УДК 681.518 : 004 : 912

А.Л. Ерохин, А.П. Турута

Харьковский национальный университет внутренних дел

РАЗРАБОТКА МЕТОДА ИДЕНТИФИКАЦИИ СЕТЕВЫХ АГЕНТОВ В WEB

Статья посвящена решению задачи повышения эффективности работы web-сервера при переходе из нормального режима в режим «отказ в обслуживании». Для идентификации сетевых агентов в web-сетях предложено использовать алгоритм вычисления оценок с использованием оценок признаков подключенных сетевых агентов без использования системных параметров серверов и каналов связи

сетевой агент, идентификация, web, алгоритм вычисления оценок

Введение

При переходе web-сервера из нормального режима в режим «отказ в обслуживании» причинами нарушения доступности сервера могут являться:

1) атака, направленная на «отказ в обслуживании» (в том числе распределенная);

2) изменение структуры сети и аудитории посетителей;

3) увеличение интенсивности использования ресурсов;

4) увеличение количества файловых ресурсов и их потребителей.

Компенсация или устранение указанных нарушений позволит увеличить эффективность работы web-сервера в целом. Под критериями эффективности понимаются:

1) увеличение суммарного объема полезных данных в ответах;

2) увеличение количества обслуженных клиентов;

3) увеличение времени доступности;

4) сохранение резерва производительности.

Традиционные подходы решают задачу не эффективно. Известны три основных подхода [1, 2]:

1) анализ на основе теории массового обслуживания. Такой анализ либо фиксирует и объясняет наступление перегрузки, либо улучшение связано с затратными методами: усиление производительности серверов, увеличение пропускной способности канала связи;

2) анализ параметров в точке подключения (например, программное обеспечение Snort – www.snort.org). Такой способ не предоставляет общей картины и не эффективен в случае распределенных атак;

3) сигнатурные методы. Узнавание перегрузки основывается на анализе контента трафика и заранее определенных признаков, но исследуемые данные могут быть закодированы или новый метод воздействия может быть неизвестен.

Указанные подходы опираются исключительно на анализ технических данных. Их недостатком является игнорирования причины возникновения нагрузки – запросов клиентов к серверу и их возможные цели. Поэтому предлагается улучшить эффективность работы сервера не за счет улучшения методов оптимизации, а за счет узнавания причин текущей ситуации и возможных путей (тенденций) ее развития. Актуальным также является исследование причин возникновения перегрузки web-сервера – запросов сетевых агентов.

Постановка задачи исследования. Общая задача улучшения эффективности обслуживания запросов состоит из ряда подзадач:

– идентификация сетевых агентов;

– анализ состояния и характеристик серверов;

– анализ состояния и характеристик каналов связи;

– оценка общих тенденций;

– оценка общей ситуации среди подключенных СА.

В работе рассматривается идентификация сетевого агента, направившего запрос к web-серверу. При решении поставленной задачи предполагается оценка признаков подключившегося сетевого агента, без использования системных параметров серверов и каналов связи. Очевидно, что на момент соединения информации о сетевом агенте недостаточно для идентификации. Ее количество будет увеличиваться во время сессии.

Особенность решаемой задачи заключается в том, что часть признаков может быть неизвестна при идентификации СА и, в случае необходимости, неизвестные признаки могут быть выяснены.

Особенность работы СА допускают работу различных СА (их классов), при одинаковых идентификаторах отправителя. И в тоже время один СА может отправлять запросы с различных идентификаторов отправителя. Отметим, что внешние факторы могут скачкообразно влиять на внутренние характеристики.

1. Разработка формальной модели признаков агентов

По способу формирования признаков предлагается выделить три группы признаков:

1) признаки, соответствующие одиночному запросу: идентификатор запрашиваемого файла, передаваемый контент от СА к серверу [3], параметры расположения источника, содержание заголовка запроса, текущее время;

2) признаки, выясняемые во время сессии (объединенной последовательности запросов): ответ на вопрос-закладку, скоростные характеристики СА, параметры настройки сетевых агентов, признаки работы web-технологий, проверка полученных параметров. Вопрос-закладка – основной способ интерактивной коммуникации с объектом управляющим СА. Рассмотрим классификацию вопросов-закладок:

– когнитивный вопрос;

– запрос данных известных определенному клиенту (заранее выданный код, данные авторизации, данные, переданные по «секретному» каналу);

– результат множественного выбора (среди области выбора, скрытых ссылок);

– указания объекту управления СА о дальнейших действиях;

– запрос данных о данных и планах объекта управляющего СА;

3) вычисляемые признаки: маршрут обхода сайта, статистические временные портреты, пространственное распределение адресов источников СА, оценка темпа работы СА.

Обозначим S – описание СА и зададим в виде вектора значений $S = (x_1(S), x_2(S), ..., x_n(S))$, где

$x_1(S)$ – адрес обращения к серверу;

$x_2(S)$ – имя файла запроса;

$x_3(S)$ – путь к запрашиваемому файлу;

$x_4(S)$ – параметры $_POST, $_GET;

$x_5(S)$ – идентификатор источника промежуточных Proxy-серверов;

$x_6(S)$ – время;

$x_7(S)$ – адрес, предшествующий запросу;

$x_8(S)$ – размер заголовка;

$x_9(S)$ – идентификатор агента;

$x_{10}(S)$ – данные, хранимые на стороне клиента (Cookie);

$x_{11}(S)$ – данные, хранимые на стороне сервера (session);

$x_{12}(S)$ – ответ на вопрос-закладку;

$x_{13}(S)$ – скорость работы с пользователем;

$x_{14}(S)$ – темп поступления заявок;

$x_{15}(S)$ – обратная скорость, вычисленная по предложенному алгоритму;

$x_{16}(S)$ – параметра настройки сетевых агентов;

$x_{17}(S)$ – признаки работающих технологий;

$x_{18}(S)$ – результаты проверки ОС;

$x_{19}(S)$ – результаты проверки браузера;

$x_{20}(S)$ – результаты проверки территориальных признаков;

$x_{21}(S)$ – маршрут обхода сайта;

$x_{22}(S)$ – статистические временные портреты;

$x_{23}(S)$ – пространственное распределение адресов источников СА;

$x_{24}(S)$ – темп работы СА.

Целью задачи идентификации СА является отнесение объекта к известному классу. По различным признакам выделим следующие подклассы.

По степени машинизации управления СА: управляемые человеком; управляемые программой.

По цели субъекта управляющего СА: просмотр страниц, небольших файлов; загрузка файлов; перегрузка системы; большое количество пустых соединений.

По регулярности посещений сетевым агентом исследуемого сервера: гости; регулярные анонимные посетители; авторизированные клиенты.

По назначению: роботы-пауки.

Из приведенных признаков выделим главные составляющие, а незначительные объединим внутри группы «безразличия».

Предлагается, в рассматриваемой задаче, использовать следующие классы: $\mathbf{K_1}$ – клиенты; $\mathbf{K_2}$ – гость – человек; $\mathbf{K_3}$ – гость-программа; $\mathbf{K_4}$ – робот; $\mathbf{K_5}$ – элемент атаки.

В результате идентификации может быть получен следующий результат:

1. Отнесение к классу **j**, принятие решение.

2. Отказ от распознавания, выяснение дополнительных признаков.

3. Отнесение к нескольким признакам, выяснение дополнительных признаков или принятие решения.

2. Разработка метода идентификации сетевых агентов

2.1. Обоснование выбора метода – прототипа. С учетом особенностей решаемой задачи и формализованных исходных данных сформулируем требования к методу идентификации СА:

1) метод должен позволять производить идентификацию объекта при неполном наборе данных, а также уточнять результат при выяснении значения неопределенных признаков;

2) метод должен позволять выделять наиболее информативные из неопределенных признаков;

3) система распознавания должна адаптировать в процессе функционирования;

4) ряд признаков во множестве эталонных объектов может в разное время соответствовать различным СА, следовательно, необходим механизм обеспечения старения отдельных признаков;

5) процессы в сети в некоторых случая зависят от внешних факторов, никак не отраженных в сети.

Поэтому необходимо предложить способ компенсации для соответствующих эталонных объектов.

В табл. 1 приведен обзор методов распознавания, основанных на обучении с учителем.

Таблица 1

Обзор методов распознавания

Для идентификации сетевых агентов удобнее всего использовать параметрическую идентификацию. Преимуществом методов работающих с частичными прецедентами является способность идентифицировать объект при неполных данных. Для решения задачи за основу был выбран метод АВО [4, 5].

Далее будем считать, что признаки принимают числовые значения, выражающие степень какого-то свойства. В случае необходимости (не числовое значение признака, корректировка степени признака, бесконечный интервал) пронормируем признак по K интервалам. При нормировании реальные значения порогов могут получаться в результате эксперимента или из иных источников. Описание объекта S задается в виде вектора значений

$$S = (x_1(S), x_2(S), ..., x_n(S)), \qquad (1)$$

где $n = 24$, а значение $y(S)$ указывает на класс объекта.

Зададим информацию об эталонных объектах $S_1, S_2, ..., S_m, y(S_1), y(S_2), ..., y(S_m)$, где m – количество записей в таблице обучения $T_{nml} = \left(a_{i,j}\right)_{m \times n}$, где $l = 5$ – количество классов, m – количество эталонных объектов, $n = 24$ – количество признаков, а $a_{i,j} = x_j(S_i)$.

2.2. Задание системы опорных множеств. Пусть Ω_A – некоторая система подмножеств множества $\{1, 2, ..., n\}$, называемая системой опорных множеств алгоритма А. Элементы $\Omega = \{i_1, i_2, ..., i_k\} \in \Omega_A$ называются опорными множествами алгоритма. Они определяют номера признаков, по которым осуществляется сравнение частей эталонных и идентифицируемых объектов. Каждому подмножеству $\Omega = \{i_1, i_2, ..., i_k\}$ можно поставить во взаимно-однозначное соответствие характеристический вектор

$\omega = \{\omega_1, \omega_2, ..., \omega_n\}$, где $\omega_j = 1, j = i_1, i_2, ..., i_k$.

Параметр **K** может задаваться экспертом или выводиться в результате оптимизации алгоритма. Особенность параметра **K** заключается в том, что при усечении множества признаков конечный алгоритм может получать следующие результаты:

– решающая функция от усеченного набора признаков будет иметь достаточную разделяющую способность, т.е. идентифицировать как на полном наборе признаков;

– решающая функция будет иметь меньшую разделяющую способность и укажет на информативность неизвестных признаков;

– оснований предполагать, что увеличится вероятность ошибок 1-го и 2-го рода – нет.

Указанную особенность предлагается использовать для идентификации объекта по неполным данным. В результате опорное множество формируется под влиянием двух факторов:

1) подбора параметра **K** – экспертом или в результате оптимизации модели;

2) фактического наличия признаков.

2.3. Задание функции близости. Пусть фиксировано некоторое опорное множество Ω и соответствующий ему характеристический вектор ω. Выделим ωS часть, которая соответствует признакам опорного множества. В исходном алгоритме предлагается два варианта функции близости. Следует отметить, что использование алгоритма для идентификации на неполном наборе признаков вносит неточность в значение базовых функций близости. Одинаковые значения получают как отсутствующий, так и неопределенный признаки. Предлагается дополнить систему идентификации отрицательным значением для не характерного признака. Однако при наличии нескольких строк в обучающей таблице отрицательный признак может вносить ошибку 1-го рода, следовательно, отрицательное значение не должно превышать

$$\Delta = -\frac{1}{\text{unique}(m_{j+1} - m_j)} * \text{positive},$$

где $\text{positive} = 1$ – значение функции близости для характерного признака, а $\text{unique}(m_{j+1} - m_j)$ – количество уникальных признаков среди записей для j-го класса. Таким образом, под функцией близости $B_\Omega(S_i, S_j)$ будем понимать функцию от соответствующих ω-частей сравниваемых объектов, принимают значение 1 (“объекты близки”), если признак характерен классу; $-\Delta$ (“объекты далеки”), если признак не характерен классу; 0, если признак не определен.

$$B_\Omega(S_\nu, S_\mu) = \begin{cases} 1, & |x_i(S_\nu) - x_i(S_\mu)| \le \varepsilon_i, \forall i : \omega_i = 1, \omega \leftrightarrow \Omega; \\ -\Delta, & \text{иначе}, \end{cases} \qquad (2)$$

где $\varepsilon_i, \forall i : \omega_i = 1$ вычисляются при оптимизации алгоритма.

Если в результате голосования функция близости регулярно принимает отрицательное значение, следует сделать вывод о малом количестве значений признаков и необходимости выявления более информативных признаков.

2.4. Оценка близости исследуемого объекта к эталонному объекту. Проведем оценку близости объекта S к эталонному S_i для ωS части, которая формируется на основе функции близости и дополнительных параметров. С учетом особенностей решаемой задачи необходимо параметрически учесть:

– вариативность влияния эталонных объектов;

– веса для признаков;

– возможность старения признаков;

– способ вычисления информативности признаков для выяснения.

В результате функция близости будет иметь вид:

$$\Gamma_\Omega(S_i, S) = \gamma_i \left(\sum_{j:\omega_j=1} p_j * \lambda_i * B_\Omega(S_i, S) \right), \quad (3)$$

где γ_i – задает важность объекта S_i, или близость текущих условий к условиям формирования эталонных объектов. Управление этим параметром позволяет корректировать влияние внешних факторов:

– реклама для информационного портала;

– резонансное событие для новостного сайта;

– день выдачи заработной платы для банка;

– рекламная акция для Е-магазина;

– сессия для сайта джистанционного обучения;

p_j – характеризует веса (информативность) признаков. Этот параметр предлагается использовать для реализации «старения» признака; λ_i – коэффициент старения признака. Для актуального признака $\lambda_i = 1$, чем дольше не обновлялась информация о значении признака, тем меньше значение λ_i. Например, адрес отправителя (IP-source) попал в черный список j-го уровня и через заданный интервал времени переведен в $j-1$-й уровень.

Для выяснения признаков необходимо вычислять их информативность: $P_i = N_i/N$, где N_i – число опорных множеств, в которые входит признак, $N = |\{T\}|$ – общее количество опорных множеств.

2.5. Определение оценок объекта S. Определим оценку объекта S за класс K_j для заданной ω-части.

Для этого используем

$$\Gamma_j^\Omega(S) = \frac{1}{(m_j - m_{j-1})} \sum_{S_i \in K_j} \Gamma_\Omega(S_i, S), \quad (4)$$

где S – исследуемый объект; S_i – эталонный объект; j – номер класса; K_j – множество эталонных объектов j-го класса; $(m_j - m_{j-1})$ – количество эталонных объектов j-го класса; Ω – опорное множество.

Для получения оценки объекта S за класс K_j используем

$$\Gamma_j(S) = \sum_{\Omega \in \Omega_A} \Gamma_j^\Omega(S). \quad (5)$$

2.6. Определение решающего правила. В результате выполнения предыдущих пунктов получен вектор оценок $\Gamma(S) = (\Gamma_1(S), \Gamma_2(S), ..., \Gamma_l(S))$. Тогда решающее правило имеет вид

$$\begin{gathered} r(\Gamma(S)) = (\alpha_1^A(S), \alpha_2^A(S), ..., \alpha_l^A(S)); \\ \alpha_i(S) \in \{0, 1, \Delta\}, \ i = 1, 2, ..., l, \end{gathered} \quad (6)$$

где $\alpha_j^A(S)$ – признак отнесения объекта S в j-й класс.

Из базовых формул вычисления признака наиболее удачной является линейное решающее правило в пространстве R^l оценок:

$$\alpha_i^A(S) = \begin{cases} 1, & \sum_{j=1}^{l} \delta_j^i \Gamma_j(S) \ge c_1^i; \\ \Delta, & c_1^i > \sum_{j=1}^{l} \delta_j^i \Gamma_j(S) > c_2^i; \\ 0, & \sum_{j=1}^{l} \delta_j^i \Gamma_j(S) \le c_2^i, \end{cases} \quad (7)$$

где $\delta_1, \delta_2, \delta_j^i, c_1^i, c_2^i$ – параметры алгоритма.

В зависимости от полученных признаков решающее правило может принимать следующие результаты:

1) отнесение объекта S к известному классу, если признаков мало, но $\alpha_j^A(S) = 1$; если признаков достаточно и $\alpha_j^A(S) = 1$;

2) невозможно отнести к определенному классу, но необходимо выявить дополнительные значения неизвестных признаков, если признаков мало и $\alpha_j^A(S) < \lambda$;

3) объект отнесен к нескольким классам, если признаков достаточно и $\alpha_j^A(S) > \tau$;

4) отказ от классификации, если признаков достаточно, но оценки невелики и равномерно распределены;

5) дополнительным решением может быть занесение признаков СА в «черный список» j-го уровня или повышение уровня для занесенного ранее.

Заметим, что λ, τ вычисляются при оптимизации алгоритма.

Выводы

1. Разработан метод идентификации сетевых агентов на основе алгоритма вычисления оценок.

2. В результате формализации исходных данных выделены группы исходных данных по способу их формирования, а также предложена классификация вопросов-закладок.

3. Выделены классы сетевых агентов для их идентификации. Предложены подходы, улучшающие качество работы АВО для решения поставленной задачи: для предварительной идентификации объектов по неполному набору признаков предлагается использовать АВО на частичных прецедентах; для уменьшения ошибки 1-го рода предлагается функцию близости дополнить отрицательной оценкой, которая характеризует удаление объекта от класса; предлагается оценивать информативность признаков «вопросов-закладок» и выявлять наиболее важные признаки в процессе идентификации.

4. Предложено корректировать вес эталонных объектов обучающей выборки в зависимости от состояния внешних факторов появления признаков эталонного объекта.

5. Обоснован критерий устаревания значения признака эталонного объекта при идентификации СА, связанный с адаптацией распознающего множества вследствие ротации, естественного изменения признаков СА.

6. Направлениями дальнейших исследований являются:

– исследование формирования N-уровневого «черного» списка, "недоверия" к записи, которое возрастает с увеличением номера, с целью разработки порядка занесения и правил удаления (понижения уровня) групп, СА, признаков из списка;

– развитие метода идентификации сетевых агентов для формирования истории СА, сеансов и отдельных признаков. Такой ресурс предполагается хранить распределено, и импортировать с различных серверов, выполняющих сходные функции. Кроме того, предлагается: характеризовать группы СА статистическими характеристиками работы и использовать эти параметры для определения тенденции развития нагрузки; строить портрет СА, который характеризуется временным распределением, используемыми файловыми ресурсами и др.; распределять приоритеты обслуживания СА в зависимости от результатов идентификации СА и заявленных (требуемых) критериев эффективности.

Список литературы

1. Вишневский В.М. Теоретические основы проектирования компьютерных сетей. – М.: Техносфера, 2003. – 512 с.

2. Олійник В.Ф. Основи теорії систем зв'язку: Математичні моделі телекомунікаційних систем. – К.: Техніка, 2000. – 152 с.

3. Shiflett C. HTTP Developer's Handbook Indianapolis: Sams Publishing, 2003. – 312 p.

4. Распознавание, классификация, прогноз. Математические методы и их применение. – М.: Наука, 1989. – 302 с.

5. Журавлев Ю.И., Рязанов В.В., Сенько О.В. Распознавание. Математические методы. Программная система. Практические применения. – М.: Фазис, 2005. – 320 с.

Поступила в редколлегию 23.02.2007

Рецензент: д-р техн. наук, проф. В.А. Краснобаев, Харьковский университет Воздушных Сил им. И. Кожедуба, Харьков.