Обзор методов обнаружения распределенных атак типа "отказ в обслуживании" на основе машинного обучения и глубокого обучения

Т.М.Клименко, Р.Р.Акжигитов

***Аннотация*— Распределенные атаки типа "отказ в обслуживании" (DDoS) представляют серьезную угрозу сетевой безопасности. При атаке типа "Отказ в обслуживании" (DOS) атаку выполняет один источник, в то время как DDoS использует несколько хостов для атаки на систему. Очень трудно идентифицировать источник атаки, когда происходит такая атака, поскольку злоумышленник скрывает свою личность, подменяя свой IP-адрес. Как обнаруживать DDoS-атаки и защищаться от них, в настоящее время является актуальной темой как в промышленности, так и в научных кругах. В этой статье обсуждаются механизм DDoS атак и модели DDoS атак, основные методы запуска DDoS атак, типы атак согласно модели OSI и более подробное описание типов DDoS атак по направленности на определенную уязвимость. В данной статье систематизированы методы машинного и глубокого обучения, применяемые для обнаружения DDoS атак. Помимо описания самих методов, также приводятся примеры исследований, где данный метод применялся для обнаружения DDoS-атак. В конце статьи даны примеры сред, уязвимых к DDoS атакам. Данная статья поможет ознакомиться с современными эффективными методами обнаружения DDoS атак.**

***Ключевые слова* — DDoS атака, Машинное обучение, Глубокое обучение, UDP-флуд, ICMP-флуд, HTTP-флуд, OSI, Agent-Handler, Reflector, IRC.**

I. ВВЕДЕНИЕ

DOS-атака с участием более чем одного компьютера, направленная на скоординированную атаку на жертву, называется распределенной атакой типа "Отказ в обслуживании"(DDoS). Распределенная атака типа "отказ в обслуживании" (DDoS) является одной из самых серьезных угроз и одной из самых серьезных проблем безопасности, с которыми сталкивается сегодня Интернет. При DDoS-атаке злоумышленник обычно использует скомпрометированные компьютеры (называемые зомби), используя преимущества известных или неизвестных ошибок и уязвимостей для отправки большого количества пакетов от этих уже захваченных зомби на сервер. Это может занимать большую часть пропускной способности сети жертвы [1].

DDoS атака запускается косвенно через множество скомпрометированных вычислительных систем. Службы, подвергшиеся атаке, называются первичной жертвой, в то время как скомпрометированные системы, используемые для запуска атаки, часто называются вторичными жертвами. Использование второстепенных жертв в DDoS-атака дает возможность вести гораздо более масштабную и разрушительную атаку, сохраняя анонимность. Вторичные жертвы на самом деле совершают атаку и тем самым затрудняют поиск реального злоумышленника[2].

DDoS атака может быстро истощить вычислительные и коммуникационные ресурсы объекта за короткий промежуток времени. DDoS - это довольно разрушительная атака. В частности, хакеры часто используют DDoS-атаки для отправки огромных запросов в систему с намерением подавить обычный сервис, предоставляемый этой системой. Кроме того, существует определенное количество легкодоступных инструментов для DDoS-атак, что делает инициирование DDoS-атаки довольно простым делом. Например, HULK, Tor's Hammer, Slowloris, LOIC, Xoic, DDOSIM, RUDY и PyLoris могут быть использованы для запуска DDoS-атаки неопытным злоумышленником. DDoS обладает такими характеристиками, как простота инициирования, трудность защиты и сильная разрушительность[1].

Учитывая серьезность ситуации, был предложен целый ряд методов обнаружения DDoS атак, одними из которых являются методы обнаружения аномалий, основанные на машинном и глубоком обучении. В то время как методы обнаружения сигнатур могут обнаруживать атаки на основе сигнатур уже изученных атак, методы обнаружения аномалий изучают сетевой трафик на основе базового профиля и обнаруживают аномалии как те, которые значительно отклоняются от базового профиля. Методы обнаружения сигнатур эффективны против известных атак, в то время как обнаружение аномалий позволяет обнаруживать неизвестные и новые атаки (атаки нулевого дня) [1].

Статья получена 7 мая 2023.
Т.М. Клименко – магистр МГУ имени М.В. Ломоносова (email: s02210114@gse.cs.msu.ru).
Р.Р. Акжигитов – магистр МГУ имени М.В. Ломоносова (email: s02210028@gse.cs.msu.ru).

Машинное обучение можно разделить на классическое или «неглубокое» обучение и глубокое обучение. На рисунке ниже показана таксономия алгоритмов машинного обучения (Рис.1).

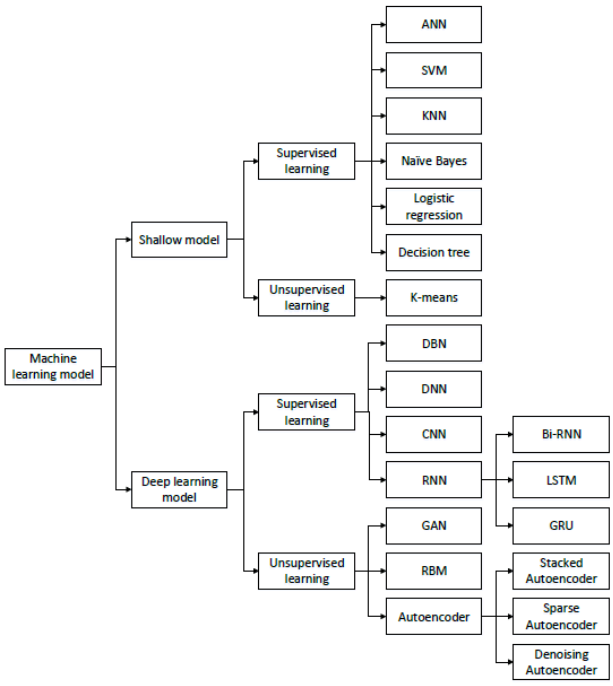

Рисунок 1 Таксономия методов машинного обучения для обнаружения DDos атак.

II. МЕХАНИЗМ DDOS АТАК

Для того чтобы выполнить DDoS-атаку, необходимо выполнить три шага. Сканирование - это первый шаг, на котором злоумышленник сканирует уязвимый компьютер с помощью различных стратегий сканирования. Следующим шагом является распространение, при котором злоумышленник набирает машины для генерации потока пакетов, которые будут выполнять атаку на уязвимую машину. Третий и последний шаг - это коммуникация, где могут быть применены три различные модели: модель агента-обработчика (Agent-Handler), модель рефлектора (Reflector) и IRC-модель[3]:

- **Модель агента-обработчика**: Эта типичная модель состоит из атакующего, обработчика, агента и целевой сети. Обработчики - это программные пакеты, расположенные по всему Интернету, которые злоумышленник использует для связи с агентами. Программное обеспечение агента существует в скомпрометированных системах, которые в конечном итоге осуществят атаку. Злоумышленник взаимодействует с обработчиками, чтобы определить, какие агенты запущены, когда планировать атаки или когда обновлять агентов)(Рис.2)[2].

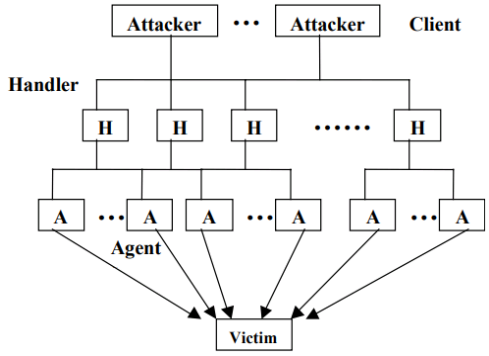

Рисунок 2 Модель агента-обработчика

- **Модель рефлектора(отражателя):** эта модель состоит из атакующего, обработчика, агента и рефлектора. Злоумышленники также имеют контроль над обработчиками, которые, в свою очередь, имеют контроль над агентами. Разница в этом типе атаки заключается в том, что обработчики заставляют агентов отправлять поток пакетов с IP-адресом жертвы в качестве исходного IP-адреса другим незараженным машинам, известным как рефлекторы, побуждая эти машины установить соединение с жертвой. Рефлектор - это любой хост, который реагирует на запросы, например веб-сервер, отвечающий на запросы TCP SYN ответом SYN-ACK. Любой хост может быть использован в качестве рефлектора путем подмены IP-адреса жертвы в поле источника запроса, обманом заставляя рефлектор направлять свой ответ жертве. Рефлекторы также можно использовать в качестве усилителей, отправляя пакеты на широковещательный адрес.
- **Модель на основе IRC:** Эта модель аналогична описанным выше моделям, за исключением того, что вместо использования программы-обработчика, установленной на сетевом сервере, для подключения клиента к агентам используется канал связи IRC (Internet Relay Chat)(Рис.3) [2].

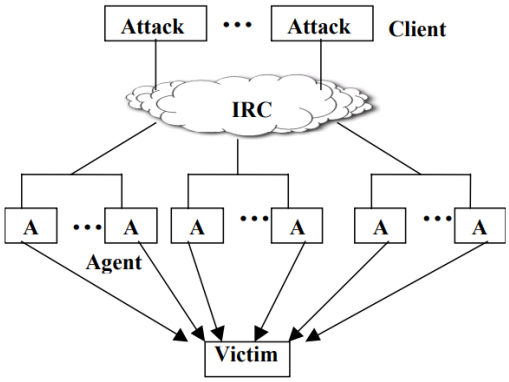

Рисунок 3 Модель на основе IRC

В настоящее время существует два основных метода запуска DDoS-атаки в Интернете.

Первый заключается в отправке жертве некоторых искаженных пакетов (т.е. атака на уязвимость).

Второй метод предполагает, что злоумышленник пытается выполнить одно или оба из следующих действий:

- Нарушить подключение законных пользователей, исчерпав пропускную способность, вычислительную мощность маршрутизатора или сетевые ресурсы. По сути, это атаки с затоплением на сетевом/транспортном уровне.

• Нарушить работу сервисов законных пользователей, истощая ресурсы сервера (например, сокеты, процессор, память, пропускную способность диска/базы данных и пропускную способность ввода-вывода). По сути, это атаки с затоплением на уровне приложений. [1]

III. ТИПЫ DDOS АТАК

Модель OSI (Open Systems Interconnection) полезна для понимания типов DDoS-атак, с которыми мы имеем дело. DDoS-атаки нацелены на определенные уровни модели сетевого взаимодействия (атаки прикладного уровня нацелены на уровень 7, атаки протокольного уровня нацелены на уровни 3 и 4)[1]. Таблица ниже показывает примеры атак на разные уровни модели OSI.

Таблица 1 Уровни модели OSI с примерами DDoS атак

Уровень сетевой модели OSI	Пример DDoS атаки
Прикладной	HTTP POST и GET
Представительный	Некорректные SSL запросы
Сеансовый	Telnet DDOS
Транспортный	Smurf , SYN-флуд атака
Сетевой	ICMP-флуд атака
Канальный	MAC-флуд атака
Физический	Неисправность физического оборудования

Также DDoS атаки можно разделить по и по направленности на определенную уязвимость или по цели атаки на атаки на истощение ресурсов и атаки на истощение пропускной способности.

A. Атаки на истощение ресурсов

Целью атак с истощением ресурсов является переполнение или сбой всех основных ресурсов системы, таких как память, сокеты и центральный процессор[4] Ресурсы потребляются таким образом, что для законных пользователей ничего не остается. Атака на истощение ресурсов предназначена для того, чтобы использовать ресурсы жертвы и сделать систему неспособной обработать законный запрос на обслуживание[2] Существует два различных способа реализации атак такого типа. В первом случае злоумышленник использует некоторые сети, протоколы транспортного и прикладного уровней для достижения своих целей. Во втором способе для выполнения атак используются искаженные пакеты[4]

- **Атаки с использованием протокола.** Существует несколько крупных атак, основанных на протоколах, которые используют слабые места протоколов различных сетевых уровней. Это вынуждает жертву использовать весь свой процессор и память для выполнения некоторых операций, требующих больших затрат памяти. Например, атаки этой группы используют протоколы транспортного уровня, такие как протокол управления передачей (TCP), и некоторые протоколы прикладного уровня, такие как протокол передачи гипертекста (HTTP) и протокол инициации сеанса (SIP) при выполнении атак [4].

(1) Медленные HTTP-атаки. "Медленная" атака направлена на то, чтобы медленно истощить все ресурсы жертвы. Атака Slowloris позволяет веб-серверу сосредоточить атаку на другом сервере, не затрагивая другие службы или порты в целевой сети. Таким образом, это позволяет проводить очень целенаправленную атаку на один конкретный сервер. Slowloris устанавливает несколько подключений к целевому серверу и удерживает эти подключения открытыми как можно дольше. При Slowloris выполняется неполный HTTP-запрос, гарантируя, что сокеты останутся открытыми. Впоследствии сервер отбрасывает все подлинные запросы, что приводит к отказу в обслуживании. Это можно свести к минимуму, установив ограничение скорости передачи данных от клиента. R.U. Dead Yet - еще одна атака. Эта атака запускается, используя области отправки форм на веб-сайтах. R.U. Dead Yet вводит только один байт информации в поле POST приложения за раз, а затем ожидает, тем самым заставляя потоки приложения ждать окончания бесконечных сообщений. R.U.D.Y. открывает несколько одновременных подключений, что в конечном итоге приводит к исчерпанию таблицы подключений сервера.[5] Система завершит работу с ошибкой и откажет в обслуживании пакетов законных пользователей.

(2) HTTP-флуд-атаки. Атаки нацелены на уровень, на котором веб-страницы генерируются на сервере и доставляются в ответ на HTTP-запросы. Выполнение одного HTTP-запроса на стороне клиента обходится дешево с точки зрения вычислений, но ответ на него может быть дорогостоящим для целевого сервера, поскольку сервер часто загружает несколько файлов и выполняет запросы к базе данных для создания веб-страницы. Злоумышленник использует ботов (скомпрометированные устройства) для отправки огромного количества запросов, расширяя масштабы атаки[5]. HTTP-флуд не использует искаженные пакеты, методы подмены или отражения и требует меньшей пропускной способности, чем другие атаки, для отключения целевого сайта или сервера. Атака наиболее эффективна, когда она вынуждает сервер или приложение выделять максимально возможные ресурсы в ответ на каждый отдельный запрос.Злоумышленник отправляет большой объем HTTP-запросов GET и POST, которые не обрабатываются сервером, но веб-сервер резервирует ресурсы для этого запроса. Это приводят к отказу в дополнительных подключениях от законных клиентов. веб-сервер резервировать ресурсы для этого запроса. Атака HTTP flood может быть осуществлена двумя способами[5].

- ❖ Атака HTTP GET происходит, когда хакер использует ботнет для отправки множества запросов GET на сервер жертвы для получения файлов, изображений и т.д. Сервер будет оставаться занятым, отвечая на эти запросы со всех компьютеров-жертв в ботнете, избегая отбрасывания подлинных запросов. Злоумышленник может вставить встроенное изображение в содержимое веб-страницы,

таким образом, любой ничего не подозревающий пользователь может принять участие в атаке и отправить непреднамеренный запрос GET на сервер жертвы, просто просмотрев веб-сайт.

- ❖ Атака HTTP POST включает в себя использование злоумышленником ботнета для ввода форм на веб-сайте. В сочетании с информацией, которую запросы передают с большого количества зараженных компьютеров, в конечном счете сервер становится перегруженным, что приводит к инциденту с отказом в обслуживании.

(3) SIP-флуд-атаки. Эта атака нацелена на SIP (Session Initiation Protocol) серверы регистрации и использование всех его ресурсов, а также пропускной способности сети, центрального процессора и памяти. Эта атака затопляет систему, не позволяя подлинным пользователям подключаться и причиняя неудобства[5]. Атака может быть осуществлена с использованием различных типов сообщений SIP-запроса (таких как SIP REQUEST, SIP INVITE) или сообщений управления SIP-вызовом (SIP INFO, SIP NOTIFY, SIP RE-INVITE. Ботнет отправляет тысячи сообщений на сервер SIP-регистратора, который ведет учет адресов и параметров пользовательских агентов. В результате сервер перегружается; законные пользователи испытывают перебои в обслуживании и не могут связаться с сервером[4].

(4) Атаки TCP SYN. В атаке TCP SYN злоумышленник использует механизм трехстороннего установления связи в процессе установления соединения TCP. Во время установления соединения TCP требует последовательных подтверждений между двумя сторонами, которые хотят создать TCP-соединение. Это достигается с помощью трехстороннего рукопожатия. При трехстороннем обмене данными сначала пакет SYN отправляется от клиента на сервер, чтобы начать обмен данными. После получения этого пакета SYN сервер подтверждает клиента, отправляя пакет SYN+ACK. Наконец, в качестве ответа на этот пакет клиент отправляет обратно окончательный пакет ACK, который завершает передачу данных и устанавливает TCP-соединение. Во время этого процесса сервер сохраняет все промежуточные состояния в стеках памяти до тех пор, пока не будет установлено соединение или не истечет тайм-айт. Чтобы завладеть памятью жертвы, злоумышленник не завершает процесс установления связи и таким образом создает огромное количество неполных подключений. Чтобы установить это неполное соединение, злоумышленник подделывает источник несуществующими IP-адресами и отправляет пакеты SYN с этими поддельными IP-адресами. После получения пакетов, сервер отвечает пакетами SYN+ACK, но поскольку исходные IP-адреса не существуют, он никогда не получает пакеты ACK от источников. Однако, поскольку сервер ожидает пакетов ACK, в конечном итоге все его таблицы подключений заполняются. Целевой хост резервирует ресурсы для каждого полуоткрытого соединения, ожидая подтверждения, которое так и не приходит. В конечном счете, все ресурсы зарезервированы, и никакие новые подключения не могут быть установлены. Таким образом, злоумышленник заполняет память жертв и успешно лишает доступа законных пользователей. Также возможно осуществить эту атаку, используя подлинный IP-адрес скомпрометированных машин. В этом случае скомпрометированные исходные компьютеры игнорируют сообщения SYN+ACK, полученные от жертвы, и, таким образом, могут выполнить успешную SYN-атаку [4].

- **Атаки с использованием искаженных пакетов.**

Основная идея атаки с искаженным пакетом заключается в атаке на жертву с использованием искаженного пакета, который может сбить жертву с толку и, как следствие, привести к сбою системы[4].

(1) Land-атаки. Эта атака происходит из-за образования бесконечного цикла. Злоумышленник настраивает адрес источника пакета так, чтобы он был IP-адресом цели. Когда цель или система реагирует на пакет, она эффективно реагирует сама на себя, создавая бесконечный цикл. В конце концов, система выходит из строя[5],[4] Таким образом, Land-атака похожа на SYN-атаку, с той лишь разницей, что вместо неправильного IP-адреса используется IP-адрес самой целевой системы[7].

(2) Ping of Death атака. При атаке ping of death злоумышленник, используя простую команду ping, намеренно формирует пакет данных, превышающий максимальный размер пакета, что приводит к зависанию или аварийному завершению работы жертвы[4]. Атакующий создает IP-пакет, размер которого превышает максимальный размер стандарта IP в 65 536 байт. Пакеты огромных размеров разделяются на небольшие сегменты, прежде чем быть отправленными в виде нескольких пакетов. Злоумышленник отправляет много больших пакетов цели, которая восстанавливает их и превышает лимит в 65 536 байт[5]. Достижение предельного значения приводит к переполнению памяти, что приводит к сбою системы. Когда система выходит из строя, она становится более уязвимой для других атак, например, для атаки троянского коня[5] Эта атака может быть инициирована только злоумышленником без необходимости использования ботнета[4].

(3) Атаки фрагментации UDP. В этой атаке злоумышленник отправляет ложные пакеты, размер которых достаточно велик для фрагментации и для повторной сборки в пункте назначения [8]. Сервер не в состоянии восстановить пакеты, используя свои ресурсы, поскольку они превышают предельный размер[5]. Безуспешные попытки собрать эти поддельные пакеты заново и фрагментировать приводят к перегрузке устройства, и, следовательно, сервер отказывается обрабатывать другие пакеты. Этот тип атаки иногда называют атакой "Пинг смерти", поскольку злоумышленник продолжает посылать множество искаженных “пингов”, которые при повторной сборке после фрагментации превышают максимальную длину IP-пакета на канальном уровне, что приводит к переполнению буферов памяти, вызывая атаку типа "Отказ в обслуживании"[8].

(4) Teardrop-атаки. Атака происходит, когда злоумышленник доставляет в систему поврежденные пакеты. Из-за ошибки в сборке TCP/IP появляются

разорванные пакеты с перекрывающимися значениями смещения. Когда пакеты накладываются друг на друга, целевая система выходит из строя [5]. Эта атака включает в себя манипулирование значением смещения, что, в свою очередь, порождает ошибки при фрагментации и повторной сборке пакетов. По сути, злоумышленник отправляет фрагментированные пакеты с перекрывающимися номерами смещений. Таким образом, во время повторной сборки пакета создаются недопустимые пакеты, которые приводят к аварийному завершению, зависанию или перезагрузке целевого компьютера [4],[7]

B. Атаки на истощение пропускной способности

Атака на истощение пропускной способности - еще один важный тип атак в мире DDoS. Цель злоумышленника - использовать всю пропускную способность сети системы жертвы, используя армию атакующих. В результате, затопленная нежелательным трафиком, жертва отказывает в обслуживании законным пользователям на небольшой или большой промежуток времени, пока атака не будет устранена[4] Чтобы усилить атаку, пакеты атаки могут быть амплифицированы или переданы по широковещательной связи. Методы наводнения и усиления являются хорошо известными методами этой атаки. Атака наводнением включает в себя отправку зомби больших объемов трафика системе-жертве, чтобы заполнить пропускную способность сети системы-жертвы IP-трафиком. Система-жертва замедляет работу, выходит из строя и предотвращает доступ законным пользователям. Примерами таких атак являются UDP-флуд и ICMP-флуд. Атака с усилением включает в себя атакующего или зомби, отправляющего сообщения на широковещательный IP-адрес, благодаря чему все системы в подсети получают сообщения с широковещательного адреса и таким образом отправляют ответ системе-жертве. Примерами таких атак являются Smurf и Fraggle. До тех пор, пока атака не будет распознана и обработана, настоящие пользователи будут сталкиваться с отказом в обслуживании[2]. Также можно выделить 2 категории:

- **Атаки с использованием протокола.** Атаки с использованием протокола могут использовать протокол транспортного уровня, такой как User Datagram Protocol (UDP), или протокол сетевого уровня, такой как Internet Control Message Protocol (ICMP)[4].

(1) UDP-флуд-атаки. Хакер предоставляет указания, а именно адрес цели, продолжительность атаки и механизм, используемый для проведения атаки с нескольких скомпрометированных машин. Злоумышленник может сначала передать указания в главное управляющее программное обеспечение, которое будет транслировать инструкции по атаке мастерам, заставляя их передавать несколько UDP-пакетов с поддельным интернет-протоколом (IP). В свою очередь, цель будет передавать пакеты протокола Internet Control Message Protocol (ICMP) в качестве требуемого ответа на поддельный адрес, но она никогда не получит ответа. Из-за огромного количества полученных пакетов и отсутствия ответа целевой компьютер будет продолжать работать медленно и в конечном итоге выйдет из строя[5] UDP-поток используется для заполнения случайных портов на удаленном хосте многочисленными UDP-пакетами. . Затем хосту необходимо проверить, нет ли приложения, прослушивающего этот порт, и, если приложение не найдено, ответить пакетом ICMP "Пункт назначения недоступен". Для фильтрации или блокирования вредоносных UDP-пакетов используются специализированные брандмауэры[4],[6].

(2) ICMP-флуд-атаки. Атака ICMP flood, также известная как ping flood attack, использует пакеты ICMP_ECHO_REQUEST протокола ICMP. Этот пакет (ping) используется для проверки того, активен удаленный хост или нет. При DDoS-атаках злоумышленник отправляет этот пакет, используя широковещательный IP-адрес. Таким образом, он доставляется на все компьютеры в сети жертвы. Машины ответят на поддельный адрес источника, предназначенный для жертвы, пакетом ICMP_ECHO_REPLY. Кроме того, злоумышленник может использовать посредническую сеть, чтобы завалить жертву. Существует несколько команд ping, таких как n, l, t, где команда n - это количество отправленных запросов, команда l сообщает нам объем данных, отправленных в пакете, а команда t используется для ping данных. Результаты ping могут сообщить вам о некоторых проблемах с подключением, и с этого момента вы можете приступить к устранению неполадок. Ping отправляет небольшой пакет информации целевому сетевому ресурсу (например веб-сайту), и этот ресурс отправляет обратно пакет информации аналогичного размера. Поток ping - это просто поток запросов ping, настолько большой, что пропускная способность сети целевой системы забивается при попытке ответить на каждый запрос, и, следовательно, отклоняет запросы от законных пользователей[4],[6].

(3) Атака Smurf. Примером атаки типа ICMP-флуд является атака ‘Smurf’, которая использует некоторые промежуточные сети, также известные как отражатели, для усиления атаки[4]. ICMP обычно отвечает за генерацию сообщений об ошибках, информирующих источник о любом сбое, произошедшем в сети или в пункте назначения, например, когда шлюз не в состоянии буферизировать данные или когда пакет недоступен для назначения. Функция Ping выполняется в ICMP, которая инициирует эхо-запрос, на который выдается эхо-ответ. Если этот ответ не получен, это указывает на то, что другой хост не активен или не имеет функции ping. В этом сценарии эхо-запросы продолжают отправляться без ожидания эхо-ответа и наводняют сеть, излишне потребляя полосу пропускания. Это также известно как атака Smurf[8]. Атака наводняет компьютер цели эхо-ответными сообщениями большого объема, в результате чего машина замедляется и, в конечном итоге, становится непригодной для использования[5].

(4) Атака Fraggle похожа на атаку Smurf. Эта атака отправляет пакеты UDP_ECHO на сетевые усилители, чтобы заполнить пропускную способность, или может отправить их на определенный порт, чтобы создать бесконечный цикл. Атаки Smurf и Fraggle используют

отражатель(ы) в качестве их пусковых установок для атаки. Любой IP-узел, который возвращает пакет в ответ на принятый пакет, известен как отражатель. Таким образом, маршрутизаторы, DNS-серверы или веб-серверы являются примерами отражателей. Эти отражатели запускают атаку, отправляя жертве ответы в качестве ответов на принятые пакеты, которые содержат поддельный IP-адрес источника в качестве IP-адреса жертвы. Отражатели используют свои собственные законные IP-адреса, поэтому их можно обнаружить. Однако злоумышленник, который задействует отражатели в атаке, остается скрытым поскольку он подделал свой исходный IP-адрес на IP-адрес жертвы[4].

- **Атаки с усилением.** Основная идея, лежащая в основе этих типов атак, заключается в генерировании большого ответа на очень маленький запрос и направлении этих ответов жертве, которые в конечном итоге потребляют всю пропускную способность сети жертвы[4]. В этой атаке хакер доставляет пакеты небольшого размера, но за счет их амплификации (усиления); он отправляет большое количество пакетов цели, используя всю доступную полосу пропускания. Атака с усилением DNS и атака по протоколу сетевого времени (NTP) являются двумя примерами таких атак.

(1) Атаки с усилением DNS. Хакер отправляет DNS-поисковый запрос на DNS-серверы с поддельным IP-адресом, то есть адресом жертвы. DNS-сервер отвечает, отправляя запись. DNS-атака является атакой с усилением, поскольку объем запроса превышает размер ответа. Поскольку ответы являются законными ответами сервера, невозможно определить, были ли пакеты отправлены злоумышленником или авторизованными пользователями[5],[7] Эта атака также является примером атаки отражения, которая использует несколько открытых рекурсивных DNS-серверов для отправки огромного количества UDP-пакетов, чтобы затопить жертву. Используя различные типы методов, связанных с усилением, злоумышленник может увеличить объем атакующего трафика, что может привести к катастрофическим последствиям для наиболее защищенной системы жертвы. Следовательно, при увеличении размера этого ответного сообщения потребляется больше полосы пропускания, чем в обычной ситуации. Для достижения этой цели злоумышленник может скомпрометировать авторитетный DNS-сервер. Здесь, сначала, злоумышленник, используя авторитетный DNS-сервер, раскрывает значительную запись ресурса (RR) типа TXT. Этот текстовый тип RR используется в DNS, чтобы сделать возможным хранение информации, такой как имя хоста, название сервера или некоторой информации о сервере или центре обработки данных. Злоумышленник манипулирует этой записью, чтобы увеличить размер сообщения. Затем злоумышленник запускает атаку с помощью ботнета и инструктирует ботов отправлять запросы на DNS-рекурсивные серверы. DNS-серверы отправляют усиленные ответы на поддельный исходный адрес, который на самом деле является адресом жертвы. Кроме того, для усиления атаки злоумышленник может отправить DNS-запрос, используя расширенный протокол DNS (EDNSO). Это расширение поддерживает большие DNS-сообщения[4],[48]

(2) Атаки с усилением NTP. Протокол NTP используется для синхронизации локальных и глобальных часов через Интернет. Однако он может быть использован для запуска DDoS-атак путем запроса запроса с поддельным исходным адресом. Хакер использует поддельный IP-адрес для доставки амплифицированных пакетов данных цели по протоколу NTP UDP. Утилита “monlist” на сервере NTP может быть использована для запуска атаки на сервер NTP[5] Злоумышленник отправляет команду MON_GETLIST на NTP-сервер[4]. Запрос требует ответа большего размера, такого как список последних 600 подключенных к серверу хостов, который используется для определения фактического времени в UTC[8]. Таким образом, ответное сообщение является огромным (примерно в 19 раз больше) по сравнению с сообщением запроса. Злоумышленник подделывает исходный IP-адрес и направляет все эти ответы на компьютер жертвы. Таким образом, эти усиленные ответы перегружают пропускную способность сети жертвы и препятствуют законному пользователям для доступа к серверу[4]

(3) Атаки с усилением CLDAP. Атака с усилением протокола light directory access без подключения к интернету отправляет поддельные пакеты на сервер CLDAP через UDP-порты. Сервер возвращает ответ на поддельный адрес. Одной из возможных атак с усилением является ответ, размер которого в 46-55 раз превышает размер подлинного пакета [5]

IV. МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ОБНАРУЖЕНИЯ DDOS АТАК

В этой главе представлены методы машинного обучения, используемые для обнаружения Ddos-атак.

1. Логистическая регрессия (Logistic Regression или LR)

LR - это тип логарифмической линейной модели. Алгоритм LR считает вероятности различных классов с помощью параметрического логистического распределения, вычисляемого как (Формула 1):

$$P(Y = k|x) = \frac{e^{w_k * x}}{1 + \sum_{k}^{K-1} e^{w_k * x}}$$

Формула 1 Функция, характеризующая искомую вероятность

где k = 1,2...K - 1. Выборка x отнесена к классу максимальной вероятности[9]. Это надежная процедура для решения задачи бинарной классификации. Логистическая регрессия используется для прогнозирования вероятности результата, имеющего только два значения. Главным компонентом логистической регрессии является логистическая функция - S-образная кривая, берущая любое вещественное число и отображающая это число в значение от 0 до 1, хотя никогда точно не в 0 и 1[10]. 0,5 считается пороговым значением, а значения,

превышающие 0,5, считаются равными 1, а ниже 0,5 – равными 0 [11]. LR не может хорошо работать с нелинейными данными, что ограничивает его применение [9]. Bakhareva и др. (2019) применили логистическую регрессию к датасету CICIDS2017 и достигли точности 0,918 в обнаружении DDoS атак [12].

2. *Наивный Байес (Naïve Bayes или NB)*

Наивный Байес - это не один алгоритм, а совокупность множества алгоритмов, которые основаны на теореме Байеса[13]. Наивный Байес - это вероятностный классификатор с сильными допущениями независимости между признаками. Условное свойство может быть разложено с помощью теоремы Байеса следующим образом(Формула 2):

$$P(C_k \mid X) = \frac{P(X \mid C_k)\,P(C_k)}{P(X)}$$

Формула 2 Формула Байеса

где X = (x_1,…, x_n) представляет вектор из n независимых

признаков, а C_k представляет каждый класс [14].

P (C_k|X) - Апостериорная вероятность класса при задании предиктора (атрибута)

P (C_k) - Априорная вероятность класса,

P (X|C_k) - Вероятность предиктора при задании класса,

P (X) - Априорная вероятность предиктора [10]

Используя теорему Байеса, мы можем определить вероятность того, что произойдет какое-то событие, так, как будто подобное событие уже произошло. Это довольно простое уравнение; мы не можем напрямую получить значение P(C_k|X), но мы можем получить значение P(X|C_k) и P(C_k) из обучающих данных. Мы делаем наивное предположение, что все признаки независимы. Исходя из этого наивного предположения, мы можем классифицировать параметры с меньшим количеством признаков. Поскольку мы можем сокращать количество призаков, этот метод действительно хорошо справляется с любыми многомерными данными [15]. Roopak и др. (2020) оценили точность Наивного Байеса для обнаружения DDoS-атак на основе набора данных CICIDS2017, которая достигла 94.19 % [16]

3. *Метод опорных векторов (Support Vector Machines или SVM)*

SVM генерирует гиперплоскость или несколько гиперплоскостей в многомерном пространстве. Лучшая гиперплоскость - это та, которая оптимально разбивает данные на разные классы. Нелинейный классификатор использует множество функций ядра для определения границ между гиперплоскостями. Основная цель этих функций ядра, таких как линейная, полиномиальная, радиальный базис и сигмоидальная, состоит в максимизации границ между гиперплоскостями [17]. В алгоритме этого типа каждый элемент данных помечается как точка в n-мерном пространстве, где n - количество объектов, причем каждый объект является значением конкретной координаты. Между различными классами рисуются разные выступы (margins), и гиперплоскость является такой, в которой среднеквадратичная ошибка минимизирована, а расстояние между выступом и классами максимизировано. Гиперплоскость выбирается таким образом, чтобы расстояние между гиперплоскостью и ближайшей к ней точкой данных было максимальным[18]. Метод опорных векторов выполняет поиск объектов, расположенных на границах классов (по крайней мере, двух), т.е. опорных векторов, и решает задачу нахождения разделения множества объектов на классы с использованием линейной решающей функции. Метод опорных векторов строит функцию классификации f(x) следующим образом (Формула 3):

$$f(x) = sign(\langle w, s \rangle + b)\ ,$$

Формула 3 Функция классификации

где < , > – скалярное произведение,

w – вектор нормали (перпендикуляра) к разделяющей гиперплоскости,

b –вспомогательный параметр, равный абсолютному модулю расстояния от гиперплоскости до начала координат. Если параметр b равен нулю, гиперплоскость проходит через начало координат.

В пространствах высокой размерности вместо прямых линий следует рассматривать гиперплоскости, размерность которых на одно измерение меньше, чем у предполагаемого пространства. В R3, например, гиперплоскость - это двумерная плоскость (Рис.2) [19].

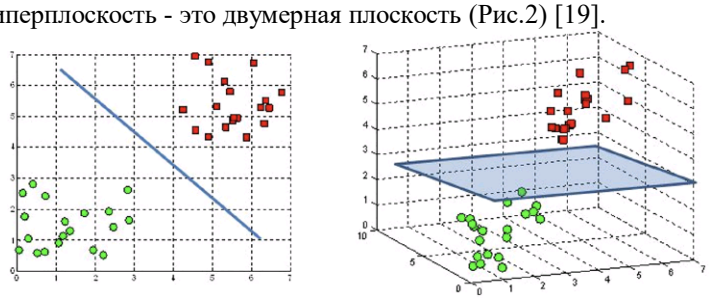

Рисунок 4 Гиперплоскость в R^2(слева) и R^3 (справа) пространствах.

Roopak и др. (2020) оценили точность SVM для обнаружения DDoS-атак на основе набора данных CICIDS2017, которая достигла 94,50 % [16]. Kasim (2020) также исследовал применение SVM к обнаружению DDoS атак, на датасете CICIDS2017 AE-SVM метод достиг точности 0.9941 и PCA-SVM метод достиг точности 0.9388 по сравнению с SVM, который достиг точности 0.9463 [20].

4. *Метод k-ближайших соседей (K-Nearest Neighbors или KNN)*

Этот метод основан на том, что похожие объекты расположены рядом друг с другом. Чтобы этот метод работал эффективно, необходимо тщательно выбирать K. Его можно подобрать, запустив эту модель с различными значениями K и выбрав наилучший вариант с лучшей робастностью и меньшим количеством ошибок [15]. KNN использует данные для поиска сходства между имеющимися данными и новыми данными [11]. KNN сохраняет доступные объекты и классифицирует новые объекты на основе функции расстояния. Объект

классифицируется большинством голосов его соседей. После этого объект присваивается классу, который имеет наибольшее к нему сходство среди K ближайших соседей. Некоторыми функциями для определения расстояния являются Евклидова, Манхэттенская и Минковского метрики расстояния [10]:

$$\text{Euclidean} \quad - \quad \sqrt{\sum_{i=1}^{k}(x_i - y_i)}$$

$$\text{Manhattan} \quad - \quad \sum_{i=1}^{k} |x_i - y_i|$$

$$\text{Minkowski} \quad - \quad \left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$$

Ramadhan и др. (2020) использовали алгоритм KNN для обнаружения DDoS атак. Точность составила 98.54% на CICIDS2017 датасете [21].

5. Дерево решений (Decision Tree или DT)

DT классифицирует выборку с помощью последовательности решений, представленных в виде древовидной структуры, в которой текущее решение помогает принять последующее решение [11].

Классификация экземпляров выполняется путем проверки атрибута, идентифицированного этим узлом, начиная с корневого узла и продолжая вниз по ветви дерева, которая соответствует значению атрибута. Наиболее часто используемым критерием расщепления является "энтропия" для примеси Джини (gini impurity) (Формула 4):

$$Entropy = -\sum_{i=1}^{n} p(xi)\, Log_2\, p(xi)$$

Формула 4 Энтропия

и "Коэффициент Джини" для прироста информации (knowledge gain) (Формула 5)[17]:

$$Gini = 1 - \sum_{i=1}^{n} pi^2$$

Формула 5 Коэффициент Джини

Весь путь от корня до листа представляет собой правило классификации. Дерево решений в основном группирует атрибуты по разным классам путем сортировки значений, связанных с атрибутами [22]. Дерево решений состоит из трех типов узлов:

- Корневой узел- это самый верхний узел. У него нет входящего ребра, но есть ноль или более исходящих ребер.
- Внутренний узел - имеет ровно одно входящее ребро и два или более исходящих ребра.
- Конечный узел - имеет ровно один входящий узел и ни одного исходящего ребра [23].

Каждая вершина (узел) дерева представляет атрибут, и каждая ветвь определяет значение, которое может иметь этот атрибут. Самая верхняя вершина в дереве называется корнем, которая содержит наибольший прирост информации (различия в энтропии) среди всех признаков и используется для оптимального разделения всех обучающих данных. Нижние узлы называются листьями. Каждый лист представляет класс. Во время классификации DT перемещается сверху вниз, удовлетворяя экземпляру, который необходимо классифицировать. Уравнение прироста информации, используемое в DT для оптимального разделения экземпляров в древовидно структурированным образом, приведено ниже(Формула 6)[24]:

$$Gain(P, Q) = Entropy(P) - \sum_{v \in D_Q} \frac{|P_v|}{|P|} Entropy(P_v)$$

Формула 6 Прирост информации

Здесь выигрыш (P,Q) - это уменьшение энтропии для сортировки P по атрибуту Q. Признаки с увеличивающимся значением прироста информации выбираются в качестве узлов нисходящим образом. Во время построения DT, предварительная (pre-pruning) и последующая (post-pruning) обрезка помогают предотвратить чрезмерную или недостаточную подгонку модели. Наконец, древовидная структура преобразуется в некоторый набор правил для классификации или прогнозирования новых экземпляров [24]. Модель похожа на дерево, что делает ее интерпретируемой. Алгоритм дерева решений может автоматически исключать нерелевантные и избыточные признаки. Процесс обучения включает в себя выбор признаков, генерацию дерева и обрезку деревьев. При обучении модели дерева решений алгоритм выбирает наиболее подходящие признаки по отдельности и генерирует дочерние узлы из корневого узла. Дерево решений - это базовый классификатор. Некоторые продвинутые алгоритмы, такие как случайный лес и экстремальный градиентный бустинг (XGBoost), состоят из множества деревьев решений [9]. Tuan и др. (2019) оценили DT алгоритм в обнаружении Ботнет DDoS атак на UNBS-NB15 датасете. Данный метод набрал 94.43% точности[25].

6. Случайный лес (Random Forest или RF)

Случайный лес (RF) - это алгоритм машинного обучения, который сочетает в себе две идеи: дерево решений и ансамблевое обучения. Лес содержит множество деревьев принятия решений, которые используют случайно выбранные атрибуты данных в качестве входных данных [14]. Предсказание класса выполняется каждым отдельным деревом путем голосования большинством голосов или взвешенного голосования. Класс, набравший наибольшее количество голосов, является предсказанием нашей модели. Одним из преимуществ случайного леса является то, что дисперсия модели уменьшается с увеличением количества деревьев в лесу, в то время как смещение остается неизменным [13],[14].

RF, как правило, намного более точен, чем отдельный классификатор. Как правило, чем больше деревьев в лесу, тем более робастным он является. Переобучение является одной из наиболее распространенных проблем в ML, но RF-классификатор не будет переобучать модель, если в лесу достаточно деревьев[11]. DT очень чувствителен к изменениям в данных. Если какое-либо изменение внесено в данные без изменения условий, то DT может привести к неправильному решению. Эта проблема была преодолена в RF-модели. Поскольку у RF есть много деревьев решений, каждое дерево может работать независимо при принятии решений со случайным набором данных. Случайные наборы данных создаются путем изменения частоты вхождений данных без

изменения длины данных. Таким образом, все деревья получают данные одинакового размера. Корреляция между деревьями должна быть меньше, чтобы повысить точность выходных данных. Если корреляция между деревьями высокая, то вероятность распространения ошибочно принятого решения тоже высокая, что снижает точность алгоритма. Затем, основываясь на голосовании большинства всех деревьев (Рис.3), принимается окончательное решение [15].

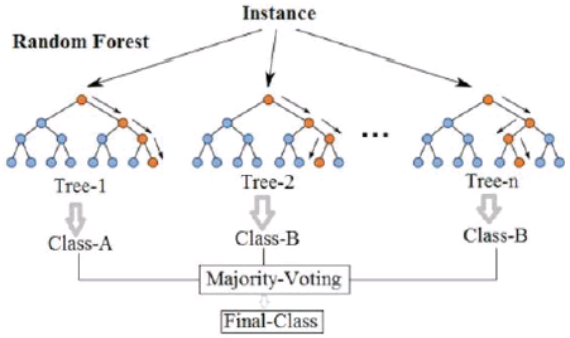

Fig. 1. Random Forest Information

Рисунок 5 Random forest классификатор

Roopak и др. (2020) оценил точность Random Forest для обнаружения DDoS-атак на основе набора данных CICIDS2017, которая достигла 93.64 % [16]. Min и др.(2018) использовали эмбеддинг слов и сверточную нейронную сеть текста (Text-CNN) для извлечения признаков и RF для классификации. Производительность метода оценивали с помощью ISCX2012 датасета. Полученная точность в обнаружении DDoS атак равняется 98,09% [26].

7. Метод k-средних (K-means)

Это один из неконтролируемых алгоритмов ML и в этом методе нет размеченных данных. Этот алгоритм работает на основе поиска групп в данных. Он группирует объекты в кластеры на основе их сходства и различий с объектами в других кластерах [11].

Данные, обладающие схожими характеристиками, группируются в один и тот же кластер (Рис. 4). Алгоритм назван K-means, потому что он создает K различных груп. Kmeans состоит из следующих шагов:

1. Установите начальные центроиды.
2. Затем отнесите каждый объект к соответствующей группе, которая имеет ближайший центр тяжести.
3. После назначения всех объектов снова вычислите положения k центроидов.
4. Затем повторяйте шаги 2 и 3 до тех пор, пока центроиды не перестанут двигаться [18],[27].

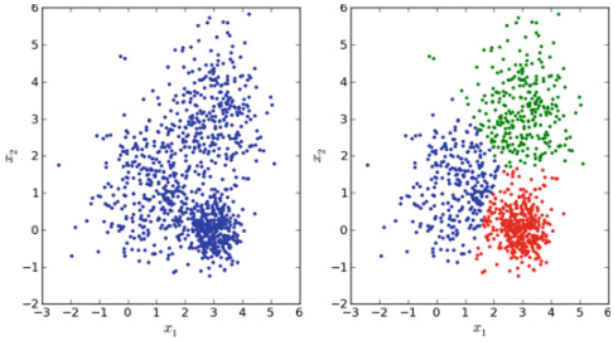

Рисунок 6 Данные до (слева) и после (справа) применения K-means

K-means - это типичный алгоритм кластеризации, где K - количество кластеров, a means - среднее значение атрибутов. Алгоритм K-средних использует расстояние в качестве критерия меры подобия. Чем меньше расстояние между двумя объектами данных, тем больше вероятность того, что они будут размещены в одном кластере [9]. Zekri и др. (2017) получили точность 95.9 % на наборе данных в реальном времени, используя метод K-Means для обнаружения DDoS атак [1]

8. Метод главных компонент (Principal Component Analysis или PCA)

Анализ основных компонентов - это неконтролируемый алгоритм машинного обучения, в котором объем данных сокращается, что делает вычисления более доступными и быстрыми. PCA преобразует двумерные данные в одномерные (Рис.5). Это делается путем преобразования набора переменных в новые, известные как главные компоненты (PC). Набор данных, к которому применяется алгоритм PCA, должен быть масштабирован, поскольку результаты чувствительны к масштабированию [18].

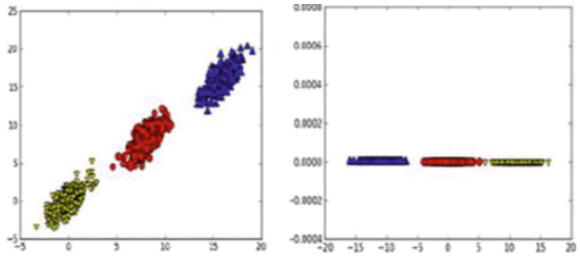

Рисунок 7 Данные до (слева) и после (справа) применения PCA

Mighan и Kahani (2018) реализовали PCA-SVM с точностью 0.856 и PCA-GMM 0.862 с точностью на датасете ISCX2012 [28]

9. Нечеткие системы (Fuzzy systems)

В начале 1960-х годов была применена теория нечетких множеств для решения таких проблем, как неполная информация. Теория нечетких множеств присваивает объектам значения в диапазоне от 0 до 1[29]. В нечеткой логике объект может принадлежать к разным классам одновременно, что полезно, когда разница между классами не определена явно. Благодаря этой концепции теория нечеткости может быть применена при обнаружении DDos-атак, когда различия между нормальными и аномальными классами четко не

определены [11]. Pillutla и Arjunan (2019) представили метод для обнаружения DDOS-атак на основе нечеткого SOM в SDN. Также в этом методе применяется улучшенная модель ANN (Artificial Neural Network), которая заменяет нейроны модели ANN Кохонена путем обновления нечетких правил. Для оценки авторы собрали датасет, содержащий атаки ICMP и UDP-флуд, а также атаки TCP SYN-флуд и достигли точности 94% [30].

10. Эволюционные вычисления (Evolutionary computation)

Эволюционные вычисления (EC) - это метод, основанный на естественной и биологической эволюции. Генетические алгоритмы (GA), генетическое программирование (GP), грамматическая эволюция (GE), эволюционные алгоритмы (EA), эволюционное программирование, стратегия эволюции, система обучающих классификаторов и т.д. являются примерами методов EC. Эти методы могут быть дифференцированы на основе представления их компонентов, так GP использует деревья; GA реализован в виде хромосомоподобных структур данных и использует параметры, операторы и процессы, такие как отбор, кроссинговер, мутация и функция приспособленности, для получения конкретного решения [11].

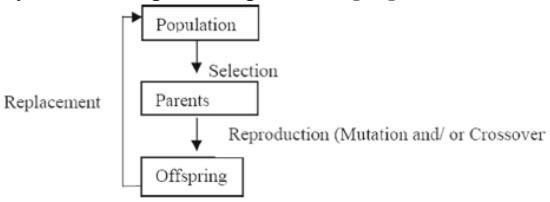

Рисунок 8 Структура эволюционного алгоритма

В исследовании Lysenko и др. (2020) генетический алгоритм используется для минимизации набора признаков, что позволяет эффективно использовать ресурсы системы для обнаружения DDoS-атак. Для обнаружения атак предлагаемый метод включает в себя генерацию правил. Признаки атак описываются набором подправил. Предлагаемый метод продемонстрировал способность обнаруживать DDoS-атаки с высокой эффективностью в диапазоне от 96,86% до 100% [31].

V. МЕТОДЫ ГЛУБОКОГО ОБУЧЕНИЯ ДЛЯ ОБНАРУЖЕНИЯ DDOS АТАК

В этой главе представлены методы глубокого обучения, используемые для обнаружения Ddos-атак.

1. Многослойный персептрон (Multi-layer Perceptron или MLP)

Многослойный персептрон - это полносвязанная сеть с входным слоем, который принимает данные, выходным слоем, который делает между этими двумя слоями суждение или вывод о входном сигнале, и одним или несколькими скрытыми слоями [17]. В MLP каждый узел одного слоя подключается с определенным весом к каждому узлу следующего слоя. Используются несколько функций активации, таких как ReLU (Rectified Linear Unit), Tanh, Sigmoid, Softmax, которые определяют выходные данные сети. Эти функции активации, также известные как передаточные функции (transfer functions), вводят нелинейные свойства в сеть для изучения сложных функциональных отображений на основе данных. MLP использует для обучения методику контролируемого обучения, называемую “Обратное распространение”. Конечной целью алгоритма обратного распространения является оптимизация весовых коэффициентов сети для точного сопоставления входных данных с целевыми выходными данными. В процессе обучения используются различные методы оптимизации, такие как стохастический градиентный спуск (SGD), BFGS с ограниченной памятью (L-BFGS), адаптивная оценка момента (Adam) [32].

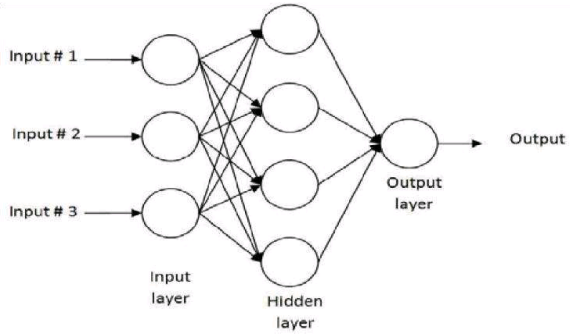

Рисунок 9 Структура многослойного персептрона

Rios и др. (2021) предложили использовать метод на основе MLP, нечеткой логики и евклидова расстояния для обнаружения DDoS атак. Данный метод был применен к набору данных CAIDA DDoS Attack 2007 и набору данных в реальном времени. Точность метода достигла 97.70% [33] Roopak и др. (2019) оценили MLP на наборе данных CICIDS2017 со скоростью обучения 0,1 и максимальным количеством эпох, равным 100. Полученная точность составила 86,34% [34] Benzaïd и др. (2020) предложили MLP с множеством скрытых слоев для обнаружения DDoS атак. Данная реализация метода достигла точности 99.65% на датасете CIC IDS 2017 [35]

2. Сверточные нейронные сети (Convolutional Neural Network или CNN)

Архитектура CNN состоит из входного и выходного слоев и множества скрытых слоев, которые включают в себя три типа слоев: сверточный, объединяющий (pooling) и полностью связанный слои. Функция ядра выполняется в сверточном слое. Веса ядра усваиваются моделью на этапе обучения таким образом, что каждое ядро представляет определенные характеристики данных. Объединяющие слои добавляются между слоями свертки для вычисления среднего значения и уменьшения размера входных данных. Полностью связанный слой выравнивает выходные данные сверточного слоя и выполняет классификацию по предоставленным признакам [36].

Примерами моделей глубокого обучения на основе CNN являются AlexNet, Exception, Inception, Visual geometry group (VGG), ResNet и т.д. [32],[38]. Архитектура CNN способна извлекать локальные признаки из данных. CNN обладает хорошей обобщающей способностью при взаимодействии с зашумленными входными данными. Методы регуляризации, такие как отсев (dropout), которые тщательно планируются, увеличивают его обобщающую способность [11]. В CNN -Сверточные слои используются для извлечения признаков, а объединяющие слои

используются для повышения обобщаемости объектов [9]. Сверточный слой использует ядра или фильтры для перемещения вдоль различных измерений (1D / 2D / 3D / 4D) данных для извлечения оптимальных признаков, которые вместе называются картами признаков (feature maps). Затем эти карты признаков передаются в объединяющий слой. Первоначально карты признаков делятся на разделы (partitions), и для уменьшения размерности карт признаков используются различные функции объединения, что является ничем иным, как операцией нелинейной понижающей дискретизации. Общими операциями объединения являются максимальная, минимальная, средняя, стохастическая, пространственная пирамида и значение деформации из раздела. Стохастическое объединение аналогично максимальному объединению, но оно также предотвращает переобучение, заменяя обычные детерминированные операции объединения стохастической процедурой, определяемой активацией в каждой области объединения в соответствии с мультиномиальным распределением. Как правило, сеть CNN может обрабатывать только входные представления фиксированной длины. Для обработки входных представлений переменной длины можно использовать объединение пространственных пирамид, поскольку оно может обрабатывать входные изображения различных масштабов, размеров и соотношений сторон [37],[38]

Архитектуры CNN хорошо известны и могут использоваться для инициализации параметров вместо случайного значения параметра в новых задачах, что представляет собой предварительное обучение. Это ускоряет процесс обучения и улучшает обобщающую способность другой модели [38].

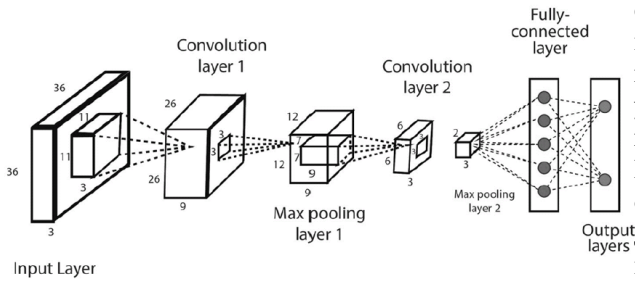

Рисунок 10 Структура сверточной нейронной сети

В статье Shaaban и др. (2019) была представлена методика сверточной нейронной сети (CNN) для обнаружения и классификации DDoS-трафика с точностью 99,33% и 99,24% для первого и второго наборов данных соответственно. Первый набор данных был получен Wireshark из смоделированной сети MCC (Mission control center) которая отвечает за управление космическим аппаратом, а другой набор данных был NSL-KDD. Другие методы, такие как DT, SVM, KNN, показали меньшую точность на этих двух наборах данных[39]. В работе Ferrag и др. (2019) сверточная нейронная сеть обеспечивает самую высокую скорость обнаружения для четырех типов атак, включая DDOS-атаку-HOIC 98,923%, DDOS-атаку-LOIC-UDP 97,888%, DDOS-атаку-LOIC HTTP 98,991% и ботнет 98,982%. В этой работе различные методы глубокого обучения, такие как DNN, RNN, RBM, DBN, DBM, CNN, были обучены на CIC IDS 2018 [40]. Xu и др. (2021) показали, что метод обнаружения, основанный на гибридной нейронной сети (CNN-GRU), может эффективно обнаруживать все шесть атак LDoS со средней точностью 97,75. Метод был оценен на датасете собранном в реальном времени из интернет-трафика [41].

3. Рекуррентные нейронные сети (Recurrent Neural Network или RNN)

Рекуррентная нейронная сеть (RNN) - это другой тип искусственной нейронной сети, которая способна обрабатывать последовательность входных данных и сохранять свое состояние при обработке следующей последовательности входных данных. Все RNN имеют петли обратной связи на рекуррентном уровне, что позволяет им сохранять информацию в "памяти" с течением времени [17]. Однако это приводит к взрыву градиента (gradient exploding) или исчезновению (gradient vanishing) во время обучения с использованием алгоритма Back Propagation Training Time. Модель LSTM была создана для предотвращения взрыва градиента [11]. Сети с длительной кратковременной памятью (LSTM) - это тип RNN, который использует специальные блоки, которые могут решать проблему исчезающего градиента, в дополнение к стандартным блокам. Каждый блок LSTM содержит три элемента управления: элемент управления памятью (forget gate), входной элемент (input gate) и выходной элемент (output gate). Forget gate определяет важность предыдущей информации и то, какую информацию следует удалить, которая больше не является полезной [36]. "Входной элемент" определяет, какая информация должна войти в состояние ячейки, а "Выходной элемент" объединяет кратковременную память с долговременной памятью для генерации текущего состояния памяти. Кроме того, устройства LSTM имеют "ячейку памяти", которая может хранить данные [32]. В действительности, стандартные RNN имеют дело только с последовательностями ограниченной длины. Для решения проблемы долгосрочной зависимости было предложено множество вариантов запуска, таких как долговременная кратковременная память (LSTM), управляемый рекуррентный блок (GRU) и bi-RNN [9],[42]. GRU был предложен Chung и др. в 2014 году. Модель GRU объединяет элемент забывания (forget gate) и элемент ввода (input gate) в единый элемент обновления (update gate) [9]. GRU - это вариант LSTM, в котором функция softmax используется в качестве конечного выходного слоя. Более того, GRU использует функцию перекрестной энтропии для расчета своих потерь [42].

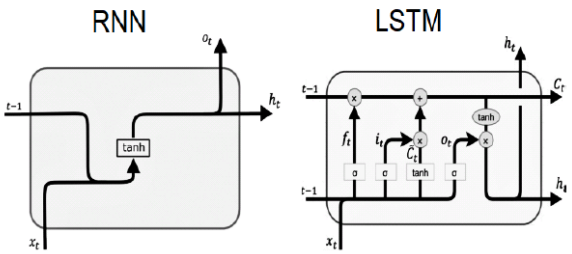

Рисунок 11 RNN и LSTM блоки

Li, и др. (2018) тренировали LSTM и GRU на ISCX2012 датасете и получили точность 89.51% и 91.56% соответственно в обнаружении DDoS атак[43] Shieh и др. (2021) применили BI-LSTM к обнаружению DDoS атак в CIC-DDoS2019 датасете. Точность обнаружения NetBIOS DDoS составила 0.898 и превзошла точность обнаружения NTP DDoS и LDAP DDoS равную 0.368 и 0.392, соответственно [44]. Liu и др.(2020) провели эксперимент на датасете CICIDS2017 с CNN-LSTM, где сериализованные предварительно обработанные данные были входными данными в нейронные сети LSTM и CNN, где LSTM предсказывает временные характеристики последовательности трафика, а CNN изучает пространственные характеристики последовательности сетевого трафика и получили точность 0.966 [45].

4. Самоорганизующаяся карта Кохонена (Self - organizing Map или SOM)

Кохонен представил модель конкурентоспособной нейронной сети, способной формировать карты признаков с помощью матричной организации искусственных нейронов [46]. Самоорганизующиеся карты (SOMs) - популярная нелинейная модель неконтролируемой нейронной сети для решения задач уменьшения размерности [27]. Самоорганизующаяся карта (SOM) использует алгоритм конкурентного обучения для обучения своей сети, в котором узлы конкурируют за право реагировать на подмножество входных данных. SOM запоминает форму набора данных, непрерывно перемещая свои нейроны ближе к точкам данных. В отличие от других искусственных нейронных сетей, использующих обучение с исправлением ошибок, таких как обратное распространение с градиентным спуском, SOM реализует функцию соседства для сохранения топологических свойств входного пространства. Алгоритм обучения SOM состоит из следующих шагов:

1. Произвольный выбор входных шаблонов и представление их в сети.
2. Вычисление расстояний карты нейронов, сгенерированной изначально, и определение ближайшего нейрона к входному вектору (минимальное расстояние).
3. Обновление весов нейрона-победителя и его соседей с топологической точки зрения.
4. Повторение предыдущих шагов до тех пор, пока не будет удовлетворен выбранный критерий остановки метода [27].

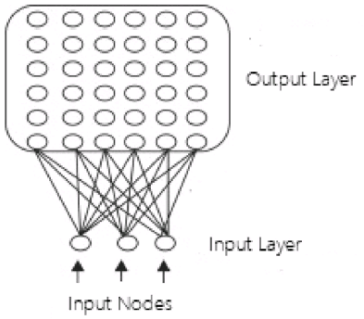

Рисунок 12 Архитектура карты Кохонена

Rafiee и Shirmarz (2022) использовали самоорганизующуюся карту (SOM) для группировки наборов данных о трафике в соответствии с их сходством. Модель была разработана и оценена с использованием набора данных CICDDoS2019. Итоговая точность составляет 98,7%. Этот метод был полезен для обнаружения DNS, LDAP, SYN DDoS-атак [47].

5. Автоэнкодер (Auto - Encoder или AE)

Автоэнкодер (AE) - это неконтролируемая нейронная сеть, которая учится минимизировать разницу между входными и выходными данными [32]. Целью автоэнкодера является изучение представления (кодировки) для набора данных. Он делает это путем выделения скрытого слоя в сети и использования его для представления данных. Когда размер этого слоя меньше входного, автоэнкодер эффективно уменьшает размерность данных [36]. Автокодер состоит из трех компонентов: кодировщика, кода и декодера. Первоначально случайные веса присваиваются как сетям кодировщика, так и сетям декодера [11]. Кодер сжимает входные данные в низкоразмерный код, извлекая таким образом ключевые характеристики из необработанных данных, а затем декодер использует этот код для восстановления входных данных [32]. Восстановленные ошибки уменьшаются с помощью процедуры обучения автоэнкодера [48]. Таким образом, во время обучения расхождение между входом кодера и выходом декодера постепенно уменьшается. Когда декодеру удается восстановить данные с помощью извлеченных признаков, это означает, что признаки, извлеченные кодером, представляют суть данных [9]. Одно из основных преимуществ AE заключается в том, что эта модель может извлекать полезные признаки и отфильтровать бесполезную информацию [32].

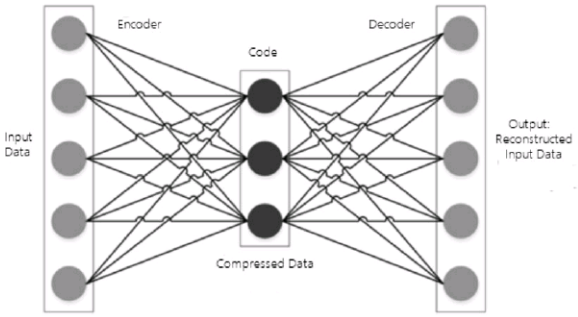

Рисунок 13 Архитектура автоэнкодера

Существуют следующие типы автоэнкодеров:

A. Многослойный автоэнкодер (Stacked Auto Encoder)

Многослойный автоэнкодер создается путем сложения входных и скрытых слоев автоэнкодера слой за слоем[49]. Mighan и Kahani (2018) разработали многослойный автоэнкодер (SAE) для уменьшения размерности и метод опорных векторов (SVM) для бинарной классификации. Эксперимент был выполнен на ISCX2012 датасете [28]

B. Автоэнкодер с шумоподавлением (Denoising Auto Encoder)

Автоэнкодеры также могут использоваться для устранения шума с входных данных, и при их использовании они известны как шумоподавляющие автоэнкодеры [36]. Шум добавляется к процедуре обучения.

C. Вариационный автоэнкодер (Variational Auto Encoder)

Вероятностная модель обучения прекрасно реализована с использованием вариационного автоэнкодера [48].

Yang и др. (2020) разработали платформу обнаружения DDoS на основе AE (AED-3F), которая обучает пятиуровневую модель AE, используя только обычный трафик из защищенной сети, а затем использует модель для обнаружения DDoS-атак. Эксперименты с различными наборами данных (синтетическими и общедоступными) показывают, что это может снизить FPR до 0 [50].

Ali и Li (2019) применили автоэнкодер для feature learning и для multiple kernel learning, чтобы объединить признаки из разных слоев. Модель была реализована поверх ISCX 2012 и UNSWNB15 датасетов[51]

6. Ограниченная машина Больцмана (Restricted Boltzmann Machine или RBM)

Машины Больцмана — это стохастические и генеративные нейронные сети только с двумя типами узлов - видимыми узлами, которые мы можем измерить и действительно измеряем, и скрытыми узлами, которые мы не можем измерить или не делаем этого. Ограниченные машины Больцмана (RBM) представляют собой особый класс машин Больцмана, и ограничены они именно с точки зрения связей между видимым слоем и скрытым слоем, т.е. между скрытым и видимым слоем переменных, но не между двумя переменными одного и того же слоя [32].

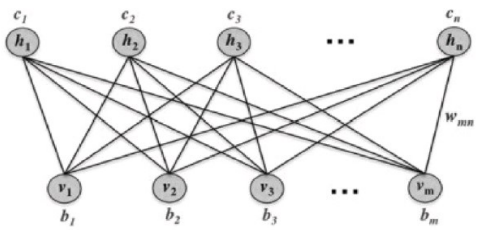

Рисунок 14 Архитектура ограниченной машины Больцмана с m видимыми и h невидимыми узлами

RBM – это также рандомизированная нейронная сеть, в которой единицы измерения подчиняются распределению Больцмана. RBM обучается с помощью алгоритма контрастной дивергенции, и обычно применяется для извлечения признаков или снижения шума [9]. Mayuranathan и др. (2019) представили исследование, в котором модель включает в себя два основных этапа, а именно выбор признаков на основе оптимизационного метода случайного гармонического поиска и классификацию на основе RBM на основе набора данных KDD Cup 99. Результат составляет 99,92% точности, тогда как RBM без выбора признаков дает 99,77% точности [52].

7. Глубокая сеть доверия (Deep Belief Network или DBN)

Основная концепция сети глубокого доверия (DBN) заключается в инициализации нейронных сетей с прямой связью неразмеченными данными с помощью неконтролируемого предварительного обучения, а затем точной настройки сети с использованием размеченных данных. DBN можно рассматривать как совокупность простых, неконтролируемых сетей, таких как ограниченные машины Больцмана (RBM) или автокодеры, где уровень каждой скрытой сети служит следующим видимым слоем [32]. Модель DBN была разработана Хинтоном и др. в 2006 году. DBN основана на модели MLP с жадным послойным обучением и может извлекать представления объектов как из размеченных, так и из неразмеченных данных. DBN состоит из множества взаимосвязанных скрытых слоев, в которых каждый слой выступает в качестве входных данных для следующего слоя и виден только следующему слою. Каждый слой в DBN не имеет боковой связи между своими узлами, присутствующими в этом слое. Сначала DBN использует преимущества эффективной послойной стратегии жадного обучения для инициализации глубокой сети, а затем точно настраивает все веса совместно с желаемыми выходными данными. DBN оптимизирует свои веса при временной сложности, линейно зависящей от глубины и размера сети. В этой модели используются неконтролируемая предварительная подготовка и контролируемые стратегии тонкой настройки [11].

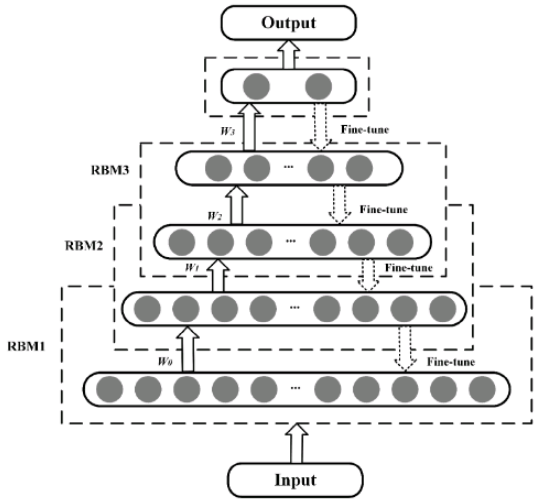

Рисунок 15 Архитектура глубокой сети доверия

DBN состоит из нескольких слоев RBM и слоя классификации softmax. Обучение DBN включает в себя два этапа: предварительную подготовку без учителя и тонкую настройку (fine tuning) с учителем. Каждая RBM обучается с использованием жадного послойного предварительного обучения. Затем по размеченным данным определяется вес слоя softmax. При обнаружении атак DBN используются как для извлечения признаков, так и для классификации [9]. В исследовании Ibrahim и др. (2021) представлен новый метод обнаружения распределенных атак типа "Отказ в обслуживании" (DDoS) в облачной среде с использованием сетей глубокого убеждения (DBN). DBN объединяется с другими классификаторами для оптимизации результатов. На входных данных CICDDoS2019 DBN+DT достиг 99,75%, а DBN+SVM - 98,5% [53].

8. Глубинная нейронная сеть (Deep Neural Network или DNN)

Глубинная нейронная сеть - это искусственная нейронная сеть (ANN), состоящая из множества входных и выходных слоев. Нейронная сеть глубока из-за множества слоев внутри нее. При этом машина неявно извлекает признаки из данных. Этот метод позволяет найти наиболее оптимальный способ преобразования входных данных в выходные. Это может быть в любой форме, будь то линейная или нелинейная зависимость. В DNN данные перемещаются с входного уровня на выходной слой без зацикливания, поэтому DNN известна как сеть с прямой связью [18],[36]. При обучении DNN параметры сначала подбираются с использованием неразмеченных данных, что является неконтролируемым этапом изучения; затем сеть настраивается с помощью размеченных данных, что является контролируемым этапом обучения [9].

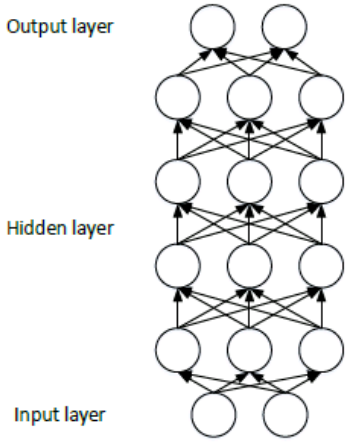

Рисунок 16 Архитектура глубинной нейронной сети

Sabeel и др. (2019) сгенерировали собственный синтетический набор данных ANTS2019 для имитации реальных атак. Была исследована производительность трех известных моделей DL, DNN и LSTM, для обнаружения неизвестных DDoS/DoS-атак. Модели сначала обучаются на предварительно обработанном наборе данных CICIDS2017 DoS/DDoS-атак, а затем их результаты оцениваются на синтезированном наборе данных ANTS2019. В этом случае точность DNN составляет 97,12%, а точность LSTM - 93,52%. Однако, если DNN и LSTM сначала обучаются на наборе данных ANTS 2019, а затем оцениваются на CICIDS2017, достигнутая точность составляет 99,59% и 99,20% соответственно[54]. Kasongo и Sun (2020) предложили глубинную нейронную сеть прямой связи (FTDNN) с алгоритмом извлечения признаков для обнаружения DDoS-атак. Метод показал точность 87,70% на датсасете UNSW NB15 [55]. Agarwal и др.(2021) улучшили точность обнаружения DDoS атак с 84.33 % до 95.35 % архитектурой DNN путем отбора признаков на основе Whale оптимизации. Эксперимент проводился на CICIDS2017 датасете [56]. Makuvaza и др. (2021) предложили глубинную нейронную сеть (DNN) для обнаружения DDoS-атак в SDN в режиме реального времени. Модель DNN достигла точности 97,59% на наборе данных CICIDS 2017 [57].

9. Генеративная состязательная сеть (Generative Adversarial Network или GAN)

GAN включает в себя две подсети, т.е. генератор и дискриминатор, которые конкурируют друг с другом [58]. Генератор предназначен для генерации синтетических данных, похожих на реальные данные, а дискриминатор предназначен для различения синтетических данных от реальных данных [9]. Благодаря непрерывной конкуренции между этими внутренними моделями, в итоге генератор способен обучиться распределенному представлению фактических данных [42]. Таким образом, генератор и дискриминатор улучшают друг друга [9]. Поскольку архитектуры DL содержат большое количество параметров и обычно применяются к большим наборам данных и поскольку получение больших наборов данных для всех классов/задач в режиме реального времени затруднено, увеличение данных с помощью GAN применяется для

увеличения выборки данных без дополнительных затрат на маркировку [38]. Shieh и др. (2022) использовали генеративные состязательные сети (GAN) Вассерштейна с градиентным штрафом для создания синтезированного трафика DDoS. Эксперименты показали, что созданный враждебный трафик может проникать через MLP, RF и KNN, не будучи обнаруженным. В качестве защитного механизма в исследовании было предложено состязательное обнаружение вторжений с двойными дискриминаторами[59].

VI. ОБЗОР ИССЛЕДОВАНИЙ В РАЗНЫХ СРЕДАХ

DDoS атаки происходят в разных средах. Кроме того, обнаружение этих атак в различных типах сетей является сложной задачей из-за их уникальных характеристик.

A. Программно-определяемые среды

Сложность традиционной сетевой архитектуры в Интернете ставит сетевого специалиста в ситуацию, которая делает невозможными настройку и управление сетью, поэтому ученые предложили новую архитектуру под названием Программно-определяемые среда (SDN). SDN включает в себя три уровня: приложение, управление и данные. Для каждого уровня существуют различные определенные задачи, и такая структура делает сеть намного более программируемой, гибкой и управляемой. SDN, в дополнение к трем уровням, имеет три API (в северном направлении, в южном направлении и с Востока на Запад). для соединения слоев и масштабирования контроллера с использованием возможностей связи контроллеров. Централизованный контроллер состоит из нескольких контроллеров, которые связаны между собой API-интерфейсами восток-запад. Плоскость данных и плоскость управления соединены с помощью API в южном направлении. Прикладной уровень основан на сетевых приложениях и связан с плоскостью управления с помощью API в северном направлении. Одним из недостатков архитектуры SDN является единственная точка отказа
контроллера; следовательно[47]. Haider и др. (2020) предложили фреймворк deep CNN для эффективного и раннего обнаружения DDoS-атак в программно-определяемых сетях. Предлагаемый метод обеспечивает наивысшую точность обнаружения DDoS атак- 99,45% на данных CICIDS2017 с более оптимальным временем обучения и классификации и включающим меньшее потребление ресурсов (% загрузки процессора). deep CNN превосходит другие предложенные DL-подходы, такие как RNN и LSTM, почти по всем метрикам[60]. Elsayed и др. (2020) предложили DDoSNet, который представляет собой метод глубокого обучения, основанный на RNN-автоэнкодере, для обнаружения DDoS-атак в программно-определяемых сетях. Они объединили RNN-автоэнкодер с регрессионной моделью softmax на выходном уровне, чтобы классифицировать сетевой трафик на вредоносный или обычный трафик. Полученные результаты показали, что модель DDoSNet с точностью 99% на наборе данных CICIDS2017 превосходит другие алгоритмы ML, такие как LR, SVM,RF, DT с точностью 95%, 93%, 86%, 77% соответственно[61].

B. Облачные среды

Для оптимального использования ресурсов облачных сред и уменьшения задержек пользователей облачных вычислений модель облачных вычислений предлагает свои сервисы, основываясь на спросе и оплате за то, что используется. Облачные вычисления стали более удобным и эффективным способом доступа к сервисам, ресурсам и приложениям через Интернет. Три основных предложения облачных вычислений - это программное обеспечение как услуга (SaaS), платформа как услуга (PaaS) и инфраструктура как услуга (IaaS). SaaS - это модель совместного использования программного обеспечения, при которой приложения размещаются продавцом или поставщиком услуг и становятся доступными для клиентов через Интернет. PaaS предлагает необходимую платформу для вычислений, а IaaS обеспечивает доступ к вычислительным ресурсам в виртуализированной инфраструктуре “облако” в Интернете. Все эти сервисы размещены в облаке, и клиенты получают к ним доступ через Интернет. Распределенные атаки типа "отказ в обслуживании". Поскольку многие организации переносят свои операции в онлайн, эти организации становятся уязвимы к DDoS-атакам, которые делают облачные сервисы недоступными для законных пользователей, создавая интенсивный трафик с нескольких устройств, называемых ботами [62]. В исследовании Virupakshar и др. (2020) были предложены различные алгоритмы машинного обучения и алгоритм глубокой нейронной сети (DNN) для обнаружения DDoS-атак в облаке на базе OpenStack. Обучающий набор данных был создан в облачной среде. LOIC ,инструмент для моделирования DDoS-атак, был использована для имитации DDoS-атак в облачной среде. Эти DDoS-атаки включают в себя ICMP-флуд, TCP-флуд и HTTP-флуд. Из результатов ясно, что глубокая нейронная сеть обладает более высокой точностью по сравнению с другими классификаторами при использовании динамически генерируемого набора данных. Таким образом, можно сказать, что DNN лучше подходит для набора данных, который динамически генерируется в облаках[63].

C. Интернет вещей

Интернет вещей (IoT) — это сеть физических объектов — устройств, транспортных средств, зданий и других предметов, - объединенных электроникой, программным обеспечением, датчиками и сетевым подключением, которое позволяет этим объектам собирать данные и обмениваться ими. В настоящее время вокруг нас существуют миллиарды различных устройств, подключенных к Интернету в различных приложениях, таких как домашняя автоматизация, социальная жизнь, системы образования, здравоохранения, развлечений и транспортные системы. Ожидается, что количество устройств превысит 75 миллиардов устройств Интернета вещей (IoT) к 2025 году. Базовая архитектура IoT состоит из уровня восприятия, сетевого уровня, уровня промежуточного программного обеспечения и

прикладного уровня [64]. Чтобы лучше обнаруживать сложные DoS- и DDoS-атаки, Hussain и др. (2020) использовали ResNet18 (модель CNN), которая состоит из 18 слоев, из которых 10 являются слоями свертки и 8 объединяющими слоями, и продемонстрировала эффективную производительность при обнаружении паттернов изображений. Для обеспечения эффективной работы ResNet авторы предложили методологию преобразования набора данных сетевого трафика CICDDoS2019, не содержащего изображений, в трехканальные изображения. ResNet предназначен для приема изображений размером 224 x 224. В исследовании предложенная методология достигла средней точности 87% в распознавании одиннадцати типов DoS-атак в сетях Интернета вещей и достигла точности 99,99% для обнаружения DoS и DDoS-атак в сетях Интернета вещей в случае двоичной классификации [65].

D. 5G

5G - это стандартная технология пятого поколения, состоящая из быстрых гетерогенных многоуровневых сетей, что неизбежно является отличным подходом в современных требовательных системах связи. Большие объемы роста трафика увеличили спрос на соединения,

что быстро приводит к сбоям в работе сети. Основными преимуществами сети 5G являются более высокие скорости при более значительном канале, широкое покрытие сети, исключительная надежность и эффективная доступность для большего числа пользователей. Amaizu и др. (2021) предложили эффективную платформу обнаружения DDoS-атак на основе DL в сетях 5G (fifth-generation) и B5G (beyond 5G). Предлагаемая структура разработана путем объединения двух по-разному спроектированных DNN. Достигнутая точность 99.66% на CICDDoS2019 датасете для этого метода обнаружения DDoS атак оказалась выше CNN и RNN [66]

E. Архитектура Fog computing

Fog computing (Туманные вычисления) - это современная вычислительная парадигма, которая предоставляет дополнительную поддержку облачной среде путем проведения некоторого локального анализа данных на периферийных устройствах, облегчая поддержку сетей, вычислений, инфраструктуры и хранилища для вычислений конечных пользователей. Несмотря на широкое использование облачных вычислений, некоторые приложения, такие как мониторинг работоспособности, игры в режиме реального времени и реагирование на чрезвычайные ситуации, чувствительны к задержкам, чтобы быть развернутыми непосредственно в облаке. Таким образом, туманные вычисления стали многообещающим дополнением к парадигме облачных вычислений, обеспечивающим лучшее время отклика, снижая затраты на транспортировку данных и место хранения. Собранные локальные данные могут агрегироваться и обрабатываться на узлах fog для обеспечения своевременной обратной связи, особенно в чрезвычайных ситуациях [67]. Priyadarshini и Barik (2019) разработали модель на основе DL для защиты от DDoS-атак в сети fog. Набор данных Hogzilla используется в этой работе для обучения и проверки предложенной модели. Этот набор данных извлекает данные из ботнета CTU-13 и наборов данных ISCX 2012 IDS. Данная модель, основанная на DL, показывает точность 98,88% на тестовых данных. [68].

VII. ЗАКЛЮЧЕНИЕ

С развитием новейших технологий, злоумышленники могут проводить DDoS-атаки с низкими затратами, и обнаружить и предотвратить DDoS-атаки становится намного сложнее, именно поэтому нужно развивать методы, способные наиболее эффективно обнаруживать DDoS-атаки. К сожалению, все еще существуют проблемы в обнаружении новых DDoS-атак, однако методы машинного и глубокого обучения дают возможность обнаружить эти атаки с высокой точностью и имеют наибольший потенциал среди имеющихся методов для обнаружения новых DDoS-атак. На сегодняшний день существует множество методов машинного и глубокого обучения для обнаруженяи DDoS-атак. В данном обзоре было представлено большинство из данных методов**.** Более того разные среды, в которых случаются DDoS атаки имеют уникальные характеристики, поэтому важно также принимать это во внимание при разработке подходов для обнаружения DDoS-атак.

БИБЛИОГРАФИЯ

[1] Zekri, M., Kafhali, S. E., Aboutabit, N., & Saadi, Y. (2017). DDoS attack detection using machine learning techniques in cloud computing environments. 2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech). doi:10.1109/cloudtech.2017.8284731

[2] Ahmad, Jalal Ale. “A Comprehensive Taxonomy of DDoS Attacks and Defense Mechanism Applying in a Smart Classification.”, WSEAS TRANSACTIONS on COMPUTERS, 2008

[3] DoS and DDoS vulnerability of IoT: A review, Emina Džaferović et al., Sustainable Engineering and Innovation, Vol. 1, No. 1., June 2019, pp.43-48 https://doi.org/10.37868/sei.v1i1.36

[4] Mahjabin, T., Xiao, Y., Sun, G., & Jiang, W. (2017). A survey of distributed denial-of-service attack, prevention, and mitigation techniques. International Journal of Distributed Sensor Networks, 13(12), 155014771774146. doi:10.1177/1550147717741463

[5] Prevention Techniques against Distributed Denial of Service Attacks in Heterogeneous Networks: A Systematic Review, Ammarah Cheema et al., 2022, Security and Communication Networks Volume 2022, https://doi.org/10.1155/2022/8379532

[6] Zhang, Boyang et al. “DDoS detection and prevention based on artificial intelligence techniques.” 2017 3rd IEEE International Conference on Computer and Communications (ICCC) (2017): 1276-1280.

[7] B. B. Gupta , R. C. Joshi & Manoj Misra (2009) Defending against Distributed Denial of Service Attacks:Issues and Challenges, Information Security Journal: A Global Perspective, 18:5,224-247,DOI:10.1080/19393550903317070

[8] Vishwakarma, R., & Jain, A. K. (2019). A survey of DDoS attacking techniques and defence mechanisms in the IoT network. Telecommunication Systems. doi:10.1007/s11235-019-00599-z

[9] Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey, Hongyu Liu and Bo Lang, Appl. Sci. 2019, 9, 4396; doi:10.3390/app9204396.

[10] Choudhary, R., & Gianey, H. K. (2017). Comprehensive Review On Supervised Machine Learning Algorithms. 2017 International Conference on Machine Learning and Data Science (MLDS). doi:10.1109/mlds.2017.11.

[11] Machine learning and deep learning methods for intrusion detection systems: recent developments and challenges, Geeta Kocher and Gulshan Kumar, Soft Computing (2021) 25:9731–9763, https://doi.org/10.1007/s00500-021-05893-0.

[12] N. Bakhareva, A. Shukhman, A. Matveev, P. Polezhaev, Y. Ushakov and L. Legashev, "Attack Detection in Enterprise Networks by Machine Learning Methods," 2019 International Russian Automation Conference (RusAutoCon), Sochi, Russia, 2019, pp. 1-6.

[13] Saini, P. S., Behal, S., & Bhatia, S. (2020). Detection of DDoS Attacks using Machine Learning Algorithms. 2020 7th International Conference on Computing for Sustainable Global Development (INDIACom). doi:10.23919/indiacom49435.2020.9083716

[14] Developing Realistic Distributed Denial of Service (DDoS) Attack Dataset and Taxonomy, Iman Sharafaldin et al.,978-1-7281-1576-4/19/$31.00 c 2019 IEEE.

[15] Priya, S. S., Sivaram, M., Yuvaraj, D., & Jayanthiladevi, A. (2020). Machine Learning based DDOS Detection. 2020 International Conference on Emerging Smart Computing and Informatics (ESCI). doi:10.1109/esci48226.2020.9167642.

[16] RoopakM, Tian GY, Chambers J (2020) An intrusion detection system against DDoS attacks in IoT networks. In: 2020 10th annual Computing and Communication Workshop and Conference, CCWC 2020. Institute of Electrical and Electronics Engineers Inc., pp 562–567

[17] A Review on Cybersecurity based on Machine Learning and Deep Learning Algorithms, Alan Fuad Jahwar and Siddeeq Y. Ameen, JOURNAL OF SOFT COMPUTING AND DATA MINING VOL.2 NO. 2 (2021) 14-25.

[18] Dhall, D., Kaur, R., Juneja, M. (2020). Machine Learning: A Review of the Algorithms and Its Applications. In: Singh, P., Kar, A., Singh, Y., Kolekar, M., Tanwar, S. (eds) Proceedings of ICRIC 2019 . Lecture Notes in Electrical Engineering, vol 597. Springer, https://doi.org/10.1007/978-3-030-29407-6_5.

[19] Machine learning methods: An overview, Ravil I Muhamedyev, COMPUTER MODELLING & NEW TECHNOLOGIES 2015 19(6) 14-29.

[20] Kasim Ö (2020) An efficient and robust deep learning based network anomaly detection against distributed denial of service attacks. Comput Netw 180:107390

[21] Ramadhan, I., Sukarno, P., & Nugroho, M. A. (2020). Comparative Analysis of K-Nearest Neighbor and Decision Tree in Detecting Distributed Denial of Service. 2020 8th International Conference on Information and Communication Technology (ICoICT). doi:10.1109/icoict49345.2020.9166380

[22] Dash, S.S., Nayak, S.K., Mishra, D. (2021). A Review on Machine Learning Algorithms. In: Mishra, D., Buyya, R., Mohapatra, P., Patnaik, S. (eds) Intelligent and Cloud Computing. Smart Innovation, Systems and Technologies, vol 153. Springer, Singapore. https://doi.org/10.1007/978-981-15-6202-0_51.

[23] Sofi, Irfan Ahmad et al., Machine Learning Techniques used for the Detection and Analysis of Modern Types of DDoS Attacks, International Research Journal of Engineering and Technology (IRJET),2017.

[24] Dasgupta, D., Akhtar, Z., & Sen, S. (2020). Machine learning in cybersecurity: a comprehensive survey. The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology, 154851292095127. doi:10.1177/1548512920951275.

[25] Tuan, T. et al., (2019). Performance evaluation of Botnet DDoS attack detection using machine learning. Evolutionary Intelligence. doi:10.1007/s12065-019-00310-w

[26] Min, E., Long, J., Liu, Q., Cui, J., Chen, W. TR-IDS: Anomaly-based Intrusion Detection Through Text- Convolutional Neural Network and Random Forest. Security and Communication Networks, 2018

[27] Martínez Torres, J., Iglesias Comesaña, C., & García-Nieto, P. J. (2019). Review: machine learning techniques applied to cybersecurity. International Journal of Machine Learning and Cybernetics. doi:10.1007/s13042-018-00906-1.

[28] Mighan, S. N., Kahani, M. Deep Learning based Latent Feature Extraction for Intrusion Detection. Proceed¬ings of Iranian Conference on Electrical Engineering (ICEE), Mashhad, Iran, May 08-10, 2018, 1511-1516. https://doi.org/10.1109/ICEE.2018.8472418.

[29] Tsoukalas L.H., Uhrig R.E. (1997) Fuzzy and neural approaches in engineering. 18216097198.

[30] Pillutla H, Arjunan A. Fuzzy self organizing maps-based DDoS mitigation mechanism for software defined networking in cloud computing. J Ambient Intell Humaniz Comput 2019;10:1547–59

[31] Lysenko, S., Bobrovnikova, K., Shchuka, R., & Savenko, O. (2020). A Cyberattacks Detection Technique Based on Evolutionary Algorithms. 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT). doi:10.1109/dessert50317.2020.9125016.

[32] Sarker, I. H. (2021). Deep Cybersecurity: A Comprehensive Overview from Neural Network and Deep Learning Perspective. SN Computer Science, 2(3). doi:10.1007/s42979-021-00535-6.

[33] Rios, V. de M., Inácio, P. R. M., Magoni, D., & Freire, M. M. (2021). Detection of reduction-of-quality DDoS attacks using Fuzzy Logic and machine learning algorithms. Computer Networks, 186, 107792. doi:10.1016/j.comnet.2020.107792.

[34] Roopak, Monika et al. “Deep Learning Models for Cyber Security in IoT Networks.” 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC) (2019): 0452-0457.

[35] C. Benzaïd, M. Boukhalfa and T. Taleb, "Robust Self-Protection Against Application-Layer (D)DoS Attacks in SDN Environment," 2020 IEEE Wireless Communications and Networking Conference (WCNC), Seoul, Korea (South), 2020, pp. 1-6, doi: 10.1109/WCNC45663.2020.9120472.

[36] Deep Learning Applications In Cyber Security: A Comprehensive Review, Challenges And Prospects, Bhavuk Sharma and Ramchandra Mangrulkar, International Journal of Engineering Applied Sciences and Technology, 2019, Vol. 4, Issue 8, ISSN No. 2455-2143, Pages 148-159.

[37] Berman, D., Buczak, A., Chavis, J., & Corbett, C. (2019). A Survey of Deep Learning Methods for Cyber Security. Information, 10(4), 122. doi:10.3390/info10040122.

[38] Deep Learning for Cyber Security Applications: A Comprehensive Survey, Vinayakumar Ravi et al.,2021.

[39] A. R. Shaaban, E. Abd-Elwanis and M. Hussein, "DDoS attack detection and classification via Convolutional Neural Network (CNN)," 2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS), Cairo, Egypt, 2019, pp. 233-238, doi: 10.1109/ICICIS46948.2019.9014826.

[40] Ferrag, M. A., Maglaras, L., Janicke, H., Smith, R. Deep Learning Techniques for Cyber Security Intrusion Detec¬tion: A Detailed Analysis. Proceedings of 6th Internation¬al Symposium for ICS & SCADA Cyber Security Research 2019 (ICS-CSR), Athens, Greece, September 10-12, 2019, 126-136. https://doi.org/10.14236/ewic/icscsr19.16.

[41] Xu, C., Shen, J., Du, X. Low-rate DoS Attack Detection Method Based on Hybrid Deep Neural Networks. Jour-nal of Information Security and Applications, 2021, 60, 102879. https://doi.org/10.1016/j.jisa.2021.102879.

[42] Wu, Y., Wei, D., & Feng, J. (2020). Network Attacks Detection Methods Based on Deep Learning Techniques: A Survey. Security and Communication Networks, 2020, 1–17. doi:10.1155/2020/8872923.

[43] Li, C., Wu, Y., Yuan, X., Sun, Z., Wang, W., Li, X., Gong, L. Detection and Defense of DDoS Attack-Based on Deep Learning in Openflow-Based SDN. International Jour¬nal of Communication Systems, 2018, 31(5), e3497 https://doi.org/10.1002/dac.3497.

[44] Shieh, C. S., Lin, W. W., Nguyen, T. T., Chen, C. H., Horng, M. F., Miu, D. Detection of Unknown DDoS At-tacks with Deep Learning and Gaussian Mixture Mod¬el. Applied Sciences, 2021, 11(11), 5213. https://doi. org/10.3390/app11115213.

[45] Liu, L., Lin, J., Wang, P., Liu, L., Zhou, R. Deep Learn-ing-Based Network Security Data Sampling and Anomaly Prediction in Future Network. Discrete Dynamics in Nature and Society, 2020. https://doi. org/10.1155/2020/4163825.

[46] Kohonen, T. Self-organized formation of topologically correct feature maps. Biol. Cybern. 43, 59–69 (1982). https://doi.org/10.1007/BF00337288.

[47] Self-Organization Map (SOM) Algorithm for DDoS Attack Detection in Distributed Software Defined Network (D-SDN), Mohsen Rafiee and Alireza shirmarz, Journal of Information Systems and Telecommunication Vol.10, No.2, April-June 2022, 120-131

[48] Dixit, P., & Silakari, S. (2021). Deep Learning Algorithms for Cybersecurity Applications: A Technological and Status Review. Computer Science Review, 39, 100317. doi:10.1016/j.cosrev.2020.100317.

[49] Yadav, S., Subramanian, S. Detection of Application Layer DDoS Attack by Feature Learning using Stacked AutoEncoder. Proceedings of International Confer¬ence on Computational Techniques in Information and Communication Technologies (ICCTICT), New Delhi, March 11-13, 2016, 361-366. https://doi.org/10.1109/ ICCTICT.2016.7514608.

[50] K. Yang, J. Zhang, Y. Xu and J. Chao, "DDoS Attacks Detection with AutoEncoder," NOMS 2020 - 2020 IEEE/IFIP Network Operations and Management Symposium, Budapest, Hungary, 2020, pp. 1-9, doi: 10.1109/NOMS47738.2020.9110372.

[51] Ali, S., and Li, Y. Learning Multilevel Auto-Encoders for DDoS Attack Detection in Smart Grid Network. IEEE Access, 2019, 7, 108647-108659. https://doi. org/10.1109/ACCESS.2019.2933304.

[52] Mayuranathan, M., Murugan, M., & Dhanakoti, V. (2019). Best features based intrusion detection system by RBM model for detecting DDoS in cloud environment. Journal of Ambient Intelligence and Humanized Computing. doi:10.1007/s12652-019-01611-9.

[53] Ibrahim Yousif IBRAHIM and Sefer KURNAZ, A NEW DISTRIBUTED DENIAL-OF-SERVICE DETECTION SYSTEM IN CLOUD ENVIRONMENT BY USING DEEP BELIEF

NETWORKS, Commun.Fac.Sci.Univ.Ank.Series A2-A3, Volume 63, Number 1, Pages 17-24 (2021), DOI: 10.33769/aupse.697067.

[54] Sabeel, U., Heydari, S. S., Mohanka, H., Bendhaou, Y., El¬gazzar, K., El-Khatib, K. Evaluation of Deep Learning in Detecting Unknown Network Attacks. Proceedings of International Conference on Smart Applications, Communications and Networking (SmartNets), South Sinai Governorate, Egypt, December 17-18, 2019, 1-6. https:// doi.org/10.1109/SmartNets48225.2019.9069788.

[55] Kasongo, S. M., Sun, Y. A Deep Learning Method with Wrapper based Feature Extraction for Wireless Intru-sion Detection System. Computers and Security, 2020, 92, 101752. https://doi.org/10.1016/j.cose.2020.101752.

[56] Agarwal, A., Khari, M., Singh, R. Detection of DDOS At-tack using Deep Learning Model in Cloud Storage Ap-plication. Wireless Personal Communications, 2021, 1-21. https://doi.org/10.1007/s11277-021-08271-z

[57] Makuvaza, A., Jat, D.S. & Gamundani, A.M. Deep Neural Network (DNN) Solution for Real-time Detection of Distributed Denial of Service (DDoS) Attacks in Software Defined Networks (SDNs). SN COMPUT. SCI. 2, 107 (2021). https://doi.org/10.1007/s42979-021-00467-1.

[58] Artificial Neural Network for Cybersecurity: A Comprehensive Review, Institute of Information and Communication Technology, Bangladesh University of Engineering and Technology, Dhaka-1205, Prajoy Podder et al., 2021.

[59] Shieh, C. et al.,Detection of Adversarial DDoS Attacks Using Generative Adversarial Networks with Dual Discriminators. Symmetry 2022, 14, 66. https://doi.org/10.3390/sym14010066

[60] S. Haider et al., "A Deep CNN Ensemble Framework for Efficient DDoS Attack Detection in Software Defined Networks," in IEEE Access, vol. 8, pp. 53972-53983, 2020, doi: 10.1109/ACCESS.2020.2976908.

[61] Elsayed M.S., Le-Khac N.A., Dev S, Jurcut A.D. (2020) DDoSNet: a deeplearning model for detecting network attacks. In: Proceedings— 21st IEEE international symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM 2020. Institute of Electrical and Electronics Engineers Inc., pp 391–396

[62] Srinivasan, K., Mubarakali, A., Alqahtani, A.S., Dinesh Kumar, A. (2020). A Survey on the Impact of DDoS Attacks in Cloud Computing: Prevention, Detection and Mitigation Techniques. In: Balaji, S., Rocha, Á., Chung, YN. (eds) Intelligent Communication Technologies and Virtual Mobile Networks. ICICV 2019. Lecture Notes on Data Engineering and Communications Technologies, vol 33. Springer, Cham. https://doi.org/10.1007/978-3-030-28364-3_24

[63] Virupakshar KB, Asundi M, Channal K, Shettar P, Patil S, Narayan DG (2020) Distributed Denial of Service (DDoS) attacks detection system for OpenStack-based Private Cloud. Procedia Comput Sci 167:2297–2307.

[64] Rozan Khader and Derar Eleyan, Survey of DoS/DDoS attacks in IoT, Sustainable Engineering and Innovation ISSN 2712-0562 Vol. 3, No. 1, January 2021, pp.23-28, https://doi.org/10.37868/sei.v3i1.124

[65] F. Hussain, S. G. Abbas, M. Husnain, U. U. Fayyaz, F. Shahzad and G. A. Shah, "IoT DoS and DDoS Attack Detection using ResNet," 2020 IEEE 23rd International Multitopic Conference (INMIC), Bahawalpur, Pakistan, 2020, pp. 1-6, doi: 10.1109/INMIC50486.2020.9318216.

[66] Amaizu G.C., Nwakanma C.I., Bhardwaj S, Lee J.M., KimDS (2021) Composite and efficient DDoS attack detection framework for B5G networks. Comput Netw 188:107871.

[67] Rani, S., Saini, P. (2020). Fog Computing: Applications and Secure Data Aggregation. In: Gupta, B., Perez, G., Agrawal, D., Gupta, D. (eds) Handbook of Computer Networks and Cyber Security. Springer, Cham. https://doi.org/10.1007/978-3-030-22277-2_19

[68] Priyadarshini R, Barik R.K. (2019) A deep learning based intelligent framework to mitigate DDoS attack in fog environment. J King Saud Univ Comput Inf Sci

machine and deep learning used to detect DDoS attacks. In addition to describing the methods themselves, examples of studies where these methods were used to detect DDoS attacks are also given. At the end of the article, examples of environments vulnerable to DDoS attacks are given. This article will help you get acquainted with modern effective methods of detecting DDoS attacks.

Keywords — DDoS attack, Machine learning, Deep learning, UDP flood, ICMP flood, HTTP flood, OSI, Agent-Handler, Reflector, IRC.

REFERENCES

1. Zekri, M., Kafhali, S. E., Aboutabit, N., & Saadi, Y. (2017). DDoS attack detection using machine learning techniques in cloud computing environments. 2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech). doi:10.1109/cloudtech.2017.8284731
2. Ahmad, Jalal Ale. "A Comprehensive Taxonomy of DDoS Attacks and Defense Mechanism Applying in a Smart Classification.", WSEAS TRANSACTIONS on COMPUTERS, 2008
3. DoS and DDoS vulnerability of IoT: A review, Emina Džaferović et al., Sustainable Engineering and Innovation, Vol. 1, No. 1., June 2019, pp.43-48 https://doi.org/10.37868/sei.v1i1.36
4. Mahjabin, T., Xiao, Y., Sun, G., & Jiang, W. (2017). A survey of distributed denial-of-service attack, prevention, and mitigation techniques. International Journal of Distributed Sensor Networks, 13(12), 155014771774146. doi:10.1177/1550147717741463
5. Prevention Techniques against Distributed Denial of Service Attacks in Heterogeneous Networks: A Systematic Review, Ammarah Cheema et al., 2022, Security and Communication Networks Volume 2022, https://doi.org/10.1155/2022/8379532
6. Zhang, Boyang et al. "DDoS detection and prevention based on artificial intelligence techniques." 2017 3rd IEEE International Conference on Computer and Communications (ICCC) (2017): 1276-1280.
7. B. B. Gupta , R. C. Joshi & Manoj Misra (2009) Defending against Distributed Denial of Service Attacks:Issues and Challenges, Information Security Journal: A Global Perspective, 18:5,224-247,DOI:10.1080/19393550903317070
8. Vishwakarma, R., & Jain, A. K. (2019). A survey of DDoS attacking techniques and defence mechanisms in the IoT network. Telecommunication Systems. doi:10.1007/s11235-019-00599-z
9. Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey, Hongyu Liu and Bo Lang, Appl. Sci. 2019, 9, 4396; doi:10.3390/app9204396.
10. Choudhary, R., & Gianey, H. K. (2017). Comprehensive Review On Supervised Machine Learning Algorithms. 2017 International Conference on Machine Learning and Data Science (MLDS). doi:10.1109/mlds.2017.11.
11. Machine learning and deep learning methods for intrusion detection systems: recent developments and challenges, Geeta Kocher and Gulshan Kumar, Soft Computing (2021) 25:9731–9763, https://doi.org/10.1007/s00500-021-05893-0.
12. N. Bakhareva, A. Shukhman, A. Matveev, P. Polezhaev, Y. Ushakov and L. Legashev, "Attack Detection in Enterprise Networks by Machine Learning Methods," 2019 International Russian Automation Conference (RusAutoCon), Sochi, Russia, 2019, pp. 1-6.

Overview of methods for detecting distributed denial-of-service attacks based on machine learning and deep learning

T.M.Klimenko, R.R.Akzhigitov

Abstract— Distributed denial of service (DDoS) attacks pose a serious threat to network security. In a Denial of Service (DOS) attack, a single source performs the attack, while DDoS uses multiple hosts to attack the system. It is very difficult to identify the source of the attack when such an attack occurs, since the attacker hides his identity by spoofing his IP address. How to detect DDoS attacks and defend against them is currently an urgent topic both in industry and in scientific circles. This article discusses the mechanism of DDoS attacks and DDoS attack models, the main methods of launching DDoS attacks, types of attacks according to the OSI model and a more detailed description of the types of DDoS attacks aimed at a specific vulnerability. This article systematizes the methods of

13. Saini, P. S., Behal, S., & Bhatia, S. (2020). Detection of DDoS Attacks using Machine Learning Algorithms. 2020 7th International Conference on Computing for Sustainable Global Development (INDIACom). doi:10.23919/indiacom49435.2020.9083716
14. Developing Realistic Distributed Denial of Service (DDoS) Attack Dataset and Taxonomy, Iman Sharafaldin et al.,978-1-7281-1576-4/19/$31.00 c 2019 IEEE.
15. Priya, S. S., Sivaram, M., Yuvaraj, D., & Jayanthiladevi, A. (2020). Machine Learning based DDOS Detection. 2020 International Conference on Emerging Smart Computing and Informatics (ESCI). doi:10.1109/esci48226.2020.9167642.
16. RoopakM, Tian GY, Chambers J (2020) An intrusion detection system against DDoS attacks in IoT networks. In: 2020 10th annual Computing and Communication Workshop and Conference, CCWC 2020. Institute of Electrical and Electronics Engineers Inc., pp 562–567
17. A Review on Cybersecurity based on Machine Learning and Deep Learning Algorithms, Alan Fuad Jahwar and Siddeeq Y. Ameen, JOURNAL OF SOFT COMPUTING AND DATA MINING VOL.2 NO. 2 (2021) 14-25.
18. Dhall, D., Kaur, R., Juneja, M. (2020). Machine Learning: A Review of the Algorithms and Its Applications. In: Singh, P., Kar, A., Singh, Y., Kolekar, M., Tanwar, S. (eds) Proceedings of ICRIC 2019 . Lecture Notes in Electrical Engineering, vol 597. Springer, https://doi.org/10.1007/978-3-030-29407-6_5.
19. Machine learning methods: An overview, Ravil I Muhamedyev, COMPUTER MODELLING & NEW TECHNOLOGIES 2015 19(6) 14-29.
20. Kasim Ö (2020) An efficient and robust deep learning based network anomaly detection against distributed denial of service attacks. Comput Netw 180:107390
21. Ramadhan, I., Sukarno, P., & Nugroho, M. A. (2020). Comparative Analysis of K-Nearest Neighbor and Decision Tree in Detecting Distributed Denial of Service. 2020 8th International Conference on Information and Communication Technology (ICoICT). doi:10.1109/icoict49345.2020.9166380
22. Dash, S.S., Nayak, S.K., Mishra, D. (2021). A Review on Machine Learning Algorithms. In: Mishra, D., Buyya, R., Mohapatra, P., Patnaik, S. (eds) Intelligent and Cloud Computing. Smart Innovation, Systems and Technologies, vol 153. Springer, Singapore. https://doi.org/10.1007/978-981-15-6202-0_51.
23. Sofi, Irfan Ahmad et al., Machine Learning Techniques used for the Detection and Analysis of Modern Types of DDoS Attacks, International Research Journal of Engineering and Technology (IRJET),2017.
24. Dasgupta, D., Akhtar, Z., & Sen, S. (2020). Machine learning in cybersecurity: a comprehensive survey. The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology, 154851292095127. doi:10.1177/1548512920951275.
25. Tuan, T. et al., (2019). Performance evaluation of Botnet DDoS attack detection using machine learning. Evolutionary Intelligence. doi:10.1007/s12065-019-00310-w
26. Min, E., Long, J., Liu, Q., Cui, J., Chen, W. TR-IDS: Anomaly-based Intrusion Detection Through Text- Convolutional Neural Network and Random Forest. Security and Communication Networks, 2018
27. Martínez Torres, J., Iglesias Comesaña, C., & García-Nieto, P. J. (2019). Review: machine learning techniques applied to cybersecurity. International Journal of Machine Learning and Cybernetics. doi:10.1007/s13042-018-00906-1.
28. Mighan, S. N., Kahani, M. Deep Learning based Latent Feature Extraction for Intrusion Detection. Proceed¬ings of Iranian Conference on Electrical Engineering (ICEE), Mashhad, Iran, May 08-10, 2018, 1511-1516. https://doi.org/10.1109/ICEE.2018.8472418.
29. Tsoukalas L.H., Uhrig R.E. (1997) Fuzzy and neural approaches in engineering. 18216097198.
30. Pillutla H, Arjunan A. Fuzzy self organizing maps-based DDoS mitigation mechanism for software defined networking in cloud computing. J Ambient Intell Humaniz Comput 2019;10:1547–59
31. Lysenko, S., Bobrovnikova, K., Shchuka, R., & Savenko, O. (2020). A Cyberattacks Detection Technique Based on Evolutionary Algorithms. 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT). doi:10.1109/dessert50317.2020.9125016.
32. Sarker, I. H. (2021). Deep Cybersecurity: A Comprehensive Overview from Neural Network and Deep Learning Perspective. SN Computer Science, 2(3). doi:10.1007/s42979-021-00535-6.
33. Rios, V. de M., Inácio, P. R. M., Magoni, D., & Freire, M. M. (2021). Detection of reduction-of-quality DDoS attacks using Fuzzy Logic and machine learning algorithms. Computer Networks, 186, 107792. doi:10.1016/j.comnet.2020.107792.
34. Roopak, Monika et al. “Deep Learning Models for Cyber Security in IoT Networks.” 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC) (2019): 0452-0457.
35. C. Benzaïd, M. Boukhalfa and T. Taleb, "Robust Self-Protection Against Application-Layer (D)DoS Attacks in SDN Environment," 2020 IEEE Wireless Communications and Networking Conference (WCNC), Seoul, Korea (South), 2020, pp. 1-6, doi: 10.1109/WCNC45663.2020.9120472.
36. Deep Learning Applications In Cyber Security: A Comprehensive Review, Challenges And Prospects, Bhavuk Sharma and Ramchandra Mangrulkar, International Journal of Engineering Applied Sciences and Technology, 2019, Vol. 4, Issue 8, ISSN No. 2455-2143, Pages 148-159.
37. Berman, D., Buczak, A., Chavis, J., & Corbett, C. (2019). A Survey of Deep Learning Methods for Cyber Security. Information, 10(4), 122. doi:10.3390/info10040122.
38. Deep Learning for Cyber Security Applications: A Comprehensive Survey, Vinayakumar Ravi et al.,2021.
39. A. R. Shaaban, E. Abd-Elwanis and M. Hussein, "DDoS attack detection and classification via Convolutional Neural Network (CNN)," 2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS), Cairo, Egypt, 2019, pp. 233-238, doi: 10.1109/ICICIS46948.2019.9014826.
40. Ferrag, M. A., Maglaras, L., Janicke, H., Smith, R. Deep Learning Techniques for Cyber Security Intrusion Detec¬tion: A Detailed Analysis. Proceedings of 6th Internation¬al Symposium for ICS & SCADA Cyber Security Research 2019 (ICS-CSR), Athens, Greece, September 10-12, 2019, 126-136. https://doi.org/10.14236/ewic/icscsr19.16.
41. Xu, C., Shen, J., Du, X. Low-rate DoS Attack Detection Method Based on Hybrid Deep Neural Networks. Jour-nal of Information Security and Applications, 2021, 60, 102879. https://doi.org/10.1016/j.jisa.2021.102879.
42. Wu, Y., Wei, D., & Feng, J. (2020). Network Attacks Detection Methods Based on Deep Learning Techniques: A Survey. Security and Communication Networks, 2020, 1–17. doi:10.1155/2020/8872923.
43. Li, C., Wu, Y., Yuan, X., Sun, Z., Wang, W., Li, X., Gong, L. Detection and Defense of DDoS Attack-Based on Deep Learning in Openflow-Based SDN. International Jour¬nal of Communication Systems, 2018, 31(5), e3497 https://doi.org/10.1002/dac.3497.
44. Shieh, C. S., Lin, W. W., Nguyen, T. T., Chen, C. H., Horng, M. F., Miu, D. Detection of Unknown DDoS At-tacks with Deep Learning and Gaussian Mixture Mod¬el. Applied Sciences, 2021, 11(11), 5213. https://doi. org/10.3390/app11115213.
45. Liu, L., Lin, J., Wang, P., Liu, L., Zhou, R. Deep Learn-ing-Based Network Security Data Sampling and Anomaly Prediction in Future Network. Discrete Dynamics in Nature and Society, 2020. https://doi. org/10.1155/2020/4163825.
46. Kohonen, T. Self-organized formation of topologically correct feature maps. Biol. Cybern. 43, 59–69 (1982). https://doi.org/10.1007/BF00337288.
47. Self-Organization Map (SOM) Algorithm for DDoS Attack Detection in Distributed Software Defined Network (D-SDN), Mohsen Rafiee and Alireza shirmarz, Journal of Information Systems and Telecommunication Vol.10, No.2, April-June 2022, 120-131
48. Dixit, P., & Silakari, S. (2021). Deep Learning Algorithms for Cybersecurity Applications: A Technological and Status Review. Computer Science Review, 39, 100317. doi:10.1016/j.cosrev.2020.100317.
49. Yadav, S., Subramanian, S. Detection of Application Layer DDoS Attack by Feature Learning using Stacked AutoEncoder. Proceedings of International Confer¬ence on Computational Techniques in Information and Communication Technologies (ICCTICT), New Delhi, March 11-13, 2016, 361-366. https://doi.org/10.1109/ ICCTICT.2016.7514608.
50. K. Yang, J. Zhang, Y. Xu and J. Chao, "DDoS Attacks Detection with AutoEncoder," NOMS 2020 - 2020 IEEE/IFIP Network Operations and Management Symposium, Budapest, Hungary, 2020, pp. 1-9, doi: 10.1109/NOMS47738.2020.9110372.
51. Ali, S., and Li, Y. Learning Multilevel Auto-Encoders for DDoS Attack Detection in Smart Grid Network. IEEE Access, 2019, 7, 108647-108659. https://doi. org/10.1109/ACCESS.2019.2933304.
52. Mayuranathan, M., Murugan, M., & Dhanakoti, V. (2019). Best features based intrusion detection system by RBM model for detecting DDoS in cloud environment. Journal of Ambient Intelligence and Humanized Computing. doi:10.1007/s12652-019-01611-9.
53. Ibrahim Yousif IBRAHIM and Sefer KURNAZ, A NEW DISTRIBUTED DENIAL-OF-SERVICE DETECTION SYSTEM IN CLOUD ENVIRONMENT BY USING DEEP BELIEF NETWORKS, Commun.Fac.Sci.Univ.Ank.Series A2-A3, Volume 63, Number 1, Pages 17-24 (2021), DOI: 10.33769/aupse.697067.
54. Sabeel, U., Heydari, S. S., Mohanka, H., Bendhaou, Y., El¬gazzar, K., El-Khatib, K. Evaluation of Deep Learning in Detecting Unknown Network Attacks. Proceedings of International Conference on Smart Applications, Communications and Networking (SmartNets), South

Sinai Governorate, Egypt, December 17-18, 2019, 1-6. https:// doi.org/10.1109/SmartNets48225.2019.9069788.
55. Kasongo, S. M., Sun, Y. A Deep Learning Method with Wrapper based Feature Extraction for Wireless Intru-sion Detection System. Computers and Security, 2020, 92, 101752. https://doi.org/10.1016/j.cose.2020.101752.
56. Agarwal, A., Khari, M., Singh, R. Detection of DDOS At-tack using Deep Learning Model in Cloud Storage Ap-plication. Wireless Personal Communications, 2021, 1-21. https://doi.org/10.1007/s11277-021-08271-z
57. Makuvaza, A., Jat, D.S. & Gamundani, A.M. Deep Neural Network (DNN) Solution for Real-time Detection of Distributed Denial of Service (DDoS) Attacks in Software Defined Networks (SDNs). SN COMPUT. SCI. 2, 107 (2021). https://doi.org/10.1007/s42979-021-00467-1.
58. Artificial Neural Network for Cybersecurity: A Comprehensive Review, Institute of Information and Communication Technology, Bangladesh University of Engineering and Technology, Dhaka-1205, Prajoy Podder et al., 2021.
59. Shieh, C. et al.,Detection of Adversarial DDoS Attacks Using Generative Adversarial Networks with Dual Discriminators. Symmetry 2022, 14, 66. https://doi.org/10.3390/sym14010066
60. S. Haider et al., "A Deep CNN Ensemble Framework for Efficient DDoS Attack Detection in Software Defined Networks," in IEEE Access, vol. 8, pp. 53972-53983, 2020, doi: 10.1109/ACCESS.2020.2976908.
61. Elsayed M.S., Le-Khac N.A., Dev S, Jurcut A.D. (2020) DDoSNet: a deeplearning model for detecting network attacks. In: Proceedings— 21st IEEE international symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM 2020. Institute of Electrical and Electronics Engineers Inc., pp 391–396
62. Srinivasan, K., Mubarakali, A., Alqahtani, A.S., Dinesh Kumar, A. (2020). A Survey on the Impact of DDoS Attacks in Cloud Computing: Prevention, Detection and Mitigation Techniques. In: Balaji, S., Rocha, Á., Chung, YN. (eds) Intelligent Communication Technologies and Virtual Mobile Networks. ICICV 2019. Lecture Notes on Data Engineering and Communications Technologies, vol 33. Springer, Cham. https://doi.org/10.1007/978-3-030-28364-3_24
63. Virupakshar KB, Asundi M, Channal K, Shettar P, Patil S, Narayan DG (2020) Distributed Denial of Service (DDoS) attacks detection system for OpenStack-based Private Cloud. Procedia Comput Sci 167:2297–2307.
64. Rozan Khader and Derar Eleyan, Survey of DoS/DDoS attacks in IoT, Sustainable Engineering and Innovation ISSN 2712-0562 Vol. 3, No. 1, January 2021, pp.23-28, https://doi.org/10.37868/sei.v3i1.124
65. F. Hussain, S. G. Abbas, M. Husnain, U. U. Fayyaz, F. Shahzad and G. A. Shah, "IoT DoS and DDoS Attack Detection using ResNet," 2020 IEEE 23rd International Multitopic Conference (INMIC), Bahawalpur, Pakistan, 2020, pp. 1-6, doi: 10.1109/INMIC50486.2020.9318216.
66. Amaizu G.C., Nwakanma C.I., Bhardwaj S, Lee J.M., KimDS (2021) Composite and efficient DDoS attack detection framework for B5G networks. Comput Netw 188:107871.
67. Rani, S., Saini, P. (2020). Fog Computing: Applications and Secure Data Aggregation. In: Gupta, B., Perez, G., Agrawal, D., Gupta, D. (eds) Handbook of Computer Networks and Cyber Security. Springer, Cham. https://doi.org/10.1007/978-3-030-22277-2_19
68. Priyadarshini R, Barik R.K. (2019) A deep learning based intelligent framework to mitigate DDoS attack in fog environment. J King Saud Univ Comput Inf Sci