3. *Парасюк И.Н., Сергиенко И.В.* Пакеты программ анализа данных: технология разработки. – М.: Финансы и статистика, 1988. – 166 с.

4. *Сильвестров Д.С., Семенов Н.А., Марищук В.В.* Пакеты прикладных программ статистического анализа. – К.: Тэхника, 1990. – 176 с.

5. *Дереза А.Ю., Приставка О.П.* Інформаційна технологія оперативного аналізу в системах моніторінгу / Актуальні проблеми автоматизації та інформаційних технологій. – Д.: Вид-во Дніпропетр. ун-ту, 2003. – Т7

6. *Сергиенко И.В., Парасюк И.Н., Тукалевская Н.И.* Автоматизированные системы обработки данных. К.: Наукова думка, – 1976.

7. *Приставка О.П., Приставка П.О., Смирнов С.О.* Статистичний аналіз в АСОД. Часові ряди: редагування, прогнозування: Навч.посіб. – Д.: РВВ ДНУ, – 2001. –92 с.

8. *Литтл Р. Дж.А., Рубин Д.Б.* Статистический анализ данных с пропусками: Пер. с англ. – М.: Финансы и статистика, 1990. – 336 с.

9. Идентификация и восстановление распределений на ЭВМ: Справочное пособие / Приставка А.Ф., Райко О.В., Малаховская Н.Л., Мымриков А.К. / Под ред. С.Н.Конюхова. – Д.: Изд-во ДГУ, 1991. – 216 с.

Надійшла 28.05.2004

УДК 681.3.06

І.А.Терейковський

ДОСЛІДЖЕННЯ СТІЙКОСТІ СЕРВЕРНИХ ТЕХНОЛОГІЙ JAVA ВІД АТАК НА ВІДМОВУ

В теперішній час більшість корпоративних комп'ютерних мереж пов'язані та використовують ресурси глобальної комп'ютерної мережі Інтернет. При цьому концепція "відкритих дверей" проголошена в Інтернет призводить до загострення проблеми забезпечення безпеки корпоративної мережі. Тепер зловмисник може атакувати корпоративну мережу інкогніто та в зручних для себе умовах. Наслідки та кількість порушень безпеки в мережі Інтернет вказують на достатньо високу ефективність хакерського програмного забезпечення і необхідність пошуку методів та засобів активної їм протидії. Навіть добре оснащені сучасні банківські мережі та мережі всесвітньо відомих корпорацій розробників програмного забезпечення не досить ефективно протидіють добре підготовленим та організованим атакам. Реальний захист від таких атак можливий тільки при веденні виваженої та інтегральної політики безпеки. Однією із найбільш важливих складових такої політики безпеки є пошук потенційно небезпечних місць як в самій корпоративні мережі, так і в системі її захисту.

Як показує практика дуже часто об'єктом атаки є Веб – сервер корпоративної мережі а метою атаки є порушення його працездатності. Атака такого типу дістала назву "атаки на відмову". Успішна атака на відмову може призвести до серйозних економічних збитків організації – експлуатанта корпоративної комп'ютерної мережі. Відзначимо, що в зв'язку широкими функціональними можливостями значна кількість Веб – серверів корпоративних мереж використовують технології Java. Хоча забезпеченню необхідного рівня безпеки таких Веб - серверів приділяється досить значна увага, проте ймовірність успішної атаки на відмову зостається досить високою.

Постановка проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями

Забезпечення належного рівня захисту Веб – серверу корпоративної комп'ютерної мережі, що використовує технології Java від атаки на відмову. Проблема безпосередньо пов'язана з важливим науково – практичним завданням забезпечення інформаційної безпеки корпоративних комп'ютерних мереж.

Аналіз останніх досягнень та публікацій, в яких започатковано розв'язання даної проблеми і на які спирається автор

В корпоративних мережах однієї з основних задач, покладених на Веб – сервер є забезпечення санкціонованого віддаленого доступу клієнта мережі Інтернет до Веб – сайту даної мережі. Веб – сервер постійно очікує одержання запиту на підключення до сайту. В випадку одержання такого запиту Веб – сервер повинен або дозволити з'єднання та передати відповідь, або відмовити в з'єднанні. Відзначимо, що відповіддю як правило є відправлення клієнту файлу з сторінкою сайту. На багатьох сайтах використовуються інтерактивні сторінки. Це Веб - сторінки зміст яких залежить від інформації переданої клієнтом мережі на Веб - сервер. Інтерактивні сторінки використовуються, наприклад, в чатах, Інтернет магазинах та пошукових системах. Для формування інтерактивних Веб – сторінок на Веб – серверах використовують певні програмні технології - ASP, PHP, Perl, C#, Java та інші. При цьому великою популярністю користується Java. Популярність Java пояснюється можливостями інтеграції з багатьма операційними системами під управліням якої працює Веб - сервер, достатньо широкими функціональними можливостями, підтримкою клієнт – серверної технології та багато поточності, наявністю великої кількості доступних як комерційних так і безкоштовних бібліотек. Крім того, до беззаперечних переваг Java відносять можливість розробки як серверного так і клієнтського програмного забезпечення, яке дістало назву аплетів. Захист аплетів традиційно вважається досить високим. На сьогодні не відомо жодного вірусу, який би міг використовувати аплети для заподіяння шкоди клієнту при стандартних установках браузера. Це є ще однією перевагою Java. Безпека аплетів реалізується завдяки жорсткій моделі захисту, яка передбачає тестування змісту версифікатором байт – коду, наявністю диспетчера безпеки, строгою типізацією мови та можливістю використання цифрового підпису.

В якості основних недоліків Java вказують високі вимоги до апаратного забезпечення Веб – сервера та до кваліфікації прикладного програміста. Високі вимоги до апаратного забезпечення пояснюються не достатньо високою швидкість роботи програм написаних на Java. Вказані недоліки та переваги зумовлюють використання Java на потужних корпоративних Веб – серверах, блокування яких може призвести до серйозних негативних наслідків.

В [3] розглянуті механізм атаки на відмову Веб – сервера. Підґрунтям атаки є те, що мережна операційна система, під управлінням якої працює Веб – сервер, здатна мати тільки обмежене число відкритих з'єднань і відповідати лише на обмежене число запитів. Ці обмеження залежать від різних параметрів системи, основними з яких є швидкодія комп'ютера, обсяг оперативної пам'яті і пропускна здатність каналу зв'язку.

Перший різновид такої атаки полягає в передачі з однієї адреси такої кількості запитів на Веб – сервер, яке дозволить трафік (спрямований "шторм" запитів). В цьому випадку, якщо в системі не передбачені правила, що обмежують число прийнятих запитів з однієї адреси в одиницю часу, то результатом цієї атаки буде переповнення черги запитів. Наслідком переповнення черги може бути як відмова телекомунікаційної служби надання віддалено доступу, так і повна зупинка комп'ютера через неможливість операційної системи займатися нічим іншим, крім обробки запитів. Ефективність таких атак тим вища, чим більша пропускна здатність каналу між зловмисником та Веб – сервером, і тим нижча, чим більша обчислювальна потужність комп'ютера, що атакується. В літературі [1,2] наведені результати здійснення таких атак на різні типи Веб – серверів та операційних систем. Аналіз результатів вказує на досить невисоку ефективність таких атак в випадку якісного адміністрування Веб – серверу, мережної операційної системи та використанні декількох потужних комп'ютерів - сервера. Наприклад атака на відмову здійснена на Веб – сервер компанії Microsoft на початку 2004 року за допомогою комп'ютерного вірусу не принесла бажаних для зловмисників результатів.

Другим різновидом атаки на відмову є передача на об'єкт, що атакується, некоректного, спеціально підібраного запиту. У цьому випадку при наявності помилок у

мережній операційний системі можливе зациклення процедури обробки запиту, переповнення буфера з наступним зависанням системи. Як правило, такі атаки здійснюються на рівні протоколу TCP/IP, що є базовим при передачі даних в мережі Інтернет. Захист від атак цього типу досить ефективно реалізується організаціями – розробниками мережних операційних систем.

Третій різновид атаки на відмову базується на санкціонованому використанні серверних програмних додатків. На багатьох Веб - серверах санкціонований запуск програмних додатків може здійснюватись анонімно, хоча і в межах установлених правил. При цьому, використовуються ресурси мережної операційної системи та обчислювальні можливості комп'ютера. Таким чином, атака на відмову здійснена на рівні санкціонованого використання програмних додатків несе потенційну загрозу. Очікувані результати такої атаки полягають в додатковому навантаженні на центральний процесор (процесори) комп'ютера – сервера, запис на жорсткий диск великого обсягу інформації, відкриття великої кількості мережених з'єднань, запуск великої кількості обчислювальних процесів. Негативними наслідками атаки можуть бути вповільнення, обмеження функціональних можливостей або навіть повна зупинка комп'ютера – сервера. Хоча атаки такого типу трапляються в мережі Інтернет досить часто структура факторів, що впливають на їх ефективність описана не достатньо чітко.

Розглянемо заходи безпеки, що рекомендуються при встановленні на Веб – сервері віртуальної машини Java та створенні Java програм [4]. Як правило, вони повинні бути інтегровані з системою безпеки Веб – серверу. Загальними заходами підтримки належного рівня безпеки Веб – серверу є обмеження прав доступу до каталогу в якому встановлюється сервер, заборона на пере визначення користувачами загальних настройок, заборона доступу до файлів, що асоційовані з сайтом, визначення кожному користувачу власної домашньої сторінки. Особлива увага приділяється настройкам CGI сценаріїв, зміст яких полягає в обмеженні прав доступу користувачів для їх запуску та модифікації, а також обмеження можливостей сценаріїв модифікації файлів на сервері.

Специфічними заходами безпеки при встановленні віртуальної машини Java є реалізація обмеження прав доступу до комп'ютерних ресурсів на основі зовнішнього файлу конфігурації. До ресурсів відносяться файли, каталоги, машини та порти. Адміністратором комп'ютера на якому буде працювати Java, встановлюється політика безпеки, яка визначає, хто може отримати доступ до кожного із ресурсів. доступ передбачає права на читання та запис для файлів та каталогів і право на з'єднання для машин та портів. Політика безпеки визначається у зовнішньому файлі конфігурації системи безпеки. При завантаженні файлу класу Java віртуальна машина перевіряє діючу політику безпеки. Якщо код містить цифровий підпис, права доступу можуть бути визначені на основі особи його власника. Права доступу можуть також призначатись в залежності від розташування файлу класу. Наприклад, файл, завантажений з локального серверу, може отримати більш високі права доступу, ніж файл завантажений з мережі Інтернет.

Заходи безпеки при створенні Java програм пов'язані в першу чергу з ідентифікацією користувачів та використанням криптографічних бібліотек для захисту потоку даних між Веб – сервером та браузером згідно протоколу SSL. Крім цього, рекомендується перевіряти введені користувачем дані. з метою заборони виконання несанкціонованих функцій. Таким чином, представлені заходи безпеки, що рекомендуються при встановленні віртуальної машини Java та створенні Java програм стосуються в основному захисту від несанкціонованого доступу та зміни інформації. При цьому, методи захисту від атаки на відмову Веб – серверу з санкціонованим використанням Java програм не достатньо надійні. Однією з можливих причин є відсутність методики реалізації атаки такого типу. Крім того, немає порівняльного аналізу ефективності атаки на відмову Java - сторінки по відношенню до звичайної HTML – сторінки.

Виділення невирішених частин загальної проблеми, котрим присвячується дана стаття

- Відсутні показники технічної ефективності атаки на відмову з санкціонованим використанням серверних технологій Java;
- Методи та засоби атаки на відмову Веб – серверу з санкціонованим використанням серверних технологій Java описані не достатньо;
- Методи та засоби захисту від атаки на відмову Веб – серверу з санкціонованим використанням серверних технологій Java не досконалі.

Формулювання цілей статті (постановка завдання)

- Формування показнику технічної ефективності атаки на відмову з санкціонованим використанням серверних технологій Java;
- Вдосконалення методики захисту від атак на відмову Веб – серверу, що санкціоновано використовують серверні технології Java.

Виклад основного матеріалу з повним обґрунтуванням отриманих наукових результатів

В загальному випадку технічну ефективність атаки на відмову можна оцінювати за допомогою показників, що характеризують доступність ресурсів Веб - сайту. При цьому, сигналом недоступності ресурсу для більшості програм – клієнтів (браузерів) є перевищення встановленого терміну отримання Веб – сторінки. Обмежимо очікувані умови атаки кваліфікованим адмініструванням Веб – серверу, відсутністю помилок в програмному забезпеченні та достатньою пропускною спроможністю каналів зв'язку, що обслуговують Веб – сервер. Відзначимо, що збільшити пропускну спроможність каналів зв'язку можливо за рахунок використання так званих дзеркал сайту. По цій причині, основним напрямком блокування ресурсів Веб – сайту є вичерпання обчислювальних можливостей комп'ютера, що обслуговує Веб - сервер, мережної операційної системи та Веб – сервера. Використання цих ресурсів призводить до збільшення терміну часу, який потрібен серверу для відповіді на запит клієнта. Використання на Веб – сторінках серверних технологій Java призводить до додаткового використання обчислювальних можливостей сервера, по відношенню до звичайних HTML сторінок. Таким чином, підвищення ефективності атаки на відмову при санкціонованому використанні серверних технологій Java пояснюється збільшенням терміну часу який потрібен серверу для формування відповіді клієнту. Ця передумова стала базою для проведення числових експериментів. В плані експериментів було передбачено моделювання атаки на HTML сторінку, на Веб – сторінки, що використовують Java технології та порівняння параметрів функціонування серверу при здійснені таких атак. Крім того, для кращого узагальнення результатів досліджень було проведено моделюванню атаки шляхом відкриття / закриття файлу, що знаходиться на комп'ютері - сервері без використання Веб – серверу.

В якості засобу здійснення атаки на відмову нами було розроблено спеціальну програму. Основними можливостями цієї програми є відкриття , зчитування та закриття файлу, що знаходиться на Web – сервері. Інтерфейс користувача програми дозволяє визначити протокол, повне ім'я файлу, кількість сеансів доступу та параметри доступу (відкриття, закриття, кількість, порядок та місцезнаходження інформації зчитаної із файлу). Програма розроблена в середовищі Microsoft VC++.NET з використання бібліотеки afxinet для роботи з Інтернет та стандартної бібліотеки MFC для роботи з файлами. Основними етапами роботи програми є відкриття сесії, багаторазовий доступ до файлу по заданому протоколу в циклі з визначеною кількістю ітерацій, закриття сесії, обчислення терміну виконання циклу, візуалізація та запис результатів. Зазначимо, що обсяг програмного коду для реалізації атаки становить близько 50 рядків.

В якості Web - серверів використано Apache та Tomcat, які працювали на комп'ютері Intel Pentium 3 під управлінням операційної системи Microsoft Windows. Атака здійснювалась з використанням комп'ютера з аналогічними характеристиками, що також працював під управлінням операційної системи Microsoft Windows. Реєстрація тривалості виконання сервером клієнтських запитів здійснювалась програмно.

На першому етапі моделювання були проведені експерименти по відкриттю / закриттю HTML файлу по протоколу HTTP та відкриттю / закриттю HTML файлу через мережу без використання Web - серверу. Результати експериментів представлені в табл.1.

Табл. 1 Термін виконання комп'ютером, що забезпечує функціонування Web – сайту клієнтських запитів

Кількість з'єднань	Термін виконання запиту (години: хвилини : секунди)	
	Відкриття/закриття HTML файлу по протоколу HTTP	Відкриття/закриття HTML файлу в мережі без використання Веб - серверу
1	00:00:00	00:00:00
10	00:00:00	00:00:00
100	00:00:01	00:00:00
1000	00:00:05	00:00:00
10000	00:00:44	00:00:01
20000	00:01:29	00:00:03
50000	00:03:41	00:00:08
100000	00:07:22	00:00:15

Аналіз даних табл.1 показує, що використання при доступі до файлу Веб – серверу збільшує термін виконання запиту в 30 – 40 разів.

На другому етапі моделювання були проведені експерименти по відкриттю / закриттю Веб – сторінок, що використовують серверні технології Java. Особливістю таких сторінок є те, що в теперішній час для формування інтерактивних Веб – сторінок використовуються дві таких технології. Це так звані сервлети (Servlet) та серверні сторінки Java (JSP).

Прикладний інтерфейс сервлетів надає загальне рішення задачі по обробці запитів клієнта, згідно протоколу передачі гіпертекста (HTTP). Сервлет є класичним поставщиком послуг. Для цього в ньому визначений метод service(), який використовується Веб – сервером для передачі сервлету запитів клієнта. Основна задача метода service() – взаємодія з HTTP – запитом клієнта та створеня HTTP – відповіді, основаної на даних запиту. Також в каркасі сервлета визначені методи, пов'язані з його життєвим циклом. Це методи init(), котрий визивається при завантаженні сервлета та метод destroy(), що визивається при вилученні сервлета із пам'яті. Важливою властивістю сервлетів є те, що як тільки сервлет визначеного класу завантажений в контейнер, всі поля цього сервлету фактично стають постійними об'єктами. Значення полів можуть бути знищені тільки якщо сервлет буде перезавантажений. Це є однією із переваг сервлетів перед класичними CGI сценаріями в яких значення полів потрібно записувати на диск.

В прикладних задачах найчастіше в якості базового класу сервлетів використовується класс HTTPServlet. Однією із переваг даного класу є визначення в ньому методів doGet() та doPost(), які розрізняють тип запиту клієнта – це може бути CGI – запит GET або CGI – запит POST. Відзначимо, що запити GET та POST відрізняються способом передачі даних від клієнта на сервер.

В загальному випадку у контейнера сервлетів буде декілька потоків обчислень, які він буде виконувати відповідно до запитів клієнтів. Достатньо висока ймовірність виникнення ситуації в якій декілька клієнтів одночасно під'єднуються до Веб –сервера та будуть одночасно викликати один і той же сервлет. Тому методи service(), doGet() та doPost() як правило пишуть з врахування багатозадачності. При цьому синхронізують доступ до розподіляємих ресурсів (файлів, баз даних).

Серверні сторінки Java є стандартним розширенням платформи Java та побудовані на основі технології сервлетів. Призначенням серверних сторінок є спрощення створення та управління динамічних Веб – сторінок. Технологія JSP дозволяє комбінувати HTML – код Веб – сторінки з фрагментами Java коду в одному документі. Код Java обмежується спеціальними тегами, які вкащзують контейнеру JSP, щоб він використовував цей код для генерації сервлета або його частини. Перевагою серверних сторінок Java є те, що вони

дозволяють прикладному програмісту підтримувати один документ, який одночасно містить і HTML - сторінку і код Java, який керує цією сторінкою.

Коли сторінка JSP перший раз завантажуєтьься контейнером JSP (який найчастіше пов'язаний з Веб – сервером або є його складовою частиною), генерується код сервлета, необхідний для виконання коду, записаного в сторінці JSP. Потім цей код компілюється та завантажується в контейнер сервлета. Статичні частини HTML – сторінки посиляються в вигляді рядків (String) в метод write(), що є записує дані в стандартний потік виводу сервлету. Динамічні частини сторінки включаються в код сервлету. Після цього, якщо код код серверної сторінки JSP не змінюється, сторінка веде себе як звичайна HTML – сторінка з асоційованим сервлетом. При модифікації початкового коду JSP вона автоматично перекомпілюється та перезавантажується при наступних зверненнях до неї. Практичний досвід свідчить, що перше завантаження серверної сторінки Java займає досить значний проміжок часу.

Таким чином механізми проектуваня та функціонування сервлетів та серверних сторінок Java, хоча і мають багато спільного, але дещо відрізняються. По цій причині на другому етапі моделювання були проведені експерименти по відкриттю / закриттю сервлету та відкриттю / закриттю серверної сторінки Java. Звернення до Веб –сервера проводилось відповідно протоколу HTTP методом GET. Результати експериментів представлені в табл.2 та на рис.1.

Табл. 2 Термін виконання комп'ютером, що забезпечує функціонування Web – сайту клієнтських запитів

Кількість з'єднань	Термін виконання запиту (години: хвилини : секунди)	
	Відкриття/закриття Servlet	Відкриття/закриття JSP
1	00:00:00	00:00:00
10	00:00:00	00:00:00
100	00:00:01	00:00:01
1000	00:00:07	00:00:11
2000	00:00:14	00:00:17
2500	00:00:15	00:00:21
2750	00:00:17	00:00:23
3000	00:00:18	00:00:26
4000	00:00:19	00:00:34
5000	00:00:33	00:00:42
6000	00:00:38	00:00:43
7000	00:00:44	00:01:00
8000	00:00:51	00:01:09
9000	00:00:57	00:01:17
10000	00:01:03	00:01:25
20000	00:02:07	00:02:51
50000	00:05:17	00:07:07
100000	00:10:31	00:14:13

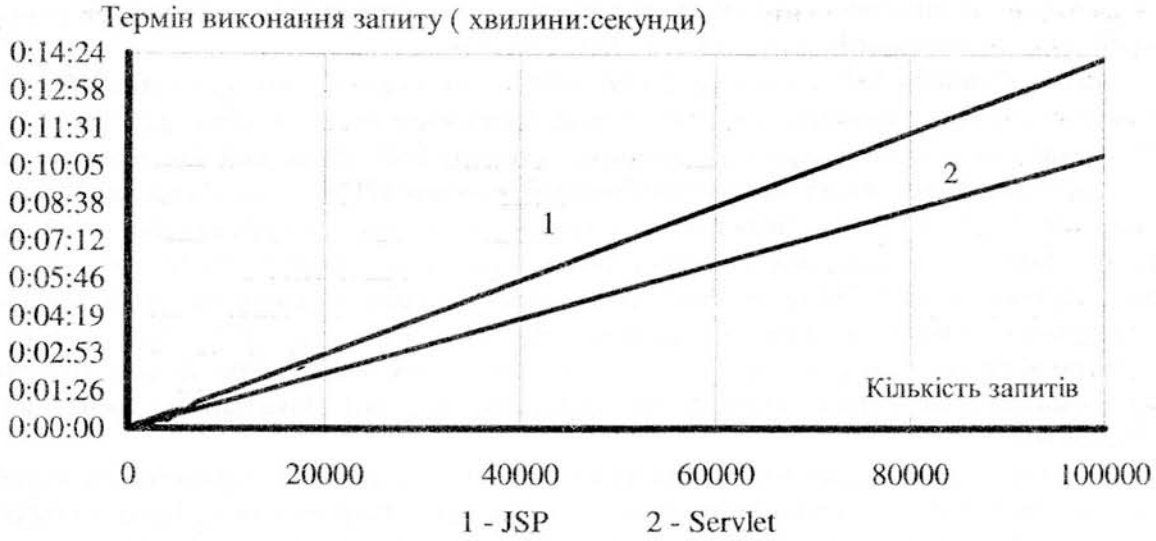

Рис.1 Графіки термінів виконаня запитів на відкриття/закриття JSP та Servlet.

Одержані експериментальні дані апроксимовані методом найменших. квадратів. Отримані функції термінів виконання запитів на відкриття/закриття JSP (1) та Servlet (2) від кількості запитів:

$$Y = 10^{-7} \times X - 4 \times 10^{-6} \quad (1),$$

$$Y = 7 \times 10^{-8} \times X - 2 \times 10^{-6} \quad (2).$$

де Y – термін виконання запиту (секунди), X- кількість запитів.

Аналіз рівнянь (1) та (2) вказує на лінійних характер зростання терміну виконаня запитів від кількості запитів. Крім того, термін виконання запитів сервлетів менший від терміну виконання запитів серверними сторінками Java. Для порівняння ефективності атаки на відмову були побудовані графіки відношення терміну відкриття/закриття Servlet до терміну відкриття/закриття HTML сторінки та відношення терміну відкриття/закриття JSP до терміну відкриття/закриття HTML сторінки для різної кількості запитів. Вказані графіки показані на рис.2.

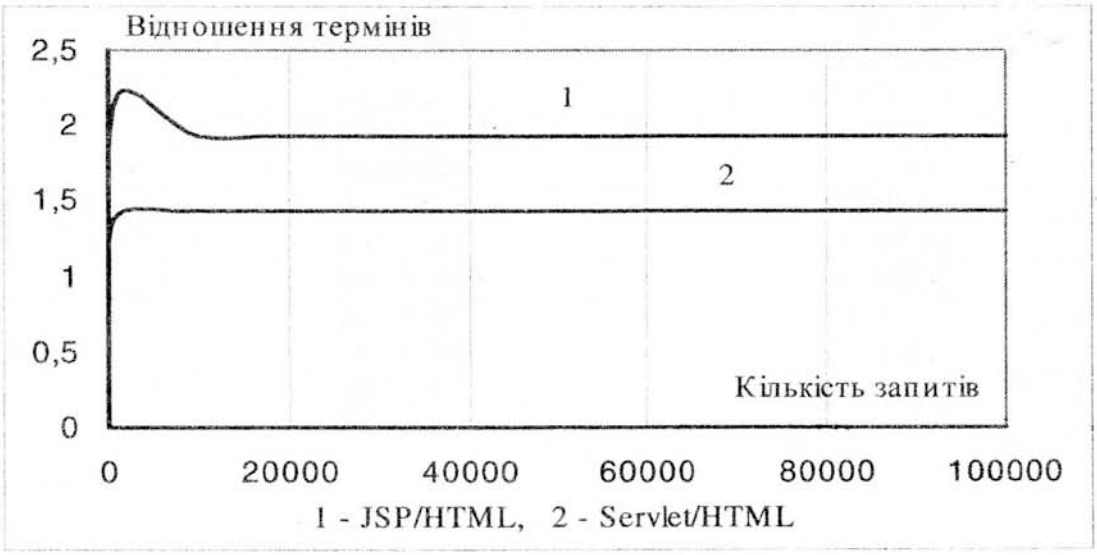

Рис.2 Відношення термінів відкриття/закриття JSP та Servlet до терміну відкриття/закриття HTML файлу.

Аналіз цих графіків показує, що при великій кількості запитів яка характерна при здійсненні атаки на відмову відношення терміну відкриття/закриття JSP файлу до HTML файлу становить 1,8 – 1,9, для сервлету це відношення 1,3 – 1,5.

Проведені експерименти реалізації атаки в яких файли тільки відкривались, зчитувались, але не закривались. Результати цих експериментів показали недоцільність атак такого типу.

Проведено моделювання реалізації даної атаки декількома зловмисниками. Тривалість термінів відкриття/закриття файлів сайту зростала прямо пропорційна кількості нападників. Відзначимо, що значного зростання використаної оперативної пам'яті не зафіксовано.

Таким чином, із розглянутих, найбільш небезпечною є синхронізована в часі атака з використанням виключно відкриття/закриття JSP – файлу.

Вдосконалення методики захисту від атаки на відмову Веб – сторінок, що використовують технології Java можливо провести шляхом доробки програмного забезпечення комп'ютера – сервера з метою заборони короткотермінового та багаторазового звернення із однієї IP – адреси до Веб - сервера. Найбільш доцільним рішенням може бути доробка програмного забезпечення Веб - сервера, що розповсюджується з відкритим кодом, наприклад Tomcat. Крім реалізації вказаної заборони доробка може включати вдосконалення програмного забезпечення Веб - сервера для погіршення характеристик функціонування комп'ютера – зловмисника.

Висновки з данного дослідження

- В якості одного з основних показників технічної ефективності атаки на відмову Веб – сторінки з санкціонованим використанням серверних технологій Java, можливо використати термін (проміжок часу) виконання сервером запиту на доступ до цієї сторінки.
- Технічна ефективність атаки на відмову з санкціонованим використанням серверних технологій Java для сервлетів в 1,8 – 1,9 разів, а для серверних сторінок Java в 1,3 – 1,5 вища від ефективності атаки на звичайну Веб – сторінку.
- Запропоновано ряд заходів для вдосконалення програмного забезпечення Веб – серверу спрямованих на захист від атаки на відмову з санкціонованим використання серверних технологій Java.

Перспективи подальших розвиток у даному напрямку

- Розробка універсальної методики захисту Веб - сайтів, що використовують сценарії на стороні сервера від атаки на відмову.
- Розробка методики доцільності застосування серверних сценаріїв з точки зору ефективності захисту від атаки на відмову.
- Дослідження методики захисту від атаки, що використовує потенційне обмеження оперативної пам'яті комп'ютера – сервера.

Список літератури

1. Белкин П.Ю., Михальский О.О., Першаков А.С. Программно-аппаратные средства обеспечения информационной безопасности. Защита программ и данных — М.: Радио и связь, 1999. - 168 с.
2. Герасименко В. А. Основы защиты информации.— М.: Изд-во «Инкомбук», 1997. - 537 с.
3. Медведовский И.Д., Семьянов П.В., Платонов В.В. Атака через Internet. - М.: ДМК, 1999. - 336 с.
4. Эккель Б. Философия Java. Библиотека программиста - СПб.: Питер, 2001. - 880 с.

Надійшла 28.04.2004