МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра математического обеспечения и применения ЭВМ

ОТЧЕТ

по научно-исследовательской работе

ТЕМА: МОДЕЛИ И МЕТОДЫ ОБНАРУЖЕНИЯ DDOS-АТАК В ОБЛАЧНЫХ ВЫЧИСЛИТЕЛЬНЫХ СРЕДАХ НА ОСНОВЕ ТЕХНОЛОГИИ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ

Студент(ка) гр. 3303 _____ Сандин А.В.

Руководитель _____ Борисенко К.А.

Санкт-Петербург
2018

ЗАДАНИЕ

НА НАУЧНО-ИССЛЕДОВАТЕЛЬСКУЮ РАБОТУ

Студент(ка) Сандин А.В.

Группа 3304

Тема НИР: модели и методы обнаружения DDoS-атак в облачных вычислительных средах на основе технологии интеллектуального анализа данных.

Задание на НИР:

1) Провести исследование современных подходов к организации систем защиты в облачных вычислительных средах;
2) Рассмотреть существующие виды DDoS –аттак;
3) Описать предполагаемое архитектурное решение.

Сроки выполнения НИР: 15.11.2018 – 20.12.2018

Дата сдачи отчета: 20.12.2018

Дата защиты отчета: 20.12.2018

Студент(ка) _____ Сандин А.В.

Руководитель _____ Борисенко К.А.

АННОТАЦИЯ

Произвести исследования в области методов обнаружения DDoS-атак в облачных вычислительных средах. Проанализировать современные подходы к организации систем защиты. Рассмотреть виды DDoS-аттак на облачные вычислительные среды. Описать предполагаемую архитектуру системы защиты облачных вычислительных сред.

SUMMARY

To conduct research in the field of detection of DDoS-attacks in cloud computing environments. Analyze modern approaches to the organization of protection systems. Consider the types of DDoS-attack on cloud computing environments. Describe the proposed cloud computing security architecture.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Технология облачных вычислений является одним из наиболее перспективных направлений развития информационных систем. Преимущественно облачные ресурсы предоставляются по следующим сервисным моделям: Software as a Service (SaaS, программного обеспечение как услуга), Platform as a Service (PaaS, платформа как услуга), Infrastructure as a Service (IaaS, инфраструктура как услуга). Модель IaaS позволяет пользователям создавать виртуализированные компьютерные сети, включающие в себя как виртуальные машины пользователей, так и сервера. Данная технология позволяет клиентам значительно снизить затраты на создание сетевой инфраструктуры, быстро реконфигурировать топологии компьютерной сети, настраивать виртуальные машины и сервисы. Однако использование технологий предоставления облачных ресурсов значительно усложняет процесс обеспечения защиты виртуализированных компьютерных сетей. В случае успешного выполнения DDoS-атак на виртуализированные компьютерные сети облачных вычислительных средах или отдельные их узлы, жертвы вредоносного воздействия начинают потреблять большее количество общих ресурсов. Вследствие чего, атака на один конкретный виртуализированный узел компьютерной сети может привести к выходу из строя всех элементов компьютерной сети облачных вычислительных средах, что повышает значимость создания эффективных методов защиты от сетевых атак. Существующие методы защиты не эффективны в условиях высоконагруженных облачных вычислительных средах. На текущем этапе развития облачных вычислений выявлен ряд уязвимостей, связанных не только с классическими угрозами для распределенных систем, но и с принципиально новыми, порожденными спецификой виртуализации, а также наличием дополнительных уязвимых компонентов реализующих предоставление

облачных услуг. В данный момент, большая часть действий по администрированию облачных вычислительных сред, а также защиты их от вредоносных воздействий требует вмешательства системного администратора. Данный способ является неэффективным и трудозатратным. Поэтому возникает необходимость разработки новых моделей, методов и алгоритмов, направленных на выявление проблем функционирования облачных вычислительных сред, вызванных DDoS-атаками.

Целью является разработка методов и модели обнаружения DDoS-атак в облачных вычислительных средах на основе методов интеллектуального анализа данных.

Для достижения цели были поставлены и решены следующие задачи:

1. Анализ современных подходов к организации систем защиты в облачных вычислительных средах.
2. Разработка методики сбора и анализа трафика.
3. Разработка метода обнаружения и блокировки сетевых атак на облачных вычислительных средах, работающего в автоматическом режиме.
4. Разработка метода моделирования сетевых атак для экспериментального исследования вредоносных воздействий на облачные вычислительные среды.
5. Программная реализация компонентов системы защиты облачных вычислительных сред, основанная на разработанных моделях и методах.
6. Тестирование эффективности разработанной системы.

1. ИССЛЕДОВАНИЕ СОВРЕМЕННЫХ ПОДХОДОВ К ОРГАНИЗАЦИИ СИСТЕМ ЗАЩИТЫ

Существующие системы обнаружения вторжений требуют постоянного обновления набора правил, которые должны соответствовать актуальным угрозам. Но даже периодического обновления набора правил может быть недостаточно для того, чтобы система всегда оставалась в актуальном состоянии.

В настоящее время разработано достаточно различных защитных методик для обнаружения вредоносного трафика и защиты от DDoS-атак.

Elastic Cloud Security System ECS2 [1] предоставляет сложную защитную систему от вредоносного трафика. ECS2 имеет антибот-таблицы и антивирусные решения, встроенные в систему защиты. Также в системе установлены межсетевые экраны, работающие в режиме реального времени. Минусом использования подобного подхода является время обновления репутационных таблиц и сигнатур вирусов. Несвоевременное обновление таблиц может привести к пропуску вредоносного трафика, генерируемого с непомеченных IP-адресов. Поэтому мой подход не основан на сигнатурах, списков вредоносных IP-адресов и других методов, данные для которых необходимо постоянно обновлять.

В статье [2] описан защитный метод, использующий ресурсы для обнаружения вредоносного трафика виртуальных машин. Метод был протестирован на 108 сервисах, запущенных на различных виртуальных машинах. Приведенные в статье методы основаны на технологии интеллектуального анализа данных и запускаются как приложения на виртуальных машинах. Минусом данного решения является расположение процессов защиты на стороне клиента. Многие клиенты не хотят видеть на своих виртуальных машинах фоновые сторонние процессы. Это не соответствует политикам безопасности многих компаний.

В статье[3] рассматривается новый метод обнаружения аномалий в трафике, использующий анализ потоков, основанный на K-mean алгоритме. Обучение данных, содержащих неподписанные записи потоков, делится на кластеры легитимного и вредоносного трафиков. Соответствующие центроиды кластеров используются как паттерны для эффективного обнаружения аномалий в анализируемом потоке трафика. Авторы указывают, что применение алгоритма раздельно для разных сервисов (использующих разные протоколы и номера портов) увеличивает точность обнаружения.

Авторы [4] описывают алгоритм для обнаружения DDoS-атак, который использует SSL/TLS протокол. Алгоритм основан на фильтрации шума в данных и кластеризации для обнаружения вредоносного трафика. Модели были обучены на данных, полученных с реалистичной компьютерной среды. Авторы заключают, что предложенная модель позволяет обнаруживать все вредоносные потоки с очень малым количеством ложных срабатываний.

Использование методов детектирования DDoS-атак на основе машинного обучения является достаточно популярной задачей среди исследователей. Один из методов детектирования, предложенный в работе [5] Keisuke Kato, с использованием алгоритма машинного обучения support vector Keisuke Kato machine with the radial basis function machine with the radial basis function (Gaussian) kernel на dataset from UCLA. Однако в работе не рассматривается возможность динамического изменения трафика.

В работе Wafa Slaibi Alsharafat [6] представлен метод детектирования на основе генерации классификаторов с использованием генетического алгоритма с набором данных KDD’99.

2. ВИДЫ DDOS-АТТАК НА ОБЛАЧНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СРЕДЫ.

DDoS-атаки затрагивают все сервисные модели ОВС, однако отличием от атак на обычные сервисы является возможность генерации вредоносного трафика внутри ОВС, исчерпывая пул ресурсов, принадлежащих большому количеству пользователей. Существует большое количество различных типов и сценариев сетевых атак. Многие из типов устарели в связи с разработкой/доработкой протоколов, программного обеспечения, устраняющих уязвимости, используемые атаками. Постоянно появляются новые типы и сценарии сетевых атак, что требует также усовершенствований методов защиты от них. Далее рассмотрены основные типы сетевых атак на облачные вычислительные среды.

Атаки с подменой IP-адреса [7]. Коммутация пакетов от клиента до сервисы может быть перехвачена вредоносной программой и в пакетах заменен IP-адрес отправителя на несуществующий или чужой. В результате сервис не сможет завершить транзакцию, и легитимный пользователь не получит ответ, что повлечет трату дополнительных ресурсов сервисом.

SYN Flooding атака [7]. TCP-протокол для установления связи между клиентом и сервисом использует метод трехэтапного рукопожатия. Клиент посылает SYN-запрос сервису, который отвечает на него с SYN-ACK флагом. Получив ответ от сервиса клиент отправляет ему пакет с флагом ACK, после чего начинается обмен пакетами. В случае атаки клиент отправляет большое количество пакетов на запрос трехэтапного рукопожатия с SYN флагом. Сервис, обрабатывая эти запросы, отмечает клиенту, который, в свою очередь, не завершает процесс соединения. Это создает большую очередь из незаконченных соединений у сервиса и замедляет или делает невозможным обработку легитимных запросов.

Использование SYN Flooding атаки в комбинации с подменой IP адреса дает возможность атакующему тратить еще большие ресурсы облачной вычислительной среды, отправляю ответ на SYN запрос внутренним узлам облачной вычислительной среды. Для достижения большего эффекта пакеты с SYN запросами генерируются с заполненным содержимым пакетом и при достижении больших мощностей атак могут заполнить пропускной канал отдельных узлов коммутации облачной вычислительной среды, что повлечет за собой невозможность приема легитимного трафика.

HTTP Flooding [8]. Одной из наиболее простых версий HTTP атак является генерация множества запросов на скачивание большого файла веб-сервиса либо загрузки веб-страницы. В результате веб-сервис затрачивает большее количество ресурсов, что уменьшает доступные ресурсы другим пользователям облачной вычислительной среды, а также замедляется обработка легитимных запросов.

NTP Flooding [9]. Данная атака использует уязвимость серверов синхронизации времени. В последних версиях программ синхронизации данная уязвимость устранена, однако до сих пор существует достаточное количество мощных не обновлённых серверов. Атакующие отправляет множество запросов списка последних синхронизировавшихся с подмененным IP-адресом на IP-адрес жертвы, в результате чего сервер синхронизации генерирует очень большой поток данных на жертву, заполняя все ресурсы и пропускной канал.

DNS Flooding [10]. Данная атака использует уязвимость серверов DNS. На сервер отправляются небольшие запросы, на которые сервер отвечает большими пакетами, при этом IP-адрес отправителя заменен на IP-адрес жертвы. В итоге атакующему нет необходимости наращивать большую полосу пропускания для генерации атаки, так как объемный трафик генерируется с помощью DNS-серверов.

Chargen Flooding [11]. Данная атака использует протокол chargen. Это служба стека протоколов TCP/IP, определённая в RFC 864 в 1983 году

Джоном Постелом. Она предназначена для тестирования, измерения и отладки. Узел сети может установить соединение с сервером, поддерживающим Character Generator Protocol с использованием TCP или UDP через порт 19. После открытия TCP-соединения, сервер начинает отправлять клиенту случайные символы и делает это непрерывно до закрытия соединения. В UDP-реализации протокола, сервер отправляет UDP- датаграмму, содержащую случайное (от 0 до 512) количество символов каждый раз, когда получает датаграмму от узла. Все полученные сервером данные игнорируются. В итоге атакующему необходимо лишь отправить запрос на открытие соединения с сервером, на котором запущен данный протокол, подменив свой IP-адрес на IP-адрес жертвы. В результате сервер начнет генерацию пакетов со случайными символами на жертву. Достигнув высокой мощность с помощью данной атаки можно забить полосу пропускания на каком-либо из узлов перед жертвой, что сделает ее недоступной для легитимных запросов.

SSDP Flooding. SSDP описывает механизм, согласно которому сетевые клиенты могут обнаружить различные сетевые сервисы. Клиенты используют SSDP без предварительной конфигурации. SSDP поддерживает обнаружение при помощи мультикаста, уведомления от серверов и маршрутизацию. Данная служба включает обнаружение UPnP-устройств в домашней сети. Например, телевизор с поддержкой DLNA/UPNP находит медиа-серверы в локальной сети с использованием этого протокола. Домашние маршрутизаторы тоже, как правило, обнаруживаются компьютерами с помощью SSDP (для отображении информации о маршрутизаторах и медиа-серверах в "Сетевом окружении" эти устройства также должны поддерживать протокол HTTP, т. к. SSDP сообщает устройствам http-ссылку на узел управления устройством). Атакующий находит с помощью данного протокола доступные уязвимые узлы, после чего начинает посылать запросы с подменой на IP-адрес жертвы.

3. ОПИСАНИЕ ПРЕДПОЛАГАЕМОЙ АРХИТЕКТУРЫ

Метод обнаружения сетевых атак в облачных вычислительных средах состоит из трех этапов: подготовительного, первичного обучения, запуска системы защиты. После выбора точек расположения компонентов системы защиты и их установки оператор запускает первичное обучение компонентов обнаружения вредоносной активности. Это необходимо, так как в основе процесса обнаружения лежат модели интеллектуального анализа данных. Алгоритм обнаружения состоит из двух уровней для сокращения потребления ресурсов. На первом уровне проверяется наличие вредоносного трафике в вычислительной среде. Если модель классифицировала трафик как вредоносной, то активируется второй уровень глубинного анализа, на котором определяются источники атак и их жертвы. После первичного обучения моделей, они внедряются в систему, после чего система может работать в автоматическом режиме.

В условиях высоконагруженных облачных вычислительных сред в качестве данных для принятия решений по классификации трафика были выбраны данные о потоках трафика, а не о каждом пакете. Это позволяет существенно сократить время обработки трафика и генерации входных векторов для моделей интеллектуального анализа данных.

Для моделирования вредоносного и легитимного трафика был создан полигон, включающий в себя методы натурного и имитационного моделирования. Сценарии поведения клиентов задаются программно. Было создано 9 типов атак, сценарии для которых можно задать в конфигурационном файле. На любом участке имитируемой сети можно установить компонент защиты, сбора трафика, что дает возможность тестировать архитектурно-зависимые методы защиты. В качестве сценариев для легитимного трафика могут быть использованы сценарии атак, к примеру, HTTP Flooding, с меньшими нагрузками и частой запросов от одного клиента.

В качестве базовых типов атак в облачных вычислительных средах были выбраны следующие атаки: SYN Flooding, HTTP Flooding, NTP Flooding. Они являются основными атаками, которые используются на сегодняшний день, а также по данным о потоках трафика похожи на многие другие типы атаки, такие как Chargen, UDP Flooding и т.д.

Для разработки системы защиты и ее тестирования была выбрана платформа OpenStack. OpenStack является системой с открытым кодом и широко поддерживается в сообществе. Также данная система предлагает системный подход управления комплексными услугами платформы облачных вычислений, которые включают в себя: обеспечение безопасности, систему хранения данных, межсетевое взаимодействие, управление работой виртуальных машин, отслеживание действий, происходящих внутри проекта, при помощи информационной панели.

ЗАКЛЮЧЕНИЕ

В ходе работы были исследованные современные подходы к организации систем защиты в облачных вычислительных средах. Также были рассмотрены виды DDoS-аттак направленные на облачные вычислительные среды и описана предполагаемая архитектура для системы защиты облачных вычислительных сред.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. About SecuCloud. URL: https://secucloud.com/en/company/about-us (дата обращения: 03.12.2018).
2. Delimitrou C.; Kozyrakis C. Security Implications of Data Mining in Cloud Scheduling // IEEE Computer Architecture Letters. 2015. №PP. pp. 64-68.
3. Munz, G., Li, S., Carle, G.: Traffic Anomaly Detection Using K-Means Clustering // Proceedings of GI/ITG Workshop MMBnet. 2007. pp. 106-110.
4. Zolotukhin, M., Hamalainen, T., Kokkonen, T., et al.: Data Mining Approach for Detection of DDoS Attacks Utilizing SSL/TLS Protocol. Proceedings of 15th International Conference, NEW2AN 2015, pp. 274-285. St. Petersburg, Russia, (2015)
5. Kato K., Klyuev V. Large-scale network packet analysis for intelligent DDoS attack detection development // Internet Technology and Secured Transactions (ICITST), 2014 9th International Conference for. – IEEE, 2014. – С. 360-365.
6. AlSharafat M. S. et al. Adaptive steady state genetic algorithm for scheduling university exams // Networking and Information Technology (ICNIT), 2010 International Conference on. – IEEE, 2010. – С. 70-74.
7. Center C. C. TCP SYN flooding and IP spoofing attacks / C.C. Center //CERT Advisory CA-1996-21. – 1996. – P. 1996-2021.
8. Lu, W. Z. An HTTP flooding detection method based on browser behavior /W.Z. Lu, S.Z. Yu //2006 International Conference on Computational Intelligence and Security, Guangzhou, China, 3-6 November, 2006 y. – IEEE, 2006. – V. 2. – P. 1151-1154.
9. The flooding time synchronization protocol / M. Maroti [et al.] //Proceedings of the 2nd international conference on Embedded networked sensor systems, New York, 3-5 November, 2004 y. – ACM, 2004. – P. 39-49.

10. Detecting DNS amplification attacks / G. Kambourakis [etal.] //International Workshop on Critical Information Infrastructures Security, Malaga, Spain, 3-5 October, 2007 y. – Springer Berlin Heidelberg, 2007. – P. 185-196.
11. Mutu, L. Improved SDN responsiveness to UDP flood attacks /L. Mutu, R. Saleh, A. Matrawy //Communications and Network Security (CNS), 2015 IEEE Conference on, San Francisco, 29-31 October, 2015 y. – IEEE, 2015. – P. 715-716.