УДК 004.056.57

А. А. Лавров, А. Р. Лисс

Метод опорных векторов в задаче идентификации версии операционной системы удаленного узла

Представлены результаты практического исследования по определению конфигурации классификатора на базе метода опорных векторов, обеспечивающей наибольшую эффективность работы метода идентификации версии операционной системы удаленного узла, основанного на совместном анализе временных и функциональных характеристик TCP/IP. Проведено практическое исследование эффективности данного метода в сравнении с существующими программными реализациями иных методов идентификации операционной системы.

Методы классификации, метод опорных векторов, идентификация ОС удаленного узла, сетевой мониторинг

В задачах сетевого мониторинга и обеспечения сетевой информационной безопасности находят широкое применение методы сбора информации об удаленных сетевых узлах, важнейшей задачей которых является идентификация версии операционной системы (ОС) удаленного узла. Методы идентификации ОС удаленного сетевого узла (методы ИОС) основаны на анализе характеристик функционирования стека протоколов TCP/IP целевой системы, по результатам которого формируется сигнатура стека TCP/IP. ОС различных версий используют различные реализации стека протоколов TCP/IP, характеризующиеся различающимися сигнатурами, что позволяет по результатам анализа стека TCP/IP некоторого неизвестного узла с достаточной степенью точности сформулировать предположение о версии его ОС.

В современных сетевых сканерах и иных программных средствах сетевого мониторинга получили распространение методы ИОС, основанные на анализе особенностей реакции целевого узла на специальным образом сформированные пакеты TCP (как правило, не соответствующие RFC) совместно с анализом значений функциональных характеристик TCP/IP, к числу которых относятся время жизни пакета (TTL), размер окна, значение Windows Scale, свойства опций TCP, алгоритм формирования ISN и др. характеристики, определяющие механизм формирования сетевых пакетов стеком протоколов TCP/IP анализируемого узла. Подобные методы обладают рядом недостатков, ограничивающих их применение в программных комплексах сетевого мониторинга и обеспечения сетевой информационной безопасности, среди которых можно выделить высокую интенсивность обмена трафиком и увеличение числа ложных срабатываний систем защиты и обнаружения вторжений. Авторами был предложен метод идентификации версии ОС удаленного сетевого узла TCP-FTA, основанный на совместном анализе функциональных и временны́х характеристик TCP/IP и не использующий алгоритмы анализа реакции целевой системы на нестандартные сетевые воздействия [1], что устраняет основные недостатки известных методов ИОС.

Предложенный метод ИОС предполагает использование метода опорных векторов (МОВ) в качестве многоклассового классификатора, что обусловлено особенностями структуры используемых векторов признаков и потенциальными свойствами базы сигнатур ОС, а именно:

– неоднородностью признаков;

– наличием составных характеристик;

– низкой размерностью векторов;

– низким уровнем шума, обусловленным наличием параметров, принимающих дискретные значения;

– ограниченным числом векторов в обучающей выборке, соответствующим количеству известных различных сигнатур ОС.

Теоретические и практические исследования МОВ в сравнении с другими методами классификации свидетельствуют об эффективности применения МОВ для классификации относительно небольших выборок (размером не более 100 тыс. векторов) с невысоким (менее 2 %) уровнем шума [2], [3]. Таким образом, можно сделать вывод о применимости МОВ для классификации векторов признаков TCP-FTA.

Кроме того, важной особенностью МОВ является его универсальность, заключающаяся в возможности выбором конфигурации классификатора на базе МОВ эмулировать другие методы классификации*.

Основной принцип МОВ – перевод векторов исходного пространства **X** в пространство более высокой размерности **H** и поиск разделяющей гиперплоскости с максимальным зазором в этом пространстве:

$$\mathbf{\Phi} : \mathbf{X} \to \mathbf{H}.$$

Поскольку задача построения разделяющей гиперплоскости сводится к задаче минимизации квадратичного функционала, заданного скалярным произведением (**x**, **y**) двух векторов в пространстве признаков, то на практике для осуществления перехода к пространству более высокой размерности выбирают не само отображение **Φ**, а сразу функцию скалярного произведения **K(x, y)**, называемую *ядром*, которая могла бы быть скалярным произведением при некотором отображении **Φ**(x):

$$K(x, y) = (\Phi(x), \Phi(y)).$$

Точность классификации МОВ зависит от используемого ядра и параметров его настройки – значений числовых параметров, входящих в состав функции отображения, соответствующей используемому ядру. Особенностью МОВ является отсутствие универсальных алгоритмов выбора ядра. В связи с этим для каждой конкретной задачи классификации, определяемой структурой классифицируемых векторов признаков, необходимо эмпирически подбирать ядро и параметры его настройки, обеспечивающие наибольшую точность классификации.

В [1] представлены результаты исследования эффективности предлагаемого авто рами метода ИОС в случае использования линейного ядра МОВ. В настоящей статье приведены результаты исследования эффективности предложенного метода для случаев использования различных ядер МОВ, на основе которого предлагается оптимальная конфигурация МОВ, обеспечивающая наибольшую эффективность предлагаемого метода с точки зрения достоверности получаемых результатов и уровня затрачиваемых вычислительных ресурсов. Под точностью результатов работы метода ИОС понимается вероятность достоверной идентификации версии ОС удаленного сетевого узла, т. е. вероятность правильной классификации неизвестного вектора признаков (сигнатуры ОС).

В качестве вектора признаков TCP-FTA использовался вектор, состоящий из временных характеристик (значений RTO) для ситуации потери пакетов при передаче данных по TCP-соединению и значений функциональных характеристик TCP/IP (размер окна TCP, значение TTL, набор опций и порядок их объявления, значение опции масштабирования окна TCP). В соответствии с [1] данная конфигурация вектора признаков обеспечивает наибольшую точность классификации МОВ в составе метода TCP-FTA.

В целях исследования была использована свободная библиотека libsvm, реализующая функциональность метода опорных векторов в задачах классификации выборок данных. Исследование выполнено для четырех типов ядер:

1. Линейное

$$(L)\colon K(x, y) = (x, y).$$

2. Полиномиальное

$$(P)\colon K(x, y) = (\gamma(x, y) + r)^d.$$

3. Радиально-базисное

$$(R)\colon K(x, y) = \exp(-\gamma \| x - y \|^2).$$

4. Сигмоидальное

$$(S)\colon K(x, y) = \tanh(\gamma(x, y) + r).$$

Исследование проводилось на более чем десяти различных версиях ОС семейств Windows и Linux и более чем 130 персональных компьютерах и серверах. Для обучения МОВ использовались обучающие выборки, включающие по 30 % векторов из набора векторов для каждого класса, соответствующего конкретной версии стека протоколов TCP/IP. Тестирование МОВ проводилось для всего набора векторов (в том числе входящих в состав обучающей выборки). Векторы для формирования обучающей выборки выбирались случайным образом. Для достижения распределения век-

* Воронцов К. В. Лекции по методу опорных векторов // http://www.ccas.ru/voron/download/SVM.pdf.

торов в обучающих выборках, близкого к равномерному, обучение и тестирование МОВ для каждого из анализируемых наборов параметров ядра проводилось 100 000 раз со случайной генерацией обучающей выборки в каждом из тестов.

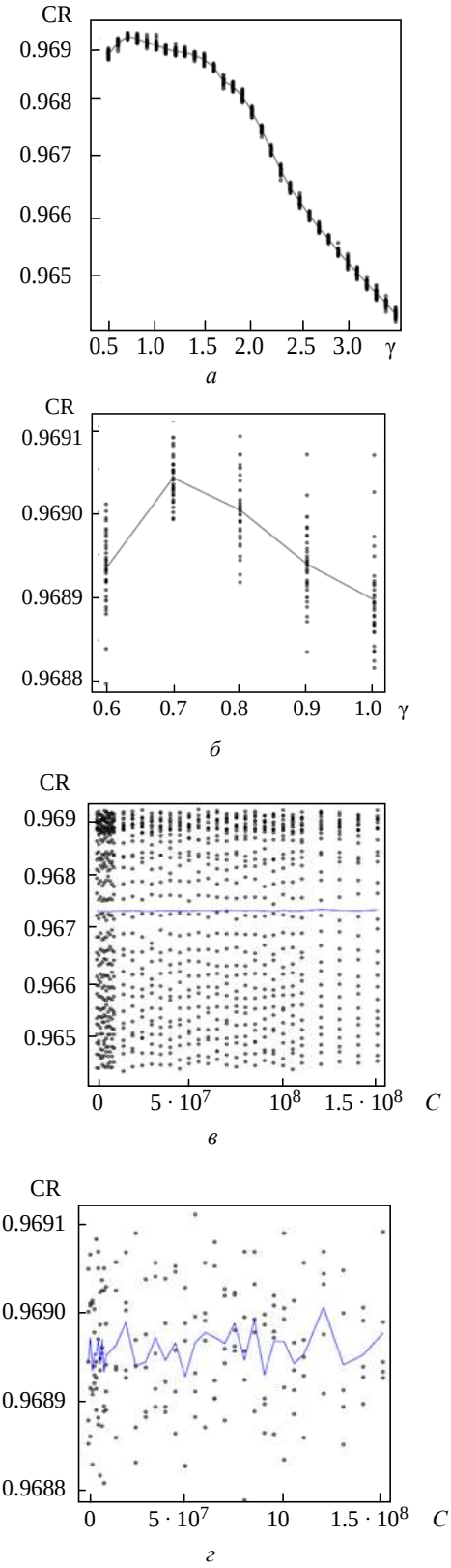
В связи со значительной вычислительной трудоемкостью математических расчетов и необходимостью провести исследование в разумный период времени для выполнения расчетов была разработана GRID-система, предназначенная для организации распределенных вычислений с использованием локальной или глобальной вычислительной сети, функционирующей на базе стека протоколов TCP/IP. Клиентские и серверный модули разработаны на языке C++, используют сетевые средства библиотеки Winsock и предназначены для исполнения под управлением операционных систем семейства Windows. Расчеты выполнялись на аппаратной базе вычислительной сети кафедры МО ЭВМ СПбГЭТУ «ЛЭТИ».

Обучение и тестирование классификатора проводилось на масштабированных данных. Масштабирование выполнялось для каждого из признаков, входящих в вектор, по формуле

$$p = \frac{p - p_{\min}}{p_{\max} - p_{\min}},$$

где $p_{\min}$ и $p_{\max}$ – минимальное и максимальное значения данного параметра среди всех векторов признаков.

Для нахождения оптимальной комбинации значений параметров ядер использовался решетчатый поиск (Grid Search). Для ядер с несколькими параметрами настройки оптимальные значения параметров выбирались с помощью построения средневзвешенных графиков зависимости отношения числа правильно классифицированных векторов признаков к общему количеству векторов в выборке (CR) от значения каждого из параметров ядра (γ, *r*, *d*) и коэффициента *C*, определяющего цену нарушения ограничений в методе *C*-классификации. На рисунке приведены графики зависимостей CR(γ) и CR(*C*) для радиально-базисного ядра, иллюстрирующие поиск оптимальных значений параметров ядра с использованием метода решетчатого поиска. Рисунки *а* и *б* иллюстрируют сужение области поиска наилучшего значения γ методом решетчатого поиска, рисунки *в* и *г* – суждение области поиска значения *C*.



Из графиков следует, что диапазон наилучших значений параметра γ находится в окрестности 0.7. Значение же параметра *C* на всем исследованном диапазоне существенно не влияет на точность классификации.

Обобщенные результаты исследования представлены в табл. 1.

По результатам исследования можно сделать вывод о том, что наиболее эффективной конфигурацией МОВ является радиально-базисное ядро с параметрами настройки γ в диапазоне от 0.7 до 0.9 и C в диапазоне от $50 \cdot 10^6$ до $130 \cdot 10^6$. В случае такой конфигурации МОВ точность классификации достигает 96.91 %. Менее эффективным (точность классификации в среднем на 0.15 % ниже) является полиномиальное ядро. Наиболее низкую точность классификации демонстрирует линейное ядро.

Таблица 1

Ядро	Значения параметров настройки				CR, %	t, мс
	γ	r	d	$C \cdot 10^{-3}$		
P	0.001...0.0011	1...1.08	529...530	236...241	96.74...96.75	0.1181
L	–	–	–	20...40	96.56...96.60	0.1038
S	0.019...0.045	0.05...0.25	–	880...930	96.67...96.68	0.1787
R	0.7...0.9	–	–	50 000...130 000	96.89...96.91	0.1236

Помимо показателя CR для полученных конфигураций каждого из ядер оценивалось среднее время, затрачиваемое на классификацию неизвестного вектора признаков. Представленные в табл. 1 значения данного показателя получены в результате вычислений на ПК, работающем под управлением CPU Intel Core2Duo E8200 2.66 ГГц с 2 Гбайт оперативной памяти. Данная характеристика является усредненным значением и определена для случая многоклассовой классификации с использованием 10 классов.

Как видно из табл. 1, наиболее производительным ядром с точки зрения затрачиваемого на классификацию времени является линейное, затем следуют полиномиальное и радиально-базисное ядра, а наименее производительным ядром является сигмоидальное. Полученные результаты соответствуют теоретическим предположениям о вычислительной сложности функций отображения, определяющих соответствующие ядра.

По результатам исследования для практического применения метода TCP-FTA, основанного на совместном анализе функциональных и временных характеристик TCP/IP, в реальных системах сетевого мониторинга следует рекомендовать выбор радиально-базисного ядра с конфигурацией, соответствующей табл. 1.

Для определения эффективности метода TCP-FTA в сравнении с существующими методами ИОС, получившими распространение в современных сетевых сканерах, проведено соответствующее практическое исследование. При этом вектор признаков TCP-FTA был аналогичен ранее рассмотренному. В качестве классификатора использовался МОВ с радиально-базисным ядром и параметрами настройки, соответствующими табл. 1.

Программная реализация метода TCP-FTA разработана на языке C++ с помощью библиотеки WinPCAP. В качестве программной реализации МОВ использована библиотека libsvm.

Эффективность метода TCP-FTA исследовалась в сравнении с результатами работы следующих сетевых сканеров, реализующих функции ИОС: NMap 6.25, XProbe2, SinFP2, SinFP3. Перечисленные программные продукты представляют собой программные реализации активных методов ИОС. Наиболее распространенным и эффективным средством ИОС в настоящее время считается сетевой сканер NMap, реализующий весь спектр методов ИОС, за исключением анализа временных характеристик TCP/IP. Программный продукт XProbe2 основан на использовании методов анализа ответов ICMP, продукты SinFP2 и SinFP3 используют методы анализа функциональных характеристик TCP/IP.

Исследование включает следующие основные этапы:

1. Формирование базы сигнатур ОС для метода TCP-FTA.

2. Тестирование разработанной программной реализации метода TCP-FTA и программных продуктов NMap 6.25, XProbe2, SinFP2, SinFP3 на ОС, входящих в сформированную базу сигнатур.

3. Обработка и анализ результатов.

Исследование проведено для 46 различных версий ОС: Windows NT 4.00, Windows 98 SE, Windows 2000, Windows 2000 SP2, Windows 2000 SP4, Windows Me, Windows XP SP2, Windows XP SP3, Windows Server 2003 SP1, Windows Server 2003 SP2, Windows Vista, Windows Server 2008 R2 SP1, Windows 7, Windows 8, Windows Server 2012, Linux 2.2.14 (RHL 6.2), Linux 2.4.27 (Debian 3.1), Linux 2.6.4 (SuSE Linux 9.1), Linux 2.6.9 (CentOS 4.4), Linux 2.6.18 (CentOS 5.8), Linux 2.6.32 (CentOS 6.3), Linux 3.1.0 (openSUSE 12.1), Linux

3.7.10 (openSUSE 12.3), Haiku R1 alpha 4, FreeBSD 6.4, FreeBSD 7.0, FreeBSD 7.4, FreeBSD 8.3, FreeBSD 9.1, OpenBSD 3.0, OpenBSD 3.4, OpenBSD 4.0, OpenBSD 5.0, NetBSD 5.0, NetBSD 6.0.1, Solaris 9 (SunOS 5.9), Solaris 10 (SunOS 5.10), Solaris 11 (SunOS 5.11), Mac OS X 10.4.7 (Darwin 8.4.1), Mac OS X 10.5.5 (Darwin 9.2.2), Mac OS X 10.6.5 (Darwin 10.5.0), OS/2 Warp 4, Syllable, QNX 6.3, Novell NetWare 6.5, Android 2.1.

Сформированная база сигнатур ОС для метода TCP-FTA включает 36 наименований.

Для каждого теста формировалась агрегированная оценка точности предположения о версии ОС целевого узла каждой программной реализации, учитывающая помимо непосредственно совпадения результата с действительной версией ОС также результаты конкурирующих программных продуктов. В итоге получена обобщенная оценка точности результатов ИОС для каждой исследованной программной реализации. Помимо точности результатов анализировался уровень потребления сетевого трафика.

Результаты исследования представлены в табл. 2.

Таблица 2

Программная реализация	CR, %	Количество пакетов, шт.	Потребляемый трафик, Кбайт
TCP-FTA	87.07	18	9.51
NMap 6.25	82.39	2173	123.11
XProbe2	55.00	12	0.98
SinFP2	34.24	8	0.50
SinFP3	68.26	14	0.95

Все значения, представленные в табл. 2, являются обобщенными показателями.

Полученные результаты демонстрируют превосходство метода TCP-FTA над NMap по критерию CR, являющемуся основным критерием эффективности методов ИОС. Значительное превосходство наблюдается также и по уровню потребления трафика. Программные реализации XProbe2, SinFP2 и SinFP3 демонстрируют низкую достоверность результатов и не являются конкурентоспособными по сравнению с NMap и TCP-FTA.

Как отмечалось в [1], основным недостатком метода TCP-FTA является длительное время сбора анализируемых характеристик. Так, для ряда ОС (FreeBSD, OpenBSD, Mac OS, устаревшие версии Linux и др.) этот процесс может занять до 15…20 мин. Вместе с тем, в большинстве случаев практического применения методов ИОС (в частности, в задачах аудита информационной безопасности) более важными критериями являются достоверность результатов и минимизация ее влияния на процесс функционирования целевого узла, а также системы IDS и фильтрации трафика. Общеизвестны* серьезные недостатки NMap, заключающиеся в генерации значительного объема сетевого трафика и использовании некорректно сформир ованных сетевых пакетов, что приводит к срабатыванию систем IDS и в ряде случаев становится причиной нарушения работоспособности целевого узла. Этот факт подтверждается в том числе и результатами проведенного исследования. Таким образом, во многих случаях практического применения метод TCP-FTA является эффективной заменой NMap.

СПИСОК ЛИТЕРАТУРЫ

1. Большев А. К., Лавров А. А. Метод идентификации версии системного программного обеспечения удаленного сетевого узла, основанный на комплексном анализе характеристик TCP/IP // Изв. СПбГЭТУ «ЛЭТИ». 2012. Вып. 1. С. 45–51.

2. Wu X. Support vector machines for text categorization: Ph. D. thesis. Buffalo. New York: State University of New York at Buffalo, 2004.

3. Williamson R., Smola A. J., Scholkopf B. New Support Vector Algorithms / Australian National University. Sydney, 1998.

A. A. Lavrov, A. R. Liss

USING SUPPORT VECTOR MACHINE FOR IDENTIFICATION OPERATION SYSTEM OF REMOTE HOST

Results of practical research for determine most effective SVM configuration for method of identification remote host operation system based on integrated analysis of TCP/IP properties is presented. A practical research of efficiency using this method in the aggregate with algorithms for OS identification of existing software realizations is conducted.

Classification algorithms, support vector machine, identification operation system of remote host, network monitoring

* Patrice Auffret. SinFP3. More Than a Complete Framework for Operating System Fingerprinting // http://www.networecon.com/files/sinfp/SinFP3-EuSecWest-ekoparty-v1.1.pdf.