# Практическое занятие: Проверка целостности данных.

Целью занятия является формирование и закрепление навыков по проверке целостности данных.

Задачи:

1. Вычислить CRC32.

2. Рассчитать MD5 и SHA сумму для текстового файла.

3. Научиться использовать возможности сервиса ftp.

4. Научиться использовать анализатор трафика tcpdump.

**Необходимые теоретические сведения.**

Контрольная сумма — некоторое значение, рассчитанное по набору данных путём применения определённого алгоритма и используемое для проверки целостности данных при их передаче или хранении. Также контрольные суммы могут использоваться для быстрого сравнения двух наборов данных на неэквивалентность: с большой вероятностью различные наборы данных будут иметь неравные контрольные суммы. Несмотря на своё название, контрольная сумма не обязательно вычисляется путём суммирования.

С точки зрения [математики](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0) контрольная сумма является результатом [хеш-функции](https://ru.wikipedia.org/wiki/%D0%A5%D0%B5%D1%88%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5), используемой для вычисления [контрольного кода](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%B4%D1%8B,_%D0%B8%D1%81%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D1%8F%D1%8E%D1%89%D0%B8%D0%B5_%D0%BE%D1%88%D0%B8%D0%B1%D0%BA%D0%B8) — небольшого количества [бит](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D1%82) внутри большого блока данных, например, сетевого [пакета](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D0%BA%D0%B5%D1%82_(%D1%81%D0%B5%D1%82%D0%B5%D0%B2%D1%8B%D0%B5_%D1%82%D0%B5%D1%85%D0%BD%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D0%B8)) или блока [компьютерного](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80) [файла](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB), применяемого для обнаружения ошибок при передаче или хранении [информации](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D1%8F). Значение контрольной суммы добавляется в конец блока данных непосредственно перед началом передачи или записи данных на какой-либо [носитель информации](https://ru.wikipedia.org/wiki/%D0%9D%D0%BE%D1%81%D0%B8%D1%82%D0%B5%D0%BB%D1%8C_%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D0%B8). Впоследствии оно проверяется для подтверждения [целостности](https://ru.wikipedia.org/wiki/%D0%A6%D0%B5%D0%BB%D0%BE%D1%81%D1%82%D0%BD%D0%BE%D1%81%D1%82%D1%8C_%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D0%B8) данных.

Популярность использования контрольных сумм для проверки целостности данных обусловлена тем, что подобная проверка просто реализуема в [двоичном](https://ru.wikipedia.org/wiki/%D0%94%D0%B2%D0%BE%D0%B8%D1%87%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%81%D1%87%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F) [цифровом оборудовании](https://ru.wikipedia.org/wiki/%D0%A6%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D0%BE%D0%B9_%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80), легко анализируется и хорошо подходит для обнаружения общих ошибок, вызванных наличием шума в каналах передачи данных.

**Практическая часть.**

Алгоритм CRC

CRC (Cyclic Redundancy Code - циклический избыточный код) - алгоритм расчёта контрольной суммы для передаваемого сообщения, основанный на полиномиальной арифметике. Основная идея алгоритма CRC состоит в представлении сообщения в виде огромного двоичного числа, делении его на другое фиксированное двоичное число и использовании остатка от этого деления в качестве контрольной суммы. Получив сообщение, приёмник должен выполнить аналогичное действие и сравнить полученный результат с принятой контрольной суммой. Сообщение считается достоверным, выполняется это равенство. Алгоритм CRC базируется на полиномиальной арифметике, а это означает, что сообщение, делитель и остаток могут быть представлены в виде полиномов с двоичными коэффициентами или в виде строки битов, каждый из которых является коэффициентом полинома. CRC чувствителен к перестановке порядка битов в сообщении. CRC может применяться не только для контроля целостности сообщений, но и для контроля целостности файлов. Если изменился CRC, то файл повреждён.

В Linux есть встроенная утилита cksum. Она выводит CRC32 в виде десятичного числа для файла, заданного ей в качестве параметра.

Хэш(Hash)-суммы

Хэш-сумма - это уникальный контрольный идентификатор одного или группы файлов. Его повторное вычисление при помощи специального алгоритма (MD5 или SHA, в контексте документа) не даёт прежнего результата даже при малейшем изменении файлов. Хэш-сумма представляет собой некоторое шестнадцатеричное число, являющееся уникальным идентификатором файла. Проверка совпадения контрольных хэш-сумм даёт возможность установить идентичность копий переданных, в том числе через сеть, данных с очень высокой точностью.

Рассчитать контрольную сумму файла можно в любом дистрибутиве Linux с помощью программ md5sum и shasum. Чтобы посчитать хэш-сумму, нужно в терминале выполнить следующие команды:

Для вычисления сумм MD5: md5sum имя\_файла

Для вычисление сумм SHA (с применением 1-, 256-, 384- или 512-битного алгоритма, соответственно):

shasum (-a 1/256/384/512) имя\_файла

Консольный ftp-клиент

Соединиться с FTP-сревером можно используя любой вёб-браузер или файловый менеджер. Но на сервере графический интерфейс может быть недоступен, и в таких случаях следует использовать консольный клиент. Программа ftp позволяет соединяться с сервером в режиме командной строки. Единственным параметром является URL или IP-адрес сервера. Протокол FTP всегда требует логин и пароль. Возможен анонимный режим, когда логин всегда anonymous, а пароль – случайный e-mail. Далее управление осуществляется при помощи команд, похожих на системные. Перечислим некоторые из них:

• dir - отображает список каталогов.

• cd - переходит в каталог, указанный параметром.

• get - скачивает файл, указанный параметром, в текущую директорию на диске.

• help - отображает справку по всем командам.

Средства командной строки для скачивания файлов

Если графический интерфейс недоступен (сервер, встраиваемое оборудование), то следует использовать средства командной строки для того, чтобы скачать какой-либо файл на сервер. Для этого служит две команды (также эти команды нужно использовать в скриптах):

1. wget - Имеет один обязательный параметр (URL, которую нужно скачать) и множество необязательных параметров. Wget работает с любым протоколом (FTP, HTTP, HTTPS). Типичное использование:

wget -c http://some-URL/

Параметр –c означает включить докачку, этот параметр может отсутствовать. Второй параметр - это полный URL файла.

2.curl - Также имеет один обязательный параметр (URL). Выводит скачанный файл сразу в терминал. Чтобы сохранить файл, можно использовать перенаправление ввода-вывода. Если используется FTP, то wget и curl автоматически соединяются с ним в анонимном режиме.

Анализатор трафика tcpdump

Tcpdump - это анализатор трафика без графического интерфейса. Является индустриальным стандартом для диагностики ЛВС. Применяется в случаях, если графический интерфейс недоступен (сервер, встраиваемое оборудование). Данный анализатор трафика запускается командой tcpdump с несколькими параметрами. Команда может требовать административных прав, поэтому её нужно запускать от имени администратора и указав полный путь sudo /usr/sbin/tcpdump

Общий формат запуска команды:

tcpdump параметры ’фильтр’

Фильтр задаёт тип пакетов, которые нужно перехватить. Фильтр заключается в одинарные кавычки. В некоторых режимах фильтр может отсутствовать.

Приведём некоторые параметры:

• -i имя\_интерфейса

Указывает имя интерфейса, на котором нужно прослушивать трафик. Например -i eth0

• -c количество\_пакетов

Указывает сколько пакетов нужно перехватить

• -xx

Показывать содержимое пакетов в виде шестнадцатеричных кодов

• -XX

Показывать содержимое пакета в виде шестнадцатеричных кодов и ASCII-символов.

• -D

Показать интерфейсы, доступные для захвата.

Параметры можно комбинировать в любой последовательности. Фильтр формируется следующим образом:

• dst host <хост>

Будет отбирать пакеты, в которых поле адреса получателя IPv4/v6 содержит адрес хоста, заданного в примитиве.

• src host <хост>

Будет выбирать все пакеты, в которых поле отправителя содержит адрес указанного хоста.

• host <хост>

Будет отбирать все пакеты, для которых адрес хоста указан в поле получателя или отправителя.

• port <порт>

Отберет все пакеты ip/tcp, ip/udp, ip6/tcp и ip6/udp, направленные в указанный порт. Номера портов могут задаваться номерами или именами из файла /etc/services.

• ip proto <протокол>

Отбирает все пакеты IP, содержащие заданный идентификатор типа в поле типа протокола. Типы протоколов IP можно указывать по именам или (icmp, icmp6, igmp, igrp, pim, ah, esp, vrrp, udp, tcp) или номерам. Поскольку tcp, udp и icmp используются также в качестве ключевых слов, перед этими идентификаторами следует помешать символ \ (слэш). Отметим, что этот примитив не проверяет цепочки протокольных заголовков.

Пример командной строки:

tcpdump -i eth 0 -c 5 -xx ’src host 192.168.4.1 and ip proto \icmp’

Захватывает 5 пакетов с интерфейса eth0, исходящие от сервера 192.168.4.1 с протоколом icmp и отобразить их в шестнадцатеричном виде.

# Задание для закрепления практических навыков:

1. Вычислить CRC32 для некоторого текстового файла при помощи cksum. Затем внести изменения в файл, снова вычислить CRC32. Переставить местами несколько символов в файле, вычислить CRC32. Что изменилось? Как можно проконтролировать целостность файлов при помощи CRC?

2. Рассчитать MD5 и SHA(1, 256, 384 и 512 битную) сумму для некоторого текстового файла.

3. Соединиться с ftp сервером 192.168.4.1 при помощи консольного клиента. Привести список каталогов, скачать 1-2 файла.

4. Скачать несколько файлов с FTP сервера используя wget. Вывести файл data.txt в терминал при помощи curl.

5. При помощи tcpdump выполнить следующие задачи:

* Запустить tcpdump –D и получить список интерфейсов, доступных для захвата. Каким устройствам соответствуют эти интерфейсы?
* Захватить 10 любых пакетов с интерфейса eth0
* Отправить при помощи netcat (nc)текстовый файл и перехватить его содержимое при помощи tcpdump. Чтобы протестировать данный режим в пределах одного компьютера нужно использовать захват на петлевом интерфейсе lo. Следует применить фильтр по номеру порта, чтобы отобразить только обмен с Netcat.