

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ  
Государственное образовательное учреждение  
высшего профессионального образования  
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ»  
Физико-технический факультет  
Кафедра радиофизики и инфокоммуникационных технологий  
Направление подготовки: 10.03.01 Информационная безопасность

## **КУРСОВАЯ РАБОТА**

на тему: **Метод обнаружения сетевых атак**

Дисциплина: Программно-аппаратные средства защиты информации

Студент: Мышкин Артем Евгеньевич

Курс: 3

Семестр: 6

Руководитель: ассистент Рушечников Я. И.

Работа представлена на кафедру «\_\_\_» \_\_\_\_ 2020г. рег. № \_\_\_\_\_

Донецк 2020г

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	3
1. КОМПЬЮТЕРНЫЕ АТАКИ И ТЕХНОЛОГИИ ИХ ОБНАРУЖЕНИЯ .....	5
1.1. Средства обнаружения компьютерных атак .....	8
1.2. Сетевые системы обнаружения атак и межсетевые экраны .....	13
2. ПРОГРАММНЫЕ СРЕДСТВА АНАЛИЗА ЗАЩИЩЕННОСТИ .....	16
2.1. Метод анализа "на лету" .....	17
2.2. Методы автоматизации процессов .....	18
3. ПРОГРАММНЫЕ СРЕДСТВА ОБНАРУЖЕНИЯ УГРОЗ .....	24
3.1. Способы защиты от атак .....	24
3.2. Сценарии защищенности .....	29
4. ОЗНАКОМЛЕНИЕ С WIRESHARK .....	33
4.1. Функциональные блоки Wireshark .....	33
4.2. Захват пакетов .....	35
4.3. Захват файлов .....	35
4.4. Обработка пакета .....	36
5. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНЫХ ИССЛЕДОВАНИЙ .....	37
5.1. Сетевые утилиты .....	37
5.2. Сетевые снифферы .....	38
5.3. Сетевые сканеры .....	40
5.4. Обнаружение снифферов и сканеров в сети .....	41
ЗАКЛЮЧЕНИЕ .....	45
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	46

## ВВЕДЕНИЕ

В связи с увеличением объемов информации, циркулирующих в локальных вычислительных сетях (ЛВС) и расширением спектра задач, решаемых с помощью информационных систем (ИС), возникает проблема, связанная с ростом числа угроз и повышением уязвимости информационных ресурсов. Это обусловлено действием таких факторов, как: расширение спектра задач, решаемых ИС; повышение сложности алгоритмов обработки информации; увеличение объемов обрабатываемой информации; усложнение программных и аппаратных компонентов ЛВС, и соответственно - повышение вероятности наличия ошибок и уязвимостей; повышение агрессивности внешних источников данных (глобальных сетей); появление новых видов угроз.

Необходимо учитывать, что конкурентоспособность предприятий, размер получаемого ими дохода, их положение на рынке существенно зависят от корректности функционирования их информационной инфраструктуры, целостности основных информационных ресурсов, защищенности конфиденциальной информации от несанкционированного доступа. Исходя из этого, возрастают требования к системам защиты ЛВС, которые должны обеспечивать не только пассивное блокирование несанкционированного доступа к внутренним ресурсам сети предприятия из внешних сетей, но и осуществлять обнаружение успешных атак, анализировать причины возникновения угроз информационной безопасности и, по мере возможности, устранять их в автоматическом режиме.

Одним из основных качеств системы защиты информации ЛВС предприятия, удовлетворяющей перечисленным требованиям, является ее адаптивность, т.е. способность анализировать информацию, генерировать на ее основе знания и автоматически изменять конфигурацию системы для блокирования обнаруженных угроз информационной безопасности.

Анализ существующих подходов к реализации систем обнаружения атак

показывает, что большинство программных продуктов, присутствующих в настоящее время на рынке, ориентируется на использование формальных описаний системной активности (сигнатур). Функции обнаружения и регистрации новых видов атак возлагаются в подобных системах на разработчика, выпускающего новые сигнатуры. Данный метод защиты является ненадежным, т.к. он ставит защищенность ИС в зависимость от действий внешнего неконтролируемого источника.

Несмотря на то, что разработка адаптивных систем защиты информации ведется уже достаточно длительное время, ни одно подобное решение не получило сколько-нибудь широкого распространения в силу сложности и малоэффективных используемых алгоритмов, отсутствия в большинстве случаев адекватных инструментов их развертывания и администрирования, а также -пользовательской документации.

Анализ работ, ведущихся в данной области, показывает, что указанная проблема требует дальнейшего изучения как с точки зрения построения адекватных математических моделей предметной области, так и реализации эффективных алгоритмов обнаружения атак и принятия решений, что подтверждает актуальность исследований в данной предметной области.

Объект исследований - система защиты информации корпоративной информационной системы.

Предмет исследований - алгоритмическое и программное обеспечение защиты информации.

Цель работы - повышение эффективности обнаружения атак и принятия решений на основе оперативной оценки риска функционирования ИС с использованием динамических моделей на основе нечетких когнитивных карт.

## 1. КОМПЬЮТЕРНЫЕ АТАКИ И ТЕХНОЛОГИИ ИХ ОБНАРУЖЕНИЯ

До сих пор нет точного определения термина "атака" (вторжение, нападение). Каждый специалист в области безопасности трактует его по-своему. Наиболее правильным и полным я считаю следующее определение.

Атакой на информационную систему называются преднамеренные действия злоумышленника, использующие уязвимости информационной системы и приводящие к нарушению доступности, целостности и конфиденциальности обрабатываемой информации.

На сегодняшний день считается неизвестным, сколько существует методов атак. Говорят, о том, что до сих пор отсутствуют какие-либо серьезные математические исследования в этой области. Но еще в 1996 году Фред Коэн описал математические основы вирусной технологии. В этой работе доказано, что число вирусов бесконечно. Очевидно, что и число атак бесконечно, поскольку вирусы - это подмножество множества атак.

Традиционная модель атаки строится по принципу "один к одному" или "один ко многим", т.е. атака исходит из одного источника. Разработчики сетевых средств защиты (межсетевых экранов, систем обнаружения атак и т.д.) ориентированы именно на традиционную модель атаки. В различных точках защищаемой сети устанавливаются агенты (сенсоры) системы защиты, которые передают информацию на центральную консоль управления. Это облегчает масштабирование системы, обеспечивает простоту удаленного управления и т.д. Однако такая модель не справляется с относительно недавно (в 1998 году) обнаруженной угрозой - распределенными атаками.

В модели распределенной атаки используются иные принципы. В отличие от традиционной модели в распределенной модели используются отношения "многие к одному" и "многие ко многим".

Распределенные атаки основаны на "классических" атаках типа "отказ в обслуживании", а точнее на их подмножестве, известном как Flood-атаки или

Storm-атаки (указанные термины можно перевести как "шторм", "наводнение" или "лавины"). Смысл данных атак заключается в отправке большого количества пакетов на атакуемый узел. Атакуемый узел может выйти из строя, поскольку он "захлебнется" в лавине посылаемых пакетов и не сможет обрабатывать запросы авторизованных пользователей. По такому принципу работают атаки SYN-Flood, Smurf, UDP Flood, Targa3 и т.д. Однако в том случае, если пропускная способность канала до атакуемого узла превышает пропускную способность атакующего или атакуемый узел некорректно сконфигурирован, то к "успеху" такая атака не приведет. Например, с помощью этих атак бесполезно пытаться нарушить работоспособность своего провайдера. Но распределенная атака происходит уже не из одной точки Internet, а сразу из нескольких, что приводит к резкому возрастанию трафика и выведению атакуемого узла из строя. Например, по данным России-Онлайн в течение двух суток, начиная с 9 часов утра 28 декабря 2000 г. крупнейший Internet-провайдер Армении "Арминко" подвергся распределенной атаке. В данном случае к атаке подключились более 50 машин из разных стран, которые посылали по адресу "Арминко" бессмысленные сообщения. Кто организовал эту атаку, и в какой стране находился хакер - установить было невозможно. Хотя атаке подвергся в основном "Арминко", перегруженной оказалась вся магистраль, соединяющая Армению с всемирной паутиной. 30 декабря благодаря сотрудничеству "Арминко" и другого провайдера - "АрменТел" - связь была полностью восстановлена. Несмотря на это компьютерная атака продолжалась, но с меньшей интенсивностью.

Можно выделить следующие этапы реализации атаки:

1. предварительные действия перед атакой или "сбор информации",
2. собственно "реализация атаки",
3. завершение атаки.

Обычно, когда говорят об атаке, то подразумевают именно второй этап, забывая о первом и последнем. Сбор информации и завершение атаки ("заметание следов") в свою очередь также могут являться атакой и могут быть

разделены на три этапа. Сбор информации - это основной этап реализации атаки. Именно на данном этапе эффективность работы злоумышленника является залогом "успешности" атаки. Сначала выбирается цель атаки и собирается информация о ней (тип и версия операционной системы, открытые порты и запущенные сетевые сервисы, установленное системное и прикладное программное обеспечение, и его конфигурация и т.д.). Затем идентифицируются наиболее уязвимые места атакуемой системы, воздействие на которые приводит к нужному злоумышленнику результату. Злоумышленник пытается выявить все каналы взаимодействия цели атаки с другими узлами. Это позволит не только выбрать тип реализуемой атаки, но и источник ее реализации. Например, атакуемый узел взаимодействует с двумя серверами под управлением ОС Unix и Windows NT. С одним сервером атакуемый узел имеет доверенные отношения, а с другим - нет. От того, через какой сервер злоумышленник будет реализовывать нападение, зависит, какая атака будет задействована, какое средство реализации будет выбрано и т.д. Затем, в зависимости от полученной информации и желаемого результата, выбирается атака, дающая наибольший эффект.

Например: SYN Flood, Teardrop, UDP Bomb - для нарушения функционирования узла; CGI-скрипт - для проникновения на узел и кражи информации; PHF - для кражи файла паролей и удаленного подбора пароля и т.п.

Традиционные средства защиты, такие как межсетевые экраны или механизмы фильтрации в маршрутизаторах, вступают в действие лишь на втором этапе реализации атаки, совершенно "забывая" о первом и третьем. Это приводит к тому, что зачастую совершаемую атаку очень трудно остановить даже при наличии мощных и дорогих средств защиты. Пример тому - распределенные атаки. Логично было бы, чтобы средства защиты начинали работать еще на первом этапе, т.е. предотвращали бы возможность сбора информации об атакуемой системе. Это позволило бы если и не полностью предотвратить атаку, то хотя бы существенно усложнить работу

злоумышленника. Традиционные средства также не позволяют обнаружить уже совершенные атаки и оценить ущерб после их реализации, т.е. не работают на третьем этапе реализации атаки. Следовательно, невозможно определить меры по предотвращению таких атак впредь.

В зависимости от желаемого результата нарушитель концентрируется на том или ином этапе реализации атаки. Например: для отказа в обслуживании подробно анализируется атакуемая сеть, в ней выискиваются лазейки и слабые места; для хищения информации основное внимание уделяется незаметному проникновению на атакуемые узлы при помощи обнаруженных ранее уязвимостей.

Рассмотрим основные механизмы реализации атак. Это необходимо для понимания методов обнаружения этих атак.

### **1.1. Средства обнаружения компьютерных атак**

Технология обнаружения атак должна решать следующие задачи:

Распознавание известных атак и предупреждение о них соответствующего персонала.

"Понимание" зачастую непонятных источников информации об атаках.

Освобождение или снижение нагрузки на персонал, отвечающий за безопасность, от текущих рутинных операций по контролю за пользователями, системами и сетями, являющимися компонентами корпоративной сети.

Возможность управления средствами защиты не-экспертами в области безопасности.

Контроль всех действий субъектов корпоративной сети (пользователей, программ, процессов и т.д.).

Очень часто системы обнаружения атак могут выполнять функции, существенно расширяющие спектр их применения. Например,

Контроль эффективности межсетевых экранов. Например, установка системы обнаружения атак после межсетевого экрана (внутри корпоративной



сети) позволяет обнаружить атаки, пропускаемые МСЭ и, тем самым, определить недостающие правила на межсетевом экране.

Контроль узлов сети с неустановленными обновлениями или узлов с устаревшим программным обеспечением.

Блокирование и контроль доступа к определенным узлам Internet. Хотя системам обнаружения атак далеко до межсетевых экранов и систем контроля доступа к различным URL, например, WEBSweeper, они могут выполнять частичный контроль и блокирование доступа некоторых пользователей корпоративной сети к отдельным ресурсам Internet, например, к Web-серверам порнографического содержания. Это бывает необходимо тогда, когда в организации нет денег на приобретение межсетевого экрана и системы обнаружения атак, и функции МСЭ разносятся между системой обнаружения атак, маршрутизатором и прокси-сервером. Кроме того, системы обнаружения атак могут контролировать доступ сотрудников к серверам на основе ключевых слов. Например, sex, job, crack и т.д.

Контроль электронной почты. Системы обнаружения атак могут использоваться для контроля неблагонадежных сотрудников, использующих электронную почту для выполнения задач, не входящих в их функциональные обязанности, например, рассылка резюме. Некоторые системы могут обнаруживать вирусы в почтовых сообщениях и, хотя до настоящих антивирусных систем им далеко, они все же выполняют эту задачу достаточно эффективно.

Лучшее использование времени и опыта специалистов в области информационной безопасности заключается в обнаружении и устранении причин реализации атак, скорее, чем, в обнаружении самих атак. Устранив причины возникновения атак, т.е. обнаружив и устранив уязвимости, администратор тем самым устраняет и сам факт потенциальной реализации атак. Иначе атака будет повторяться раз за разом, постоянно требуя усилий и внимания администратора.

Существует большое число различных классификаций систем

обнаружения атак, однако самой распространенной является классификация по принципу реализации:

1. host-based, то есть обнаруживающие атаки, направленные на конкретный узел сети.
2. network-based, то есть обнаруживающие атаки, направленные на всю сеть или сегмент сети.

Системы обнаружения атак, контролирующие отдельный компьютер, как правило, собирают и анализируют информацию из журналов регистрации операционной системы и различных приложений (Web-сервер, СУБД и т.д.). По такому принципу функционирует RealSecure OS Sensor. Однако в последнее время стали получать распространение системы, тесно интегрированные с ядром ОС, тем самым, предоставляя более эффективный способ обнаружения нарушений политики безопасности. Причем такая интеграция может быть реализовано двояко. Во-первых, могут контролироваться все системные вызовы ОС (так работает Entercept) или весь входящий/исходящий сетевой трафик (так работает RealSecure Server Sensor). В последнем случае система обнаружения атак захватывает весь сетевой трафик напрямую с сетевой карты, минуя операционную систему, что позволяет уменьшить зависимость от нее и тем самым повысить защищенность системы обнаружения атак.

Системы обнаружения атак уровня сети собирают информацию из самой сети, то есть из сетевого трафика. Выполняться эти системы могут на обычных компьютерах (например, RealSecure Network Sensor), на специализированных компьютерах (например, RealSecure for Nokia или Cisco Secure IDS 4210 и 4230) или интегрированы в маршрутизаторы или коммутаторы (например, CiscoSecure IOS Integrated Software или Cisco Catalyst 6000 IDS Module). В первых двух случаях анализируемая информация собирается посредством захвата и анализа пакетов, используя сетевые интерфейсы в беспорядочном (promiscuous) режиме. В последнем случае захват трафика осуществляется с шины сетевого оборудования. Обнаружение атак требует выполнения одного

из двух условий - или понимания ожидаемого поведения контролируемого объекта системы или знания всех возможных атак и их модификаций. В первом случае используется технология обнаружения аномального поведения, а во втором случае - технология обнаружения злоумышленного поведения или злоупотреблений. Вторая технология заключается в описании атаки в виде шаблона или сигнатуры и поиска данного шаблона в контролируемом пространстве (например, сетевом трафике или журнале регистрации). Эта технология очень похожа на обнаружение вирусов (антивирусные системы являются ярким примером системы обнаружения атак), т.е. система может обнаружить все известные атаки, но она мало приспособлена для обнаружения новых, еще неизвестных, атак. Подход, реализованный в таких системах, очень прост и именно на нем основаны практически все предлагаемые сегодня на рынке системы обнаружения атак. Практически все системы обнаружения атак основаны на сигнатурном подходе.

Достоинства системы обнаружения атак:

1) Коммутация позволяет управлять крупномасштабными сетями, как несколькими небольшими сетевыми сегментами. В результате бывает трудно определить наилучшее место для установки системы, обнаруживающей атаки в сетевом трафике. Иногда могут помочь специальные порты (span ports) на коммутаторах, но не всегда. Обнаружение атак на уровне конкретного узла обеспечивает более эффективную работу в коммутируемых сетях, так как позволяет разместить системы обнаружения только на тех узлах, на которых это необходимо.

2) Системы сетевого уровня не требуют, чтобы на каждом хосте устанавливалось программное обеспечение системы обнаружения атак. Поскольку для контроля всей сети число мест, в которых установлены IDS невелико, то стоимость их эксплуатации в сети предприятия ниже, чем стоимость эксплуатации систем обнаружения атак на системном уровне. Кроме того, для контроля сетевого сегмента, необходим только один сенсор, независимо от числа узлов в данном сегменте.

3) Сетевой пакет, будучи ушедшим с компьютера злоумышленника, уже не может быть возвращен назад. Системы, функционирующие на сетевом уровне, используют "живой" трафик при обнаружении атак в реальном масштабе времени. Таким образом, злоумышленник не может удалить следы своей несанкционированной деятельности. Анализируемые данные включают не только информацию о методе атаки, но и информацию, которая может помочь при идентификации злоумышленника и доказательстве в суде. Поскольку многие хакеры хорошо знакомы с механизмами системной регистрации, они знают, как манипулировать этими файлами для скрывания следов своей деятельности, снижая эффективность систем системного уровня, которым требуется эта информация для того, чтобы обнаружить атаку.

4) Системы, функционирующие на уровне сети, обнаруживают подозрительные события и атаки по мере того, как они происходят, и поэтому обеспечивают гораздо более быстрое уведомление и реагирование, чем системы, анализирующие журналы регистрации. Например, хакер, инициирующий сетевую атаку типа "отказ в обслуживании" на основе протокола ТСР, может быть остановлен системой обнаружения атак сетевого уровня, посылающей ТСР-пакет с установленным флагом Reset в заголовке для завершения соединения с атакующим узлом, прежде чем атака вызовет разрушения или повреждения атакуемого узла. Системы анализа журналов регистрации не распознают атаки до момента соответствующей записи в журнал и предпринимают ответные действия уже после того, как была сделана запись. К этому моменту наиболее важные системы или ресурсы уже могут быть скомпрометированы или нарушена работоспособность системы, запускающей систему обнаружения атак на уровне узла. Уведомление в реальном масштабе времени позволяет быстро среагировать в соответствии с предварительно определенными параметрами. Диапазон этих реакций изменяется от разрешения проникновения в режиме наблюдения для того, чтобы собрать информацию об атаке и атакующем, до немедленного завершения атаки.

И, наконец, системы обнаружения атак, функционирующие на сетевом уровне, не зависят от операционных систем, установленных в корпоративной сети, так как они оперируют сетевым трафиком, которым обмениваются все узлы в корпоративной сети. Системе обнаружения атак все равно, какая ОС сгенерировала тот или иной пакет, если он в соответствии со стандартами, поддерживаемыми системой обнаружения. Например, в сети могут работать ОС Windows 98, Windows NT, Windows 2000 и XP, Netware, Linux, MacOS, Solaris и т.д., но если они общаются между собой по протоколу IP, то любая из систем обнаружения атак, поддерживающая этот протокол, сможет обнаруживать атаки, направленные на эти ОС. Совместное применение систем обнаружения атак на уровне сети и уровне узла повысит защищенность вашей сети.

## **1.2. Сетевые системы обнаружения атак и межсетевые экраны**

Наиболее часто сетевые системы обнаружения атак пытаются заменить межсетевыми экранами, уповая на то, что последние обеспечивают очень высокий уровень защищенности. Однако не стоит забывать, что межсетевые экраны - это просто системы, основанные на правилах, которые разрешают или запрещают прохождение трафика через них. Даже межсетевые экраны, построенные по технологии "", не позволяют с уверенностью сказать, присутствует ли атака в контролируемом ими трафике или нет. Они могут сказать, соответствует ли трафик правилу или нет. Например, МСЭ сконфигурирован так, чтобы блокировать все соединения кроме TCP-соединений на 80 порту (то есть HTTP-трафик). Таким образом, любой трафик через 80-ый порт законен с точки зрения МСЭ. С другой стороны, система обнаружения атак также контролирует трафик, но ищет в нем признаки атаки. Ее мало заботит, для какого порта предназначен трафик. По умолчанию весь трафик для системы обнаружения атак подозрителен. То есть, несмотря на то, что система обнаружения атак работает с тем же источником данных, что и

МСЭ, то есть с сетевым трафиком, они выполняют дополняющие друг друга функции. Например, HTTP-запрос "GET ../../etc/passwd HTTP/1.0". Практически любой МСЭ разрешает прохождение данного запроса через себя. Однако система обнаружения атак легко обнаружит эту атаку и блокирует ее. Мало обнаружить атаку, - необходимо на нее соответствующим образом отреагировать. Именно варианты реагирования во многом определяют эффективность системы обнаружения атак. На сегодняшний день предлагаются следующие варианты реагирования:

Уведомление на консоль (включая резервную) системы обнаружения атак или на консоль интегрированной системы (например, межсетевого экрана).

Звуковое оповещение об атаке.

Генерация управляющих последовательностей SNMP для систем сетевого управления.

Генерация сообщения об атаке по электронной почте.

Дополнительные уведомления на пейджер или факс. Очень интересная, хотя и редко применяемая возможность. Оповещение об обнаружении несанкционированной деятельности посылается не администратору, а злоумышленнику. По мнению сторонников данного варианта реагирования, нарушитель, узнав, что его обнаружили, вынужден прекратить свои действия.

Обязательная регистрация обнаруживаемых событий. В качестве журнала регистрации могут выступать: текстовый файл, системный журнал (например, в системе Cisco Secure Integrated Software), текстовый файл специального формата (например, в системе Snort), локальная база данных MS Access, SQL-база данных (например, в системе RealSecure). Надо только учитывать, что объемы регистрируемой информации требуют, как правило, SQL-базу - MS SQL или Oracle.

Трассировка событий (event trace), т.е. запись их в той последовательности и с той скоростью, с которыми их реализовывал злоумышленник. Затем администратор в любое заданное время может

прокрутить (replay или playback) необходимую последовательность событий с заданной скоростью (в реальном режиме времени, с ускорением или замедлением), чтобы проанализировать деятельность злоумышленника. Это позволит понять его квалификацию, используемые средства атаки и т.д.

Прерывание действий атакующего, т.е. завершение соединения. Это можно сделать, как:

перехват соединения (session hijacking) и посылка пакета с установленным флагом RST обоим участникам сетевого соединения от имени каждого из них (в системе обнаружения атак, функционирующей на уровне сети);

блокировка учетной записи пользователя, осуществляющего атаку (в системе обнаружения атак на уровне узла). Такая блокировка может быть осуществлена либо на заданный промежуток времени, либо до тех пор, пока учетная запись не будет разблокирована администратором. В зависимости от привилегий, с которыми запущена система обнаружения атак, блокировка может действовать как в пределах самого компьютера, на который направлена атака, так и в пределах всего домена сети.

Реконфигурация сетевого оборудования или межсетевых экранов. В случае обнаружения атаки на маршрутизатор или межсетевой экран посылается команда на изменение списка контроля доступа. Впоследствии все попытки соединения с атакующего узла будут отвергаться. Как и блокировка учетной записи злоумышленника, изменение списка контроля доступа может быть осуществлено или на заданный интервал времени или до того момента, как изменение будет отменено администратором реконфигурируемого сетевого оборудования.

Блокирование сетевого трафика так, как это реализовано в межсетевых экранах. Этот вариант позволяет ограничить трафик, а также адресатов, которые могут получить доступ к ресурсам защищаемого компьютера, позволяя выполнять функции доступные в персональных межсетевых экранах.

## 2. ПРОГРАММНЫЕ СРЕДСТВА АНАЛИЗА ЗАЩИЩЕННОСТИ

Сетевые и информационные технологии меняются настолько быстро, что статичные защитные механизмы, к которым относятся и системы разграничения доступа, и межсетевые экраны, и системы аутентификации, сильно ограничены и во многих случаях не могут обеспечить эффективной защиты. Следовательно, необходимы динамические методы, позволяющие обнаруживать и предотвращать нарушения безопасности. Одной из технологий, которая может быть применена для обнаружения нарушений, которые не могут быть идентифицированы при помощи моделей контроля доступа, является технология обнаружения атак.

Средства защиты информации на основе методов построения систем обнаружения атак (СОА) принято условно делить на два класса:

1. СОА на уровне сети анализируют сетевой трафик. Принципиальное преимущество сетевых СОА в том, что они идентифицируют нападения прежде, чем они достигнут атакуемого узла. Эти системы проще для развертывания в крупных сетях, потому что они не требуют установки на различные платформы, используемые в организации, практически не снижают производительности сети.

2. СОА на уровне хоста анализирует регистрационные журналы операционной системы или приложений. Они были разработаны для работы под управлением конкретной операционной системы, что накладывает на них определенные ограничения. Используя знание того, как должна себя "вести" операционная система, средства, построенные с учетом этого подхода, иногда могут обнаружить вторжения, пропускаемые сетевыми СОА.

Однако зачастую это достигается дорогой ценой, потому что постоянная регистрация, необходимая для выполнения такого рода обнаружения, существенно снижает производительность защищаемого хоста. Такие системы сильно загружают процессор и требуют больших объемов дискового пространства для хранения журналов регистрации.



Системы, входящие в первый класс, анализируют сетевой трафик, используя, как правило, сигнатуры атак и анализ "на лету", в то время как системы второго класса проверяют регистрационные журналы операционной системы или приложения.

## **2.1. Метод анализа "на лету"**

Заключается в мониторинге сетевого трафика в реальном или близком к реальному времени и использовании соответствующих алгоритмов обнаружения. Часто используется механизм поиска в трафике определенных строк, которые могут характеризовать несанкционированную деятельность. К таким строкам, к примеру, можно отнести `/etc/passwd` (описывает путь к списку паролей ОС UNIX).

Анализ журналов регистрации - один из самых первых реализованных методов обнаружения атак. Он заключается в анализе журналов регистрации (`log`, `audit trail`), создаваемых операционной системой, прикладным программным обеспечением, маршрутизаторами и т. д. Записи журнала регистрации анализируются и интерпретируются системой обнаружения атак.

Уровень защищенности компьютерных систем от угроз безопасности определяется многими факторами. Одним из определяющих факторов является адекватность конфигурации системного и прикладного ПО, средств защиты информации и активного сетевого оборудования существующим рискам.

Для проведения активного аудита безопасности могут использоваться специализированные программные средства, выполняющие обследование АС с целью выявления уязвимых мест (наличия "дыр") для электронного вторжения, а также, обеспечивающие комплексную оценку степени защищенности от атак нарушителей. Специальные открытые и коммерческие средства анализа защищенности позволяют оперативно проверить десятки и сотни территориально разнесенных узлов сети. При этом они не только

выявляют большинство угроз и уязвимых мест информационной системы, но и позволяют выработать рекомендации администраторам безопасности по их устранению.

## **2.2. Методы автоматизации процессов**

Существуют два метода автоматизации процессов анализа, защищенности:

использование технологии интеллектуальных программных агентов;

активное тестирование механизмов защиты путем эмуляции действий злоумышленника по осуществлению попыток сетевого вторжения в АС.

В первом случае система защиты строится на архитектуре консоль/менеджер/ агент. На каждую из контролируемых систем устанавливается программный агент, который выполняет настройки ПО и проверяет их правильность, контролирует целостность файлов, своевременность установки пакетов программных коррекций, а также выполняет другие полезные задачи по контролю защищенности АС. Менеджеры являются центральными компонентами подобных систем. Они посылают управляющие команды всем агентам контролируемого ими домена и сохраняют все данные, полученные от агентов в центральной базе данных. Администратор управляет менеджерами при помощи графической консоли, позволяющей выбирать, настраивать и создавать политики безопасности, анализировать изменения состояния системы, осуществлять ранжирование уязвимостей и т. п. Все взаимодействия между агентами, менеджерами и управляющей консолью осуществляются по защищенному клиент-серверному протоколу. Для активного тестирования механизмов защиты путем эмуляции действий злоумышленника по осуществлению попыток сетевого вторжения в АС применяются сетевые сканеры, эмулирующие действия потенциальных нарушителей. В основе работы сетевых сканеров лежит база данных, содержащая описание известных уязвимостей ОС, МЭ, маршрутизаторов и

сетевых протоколов, а также алгоритмов осуществления попыток вторжения (сценариев атак). Например, сетевые сканеры Nessus и Symantec NetRecon являются достойными представителями данного класса программных средств анализа защищенности.

Таким образом, программные средства анализа защищенности условно можно разделить на два класса:

Первый класс, к которому принадлежат сетевые сканеры, иногда называют средствами анализа защищенности сетевого уровня.

Второй класс, к которому относятся все остальные рассмотренные здесь средства, иногда называют средствами анализа защищенности системного уровня.

Данные классы средств имеют свои достоинства и недостатки, а на практике взаимно дополняют друг друга.

Сетевые сканеры являются, пожалуй, наиболее доступными и широко используемыми средствами анализа защищенности. Основной принцип их функционирования заключается в эмуляции действий потенциального злоумышленника по осуществлению сетевых атак. Поиск уязвимостей путем имитации возможных атак является одним из наиболее эффективных способов анализа защищенности АС, который дополняет результаты анализа конфигурации по шаблонам, выполняемый локально с использованием шаблонов (списков проверки). Сканер является необходимым инструментом в арсенале любого администратора безопасности АС.

Современные сканеры способны обнаруживать сотни уязвимостей сетевых ресурсов, предоставляющих те или иные виды сетевых протоколов, они выполняют четыре основные задачи:

- идентификацию доступных сетевых ресурсов;
- идентификацию доступных сетевых служб;
- идентификацию имеющихся уязвимостей сетевых служб;
- выдачу рекомендаций по устранению уязвимостей.

В функциональность сетевого сканера не входит выдача рекомендаций

по использованию найденных уязвимостей для реализации атак на сетевые ресурсы. Возможности сканера по анализу уязвимостей ограничены той информацией, которую могут предоставить ему доступные сетевые службы.

Принцип работы сканера заключается в моделировании действий злоумышленника, производящего анализ сети при помощи стандартных сетевых утилит. При этом используются известные уязвимости сетевых служб, сетевых протоколов и ОС для осуществления удаленных атак на системные ресурсы и осуществляется документирование удачных попыток.

Число уязвимостей в базах данных современных сканеров медленно, но уверенно приближается к 1000. Одним из наиболее продвинутых коммерческих продуктов этого класса является сетевой сканер NetRecon компании Symantec, база данных которого содержит около 800 уязвимостей UNIX, Windows и NetWare систем и постоянно обновляется через Web.

Преимущества сетевых сканеров. Для функционирования сетевого сканера необходим только один компьютер, имеющий сетевой доступ к анализируемым системам, поэтому в отличие от продуктов, построенных на технологии программных агентов, нет необходимости устанавливать в каждой анализируемой системе своего агента (своего для каждой ОС). Кроме того, сканеры являются более простым, доступным, дешевым и, во многих случаях, более эффективным средством анализа защищенности.

К недостаткам сетевых сканеров можно отнести большие временные затраты, необходимые для сканирования всех сетевых компьютеров из одной системы, и создание большой нагрузки на сеть. Кроме того, в общем случае трудно отличить сеанс сканирования от действительных попыток осуществления атак. Сетевыми сканерами также с успехом пользуются злоумышленники.

Средства анализа защищенности системного уровня выполняют проверки конфигурационных параметров ОС и приложений "изнутри". Такого рода системы зачастую строятся с применением интеллектуальных программных агентов. Это обусловлено тем, что системы анализа

защищенности, построенные на интеллектуальных программных агентах, обладают следующими достоинствами:

- являются потенциально более мощным средством, чем сетевые сканеры;
- обычно способны выполнять более сложные проверки и анализировать параметры ПО, недоступные сетевым сканерам, поскольку действуют изнутри;
- анализ защищенности может планироваться по времени и выполняться одновременно на всех контролируемых компьютерах;

- не оказывают большого влияния на пропускную способность сети;

- осуществляют шифрование (защита данных путем использования криптографических методов и гарантия невозможности чтения информации без знания секретного ключа) результатов проверок при передаче данных по сети.

Примером развитого средства анализа защищенности системного уровня рассматриваемого типа является автоматизированная система управления безопасностью предприятия ESM компании Symantec и System Scanner компании ISS. Система ESM построена на архитектуре консоль/менеджер/агент.

На сегодняшний день практически ни одна компания, использующая Интернет в связи с ИБ, не обходится без применения технологии VPN - виртуальной частной сети.

В основе построения лежит следующая идея: если в глобальной сети есть два узла, которые хотят обменяться информацией, то для обеспечения конфиденциальности и целостности передаваемой по открытым каналам информации необходимо построить виртуальный туннель, доступ к которому должен быть затруднен всем возможным активным и пассивным внешним наблюдателям.

Под термином виртуальная частная сеть чаще всего понимается организация защищенных информационных потоков между объектами виртуальной сети, организованных через сети общего пользования.

При этом потоки данных локальной и общей сетей не должны влиять

друг на друга. Термин виртуальная указывает на то, что соединение между двумя узлами сети не является постоянным и существует только во время прохождения трафика по сети. Объектами виртуальной корпоративной сети могут выступать объединения локальных сетей и отдельных компьютеров.

Инфраструктура сети VPN моделируется на основе реальных каналов связи: выделенных линий - проводных линий, соединенных с провайдером, который обладает высокоскоростными магистральными каналами (оптоволоконными, спутниковыми, радиорелейными), объединенными в Интернет, или коммутируемых линий - обычных телефонных каналов. При этом реальная открытая сеть может служить основой для целого множества VPN, конечное число которых определяется пропускной способностью открытых каналов связи. Позволяют организовать прозрачное для пользователей соединение локальных сетей, сохраняя секретность и целостность передаваемой информации с помощью шифрования. При этом при передаче по Интернет шифруется не только данные пользователя, но и сетевая информация - сетевые адреса, номера портов и т.д. Технология виртуальных частных сетей позволяет использовать сети общего пользования для построения защищенных сетевых соединений.

Технология VPN выполняет две основные функции: шифрование данных для обеспечения безопасности сетевых соединений и туннелированные протоколы.

Под туннелирование понимают безопасную передачу данных через открытые сети при помощи безопасного логического соединения, позволяющего упаковывать данные одного протокола в пакеты другого.

Защищенные потоки (каналы) виртуальной частной сети могут быть созданы между VPN: -шлюзами сети, VPN-шлюзами и VPN-клиентами, а также между VPN-клиентами. Создание виртуальных защищенных каналов достигается за счет шифрования трафика и туннелирования протоколов между объектами VPN-сети. VPN-шлюз - сетевое устройство, установленное на границе сети, выполняющее функции образования защищенных VPN-каналов,

аутентификации и авторизации клиентов VPN-сети. VPN-шлюз располагается аналогично МЭ таким образом, чтобы через него проходил весь сетевой трафик организации. В большинстве случаев VPN-сеть для пользователей внутренней сети остается прозрачной и не требует установки специального программного обеспечения. VPN-клиент - программное обеспечение (иногда с аппаратным акселератором), устанавливаемое на компьютеры пользователей, осуществляющих подключение к сети VPN (через VPN-шлюзы). VPN-клиент выполняет функции передачи параметров аутентификации и шифрования/дешифрования трафика.

В большинстве случаев необходимо одновременно обеспечить функционирование двух каналов - Internet и VPN. При этом можно использовать или различные физические линии связи, или одну. Однако стоимость эксплуатации одного канала связи для доступа к сети Internet и поддержки VPN обходится значительно ниже.

### 3. ПРОГРАММНЫЕ СРЕДСТВА ОБНАРУЖЕНИЯ УГРОЗ

Процесс осуществления атаки на АС включает три этапа. Первый этап, подготовительный, заключается в поиске предпосылок для осуществления той или иной атаки. На этом этапе ищутся уязвимости, использование которых приводит к реализации атаки, т. е. ко второму этапу. На третьем этапе атака завершается, "замечаются" следы и т. д. При этом первый и третий этапы сами по себе могут являться атаками.

#### 3.1. Способы защиты от атак

Обнаруживать, блокировать и предотвращать атаки можно несколькими путями. Первый способ, и самый распространённый, - это обнаружение уже реализуемых атак. Данный способ функционирует на втором этапе осуществления атаки. Этот способ применяется в "классических" системах обнаружения атак:

- серверах аутентификации;
- системах разграничения доступа;
- межсетевых экранах и т. п.

Основным недостатком средств данного класса является то, что атаки могут быть реализованы повторно. Они также повторно обнаруживаются и блокируются. И так далее, до бесконечности.

Второй способ - предотвратить атаки ещё до их реализации. Осуществляется это путём поиска уязвимостей, которые могут быть использованы для реализации атаки.

И наконец, третий путь - обнаружение уже совершённых атак и предотвращение их повторного осуществления.

Таким образом, системы обнаружения атак могут быть классифицированы по этапам осуществления атаки.

1. Системы, функционирующие на первом этапе осуществления атаки и



позволяющие обнаружить уязвимости информационной системы, используемые нарушителем для реализации атаки. Средства этой категории называются системами анализа защищенности (security assessment systems) или сканерами безопасности (security scanners).

Системы анализа защищённости проводят всесторонние исследования систем с целью обнаружения уязвимостей. Результаты, полученные от средств анализа защищённости, представляют "мгновенный снимок" состояния защиты системы в данный момент времени. Несмотря на то что эти системы не могут обнаруживать атаку в процессе её развития, они могут определить возможность реализации атак.

Эти системы реализуют две стратегии. Первая стратегия - пассивная, реализуемая на уровне операционной системы, СУБД и приложений, при которой осуществляется анализ конфигурационных файлов и системного реестра на наличие неправильных параметров, файлов паролей на наличие легко угадываемых паролей, а также других системных объектов на нарушения политики безопасности. Вторая стратегия - активная, осуществляется в большинстве случаев на сетевом уровне. Она заключается в воспроизведении наиболее распространенных сценариев атак и анализе реакции системы на эти сценарии.

2. Системы, функционирующие на втором этапе осуществления атаки и позволяющие обнаружить атаки в процессе их реализации, т.е. в режиме реального (или близкого к реальному) времени. Именно эти средства и принято считать системами обнаружения атак в классическом понимании. Помимо этого, в последнее время выделяется новый класс средств обнаружения атак - обманные системы.

Обнаружение атак реализуется посредством анализа или журналов регистрации операционной системы и прикладного программного обеспечения, или сетевого трафика в реальном времени. Компоненты обнаружения атак, размещенные на узлах или сегментах сети, оценивают различные действия, в том числе и использующие известные уязвимости,

сравнивая контролируемое пространство (сетевой трафик или журналы регистрации) с известными шаблонами (сигнатурами) несанкционированных действий.

Обманные системы могут использовать следующие методы: сокрытие, камуфляж и дезинформацию. Ярким примером использования первого метода является сокрытие сетевой топологии при помощи межсетевого экрана. Примером камуфляжа можно назвать использование Unix-подобного графического интерфейса в системе, функционирующей под управлением операционной системы Windows NT. Если злоумышленник случайно увидел такой интерфейс, то он будет пытаться реализовать атаки, характерные для ОС Unix, а не для ОС Windows NT. Это существенно увеличит время, необходимое для "успешной" реализации атаки. И наконец, в качестве примера дезинформации можно назвать использование заголовков, которые бы давали понять злоумышленнику, что атакуемая им система уязвима.

Системы, реализующие камуфляж и дезинформацию, эмулируют те или иные известные уязвимости, которых в реальности не существует.

Использование таких систем приводит к следующему:

- Увеличение числа выполняемых нарушителем операций и действий.

Так как невозможно заранее определить, является ли обнаруженная нарушителем уязвимость истинной или нет, злоумышленнику приходится выполнять много дополнительных действий, чтобы выяснить это. И даже дополнительные действия не всегда помогают. Например, попытка запустить программу подбора паролей на сфальсифицированный и несуществующий в реальности файл приведёт к бесполезной трате времени без какого-либо видимого результата. Нападающий будет думать, что он не смог подобрать пароли, в то время как на самом деле программа "взлома" была просто обманута.

- Получение возможности отследить нападающих. За тот период времени, когда нападающие пытаются проверить все обнаруженные уязвимости, в том числе и фиктивные, администраторы безопасности могут

проследить весь путь до нарушителя или нарушителей и предпринять соответствующие меры.

1. Системы, функционирующие на третьем этапе осуществления атаки и позволяющие обнаружить уже совершённые атаки. Эти системы делятся на два класса - системы контроля целостности, обнаруживающие изменения контролируемых ресурсов, и системы анализа журналов регистрации.

Системы контроля целостности работают по замкнутому циклу, обрабатывая файлы, системные объекты и атрибуты системных объектов с целью получения контрольных сумм; затем они сравнивают их с предыдущими контрольными суммами, отыскивая изменения. Когда изменение обнаружено, система посылает сообщение администратору, фиксируя вероятное время изменения.

Существует ещё одна распространённая классификация систем обнаружения нарушения политики безопасности - по принципу реализации: host-based, т.е. обнаруживающие атаки, направленные на конкретный узел сети, и network-based, направленные на всю сеть или сегмент сети. Существуют три основных вида систем обнаружения атак на уровне узла.

1. Системы, обнаруживающие атаки на конкретные приложения.
2. Системы, обнаруживающие атаки на операционные системы.
3. Системы, обнаруживающие атаки на системы управления базами данных (СУБД).
4. Внедрение программных средств обнаружения атак для информационной системы предприятия

Вопросы реализации и обеспечения ИБ прямо входят в сферу ответственности руководителя ИТ-департамента (если компания крупная) или ИТ-отдела или ИТ-службы. Экономия на информационной безопасности может выражаться в различных формах, крайними из которых являются: принятие только самых общих организационных мер обеспечения безопасности информации в ИС, использование только простых дополнительных средств защиты информации (СЗИ). В первом случае, как

правило, разрабатываются многочисленные инструкции, приказы и положения, призванные в критическую минуту переложить ответственность с людей, издающих эти документы, на конкретных исполнителей. Естественно, что требования таких документов (при отсутствии соответствующей технической поддержки) затрудняют повседневную деятельность сотрудников организации и, как показывает опыт, не выполняются. Во втором случае приобретаются и устанавливаются дополнительные средства защиты.

Разрабатываемая технология информационной безопасности должна обеспечивать: дифференцированный подход к защите различных АРМ и подсистем (уровень защищенности должен определяться с позиций разумной достаточности с учетом важности обрабатываемой информации и решаемых задач); максимальную унификацию средств защиты информации с одинаковыми требованиями к безопасности; реализацию разрешительной системы доступа к ресурсам ИС; минимизацию, формализацию (в идеале - автоматизацию) реальной выполнимости рутинных операций и согласованность действий различных подразделений по реализации требований разработанных положений и инструкций, не создавая больших неудобств при решении сотрудниками своих основных задач; учет динамики развития автоматизированной системы, регламентацию не только стационарного процесса эксплуатации защищенных подсистем, но и процессов их модернизации, связанных с многочисленными изменениями аппаратно-программной конфигурации АРМ; минимизацию необходимого числа специалистов отдела, занимающихся защитой информации. Надо совершенно четко понимать, что соблюдение необходимых требований по защите информации, препятствующих осуществлению несанкционированных изменений в системе, неизбежно приводит к усложнению процедуры правомочной модификации ИС. В этом состоит одно из наиболее остро проявляющихся противоречий между обеспечением безопасности и развитием и совершенствованием автоматизированной системы. Технология обеспечения информационной безопасности должна быть достаточно гибкой

и предусматривать особые случаи экстренного внесения изменений в программно-аппаратные средства защищаемой ИС.

В зависимости от масштаба компании можно выделить три основных класса сетей: IECO (International Enterprise Central Office) - центральная сеть международной распределенной компании, которая может насчитывать сотни и тысячи узлов; ROBO (Regional Office / Branch Office) - сеть регионального филиала, насчитывающего несколько десятков или сотен узлов; SOHO (Small Office / Home Office), - сети небольших филиалов или домашние (мобильные) компьютеры, подключаемые к центральной сети. Можно также выделить три основных сценария обеспечения информационной безопасности для этих классов сетей, различающихся различными требованиями по обеспечению защиты информации.

### **3.2. Сценарии защищенности**

При первом сценарии минимальный уровень защищенности обеспечивается за счет возможностей, встроенных в сетевое оборудование, которое установлено на периметре сети (например, в маршрутизаторах). В зависимости от масштабов защищаемой сети эти возможности (защита от подмены адресов, минимальная фильтрация трафика, доступ к оборудованию по паролю и т. д.) реализуются в магистральных маршрутизаторах - например, Cisco 7500 или Nortel BCN, маршрутизаторах региональных подразделений - например, Cisco 2500 или Nortel ASN, и маршрутизаторах удаленного доступа - например, Cisco 1600 или 3ComOfficeConnect. Больших дополнительных финансовых затрат этот сценарий не требует.

Второй сценарий, обеспечивающий средний уровень защищенности, реализуется уже при помощи дополнительно приобретенных средств защиты, к которым могут быть отнесены несложные межсетевые экраны, системы обнаружения атак и т. п. В центральной сети может быть установлен межсетевой экран (например, CheckPoint Firewall-1), на маршрутизаторах

могут быть настроены простейшие защитные функции, обеспечивающие первую линию обороны (списки контроля доступа и обнаружение некоторых атак), весь входящий трафик проверяется на наличие вирусов и т. д. Региональные офисы могут защищаться более простыми моделями межсетевых экранов. При отсутствии в регионах квалифицированных специалистов рекомендуется устанавливать программно-аппаратные комплексы, управляемые централизованно и не требующие сложной процедуры ввода в эксплуатацию (например, CheckPoint VPN-1 Appliance на базе Nokia IP330).

Третий сценарий, позволяющий достичь максимального уровня защищенности, предназначен для серверов e-Commerce, Internet-банков и т. д. В этом сценарии применяются высокоэффективные и многофункциональные межсетевые экраны, серверы аутентификации, системы обнаружения атак и системы анализа защищенности. Для защиты центрального офиса могут быть применены кластерные комплексы межсетевых экранов, обеспечивающих отказоустойчивость и высокую доступность сетевых ресурсов (например, CheckPoint VPN-1 Appliance на базе Nokia IP650 или CheckPoint VPN-1 с High Availability Module). Также в кластер могут быть установлены системы обнаружения атак (например, RealSecure Appliance).

Для обнаружения уязвимых мест, которые могут быть использованы для реализации атак, могут быть применены системы анализа защищенности (например, семейство SAFE - suite компании Internet Security Systems). Аутентификация внешних и внутренних пользователей осуществляется при помощи серверов аутентификации (например, CiscoSecure ACS). Ну и, наконец, доступ домашних (мобильных) пользователей к ресурсам центральной и региональных сетей обеспечивается по защищенному VPN-соединению. Виртуальные частные сети (Virtual Private Network - VPN) также используются для обеспечения защищенного взаимодействия центрального и региональных офисов. Функции VPN могут быть реализованы как при помощи межсетевых экранов (например, CheckPoint VPN-1), так и при

помощи специальных средств построения VPN. Авторизованное обучение и поддержка помогут быстро ввести систему защиты в эксплуатацию и настроить ее на технологию обработки информации, принятую в организации. Примерная стоимость обновления составляет около 15-20% стоимости программного обеспечения. Стоимость годовой поддержки со стороны производителя, которая, как правило, уже включает в себя обновление ПО, составляет около 20-30% стоимости системы защиты. Таким образом, каждый год нужно тратить не менее 20-30% стоимости ПО на продление технической поддержки средств защиты информации. Стандартный набор средств комплексной защиты информации в составе современной ИС обычно содержит следующие компоненты:

1. средства обеспечения надежного хранения информации с использованием технологии защиты на файловом уровне (FileEncryption System - FES);
2. средства авторизации и разграничения доступа к информационным ресурсам, а также защиту от несанкционированного доступа к информации с использованием систем биометрической авторизации и технологии токенов (смарт-карты, touch-memory, ключи для USB-портов и т.п.);
3. средства защиты от внешних угроз при подключении к общедоступным сетям связи (Internet), а также средства управления доступом из Internet с использованием технологии межсетевых экранов (Firewall) и содержательной фильтрации (Content Inspection);
4. средства защиты от вирусов с использованием специализированных комплексов антивирусной профилактики;
5. средства обеспечения конфиденциальности, целостности, доступности и подлинности информации, передаваемой по открытым каналам связи с использованием технологии защищенных виртуальных частных сетей (VPN);

6. средства обеспечения активного исследования защищенности информационных ресурсов с использованием технологии обнаружения атак (Intrusion Detection);

7. средства обеспечения централизованного управления системой информационной безопасности в соответствии с согласованной и утвержденной "Политикой безопасности компании".

В зависимости от масштаба деятельности компании методы и средства обеспечения ИБ могут различаться, но любой квалифицированный СЮ или специалист IT-службы скажет, что любая проблема в области ИБ не решается односторонне - всегда требуется комплексный, интегральный подход.



## 4. ОЗНАКОМЛЕНИЕ С WIRESHARK

### 4.1. Функциональные блоки Wireshark

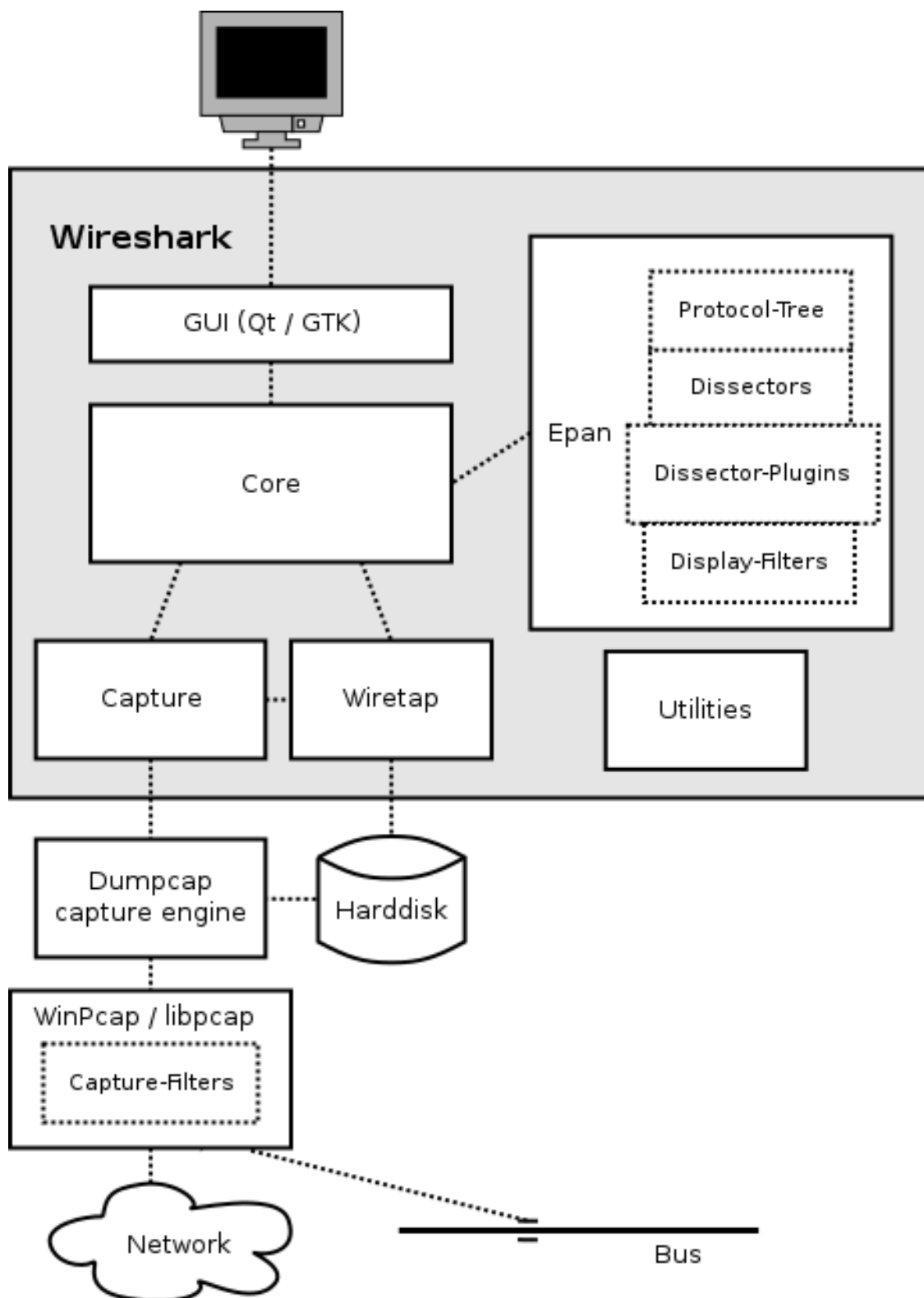


Рисунок 1. Функциональные блоки

## **Функциональные блоки более подробно:**

### **GUI**

Обработка всего пользовательского ввода / вывода (все окна, диалоги и тому подобное). Исходный код можно найти в каталоге `ui / qt`.

### **Core**

Основной «клеевой код», который скрепляет остальные блоки. Исходный код можно найти в корневом каталоге.

### **Epan**

Enhanced Packet ANalyzer - механизм анализа пакетов. Исходный код можно найти в каталоге `epan`. Epan предоставляет следующие API:

Дерево протокола. Информация о рассечении для отдельного пакета.

Диссекторы. Различный протокол диссекторов в эпане / диссекторов.

Плагины Dissector - Поддержка реализации диссекторов в виде отдельных модулей. Исходный код можно найти в плагинах.

Фильтры дисплея - двигатель фильтра дисплея на `epan / dfilter`.

### **Wiretap**

Библиотека прослушивания используется для чтения и записи файлов захвата в `libpcap`, `pcapng` и многих других форматах файлов. Исходный код находится в каталоге прослушивания.

### **Capture**

Интерфейс для механизма захвата. Исходный код находится в корневом каталоге.

### **Dumpcap**

Сам двигатель захвата. Это единственная часть, которая выполняется с повышенными привилегиями. Исходный код находится в корневом каталоге.

### **Npcap and libpcap**

Это внешние библиотеки, которые обеспечивают захват пакетов и поддержку фильтрации на разных платформах. Фильтрация в Npcap и libpcap работает на гораздо более низком уровне, чем фильтры отображения

Wireshark, и использует существенно другой механизм. Вот почему существуют разные синтаксисы фильтра отображения и захвата

## **4.2. Захват пакетов**

Захват принимает пакеты от сетевого адаптера и сохраняет их в файл на жестком диске.

Поскольку доступ к сетевому адаптеру требует повышенных привилегий, эти функции изолированы от программы `dumpcap`. Размещение функции захвата в `dumpcap` позволяет запускать остальную часть кода (анализаторы, пользовательский интерфейс и т. д.) С обычными привилегиями пользователя. Чтобы скрыть все низкоуровневые машинно-зависимые детали от Wireshark, используются библиотеки `libpcap` и `Npcap` («`libpcap` или `Npcap` (необязательно, но настоятельно рекомендуется)»). Эти библиотеки предоставляют интерфейс общего назначения для захвата пакетов и используются широким спектром приложений.

## **4.3. Захват файлов**

Wireshark может читать и записывать файлы захвата в своих естественных форматах, `pcapng` и `pcap`, которые используются многими другими инструментами захвата сети, такими как `tcpdump`. Кроме того, Wireshark поддерживает чтение и запись файлов захвата пакетов в форматах, используемых другими инструментами захвата сети. Эта поддержка реализована в библиотеке прослушивания Wireshark, которая предоставляет универсальный интерфейс для чтения и записи форматов захвата пакетов и поддерживает более двадцати форматов захвата пакетов.

#### 4.4. Обработка пакета

Wireshark пакеты так называемым двухпроходным анализом.

Wireshark выполняет первый проход анализа всех пакетов, когда они загружаются из файла. Все пакеты рассекаются последовательно, и эта информация используется для заполнения панели списка пакетов Wireshark и для создания состояния и другой информации, необходимой при отображении пакета.

Позднее Wireshark выполняет специальные вскрытия «второго прохода» для пакетов, из которых ему нужны данные. Это позволяет Wireshark заполнять поля, требующие знаний в будущем, например, поля «response in frame #», и правильно вычислять зависимости кадров сборки.

Например, Wireshark будет выполнять специальное вскрытие, когда пользователь выбирает пакет (для отображения подробной информации о пакете), вычисляет статистику (таким образом, все значения вычисляются) или выполняет другое действие, требующее пакетных данных. Однако, поскольку Wireshark может анализировать только те пакеты, которые необходимы, нет гарантии, что Wireshark снова будет анализировать все пакеты, а также нет никакой гарантии в отношении порядка, в котором пакеты будут анализироваться после первого прохода.

## 5. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНЫХ ИССЛЕДОВАНИЙ

### 5.1. Сетевые утилиты

1. Откроем параметры своего сетевого интерфейса (см. Приложение 1)

#### **ARP**

Отображает и изменяет записи в кэше протокола ARP. Кэш ARP содержит одну или несколько таблиц, которые используются для хранения IP-адресов и разрешенных физических адресов Ethernet или Token Ring. Для каждого сетевого адаптера Ethernet или Token Ring, установленного на компьютере, существует отдельная таблица. При использовании без параметров в ARP отображаются справочные сведения.

Чтобы отобразить таблицы кэша ARP для всех интерфейсов, введите: -a

#### **Ipconfig**

Отображает все текущие значения конфигурации сети TCP/IP и обновляет параметры протокола DHCP и системы доменных имен (DNS). При использовании без параметров ipconfig отображает IP-адреса версии 4 (IPv4) и IPv6, маску подсети и шлюз по умолчанию для всех адаптеров.

2. Сравниваем статистику по протоколу tcp и со статистикой udp.

(см. Приложение 2)-**n** Отображает активные TCP-подключения, однако адреса и номера портов выражаются в числовом виде, и для определения имен не выполняется никаких попыток.

(см. Приложение 3)

```
netstat -s -p
```

данные параметры включают подробную информацию для tcp / udp

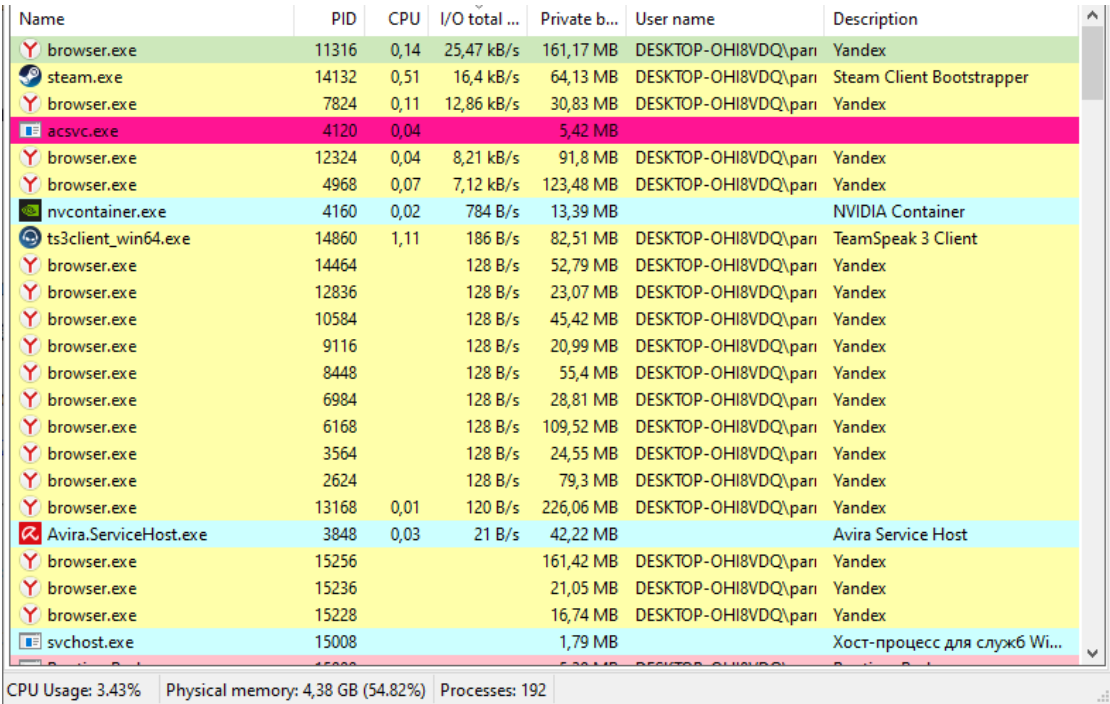
3. Посмотрим список сетевых служб на нашем компьютере

(см. Приложение 4)

**Arp -d \*** - очистка таблицы ARP

Мы научились использовать сетевые утилиты, как встроенные в систему так и поставляемые отдельно.

Скачаем программу Process Hacker, с официального сайта  
Эта программа является “улучшенным” диспетчером задач



Name	PID	CPU	I/O total ...	Private b...	User name	Description
browser.exe	11316	0,14	25,47 kB/s	161,17 MB	DESKTOP-OHI8VDQ\pari	Yandex
steam.exe	14132	0,51	16,4 kB/s	64,13 MB	DESKTOP-OHI8VDQ\pari	Steam Client Bootstrapper
browser.exe	7824	0,11	12,86 kB/s	30,83 MB	DESKTOP-OHI8VDQ\pari	Yandex
acsvic.exe	4120	0,04		5,42 MB		
browser.exe	12324	0,04	8,21 kB/s	91,8 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	4968	0,07	7,12 kB/s	123,48 MB	DESKTOP-OHI8VDQ\pari	Yandex
nvcontainer.exe	4160	0,02	784 B/s	13,39 MB		NVIDIA Container
ts3client_win64.exe	14860	1,11	186 B/s	82,51 MB	DESKTOP-OHI8VDQ\pari	TeamSpeak 3 Client
browser.exe	14464		128 B/s	52,79 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	12836		128 B/s	23,07 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	10584		128 B/s	45,42 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	9116		128 B/s	20,99 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	8448		128 B/s	55,4 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	6984		128 B/s	28,81 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	6168		128 B/s	109,52 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	3564		128 B/s	24,55 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	2624		128 B/s	79,3 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	13168	0,01	120 B/s	226,06 MB	DESKTOP-OHI8VDQ\pari	Yandex
Avira.ServiceHost.exe	3848	0,03	21 B/s	42,22 MB		Avira Service Host
browser.exe	15256			161,42 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	15236			21,05 MB	DESKTOP-OHI8VDQ\pari	Yandex
browser.exe	15228			16,74 MB	DESKTOP-OHI8VDQ\pari	Yandex
svchost.exe	15008			1,79 MB		Хост-процесс для служб Wi...

CPU Usage: 3.43%   Physical memory: 4,38 GB (54.82%)   Processes: 192

Рисунок 2. Диспетчером задач

В рисунке 2, я показал какой процесс у меня использует больше всех трафика

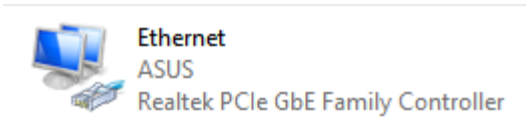
5.2. Сетевые sniffеры

Захват

...используя этот фильтр:

- Беспроводная сеть \_
- Подключение по локальной сети\* 11 \_
- Ethernet 5 \_
- Подключение по локальной сети\* 12 \_
- Подключение по локальной сети\* 8 \_
- Подключение по локальной сети\* 9 \_
- Ethernet ^
- Adapter for loopback traffic capture \_
- Подключение по локальной сети\* 7 \_

Тут вы выбираем сеть, в моем случае Ethernet, т.к мой компьютер подключен



через этот адаптер

(см. Приложение 5)

Нужно отсортировать наш трафик, с помощью фильтра, “http” в строке фильтров, для нахождения нужных нам пакет. Для это вводим в специальную строку “http”

(см. Приложение 6)

Через wireshark можно посмотреть подробную статистику, нажав дважды на один из пакетов

(см. Приложение 7)

И посмотреть какой язык использует сайт. В данном случае en-US

Здесь видно, что можно перехватить передачу информации, я логинился на сайт и через wireshark нашел пакет со своим логином и паролем, который отправлял на сайте

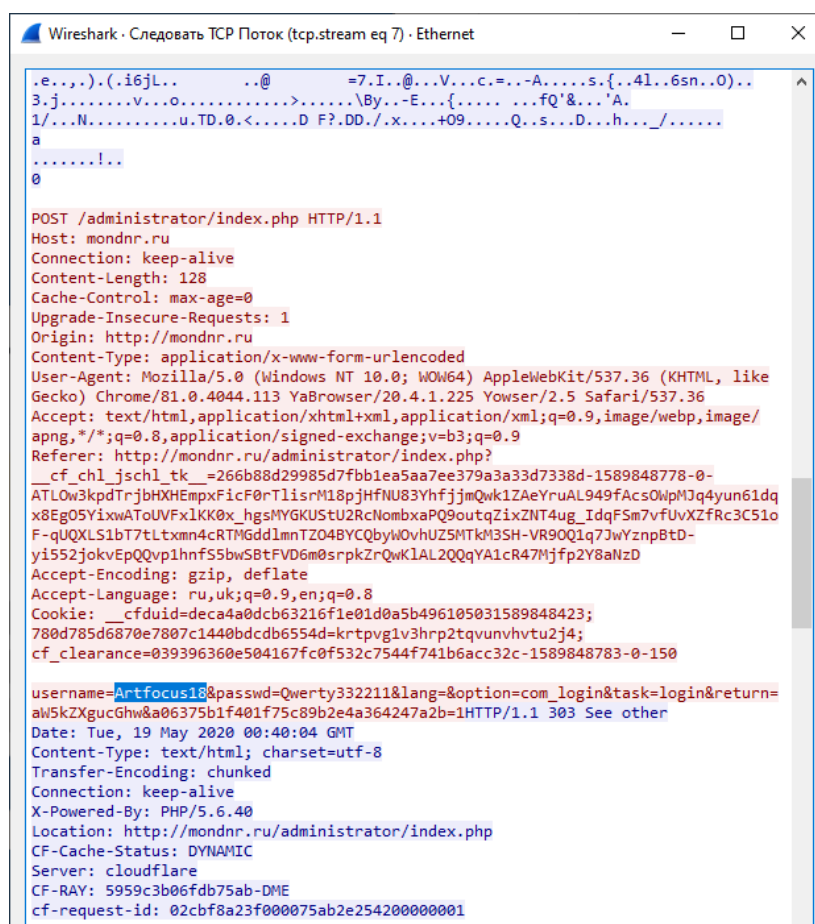


Рисунок 3. Отчет TCP потока

### 5.3. Сетевые сканеры

Необходимо провести подробный тест сети

Для этого выберем профиль - Intense scan. (Тестировал свою локальную сеть)  
(см. Приложение 8)

На скриншоте есть информация про мой роутер (его производитель ),  
показывает какие порты открыты и информацию про них  
(см. Приложение 9)

Quick scan показывает краткую информацию про мои порты

Подробная информация про параметры

**ОПЦИИ УПРАВЛЕНИЯ ВРЕМЕНЕМ И ПРОИЗВОДИТЕЛЬНОСТЬЮ:**

Опции, принимающие аргумент <время>, задаются в миллисекундах, пока вы не добавите 's' (секунды), 'm' (минуты), или 'h' (часы) к значению (напр. 30m).

-T[0-5]: Установить шаблон настроек управления временем (больше - быстрее)

--min-hostgroup/max-

Включить IPv6 сканирование

A: Активировать функции определения ОС и версии, сканирование использованием скриптов и трассировку

--datadir <имя\_директории>: Определяет место расположения файлов Nmap

--send-eth/--send-ip: Использовать сырой уровень Ethernet/IP

--privileged: Подразумевать, что у пользователя есть все привилегии  
Интенсивное, всестороннее сканирование.

Опция -A включает обнаружение ОС (-O), определение версии (-sV), сканирование скриптов (-sC) и трассировку (--traceroute). Без прав root запускаются только обнаружение версий и сканирование скриптов. Это считается навязчивым сканированием.

Дальше я решил проверить 139 порт на уязвимости, вот что мне выдал nmap  
(см. Приложение 10)



И вот скрипт на проверку уязвимостей (см. Приложение 11)

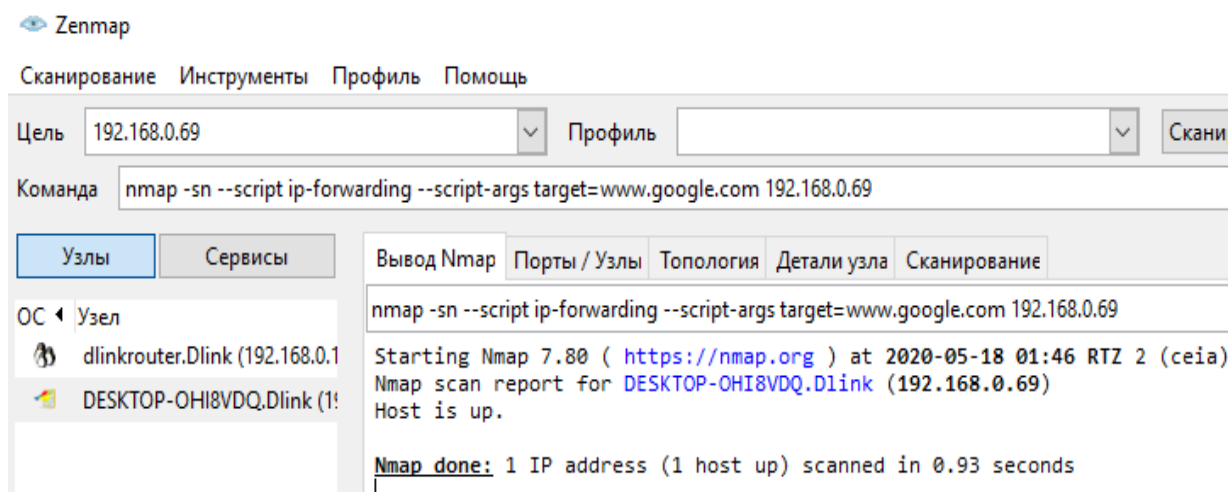


Рисунок 4. Nmap

Отправляю запрос со своего компьютера на сайт google.com

nmap полезная программа для подробного просмотра настроек сети. На сайте производителя указано что Профиль "Intense scan" опасен для маршрутизаторов, по всей видимости очень часто его использовать нельзя изза нагрузки

#### 5.4. Обнаружение sniffеров и сканеров в сети

Продолжаем работу с wireshark

Тут я хочу показать вам как можно на одном компьютере и виртуальной машине протестировать работу sniffеров и сканеров сети

Я использовал свою обычную систему windows 10 и Kali Linux

Возникли проблемы с настройкой виртуалки(Kali linux), в Параметрах машины-Сеть по стандарту стоит параметр NAT- это общая сеть.

То есть виртуальная машина, для использования интернета, обращается к моему устройству (не создавая своего). Для различных манипуляций и создания устройства в локальной сети нужно указать параметр Сетевой мост и задать отдельное устройство (WiFi адаптер у меня) (см. Приложение 12)

Я выбрал Kali Linux, т.к все утилиты которые нам понадобятся уже есть в самой системе и ничего настраивать не нужно.

Мы зашли в Kali и запустили там программу nmap, провели скан, моей основной системы, с профилем Intense scan. А в Windows, в это время, я включить Wireshark, которым “прослушивал” сеть. Получил такие результаты(см. Приложение 13)

Провел сканирования IP-сетей второго компьютера

Видно, что была куча запросов на мою сеть. Проверка разных портов и поиск любой информации (см. Приложение 14)

Но т.к человек не способен за всем уследить я использовал script, написанный на python

Файл который я сохранил на windows wireshark, не заметил ничего подозрительного

А вот тот который я взял из linux, script обнаружил угрозу и флаг SYN-ACKs

```
root@kali:/media/sf_# python detector.py testwin.pcap
Analyzed 20357 packets:
no suspicious packets detected...
root@kali:/media/sf_# python detector.py testlinux.pcap.pcap
Cannot open file: testlinux.pcap.pcap
root@kali:/media/sf_# python detector.py testlinux.pcap
Analyzed 2067 packets:
192.168.0.69 had 1002 SYNs and 0 SYN-ACKs
```

Показываю что человек также способен увидеть угрозу (см. Приложение 15)

Теперь мы зайдем в раздел “Информация эксперта”(см. Приложение 16)

Я провел запросы nmap quick scan plus и получил результат в wireshark и в режиме эксперта(см. Приложение 17)

В данном случае рассмотрена статистика NMAP сканирования, по которой ясно, что если в сети мощного Warning и Error флагов, то происходит что то нетипичное (но это не является однозначным критерием того, что сеть сканируют).

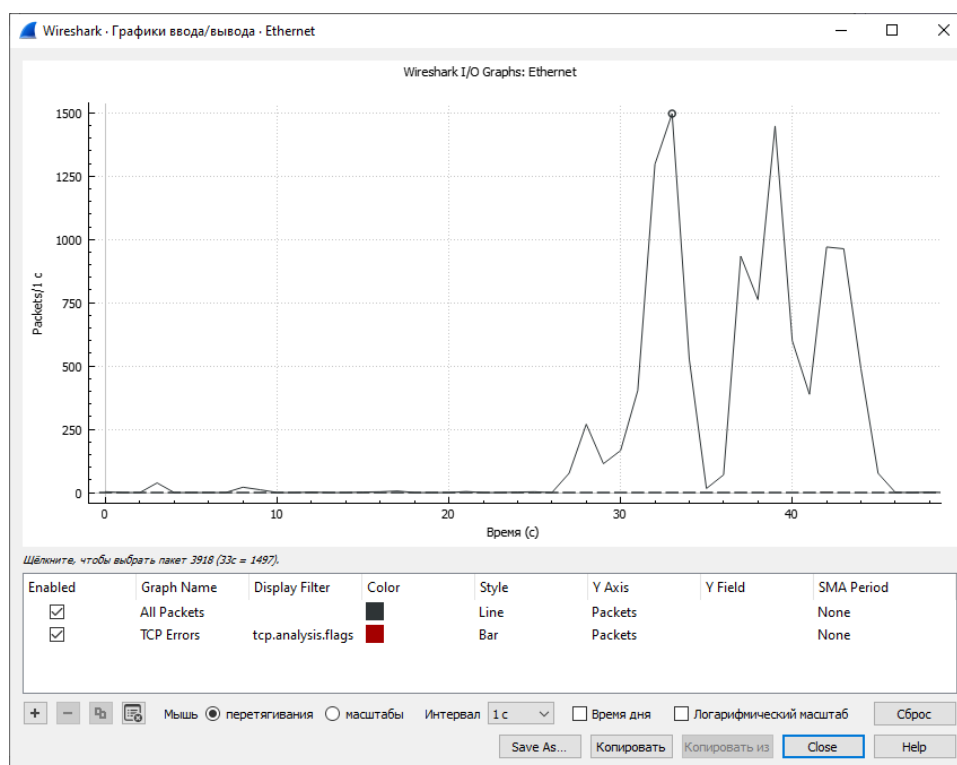


Рисунок 5. График ввода/вывода

Обычная активность(зашел на пару сайтов)

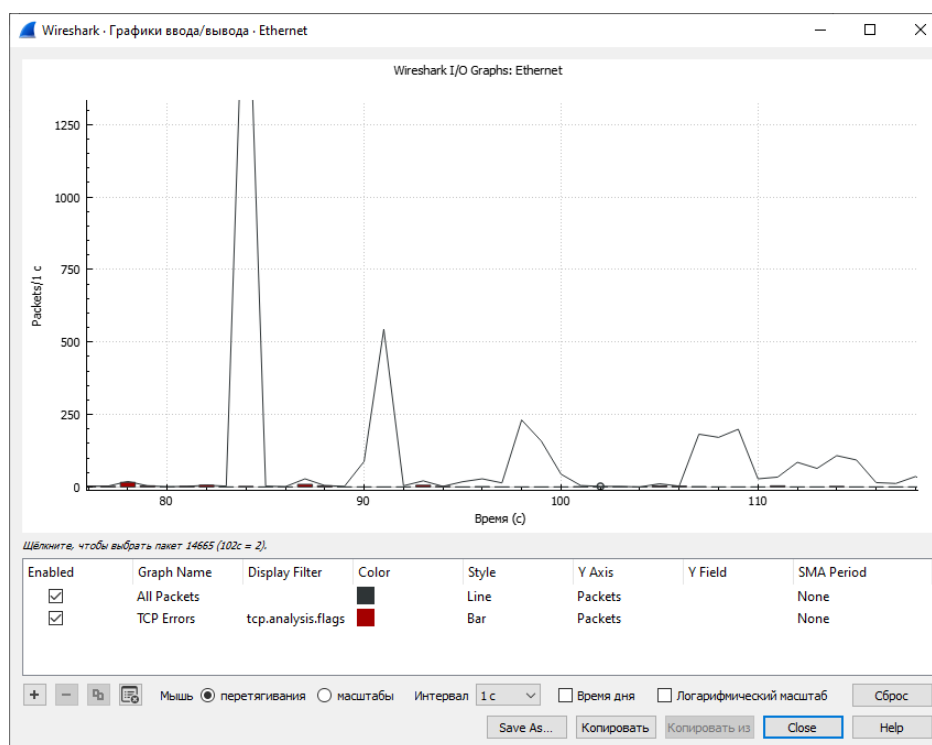


Рисунок 6. График ввода/вывода

А это при запуске nmap

Можно заметить, что в случае нормального поведения количество данных по отношению к пакетам, переданным за секунду несколько разнится. Но опять-таки это не стопроцентный показатель того, что сеть подверглась сканированию

Сканирование можно определять и в режиме реального времени. Есть такие способы:

1. Мониторить Wireshark (но это делается человеком и высока вероятность пропуска важной информации).
2. Использовать специальные скрипты снифферы-анализаторы.
3. Использовать IDS (системы обнаружения вторжений), например Snort, в котором выставлен флаг обнаружения сканирования в конфигурационном файле.

## ЗАКЛЮЧЕНИЕ

Не будь уязвимостей в компонентах информационных систем, нельзя было бы реализовать многие атаки и, следовательно, традиционные системы защиты вполне эффективно справлялись бы с возможными атаками. Но программы пишутся людьми, которым свойственно делать ошибки. Вследствие чего и появляются уязвимости, которые используются злоумышленниками для реализации атак. Если бы все атаки строились по модели "один к одному", то с некоторой натяжкой, но межсетевые экраны и другие защитные системы смогли бы противостоять и им. Но появились скоординированные атаки, против которых традиционные средства уже не так эффективны. Поэтому появляются новые технологии - технологии обнаружения атак. Приведенная систематизация данных об атаках и этапах их реализации дает необходимый базис для понимания технологий обнаружения атак.

Система обнаружения атак - это всего лишь необходимое, но явно недостаточное условие для обеспечения эффективной системы защиты организации. Необходимо провести целый спектр организационных и технических мероприятий для построения целостной системы защиты организации, это: анализ рисков, разработка политики безопасности, установка, настройка различных средств защиты, обучение специалистов, и т.д. Эффективная и надежная система обнаружения атак позволяет собирать, обобщать и анализировать информацию от множества удаленных сенсоров на центральной консоли. Она позволяет сохранять эту информацию для более позднего анализа, и предоставляет средства для проведения такого анализа. Эта система постоянно контролирует все установленные модули слежения и мгновенно реагирует в случае возникновения тревоги. Использование всех этих компонентов в комплексе образует реальную и эффективную систему обнаружения атак.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Комплексный подход к построению интеллектуальной системы обнаружения атак / Васильев В. И., Свечников Л. А., Кашаев Т. Р. // Системы управления и информационные технологии, Воронеж, №2, 2007. - С. 76-82.
2. Проблема построения защищенных Internet-серверов / Свечников Л.А., Кашаев Т.Р. // Интеллектуальные системы управления и обработки информации: Материалы Всероссийской молодежной науч.-технич. конференции. Уфа: УГАТУ, 2003. - С. 9.
3. Использование AD для разработки защищенных приложений / Свечников Л.А., Кашаев Т.Р., Кустов Г.А. // Интеллектуальные системы управления и обработки информации: Материалы Всероссийской молодежной науч.-технич. конференции. Уфа: УГАТУ, 2003. -С. 21.
4. Структура интеллектуальной распределенной системы обнаружения атак / Васильев В.И., Свечников Л.А., Калабухов М.С. // Компьютерные науки и информационные технологии: Труды 7-й Международной конференции (CSIT'2005), Т. 2, Уфа: Изд-во УГАТУ, 2005. - С. 200-206 (на англ. языке).
5. Архитектура распределенной системы обнаружения атак / Васильев В.И., Свечников Л.А., // Информационная безопасность: Материалы 8-й международной научно-практической конференции. Часть 1 - Таганрог: Изд-во ТРТУ, 2006. - С. 180-184.

ПРИЛОЖЕНИЕ 1

ARP -a

```
Microsoft Windows [Version 10.0.18363.836]
(c) Корпорация Майкрософт (Microsoft Corporation), 2019. Все права за...

C:\Users\parm2>Arp -a

Интерфейс: 192.168.56.1 --- 0x15
  адрес в Интернете      Физический адрес      Тип
192.168.56.255           ff-ff-ff-ff-ff-ff     статический
224.0.0.2                01-00-5e-00-00-02     статический
224.0.0.22               01-00-5e-00-00-16     статический
224.0.0.251              01-00-5e-00-00-fb     статический
224.0.0.252              01-00-5e-00-00-fc     статический
255.255.255.255          ff-ff-ff-ff-ff-ff     статический

Интерфейс: 192.168.0.69 --- 0x16
  адрес в Интернете      Физический адрес      Тип
192.168.0.1              28-3b-82-47-49-08     динамический
192.168.0.255            ff-ff-ff-ff-ff-ff     статический
224.0.0.2                01-00-5e-00-00-02     статический
224.0.0.22               01-00-5e-00-00-16     статический
224.0.0.251              01-00-5e-00-00-fb     статический
224.0.0.252              01-00-5e-00-00-fc     статический
239.255.255.250          01-00-5e-7f-ff-fa     статический
255.255.255.255          ff-ff-ff-ff-ff-ff     статический

C:\Users\parm2>ipconfig

Настройка протокола IP для Windows

Адаптер Ethernet Ethernet 4:

    DNS-суффикс подключения . . . . . :
    Локальный IPv6-адрес канала . . . : fe80::800a:ded4:eac6:fcd0%21
    IPv4-адрес. . . . . : 192.168.56.1
    Маска подсети . . . . . : 255.255.255.0
    Основной шлюз. . . . . :

Адаптер Ethernet Ethernet:

    DNS-суффикс подключения . . . . . : Dlink
    IPv6-адрес. . . . . : fd01::a965:edff:1d2d:372e
    Временный IPv6-адрес. . . . . : fd01::d8f3:185f:269c:3350
    Локальный IPv6-адрес канала . . . : fe80::a965:edff:1d2d:372e%22
    IPv4-адрес. . . . . : 192.168.0.69
    Маска подсети . . . . . : 255.255.255.0
    Основной шлюз. . . . . : 192.168.0.1

Адаптер Ethernet Ethernet 2:

    Состояние среды. . . . . : Среда передачи недоступна.
    DNS-суффикс подключения . . . . . :
```

ПРИЛОЖЕНИЕ 2

Netstat -n

```
C:\Users\parm2>netstat -n

Активные подключения

Имя      Локальный адрес      Внешний адрес      Состояние
TCP      127.0.0.1:56640      127.0.0.1:65001     ESTABLISHED
TCP      127.0.0.1:65001      127.0.0.1:56640     ESTABLISHED
TCP      192.168.0.69:56630   88.221.132.10:80     CLOSE_WAIT
TCP      192.168.0.69:56631   88.221.132.10:80     CLOSE_WAIT
TCP      192.168.0.69:56643   51.105.249.223:443   ESTABLISHED
TCP      192.168.0.69:56708   23.54.61.228:443     CLOSE_WAIT
TCP      192.168.0.69:56709   23.54.61.228:443     CLOSE_WAIT
TCP      192.168.0.69:56710   23.54.61.228:443     CLOSE_WAIT
TCP      192.168.0.69:56711   23.54.61.228:443     CLOSE_WAIT
TCP      192.168.0.69:56712   23.54.61.228:443     CLOSE_WAIT
TCP      192.168.0.69:56713   23.54.61.228:443     CLOSE_WAIT
TCP      192.168.0.69:56717   72.247.174.86:80     CLOSE_WAIT
TCP      192.168.0.69:56718   72.247.174.86:80     CLOSE_WAIT
TCP      192.168.0.69:56719   72.247.174.86:80     CLOSE_WAIT
TCP      192.168.0.69:56720   72.247.174.86:80     CLOSE_WAIT
TCP      192.168.0.69:56721   72.247.174.86:80     CLOSE_WAIT
TCP      192.168.0.69:56722   72.247.174.86:80     CLOSE_WAIT
TCP      192.168.0.69:56731   23.54.61.228:443     CLOSE_WAIT
TCP      192.168.0.69:56772   72.247.172.15:443    CLOSE_WAIT
TCP      192.168.0.69:56786   80.239.142.165:443   ESTABLISHED
TCP      192.168.0.69:56795   213.180.204.179:443  ESTABLISHED
TCP      192.168.0.69:56803   77.88.55.60:443      ESTABLISHED
TCP      192.168.0.69:56823   87.240.129.186:443   ESTABLISHED
TCP      192.168.0.69:56830   64.233.162.188:5228  ESTABLISHED
TCP      192.168.0.69:56834   213.180.204.179:443  ESTABLISHED
```

ПРИЛОЖЕНИЕ 3

Статистика сети

Статистика IPv4

Получено пакетов

= 200800

Получено ошибок в заголовках

= 0

Получено ошибок в адресах

= 6

Направлено датаграмм

= 0

Получено неизвестных протоколов

= 0

Отброшено полученных пакетов

= 5518

Доставлено полученных пакетов

= 205428

Запросов на вывод

= 162038

Отброшено маршрутов

= 0

Отброшено выходных пакетов

= 1601

Выходных пакетов без маршрута

= 0

Требуется сборка

= 0

Успешная сборка

= 0

Сбоев при сборке

= 0

Успешно фрагментировано датаграмм

= 0

Сбоев при фрагментации датаграмм

= 0

Создано фрагментов

= 0

Статистика IPv6

Получено пакетов

= 2914

Получено ошибок в заголовках

= 0

Получено ошибок в адресах

= 0

Направлено датаграмм

= 0

Получено неизвестных протоколов

= 0

Отброшено полученных пакетов

= 701

Доставлено полученных пакетов

= 3723

Запросов на вывод

= 4572

Отброшено маршрутов

= 0

Отброшено выходных пакетов

= 0

Выходных пакетов без маршрута

= 0

Требуется сборка

= 0

Успешная сборка

= 0

Сбоев при сборке

= 0

Успешно фрагментировано датаграмм

= 0

Сбоев при фрагментации датаграмм

= 0

Создано фрагментов

= 0

Статистика ICMPv4

Сообщений

2257

2684

Ошибок

0

0

'Назначение недостижимо'

2249

2684

Превышений времени

0

0

Ошибок в параметрах

0

0

Просьба "снизить скорость"

0

0

Переадресовано

0

0

Ответных пакетов

8

0

Эхо-сообщений

0

0

Отметок времени

0

0

Ответы на отметки времени

0

0

Масок адресов

0

0

Ответов на маски адресов

0

0

Маршрутизатор

0

0

Маршрутизатор

0

0

ICMPv6 Статистика

Сообщений

618

539

Ошибок

0

0

'Назначение недостижимо'

8

150

Пакет слишком велик

0

0

Превышений времени

0

0

Ошибок в параметрах

0

0

Эхо-сообщений

0

0

Ответных пакетов

0

0

MLD-запросы

0

0

MLD-отчеты

0

0

MLD выполнено

0

0

Маршрутизатор

0

12

Маршрутизатор

252

0

Окружение

182

187

Окружение

176

190

Переадресовано

0

0

Перенумер. маршрутизатора

0

0

Статистика TCP для IPv4

Активных открыто

= 1880

Пассивных открыто

= 9

Сбоев при подключении

= 36

Сброшено подключений

= 320

Текущих подключений

= 35

Получено сегментов

= 187056

Отправлено сегментов

= 144503

Повторно отправлено сегментов

= 168

Статистика TCP для IPv6

Активных открыто

= 18

Пассивных открыто

= 0

Сбоев при подключении

= 16

Сброшено подключений

= 0

Текущих подключений

= 0

Получено сегментов

= 32

Отправлено сегментов

= 38

Повторно отправлено сегментов

= 4

Статистика UDP для IPv4

Получено датаграмм

= 14879

Отсутствие портов

= 3199

Ошибки при получении

= 2106

Отправлено датаграмм

= 10138

Статистика UDP для IPv6

Получено датаграмм

= 3079

Отсутствие портов

= 696

Ошибки при получении

= 0

Отправлено датаграмм

= 3901

Активные подключения

Имя

Локальный адрес

Внешний адрес

C:\Users\pam2>

ПРИЛОЖЕНИЕ 4

Арг с доп. параметрами

C:\Windows\system32>arp -d \*

C:\Windows\system32>arp -a

Интерфейс: 192.168.56.1 --- 0x15

адрес в Интернете

Физический адрес

Тип

224.0.0.22

01-00-5e-00-00-16

статический

255.255.255.255

ff-ff-ff-ff-ff-ff

статический

Интерфейс: 192.168.0.69 --- 0x16

адрес в Интернете

Физический адрес

Тип

192.168.0.1

28-3b-82-47-49-08

динамический

224.0.0.2

01-00-5e-00-00-02

статический

224.0.0.22

01-00-5e-00-00-16

статический



# Wireshark

The screenshot displays the Wireshark network protocol analyzer interface. The top menu bar includes options like File, Edit, View, Capture, Analyze, Statistics, Telephony, Wireless, Instruments, and Help. Below the menu is a toolbar with various icons for file operations, packet capture, and analysis. The main window is divided into three panes: the packet list, packet details, and packet bytes.

The packet list pane shows a list of 24 captured packets. The selected packet is packet 118, which is an HTTP GET request. The details pane for packet 118 shows the following structure:

- Frame 118: 524 bytes on wire (4192 bits), 524 bytes captured (4192 bits) on interface \Device\NPF\_{F8F5449A-6B18-48A}
- Ethernet II, Src: Giga-Byt\_3f:f5:81 (50:e5:49:3f:f5:81), Dst: D-LinkIn\_47:49:08 (28:3b:82:47:49:08)
- Internet Protocol Version 4, Src: 192.168.0.69, Dst: 104.18.32.61
- Transmission Control Protocol, Src Port: 54021, Dst Port: 80, Seq: 1, Ack: 1, Len: 470
- Hypertext Transfer Protocol

The packet bytes pane shows the raw data of the selected packet, which is a GET request for the path /administrator/.

## ПРИЛОЖЕНИЕ 6

## Разбор пакета сети

```

Domain Name System (response)
  Transaction ID: 0x9435
  > Flags: 0x8180 Standard query response, No error
  Questions: 1
  Answer RRs: 2
  Authority RRs: 0
  Additional RRs: 0
  > Queries
  < Answers
    > update.microsoft.com: type CNAME, class IN, cname update.microsoft.com.nsatsc.net
    < update.microsoft.com.nsatsc.net: type A, class IN, addr 65.55.184.151
      Name: update.microsoft.com.nsatsc.net
      Type: A (Host Address) (1)
      Class: IN (0x0001)
      Time to live: 150 (2 minutes, 30 seconds)
      Data length: 4
      Address: 65.55.184.151
[Request In: 528]
[Time: 0.028467000 seconds]

```

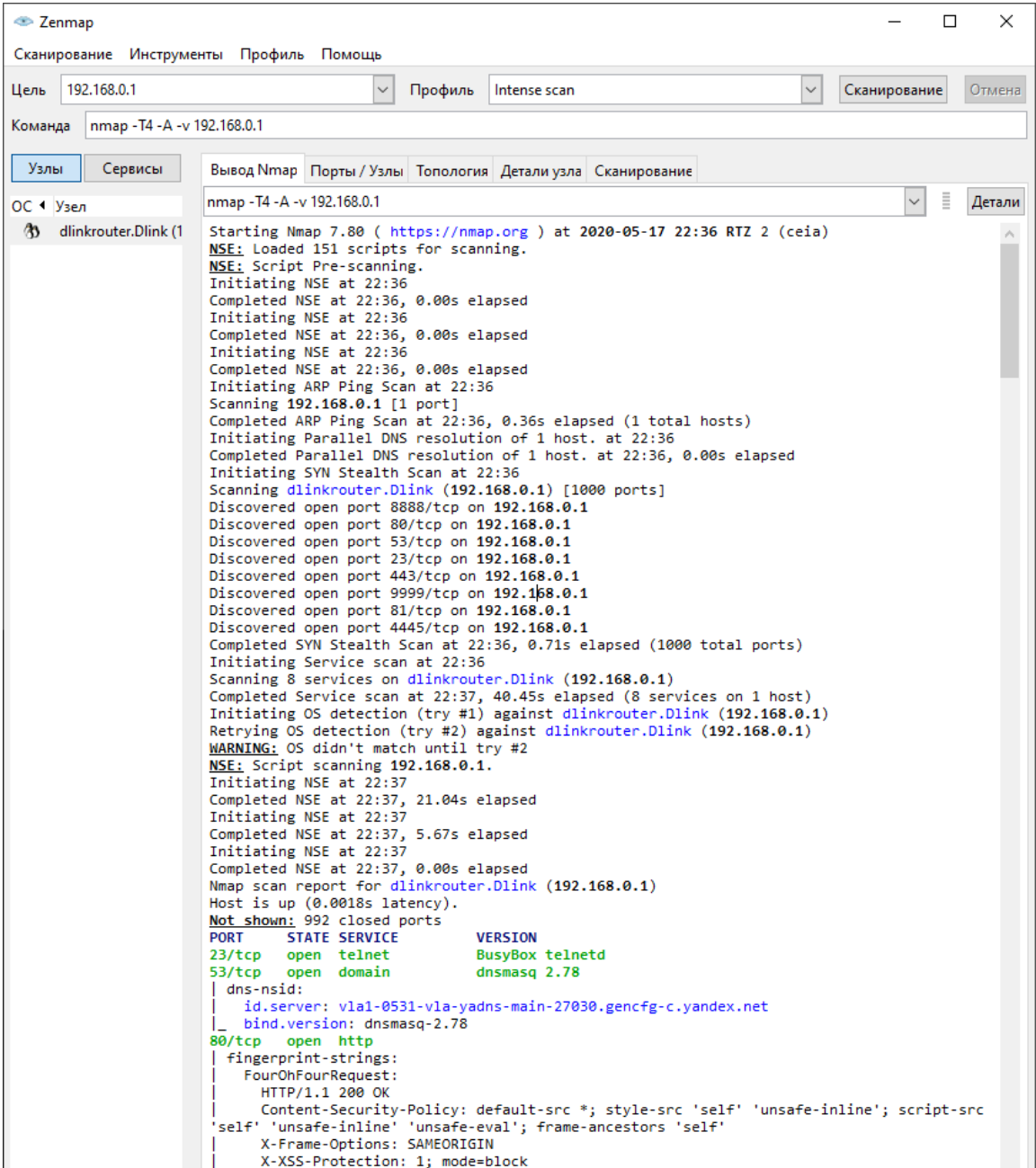
ПРИЛОЖЕНИЕ 7

Статистика сайта

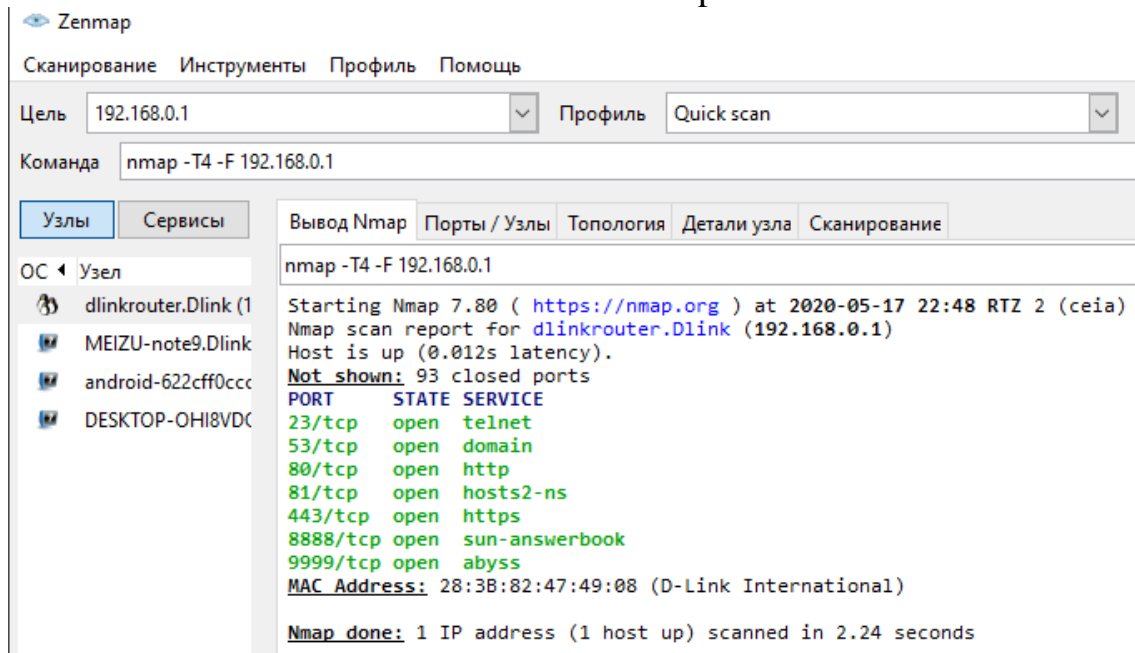
```
▼ Hypertext Transfer Protocol
> GET /images/layout/logo.png HTTP/1.1\r\n
Host: packetlife.net\r\n
User-Agent: Mozilla/5.0 (X11; U; Linux x86_64; en-US; rv:1.9.2.3) Gecko/20100423 Ubuntu/10.04.2 LTS Firefox/3.6.8\r\n
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
Accept-Language: en-us,en;q=0.5\r\n
Accept-Encoding: gzip,deflate\r\n
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7\r\n
Keep-Alive: 115\r\n
Connection: keep-alive\r\n
```

ПРИЛОЖЕНИЕ 8

Port

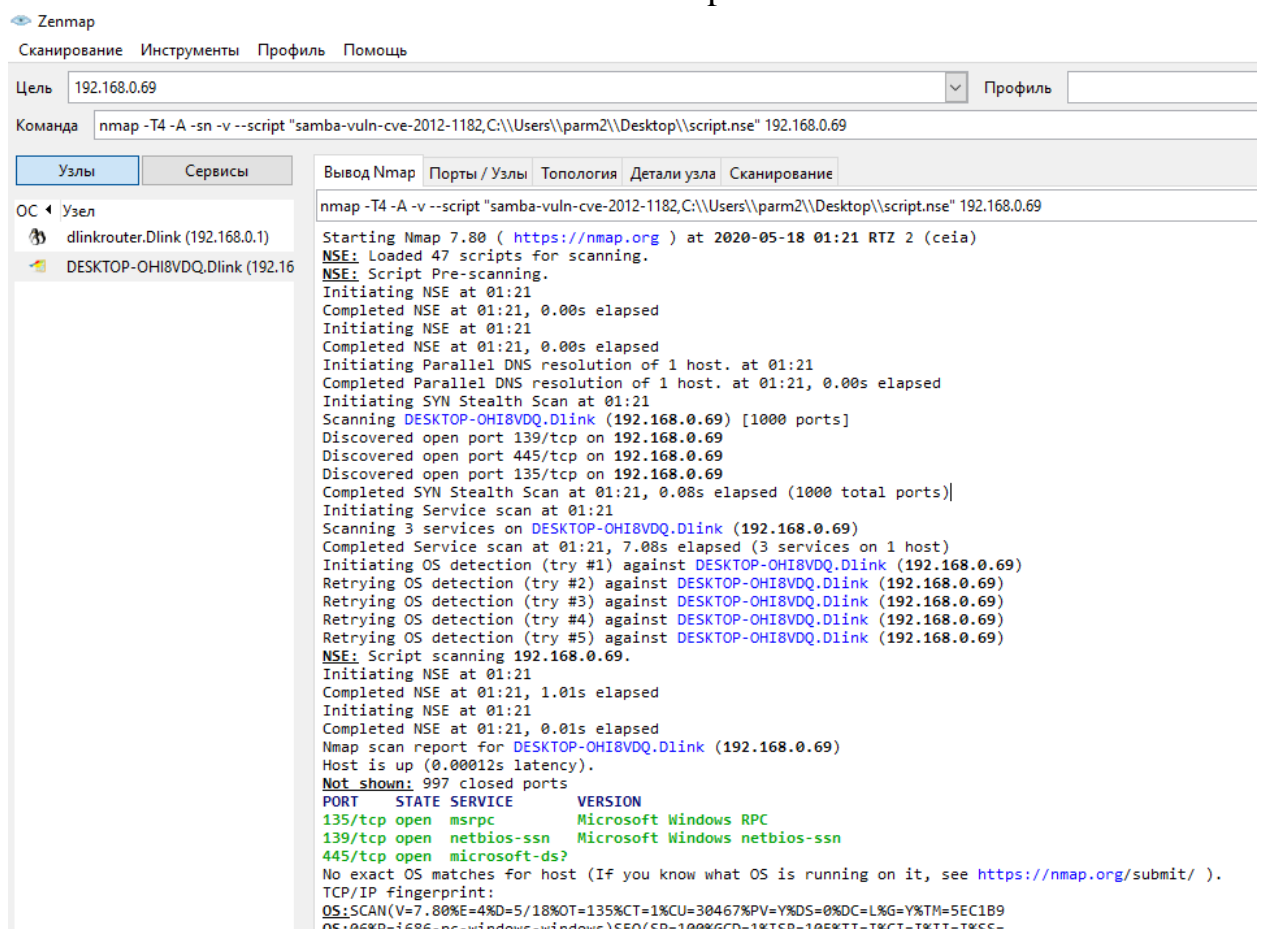


## Статистика port



## ПРИЛОЖЕНИЕ 10

## Атака nmap



## Код script python

```

hostrule = function(host)
    return smb.get_port(host) ~= nil
end

action = function(host,port)

    local result, stats
    local response = { }

    local samba_cve = {
        title = "SAMBA remote heap overflow",
        IDS = { CVE = 'CVE-2012-1182' },
        risk_factor = "HIGH",
        scores = {
            CVSSv2 = "10.0 (HIGH) (AV:N/AC:L/Au:N/C:C/I:C/A:C)",
        },
        description = [[
Samba versions 3.6.3 and all versions previous to this are affected by
a vulnerability that allows remote code execution as the "root" user
from an anonymous connection.
]],
        references = {
            'http://www.samba.org/samba/security/CVE-2012-1182',
        },
        dates = {
            disclosure = { year = '2012', month = '03', day = '15' },
        },
        exploit_results = { },
    }

    local report = vulns.Report:new(SCRIPT_NAME, host, port)
    samba_cve.state = vulns.STATE.NOT_VULN

    -- create SMB session
    local status, smbstate
    status, smbstate = msrpc.start_smb(host, msrpc.SAMR_PATH,true)

```

```

if(status == false) then
    return false, smbstate
end

-- bind to SAMR service
local bind_result
status, bind_result = msrpc.bind(smbstate, msrpc.SAMR_UUID, msrpc.SAMR_VERSION, nil)
if(status == false) then
    msrpc.stop_smb(smbstate)
    return false, bind_result
end

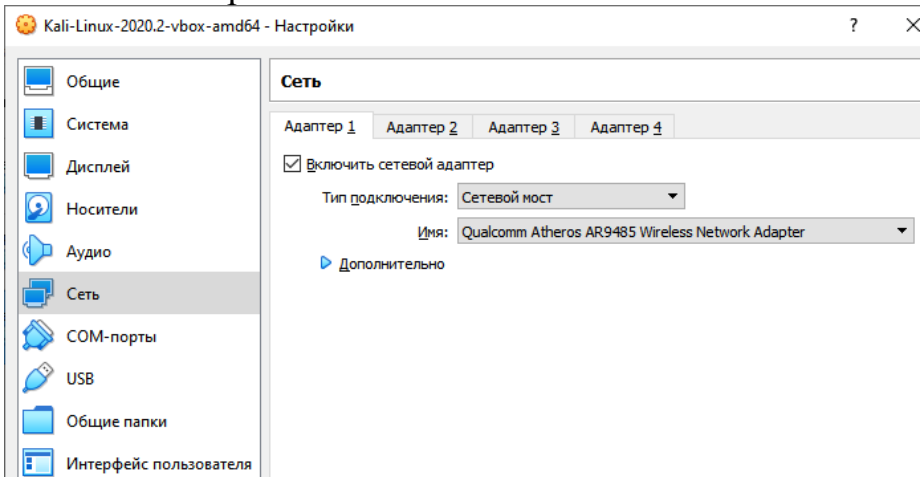
-- create malicious packet, same as in the PoC
local data = string.pack("<I4",4096) -- num_sids
    .. "abcd"
    ..string.pack("<I4I4I4",100
        ,0
        ,100)
    ..string.rep("a",1000)
local marshaledHandle = string.rep("X",20)
status, result = msrpc.samr_getaliasmembership(smbstate,marshaledHandle, data)
stdnse.debug2("msrpc.samr_getaliasmembership: %s, '%s'", status, result)
if(status == false and string.find(result,"Failed to receive bytes after 5 attempts") ~= nil) then
    samba_cve.state = vulns.STATE.VULN -- connection dropped, server crashed
end
return report:make_output(samba_cve)

```

end

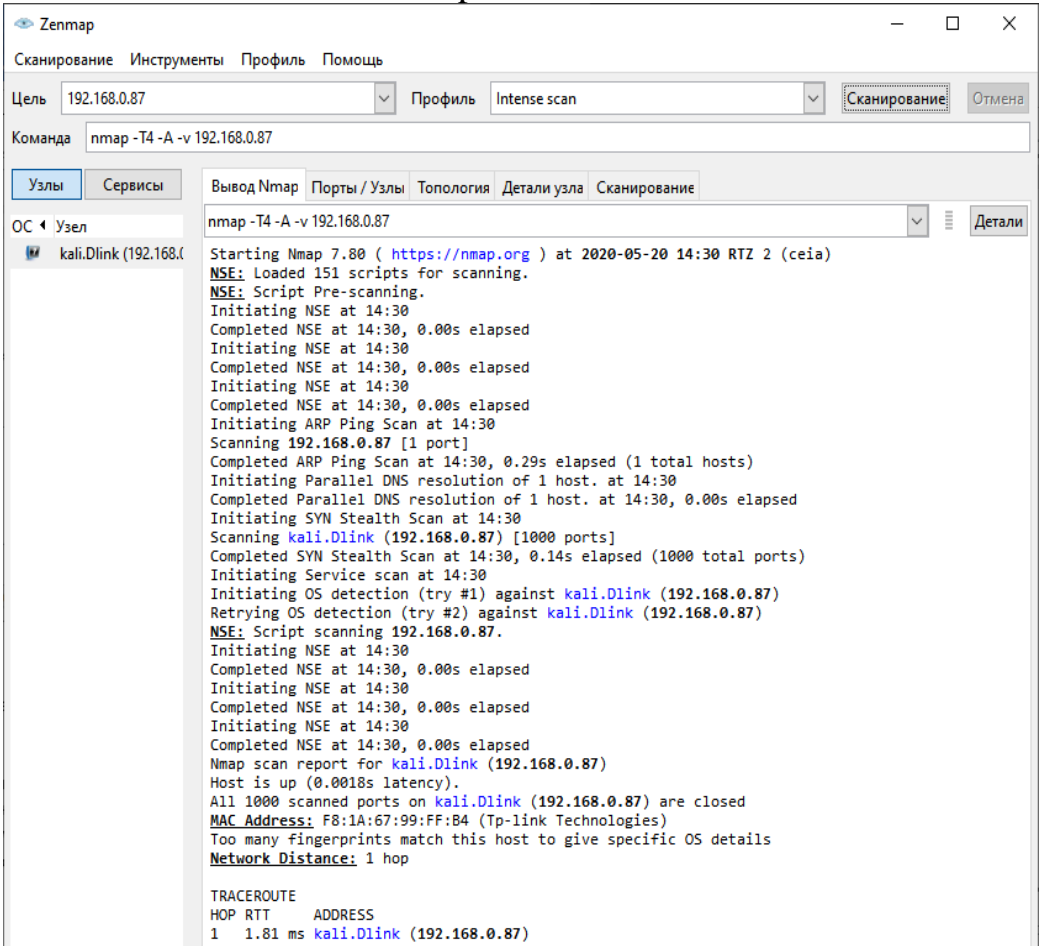
ПРИЛОЖЕНИЕ 12

Настройка vm box



ПРИЛОЖЕНИЕ 13

Zenmap



## ПРИЛОЖЕНИЕ 14

## Отчет wireshark

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
2	0.839984	192.168.0.30	224.0.0.251	MDNS	103	Standard query 0x012e PTR
3	1.024189	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
4	2.017440	192.168.0.87	192.168.0.69	TCP	74	54790 → 80 [SYN] Seq=0 Win=...
5	2.017525	192.168.0.87	192.168.0.69	TCP	74	55492 → 443 [SYN] Seq=0 Win=...
6	2.047728	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
7	2.302516	192.168.0.38	239.255.255.250	UDP	85	5050 → 5050 Len=43
8	2.326556	192.168.0.69	192.168.0.255	UDP	85	5050 → 5050 Len=43
9	2.562541	fe80::9460:664d:7476:5790	fe80::2a3b:82ff:fe47:4908	DNS	94	Standard query 0x97fa AAAA
10	2.604036	fe80::2a3b:82ff:fe47:4908	fe80::9460:664d:7476:5790	DNS	253	Standard query response 0xc...
11	3.074452	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
12	3.304152	fe80::9460:664d:7476:5790	fe80::2a3b:82ff:fe47:4908	DNS	110	Standard query 0x8b32 AAAA
13	3.401056	fe80::2a3b:82ff:fe47:4908	fe80::9460:664d:7476:5790	DNS	282	Standard query response 0xc...
14	3.518323	192.168.0.87	192.168.0.69	TCP	74	55494 → 443 [SYN] Seq=0 Win=...
15	3.518401	192.168.0.87	192.168.0.69	TCP	74	54796 → 80 [SYN] Seq=0 Win=...
16	4.094357	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
17	5.117445	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
18	6.140910	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
19	7.166588	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
20	7.310249	192.168.0.38	239.255.255.250	UDP	85	5050 → 5050 Len=43
21	7.343978	192.168.0.69	192.168.0.255	UDP	85	5050 → 5050 Len=43
22	7.369204	fe80::9460:664d:7476:5790	fe80::2a3b:82ff:fe47:4908	ICMPv6	86	Neighbor Solicitation for
23	7.371076	fe80::2a3b:82ff:fe47:4908	fe80::9460:664d:7476:5790	ICMPv6	78	Neighbor Advertisement fe80::...
24	7.603344	fe80::2a3b:82ff:fe47:4908	fe80::9460:664d:7476:5790	ICMPv6	86	Neighbor Solicitation for
25	7.603411	fe80::9460:664d:7476:5790	fe80::2a3b:82ff:fe47:4908	ICMPv6	86	Neighbor Advertisement fe80::...
26	8.187927	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
27	9.212228	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
28	10.235510	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
29	11.259124	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
30	12.282471	192.168.0.87	192.168.0.69	ICMP	98	Echo (ping) request id=0xc...
31	12.317957	192.168.0.38	239.255.255.250	UDP	85	5050 → 5050 Len=43
32	12.361370	192.168.0.69	192.168.0.255	UDP	85	5050 → 5050 Len=43

> Frame 14: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)  
 > Ethernet II, Src: Tp-LinkT\_99:ff:b4 (f8:1a:67:99:ff:b4), Dst: Giga-Byt\_3f:f5:81 (50:e5:49:3f:f5:81)  
 > Internet Protocol Version 4, Src: 192.168.0.87, Dst: 192.168.0.69  
 > Transmission Control Protocol, Src Port: 55494, Dst Port: 443, Seq: 0, Len: 0

## ПРИЛОЖЕНИЕ 15

## Запрос TCP

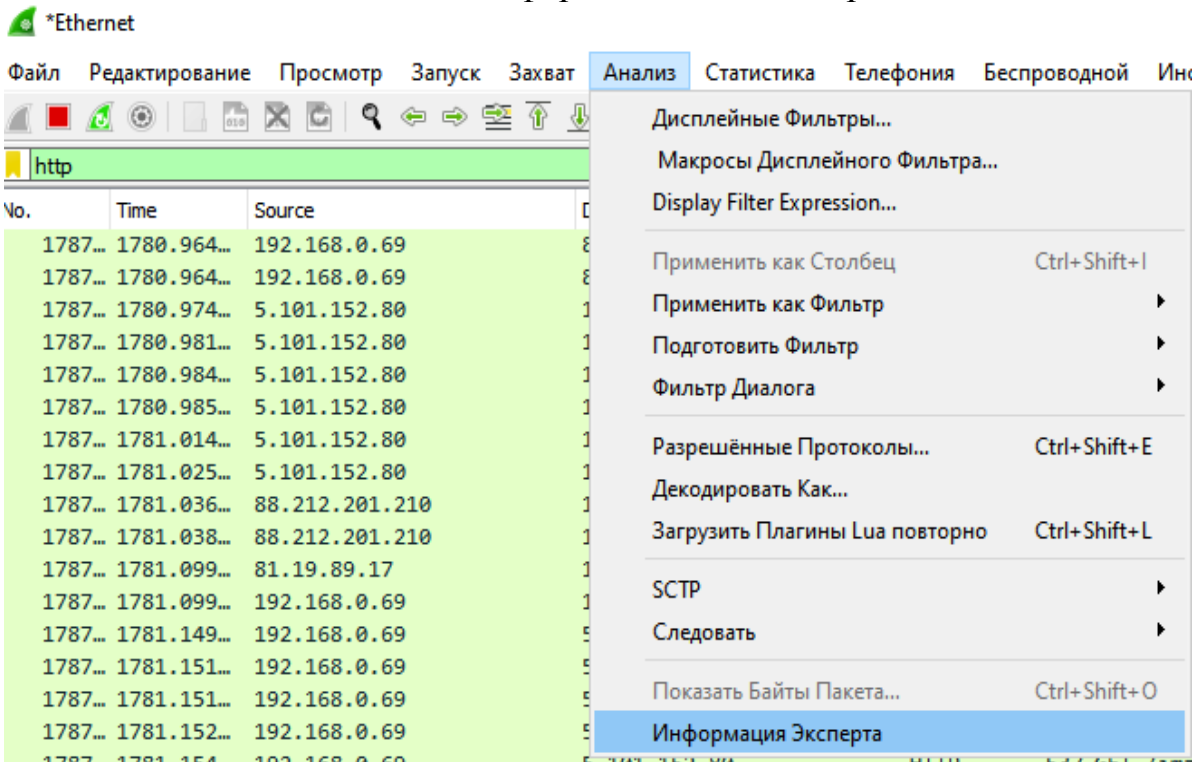
No.	Time	Source	Destination	Protocol	Length	Info
7159	177.382819	93.186.225.201	192.168.0.69	TCP	66	443 → 49794 [ACK] Seq=3191 Ack=1351 W...
7160	178.505914	192.168.0.69	178.154.131.215	TCP	55	[TCP Keep-Alive] 49918 → 443 [ACK] Seq=...
7161	178.523677	178.154.131.215	192.168.0.69	TCP	78	[TCP Keep-Alive ACK] 443 → 49918 [ACK] Seq=...
7162	178.832351	192.168.0.69	80.239.142.164	TLSv1.2	112	Application Data
7163	178.832402	192.168.0.69	80.239.142.164	TLSv1.2	451	Application Data
7164	178.832461	192.168.0.69	80.239.142.164	TLSv1.2	143	Application Data
7165	178.874336	80.239.142.164	192.168.0.69	TCP	66	443 → 49748 [ACK] Seq=889465 Ack=40760
7166	178.874540	80.239.142.164	192.168.0.69	TLSv1.2	112	Application Data
7167	178.874540	80.239.142.164	192.168.0.69	TCP	66	443 → 49748 [ACK] Seq=889511 Ack=40813
7168	178.915670	192.168.0.69	80.239.142.164	TCP	66	49748 → 443 [ACK] Seq=408130 Ack=88951
7169	178.931181	80.239.142.164	192.168.0.69	TLSv1.2	160	Application Data
7170	178.931451	192.168.0.69	80.239.142.164	TLSv1.2	143	Application Data
7171	178.932172	80.239.142.164	192.168.0.69	TLSv1.2	192	Application Data
7172	178.972917	192.168.0.69	80.239.142.164	TCP	66	49748 → 443 [ACK] Seq=408207 Ack=88973
7173	179.010214	80.239.142.164	192.168.0.69	TCP	66	443 → 49748 [ACK] Seq=889731 Ack=408207

> Frame 4683: 371 bytes on wire (2968 bits), 371 bytes captured (2968 bits) on interface \Device\NPF\_{F8F5449A-6B18-48...  
 > Ethernet II, Src: D-LinkIn\_47:49:08 (28:3b:82:47:49:08), Dst: Giga-Byt\_3f:f5:81 (50:e5:49:3f:f5:81)  
 > Internet Protocol Version 4, Src: 213.196.40.55, Dst: 192.168.0.69  
 > Transmission Control Protocol, Src Port: 443, Dst Port: 49923, Seq: 3950, Ack: 1749, Len: 305  
 > Transport Layer Security



ПРИЛОЖЕНИЕ 16

Как попасть в Информацию для эксперта



ПРИЛОЖЕНИЕ 17

Информация Эксперта

Wireshark · Информация Эксперта · Ethernet				
Степень тяжести	Сводка	Группа	Протокол	Подсчёт
> Error	TLSCiphertext length MUST NOT exceed 2^14 + 2048	Protocol	TLS	1
> Warning	DNS query retransmission. Original request in frame 2855	Protocol	LLMNR	12
> Warning	DNS query retransmission. Original request in frame 2840	Protocol	mDNS	44
> Warning	This frame is a (suspected) out-of-order segment	Sequence	TCP	30
> Warning	Previous segment(s) not captured (common at capture sta...	Sequence	TCP	11
> Warning	Window scale shift exceeds 14	Protocol	TCP	8
> Warning	The non-SYN packet does contain a MSS option	Protocol	TCP	20
> Warning	Ignored Unknown Record	Protocol	TLS	147
> Warning	Connection reset (RST)	Sequence	TCP	129
> Note	"Time To Live" != 255 for a packet sent to the Local Networ...	Sequence	IPv4	24
> Note	This frame is a (suspected) fast retransmission	Sequence	TCP	6
> Note	ACK to a TCP keep-alive segment	Sequence	TCP	78
> Note	TCP keep-alive segment	Sequence	TCP	78
> Note	A new tcp session is started with the same ports as an earli...	Sequence	TCP	9
> Note	The urgent pointer field is nonzero while the URG flag is no...	Protocol	TCP	2
> Note	The SYN packet does not contain a MSS option	Protocol	TCP	2
> Note	The acknowledgment number field is nonzero while the A...	Protocol	TCP	38
> Note	This session reuses previously negotiated keys (Session res...	Sequence	TLS	7
> Note	This frame is a (suspected) retransmission	Sequence	TCP	87
> Note	Duplicate ACK (#1)	Sequence	TCP	223
> Chat	GET / HTTP/1.0\r\n	Sequence	HTTP	27
> Chat	Connection establish acknowledge (SYN+ACK): server por...	Sequence	TCP	151
> Chat	Connection establish request (SYN): server port 5900	Sequence	TCP	194
> Chat	Connection finish (FIN)	Sequence	TCP	151
> Chat	M-SEARCH * HTTP/1.1\r\n	Sequence	SSDP	104