Анализ безопасности при атаке MITM на сети АСУ ТП (метод Packet Inspection)

Кибербезопасность - одна из основных задач в автоматизированных системах управления технологическими процессами (АСУ ТП, SCADA, Supervisory Control and Data Acquisition Technique). В АСУ ТП злоумышленники могут либо отключить систему, либо попытаться повредить сеть, отправляя неправильные данные или команды для нарушения связи. Системы АСУ ТП наиболее уязвимы для злоумышленников из-за их взаимосвязи в интеллектуальных сетях и использования стандартных текстовых протоколов. В системах АСУ ТП из-за унаследованной инфраструктуры связи и протоколов они имеют уязвимости кибербезопасности, поскольку изначально системы проектировались с меньшим учетом киберугроз. В данном материале анализируется возможность проведения атак типа «человек посередине» (MITM) на системы связи АСУ ТП, которая использует стандарт протокола связи SCADA Международной электротехнической комиссии (IEC 60870-5-104). Метод Packet Inspection в системах АСУ ТП используется для обнаружения MITM-атак, проводимых на основе метода ARP Poisoning. Сети связи АСУ ТП используются для управления различными инфраструктурами и играют жизненно важную роль для коммунальных предприятий и обрабатывающих производств, включая энергетический сектор, газ, нефть, водоснабжение и т. д.

Диспетчерский контроль и сбор данных, управление оборудованием, широко известные как АСУ ТП (SCADA), можно рассматривать как распределённую компьютерную систему, используемую для сбора данных, анализа и управления в режиме реального времени. В данных системах, как правило, есть удаленный терминал (RTU, Remote Terminal Unit) и главный терминал (MTU, Master Terminal Unit), человеко-машинный интерфейс, архиватор, сервер аналитики, сервер отчетов и т. д. Связь между RTU и MTU может быть прослушана посредством атаки типа MITM. Сети связи АСУ ТП, имеющие сетевую инфраструктуру, и применяемые протоколы ранее не

считали проблемными с точки зрения угроз кибербезопасности. АСУ ТП могут рассматриваться злоумышленниками как цель для получения несанкционированного доступа к системе в уязвимых точках. Таким образом, необходимо обеспечить необходимый уровень защиты сети связи АСУ ТП от несанкционированного доступа [13].

Сеть АСУ ТП состоит из MTU для взаимодействия с RTU для сбора и управления данными. В системе также есть HMI для отображения данных, полученных от RTU. Этот сервер MTU подключен к локальной сети с другими системами, такими как архиваторы, системы отчетности и т. д. MTU с помощью маршрутизатора подключается к RTU, находящимся в другой сети. На рисунке 1 показана типовая архитектура АСУ ТП.

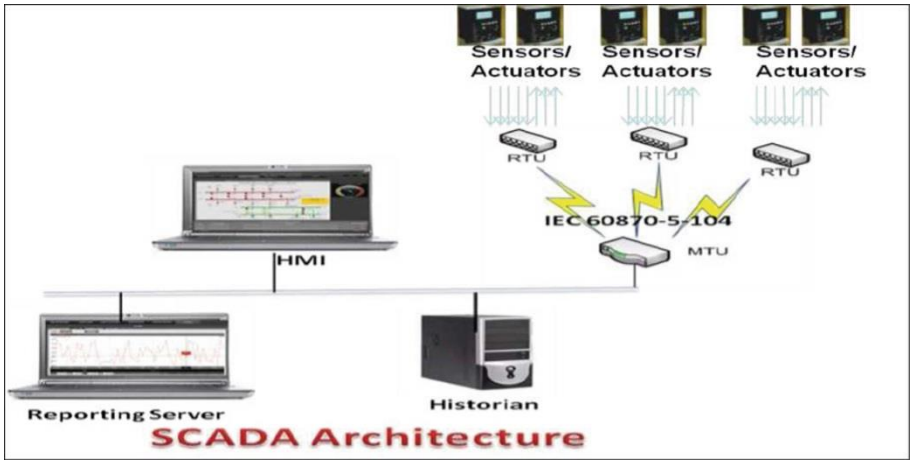


Рисунок 1. Типовая архитектура АСУ ТП

Возможны следующие воздействия на сети АСУ ТП:

- перехват данных между RTU и MTU;
- изменение данных между RTU и MTU;
- сохранение данных во временном буфере и воспроизведение данных между RTU и MTU;
- модификация данных / Атака «Человек посередине»: после перехвата связи между RTU и MTU -измерение данных;
- после прослушивания сообщения данные могут передаваться в MTU для каждого запроса. В этом случае злоумышленник ведет себя как RTU;

- проведение злоумышленником DoS-атаки на RTU или MTU. DoS-атака может проводиться на различных уровнях, то есть на уровне канала передачи данных, транспортном уровне и уровне приложений.

IEC 60870-5-104 (IEC 104) - это стандартный протокол, основанный на Ethernet, который в основном используется для дистанционного управления в АСУ ТП. IEC 104 - это протокол, который используется для сетевой связи в области управления АСУ ТП, то есть связи между MTU / центром управления и устройствами RTU. Этот протокол использует аналогичный формат кадра, используемый МЭК 60870-5-101, протокол последовательной линии и использует открытый интерфейс TCP / IP для связи, а стандартный номер порта прослушивания RTU - 2404. Как показано на рисунке 2, МЭК 60870-5 формат кадра -104 разделен на две части, а именно: информацию управления протоколом приложений (APCI, Application Protocol Control Information) и блок данных службы приложений (ASDU, Application Service Data Unit). Часть ASDU имеет дело с фактической передаваемой информацией, а также с ее метаданными. Часть APCI имеет длину 6 октетов и содержит начальный символ, спецификацию длины ASDU и 4 октета поля управления. Протокол IEC 60870-5-104, в основном, поддерживает множество типов данных, которыми необходимо обмениваться в АС ТП-коммуникациях, например, измеренные (напряжение, ток, мощность, температура масла, температура обмотки, частота и т. д.), индикация (состояние выключателя, состояние изолятора и т. д.), сбор данных из RTU в MTU, команды управления из MTU в RTU, передача файлов, данных синхронизации времени и т.д. Сбор данных и команды управления являются основным ядром коммуникаций АСУ ТП.

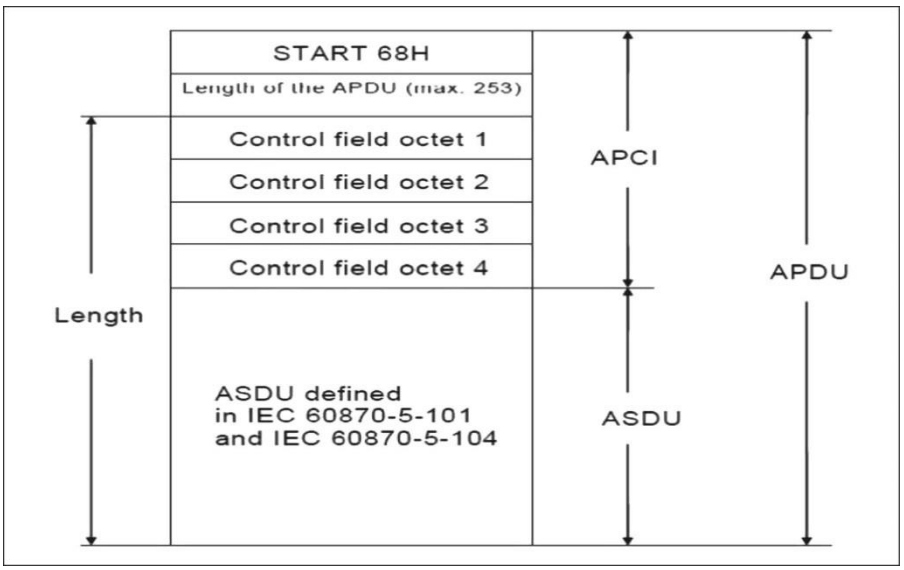


Рисунок 2. Формат пакета протокола IEC104

Уязвимости IEC 60870-5-104 можно в общих чертах перечислить следующим образом:

1) Связанные со стеком TCP / IP. Обычное TCP-соединение может быть атаковано различными способами, что приведет к нежелательным / скомпрометированным ситуациям, например следующим:

- отказ в обслуживании - это ситуация, когда потребляются большие объемы ресурсов в сети и на сервере из-за лавинной рассылки пакетов от злоумышленника, использующего поддельный IP-адрес;

- перехват соединения: перехватывая сеанс TCP, злоумышленник может перенаправить пакеты и захватить TCP-соединение. Злоумышленник использует некоторые способы прогнозирования порядковых номеров на основе текущего обмена данными и отправляет свой собственный пакет с новым порядковым номером. Когда этот пакет подтверждается на другой стороне соединения, синхронизация теряется. Этот метод можно комбинировать с атакой ARP, чтобы получить постоянный контроль над захваченным TCP-соединением;

- вето TCP: злоумышленник, который может перехватить сообщение и может правильно предсказать порядковый номер и размер следующего пакета, который будет отправлен, может внедрить вредоносный пакет. Поскольку

пакет с правильным порядковым номером и размером полезной нагрузки уже получен, подлинный пакет, отправленный отправителем, будет проигнорирован получателем. Это относительно менее эффективно, но практически невозможно обнаружить (рисунок 3);

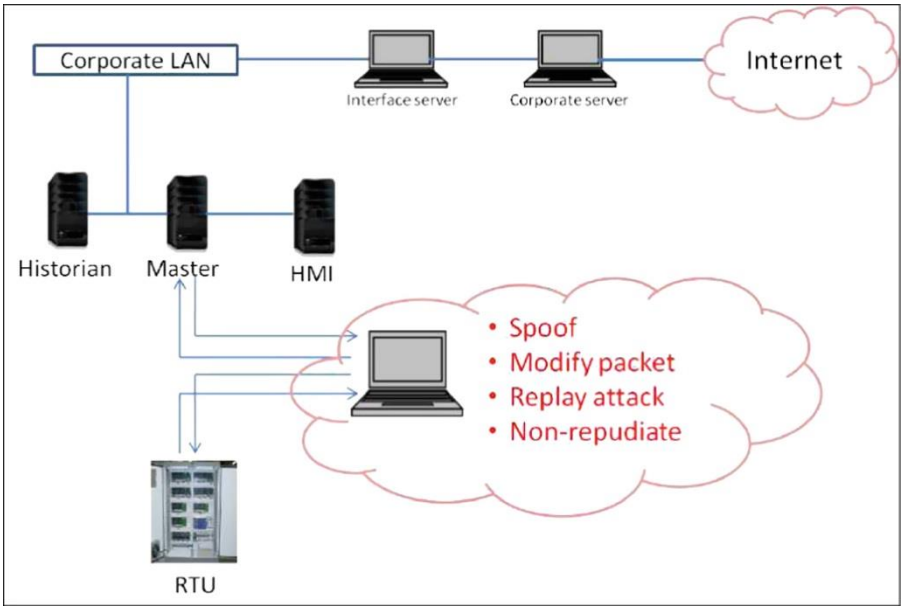


Рисунок 3. Уязвимости связи АСУ ТП

2) Уязвимости в протоколе из-за отсутствия встроенных механизмов безопасности. Из-за того, что для обеспечения безопасности в протоколе не принимается никаких мер, связь IEC 60870-5-104 всегда уязвима для атак, как показано ниже:

- конфиденциальность: поскольку это обычный текстовый обмен данными, перехватчик может легко получить доступ к фактическим передаваемым данным. Протокол, являющийся открытым стандартом, позволяет злоумышленнику правильно интерпретировать пакеты;

- целостность: злоумышленник, успешно запустивший атаку типа MITM, обладая достаточными знаниями о протоколе, может изменить пакет на пути к MTU, чтобы указать неверные значения измерений и неправильный статус устройств. Это, в свою очередь, может заставить оператора предпринять несоответствующие действия;

- аутентичность: протокол не имеет возможности аутентифицировать устройство связи. Это позволяет злоумышленнику ввести пакет для его

отправки в RTU. Введенный пакет может иметь команду для выполнения задачи, которая приводит к катастрофической ситуации на подстанции.

Существуют и другие угрозы, такие как атака MITM и отказ от авторства. Хотя атакам воспроизведения можно противодействовать, только в определенной степени и не полностью, путем эффективного использования порядковых номеров, которые по своей сути присутствуют в протоколе, неотказуемость не попадает в тот же уровень угрозы.

MITM - это кибератака, при которой злоумышленник вступает в диалог между двумя устройствами, выдает себя за оба устройства и получает доступ к информации двух сторон при обмене данными и сообщениями управления. Основная цель атаки MITM состоит в том, чтобы получить конфиденциальную информацию, включая технологические данные и команды управления. Атака MITM использует метод, называемый ARP poisoning. Это можно сделать, используя защиту двух устройств. Во-первых, ответ протокола разрешения адресов является доверенным. Любому подключенному вредоносному устройству сеть будет доверять для отклонения и перехвата сетевого трафика АСУ ТП. Кроме того, атака MITM позволяет третьей стороне перехватывать и захватывать данные для получения дальнейших результатов, а также изменять данные в реальном времени перед их отправкой жертве. После завершения ARP-атаки пользователи могут быть перехвачены, пока отправляются технологические данные и принимаются команды управления. Эту активность можно отслеживать с помощью ПО Wireshark, которое представляет собой открытый и бесплатный анализатор пакетов. Wireshark - это платформа, которая может работать в Windows, Linux и т. д. Защита сети может осуществляться с помощью системы обнаружения вторжений. Сетевые администраторы должны соблюдать такие правила сети, чтобы предотвратить отравление MITM. Анализ моделей трафика для выявления необычного поведения. Сеть должна иметь надежные брандмауэры и интернет-протоколы для

предотвращения несанкционированного доступа. Необходимо использовать стороннее программное обеспечение для проникновения, инструменты и шифрование по протоколу HTTPS, чтобы помочь обнаружить и заблокировать попытки перехвата. Для безопасности необходимо установить высокоактивное защищенное от вирусов и вредоносных программ программное обеспечение, которое включает сканер и запускается в системе для перезагрузки. Атаки типа «злоумышленник посередине» (MITM) часто основываются на вредоносном ПО. Запуск обновленного антивирусного программного обеспечения безопасен. Лучшая защита от перехвата связи - это зашифрованная форма. Эффективная процедура предотвращения перехвата Интернет-протокола (IP-адресов) - это включение двухфакторной аутентификации. Помимо аутентификации вашей системы, необходимо использовать парольную информацию.

ARP - это аббревиатура протокола разрешения адресов, при котором злоумышленники отправляют измененные сообщения ARP по локальной сети, чтобы связать MAC-адреса злоумышленников и сделать сервер легитимным. Наконец, когда третий системный MAC-адрес подключен к аутентифицированному IP-адресу, он начнет получать любые данные, предназначенные для IP-адреса. ARP poisoning может перехватить любую систему и изменить или остановить передачу данных. Атаки с ARP poisoning могут происходить только в локальной сети, которая как сеть АСУ ТП является проводным соединением.

Захват Ethernet-пакетов в опытной установке будет осуществляться ПО Ettercap-G, который представляет собой графический пользовательский интерфейс, подходящий для атаки типа «человек посередине». Во-первых, был проверен порт Ethernet, в котором осуществлялась связь как RTU, так и MTU. В этом эксперименте была выбрана третья система, на которую установили Ettercap-G и выполнили отравление MITM. Базовое представление об атаке поясняется на рисунке 4. Во-вторых, система с ПО Ettercap-G к АСУ

ТП через коммутатор, затем были просканированы доступные хосты в сети. После сканирования были обнаружены IP-адреса RTU и MTU и MAC-адреса в сети. Эти два IP-адреса были выбраны для кибератаки с помощью инструмента Ettercap-G.

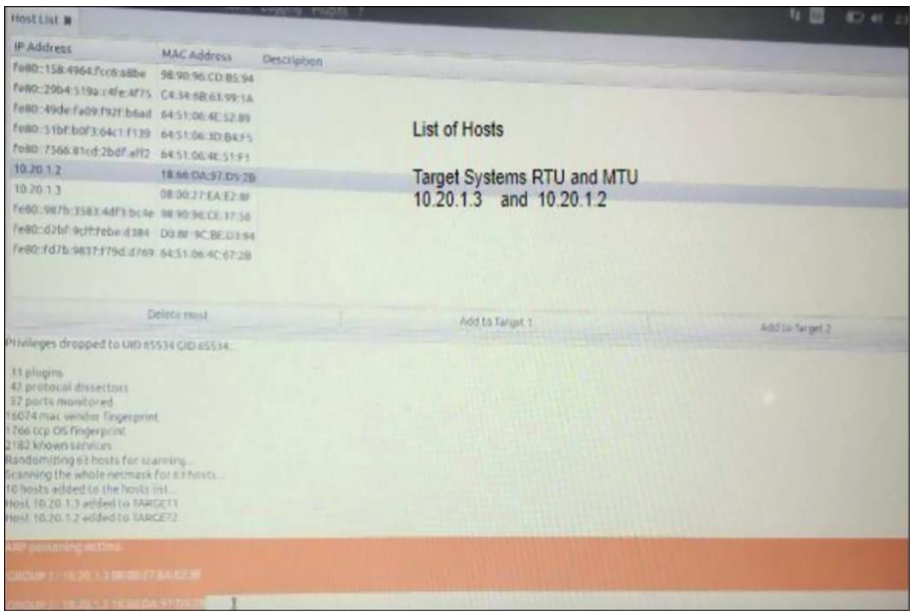


Рисунок 4. Сканирование на предмет наличия RTU и MTU IP

Учитывая все уязвимости, упомянутые ранее, можно сделать вывод, что сеть АСУ ТП не застрахована от кибератак. Теперь необходимо установить последствия любого такого кибер-злоумышленника. С этой целью в исследовательской лабораторной установке был проведен эксперимент с использованием набора COPS SCADA Lab и других инструментов, связанных с этим экспериментом: Ettercap-G для атаки на конкретный IP-адрес, RTU или MTU, и Wireshark для мониторинга пакетных данных. Цель эксперимента - провести исследовательскую работу, которая может реализовать или повлиять на кибератаку на сеть АСУ ТП. Сторонняя организация, известная как MITM Attacker, входит в сеть АСУ ТП и нарушает связь RTU и MTU, изменяя адрес соответствующей системы. В более ранних случаях связь между RTU и MTU была непрерывной, и обе системы имели одинаковый машинный адрес. Был использован инструмент Ettercap-G, который перехватил пакеты и изменил машинный адрес RTU и MTU в сети. Это означает, что в более раннем случае

RTU получает ответ от MTU, а MTU получает ответ от RTU, но когда из-за прерывания третьей стороны в сети АСУ ТП третья сторона меняет системный адрес и заменяет адрес системы на его адрес. Лабораторная установка АСУ ТП с параметрами сети показана на рисунке 5.

Рисунок 5. Лабораторная установка

MAC-адреса RTU и MTU были изменены MAC злоумышленника, как показано на рисунке 6:

1) MTU / 10.20.1.2 18: 66: da: 97: d5: 2b;
2) RTU / 10.20.1.3 08: 00: 27: ea: e2: 8f;
3) Атакующий / 10.20.1.4 8c: 16: 45: c4: 98: 71.

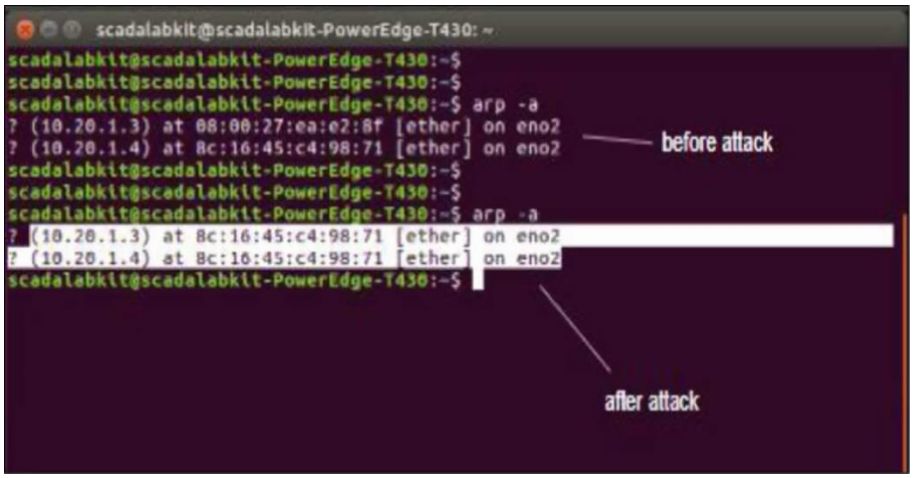


Рисунок 6. Атака тип MITM

В таблице 1 показано изменение MAC RTU после атаки MITM.

Таблица 1 - Изменение MAC RTU после атаки MITM

Источник	MAC1	Приемник	MAC2	Примечание
MTU	18:66:da:97:d5:2b	RTU	18:66:da:97:d5:2b	Без сбоя
RTU	08:00:27:ea:e2:8f	MTU	18:66:da:97:d5:2b	Без сбоя
Атакующий	8c:16:45:c4:98:71	RTU	8c:16:45:c4:98:71	Сбой

В таблице 2 показано изменение MAC MTU после атаки MITM.

Таблица 2 - Изменение MAC MTU после атаки MITM

Источник	MAC1	Приемник	MAC2	Примечание
MTU	08:00:27:ea:e2:8f	MTU	18:66:da:97:d5:2b	Без сбоя
RTU	18:66:da:97:d5:2b	RTU	08:00:27:ea:e2:8f	Без сбоя
Атакующий	8c:16:45:c4:98:71	MTU	8c:16:45:c4:98:71	Сбой

Таким образом, сети АСУ ТП, как и любые коммуникационные сети, уязвимы для кибератак. Принятие протоколов открытого стандарта IEC 60870-5-104 позволило преодолеть проблемы взаимодействия в системах АСУ ТП, но имело место уязвимость, поскольку эти протоколы не были разработаны с учетом требований безопасности. Из-за дополнительных уязвимостей, которые несут сети TCP / IP, связь АСУ ТП становится более уязвимой. IEC 60870-5-104 - это простой текстовый протокол без механизма аутентификации. Атаки MITM легче проводить в сети АСУ ТП, чем передача данных в

текстовом формате, и доказали, что она более уязвима для различных атак. Связь АСУ ТП требует аутентификации, авторизации, шифрования, безопасности конечных точек, безопасности связи и т. д. Для надежной работы энергосистемы. Таким образом, необходимо активно изучать уязвимости и анализировать связанный с этим риск, а также принимать соответствующие меры безопасности для предотвращения доступа неавторизованных пользователей из сторонней системы или сети связи.

1.1.1. Безопасный метод маршрутизации и методы идентификации MITM - атак

Рассмотрим метод маршрутизации с использованием теории графов, который добавляет задачу безопасности при отправке сообщений, чтобы обойти атаки, основанные на сниффинге, такие как MITM-атаки. Чтобы доказать действенность этого метода, было проведено моделирование для подтверждения того факта, что этот новый способ обработки не оказывает большого влияния на производительность маршрутизатора при расчете маршрутов и диспетчеризации сообщений.

Для того чтобы противостоять такой атаке, мы предлагаем усилить операции маршрутизатора, чтобы иметь возможность пересылать сегменты одного и того же пакета по разным путям (рисунок 7). Для этого предлагается выполнить два основных шага:

- используя теорию графов, создать эффективный алгоритм (pathFinder), который вычисляет все возможные пути от источника к месту назначения на основе матрицы смежности.
- выбрать один из возможных путей, указанных в Pathfinder алгоритме, все комбинации, отвечающие ряду критериев такие как безопасность, скорость, размер буфера и т. д.

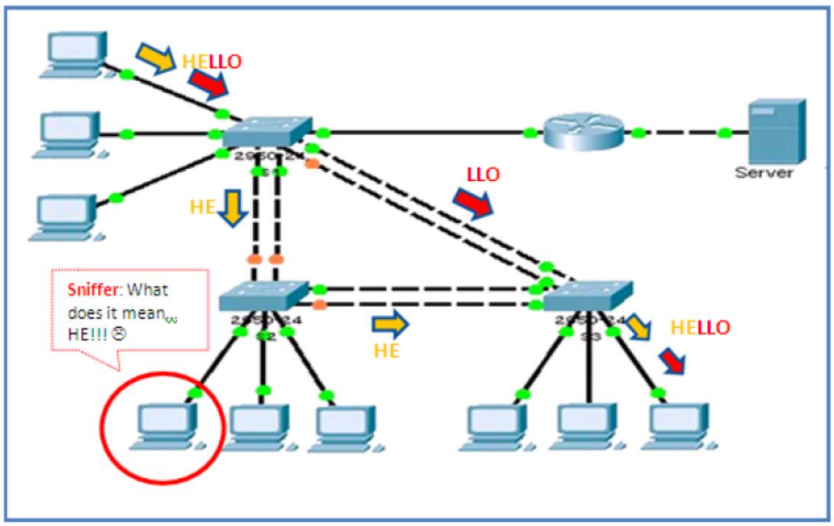


Рисунок 7. Сниффинг в описанной ситуации

1.1.1.1. Теория графов и её применение в компьютерных сетях

В информатике теория графов - это изучение графов, которые представляют собой математические структуры, используемые для моделирования парных взаимоотношения между объектами. «Граф» в этом контексте состоит из «вершин» или «узлов» и линий, называемых ребрами, которые их соединяют. Граф может быть неориентированным, что означает, что нет различия между двумя вершинами связанных с каждым ребром, или его ребра могут быть направлены от одной вершины к другой.

Для простоты рассмотрим простой граф; без параллельных краев или петель; который можно просто определить парой наборов: $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, где V - множество вершин, E - множество ребер, сформированных парами вершин.

В этом состоянии каждый граф может быть представлен как простая матрица, называемая матрицей смежности. Матрица смежности $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, это матрица размерности n*n $\mathbf{D} = \mathbf{d_{ij}}$, где n – количество узлов в G и $\mathbf{d_{ij}}$ представляет собой вес каждого ребра:

$$
\begin{array}{c} \\ n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \\ n_6 \end{array}
\begin{array}{c}
\begin{array}{cccccc} n_1 & n_2 & n_3 & n_4 & n_5 & n_6 \end{array} \\
\begin{pmatrix}
0 & w_1 & 0 & 0 & 0 & 0 \\
w_1 & 0 & w_2 & 0 & w_4 & 0 \\
0 & w_2 & 0 & 0 & w_5 & w_6 \\
0 & 0 & 0 & 0 & w_3 & 0 \\
0 & w_4 & w_5 & w_3 & 0 & 0 \\
0 & 0 & w_6 & 0 & 0 & 0
\end{pmatrix}
\end{array}
$$

Моделируется компьютерная сеть в виде графа (рисунок 8), а затем матрица, для того чтобы упростить для роутера с соответствующим алгоритмом прохождение через него и найти все возможные пути от одного узла к другому самым быстрым способом.

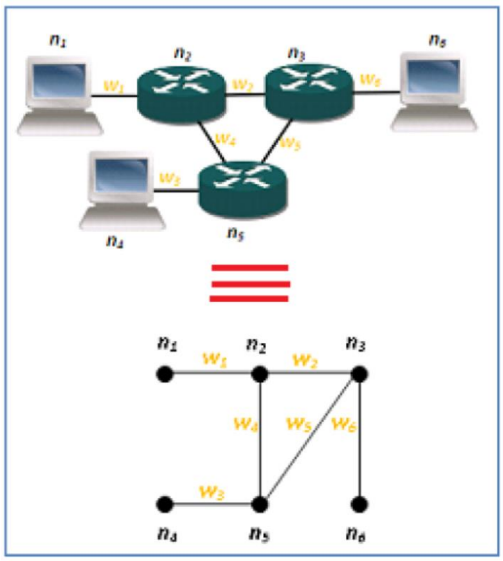


Рисунок 8. Моделирование компьютерной сети в виде графа

Вручную; используя граф, легко найти все возможные пути из одной точки в другую. Из последнего примера для перехода от n1 к n6 имеем:

$$n1 \rightarrow n2 \rightarrow n3 \rightarrow n6$$

и

$$n1 \rightarrow n2 \rightarrow n5 \rightarrow n3 \rightarrow n6$$

Автоматически; Алгоритм будет использован как смоделированный (рисунок 9).

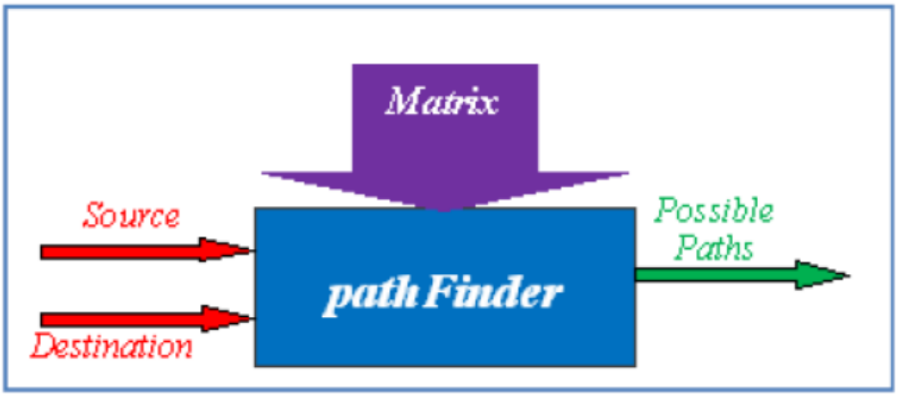


Рисунок 9. Ввод / вывод для pathFinder

Алгоритм, показанный на рисунке 10, построен на основе одной рекурсивной функции для непрерывного просмотра матрицы. Также необходимы некоторые внутренние параметры, такие как:

- путь → хранить путь во время рекурсивного исследования;
- используемый →для отметки используемой и текущей точки;
- n →Порядок квадратной матрицы.

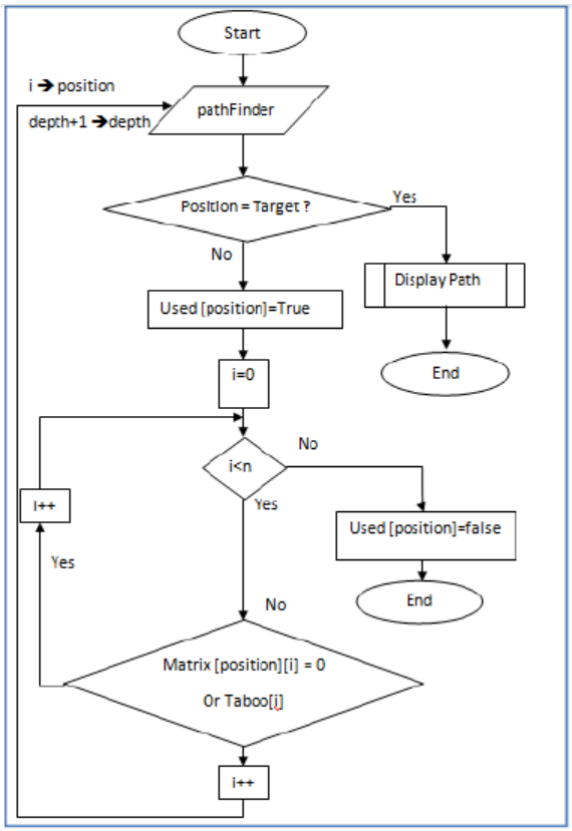
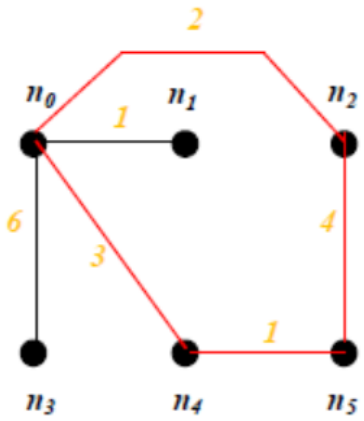


Рисунок 10. Алгоритм работы для pathFinder

Пример сети:



AdjaMatrix [n][n] = {

{0, 1, 2, 6, 3, 0},

{1, 0, 0, 0, 0, 0},

{2, 0, 0, 0, 0, 4},

{6, 0, 0, 0, 0, 0},

{3, 0, 0, 0, 0, 1},

{0, 0, 4, 0, 1, 0}};

pathFinder пример вывода для:

- источник = n0;
- пункт назначения = n2.

```
From 0 to 2 possible routs are :
0 2   = 2
0 4 5 2   = 8
```

При моделировании, выполненном с помощью вышеуказанного алгоритма для сети, содержащую от 2 до 100 узлов, можно сделать вывод о том, что влияние на маршрутизаторы отсутствует, если количество переходов не превышает 12, как показано на рисунке 11.

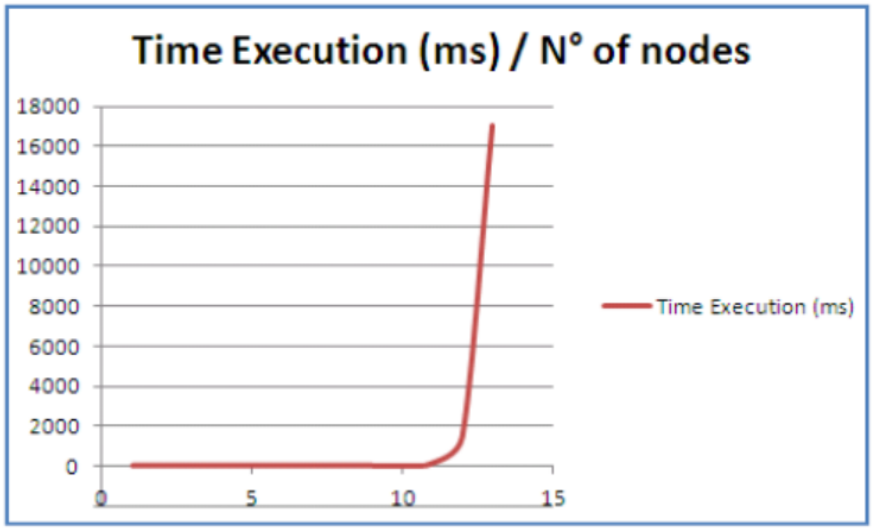


Рисунок 11. Влияние количества узлов (переходов) на производительность

Также возможно логически разделить сеть на маленькую ячейку, содержащую не более 12 переходов (аналогично понятию ОБЛАСТЬ, используемому OSPF [5], рисунок 12).

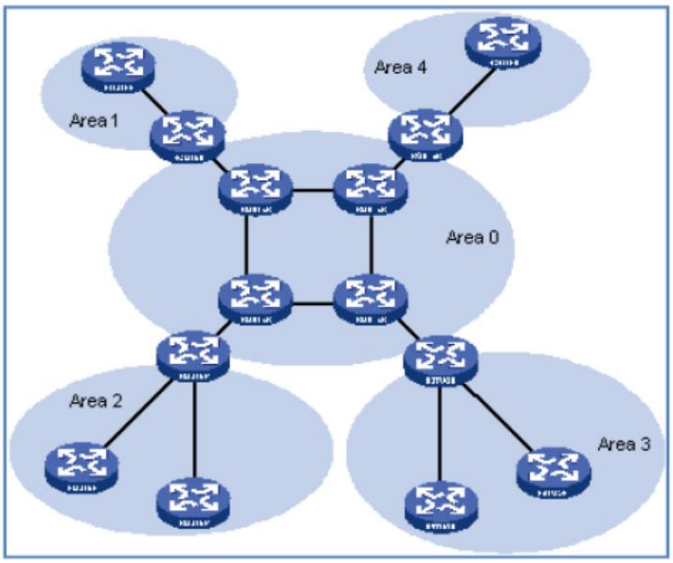


Рисунок 12. Область OSPF

1.1.1.2.Выбор безопасного сочетания пути при отправке сообщения

Рассмотрим некоторые параметры, которые будут использоваться в данном разделе:

- Pn - путь n;

- EPn - набор промежуточных узлов, используемых для Pn;

- wn - общий вес для Pn ($\mathbf{w_n} \in \mathbf{N}$);

- mn - нагрузка, которую следует учитывать на Pn.

В проведенном ранее моделировании и простой пример с очень небольшим количеством возможных путей; для более сложных сетей:

adjaMatrix [n][n] = {

{0, 1, 0, 3, 2, 0, 0, 0, 0},

{1, 0, 1, 0, 1, 0, 0, 0, 0},

{0, 1, 0, 0, 1, 1, 0, 0, 0},

{3, 0, 0, 0, 0, 0, 1, 0, 0},

{2, 1, 1, 3, 0, 4, 1, 1, 2},

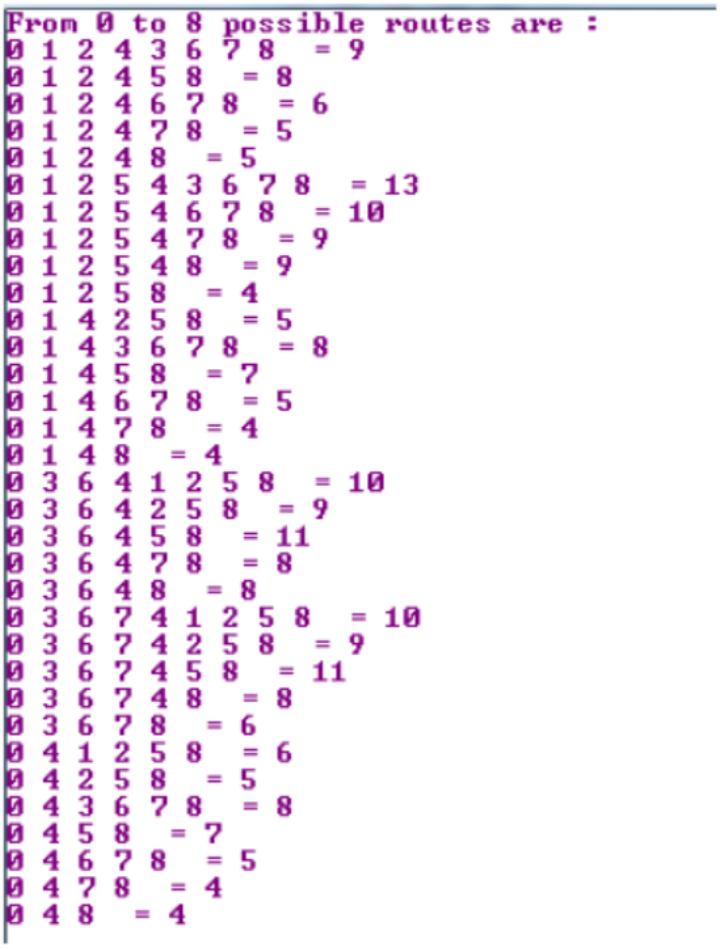
{0, 0, 1, 0, 4, 0, 0, 0, 1},

{0, 0, 0, 1, 2, 0, 0, 1, 0},

{0, 0, 0, 0, 1, 0, 1, 0, 1},

{0, 0, 0, 0, 2, 1, 0, 1, and 0},};

Решение в этом случае предоставит больше вариантов: (Например, источник n0 и пункт назначения n8)



Существует 33 возможных пути (Pn с $\mathbf{n} \in [\mathbf{1} - \mathbf{33}]$) для того чтобы использовать их при отправке сообщений от 0 до 8, но мы не собираемся чтобы использовать все эти комбинации. Лучшим решением будет как минимум два пути (Pi & Pj) с одинаковым весом (wi = wj), а также с использованием разных промежуточных узлов ($EPi \cap EPj = Ø$). Это условие является основным отличием от метода балансировки нагрузки по пакетам, уже предложенным Cisco Маршрутизатором, со стандартным протоколом маршрутизации, таким как RIP, RIPv2, EIGRP, OSPF.

Проверив результат последней симуляции, мы обнаружили, что у нас есть два пути с одинаковым наименьшим общим весом (4), а также с использованием разных узлов - n4 для первого пути и n1, n2 & n5 для второго пути:

$$n_0 \rightarrow n_4 \rightarrow n_8 = 4$$

и

$$n_0 \rightarrow n_1 \rightarrow n_2 \rightarrow n_5 \rightarrow n_8 = 4$$

Поскольку имеется одинаковый вес, количество сообщений для отправки по каждому пути будет одинаковым, следовательно, нагрузка будет 50% на каждый путь.

Когда такое решение невозможно, возможно рассмотреть другой вариант, который заключается в поиске двух путей, удовлетворяющих условию:

$$m_i \times w_i = m_j \times w_j$$

Рассмотрим решение из предыдущего примера:

$$n_0 \rightarrow n_1 \rightarrow n_2 \rightarrow n_5 \rightarrow n_8 = 4$$

и

$$n_0 \rightarrow n_3 \rightarrow n_4 \rightarrow n_8 = 4,$$

где:

$m_1 = 2$ (2 сообщения на P_1, нагрузка 67%)

$w_1 = 4$

$m_2 = 1$ (1 сообщение на P_2, загрузка 33%)

$w_2 = 8$

1.1.1.3.Облегченный алгоритм обнаружения и защиты от MITM-атак для шлюзов IoT с поддержкой WiFi

Интернет-революция изменила определение индустрии с отношениями типа «бизнес-покупатель» (B2C), таких как средства массовой информации, розничная торговля и финансовые услуги. Эта революция привела к появлению Интернета вещей (IoT), вездесущей глобальной вычислительной сети, в которой все и вся подключены к Интернету. Количество подключенных

к сети устройств постоянно увеличивается. Предполагается, что к 2020 году будет подключено около 50 миллиардов устройств (показано на рисунке 13).

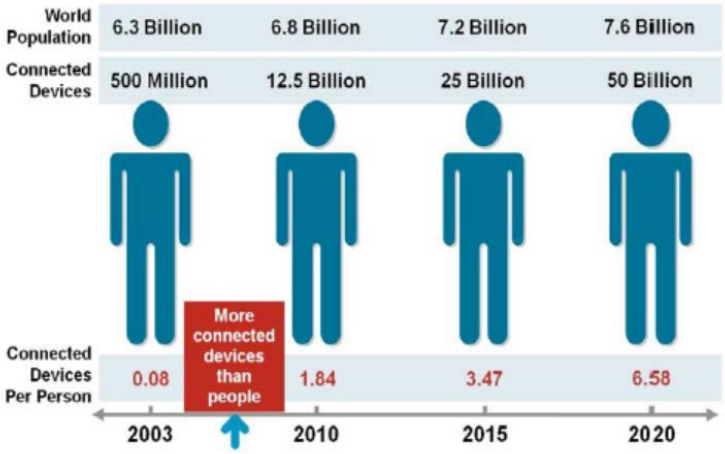


Рисунок 13. Будущее связанных устройств

Прямое подключение этих разнородных устройств к Интернету увеличивает их шансы подвергнуться нескольким угрозам безопасности. Эти угрозы включают в себя [19]:

- клонирование устройств IoT;
- злонамеренную подмену устройств IoT;
- замену прошивки устройств IoT;
- извлечение параметров безопасности;
- подслушивание;
- и, конечно же, атаку типа MITM.

Большинство устройств Интернета вещей используют технологию Wi-Fi, следовательно, они могут быть уязвимы для обычных атак по беспроводной сети.

Рассмотрим восприимчивость IoT-устройств к MITM-атакам (рисунок 14).

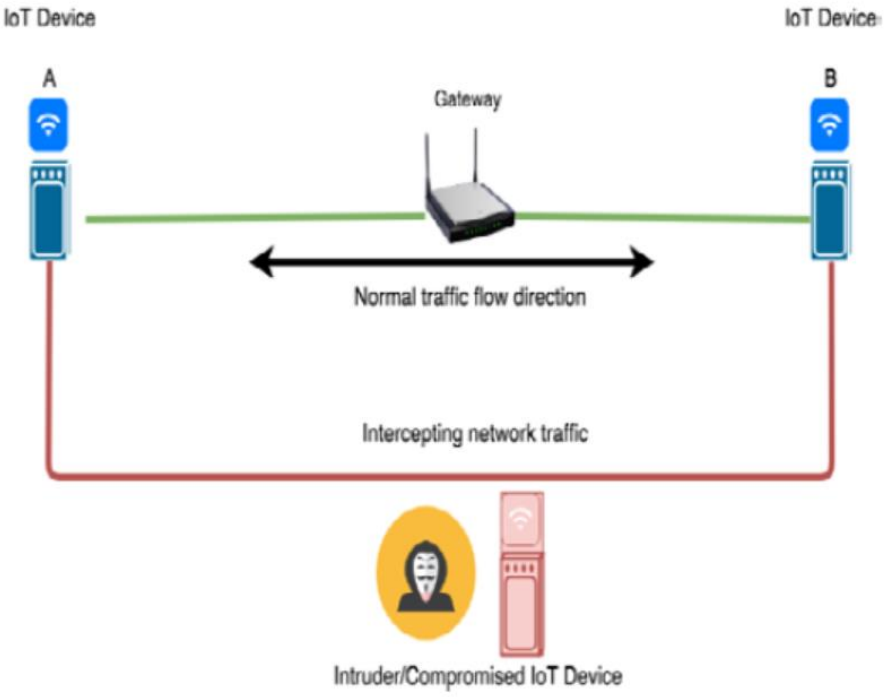


Рисунок 14. MITM атака на IoT-устройства

Эта атака возможна из-за слабости протокола разрешения адресов (ARP). ARP - это протокол, используемый уровнем канала передачи данных для сопоставления IP-адресов с MAC-адресами. Перед инкапсуляцией пакета сетевого уровня во фрейм канального уровня хост, отправляющий пакет, должен знать MAC-адрес получателя. Учитывая IP-адрес хоста, чтобы найти его MAC-адрес, исходный узел передает пакет запроса ARP, который запрашивает MAC-адрес владельца IP-адреса. Этот запрос получают все узлы внутри локальной сети (LAN). Узел, которому принадлежит этот IP-адрес, отвечает своим MAC-адресом (одноадресная передача).

ARP - это протокол без сохранения состояния, и его система кеширования недостаточно безопасна. Он принимает ответы ARP, не учитывая, был ли отправлен запрос ARP. Эта уязвимость может быть использована злоумышленником для инициирования атаки MITM. Атака типа «отказ в обслуживании» (DoS) может произойти, если злоумышленник отбрасывает полученный пакет, не пересылая его по назначению.

Некоторые решения защиты от MITM не подходят для систем IoT из-за определенных характеристик IoT, которые могут повлиять на работу систем обнаружения вторжений (IDS). Рассмотрим облегченный алгоритм обнаружения и защиты MITM для шлюзов Интернета вещей с поддержкой WiFi. Алгоритм работает для статических IP-адресов хоста и IP-адресов, назначаемых через DHCP. Применяется технология диспетчеризации асинхронного метода (AMD, Asynchronous Method Dispatch) для обнаружения и предотвращения MITM атак, чтобы снизить накладные расходы на производительность. Используя технологию AMD, сервер может получить запрос, но затем приостановить его обработку, чтобы как можно было освободить поток отправки. Когда обработка возобновляется и результаты доступны, сервер может предоставить свои услуги для доставки клиенту необходимой информации.

В рамах данного исследования были разработаны ряд традиционных механизмов обнаружения MITM ARP-спуфинга. Была представлена недорогая встроенная система IDS, способная автоматически эффективно обнаруживать и предотвращать атаки с подменой ARP, но для этого ее необходимо было подключить к коммутатору или концентратору.

Одноадресный запрос ARP был предложен вместо широковещательного запроса ARP путем назначения IP-адресов через DHCP. Предполагается, что DHCP разрешит сопоставление IP / MAC без необходимости широковещательной передачи. Этот подход не применим для статических IP-адресов.

Рассмотрим типовые решения:

- расширение с обратной совместимостью для ARP, основанное на криптографии с открытым ключом для аутентификации ответов ARP. Все хосты создают пары открытого и закрытого ключей во время первоначального контакта с сетью и отправляют их с подписанными сертификатами

уполномоченному распространителю ключей (AKD). Этот метод приводит к накладным расходам производительности и также невыполним в беспроводной сети;

- протокол ARP на основе билетов (TARP), который реализует безопасность путем распространения централизованно выпущенных защищенных билетов IP/MAC через DHCP. Эти билеты отправляются клиентам, когда они присоединяются к сети, и впоследствии распределяются через существующие сообщения ARP. Это приводит к снижению производительности при генерации пар открытого/закрытого ключей и не подходит для динамических сетей, где узлы могут присоединяться и покидать сеть в любое время.

Предложен метод предотвращения отравления кеша ARP в беспроводной локальной сети путем реализации механизма защиты в точке доступа (AP, access point). AP составляет список правильных сопоставлений IP-адресов с MAC-адресами, отслеживая сообщения DHCP ACK или ссылаясь на файл аренды DHCP, и блокирует все пакеты ARP с ложным отображением на основе созданного списка.

MR-ARP, первый протокол, устойчивый к спуфингу ARP на основе голосования. Когда устройство MR-ARP получает запрос или ответ ARP, объявляющий отображение (IP, MAC) для нового IP-адреса, она опрашивает соседние устройства для получения нового IP-адрес. Для этого механизма голосование может быть справедливым только в том случае, если скорость трафика для отвечающих машин почти одинакова. Это условие может быть выполнено в Ethernet, но может быть недействительно в сети 802.11 из-за адаптации скорости трафика на основе отношения сигнал / шум (SNR).

Для преодоления ограничений MR-ARP был предложен EMR-ARP. Новый протокол улучшает процедуру голосования за счет включения вычислительных задач. Этот механизм требует от устройств слишком много вычислительного времени.

Метод GMR-ARP является улучшением по сравнению с EMR-ARP. Он уменьшил накладные расходы трафика голосования (ниже, чем MR-ARP и EMR-ARP). Поскольку запросы на голосование используются в широковещательной рассылке, этот подход также может вызвать дополнительные накладные расходы.

Из связанных работ можно сделать вывод о том, что традиционный подход, используемый при обнаружении атак MITM, не применим к устройствам IoT из-за ограничений ресурсов; отсюда и потребность в легком алгоритме обнаружения MITM.

Алгоритм обнаружения и защиты от MITM состоит из трех подуровневых процессов, координируемых контроллером межпроцессного взаимодействия (показано на рисунке 15). Три процесса подуровня состоят из анализатора пакетов, подпроцесса обнаружения и подпроцесса защиты. Асинхронная диспетчеризация метода (AMD) используется в межпроцессном взаимодействии.

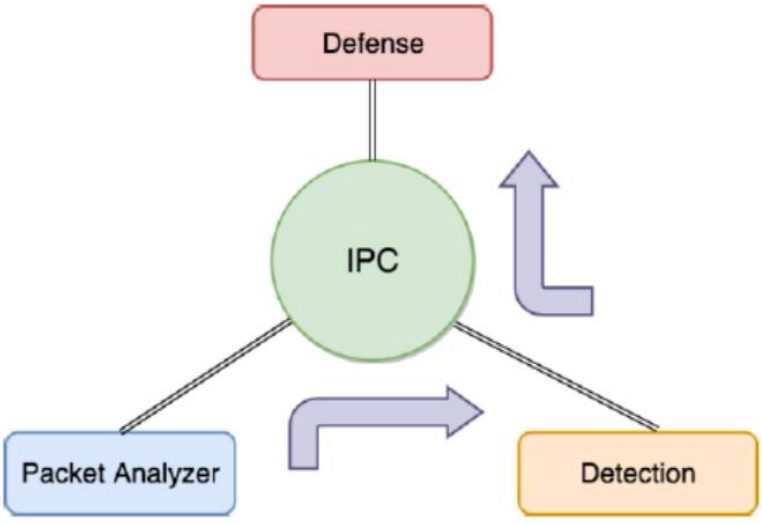


Рисунок 15. Взаимодействие в рамках обнаружения атак

Алгоритм работает, сначала обнаруживая (IP, MAC) отображения клиентских узлов, подключенных к шлюзу. Допустимые отображения (IP, MAC) добавляются в кеш ARP шлюза, которым в данном случае является AP.

Подпроцесс анализатора пакетов отвечает за захват и декодирование беспроводного трафика. Были записаны и проанализированы следующие пакеты:

- EAPOL/EAP (Расширяемый протокол аутентификации), показан на рисунке 16;

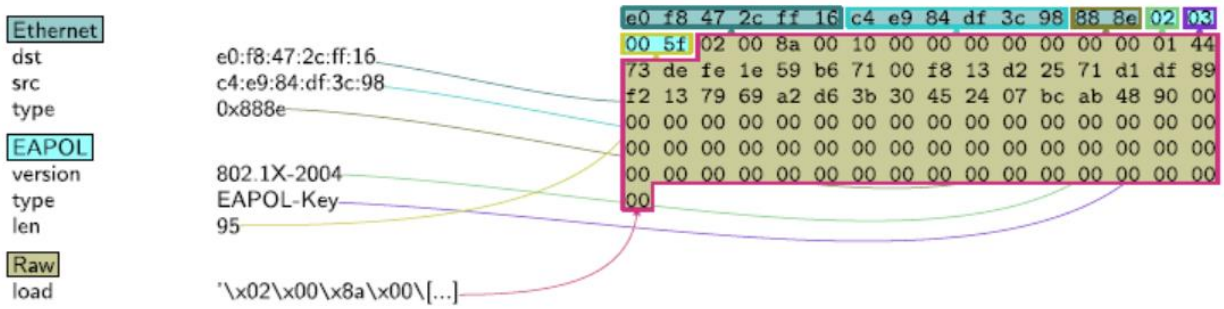


Рисунок 16. Декодированная структура пакета EAPOL

- DHCP (показан на рисунке 17);

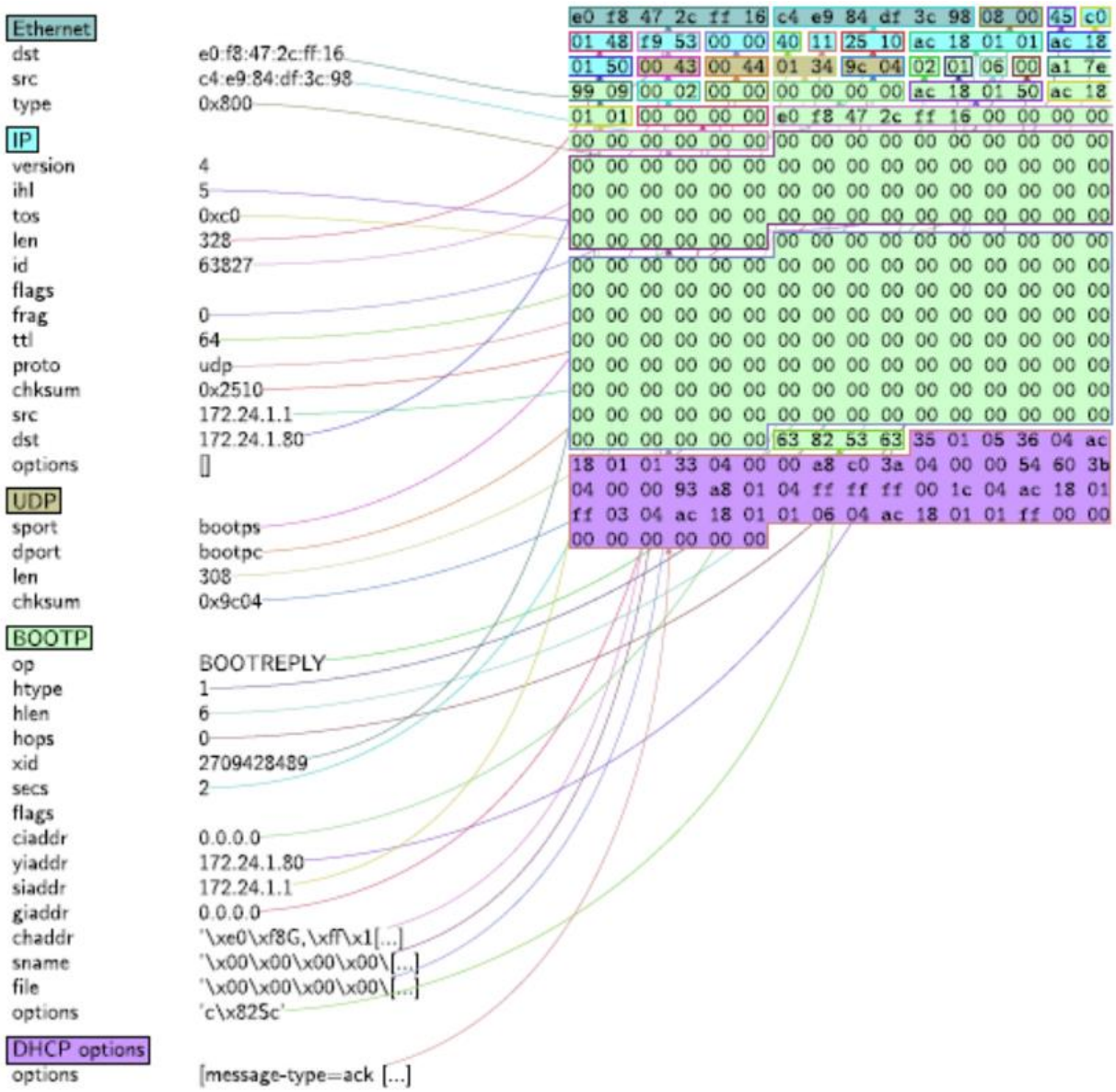


Рисунок 17. декодированная структура пакета DHCP

- IP (показан на рисунке 18);

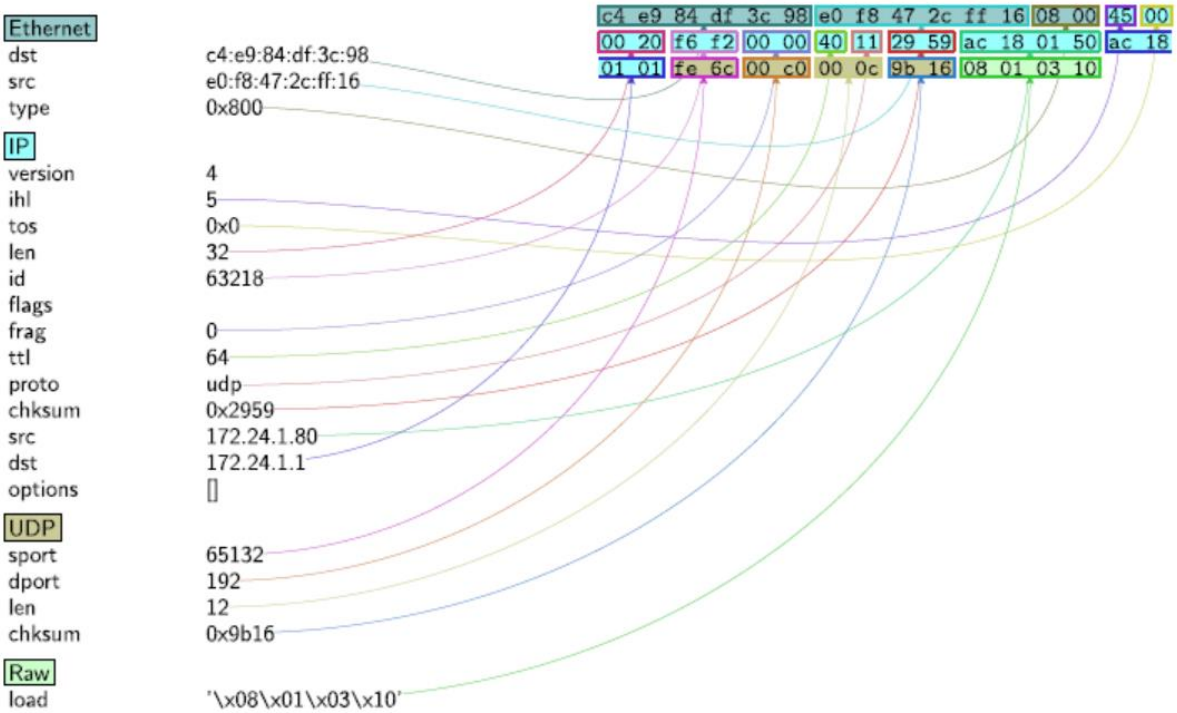


Рисунок 18. декодированная структура IP-пакета

- ARP (показан на рисунке 19).

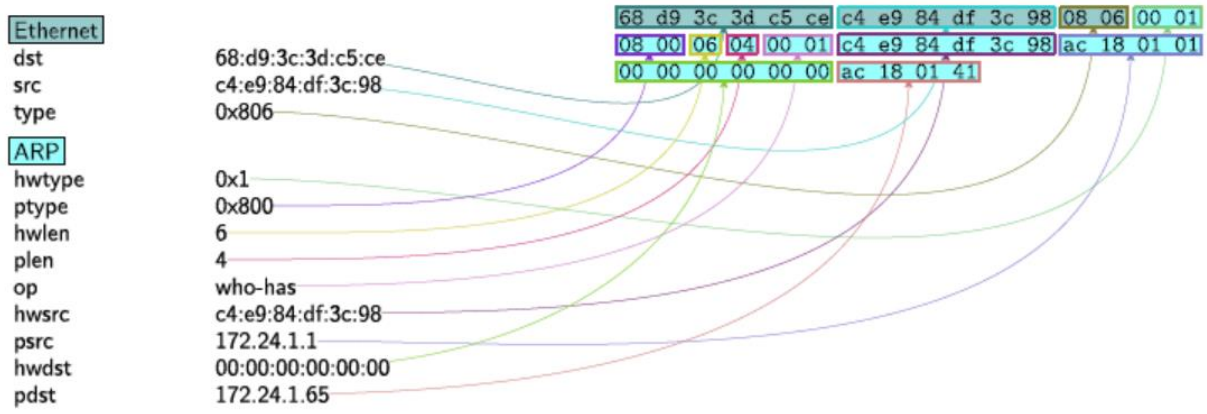


Рисунок 19. Декодированная структура пакета ARP

Алгоритм декодирования пакетов показан на рисунке 20.

Algorithm 1 Packet Decoding Algorithm

```
1: while True do
2:     if sniff(interface) then
3:         if packet.hasLayer(EAPOL) then
4:             ← IP, MAC, Time-Seen
5:         else if packet.hasLayer(DHCP) then
6:             ← IP, MAC, Time-Seen
7:         else if packet.hasLayer(IP) then
8:             ← IP, MAC, Time-Seen
9:         else if packet.hasLayer(ARP) then
10:            ← IP, MAC, Time-Seen
11:        end if
12:    end if
13: end while
```

Рисунок 20. Алгоритм декодирования пакета

Подпроцесс обнаружения хранит записи виртуального кеша ARP с допустимыми отображениями IP и MAC и временем последнего посещения. При получении нового ответа ARP подпроцесс обнаружения проверяет виртуальный кеш ARP, чтобы проверить, существует ли такая запись (IP, MAC). Если запись не существует, новое отображение (IP, MAC) добавляется в виртуальный кеш ARP, а также в кеш ARP шлюза.

Если полученный (IP, MAC) ответа ARP содержит тот же IP и MAC-адрес, что и одна из записей в виртуальном кеше ARP, значение времени просмотра этой записи обновляется. Если MAC-адрес ответа ARP совпадает с одной из записей в виртуальном кеше ARP, но IP-адрес изменяется, подпроцесс обнаружения выполняет обратный ARP, для того чтобы определить, является ли хост, который ранее имел связанный MAC-адрес, действующим. Если хост действующий, то он помечается как атака MITM. Если хост не действующий, выполняется два теста. Первый тест состоит в том, чтобы определить число переходов путем выполнения трассировки IP-адреса хоста. Если количество переходов больше 1, это означает, что трафик перехватывается неавторизованным клиентом. В таком случае он помечается как атака MITM. Если счетчик переходов равен 0, возможно, хосту отказано в

обслуживании. Второй тест выполняется для проверки того, является ли данная атака атакой MITM. Вычисляется разница во времени между последней просмотренной записью MAC-адреса в виртуальном кеше ARP и временем входящего ответа ARP. Входящий ответ ARP помечается как атака MITM, если результирующая разница во времени меньше, чем время жизни записи кеша ARP (TTL). Алгоритм обнаружения представлен на рисунке 21.

```
Algorithm 2 Detection Algorithm
 1: if arpData then
 2:     exists, res = arp.findMac(arpData.mac)
 3:     if exists then
 4:         if arpData.ip != res.ip then
 5:             if InvARP(res.ip) then
 6:                 ← Host alive, MITM detected
 7:             else
 8:                 if hopCount(res.ip) != 1 then
 9:                     tDiff = (arpData.t - res.t)
10:                     if tDiff < ARPTTL then
11:                         ← MITM leading to DoS
12:                     end if
13:                 end if
14:             end if
15:         else
16:             ← Update Last Seen
17:         end if
18:     else
19:         ← New ARP Entry
20:     end if
21: end if
```

Рисунок 21. Алгоритм детектирования

Когда конкретный ответ ARP помечен как атака MITM, подпроцесс защиты удаляет IP-адрес хоста, выполняющего атаку подделки, из записи ARP и блокирует весь трафик, исходящий от этого хоста. Алгоритм защиты представлен на рисунке 22.

```
Algorithm 3 Defense Algorithm
1: if mitmData then
2:     deleteARPEntry(mitmData.IP)
3:     dropPackets(mitmData.IP)
4: end if
```

Рисунок 22. Алгоритм защиты

Алгоритм был реализован на Raspberry Pi, выступающей в качестве шлюза для семи устройств IoT (NodeMcu), при этом один из узлов выступал в роли вредоносного клиента (показано на рисунке 23).

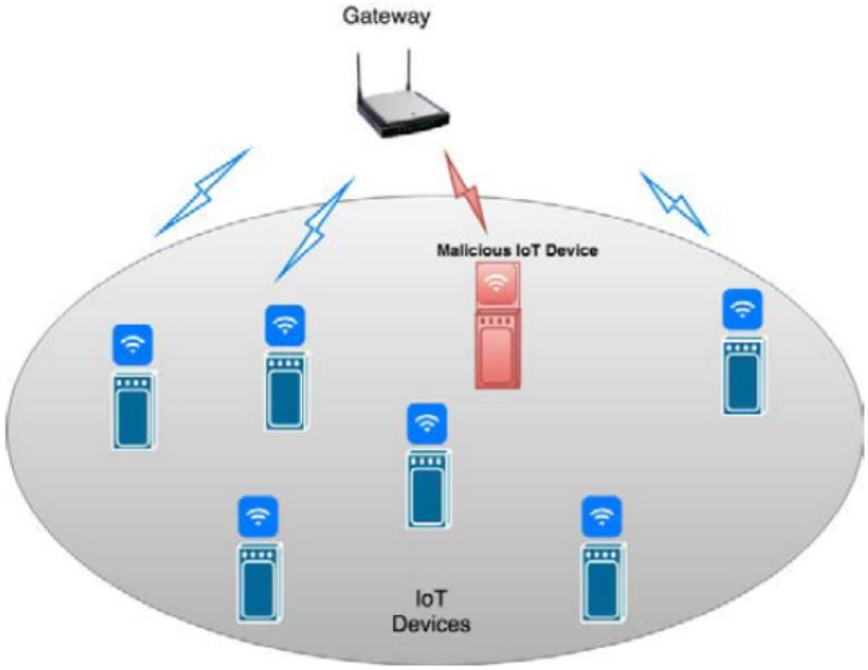


Рисунок 23. Структура исследовательской установки

Для определения эффективности предложенного алгоритма использовались три ключевых показателя производительности: эффективность использования ЦП, скорость обнаружения и задержка в сети (с использованием времени приема-передачи (RTT)).

На рисунке 24 показаны накладные расходы на производительность при реализации алгоритма; в среднем 0,9545%. Результат превосходит показатель, который составлял 1,65%.

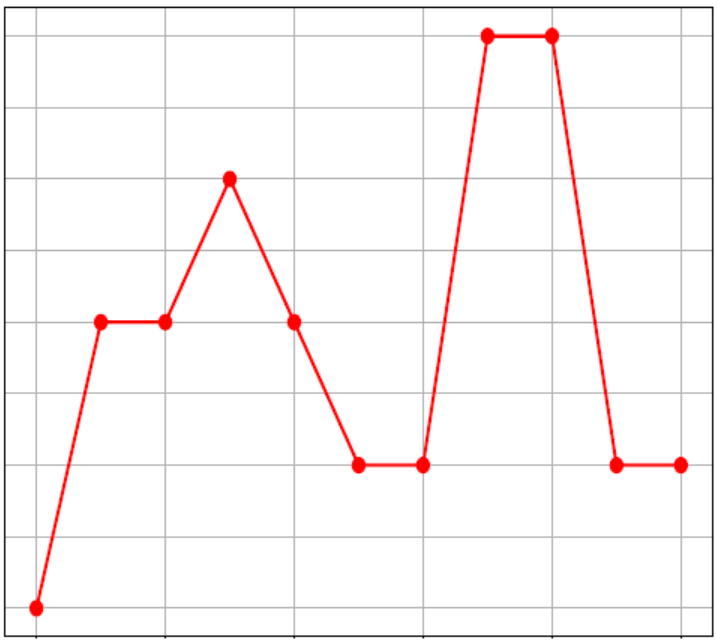


Рисунок 24. График накладных расходов на производительность

Среднее время (показано на рисунке 25) для обнаружения атаки MITM составляет 0,1686 секунды.

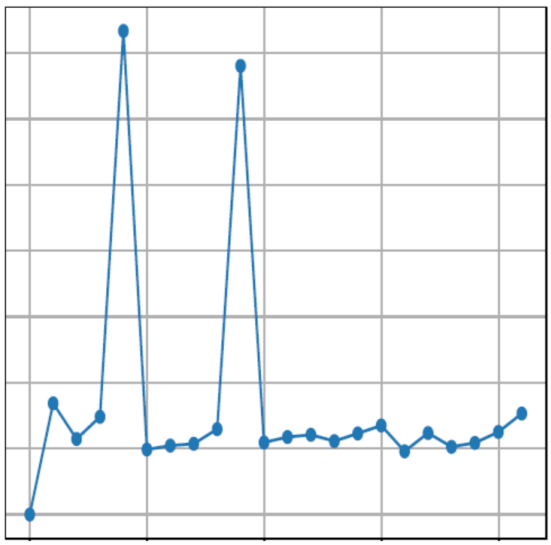


Рисунок 25. Уровень детектирования

Время приема-передачи, которое является мерой задержки в сети, для случая, когда алгоритм не был реализован, составляет 1,298 секунды, в то время как когда алгоритм был реализован, оно составляет 1,335 секунды. Данный показатель демонстрирует, что алгоритм обнаружения и защиты MITM существенно не влияет на задержку сети (показано на рисунке 26).

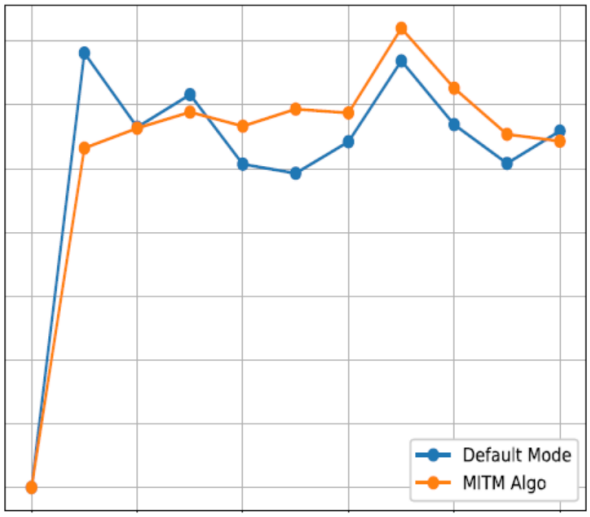


Рисунок 26. График времени приема-передачи сигнала в обоих направлениях

Список использованных источников

1 New secure routing method & applications facing MitM attacks. [Электронный ресурс] - 2020 - Режим доступа: https://www.academia.edu/27555716/New_secure_routing_method_and_amp_applications_facing_MitM_attacks (дата обращения 25.09.2020).

2 LAN security analysis and design. [Электронный ресурс] - 2020 - Режим доступа: https://www.researchgate.net/publication/329183543_LAN_security_analysis_and_design/download (дата обращения 25.09.2020).

3 Man-in-the-middle attack in wireless and computer networking - A review. [Электронный ресурс] - 2020 - Режим доступа: https://ieeexplore.ieee.org/document/8344724 (дата обращения 25.09.2020).

4 Анализ сетевого трафика в режиме реального времени: обзор прикладных задач, подходов и решений. [Электронный ресурс] - 2020 - Режим доступа: https://www.ispras.ru/preprints/docs/prep_28_2015.pdf (дата обращения 25.09.2020).

5 Introduction to TCP Offload Engines. [Электронный ресурс] - 2020 - Режим доступа: https://www.dell.com/downloads/global/power/1q04-her.pdf (дата обращения 28.09.2020).

6 XMC Modules. [Электронный ресурс] - 2020 - Режим доступа: https://www.ecrin.com/datasheets/Acromag/XMC-6280.pdf (дата обращения 26.09.2020).

7 Extending OSI to Network Security. [Электронный ресурс] - 2020 - Режим доступа: https://www.sciencedirect.com/topics/computer-science/protocol-analyzer (дата обращения 21.09.2020).

8 Pcap. [Электронный ресурс] - 2020 - Режим доступа: https://en.wikipedia.org/wiki/Pcap (дата обращения 21.09.2020).

9 Wireshark. [Электронный ресурс] - 2020 - Режим доступа: https://www.wireshark.org (дата обращения 21.09.2020).

10CommView® for WiFi. [Электронный ресурс] - 2020 - Режим доступа:

https://www.tamos.com/products/commwifi/ (дата обращения 28.09.2020).

11Microsoft Message Analyzer Operating Guide. [Электронный ресурс] - 2020 - Режим доступа: https://docs.microsoft.com/en-us/message-analyzer/message-analyzer-tutorial (дата обращения 23.09.2020).

12Добываем Wi-Fi соседа стандартными средствами MacOS. [Электронный ресурс] - 2020 - Режим доступа: https://habr.com/ru/post/347658/ (дата обращения 29.09.2020).

13Security Analysis of MITM Attack on SCADA Network. [Электронный ресурс] - 2020 - Режим доступа: https://www.researchgate.net/publication/342406756_Security_Analysis_of_MITM_Attack_on_SCADA_Network (дата обращения 29.09.2020).

14Detection of MITM attack in LAN environment using payload matching. [Электронный ресурс] - 2020 - Режим доступа: https://www.researchgate.net/publication/279192958_Detection_of_MITM_attack_in_LAN_environment_using_payload_matching (дата обращения 29.09.2020).

15Detection and Prevention of Man-in-the-Middle Spoofing Attacks in MANETs Using Predictive Techniques in Artificial Neural Networks (ANN). [Электронный ресурс] - 2020 - Режим доступа: https://www.researchgate.net/publication/330520159_Detection_and_Prevention_of_Man-in-the-Middle_Spoofing_Attacks_in_MANETs_Using_Predictive_Techniques_in_Artificial_Neural_Networks_ANN (дата обращения 29.09.2020).

16Regulators Warn of Man-in-the-Middle Attack Risks. [Электронный ресурс] - 2020 - Режим доступа: https://www.bankinfosecurity.com/regulators-warn-man-in-the-middle-attack-risks-a-9813 (дата обращения 29.09.2020).

17man-in-the-middle attack (MitM) [Электронный ресурс] - 2020 - Режим доступа: https://csrc.nist.gov/glossary/term/man_in_the_middle_attack (дата обращения 29.09.2020).

18ENISA. [Электронный ресурс] - 2020 - Режим доступа: https://www.enisa.europa.eu (дата обращения 29.09.2020).

19Lightweight Man-In-The-Middle (MITM) Detection and Defense Algorithm for WiFi-Enabled Internet of Things (IoT) Gateways. [Электронный ресурс] - 2020 - Режим доступа: https://www.academia.edu/38250498/Lightweight_Man_In_The_Middle_MITM_Detection_and_Defense_Algorithm_for_WiFi_Enabled_Internet_of_Things_IoT_Gateways (дата обращения 29.09.2020).

20MITM 2: The OSI model (layer 1-2-3). [Электронный ресурс] - 2020 - Режим доступа: https://toschprod.wordpress.com/2012/01/17/mitm-2-the-osi-model/ (дата обращения 29.09.2020).

21Как перехватить трафик? [Электронный ресурс] - 2020 - Режим доступа: https://itsecforu.ru/2020/02/05/как-перехватить-трафик-в-коммутируем/ (дата обращения 29.09.2020).

22Perform A Man In The Middle Attack With Kali Linux & Ettercap. [Электронный ресурс] - 2020 - Режим доступа: https://medium.com/@thamihardik8/perform-a-man-in-the-middle-attack-with-kali-linux-ettercap-6cd848e1a407 (дата обращения 30.09.2020).

23Creating an Evil Twin Wireless Access Point to Eavesdrop on Data. [Электронный ресурс] - 2020 - Режим доступа: https://null-byte.wonderhowto.com/how-to/hack-wi-fi-creating-evil-twin-wireless-access-point-eavesdrop-data-0147919/ (дата обращения 30.09.2020).