## Лекция 2.4. Сети. Инструменты по работе с сетевыми пакетами

В данной лекции рассмотрены следующие вопросы:  
1. Структура сетевых моделей.  
2. Знакомство с моделью ISO/OSI.  
3. Сходства и различия моделей OSI и TCP/IP.  
4. Инкапсуляция и деинкапсуляция данных.

### Глава 1. Сети. Инструменты по работе с сетевыми пакетами

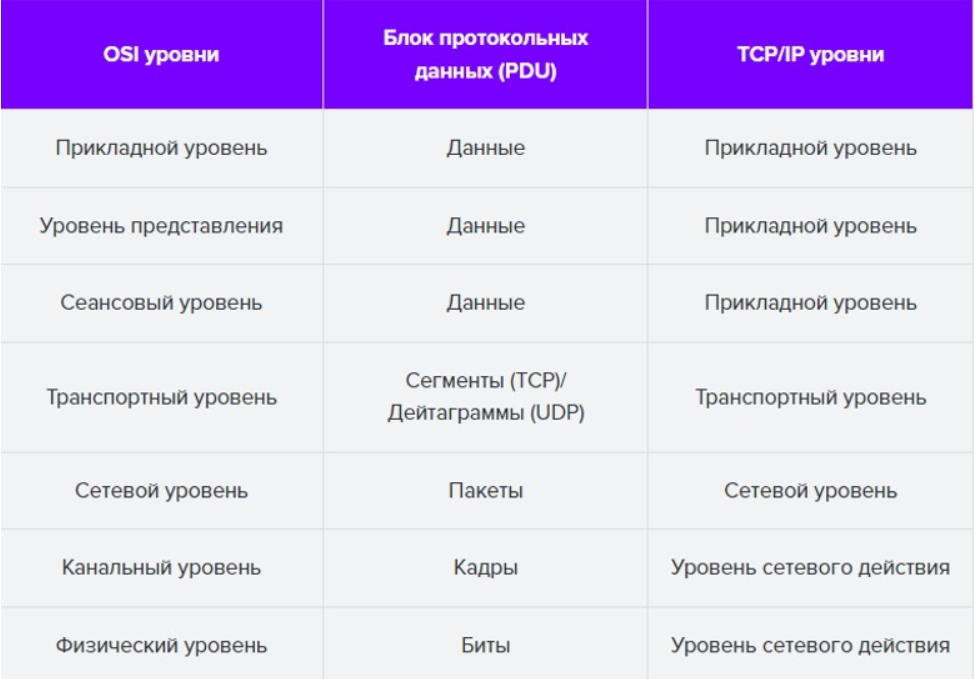
Изучение сетей начинается с моделей OSI и TCP/IP. Именно они дают понимание того, как происходит взаимодействие между двумя хостами как в одной локальной сети, так и на разных уголках планеты. Они идеально описывают взаимодействие между процессами и приложениями на конечных системах.  
Взаимодействие в сетях происходит между хостами — узлами сети, любого компьютера, сервера (физического или виртуального), подключённого к локальной или глобальной сети.  
**Хостинг** — это услуга по предоставлению ресурсов для размещения информации на сервере, постоянно имеющем доступ к сети (обычно Интернет):  
1.  Для описания взаимодействия двух хостов на одном уровне используется понятие **«Протокол».**  
2.  Для описания взаимодействия между двумя уровнями в пределах одного хоста используется понятие **«Интерфейс».**  
**Сетевой интерфейс** — физическое или виртуальное устройство, предназначенное для передачи данных между программами через компьютерную сеть.  
Это могут быть как физические интерфейсы сетевых карт и телекоммуникационных устройств, так и открытые логические соединения, через которые передаются данные. Взаимодействие между интерфейсами определяется сетевыми протоколами.  
**Сетевой протокол** — набор правил, определяющих способы управления соединениями на физическом и логическом уровнях. Для выполнения своих функций одни протоколы часто используют другие протоколы.  
Знакомство с моделью ISO/OSI  
Модель ISO/OSI (ISO Open System Interconnection, модель взаимодействия открытых систем) — это эталонная модель сети, которая описывает взаимодействие устройств в сети.  
Классическая модель OSI состоит из семи уровней, каждый из которых решает свои задачи и взаимодействует со смежными уровнями.  
7 уровень: Прикладной.  
6 уровень: Представления.  
5 уровень: Сеансовый.  
4 уровень: Транспортный.  
3 уровень: Сетевой.  
2 уровень: Канальный.  
1 уровень: Физический.

Глава 2. Сходства и различия моделей OSI и TCP/IP

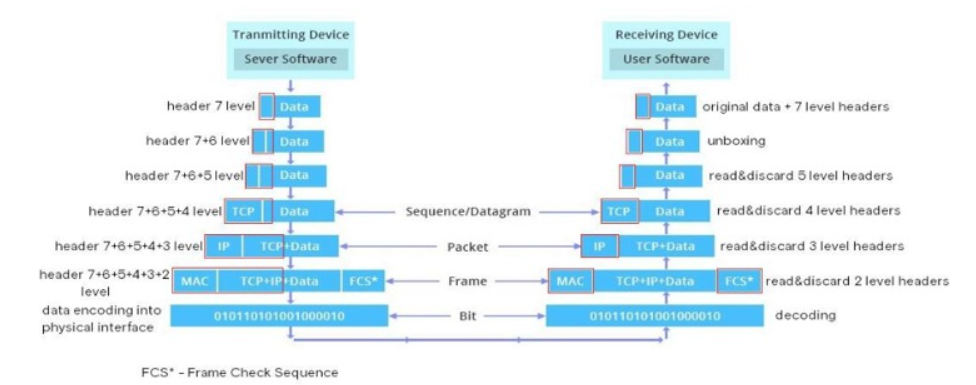
### Сходства и различия моделей OSI и TCP/IP С точки зрения ИБ, при разделении сетей на уровни инженеру по безопасности становится легче определять и описывать сетевые взаимодействия в процессе расследования инцидентов ИБ. Кроме модели OSI также существует сетевая модель TCP/IP. Посмотрим, чем они отличаются:



Обычно последние три уровня (прикладной, представления и сеансовый) объединяют в один и называют прикладным, а первые два (физический и канальный) объединяют в канальный (его также называют «уровень доступа к сети»). Транспортный уровень протокола TCP/IP включает в себя функции транспортного уровня OSI и некоторые функции сеансового уровня модели OSI. Таким образом, теоретическая модель OSI сжимается до применяемого на практике четырёхуровневого стека протоколов TCP/IP.  
Следует отметить, что в модели OSI не все уровни могут быть использованы для передачи данных.  
Модель TCP/IP широко известна как TCP/IP, потому что её основополагающие протоколы — TCP и IP, однако, не только эти два протокола используются в этой модели.  
Идеология сетевых моделей  
Модель OSI является концептуальной теоретической моделью. Она в основном используется для описания, обсуждения и понимания отдельных сетевых функций. Однако, TCP/IP в первую очередь сконструирована для того, чтобы разрешить специфический круг проблем, а не действовать как описание поколений сетей для всех сетевых взаимодействий (как это происходит в модели OSI). Вспоминаем, что IP-сети появились не сразу.  
Цифровая передача данных представляет собой физический процесс, в котором данные переносятся в виде сигналов между точками (абонентами).  
Выполняется передача данных по определённым каналам связи.  
В качестве таких каналов могут выступать:  
1. Проводные — витая пара, коаксиал.  
2. Акустические и оптические — оптоволокно.  
3. Инфракрасные и радиоканалы — беспроводные.  
Инкапсуляция и деинкапсуляция данных  
В многоуровневой системе устройства обмениваются данными в формате, известном как блок протокольных данных (PDU). В таблице ниже показаны PDU по уровням сетевых моделей.



Во время передачи каждый уровень добавляет новые заголовки к PDU, поступающему от вышестоящего уровня. Этот процесс называется инкапсуляцией.



В приёмном устройстве происходит обратный процесс, деинкапсуляции данных на каждом уровне.

### Глава 3. Сети. Инструменты по работе с сетевыми пакетами

### Физический уровень модели OSI Физический уровень. Низший уровень модели OSI. Он регламентирует передачу данных непосредственно по физическому каналу связи. На этом уровне данные представляют из себя электрический сигнал, модулированный информацией, переданной от вышестоящих уровней модели OSI. Функции физического уровня реализуются во всех устройствах, подключённых к сети. Со стороны компьютера функции физического уровня выполняются сетевым адаптером. К физическому уровню относится та часть функций сетевого адаптера, которая связана с приёмом и передачей сигналов по линии связи (проводной или беспроводной). Если сказать проще, то задача физического уровня — выпустить трафик в сеть, а дальше будь что будет. Протокол ARP ARP (англ. Address Resolution Protocol — протокол разрешения адресов) представляет собой сетевой протокол, предназначенный для преобразования IP-адресов в MAC-адреса в компьютерных сетях. Такое преобразование требуется, когда IP-пакет требуется передать по локальной сети. Как передаются данные в сети Интернет MAC-адрес идентифицирует твой компьютер в сети. IP-адрес определяет его местоположения, а протоколы создают правила передачи информации. ARP-запросы позволяют определить получателя пакетов, в то время как ARP-таблицы сохраняют всю информацию для дальнейших запросов. Инструменты для работы с сетью Популярные инструменты для работы с сетью на Python включают: 1. «Socket» — это стандартная библиотека Python для работы с сокетами. Она предоставляет простой и удобный интерфейс для создания клиентов и серверов TCP и UDP. 2. «Requests» — это библиотека Python для работы с HTTP-запросами. Она позволяет легко отправлять GET-, POST-, PUT- и другие запросы на удалённый сервер и получать ответы. 3. «Urllib» — это стандартная библиотека Python для работы с URL-адресами. Она позволяет извлекать данные с веб-страниц, скачивать файлы и отправлять запросы на сервер. 4. «Paramiko» — это библиотека Python для работы с протоколом SSH. Она позволяет подключаться к удалённому серверу по SSH, выполнять команды и передавать файлы. 5. «Netifaces» — это библиотека Python для получения информации о сетевых интерфейсах. Она позволяет получать информацию о IP-адресах, MAC-адресах, маршрутизации и других параметрах сетевых интерфейсов. 6. «Scapy» — это библиотека Python для создания и обработки сетевых пакетов на низком уровне. Она позволяет создавать, отправлять, перехватывать и анализировать пакеты на любом уровне сетевой модели. 7. «Pyshark» — это библиотека Python для работы с файлами захвата сетевых пакетов формата pcap. Она позволяет извлекать информацию из сетевых пакетов, фильтровать их и анализировать на любом уровне сетевой модели. 8. «Pexpect» — это библиотека Python для работы с интерактивными консольными приложениями. Она позволяет автоматизировать взаимодействие с консольными приложениями, такими как Telnet, SSH и другие. Это только некоторые из инструментов для работы с сетью на Python. Python предоставляет множество других библиотек и модулей для решения различных задач, связанных с сетевой работой.

1. Устройство или приложение, которое инициирует запросы к серверу:
   1. API
   2. Сервер
   3. Клиент
2. Устройство или приложение, которое обрабатывает запросы, и предоставляет ответы:
   1. Сервер
   2. Клиент
   3. API
3. Интерфейс, который позволяет приложениям взаимодействовать друг с другом называется:
   1. API
   2. Сервер
   3. Клиент
4. Какого элемента нет в структуре запроса?
   1. Клиент
   2. Метод
   3. Тело
   4. Заголовок
   5. URL
5. Какой запрос используется для получения данных с сервера, добавляя параметры запроса в URL и передавая их в открытом виде?
   1. Нет верного ответа
   2. POST-запрос
   3. GET-запрос
6. Какой запрос используется для отправки данных на сервер, добавляя параметры запроса в тело запроса, и передавая их скрыто от пользователя.
   1. Нет верного ответа
   2. GET-запрос
   3. POST-запрос
7. Какие методы не используются в HTTP протоколе?
   1. OSI
   2. DELETE
   3. GET
   4. POST