УДК 004.056
doi:10.15217/issn1684-8853.2017.6.76

ОЦЕНКА ЗАЩИЩЕННОСТИ КОМПЬЮТЕРНЫХ СЕТЕЙ НА ОСНОВЕ МЕТРИК CVSS

Е. В. Дойникова[а, б], *канд. техн. наук, научный сотрудник*
А. А. Чечулин[а, б], *канд. техн. наук, ведущий научный сотрудник*
И. В. Котенко[а, б], *доктор техн. наук, профессор*
[а]*Санкт-Петербургский институт информатики и автоматизации РАН, Санкт-Петербург, РФ*
[б]*Университет ИТМО, Санкт-Петербург, РФ*

Введение: *в современных условиях чрезвычайно актуальной является проблема улучшения качества оценки защищенности компьютерных сетей для автоматизированных систем защиты, целью которых является проактивная защита от атак, за счет применения объективных количественных показателей защищенности, вычисляемых с использованием метрик открытой системы оценки защищенности CVSS (Common Vulnerability Scoring System) и аналитических моделей.* ***Цель исследования:*** *совершенствование разработанного ранее подхода к оценке защищенности на основе аналитического моделирования, применения открытых стандартов представления данных безопасности и метрик CVSS благодаря новой версии CVSS. В основе подхода лежит графовая модель действий злоумышленника, формируемая с использованием метрик CVSS. Неточности предыдущей версии формата CVSS привели к некоторым допущениям при автоматической генерации графа действий злоумышленника. Предполагается, что применение метрик новой версии позволит усовершенствовать процедуру генерации графа и оценки защищенности на его основе.* ***Результаты:*** *исследования показали, что результаты оценки защищенности существенно зависят от корректности входных данных. Для получения исходных оценок уязвимостей применена система CVSS (метрики для оценки уязвимостей, их описание и критерии назначения оценок) и открытые базы уязвимостей. Выявлены недостатки описания метрик CVSS версии 2 и отличия CVSS версии 3, а именно: введены новые качественные значения метрик для учета существенных для безопасности характеристик, которые ранее игнорировались; определены области возможных значений метрик, снимающие существовавшую ранее неопределенность. Разработан новый подход к моделированию атак и анализу защищенности компьютерных сетей на основе формата CVSS версии 3, описаны его преимущества по сравнению с предложенным авторами ранее подходом, заключающиеся в устранении допущений при формировании графа действий злоумышленника, связанных с определением предусловий и постусловий атакующих действий. Приведены примеры применения метрик CVSS версии 3 для оценки защищенности компьютерных сетей на основе использования методов аналитического моделирования.* ***Практическая значимость:*** *новизна и практическая значимость предложенного подхода заключаются в совершенствовании процедур генерации графа атак и анализа защищенности компьютерных сетей на основе использования нового формата оценки уязвимостей CVSS версии 3 в рамках программной системы оценки защищенности компьютерных сетей.*

Ключевые слова *— аналитическое моделирование, оценка защищенности, показатели защищенности, Common Vulnerability Scoring System, защита от атак, компьютерные сети, графы атак, деревья атак.*

Введение

В современных условиях, когда деятельность все большего количества организаций зависит от защищенного и надежного функционирования компьютерных сетей, проблема мониторинга безопасности компьютерных сетей является особенно важной и актуальной.

Злоумышленники для достижения своих целей зачастую реализуют сложные многошаговые атаки, которые включают последовательность шагов, основанных на эксплуатации различных уязвимостей, ошибок конфигурации и особенностей реализации программно-аппаратного обеспечения. Своевременное обнаружение атакующего в системе и точное прогнозирование его целей может помочь предотвратить серьезный урон системе и избежать больших потерь. Для этих целей исследователями разрабатывались подходы на основе аналитического моделирования.

Было предложено большое количество методов моделирования шагов атакующего в системе, в том числе в виде графов атакующих действий.

Вопросы формирования графов атак и анализа защищенности на их основе рассматриваются во множестве работ [1–12].

Выделяется несколько типов графов атак: полный граф атак [1] включает все пути, которыми атакующий может скомпрометировать сеть; в предиктивном графе [2] узел добавляется в граф в том случае, если ни один предок данного узла не использует ту же уязвимость для перехода в то же состояние, что и новый узел; граф с множеством предусловий [3] включает три типа узлов: состояние, предусловия, уязвимость — и дополнительные циклические дуги для отображения связей с уже существующими узлами.

В ряде работ рассматривается проблема оперативности построения графов атак [3, 4, 16].

На основе графов атак был разработан ряд вероятностных моделей для анализа защищенности системы. В работах [4–7] предлагается использовать вероятностные графы атак, а байесовские графы атак применяются в работах [8–12].

В предыдущих работах авторов были представлены разные варианты алгоритмов построе-

ния и анализа деревьев атак [13, 14], являющихся подклассом графов атак, а также метрики, рассчитываемые на основе деревьев атак [15], и модификация алгоритмов построения деревьев атак для генерации и анализа моделей в режиме, близком к реальному времени [16].

Важной особенностью авторского подхода является применение открытых баз, что, с одной стороны, позволяет учитывать максимальное количество известных уязвимостей, а с другой — автоматизировать процесс. Другой важной особенностью предложенного ранее подхода является оперативность построения графа, что критически важно в динамическом режиме работы системы, когда от своевременности реагирования на атаку зависит уровень итоговых потерь.

В основе предложенного подхода лежит Common Vulnerability Scoring System (CVSS) [17]. Ключевыми факторами, благодаря которым данный формат был выбран, является открытость CVSS-оценок, что позволяет использовать их для формирования собственных показателей защищенности, и наличие связей между CVSS и стандартами Common Platform Enumeration (CPE) [18] и Common Configuration Enumeration (CCE) [19], что позволяет автоматизировать идентификацию и оценку уязвимостей.

Предыдущий формат имел ряд неточностей, которые привели к некоторым допущениям при автоматической генерации графа атакующих действий. В 2015 г. вышла новая версия CVSS, которая учитывает проблемы предыдущей версии.

В статье рассматривается новый формат, анализируются его достоинства и его влияние на подход, предложенный авторами ранее. На основе проведенного анализа предлагаются изменения в процедуре генерации графа и оценки защищенности и рассматриваются преимущества и недостатки нового метода на примере.

Применение CVSS для формирования графов атак

CVSS версии 2.0

CVSS включает ряд метрик, характеризующих уязвимости программного и аппаратного обеспечения и позволяющих получить итоговую интегральную оценку уязвимости, характеризующую ее критичность по сравнению с другими уязвимостями [20]. CVSS включает три группы метрик: основные, временные и контекстные. На данный момент для построения графа используются только основные метрики, поэтому ниже приведено их краткое описание. Значения метрик для известных уязвимостей можно найти в открытой базе уязвимостей NVD [21].

В группу базовых метрик CVSS версии 2.0 входят две группы: *Exploitability* (эксплуатируемость) и *Impact* (ущерб). Метрики группы *Exploitability* определяют способ доступа к уязвимости и необходимость дополнительных условий для ее эксплуатации. Метрики группы *Impact* измеряют влияние уязвимости на актив информационных технологий в случае эксплуатации.

Рассмотрим эти метрики и соответствующие неточности в задании их значений.

К метрикам группы *Exploitability* относятся: *Access Vector* (вектор доступа), *Access Complexity* (сложность доступа) и *Authentication* (аутентификация). *Access Vector* определяет, как эксплуатируется уязвимость (чем более удаленный нарушитель может атаковать хост, тем выше оценка уязвимости). Если уязвимость может быть использована несколькими способами, то выбирается наиболее удаленный доступ. При формировании графа данный показатель используется для определения предусловий эксплуатации уязвимостей, что позволяет сформировать последовательные связи между ними для объединения отдельных шагов в многошаговую атаку. *Неточность 1* состоит в том, что значение показателя «локальный доступ» не определяет, какой именно подразумевается тип доступа — физический или логический. *Access Complexity* задает сложность атаки, которую необходимо провести для эксплуатации уязвимости, после того, как нарушитель получил доступ к системе. Чем ниже сложность, тем выше оценка уязвимости. Данная метрика позволяет определить, насколько вероятна успешная эксплуатация уязвимости. *Неточность 2* состоит в том, что при этом не выделяются отдельно уязвимости, требующие дополнительных действий от пользователя. *Authentication* определяет, сколько раз атакующий должен аутентифицироваться в системе, чтобы использовать уязвимость (сложность процесса не учитывается, только количество). Чем меньше количество процедур аутентификации, тем выше оценка. Данная метрика отличается от метрики *Access Vector*, т. е. считается, что доступ к системе уже есть (кроме логина нужно предоставить еще дополнительную аутентификацию). *Неточность 3* состоит в том, что метрика не определяет, какой именно уровень привилегий требуется при дополнительной аутентификации.

К метрикам группы *Impact* относятся: *Confidentiality Impact* (влияние на конфиденциальность), *Integrity Impact* (влияние на целостность) и *Availability Impact* (влияние на доступность). *Confidentiality Impact* определяет ущерб конфиденциальности в результате успешной эксплуатации уязвимости. *Integrity Impact* задает ущерб целостности после успешной эксплуатации уязвимости. *Availability Impact* определяет ущерб доступности в результате успешной эксплуатации уязвимости. Увеличение ущерба для

любого из трех свойств безопасности увеличивает оценку уязвимости. При этом *неточность 4* состоит в том, что данные метрики не учитывают область действия уязвимости.

CVSS версии 3.0

Важным требованием к CVSS является то, что определение характеристик уязвимостей должно быть понятным для любого эксперта и однозначным. Ряд неточностей, возникающих при применении формата версии 2.0, был устранен в новой версии.

CVSS версии 3.0 [22], как и предыдущая версия, включает две группы метрик: *Exploitability* и *Impact*. Метрики группы *Exploitability* отображают характеристики уязвимого компонента. К ним относятся: *Attack Vector* (вектор атаки), *Attack Complexity* (сложность атаки), *Privileges Required* (требуемые привилегии), *User Interaction* (взаимодействие с пользователем). Метрики группы *Impact* отображают последствия атаки на уязвимый компонент. К ним относятся: *Confidentiality Impact* (влияние на конфиденциальность), *Integrity Impact* (влияние на целостность), *Availability Impact* (влияние на доступность).

Самым важным отличием нового формата является то, что была дополнительно добавлена метрика *Scope* (область действия). Данная метрика позволяет отделить уязвимый компонент (компонент, содержащий уязвимость, например, программное обеспечение, модуль, драйвер и др.) от компонента, которому наносится ущерб (программное обеспечение, аппаратное обеспечение или сетевой ресурс).

Группа метрик *Impact* не изменилась по сравнению с предыдущей версией (тем не менее влияние теперь определяется по области действия *Scope* с учетом максимального воздействия). В группе метрик *Exploitability* метрика *Authentication* заменена на *Privileges Required* и добавлена метрика *User Interaction* (взаимодействие с пользователем раньше учитывалось при определении оценки метрики *Access Complexity*).

Детальное сравнение метрик CVSS версий 2.0 и 3.0 приведено в табл. 1. Сравнение показало, что все метрики были изменены в той или иной степени. В табл. 1 отдельно выделены серым цветом сильно измененные метрики/значения (например, вновь добавленные или удаленные). Остальные метрики подверглись небольшим изменениям (например, численных значений). Рассмотрим эти изменения подробнее.

Важность метрики *Scope* объясняется тем, что в CVSS версии 2.0 не была явно определена область действия ущерба, наносимого уязвимостью. Метрика *Scope* решает эту проблему, четко ограничивая область влияния и устраняя таким образом *неточность 4*. *Scope* относится к набору привилегий, определяемому при назначении доступа ресурсам (файлам, ЦПУ, памяти и т. п.). Когда уязвимость компонента программного обеспечения, управляемая одной областью полномочий, может влиять на ресурсы, управляемые другой областью

■ ***Таблица 1***. Сравнение метрик CVSS версий 2.0 и 3.0

■ ***Table 1***. Comparison of the metrics of CVSS of version 2.0 and CVSS of version 3.0

CVSS версии 2.0	CVSS версии 3.0
Группа *Exploitability*	
Access Vector (*AV*)	*Attack Vector* (*AV*)
Значения *AV*	Значения *AV*
Local (L): 0.395 — для эксплуатации уязвимости требуется локальный доступ к хосту	Local (L): 0.55 — атакующему необходимы права чтение/запись/запуск, чтобы эксплуатировать уязвимость. То есть атакующий должен либо быть залогинен в системе, либо положиться на взаимодействие с пользователем
	Physical (P): 0.2 — требует физических манипуляций с уязвимым компонентом
Adjacent Network (A): 0.646 — для эксплуатации уязвимости требуется доступ к смежной сети	Adjacent (A): 0.62 — для эксплуатации уязвимости требуется доступ к смежной сети
Network (N): 1.0 — для эксплуатации уязвимости требуется сетевой доступ	Network (N): 0.85 — для эксплуатации уязвимости требуется сетевой доступ
Access Complexity (*AC*)	*Attack Complexity* (*AC*)
Значения *AC*	Значения *AC*
High (H): 0.35 — высокая сложность эксплуатации уязвимости	High: 0.44 — успех атаки зависит от условий вне контроля атакующего
Medium (M): 0.61 — средняя сложность эксплуатации уязвимости	

■ ***Окончание табл. 1***

■ ***Table 1.***

CVSS версии 2.0	CVSS версии 3.0
Low (L): 0.71 — низкая сложность эксплуатации уязвимости	Low: 0.77 — нет особых условий доступа. Атакующий может ожидать повторяемого успеха против уязвимого компонента
Authentication (*Au*)	*Privileges Required* (*PR*)
Значения *Au*	Значения *PR*
Multiple (M): 0.45 — для эксплуатации уязвимости требуется дополнительно пройти множество процедур аутентификации	High: 0.27 (0.5, если изменилось значение *Scope*) — атакующий авторизован и имеет привилегии, дающие значительный (административный) доступ к уязвимому компоненту, что может повлиять на настройки и файлы всей системы
Single (S): 0.56 — для эксплуатации уязвимости требуется дополнительно пройти одну процедуру аутентификации	Low: 0.62 (0.68, если изменилось значение *Scope*) — атакующий авторизован и имеет привилегии, дающие базовые пользовательские возможности, которые влияют только на файлы и настройки данного пользователя или только на неконфиденциальные ресурсы
None (N): 0.704 — для эксплуатации уязвимости дополнительной аутентификации не требуется	None: 0.85 — атакующий не авторизован, т. е. доступа к настройкам и файлам не требуется
–	*User Interaction* (*UI*)
	Значения *UI*
	None: 0.85 — система может быть скомпрометирована без участия пользователя
	Required: 0.62 — пользователь должен совершить какие-то действия до того, как уязвимость будет проэксплуатирована. Например, успешный эксплоит возможен только в случае установки приложения системным администратором
Группа *Impact*	
Confidentiality Impact (*C*)	*Confidentiality Impact* (*C*)
Значения *C*	Значения *C*
None (N): 0.0 — нет ущерба	None (N): 0
Partial (P): 0.275 — частичный ущерб	Low (L): 0.22
Complete (C): 0.660 — полный ущерб	High (H): 0.56
Integrity Impact (*I*)	*Integrity Impact* (*I*)
Значения *I*	Значения *I*
None (N): 0.0	None (N): 0
Partial (P): 0.275	Low (L): 0.22
Complete (C): 0.660	High (H): 0.56
Availability Impact (*A*)	*Availability Impact* (*A*)
Значения *A*	Значения *A*
None (N): 0.0	None (N): 0
Partial (P): 0.275	Low (L): 0.22
Complete (C): 0.660	High (H): 0.56
–	*Scope* (*S*)
	Значения *S*
	Unchanged (U) — уязвимость влияет только на ресурсы в рамках привилегий уязвимого компонента: уязвимый компонент и подверженный влиянию компонент — один и тот же
	Changed (C) — уязвимость влияет на ресурсы вне привилегий уязвимого компонента, в этом случае уязвимый компонент и подверженный влиянию компонент — разные

■ ***Таблица 2.*** Формулы CVSS в версии 2.0 и версии 3.0

■ ***Table 2.*** The equations of CVSS of version 2.0 and CVSS of version 3.0

CVSS версии 2.0	CVSS версии 3.0
$CVSS\_Score =$ $= \text{round\_to\_1\_decimal}(((0.6 \times Impact) +$ $+ (0.4 \times Exploitability) - 1.5) \times f(Impact))$	Если ($Impact <= 0$) $CVSS\_Score = 0$, иначе если ($Scope$ = Unchanged) $CVSS\_Score =$ $= \text{Roundup}(\min[(Impact + Exploitability), 10])$, иначе если ($Scope$ = Changed) $CVSS\_Score =$ $= \text{Roundup}(\min[1.08 \times (Impact + Exploitability), 10])$
$Impact = 10.41 \times (1 - (1 - C) \times (1 - I) \times (1 - A))$	Если ($Scope$ = Unchanged) $Impact = 6.42 \times ISC_{Base}$, иначе если ($Scope$ = Changed) $Impact =$ $= 7.52 \times (ISC_{Base} - 0.029) - 3.25 \times (ISC_{Base} - 0.02)^{15}$
	$ISC_{Base} = 1 - [(1 - C) \times (1 - I) \times (1 - A)]$
$Exploitability = 20 \times AV \times AC \times Au$	$8.22 \times AV \times AC \times PR \times UI$
Если ($Impact = 0$) $f(Impact) = 0$, иначе $f(Impact) = 1.176$	–
round_to_1_decimal — функция округления до одного десятичного знака после запятой в большую сторону	Roundup — округление до одного десятичного знака после запятой в большую сторону

полномочий, происходит смена *Scope*. В качестве примера здесь можно привести уязвимость виртуальной машины, которая позволяет атакующему удалять файлы на хостовой операционной системе, возможно, даже саму виртуальную машину. Базовая оценка растет в случае смены *Scope*.

Метрика *Attack Vector* сохранилась из предыдущей версии, но изменились ее возможные значения. В CVSS версии 3.0 физический доступ выделен в отдельное значение, что устраняет *неточность 1* (путаницу между локальным и физическим доступом). Значение метрики тем больше, чем более удаленный (логически и физически) атакующий может ее использовать (удаленных атакующих намного больше, чем тех, у кого есть физический доступ к устройству).

Метрика *Attack Complexity* унаследована из CVSS версии 2.0 от метрики *Access Complexity*. Однако она больше не учитывает взаимодействие с пользователем и изменились ее значения (см. табл. 1).

Заменившая метрику *Authentication* метрика *Privileges Required* определяет уровень привилегий, необходимый атакующему до того, как уязвимость будет успешно проэксплуатирована. Значение данной метрики выше всего, если не требуется никаких привилегий. Она снимает *неточность 3*, позволяя связать постусловия эксплуатации уязвимости на одном хосте с предусловиями на другом.

Новая метрика *User Interaction* определяет требования к пользователю, отличному от атакующего, необходимые для успешной компрометации уязвимого компонента. Метрика определяет, может ли уязвимость эксплуатироваться без участия атакуемого пользователя. Например, для успешного проведения атаки может требоваться, чтобы жертва открыла в браузере сформированную атакующим вредоносную ссылку. Метрика имеет наибольшее значение, когда не требуется взаимодействие с пользователем. Введение данной метрики устраняет *неточность 2*.

Описанные метрики используются для вычисления общей оценки CVSS для уязвимостей. Для этой цели в формате CVSS приведены специальные формулы. В связи с изменениями метрик CVSS и их значений формулы в CVSS 3.0 были также изменены. Формулы CVSS версии 2.0 и версии 3.0 приведены в табл. 2 для сравнения (обозначения метрик взяты из табл. 1).

Таким образом, изменения, внесенные в CVSS версии 3.0, снимают многие проблемы, возникавшие ранее. В следующем разделе рассматривается влияние данных изменений на процесс построения и анализа графа атак.

Графы атак

Формирование графов атак на основе формата CVSS версии 2.0

Общий алгоритм построения графов атак состоит из трех шагов:

1) формирование матриц по базам уязвимостей и конфигурации программно-аппаратного обеспечения хостов;

2) формирование списков доступных нарушителям атакующих действий;

3) генерация графа атак на основе графа связей сети и списков атакующих действий.

Рассмотрим данные шаги более подробно.

Шаг 1. Для построения графов атак формируется список возможных атакующих действий, разбитых на группы в соответствии с метриками

CVSS версии 2.0. Для этого для каждого хоста из сети строится трехмерная матрица по следующим данным:

1) класс атак (сбор данных, подготовительные действия, повышение привилегий, выполнение цели атаки) — определяется на основе используемых баз (база атак CAPEC [23] или база уязвимостей CVE) с учетом влияния на конфиденциальность (*Confidentiality Impact*), целостность (*Integrity Impact*), доступность (*Availability Impact*) и получаемых прав (*Gained Access Level*);

2) тип доступа (удаленный источник без прав доступа, удаленный пользователь системы, локальный пользователь системы, администратор) — определяется на основе вектора доступа (*Access Vector*) и аутентификации (*Authentication*);

3) уровень знаний нарушителя (типы уязвимостей, которые нарушитель сможет реализовывать) — определяется на основе сложности доступа (*Access Complexity*).

При этом ячейки матрицы (т. е. пересечения класса атаки, типа доступа и уровня знаний нарушителя) являются списками уязвимостей, соответствующих этим параметрам. После формирования самой матрицы ее ячейки заполняются конкретными атакующими действиями на основе списков существующих уязвимостей, соответствующих конфигурации программно-аппаратного обеспечения хоста, и атак, направленных на сбор информации. Списки возможных атак ограничиваются параметрами безопасности хоста (т. е. ограничениями на список возможных уязвимостей и атак, направленных на сбор информации). В результате для каждого хоста формируется список возможных атакующих действий, разбитых на группы по следующим параметрам: класс атаки, необходимый тип доступа и необходимый уровень знаний нарушителя. Для каждой группы в свою очередь формируется список конкретных атак и уязвимостей, которые эти атаки реализуют. Например, уязвимость CVE-2016-10108 позволяет получить права администратора (*Gained Access Level* = “administrator”) на некоторых версиях Western Digital MyCloud NAS. Эта уязвимость (или атакующее действие, реализующее данную уязвимость) относится к классам атак «повышение привилегий» и «выполнение цели атаки», может быть проэксплуатирована удаленно (*Access Vector* = “Network”) и не требует предварительного получения прав доступа (*Authentication* = “None”) и знаний нарушителя (*Access Complexity* = “Low”). Таким образом, данная уязвимость относится к группе «повышение привилегий» (класс атаки): удаленный источник без прав доступа (тип доступа), не требуется специализированных знаний (уровень знаний нарушителя).

Кроме отдельных уязвимостей при построении графа атак используются шаблоны атак в формате CAPEC, которые могут выступать не только в качестве входной информации для построения графов атак, но и как результат анализа безопасности — они могут описывать наиболее часто встретившиеся последовательности эксплуатаций уязвимостей и других действий атакующего. Также шаблоны содержат описания атак, которые не используют уязвимости, например, первая стадия проведения атаки — это сбор информации о доступных хостах. Для этого применяется шаблон CAPEC-292 (Host Discovery), описывающий группу различных способов проведения сканирования хостов и портов. Следующая стадия атаки — поиск уязвимого программного обеспечения. Для этого используются следующие шаблоны: CAPEC-310 (Scanning for Vulnerable Software), CAPEC-311 (Fingerprinting Remote Operating Systems), CAPEC-300 (Port Scanning) и т. д. На третьей стадии проведения атаки используются как отдельные уязвимости из словаря CVE, так и шаблоны, например CAPEC-233 (Privilege Escalation) и т. д.

Метрики CVSS *Access Vector*, *Authentication* и *Access Complexity* являются предусловиями эксплуатации уязвимостей, т. е. предусловиями, необходимыми для успешной реализации атаки, и помимо формирования графа применяются при оценке защищенности компьютерной сети для определения вероятности успешной реализации атаки [23, 24]. Метрики *Confidentiality Impact*, *Integrity Impact*, *Availability Impact* являются постусловиями эксплуатации уязвимостей, т. е. постусловиями успешной реализации атаки, и помимо формирования графа применяются при оценке защищенности компьютерной сети для определения ущерба в результате реализации атаки [23, 24].

Шаг 2. После формирования матриц возможных атакующих действий для каждого хоста анализируемой сети на основе уровня знаний нарушителя выбираются атакующие действия, доступные конкретной модели нарушителя. На данном этапе могут быть использованы сразу несколько моделей нарушителей.

Далее на основе анализа связей компьютерной сети и множества атакующих действий, ограниченного возможностями нарушителя, формируется граф доступности хостов одновременно для всех нарушителей.

Шаг 3. На основе графов доступности формируются графы атак для начальных точек доступа, доступных каждому нарушителю. Данный шаг включает ряд действий для каждого нарушителя. Рассмотрим данные действия более подробно.

Шаг 3.1. Формирование множества хостов, к которым есть доступ у нарушителя, в соответствии с исходными данными.

Шаг 3.2. Получение максимально возможных привилегий на каждом доступном хосте на ос-

нове использования доступных атакующих действий (на основе анализа поля *Gained Privileges* уязвимостей соответствующей группы).

Шаг 3.3. Выполнение атакующих действий, направленных на нарушение конфиденциальности, целостности и доступности информации, хранящейся на хосте. Если для нарушителя доступны только права пользователя в соответствии с метриками *Access Vector* и *Authentication*, то атакующие действия ограничиваются доступными только локальным и удаленным пользователям. Влияние определяется на основе анализа полей *Confidentiality Impact*, *Integrity Impact* и *Availability Impact* уязвимостей соответствующей группы.

Шаг 3.4. Для каждого доступного хоста, на котором нарушитель может получить права администратора, составляется список обнаруженных связанных с ним хостов, для которых возможно проведение атаки сбора информации.

Шаг 3.5. Составление списка обнаруженных связанных хостов, для которых нарушитель может определить конфигурацию программно-аппаратного обеспечения.

Шаг 3.6. Выполнение шага 3.2 для полученного на шаге 3.4 списка. Каждое действие шагов 3.2–3.4 алгоритма добавляет новые атакующие действия, относящиеся к выбранной модели нарушителя, в граф атак.

При этом для каждого хоста формируется направленный граф использования уязвимостей, определяющий возможные последовательности эксплуатации уязвимостей нарушителем. Так, в качестве первого шага нарушитель может реализовать атаки, которые не требуют наличия локального доступа и учетной записи и направленные на: 1) нарушение конфиденциальности, целостности и доступности информации; 2) получение прав доступа учетной записи пользователя системы; 3) получение прав доступа учетной записи администратора системы.

Далее если нарушитель получил доступ к любой учетной записи, он может выполнять атаки, направленные на нарушение конфиденциальности, целостности и доступности информации, которые требуют наличия локального доступа. Если нарушитель получил доступ к учетной записи пользователя, он может повысить свой уровень доступа до администраторского с помощью атак, направленных на повышение привилегий. Далее при наличии прав администратора нарушитель может выполнять любые атаки, направленные на нарушение конфиденциальности, целостности и доступности информации.

В результате работы данного алгоритма для каждого нарушителя формируется граф связанных хостов, включающий в себя множество пересекающихся графов, начинающихся от начальных хостов нарушителя и включающих в себя подграфы эксплуатируемых уязвимостей и атакующие действия, направленные на сбор информации. При этом каждый хост в графе характеризуется уровнем нарушения свойств конфиденциальности, целостности и доступности, а также правами доступа, полученными нарушителем в результате эксплуатации уязвимостей.

Формирование графов атак на основе формата CVSS версии 3.0

Основными недостатками подхода к построению графов атак, представленного в предыдущем разделе, являются неточности при использовании описаний уязвимостей. Так, например, ущерб конфиденциальности, целостности и доступности определялся по метрикам уязвимости группы *Impact*. Но эти метрики не определяют область влияния уязвимости (информация в приложении, информация в операционной системе или вся информация на жестком диске). В построенный граф также попадали уязвимости, требующие активных действий от атакуемого хоста (например, перехода по вредоносной ссылке), что не всегда возможно (например, использование таких атак невозможно против серверных хостов). Данные проблемы обусловливают необходимость перехода на стандарт описания уязвимостей CVSS версии 3.0.

Общая структура алгоритма построения графов атак при переходе от CVSS версии 2.0 к версии 3.0 практически не меняется. Но использование стандарта CVSS версии 3.0 позволяет уточнить построенные графы атак, что приводит к повышению обоснованности построенной модели и, как следствие, повышает точность оценки защищенности компьютерной сети [25, 26].

Изменения коснутся шага 1 алгоритма, представленного в предыдущем подразделе, так как изменятся как сформированные на данном шаге группы, так и значения соответствующих метрик CVSS. Вследствие этого изменятся результаты, получаемые на шаге 3 алгоритма.

Поскольку определяемый класс атак учитывает метрики *Confidentiality Impact*, *Integrity Impact* и *Availability Impact*, а также получаемые привилегии, изменятся формируемые классы. Это обусловлено изменением значения метрик ущерба (что повлияет также на численные оценки уровня ущерба и уровня риска при оценке защищенности). Кроме того, появилась метрика *Scope*, на основе которой уточняется область влияния уязвимостей (приложение, операционная система, песочница) и определяется возможный доступ к ресурсам (файлам, ЦПУ, памяти и т. п.). Это приведет к тому, что для ряда атак, для которых *Scope* является неизменным и не затрагивает системные ресурсы, получение прав

не приведет к получению прав на хосте. Кроме того, будет уточнена область наносимого ущерба. Как следствие, меняется результат шага 3 (шаги 3.2, 3.3) алгоритма и список доступных хостов, формируемый на шаге 3 (шаг 3.4) алгоритма. Использование метрики *Scope* не меняет общую структуру алгоритма, но уточняет результаты анализа последствий атакующих действий на компьютерную сеть.

При формировании необходимого типа доступа и знаний нарушителя на шаге 1 алгоритма используются метрики CVSS версии 2.0 *Access Vector*, *Authentication* и *Access Complexity*.

Метрика *User Interaction*, появившаяся в CVSS версии 3.0, определяет, может ли уязвимость эксплуатироваться без участия атакуемого. На основе данной метрики и типа хостов выделяются группы уязвимостей в матрице уязвимостей на шаге 1 алгоритма, приведенного в предыдущем подразделе. Так, уязвимости, не требующие участия атакуемого, могут быть эксплуатированы без ограничений. Возможность эксплуатации уязвимостей, требующих участия атакуемого, определяется на основе дополнительных параметров хостов, задаваемых оператором. По умолчанию данный тип уязвимостей не может эксплуатироваться для серверных хостов. Для пользовательских хостов данные уязвимости по умолчанию считаются доступными.

Для метрики *Access Vecto*r изменились возможные значения, что, с одной стороны, повлияет на связи в графе, так как выделилась отдельная категория физического доступа, и ряд уязвимостей уйдет из графа, с другой стороны, изменится значение вероятности успешного выполнения атаки, применяемого при оценке защищенности.

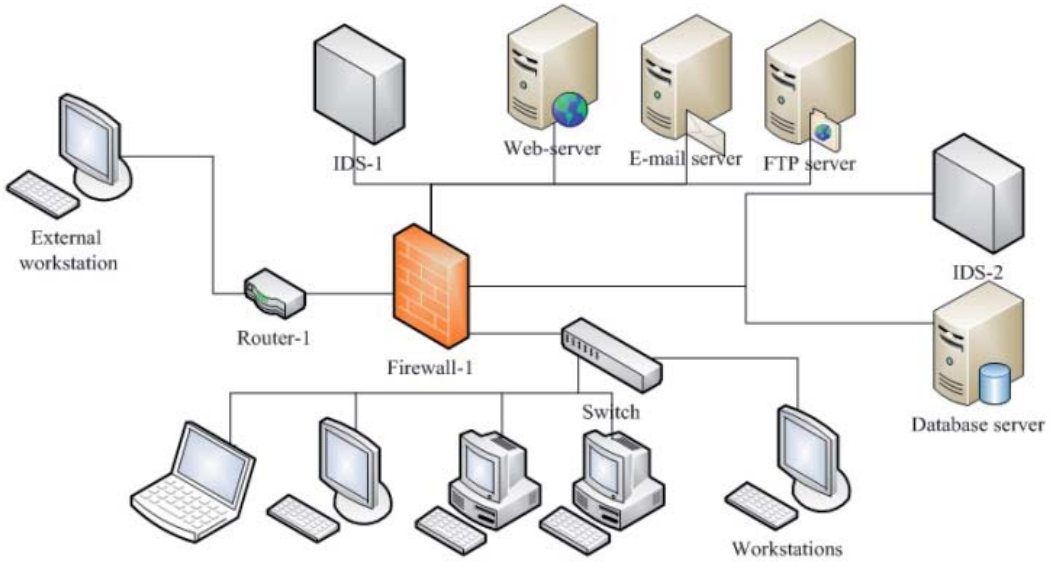
Метрика *Authentication* была заменена на метрику *Privileges Required*, что позволит более точно формировать список доступных атакующих действий при переходах между хостами и внутри хоста, а также повлияет на значение вероятности успешного выполнения атаки.

Пример применения

Рассмотрим влияние перехода от CVSS версии 2.0 к CVSS версии 3.0 на процесс генерации графа атак и оценки защищенности.

На рисунке представлен пример сети, включающей: веб-сервер Web-server (с системой Windows Server 2008 R2 (64 бит), JBoss AS 5.0.1, фреймворком ApacheStruts2); сервер баз данных Database server (с системой Windows Server 2008 R2 (64 бит), MS SQL Server 2008 R2, CA Spectrum 9.2, EMC Unisphere для VMAX 8.1); почтовый сервер E-mail server (с системой SUSE Enterprise Linux 11 SP1 (32 бит), почтовым сервером Postfix, почтовым сервером Dovecot, MySQL); FTP сервер FTP-server (с системой Windows Server 2008 R2 (64 бит), Ipswitch WS_FTP Server 6.1.0.0); межсетевой экран Firewall-1 (с системой Novell SUSE Linux Enterprise Server 11.0 Service Pack 3 Long Term Service Pack Support, Netfilter); рабочие станции Workstations (с системой Microsoft Windows 7 64-bit, Apple ITunes 9.0.3, Microsoft Office 2007 SP1, Microsoft Internet Explorer 7).

На небольшом фрагменте графа (примере атак) рассмотрим влияние CVSS версии 3.0.



- Пример компьютерной сети
- Test network

Удаленный пользователь (атакующий) имеет удаленный доступ к Firewall-1, на котором установлена операционная система Novell SUSE Linux Enterprise Server 11.0 Service Pack 3 Long Term Service Pack Support. В этой системе присутствует уязвимость CVE-2016-4448. Данной уязвимости присвоена как оценка CVSS версии 2.0 (10.0), так и оценка CVSS версии 3.0 (9.8). Соответствующие метрики CVSS версии 2.0 и их значения: *Access Vector* = "Network", *Access Complexity* = "Low", *Authentication* = "None", *Confidentiality/Integrity* и *Availability Impact* = = "Complete". Соответствующие метрики CVSS версии 3.0 и их значения: *Attack Vector* = "Network", *Attack Complexity* = "Low", *Privileges Required* = "None", *User Interaction* = "None", *Scope* = "Unchanged", *Confidentiality/Integrity* и *Availability Impact* = "High". С точки зрения генерации графа атак предусловия эксплуатации уязвимости не изменились. В обоих случаях требуется сетевой уровень доступа, не требуется дополнительных привилегий и сложность атаки низкая. Отличием является то, что в случае применения CVSS версии 3.0 очевиднее, что атакующему не требуются дополнительные привилегии на хосте и не требуется взаимодействовать с пользователем для эксплуатации уязвимости. Постусловия эксплуатации уязвимости также не изменились. В обоих случаях атакующий получает привилегии администратора и может продолжить атаковать следующие доступные хосты сети, а влияние на свойства безопасности остается высоким (поскольку значение метрики *Scope* не изменилось, она не влияет на постусловия).

Далее рассмотрим, как повлияют изменения в CVSS версии 3.0 на результаты оценки защищенности, на примере упрощенного вычисления вероятности атаки (без учета предыдущих шагов атаки). Вероятность атаки с применением CVSS версии 2.0 вычислялась с использованием *Exploitability subscore* [25, 26]. Для выбранной уязвимости *Exploitability subscore* определяется следующим образом: $2 \times AV \times AC \times Au = 1.0$ (см. табл. 2). В случае CVSS версии 3.0 максимальное значение *Exploitability subscore* 3.9, а минимальное — 0.2. Чтобы получить значение между 0 и 1.0, вычтем 0.2, поделим полученное значение на 10 и умножим на 2.7. Тогда вероятность успеха атакующего действия, использующего выбранную уязвимость, будет вычисляться с применением CVSS версии 3.0 следующим образом: $(8.22 \times AV \times \times AC \times PR \times UI - 0.2) \times 2.7/10 = 1.0$ (см. табл. 2). Полученный результат совпадает с результатом согласно CVSS версии 2.0. Ущерб от атаки при использовании CVSS версии 2.0 вычислялся на основе метрик группы *Impact* с предположением, что ущерб распространяется только на уязвимый компонент [25, 26]. Поскольку для выбранной уязвимости значение метрики *Scope* = "Unchanged", подверженный влиянию компонент совпадает для обеих версий. В обоих случаях ущерб является высоким. Но согласно CVSS версии 2.0 значение ущерба для всех свойств безопасности количественно равно 0.66, а для CVSS версии 3.0 — 0.56. Значение *Impact subscore* по CVSS версии 2.0 равно 10.0, а по CVSS версии 3.0 — 5.9 (что не является максимальным значением данной метрики).

После компрометации межсетевого экрана Firewall атакующий может обнаружить другие хосты сети, например Database server. На данном сервере установлено программное обеспечение EMC Unisphere для VMAX 8.1, которое имеет уязвимость CVE-2016-6645. Данной уязвимости назначена как оценка CVSS версии 2.0 (9.0), так и оценка CVSS версии 3.0 (8.8). Соответствующие метрики и их значения согласно CVSS версии 2.0: *Access Vector* = "Network", *Access Complexity* = = "Low", *Authentication* = "Single", *Confidentiality/ Integrity* и *Availability Impact* = "Complete". Соответствующие метрики и их значения согласно CVSS версии 3.0: *Attack Vector* = "Network", *Attack Complexity* = "Low", *Privileges Required* = "Low", *User Interaction* = "None", *Scope* = "Unchanged", *Confidentiality/Integrity* и *Availability Impact* = = "High". С точки зрения генерации графа атак предусловия остаются одинаковыми в случае применения обеих версий: в обоих случаях для эксплуатации уязвимости требуется сетевой доступ и привилегии на хосте. Единственным отличием является то, что в случае применения CVSS версии 3.0 очевиднее, что атакующему требуются привилегии на хосте, дающие базовые пользовательские возможности, которые влияют только на файлы и настройки данного пользователя. Это означает, что атакующий не может сразу реализовать данное атакующее действие, так как он вначале должен получить привилегии пользователя на хосте. В отличие от CVSS версии 2.0, в CVSS версии 3.0 понятнее связь между полученными и требуемыми привилегиями. В этом примере сложность атаки низкая (но в случае применения CVSS версии 3.0 очевиднее, что атакующему не требуется взаимодействовать с пользователем для эксплуатации уязвимости). Постусловия эксплуатации уязвимости также не изменились. В обоих случаях атакующий получает привилегии администратора и может продолжить атаковать следующие доступные хосты сети, а влияние на свойства безопасности остается высоким (поскольку значение метрики *Scope* не изменилось, она не влияет на постусловия).

Далее рассмотрим, как повлияют изменения в CVSS версии 3.0 на процесс оценки защищенности. Вероятность атаки с применением CVSS версии 2.0 для выбранной уязвимости вычислялась следующим образом: $2 \times AV \times AC \times Au = 0.8$ (см.

табл. 2). В случае применения CVSS версии 3.0 вероятность успеха атакующего действия, использующего выбранную уязвимость, будет вычисляться следующим образом: $(8.22 \times AV \times AC \times \times PR \times UI - 0.2) \times 2.7/10 = 0.7$ (см. табл. 2). В данном случае результат при применении CVSS версии 2.0 выше, чем результат при применении CVSS версии 3.0. Поскольку для выбранной уязвимости значение метрики *Scope* = "Unchanged", подверженный влиянию компонент совпадает для обеих версий. В обоих случаях ущерб является высоким. Однако по CVSS версии 2.0 значение ущерба для всех свойств безопасности равно 0.66, а по CVSS версии 3.0 — 0.56. *Impact subscore* по CVSS версии 2.0 равен 10.0, а по CVSS версии 3.0 — 5.9 (что не является максимальным значением данной метрики).

Таким образом, хотя CVSS версии 3.0 принципиально не влияет на алгоритм построения графа, она позволяет снять некоторые неточности и допущения. Хотя она в то же время создает некоторые дополнительные сложности для процесса оценки защищенности.

Описанный в данной статье подход был реализован как приложение на языке Java. На сегодня полный переход к CVSS версии 3.0 в приложении невозможен, так как, во-первых, оценки по CVSS версии 3.0 существуют только для новых уязвимостей, и, во-вторых, в настоящий момент файл .xml данных CVSS версии 3.0 на сайте NVD [21] отсутствует.

Заключение

В статье проведен анализ изменений, введенных в новую версию системы оценки уязвимостей CVSS, а также влияния этих изменений на предложенный нами алгоритм формирования графа атакующих действий. Применение изменений при генерации графа атак и оценке защищенности показано на примере. На основе проведенного анализа сделан вывод, что применение CVSS версии 3.0 позволит устранить многие неточности, существовавшие ранее, хотя и не все. В настоящий момент невозможно автоматизировать применение нового стандарта, но в будущем планируется использовать его в приложении, разработанном авторами, наравне с CVSS версии 2.0. Кроме того, в последующей работе планируется продолжить улучшение процесса генерации графов атак с точки зрения применения шаблонов атак и дальнейшего автоматизированного выбора защитных мер.

Работа выполнена при финансовой поддержке РФФИ (проекты № 15-07-07451, 16-37-00338, 16-29-09482 офи_м), гранта Президента РФ № МК-314.2017.9 и при частичной поддержке бюджетных тем № 0073-2015-0004 и 0073-2015-0007 в СПИИРАН.

Литература

1. **Artz M.** NetSPA, a Network Security Planning Architecture: Master's Thesis. — Massachusetts Institute of Technology, 2002. — 96 p.
2. **Lippmann R. P.** Validating and Restoring Defense in Depth using Attack Graphs // Proc. of MILCOM 2006. Washington, DC. P. 1–10.
3. **Ingols K., Lippmann R., Piwowarski K.** Practical Attack Graph Generation for Network Defense//Proc. of 22nd Annual Conf. on the Computer Security Applications, Miami Beach, FL. IEEE, 2006. P. 121–130.
4. **Singhal A. Ou X.** Security Risk Analysis of Enterprise Networks using Probabilistic Attack Graphs: NIST Interagency Report 7788. — Gaithersburg: National Institute of Standards and Technology, 2011. — 24 p.
5. **Man D., Yang W., Yang Y., Wang W., Zhang L.** A Quantitative Evaluation Model for Network Security // Proc. of the 2007 Intern. Conf. on Computational Intelligence and Security. Dec. 2007. P. 773–777.
6. **Wu Y.-S., Foo B., Mao Y.-C., Bagchi S., Spafford E. H.** Automated Adaptive Intrusion Containment in Systems of Interacting Services // The Intern. Journal of Computer and Telecommunications Networking. 2007. Vol. 51. P. 1334–1360.
7. **Stakhanova N., Basu S., Wong J.** A Cost-Sensitive Model for Preemptive Intrusion Response Systems // Proc. of the 21st Intern. Conf. on Advanced Networking and Applications. 2007. P. 1–8.
8. **Liu Y., Man Y.** Network Vulnerability Assessment using Bayesian Networks // Proc. of the SPIE. 2005. Vol. 5812. P. 61–71.
9. **Frigault M., Wang L., Singhal A., Jajodia S.** Measuring Network Security using Dynamic Bayesian Network //Proc. of the ACM Workshop on Quality of Protection. October 2008. P. 23–30.
10. **Dantu R., Kolan P., Cangussu J., Dantu R., Kolan P.** Network Risk Management using Attacker Profiling // Security and Communication Networks. 2009. Vol. 2. N 1. P. 83–96.
11. **Wang L., Islam T., Long T., Singhal A., Jajodia S.** An Attack Graph-Based Probabilistic Security Metric //Proc. of the 22nd Annual IFIP WG 11.3 Working Conf. on Data and Applications Security, Heidelberg. Springer-Verlag Berlin, 2008. P. 283–296.
12. **Poolsappasit N., Dewri R., Ray I.** Dynamic Security Risk Management using Bayesian Attack Graphs // IEEE Transactions on Dependable and Security Computing. 2012. Vol. 9. N 1. P. 61–74.
13. **Kotenko I., Chechulin A.** Computer Attack Modeling and Security Evaluation based on Attack Graphs // Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2013):

Proc. of the IEEE 7th Intern. Conf. Berlin, Germany, September 2013. P. 614–619.
14. **Kotenko I., Chechulin A.** A Cyber Attack Modeling and Impact Assessment Framework // Proc. of the 5th Intern. Conf. on Cyber Conflict 2013 (CyCon 2013). Tallinn, Estonia: IEEE and NATO COE Publications, June 2013. P. 119–142.
15. **Дойникова Е. В., Котенко И. В.** Методики и программный компонент оценки рисков на основе графов атак для систем управления информацией и событиями безопасности // Информационно-управляющие системы. 2016. № 5. С. 54–65. doi:10.15217/issn1684-8853.2016.5.54
16. **Chechulin A., Kotenko I.** Attack Tree-based Approach for Real-Time Security Event Processing // Automatic Control and Computer Sciences. 2015. Vol. 49. N 8. P. 701–704.
17. Common Vulnerability Scoring System (CVSS-SIG). FIRST website. https://www.first.org/cvss (дата обращения: 18.09.2017).
18. Common Platform Enumeration (CPE). NVD website. https://nvd.nist.gov/cpe.cfm (дата обращения: 18.09.2017).
19. Common Configuration Enumeration (CCE). NVD website. https://nvd.nist.gov/cce/index.cfm (дата обращения: 18.09.2017).
20. **Mell P., Scarforne K., Romanosky S.** A Complete Guide to the Common Vulnerability Scoring System Version 2.0 (CVSS). 2007. — 23 p. https://www.first.org/cvss/v2/guide (дата обращения: 18.09.2017).
21. NVD website. https://nvd.nist.gov (дата обращения: 18.09.2017).
22. Common Vulnerability Scoring System v3.0: Specification Document. FIRST Org. Inc, 2015. — 21 p. https://www.first.org/cvss/specification-document (дата обращения: 18.09.2017).
23. **Barnum S.** Common Attack Pattern Enumeration and Classification (CAPEC). — Schema Description, 2008. — 26 p.
24. Common Vulnerabilities and Exposures (CVE). http://cve.mitre.org. (дата обращения: 18.09.2017).
25. **Kotenko I., Doynikova E.** Dynamical Calculation of Security Metrics for Countermeasure Selection in Computer Networks // Proc. of the 24th Euromicro Intern. Conf. on Parallel, Distributed and Network-based Processing (PDP 2016), Heraklion, Crete, Greece, Feb. 2016, Los Alamitos, California. IEEE Computer Society, 2016. P. 558–565.
26. **Котенко И. В., Дойникова Е. В.** Методика выбора контрмер в системах управления информацией и событиями безопасности // Информационно-управляющие системы. 2015. № 3. С. 60–69. doi:10.15217/issn1684-8853.2015.3.60

UDC 004.056
doi:10.15217/issn1684-8853.2017.6.76

Computer Network Security Evaluation based on CVSS Metrics

Doynikova E. V.[a,b], PhD, Tech., Researcher, doynikova@comsec.spb.ru
Chechulin A. A.[a,b], PhD, Tech., Leading Researcher, chechulin@comsec.spb.ru
Kotenko I. V.[a,b], Dr. Sc., Tech., Professor, ivkote@comsec.spb.ru
[a]Saint-Petersburg Institute for Informatics and Automation of the RAS, 39, 14 Line, V. O., 199178, Saint-Petersburg, Russian Federation
[b]Saint-Petersburg National Research University of Information Technologies, Mechanics and Optics, 49, Kronverkskii St., 197101, Saint-Petersburg, Russian Federation

Introduction: In modern conditions, an extremely relevant issue is the enhancement of computer networks security assessment for automated defence systems targeted on a preventive response to attacks through the application of objective quantitative security metrics calculated using the metrics of CVSS (Common Vulnerability Scoring System) and analytical models. **Purpose:** The enhancement of the previously developed approach to the security assessment based on analytical modeling, usage of open standards for security data representation and CVSS metrics through the application of new CVSS version. The approach is based on a graph model of malefactor's actions generated with CVSS metrics. Some inaccuracies in the previous CVSS version led to certain limitations in generating the graph of malefactor's actions. We assume that applying the new CVSS version metrics will enhance the graph generation procedure and the assessment of security. **Results:** The security assessment results significantly depend on the input data correctness. To get initial vulnerability scores, we use CVSS (vulnerability assessment metrics, their specification and scoring criteria) and open vulnerability databases. Some disadvantages of metrics specification in CVSS v.2 and differences in CVSS v.3 have been revealed, namely: in order to take into account some essential security characteristics, new qualitative values of metrics were introduced which were neglected before; areas of possible values of the metrics were defined, removing the uncertainty which existed before. A novel approach to attack modeling and computer network security assessment has been developed on the basis of CVSS v.3. The advantages of this approach as compared to the approach proposed by the authors earlier are described, namely: when forming the graph of malefactor's actions, the assumptions about the specification of pre and post conditions of an attack are removed. Examples of applying CVSS v.3 metrics to assess computer network security on the basis of the analytical modeling are given. **Practical relevance:** The proposed approach improves the procedures of attack graph generation and security assessment due to the novel vulnerability assessment format CVSS v.3 in the framework of computer network security assessment software.

Keywords — Analytical Modeling, Security Assessment, Security Metrics, Common Vulnerability Scoring System, Attack Protection, Computer Networks, Attack Graphs, Attack Trees.

References

1. Artz M. *NetSPA, a Network Security Planning Architecture*. Master's thesis. Massachusetts Institute of Technology, 2002. 96 p.
2. Lippmann R. P. Validating and Restoring Defense in Depth using Attack Graphs. *Proc. of MILCOM 2006*, Washington, DC, pp.1–10.
3. Ingols K., Lippmann R., Piwowarski K. Practical Attack Graph Generation for Network Defense. *Proc. of 22nd Annual Conf. on the Computer Security Applications*, Miami Beach, FL, IEEE, 2006, pp. 121–130.
4. Singhal A. Ou X. *Security Risk Analysis of Enterprise Networks using Probabilistic Attack Graphs*. NIST Interagency Report 7788. Gaithersburg, National Institute of Standards and Technology, 2011. 24 p.
5. Man D., Yang W., Yang Y., Wang W., Zhang L. A Quantitative Evaluation Model for Network Security. *Proc. of the 2007 Intern. Conf. on Computational Intelligence and Security*, Dec. 2007, pp. 773–777.
6. Wu Y.-S., Foo B., Mao Y.-C., Bagchi S., Spafford E. H. Automated Adaptive Intrusion Containment in Systems of Interacting Services. *The Intern. Journal of Computer and Telecommunications Networking*, 2007, vol. 51, pp. 1334–1360.
7. Stakhanova N., Basu S., Wong J. A Cost-Sensitive Model for Preemptive Intrusion Response Systems. *Proc. of the 21st Intern. Conf. on Advanced Networking and Applications*, 2007, pp. 1–8.
8. Liu Y., Man Y. Network Vulnerability Assessment using Bayesian Networks. *Proc. of the SPIE*, 2005, vol. 5812, pp. 61–71.
9. Frigault M., Wang L., Singhal A., Jajodia S. Measuring Network Security using Dynamic Bayesian Network. *Proc. of the ACM Workshop on Quality of Protection*, October 2008, pp. 23–30.
10. Dantu R., Kolan P., Cangussu J., Dantu R., Kolan P. Network Risk Management using Attacker Profiling. *Security and Communication Networks*, 2009, vol. 2, no. 1, pp. 83–96.
11. Wang L., Islam T., Long T., Singhal A., Jajodia S. An Attack Graph-Based Probabilistic Security Metric. *Proc. of the 22nd Annual IFIP WG 11.3 Working Conf. on Data and Applications Security*, Heidelberg, Springer-Verlag Berlin, 2008, pp. 283–296.
12. Poolsappasit N., Dewri R., Ray I. Dynamic Security Risk Management using Bayesian Attack Graphs. *IEEE Transactions on Dependable and Security Computing*, 2012, vol. 9, no. 1, pp. 61–74.
13. Kotenko I., Chechulin A. Computer Attack Modeling and Security Evaluation based on Attack Graphs. *Proc. of the IEEE 7th Intern. Conf. "Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications" (IDAACS'2013)*, Berlin, Germany, September 2013, pp. 614–619.
14. Kotenko I., Chechulin A. A Cyber Attack Modeling and Impact Assessment Framework. *Proc. of the 5th Intern. Conf. on Cyber Conflict 2013 (CyCon 2013)*, Tallinn, Estonia, IEEE and NATO COE Publications, June 2013, pp. 119–142.
15. Doynikova E. V., Kotenko I. V. Techniques and Tool for the Risk Assessment on the Base of Attack Graphs in Information and Security Event Management Systems. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2016, no. 5, pp. 54–65 (In Russian). doi:10.15217/issn1684-8853.2016.5.54
16. Chechulin A., Kotenko I. Attack Tree-based Approach for Real-Time Security Event Processing. *Automatic Control and Computer Sciences*, 2015, vol. 49, no. 8, pp. 701–704.
17. *Common Vulnerability Scoring System (CVSS-SIG). FIRST website.* Available at: https://www.first.org/cvss (accessed 18 September 2017).
18. *Common Platform Enumeration (CPE)*. NVD website. Available at: https://nvd.nist.gov/cpe.cfm (accessed 18 September 2017).
19. *Common Configuration Enumeration (CCE)*. NVD website. Available at: https://nvd.nist.gov/cce/index.cfm (accessed 18 September 2017).
20. Mell P., Scarforne K., Romanosky S. A *Complete Guide to the Common Vulnerability Scoring System Version 2.0 (CVSS)*. 2007. 23 p. Available at: https://www.first.org/cvss/v2/guide (accessed 18 September 2017).
21. *NVD website*. Available at: https://nvd.nist.gov (accessed 18 September 2017).
22. *Common Vulnerability Scoring System v3.0: Specification Document*. *FIRST Org. Inc*, 2015. 21 p. Available at: https://www.first.org/cvss/specification-document (accessed 18 September 2017).
23. Barnum S. *Common Attack Pattern Enumeration and Classification (CAPEC)*. Schema Description, 2008. 26 p.
24. *Common Vulnerabilities and Exposures (CVE)*. Available at: http://cve.mitre.org (accessed 18 September 2017).
25. Kotenko I., Doynikova E. Dynamical Calculation of Security Metrics for Countermeasure Selection in Computer Networks. *Proc. of the 24th Euromicro Intern. Conf. on Parallel, Distributed and Network-based Processing (PDP 2016)*, Heraklion, Crete, Greece, Feb. 2016, Los Alamitos, California, IEEE Computer Society, 2016, pp. 558–565.
26. Kotenko I. V., Doynikova E. V. Countermeasure Selection in Security Management Systems. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2015, no. 3, pp. 60–69 (In Russian). doi:10.15217/issn1684-8853.2015.3.60