УДК 004.056.5

Д.А. Кавчук, Е.П. Тумоян, Г.А. Евстафьев

ИНТЕЛЛЕКТУАЛЬНЫЙ ПОДХОД К АНАЛИЗУ РИСКОВ И УЯЗВИМОСТЕЙ ИНФОРМАЦИОННЫХ СИСТЕМ

Современные системы управления рисками основаны на анализе большого количества различных факторов, включающих в себя организационные, правовые и иные «нетехнические» факторы, вследствие чего такие системы не способны с достаточной точностью оценить состояние защищенности компьютерной системы организации. В данной работе предлагается интеллектуальный подход к анализу рисков и уязвимостей информационных систем, учитывающий в своей работе реальные данные о состоянии исследуемой системы, а не субъективные оценки экспертов. Целью настоящего исследования является исключение субъективизма в оценке рисков и сокращение времени, требуемого для процесса анализа рисков и уязвимостей компьютерных системы, что достигается за счет применения математического аппарата искусственной нейронной сети и вероятностного дерева атак. Для оценки эффективности разработанной системы производилась оценка защищенности систем семейства Windows, которая доказала соответствие присвоенных системой оценок реальной защищенности исследуемой информационной среды.

Анализ рисков; валидация уязвимостей; дерево атак; искусственная нейронная сеть.

D.A. Kavchuk, E.P. Tumoyan, G.A. Evstafiev

THE INTELLECTUAL APPROACH TO RISK AND VULNERABILITY ANALYSIS FOR THE INFORMATION SYSTEMS

The existing risk management systems are based on the analyzing plenty of various factors, including organizational, legal and other «non-technical» ones. This leads to inability of such a systems to evaluate the security state of computer system thoroughly. This work suggests the intellectual approach to risk and vulnerability analysis of the information systems. The approach consider the real data about the system under research state rather, than the subjective expert evaluations. The aim of the current research is the exclusion of the subjectivity in risk assessments and the decreasing of time consuming for the process of risk and vulnerability analysis for the computer system. This and is achieved through the use of the mathematical apparatus of the artificial neural network and the probabilistic attack tree. The security assessment for the Windows systems was performed in order to evaluate the efficiency of the developed system. As the result the conformity between the real security of the information system and the assigned evaluations was proved.

Risk analysis; vulnerability validation; attack tree; artificial neural network.

Введение. Использование информационных систем непосредственно связано с определенной совокупностью рисков, основной причиной которых являются уязвимости информационных технологий и систем. Когда риск становится неприемлемо велик, необходимо применять экономически оправданные защитные меры для снижения уровня риска либо же полной ликвидации риска. С количественной точки зрения величина риска является функцией вероятности реализации определенной уязвимости. Все современные системы анализа и управления рисками, как отечественные, так и зарубежные, опираются в своей работе на весьма субъективные мнения экспертов, либо же предполагают применение специальных опросников, на вопросы которых зачастую вынуждены отвечать некомпетентные лица. В этом случае даже использование математически сложных алгоритмов анализа введенных данных никак не поможет повысить качество таких оценок. К тому же, подавляющее большинство систем ориентировано на комплексный подход к анализу и управлению рисками. А при комплексном подходе рассматривается множество лишних для анализа состояния компьютерной системы факторов, поскольку учитываются не только

технические сферы деятельности исследуемой организации, и вследствие этого анализу уровня угроз непосредственно компьютерной части, обеспечивающей любую деятельность предприятия, не уделяется должного внимания. В данной работе предлагается метод и программная система интеллектуального анализа рисков и уязвимостей информационных систем, опирающиеся на применение математического аппарата вероятностных графов и искусственных нейронных сетей и использующие в своей работе реальные данные о состоянии исследуемой системы.

Разработанный метод позволяет произвести количественную оценку рисков (оценить вероятность успешной реализации уязвимостей) для данной информационной системы, а также провести проверку найденных уязвимостей с учетом присвоенных оценок и сформировать рекомендации по уменьшению и ликвидации валидных рисков.

1. Существующие методы и средства. Как уже отмечалось выше, подавляющее большинство систем анализа и управления рисками являются комплексными и представляют собой в основном экспертные системы и специализированные опросники, дополненные сложными средствами математического анализа введенных данных. Среди же систем и методик, делающих основной упор на анализ рисков компьютерных систем можно выделить следующие [1].

а) Методика и система RiskWatch

Программная система RiskWatch [2] предназначена для проведения анализа рисков и выбора обоснованных мер и средств защиты. В своей работе система использует методику, состоящую из четырех этапов.

- Определение предмета исследования. На данном этапе в систему вносится некоторое множество входных параметров. Среди них: тип исследуемой организации, состав исследуемой системы, базовые требования предприятия в области безопасности. Внутри системы описание входных параметров определенным образом формализуется, причем принятая формализация предусматривает возможность дальнейшей детализации введенных параметров. RiskWatch содержит набор шаблонов, включающих такие категории, как защищаемые ресурсы организации, потенциальные потери, типовые угрозы и уязвимости, возможные меры защиты, из которых на текущем этапе оператору предлагается отобрать те, которые для организации являются критичными. При этом имеется возможность добавления новых категорий, а также модификации существующих.
- Внесение данных, конкретизирующих текущее состояние системы. Данные можно импортировать из отчетов инструментальных средств исследования уязвимостей, либо же ввести вручную. На этом этапе потенциальные уязвимости определяются при помощи опросника, содержащего более 600 вопросов, и определяются частота возникновения каждой из выделенных на предыдущем этапе угроз, степень критичности уязвимости и ценность ресурсов. Полученная информация в дальнейшем используется для расчета эффективности внедренных средств защиты.
- Оценка рисков. На этом этапе, после установления связей между угрозами, ресурсами и потерями, производится расчет математического ожидания потерь за год, полученное значение которого принимается в качестве риска.
- На последнем этапе работы система генерирует отчеты.

К достоинствам данного метода можно отнести следующее: метод анализа рисков достаточно гибок и прост, а трудоемкость работ сравнительно невелика. В числе недостатков основным является метод расчета оценок рисков – математическое ожидание потерь за год не является приемлемой оценкой для любого из оцениваемых рисков.

Программная система RiskWatch выпускается только на английском языке, и кроме того является платной, причем стоимость лицензии достаточно велика: от 15000$ за одно рабочее место для небольшой компании; от 125000$ за корпоративную лицензию [3].

б) Методика OCTAVE

OCTAVE (Operationally Critical Threat, Asset, and Vulnerability Evaluation) – методика поведения оценки рисков в организации, разработанная в институте Software Engineering Institute (SEI) при университете Карнеги Меллон (Carnegie Mellon University) [4]. Для оценки потенциальных инцидентов в области безопасности и разработки адекватных контрмер разработчики методики предлагают создать группу специалистов, состоящую только из сотрудников организации. При этом особенный упор делается на то, что весь процесс анализа происходит без участия сторонних экспертов и консультативных предприятий в области, т.е. непосредственно силами организации.

OCTAVE предполагает три фазы анализа:

- разработка профиля угроз, критичных для предприятия. Для описания профиля предлагается использовать так называемые «деревья вариантов», которые позволяют представить сочетание угрозы и ресурса в формализованном виде;
- идентификация инфраструктурных уязвимостей. На данном этапе определяется инфраструктура, поддерживающая выделенные на предыдущем этапе уязвимые места, и рассматриваются технические компоненты системы, такие как, например, сервера и персональные компьютеры, которые затем проверяются на наличие уязвимостей с использованием сканеров безопасности;
- разработка стратегии и планов по обеспечению безопасности. Данная стадия включает в себя оценку рисков, которая проводится на базе отчетов по двум предыдущим этапам, причем дается только оценка ожидаемого ущерба, без оценки вероятности, по шкале высокий (high), средний (middle), низкий (low); и формирование планов снижения рисков вместе с определением мер противодействия угрозам.

Сильной стороной OCTAVE является высокая степень гибкости методологии при адаптации под нужны конкретного предприятия, достигаемая благодаря возможности самостоятельного выбора критериев. А к недостаткам можно отнести высокую степень участия человека, что неизменно ведет к субъективизму оценки. Методология разработана в основном для применения в крупных компаниях.

2. Предлагаемый метод анализа рисков и уязвимостей. В предлагаемом методе анализ рисков производится на основании информации об исследуемой системе и об уязвимостях системы, а также на основании результатов валидации найденных и предполагаемых уязвимостей. Для осуществления валидации уязвимостей строится дерево атак для рассматриваемой системы.

Предлагаемый метод можно сформулировать следующим образом:

1) Получение информации об исследуемой системе – семейство, версия, service pack, список и баннеры сервисов.

$$osInf = \{S, Ver, SP, serv, servB\},$$

где S – семейство ОС; $Ver = \{ver\_i\}$ – конечное множество (к.м.) предположительных версий ОС; $SP = \{sp\_i\}$ – к.м. предположительных service pack ОС; $serv = \{serv\_i\}$ – к.м. сервисов; $servB = \{servB\_i\}$ – к.м. баннеров сервисов.

2) Получение информации об уязвимостях.

$$vulnInf = \{vuln\_i\},$$

где $vuln\_i$ – предположительная уязвимость системы.

3) Построение дерева атак для целевой системы. Узлы дерева представляют собой эксплойты, ребра – оценки рисков для каждого конкретного листа дерева (эксплойта). Дерево включает все возможные эксплойты для данной операционной системы.

$$G = (V, E),$$

где V – множество вершин: $V = \{exp\_i\}$, exp_i – допустимый эксплойт: $exp\_i = F(osInf, vulnInf)$; E – множество ребер: $E = \{Ri\}$, Ri — оценка риска для i-ого эксплойта.

Построение дерева выполняется с использованием эвристического алгоритма на основе информации об уязвимостях системы. В частности, если эксплойт может сработать без предварительных условий, то он размещается в узле с уровнем равным 1 (на следующем за корнем уровне дерева). Если же для успешного срабатывания эксплойта необходима реализация некоторых предварительных условий, то он размещается в поддереве, корнем которого является реализующий данные условия узел [5].

4) Уточнение дерева атак. Производится оценка риска для каждого выбранного эксплойта с учетом информации об операционной системе и эксплойте {*osInf* | *vulnInf* }. Оценка риска в данном случае является количественной и представляет собой вероятность успешного срабатывания эксплойта. Для получения вероятностной оценки могут использоваться различные техники, в частности таблицы. Однако, поскольку пространство анализируемых данных имеет большую размерность, то преобразование пространства данных в вероятность с использованием правил или таблиц не позволяет покрыть все варианты входных данных. Для решения данной проблемы используются обобщающие возможности нейронной сети. Вычисление риска производится по следующей формуле:

$$Ri = neuronet(normRank, confRef, confTarget),$$

где Ri – оценка риска для i-го эксплойта, $normRank$ – нормированное значение критичности эксплойта, $confRef$ – уровень соответствия эксплойта найденным уязвимостям, $confTarget$ – уровень соответствия эксплойта исследуемой операционной системе.

Для решения данной задачи была выбрана искусственная нейронная сеть на основе радиальных базисный функций [6]. На вход сети поступает вектор свойств, ассоциированный с рассматриваемым состоянием дерева, а обучение сети производится на основе экспертной оценки, уточненной при построении конкретного дерева атаки.

5) Обход дерева атак. Целью обхода дерева является нахождение упорядоченного по значению рисков множества всех поддеревьев, эксплойты в которых актуальны для данной системы. Мы обходим дерево от корня к листьям по пути с наибольшим уровнем риска. При обработке очередного узла мы запускаем эксплойт с параметрами, наиболее подходящими для данной операционной системы и набора сервисов. Если очередной узел не сработал, то мы производим усечение всего поддерева с корнем в данном узле, и переходим далее, обходя оставшуюся часть дерева по убыванию условной критичности путей, т.е. по убыванию степени рисков.

6) Построение последовательностей эксплойтов и формирование рекомендаций по снижению уровня рисков. Мы строим цепочки сработавших эксплойтов с указанием возможностей, которые может получить атакующий при выполнении эксплойтов из цепочки. Вместе с тем, в том случае, если это возможно, предоставляется информация по устранению уязвимостей из цепочки, взятая из открытых источников, таких как стандарты CWE (Common Weakness Enumeration) [7] и OSVDB (Open Sourced Vulnerability Database) [8].

На рис. 1 представлен пример работы предложенной модели. С учетом правил построения и обхода дерева можно утверждать, что в процессе обхода будут достигнуты все допустимые вершины дерева, т. е. эксплойты, которые могут сработать на данной ОС. Кроме того, для данного метода можно рассчитать вероятность того, что все актуальные для данной системы эксплойты будут обнаружены за заданное количество шагов.

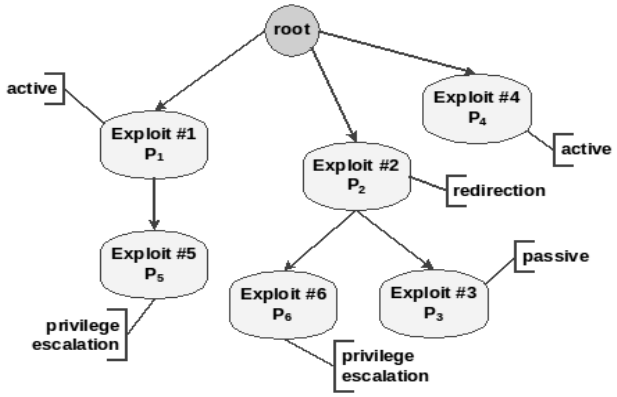

Рис. 1. Пример работы предложенной модели

3. Архитектура системы анализа рисков и уязвимостей. Разработанная система включает следующие взаимодействующие компоненты: сканер уязвимостей Nessus Security Scanner, сервер системы Metasploit Framework и анализатор, разработанный в среде Matlab и взаимодействующий с другими компонентами системы. Для получения значимой информации для формирования рекомендаций из открытых источников используются скрипты на языке Python. Архитектура системы представлена на рис. 2.

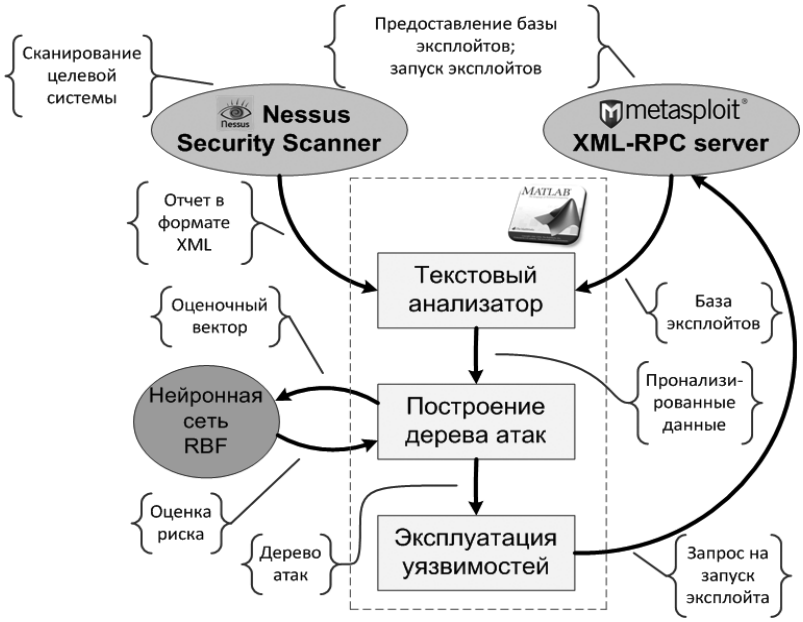

Рис. 2. Архитектура системы

Nessus Security Scanner

Nessus применяется для получения информации об исследуемой системе, а именно информации об уязвимостях системы, о типе и версии операционной системы, об открытых портах и активных сервисах, предоставляемых системой.

В настоящее время данная информация получается в ручном режиме и сохраняется в виде отчета сканера в формате XML, и затем этот отчет в качестве входного параметра передается программе-анализатору.

Metasploit Framework

Анализатор взаимодействует с сервером Metasploit Framework в автоматическом режиме.

Взаимодействие осуществляется посредством протокола HTTP с применением сериализации данных в формат msgpack. После подключения к серверу запрашивается база доступных эксплойтов, которая передаётся скрипту для дальнейшего анализа. Впоследствии системе Metasploit Framework отдается команда на запуск определенного эксплойта и получение результатов работы эксплойта.

Анализатор

Первым этапом работы анализатора в соответствии с предлагаемым методом является сканирование целевой системы. Сканирование производится с использованием сканера уязвимостей Nessus с целью определения типа и версии операционной системы, под управлением которой работает ПЭВМ, открытых портов и сервисов, работающих на ПЭВМ, и уязвимостей исследуемой системы. В настоящее время сканирование производится в ручном режиме, результаты импортируются в систему анализа. Однако, нет теоретических ограничений на получение информации от другого сканера.

Вторым этапом является получение базы эксплойтов из Metasploit Framework. Данный этап выполняется в автоматическом режиме.

Далее следует построение дерева атаки. Алгоритм построения дерева атаки базируется на модели деревьев атак с использованием искусственной нейронной сети для решения задачи оценки рисков для узлов построенного дерева и эвристическим алгоритмом построения связей между узлами.

При построении дерева атаки выполняется выбор подходящих для эксплуатации целевой системы модулей Metasploit на основе анализа полученной на предыдущих этапах информации с последующей генерацией вероятностных оценок рисков для выбранных эксплойтов посредством нейронной сети [9].

На последнем этапе работы анализатора выполняется обход дерева и визуальное отображение построенного дерева с цветовой индикацией риска для эксплойта. Для каждого узла дерева запрос на запуск эксплойта с необходимыми параметрами отсылается серверу Metasploit Framework. Производится проверка успешности срабатывания эксплойта. В случае, если эксплойт не сработал, все дерево, лежащее ниже, отсекается. В случае срабатывания эксплойта узел помечается как успешный, и в соответствии с открытыми стандартами формируются рекомендации по устранению уязвимостей и снижению уровня данного риска.

4. Результаты тестирования системы. В ходе работы была подобрана следующая размерность сети: 8 нейронов входного слоя, 50 нейронов скрытого слоя и 1 нейрон выходного слоя. Для обучения сети используется выборка из приблизительно шестидесяти векторов, сформированная синтетически.

Поскольку разработанная система не имеет на данный момент своих прямых аналогов, оценка эффективности работы системы производилась на основе соответствия присвоенных оценок реальной защищенности исследуемой системы. В процессе тестирования был произведен анализ защищенности и рисков для систем семейства Windows. Результаты приведены в табл. 1.

Таблица 1

Результаты экспериментов

ОС	Найдено предполагаемых уязвимостей	Количество подтвержденных уязвимостей	Диапазон назначенных системой уровней риска для подтвержденных уязвимостей
Windows XP SP3	200	4	0,6–0,85
Windows XP SP2	200	5	0,6–0,85
Windows 2000	236	7	0,7–0,9

Как видно из представленных результатов, для каждой из систем все подтвердившиеся уязвимости вошли в группу высоких показателей уровней рисков, присвоенных системой, что доказывает эффективность системы при назначении вероятностных оценок рисков, а также ее способность сократить время анализа рисков и уязвимостей до минимального.

В соответствии с результатами исследований система имеет ряд преимуществ перед существующими средствами:

- Анализирует и учитывает в своей работе реальную информацию об операционной системе и сервисах исследуемой информационной системы.
- Рассчитывает вероятностные уровни рисков для эксплойтов и осуществляет валидацию найденных уязвимостей в соответствии с присвоенными уровнями (наиболее критичные эксплойты проверяются первыми).
- Обеспечивает визуальное отображение возможных путей атак, формирует рекомендации по устранению уязвимостей и снижению рисков.

5. Дальнейшие исследования. В рамках дальнейших исследований предполагается производить построение и обход графа атаки, что включает запуск не только активных эксплойтов, но и пассивных, а также реализацию многостадийных атак. Для более детального анализа безопасности в максимально приближенных к реальности условиях предполагается производить учет ответной реакции системы на атакующие воздействия, и в соответствии с этим осуществлять коррекцию построенного графа атак.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Евстафьев Г.А.* Сравнительный анализ существующих решений управления рисками ИБ // Неделя науки. – Таганрог: Изд-во ТТИ ЮФУ, 2008. – С. 360-363.
2. RiskWatch Risk assessment [Электронный ресурс] URL: http://riskwatch.com/ (дата обращения: 17.10.2013).
3. *Куканова Н.* Современные методы и средства анализа и контроля рисков информационных систем компаний [Электронный ресурс] URL: http://dsec.ru/ipm-research-center/article/modern_methods_and_means_for_analysis_and_risk_management_of_information_systems_of_companies/ (дата обращения: 17.10.2013).
4. OCTAVE Information Security Risk Evaluation [Электронный ресурс] URL: http://www.cert.org/octave/ (дата обращения: 17.10.2013).
5. *Тумоян Е.П., Кавчук Д.А.* Метод оптимизации автоматической проверки уязвимостей удаленных информационных систем // Безопасность информационных технологий. – 2013. – № 1. – С. 25-30.
6. *Хайкин С.* Нейронные сети: полный курс. – 2-е изд.: Пер. с англ. – М.: Изд. дом "Вильямс", 2006. – 1104 с.
7. Проект CWE [Электронный ресурс] URL: http://cwe.mitre.org/ (дата обращения: 12.08.2013).

8. Проект OSVDB [Электронный ресурс] URL: http://osvdb.org/ (дата обращения: 12.08.2013).
9. *Tumoyan E., Kavchuk D*. The method of optimizing the automatic vulnerability validation // Proceedings of the Fifth International Conference on Security of Information and Networks SIN 2012. – 25-27 October 2012. – P. 205-208.

Статью рекомендовал к опубликованию д.т.н., профессор Я.Е. Ромм.

Кавчук Дарья Александровна – Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Южный федеральный университет»; e-mail: dar.ushka.k@gmail.com; 347928, г. Таганрог, ул. Чехова, 2; тел.: 88634371905; кафедра безопасности информационных технологий; аспирантка.

Тумоян Евгений Петрович – e-mail: e.tumoyan@gmail.com; кафедра безопасности информационных технологий; к.т.н.; доцент.

Евстафьев Георгий Александрович – ООО "Комплексные программные решения"; e-mail: gaevstafiev@yahoo.com; 347900, г. Таганрог, Мариупольское шоссе, 27/2, кв. 305; тел.: 88634605377; инженер-программист.

Kavchuk Daria Alexandrovna – Federal State-Owned Autonomy Educational Establishment of Higher Vocational Education "Southern Federal University"; e-mail: dar.ushka.k@gmail.com; 2, Chekhov street, Taganrog, 347928, Russia; phone: +78634371905; the department of security in data processing technologies; postgraduate student.

Tumoyan Evgenie Petrovich – e-mail: e.tumoyan@gmail.com; the department of security in data processing technologies; cand. of eng. sc.; associate professor.

Evstafiev Georgyi Alexandrovich – OOO "Complex Program Solutions"; e-mail: gaevstafiev@yahoo.com; 27/2, Mariupolskoe shosse, apt. 305, Taganrog, 347900, Russia; phone: +78634605377; programming engineer.

УДК: 004.056

М.А. Кобилев, Е.С. Абрамов

РАЗРАБОТКА АЭРОМОБИЛЬНОГО КОМПЛЕКСА ДЛЯ ПРОВЕДЕНИЯ АУДИТА БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ С ВЫСОКОЙ ФИЗИЧЕСКОЙ ЗАЩИЩЁННОСТЬЮ*

Описываются эксперименты по созданию прототипа лёгкого малозаметного летательного аппарата для проведения удаленного аудита безопасности информационных систем [пентеста]. Была предложена модификация квадрокоптера Parrot AR.Drone 2, предполагающая рациональное использование устройства в задачах анализа безопасности. Прототип работает под управлением Raspberry Pi, дополнительно используется специализированный беспроводной сетевой адаптер ALFA AWUS036NHR с направленной антенной и GPS-приёмник.

Рассмотрены возможные алгоритм функционирования атакующей системы. Предполагается использование прототипа в двух режимах - активного аудита и сервера-ретранслятора. В режиме аудита квадрокоптер непосредственно осуществляет атакующее воздействие на целевую сеть. В режиме ретранслятора квадрокоптер может становится передающим сервером для других устройств, расширяя радиус действия комплекса, либо передавая данные для анализа на мощный стационарный сервер.

* Работа выполнена при поддержке гранта РФФИ № 12-07-00014-а.