленими системами. Вони можуть взаємодіяти з іншими хостами, отримувати або передавати дані. В цьому випадку існує велика спокуса розмістити ці комп'ютери за міжмережевим екраном усередині мережі організації, забезпечивши таким чином безпеку даних, що зберігаються на них. Проте якщо зловмисникові вдасться скомпрометувати Web-сервер, він може бути використаний як стартовий майданчик для атаки на ці системи. Для вилучення такої ситуації необхідно відокремити системи, що спілкуються з Web-сервером, від іншої мережі внутрішнім міжмережевим екраном. Тоді проникнення на Web-сервер і звідти на системи, що спілкуються з ним, не призведе до компрометації всієї корпоративної мережі.

Незважаючи на усі спроби реалізувати безпечну конфігурацію, неможливо добитися гарантованого вилучення усіх вразливостей. Тим більше що Web-сервер, захищений від зовнішніх атак, може бути виведений з ладу порушенням роботи одного з сервісів. У цьому випадку важливо отримувати оперативну інформацію про подібні події, для мінімізації наслідків атаки або швидкого відновлення працездатності сервісу. Для отримання такої інформації використовують мережеві системи виявлення вторгнень (IDS). Вони сканують увесь трафік мережі і виявляють несанкціоновану активність, порушення захисту чи блокування сервера. Сучасні IDS створюють звіт про усі виявлені порушення, одночасно повідомляючи про них адміністраторів шляхом виведення повідомлень на електронну поштову скриньку або монітор. Типові автоматизовані звіти включають також збої мережевих з'єднань і список заблокованих IP-адрес.

Системи виявлення вторгнень, що розміщуються на серверах, краще справляються із завданням визначення стану мережі, ніж мережеві IDS. Маючи всі можливості мережевих IDS, у багатьох випадках серверні IDS краще виявляють спроби порушення захисту, оскільки мають вищий рівень доступу до стану Web-сервера.

Проте і цей спосіб не позбавлений своїх недоліків. Якщо хакер проникне на Web-сервер, він зможе відключити серверні IDS,

блокувавши тим отримання повідомлень про атаку адміністратором. Видалені атаки на відмову сервісу (DoS-атаки) також часто блокують IDS на час виходу з ладу сервера. А оскільки DoS-атаки дозволяють зловмисникам блокувати сервер без проникнення на нього, то IDS, що розташований на сервері, має бути доповнений мережевою системою виявлення вторгнень.

Усі фахівці з безпеки радять використовувати захищене програмне забезпечення (ПЗ), але в деяких випадках установити його неможливо через дорожнечу або брак часу. Мало того, безпечне ПЗ через деякий час застаріває, і необхідно встановлювати нову версію. Тому використання застарілого ПЗ і стандартних методів забезпечення безпеки не може служити гарантією захищеності серверів. Але стійкість Web-сервера до атак може бути досягнута за умови використання сформульованих рішень забезпечення безпеки спільно з надійним ПЗ, під яким у цьому випадку слід розуміти деяке ПЗ, що має певний рівень безпеки.

Висновки. На цей час нормальне функціонування Web-сервера, підключеного до мережі Internet, практично неможливе, якщо не приділяти належну увагу питанням забезпечення його інформаційної безпеки. Ця проблема може бути вирішена шляхом використання комплексного підходу до захисту ресурсів сервера від можливих атак. Для цього до складу комплексу засобів захисту сервера повинні входити системи антивірусного захисту, контролю цілісності, виявлення вторгнень, розмежування доступу, криптографічного захисту, а також підсистема управління. При цьому кожна із систем має бути оснащена елементами власної безпеки.

Список використаних джерел: **1.** Статистика уязвимостей веб-приложений (2013 год) [Електронний ресурс] / Postive Technologies. – 19 с. – Режим доступу: http://www.ptsecurity.ru/download/PT_Web_application_vulnerability_2014_rus.pdf. **2.** Новиков А. Некоторые аспекты безопасности Веб-серверов на Unix платформах [Електронний ресурс] / Андрей Новиков. – Режим доступу: http://www.opennet.ru/base/dev/apache_sec.txt.html. **3.** Мелл П. Обеспечение безопасности web-серверов [Електронний ресурс] / Питер Мелл, Дєвид Феррэйоло // Default Web Project : [сайт]. – Режим доступу: http://www.dflt.ru/articles/networks/obespechenie-bezopastnosti-web-serverov. **4.** Обеспечение безопасности для виртуальных серверов и приложений) [Електронний ресурс] / CISCO. – [2012]. – 8 с. – Режим доступу: http://www.cisco.com/web/RU/downloads/broch/white_paper_c11-652663.pdf. **5.** 10 самых популярных web-уязвимостей OWASP [Електронний ресурс] // UFN-Review. – 2010. – С. 3–4. – Режим доступу: http://www.ufn.com.ua/pub_user/review10.06.pdf. **6.** Web Application Firewall [Електронний ресурс]. – Режим доступу: https://www.owasp.org/index.php/Web_Application_Firewall. **7.** Ошибки ModSecurity и способы их устранения [Електронний ресурс]. – Режим доступу: http://www.remoteshaman.com/server/apache/apache-modsecurity-errors.

Надійшла до редколегії 02.06.2015

Петров К. Э., Кобзев И. В., Онищенко Ю. Н. Политика безопасности Web-приложений и серверов

Рассмотрены вопросы уязвимости использования Web-приложений и серверов, а также каким образом могут быть использованы наиболее общие способы защиты для предотвращения программных атак на серверы и приложения. Функционирование Web-сервера невозможно, если не уделять надлежащее внимание вопросам обеспечения его информационной безопасности. Эта проблема может быть решена путём использования комплексного подхода к защите ресурсов сервера от возможных атак.

Ключевые слова: Web-приложение, сервер, уязвимость, программное обеспечение, безопасность, защита, атака.

Petrov K. E., Kobzev I. V., Onishchenko Y. M. Security policy of Web-applications and servers

Today the security breaches in Web-applications, same like before, remains one of significant shortcomings in maintaining the information security. Probably, one of the main reasons of low level of security for many Web-applications is negligence and underestimation of risk gravity for information security breaches with integrated Web-applications, which are in free access in Internet.

In the article the most common protection means are discussed, and organizations may adopt such practices for attack prevention on servers or easing the negative consequences.

In the first place for correct organization of Web-server protection there are needs to find out the weak points in Web-applications and Web-server's security policy. Performance of such audit allows reveal faults and evaluate the level of Web-application protection, as well produce the set of recommendations in security enhancement.

Regardless the measures taken, that aimed on providing of Web-server security, the probability of intrusion can't be completely excluded. Therefore, if that security breach happened, it is important to minimize the attack outcome.

One of the successful solutions of this task is the users' privileges distribution, when concrete user is granted rights to run certain programs.

Nowadays, the normal functioning of Web-server, connected to Internet is practically impossible, if do not pay due attention to its information security maintenance. This problem can be solved by utilization of complex approaches in server resources protection from possible threats. In order to solve this problem such approaches should include systems of antivirus protection, integrity checks, intrusions detection, access rights division, cryptography protection and sub-system of control.

Keywords: Web-application, server, vulnerability, software, security, defense, attack.