Методика оценки устойчивости программно-конфигурируемых сетей в условиях компьютерных атак

Саенко Игорь Борисович
доктор технических наук, профессор, ведущий научный сотрудник, Санкт-Петербургский федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН), Санкт-Петербург, Россия, ibsaen@comsec.spb.ru

Котенко Игорь Витальевич
доктор технических наук, профессор, главный научный сотрудник, СПб ФИЦ РАН, Санкт-Петербург, Россия, ivkote@comsec.spb.ru

Лаута Олег Сергеевич
доктор технических наук, профессор кафедры Государственного университета морского и речного флота им. адмирала С.О. Макарова (ГУМРФ), Санкт-Петербург, Россия, laos-82@yandex.ru

Скоробогатов Сергей Юрьевич
адъюнкт 32 кафедры Военной академии связи имени маршала Советского союза С.М. Буденного, Санкт-Петербург, Россия, skorobogatovsu-vas@yandex.ru

АННОТАЦИЯ

Введение: Важной особенностью технологии SDN является централизованное управление сетью с помощью контроллера, реализованное при помощи протокола управления OpenFlow и позволяющее не только управлять сетевыми устройствами, но и собирать сетевую статистику, что позволяет более эффективно решать возникающие проблемы в сети, конфигурируя одновременно все устройства сети. Контроллер является самым уязвимым элементом SDN, атака на который может повлиять на устойчивость ее функционирования. **Постановка задачи:** разработка математических основ оценки устойчивости SDN позволит с помощью аналитических выражений вычислить показатели устойчивости SDN. В качестве основного показателя предлагается использовать коэффициент исправного действия по устойчивости SDN. **Методы:** оценка показателей устойчивости SDN осуществляется с использованием методов теории Марковских процессов. С целью обеспечения устойчивости функционирования SDN в статье обоснован алгоритм контроля за состоянием контроллеров и их автоматической перестройки. **Результаты:** осуществлена вербальная и математическая постановка научной задачи на исследование, а также общая задача декомпозирована на частные задачи: концептуальное моделирование подсистемы интеллектуального мониторинга состояния информационно-телекоммуникационной сети общего пользования, разработка метода синтеза ее подсистемы интеллектуального мониторинга состояния, а также формирование научно-технических предложений по реализации данной метода. **Практическая значимость:** предложенная методика позволяет оценить устойчивость программно-конфигурируемой сети в условиях характерных для нее компьютерных атак, а также используя полученные показатели устойчивости сформировать общие требования к системе защиты. **Обсуждение:** новизна полученных результатов определяется развитием подходов к аналитическому моделированию атак на SDN, получении исходных данных для оценки устойчивости SDN, особенностью которой является, что уровень управления реализуется программным образом. Новизна полученных результатов определяется тем, что для обоснования устойчивой топологии SDN в условиях кибератак одновременно используются Марковские процессы и метод топологического преобразования стохастических сетей.

КЛЮЧЕВЫЕ СЛОВА: компьютерные атаки; устойчивость; программно-конфигурируемые сети; Марковские цепи.

Введение

Важной особенностью технологии программно-конфигурируемых сетей (англ. – Software-defined networking, SDN) является централизованное управление сетью с помощью контроллера реализованное при помощи протокола управления OpenFlow и позволяющего не только управлять сетевыми устройствами, но и собирать сетевую статистику, что позволяет более эффективно решать возникающие проблемы в сети, реконфигурируя одновременно все устройства сети.

Протокол управления OpenFlow предлагает следующий ряд возможностей, которые позволяют реализовать управляемую и высоконадежную среду [1]:

модель потока великолепно подходит для обеспечения безопасности;

централизованное управление дает возможность рационально контролировать производительность сети, в том числе и в условиях кибератак;

настройка политики безопасности обеспечивается программным контролем;

сдерживание и изоляция от кибератак обеспечивается через гибкое управление трафиком.

Концептуальная структура SDN включает уровень прикладных приложений, API, уровень управления, OpenFlow и уровень передачи данных.

Уровень передачи данных, представленных программными или аппаратно-программными коммутаторами, выполняет функции коммутаторов L2 и L3 уровня по обработки и передачи сетевого трафика. Набор правил каждый коммутатор получает по каналу управления и протоколу управления OpenFlow от контроллера. В свою очередь протокол управления OpenFlow предоставляет контроллеру возможность использования специальных таблиц маршрутизации и/или модификации пакетов, передаваемых коммутаторам. Правила, передаваемые с помощью протокола OpenFlow, могут быть как групповыми, так и дискретными для каждого потока в отдельности. Пакеты, поступившие на входной буфер коммутатора, сначала проверяются на соответствие их заголовков шаблонам правил из нулевой таблицы, а заголовки пакетов сравниваются с шаблонами правил и в случае совпадения выполняется инструкция согласно выбранному правилу.

В программно-конфигурируемой сети можно выделить три основных составляющих: контроллер, канал управления (OpenFlow) и маршрутизатор/коммутатор OpenFlow. Как и в классической архитектуре, основными элементами являются маршрутизаторы/коммутаторы, которые выполняют обработку сетевого трафика уровня L2/L3. Однако в данном случае на сетевые устройства возложена функция пересылки трафика между оконечными пользователями, а все решения, связанные с фильтрацией и перестроениями маршрутов, которые в клас-сической реализации сети выполняли протоколы динамической маршрутизации в данном случае выполняет контроллер.

Контроллер в свою очередь обладает двумя интерфейсами: OpenFlow server который непосредственно управляет сетью и проверяет состояние портов/устройств по средствам протокола OF-CONFIG и API интерфейс, предоставляемый сетевым приложениям.

Понимание процессов функционирования определяется двумя уровнями SDN технологии:

уровень управления (Control Plane);

уровень передачи данных (Data Plane).

Связь между уровнями осуществляет протокол управления Open Flow. Для работы протокола управления реализуется соответствующий защищенный канал. Он может быть, как отдельным физическим соединением контроллер-коммутатор, так и логическим каналам проходящем через другие устройства SDN. По каналу управления происходит информационный обмен. Команды управления от контроллера к коммутатору и информация о состоянии логических переключателей и канала связи между устройствами от коммутатора к контроллеру. Одним из основных плюсов данного решения является централизованное управление. Такая централизация позволяет динамически изменять маршруты передачи трафика в сети исходя из меняющейся обстановки. При создании новых маршрутов и подключении новых каналов контроллер, отвечающий за конкретный сегмент, рассылает каждому устройству необходимые правила. Что отличает SDN от классического подхода, где при изменениях в структуре администратор поэлементно вручную или по протоколам управления SSH/TELNET вынужден прописывать нужные правила [2].

Все это позволяет выделить основные векторы угроз для данной концепции:

пользователи сети SDN, получающие сетевые услуги;

канал от пользовательского до сетевого устройства;

сетевое устройство Open Flow;

канал управления и мониторинга Open Flow;

контроллер SDN.

Контроллер является наиболее уязвимым элементом SDN, атака на который может повлиять на устойчивость ее функционирования, так как он является ключевым компонентом в управлении всей инфраструктурой SDN.

Экономическая парадигма современного мира привела к тому, что разработчики сетевого оборудования экономят на всем, что в свою очередь влияет на общую устойчивость информационно-телекоммуникационных сетей. Об этом не следует забывать, рассматривая технологии программно-конфигурируемых сетей. Таким образом, SDN, с одной стороны, создает определенный риск, открывая злоумышленникам новые горизонты, а с другой, даёт новые возможности по созданию альтернативных сервисов информационной безопасности [3–5].

Степень разработанности темы

На сегодняшний день известны три основных направления по обеспечению устойчивости SDN в условиях таргетированных информационно-технических воздействий [6]. Первый способ – это оптимизация маршрута, используемого для сокращения технологического цикла управления центральным контроллером [7]. В работах [9,10] были рассмотрены вариации усовершенствованного алгоритма Дейкстры по модели взвешенного графа. В работе [11] для решения задачи маршрутизации потоков в условиях коллизий использован итерационный метод Кларка-Райта для оценки операции слияния между маршрутами, особенностью которого является получение выигрышей путем сокращения стоимости комбинированием двух коротких маршрутов в один больший маршрут. Эвристический метод вставок по принципу «ближайшего соседа», а также его ответвление «табу-поиск» рассматривались в работе [12]. Но, не смотря на простоту решения, данные подходы базируются на формально не обоснованных соображениях.

Второй подход к обеспечению устойчивости SDN – структурный – основан на особенностях архитектуры сети. Зачастую важным фактором является устойчивость не всей сети, а ее главной части – системы управления. Кибератаки на SDN в 89% случаях направлены на подсистему управления, так как сбой в её работе приводит к общему «падению». В работе [13] структурная устойчивость SDN оценивается через четыре основных показателя: робастность, возможность резервирования, гибкость управления ресурсом и быстродействие. Такая оценка позволяет отделить процесс маршрутизации от пересылки данных, что является ключевым для сетей подобного рода.

Ряд исследований в области структурной устойчивости SDN направлены на резервирование ключевых контроллеров [14–16] с большим количеством дублируемых трибутарных каналов, использование альтернативных алгебраических то-пологий (например «FatTree») [17], а также гибриды [18–20] разноуровневых то-пологий, таких как «звезда» и «двойное кольцо», с организацией защиты каналов доступа по принципу «1+1». Для всех рассмотренных типов организации структурной устойчивости SDN характерны общие недостатки: несовместимость виртуальной конфигурации с сетевыми контроллерами и высокая стоимость согласованного однотипного «железа» [21].

К третьему варианту повышения устойчивости сети отнесём комбинированные способы, которые сочетают в себе характерные особенности первых двух вариантов. Так, в работе [22] рассмотрен схожий с предлагаемым нами метод превентивного выявления факта воздействия на центральный контроллер SDN, но в отличие от нашего подхода, предлагается систему защиты сети эмулировать на уровне приложений и для отсева аномальных запросов к управляющей подсистеме использовать комплементарный фильтр, который, как известно, имеет характерные временные задержки при переходных процессах.

Общим вопросам количественной оценки устойчивости сложных динамических систем, к числу которых относятся SDN, посвящен ряд работ, например [23,24]. Они рассматривают устойчивость системы как способность «планировать и готовиться к стихийным бедствиям, поглощать их, реагировать на них и восстанавливаться после них, а также адаптироваться к новым условиям». В этих работах предлагается подход к оценке устойчивости компьютерной сети, основанный на учете критической функциональности и особенностей внешних воздействий на элементы сети. Критическая функциональность может быть определена как качество системы [25], а также как показатель производительности системы, который вводится для получения интегрированного показателя устойчивости (например, критическая функциональность может вычисляться как процент функционирующих узлов).

Таким образом, анализ известных работ по устойчивости компьютерных сетей в условиях воздействия на них компьютерных атак позволяет сделать следующие выводы:

стохастическое аналитическое моделирование и методы теории Марковских процессов имеют большое значение для обоснования мер защиты в современных системах информационной безопасности;

с минимальными вычислительными затратами стохастические модели должны обеспечивать вычисление функции распределения интересующих нас случайных величин;

стохастические модели должны обеспечивать моделирование любых атак и высокую гибкость.

Подходы, рассмотренные выше, не в полной мере соответствуют этим требованиям. В основе описываемого ниже подхода к оценке устойчивости SDN лежат методы теории Марковских процессов, так как они позволяют устранить этот недостаток.

Методика оценки устойчивости программно-конфигурируемых сетей в условиях компьютерных атак

При оценке устойчивости сети необходимо определить критерии, при которых она перестанет выполнять возложенные на нее функции.

При рассмотрении транспортной составляющей разумно предположить, что сеть перестанет быть работоспособной при следующих условиях:

отказ контроллера транспортной сети либо подмена контроллера с целью управления нарушителем сетью в своих интересах;

отказ маршрутизаторов, отвечающих за транспортную составляющую сети;

подмена топологии, при которой нарушитель, выдавая себя за маршрутизатор транспортной сети и создание черных дыр для передаваемого трафика;

отказ каналов связи между узлами сети.

Учитывая описанные выше условия оценим устойчивость программно-конфигурируемой сети при резервировании OpenFlow коммутаторов, контроллера. Для этого необходимо представить сеть в виде Марковского процесса с дискретными состояниями в непрерывном времени, время пребывания в одном состоянии распределено по показательному закону.

На рисунке 1 представлен граф дискретных состояний и условных переходов.

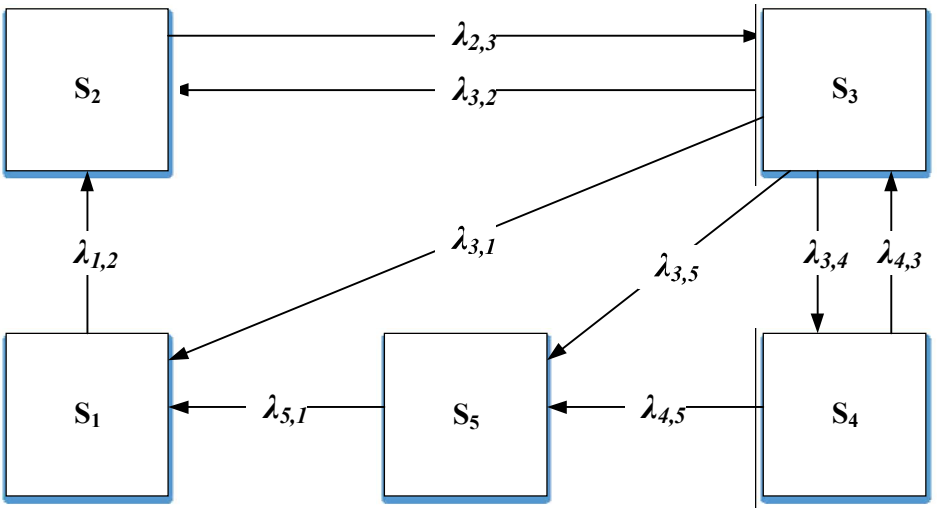

Рис. 1. Граф условных состояний СПД программно-конфигурируемой сети

В таблице 1 приведено описание условных дискретных состояний распределенной корпоративной SDN в условиях кибератак.

Исходные данные для задачи:

1. Граф укрупненных устойчивых состояний SDN в условиях проведения КА $G = (S, \lambda)$.

2. Множество состояний SDN в условиях ведения КА

$$S = (S_1, S_2, S_3, S_4, S_5) \quad (1)$$

3. Множество потоков событий, при изменении состояний SDN в условиях проведения КА

$$\Lambda = (\lambda_{12}, \lambda_{21}, \lambda_{23}, ..., \lambda_{ij}) \quad (2)$$

4. Характеристики устойчивых укрупненных состояний SDN при воздействии кибератак.

5. Значения интенсивностей потоков событий.

6. Вектор вероятностей начальных состояний системы $p_i(0) = |1000000|$:

7. Нормировочное условие:

$$\sum_{i=0}^{4} p_i(t) = 1 \quad (3)$$

Таб.1. Описание условных дискретных состояний распределенной корпоративной SDN в условиях кибератак

Условное обозначение состояния	Описание условного дискретного состояния
S1	Стабильное устойчивое функционирование без отказов.
S2	Функционирование в условиях технической компьютерной разведки. (осуществление нарушителем сбора информации о будущем объекте кибератаки)
S3	Функционирование в условиях проведения кибератак и в отношении SDN.
S4	Функционирование при успешной атаке (успешное подключение к атакуемой сети, получение доступа к атакуемому контроллеру)
S5	Обнаружение аномалий в сети, выявление кибератаки, устранение последствий успешной атаки.

Моменты вероятностных переходов SDN из состояния в состояние при использовании стратегии защиты неопределенны, случайны и происходят под действием потоков событий, которые характеризуются интенсивностями λ_{ij}. Интенсивности являются важной характеристикой потоков событий и представляют среднее число событий, приходящих за единицу времени. Численные значения интенсивностей λ зададим в соответствии с имитационной моделью. При решении системы линейных дифференциальных уравнений с постоянными коэффициентами (однородный Марковский процесс) переходим к непрерывному времени $t \to 0$. По размеченному графу G сформируем систему дифференциальных уравнений с неизвестными функциями $p_i(t)$. Вектор вероятностей начальных состояний системы $p_i(0)$ необходим для точного решения этой системы.

$$D(P,T) = \begin{cases} \frac{dp_0(t)}{dt} = \lambda_{51} p_4(t) + \lambda_{31} p_2(t) - \lambda_{12} p_0(t), \\ \frac{dp_1(t)}{dt} = \lambda_{12} p_0(t) + \lambda_{32} p_2(t) - \lambda_{23} p_1(t), \\ \frac{dp_2(t)}{dt} = \lambda_{23} p_1(t) + \lambda_{31} p_0(t) + \lambda_{35} p_4(t) + \lambda_{34} p_3(t) - (\lambda_{23} + \lambda_{34}) p_2(t), \\ \frac{dp_3(t)}{dt} = \lambda_{43} p_2(t) + \lambda_{45} p_4(t) - \lambda_{34} p_3(t), \\ \frac{dp_4(t)}{dt} = \lambda_{51} p_0(t) - (\lambda_{35} + \lambda_{45}) p_4(t), \\ \sum_{i=0}^{4} p_i(t) = 1. \end{cases} \quad (4)$$

Устойчивость SDN является достаточно обширным понятием, даже в объёме устойчивости SDN в условиях кибератак, потому что, как было сказано ранее, появление новых угроз информационной безопасности есть процесс постоянный. Поэтому устойчивость следует рассматривать как способность распределенной корпоративной сети с использованием технологии SDN противостоять к определённому классу кибератак, а *S*4 – состояние функционирования SDN при успешном осуществлении кибератаки.

Таким образом, блок-схема предлагаемой методики оценки устойчивости, распределенной SDN в условиях кибератак, представлена на рис. 2.

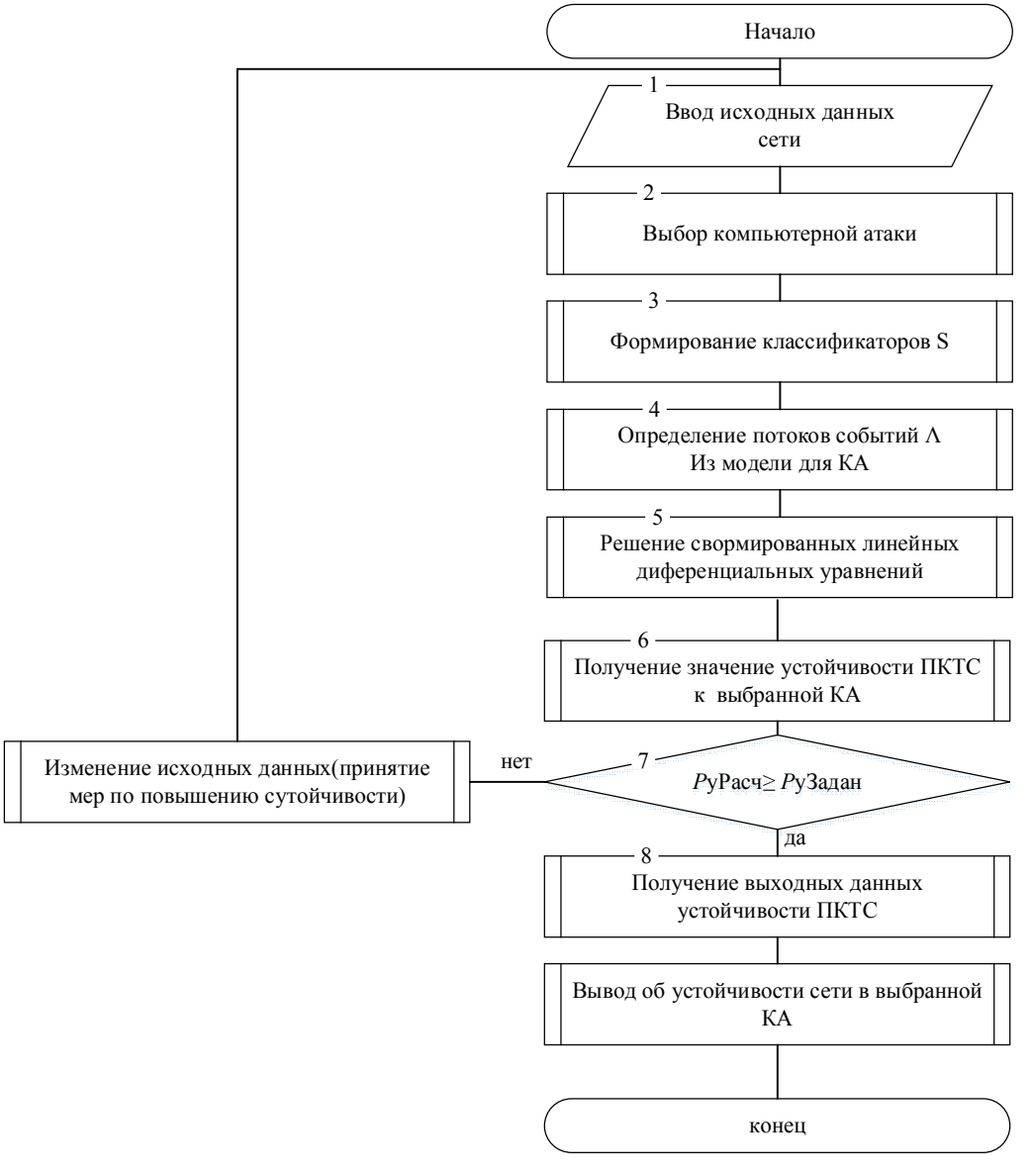

Рис. 2. Блок-схема оценки устойчивости SDN в условиях кибератак

Методика позволяет при определении наиболее актуальных атак для корпоративной SDN определить степень ее устойчивости. Получаемые результаты и выводы, основанные на них, вызволяют получить адекватную оценку устойчивости SDN в моделируемых условиях к КА характерным для данной сети.

Оценка устойчивости SDN в условиях КА

Для оценки устойчивости SDN в условиях КА были разработаны и реализованы три структуры сети:

Структура 1 – структура SDN, состоящая из трех элементов с одним контроллером;

Структура 2 – структура SDN с двумя контроллерами с разделением функцией управления и перехватом функций управления друг друга по заданному алгоритму в условиях КА;

Структура 3 – структура SDN с двумя контроллерами, когда один контроллер является основным и выполняет функции управления, а второй контроллер находится в режиме горячего резервирования.

Результаты расчета представлены в виде графиков (рис. 3-5).

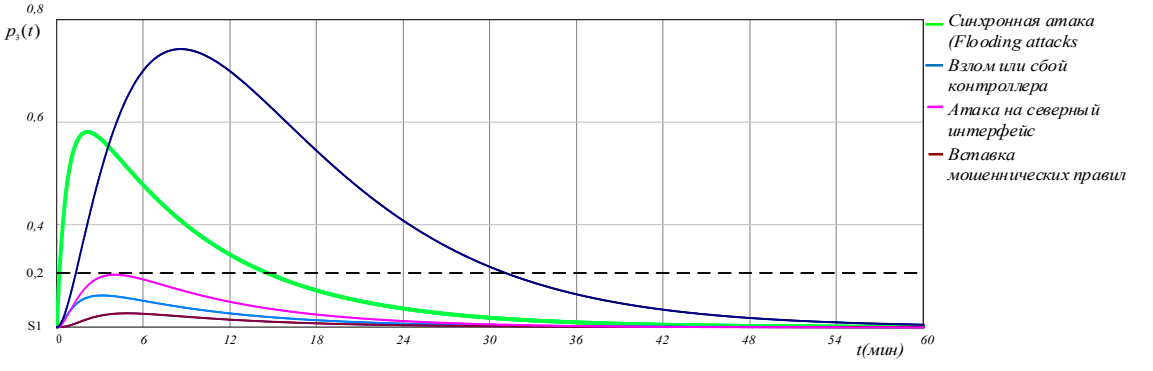

Рис. 3. Зависимость вероятности устойчивой работы SDN от времени реализации КА для Структуры 1

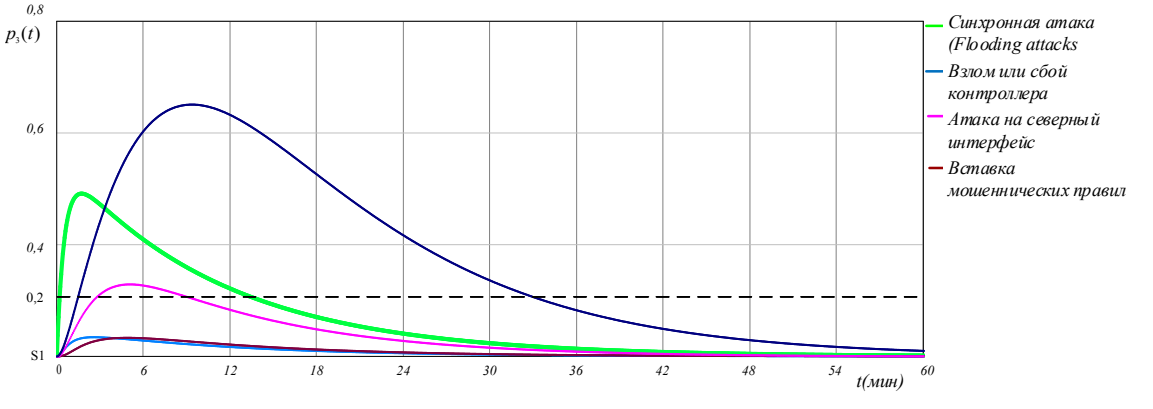

Рис 4. Зависимость вероятности устойчивой работы SDN от времени реализации КА для Структуры 2

Анализ результатов показал, что рассматриваемые структуры SDN в условиях воздействия КА типов «Синхронная атака» и «Взлом или сбой контроллера» не соответствуют требованиям по устойчивости. С целью обеспечения устойчивости функционирования SDN в условиях КА необходимо разработать алгоритм контроля за состоянием контроллеров и их автоматической перестройки, так как через 18 минут успешной реализации КА вероятность устойчивого функционирования сети начинает стремиться к 0.

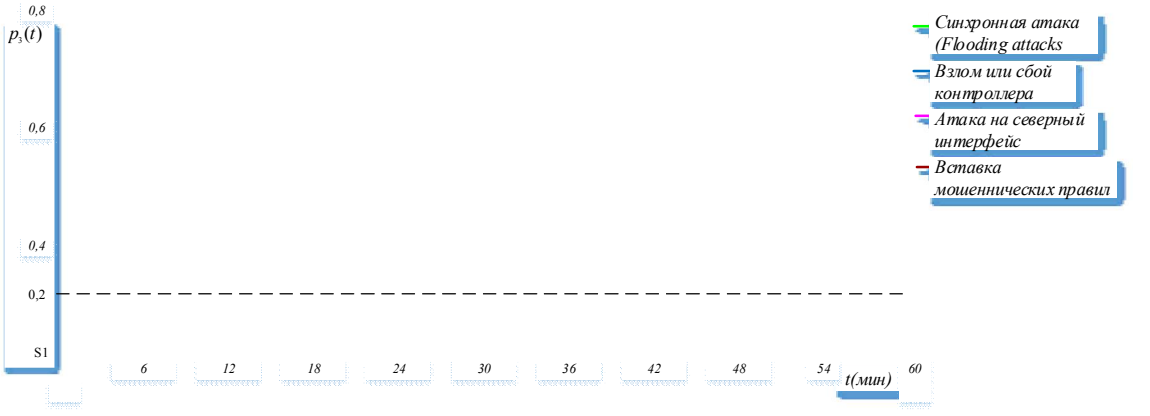

Рис. 5. Зависимость вероятности устойчивой работы SDN от времени реализации КА для Структуры 3

Таким образом, на основании проведенных исследований по применению SDN, а также их устойчивости к КА были сформированы общие требования к системе противодействия. Основным способом достижения требуемого уровня устойчивости SDN может стать разработка алгоритмов и способов резервирования контроллеров, а также алгоритмов резервирования и переключения программных коммутаторов.

Алгоритм контроля за состоянием контроллеров и их автоматической перестройки

Учитывая вышеизложенное, можно сформулировать требования к отказоустойчивому контуру управления SDN, который включает в себя три основных уровня:

1. Контур управления уровнем передачи.
2. Инфраструктура мониторинга и управления OpenFlow.
3. Межконтроллерная коммуникационная инфраструктура.

Сложность функционирования такого контура заключается в формировании условий инициирования процессов восстановления SDN при реализации различных КА злоумышленником.

Для решения задачи повышения устойчивости SDN в условиях КА необходима система защиты (рисунок 6), в основу которой заложен алгоритм восстановления сети, выделяющий два уровня:

уровень передачи;

уровень управления.

При проведении КА злоумышленник осуществляет ряд последовательных действий, при обнаружении которых система защиты должна сигнализировать об этом сетевому администратору, а также рекомендовать применение сценария противодействия. При этом признаки КА для уровней SDN будут различными. Например, для атаки типа «Сбой или взлом контроллера», направленной на программный коммутатор, будет характерно повышение трафика, проходящего через контроллер, повышение задержки или отсутствие ответом от контроллера, а соответственно для контроллера – повышение нагрузки процессора, количества запросов и т.д.

В этом случае очевидна необходимость разделения условий реагирования программных коммутаторов и контроллеров на признаки КА.

Для построения системы отказоустойчивости SDN в условиях КА разработана клиент-серверная структура, которая состоит из агентов, функционирующих на программных маршрутизаторах и сервере, функционирующем на контролере (рис. 7).

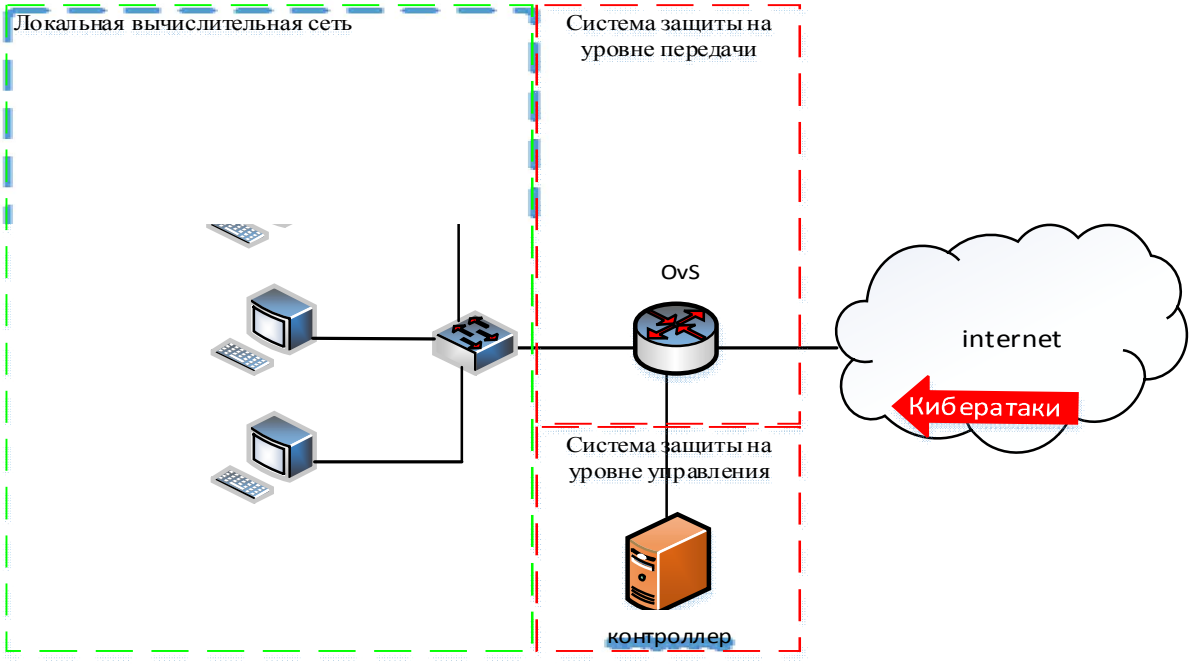

Рис. 6. Вариант предлагаемой структуры построения системы защиты SDN в условиях КА

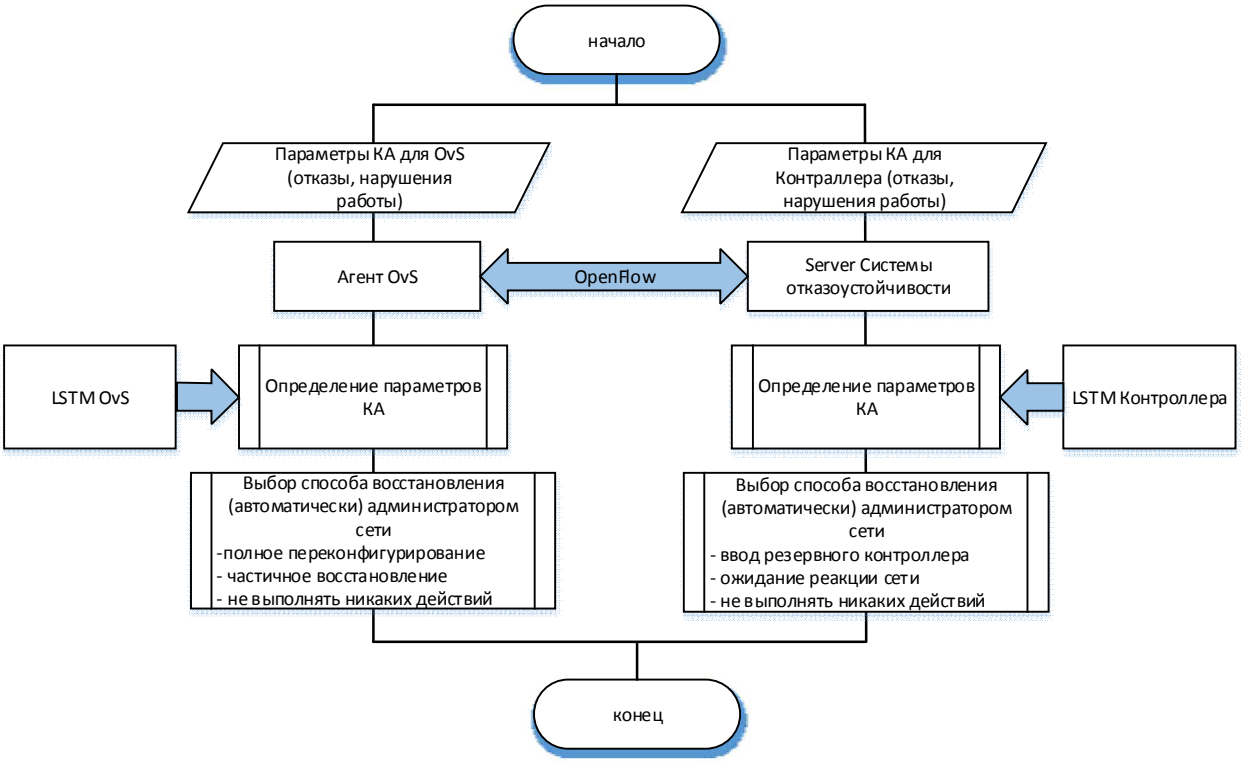

Рис. 7. Обобщенная структура системы обеспечения отказоустойчивости сегмента SDN

На программном маршрутизаторе работает Агент OvS, а на контроллере развернут сервер системы отказоустойчивости, которые взаимодействуют по протоколу OpenFlow.

Далее после принятия решения должностным лицом выполняется выбранный сценарий. Из базы данных конфигураций загружается порядок действий, производятся автоматические настройки и подключение к резервному контроллеру при выборе режима полной пере конфигурации.

Сервер системы отказоустойчивости запускается в двух версиях – “master” (ведущий) и “slave” (подчиненный). Для их совместной работы реализуется сервис синхронизации, который отвечает за переключение контроллеров между собой, а также зеркалирование служебной информации, необходимой для принятия решений на проведение сценария восстановления.

Для определения КА, также как и в Агенте OvS, используется LSTM-сеть. Набор данных, предназначенный для ее обучения, будет отличаться от используемого в агенте. Проводимые КА могут быть направлены на различные элементы сети и вызывать разные последствия. Таким образом, важным фактором является отсутствие противоречий между выполняемыми агентом сценариями противодействия и сервером системы, даже при отсутствии канала управления OpenFlow между ними.

Блок-схема функционирования сервера предлагаемой системы обеспечения устойчивости SDN представлена на рисунке 8.

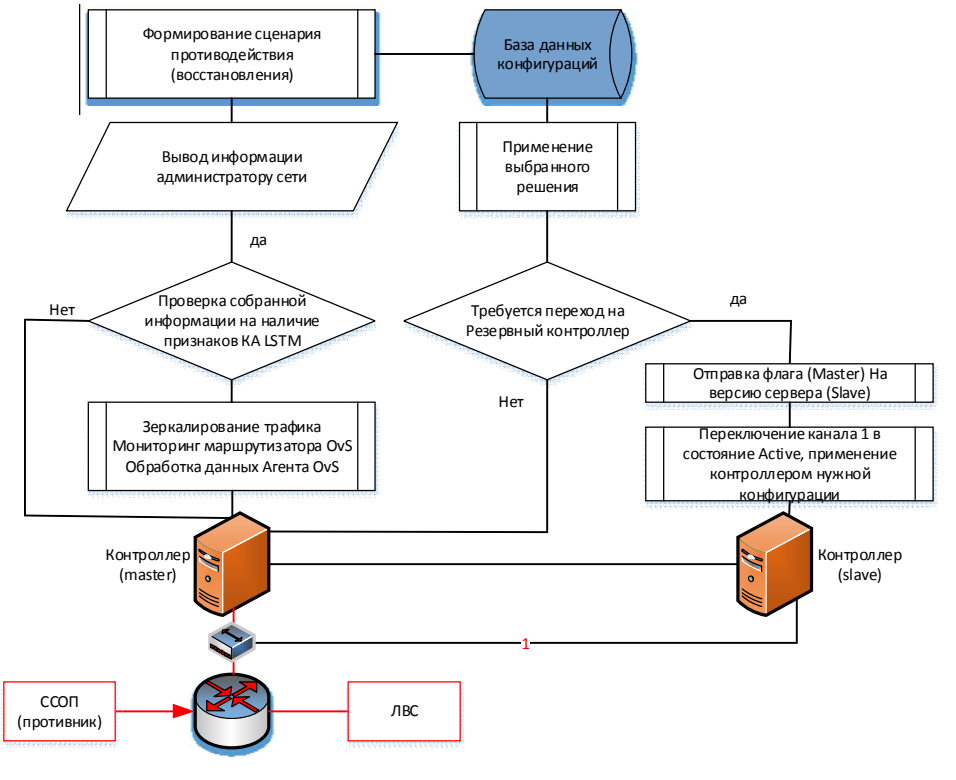

Рис. 8. Блок-схема функционирования сервера системы обеспечения устойчивости SDN

Таким образом, архитектура системы обеспечения устойчивости SDN состоит из Агентов OvS, запускаемых вместе с программными маршрутизаторами уровня передачи технологии ПКС, отвечающей за принятие мер по реконфигурации сети в случае обнаружения КА с

помощью нейронной сети LSTM, и сервера системы, отвечающий за принятие мер на уровне управления, вплоть до ввода в работу резервного контроллера (slave).

Заключение

Таким образом, предложенная методика оценки устойчивости SDN в условиях КА позволяет обосновать наиболее устойчивую топологию сети и рассчитать показатели компьютерных атак. Расчет вероятностно-временных характеристик известных КА осуществляется на основе разработки их профильных моделей. Полученные значения в дальнейшем используются в качестве исходных данных при оценке угроз и обосновании требований по защите SDN от КА. Кроме того, разработаны алгоритмы резервирования контроллеров и переключения программных коммутаторов.

Дальнейшие исследования связываются с разработкой аналитических моделей для реализации контрмер в сетях SDN и интеграцией их с моделями кибератак.

Благодарность. Работа выполнена при частичной финансовой поддержке бюджетной темы FFZF-2022-0007.

Литература

1. *H. E. Egilmez, S. T. Dane, K. T. Bagci and A. M. Tekalp*, "OpenQoS: An OpenFlow controller design for multimedia delivery with end-to-end Quality of Service over Software-Defined Networks," Proceedings of The 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference, Hollywood, CA, USA, 2012, pp. 1-8.

2. *Lei, Yunsen; Lanson, Julian; Kaldawy, Remy; Estrada, Jeffrey; Shue, Craig* (11 November 2020). "Can Host-Based SDNs Rival the Traffic Engineering Abilities of Switch-Based SDNs?". IEEE Network of the Future Conference: 91–99. doi:10.1109/NoF50125.2020.9249110.

3. *W. Xia, Y. Wen, C. H. Foh, D. Niyato and H. Xie,* "A Survey on Software-Defined Networking," in IEEE Communications Surveys & Tutorials, vol. 17, no. 1, pp. 27-51, Firstquarter 2015, doi: 10.1109/COMST.2014.2330903.

4. *J. Vestin, A. Kassler and J. Akerberg*, "Resilient software defined networking for industrial control networks," 2015 10th International Conference on Information, Communications and Signal Processing (ICICS), Singapore, 2015, pp. 1-5, doi: 10.1109/ICICS.2015.7459981.

5. *D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky and S. Uhlig*, "Software-Defined Networking: A Comprehensive Survey," in Proceedings of the IEEE, vol. 103, no. 1, pp. 14-76, Jan. 2015, doi: 10.1109/JPROC.2014.2371999.

6. *V. Ahmadi.* A hybrid NSGA-II for solving multiobjective controller placement in SDN / V.Ahmadi, A.Jalili, S.M.Khor, and M.Keshtgari // in KBEI, 2015.

7. *S.Agarwal*. Traffic engineering in SDN / S.Agarwal, M.Kodialam, T.Lakshman // 2013 Proceedings IEEE INFOCOM, paper no 06567024, pp. 2211-2219.

8. *H. Engilmez*. Open QoS: an OpenFlow controller Design for multimedia delivery with End-to-End QoS over SDN / H.Engilmez, S.Dane, K.Bagci // APSIPA Annual summit and conf., LA, CA, Dec. 2012 pp. 194-202.

9. *Nencioni G*. Impact of SDN Controllers Deployment / Nencioni G., Helvik B.E., Gonzalezy A.J., Heegaard P.E., Kamisinski A. // on Network Availability — Cornell University Library [URL accessed on 20 Sept. 2022].

10. *F. Bannour*. Scalability and Relia bility Aware SDN Controller Placement Strategies / Fetia Bannour, Sami Souihi and Abdelhamid Mellouk // LISSI TincNetwork Research Team, University Paris-Est Cr´eteil (UPEC), France, November 2017 [DOI:10. 23919/CNSM.2017.8255989].

11. *Kenneth Sörensen. Florian Arnold. Daniel Palhazi Cuervo*. A critical analysis of the “improved Clarke and Wright savings algorithm”: Sörensen et al. University of Antwerp. Departement of Engineering Management,ANT/OR - Operations Research GroupJune 13, 2017.

12. *F.J. Ros and P. M. Ruiz*, On reliable controller placements in software-defined networks // in Comput. Commun., vol. 77, pp. 41–51, Mar. 2016.

13. *G. Yao*. On the capacitated controller placement problem in software defined networks / G. Yao, J. Bi, Y. Li, and L. Guo // in IEEE Commun. Lett., pp. 141–159, Dec. 2017.

14. *Jaiyong, L.* E□cient routing for tra□c o□oading in software-defined network / Jaiyong, L., Min Park, S., Seungbum, Ju. // The 9th International Conference on Future Networks and Communications, Vol. 34, pp. 674679 (2019).

15. *Singh, S.* A survey on Software Defined Networking: Architecture for next generation network / Singh, S., Jha, R. K. // In: Journal of Network and Systems Management, vol. 25 no. 2, pp. 321374 (2017). doi:10.1007/s10922-016-9393-9.

16. *S. Lange*. Specialized heuristics for the controller placement problem in large scale SDN networks / S. Lange, S. Gebert, J. Spoerhase, P. Rygielski, T. Zinner, S. Kounev, and P. Tran-Gia // in Proc. ITC, Sept 2015, pp. 210–218.

17. *M.Ashouri*. Enhancing the Performance and Stability of SDN Architecture with a Fat-Tree Based Algorithm / Mohammadreza Ashouri, Shirin Setayesh / HAL Open science, Id: hal-01858528. Preprint submitted on 20 Aug 2018 [https://hal.archives-ouvertes.fr/hal-01858528].

18. *J. Li, X. Chang, Y. Ren, Z. Zhang, & G. Wang*, "An Effective Path Load Balancing Mechanism Based on SDN." Trust, Security and Privacy in Computing and Communications (TrustCom), 2014 IEEE 13th International Conference on IEEE, (2014), pp. 527-533.

19. *Celenlioglu, M.R.* Design, implementation and evaluation of SDN-based resource management model / Celenlioglu, M. R., Alsadi M., Mantar, H. A. // In: 7th International Conference on New Technologies, Mobility and Security (NTMS), Paris, pp. 1-8 (2015).

20. *Li, W.* A survey on OpenFlow-based Software Defined Networks: Security challenges and countermeasures / Li, W., Meng, M., Kwok, L.M. // In: Journal of Network and Computer Applications, Issue 68, pp. 126139 (2016).

21. *A.M. Koushika, S.T. Selvi.* "Load balancing Using Software Defined Networking in cloud environment" Recent Trends in Information Technology (ICRTIT), 2014 International Conference on. IEEE, (2019), pp. 1-8.

22. *K. Govindarajan*. Realizing the Quality of Service (QoS) in Software-Defined Networking (SDN) based Cloud infrastructure / Govindarajan, K., Meng, K. C., Ong, H., Tat, W. M., Sivanand, S., And Leong, L. S. // In Information and Communication Technology (ICoICT), 2020 2nd International Conference on(pp. 505-510). IEEE. (2020).

23. *Котенко В.И., Саенко И.Б., Коцыняк М.А., Лаута О.С.* Оценка киберустойчивости компьютерных сетей на основе моделирования кибератак методом преобразования стохастических сетей // Труды СПИИРАН. 2017. № 6(55). С.160-184.

24. *Kotenko I., Saenko I., Lauta O.* Modeling the Impact of Cyber Attacks // Kott A., Linkov I. (eds) Cyber Resilience of Systems and Networks. Risk, Systems and Decisions. Springer, Cham. 2019. Pp. 135-169. ISBN 978-3-319-77491-6. https://doi.org/10.1007/978-3-319-77492-3_7.

25. *Briana Lucero, Vimal Viswanathan, Julie Linsey Georgia*. Analysis of Critical Functionality for Meta Analogy via Performance Specification. Proceedings of the International Design Engineering Technical Conference IDETC August 4-7, 2013, Portland, Oregon, USA DRAFT IDETC2013-13472.

SUSTAINABILITY ASSESSMENT METHODOLOGY SOFTWARE-CONFIGURABLE NETWORKS IN THE CONDITIONS OF COMPUTER ATTACKS

IGOR B. SAENKO
leading researcher, St. Petersburg Federal Research Center of the Russian Academy of Sciences, doctor of technical sciences, professor, ibsaen@comsec.spb.ru

IGOR V. KOTENKO
chief researcher, St. Petersburg Federal Research Center of the Russian Academy of Sciences, doctor of technical sciences, professor, ivkote@comsec.spb.ru

OLEG S. LAUTA
professor, Admiral Makarov State University of Mari-time and inland shipping, doctor of technical sciences, laos-82@yandex.ru

SERGEY Y. SKOROBOGATOV
graduate student, Military Academy of Communications named after Marshal of the Soviet Union S.M. Budyonny, skorobogatovsu-vas@yandex.ru

ABSTRACT

Introduction: An important feature of SDN technology is centralized network management using a controller, implemented using the OpenFlow control protocol and allowing not only to manage network devices, but also to collect network statistics, which allows you to more effectively solve emerging network problems by configuring all network devices simultaneously. The controller, as a key component in the management of the entire SDN infrastructure, is the most vulnerable element, an attack on which can entail consequences critical for the entire infrastructure, i.e. affect the stability of its functioning. **Problem statement:** the development of mathematical foundations for assessing SDN stability will allow using analytical expressions to calculate SDN stability indicators. As the main indicator, it is proposed to use the coefficient of serviceable action for SDN stability. **Methods:** assessment of SDN stability indicators is carried out using methods of the theory of Markov processes. In order to ensure the stability of the SDN operation, the article substantiates an algorithm for monitoring the state of controllers and their automatic adjustment. **Results:** the proposed methodology makes it possible to assess the stability of a software-configured network in the conditions of computer attacks characteristic of it, and also using the obtained stability indicators to form general requirements for the protection system. **Practical significance:** the proposed methodology makes it possible to assess the stability of a software-configured network in the conditions of computer attacks characteristic of it, as well as using the obtained stability indicators to form general requirements for the protection system. **Discussion**: the novelty of the results obtained is determined by the development of approaches to analytical modeling of attacks on SDN, obtaining initial data for assessing the stability of SDN, the peculiarity of which is that the control level is implemented programmatically. The novelty of the obtained results is determined by the fact that Markov processes and the method of topological transformation of stochastic networks are simultaneously used to substantiate the stable topology of SDN in the conditions of cyberattacks.

Keywords: computer attacks; stability; software-configurable networks; Markov chains.

REFERENCES

1. H. E. Egilmez, S. T. Dane, K. T. Bagci and A. M. Tekalp, "OpenQoS: An OpenFlow controller design for multimedia delivery with end-to-end Quality of Service over Software-Defined Networks," Proceedings of The 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference, Hollywood, CA, USA, 2012, pp. 1-8.
2. Lei, Yunsen; Lanson, Julian; Kaldawy, Remy; Estrada, Jeffrey; Shue, Craig (11 November 2020). "Can Host-Based SDNs Rival the Traffic Engineering Abilities of Switch-Based SDNs?". IEEE Network of the Future Conference: 91–99. doi:10.1109/NoF50125.2020.9249110.
3. W. Xia, Y. Wen, C. H. Foh, D. Niyato and H. Xie, "A Survey on Software-Defined Networking," in IEEE Communications Surveys & Tutorials, vol. 17, no. 1, pp. 27-51, Firstquarter 2015, doi: 10.1109/COMST.2014.2330903.
4. J. Vestin, A. Kassler and J. Akerberg, "Resilient software defined networking for industrial control networks," 2015 10th International Conference on Information, Communications and Signal Processing (ICICS), Singapore, 2015, pp. 1-5, doi: 10.1109/ICICS.2015.7459981.
5. D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky and S. Uhlig, "Software-Defined Networking: A Comprehensive Survey," in Proceedings of the IEEE, vol. 103, no. 1, pp. 14-76, Jan. 2015, doi: 10.1109/JPROC.2014.2371999.
6. V. Ahmadi. A hybrid NSGA-II for solving multiobjective controller placement in SDN / V.Ahmadi, A.Jalili, S.M.Khor, and M.Keshtgari // in KBEI, 2015.
7. S.Agarwal. Traffic engineering in SDN / S.Agarwal, M.Kodialam, T.Lakshman // 2013 Proceedings IEEE INFOCOM, paper no 06567024, pp. 2211-2219.
8. H. Engilmez. Open QoS: an OpenFlow controller Design for multimedia delivery with End-to-End QoS over SDN / H.Engilmez, S.Dane, K.Bagci // APSIPA Annual summit and conf., LA, CA, Dec. 2012 pp. 194-202.
9. Nencioni G. Impact of SDN Controllers Deployment / Nencioni G., Helvik B.E., Gonzalezy A.J., Heegaard P.E., Kamisinski A. // on Network Availability — Cornell University Library [URL accessed on 20 Sept. 2022].
10. F. Bannour. Scalability and Relia bility Aware SDN Controller Placement Strategies / Fetia Bannour, Sami Souihi and Abdelhamid Mellouk // LISSI TincNetwork Research Team, University Paris-Est Cr´eteil (UPEC), France, November 2017 [DOI:10. 23919/CNSM.2017.8255989].
11. Kenneth Sörensen. Florian Arnold. Daniel Palhazi Cuervo. A critical analysis of the “improved Clarke and Wright savings algorithm”: Sörensen et al. University of Antwerp. Departement of Engineering Management,ANT/OR - Operations Research GroupJune 13, 2017.
12. F.J. Ros and P. M. Ruiz, On reliable controller placements in software-defined networks // in Comput. Commun., vol. 77, pp. 41–51, Mar. 2016.
13. G. Yao. On the capacitated controller placement problem in software defined networks / G. Yao, J. Bi, Y. Li, and L. Guo // in IEEE Commun. Lett., pp. 141–159, Dec. 2017.
14. Jaiyong, L. E□cient routing for tra□c o□oading in software-defined network / Jaiyong, L., Min Park, S., Seungbum, Ju. // The 9th International Conference on Future Networks and Communications, Vol. 34, pp. 674679 (2019).
15. Singh, S. A survey on Software Defined Networking: Architecture for next generation network / Singh, S., Jha, R. K. // In: Journal of Network and Systems Management, vol. 25 no. 2, pp. 321374 (2017). doi:10.1007/s10922-016-9393-9.
16. S. Lange. Specialized heuristics for the controller placement problem in large scale SDN networks / S. Lange, S. Gebert, J. Spoerhase, P. Rygielski, T. Zinner, S. Kounev, and P. Tran-Gia // in Proc. ITC, Sept 2015, pp. 210–218.
17. M.Ashouri. Enhancing the Performance and Stability of SDN Architecture with a Fat-Tree Based Algorithm / Mohammadreza Ashouri, Shirin Setayesh / HAL Open science, Id: hal-01858528. Preprint submitted on 20 Aug 2018 [https://hal.archives-ouvertes.fr/hal-01858528].

18. J. Li, X. Chang, Y. Ren, Z. Zhang, & G. Wang, "An Effective Path Load Balancing Mechanism Based on SDN." Trust, Security and Privacy in Computing and Communications (TrustCom), 2014 IEEE 13th International Conference on IEEE, (2014), pp. 527-533.
19. Celenlioglu, M.R. Design, implementation and evaluation of SDN-based resource management model / Celenlioglu, M. R., Alsadi M., Mantar, H. A. // In: 7th International Conference on New Technologies, Mobility and Security (NTMS), Paris, pp. 1-8 (2015).
20. Li, W. A survey on OpenFlow-based Software Defined Networks: Security challenges and countermeasures / Li, W., Meng, M., Kwok, L.M. // In: Journal of Network and Computer Applications, Issue 68, pp. 126139 (2016).
21. A.M. Koushika, S.T. Selvi. "Load balancing Using Software Defined Networking in cloud environment" Recent Trends in Information Technology (ICRTIT), 2014 International Conference on. IEEE, (2019), pp. 1-8.
22. K. Govindarajan. Realizing the Quality of Service (QoS) in Software-Defined Networking (SDN) based Cloud infrastructure / Govindarajan, K., Meng, K. C., Ong, H., Tat, W. M., Sivanand, S., And Leong, L. S. // In Information and Communication Technology (ICoICT), 2020 2nd International Conference on(pp. 505-510). IEEE. (2020).
23. Kotenko V.I., Saenko I.B., Kotsynyak M.A., Lauta O.S. Assessment of cyber-stability of computer networks based on the simulation of cyber attacks by the method of transformation of stochastic networks // Proceedings of SPIIRAN. 2017. No. 6(55). pp.160-184. (In Rus).
24. Kotenko I., Saenko I., Lauta O. Modeling the Impact of Cyber Attacks // Kott A., Linkov I. (eds) Cyber Resilience of Systems and Networks. Risk, Systems and Decisions. Springer, Cham. 2019. Pp. 135-169. ISBN 978-3-319-77491-6. https://doi.org/10.1007/978-3-319-77492-3_7.
25. Briana Lucero, Vimal Viswanathan, Julie Linsey Georgia. Analysis of Critical Functionality for Meta Analogy via Performance Specification. Proceedings of the International Design Engineering Technical Conference IDETC August 4-7, 2013, Portland, Oregon, USA DRAFT IDETC2013-13472.