A. V. Arkhangelskaya, V. G. Arkhangelskii, V. V. Kalmykov

the Technique to Test the Software Prototype of One-Way Data Gateway

Keywords: *one-way data transfer, one-way data gateway, secured data delivery, data accessibility, data integrity, software prototype, estimation procedure*

The purpose of the article was to investigate the interaction between a data source, a data sink and a one-way data gateway. The main steps are as follows: development of the one-way data gateway architecture and the data communication protocol, design of software data transfer prototype and investigation of its properties. The main result of the research was the technique to reduce a performance penalty.

А. В. Архангельская, В. Г. Архангельский, В. В. Калмыков

О ТЕСТИРОВАНИИ МАКЕТА ОДНОНАПРАВЛЕННОГО ШЛЮЗА

Введение

В настоящее время шлюзы однонаправленной передачи информации активно используются для обеспечения безопасной передачи данных из одного сегмента сети в другой сегмент с более высоким уровнем конфиденциальности, другими правилами разграничения доступа. Применение однонаправленного шлюза позволяет защитить конфиденциальную информацию сети-получателя от доступа извне. На данный момент представлено относительно небольшое количество таких решений и ни одно из них не обеспечивает однонаправленной гарантированной передачи больших объемов информации в автоматическом режиме. Авторами разработаны архитектура высокоскоростного шлюза однонаправленной гарантированной передачи данных, протокол обмена, обеспечивающий строго однонаправленную передачу данных с возможностью подтверждения их получения, и алгоритм функционирования высокоскоростного шлюза [1]. Настоящая статья посвящена исследованию свойств программного макета, осуществляющего моделирование взаимодействия участников обмена с однонаправленным шлюзом.

1. Архитектура однонаправленного высокоскоростного шлюза

Обеспечение гарантированной передачи данных в предлагаемом однонаправленном шлюзе достигается за счет разработанной архитектуры аппаратной реализации шлюза, а также благодаря использованию специального алгоритма однонаправленной гарантированной передачи информации и нового протокола взаимодействия источника и получателя, исключающего возможность обратной передачи данных из сети-получателя в сеть-источник, но позволяющего отправлять подтверждения получения данных сетью-получателем и противодействующего функционированию скрытых логических каналов выноса информации из сети-получателя.

Указанный протокол взаимодействия позволяет отправлять подтверждение о доставке блока данных без установления непосредственного соединения между источником и получателем. Исходя из необходимости простоты реализации и достижения высоких скоростных характеристик принято решение строить протокол на базе протокола UDP, не требующего предварительного установления соединения между участниками обмена информацией. При потере одного или нескольких пакетов все сообщение не будет передаваться повторно — достаточно будет повторить передачу его недоставленных частей, что способствует эффективному использованию пропускной способности канала связи.

Безусловно, такой подход нарушает условие строгой однонаправленности при передаче данных через шлюз, а значит, позволяет построить канал утечки информации. Для его нейтрализации предлагается использовать в протоколе обмена единственный формат пакетов, в котором часть

полей, отвечающих за идентификационную информацию пакета, будет контролироваться аппаратно, остальные поля — программно.

Для предотвращения обратной передачи информации с использованием скрытых каналов передачи данных [2] в разработанном однонаправленном шлюзе предлагается реализовать передачу информации, разделив процесс передачи на два тракта:

«Источник» — «Шлюз»;

«Шлюз» — «Получатель».

При получении блока информации от источника шлюз отправляет пакет-подтверждение, затем передает получателю принятый от источника блок данных. Получатель, в свою очередь, отправляет пакет-подтверждение шлюзу, который в случае отсутствия подтверждения со стороны получателя повторяет передачу неподтвержденных пакетов.

При такой схеме передачи информации формирование пакета-подтверждения, передаваемого источнику, происходит в шлюзе, который является доверенным устройством. При этом момент отправки подтверждения принципиально не может нести никакой информации о времени доставки блока данных получателю, поскольку этот момент еще не наступил. Данные о доставке информации получателю передаются шлюзу, который повторно отправляет непереданные пакеты, попутно производя журналирование и сбор статистики о переданных получателю пакетах.

Для обеспечения гарантии передачи информации предлагается использовать в шлюзе достаточно большой блок энергонезависимой памяти, который позволяет сохранять принятую от источника информацию до полной передачи получателю. Воздействие со стороны получателя на блок памяти шлюза невозможно, поскольку все пакеты от получателя, за исключением пакетов-подтверждений, будут проигнорированы, а пакеты-подтверждения не попадут в блок памяти шлюза согласно протоколу обмена.

Предъявляемые требования к надежности реализации свойства однонаправленности передачи данных обусловливают необходимость принятия специальных инженерных мер по повышению надежности верификации реализации критических алгоритмов и снижению вероятности влияния сбоев и отказов аппаратуры на корректность выполнения критических функций. Для решения обеих задач предлагается использовать аппаратную реализацию критических функций, каковыми являются управление потоками данных и контроль корректности маршрута пакетов данных.

В условиях малой тиражности изделия предлагается сложные электронные устройства реализовывать с использованием программируемых логических интегральных схем (ПЛИС). Важным преимуществом аппаратной реализации является возможность исчерпывающего анализа корректности функционирования, поскольку в этом случае, в отличие от применения универсального процессора, не требуется, во-первых, анализировать влияние избыточных для выполнения требуемой функции аппаратных средств, присутствующих в универсальном процессоре, во-вторых, проводить анализ взаимодействия прикладного программного обеспечения с операционной системой. Требуемые функции реализуются непосредственно электронной схемой, выступающей в качестве специализированного процессора, на который имеется вся необходимая техническая документация. Кроме того, в отличие от универсального процессора, эта документация существенно меньше по объему и, следовательно, в большей степени пригодна для анализа.

Не менее важным преимуществом аппаратной реализации на ПЛИС является возможность дублирования отдельных операций в физически разных элементах ПЛИС. В отличие от дублирования функций в программной реализации, при которой продублированные операции выполняются одними и теми же аппаратными средствами процессора, аппаратное дублирование позволяет обеспечить максимальное увеличение надежности выполнения критических функций, поскольку сами дублирующие схемы, а следовательно, и их отказы являются независимыми.

В то же время реализация на ПЛИС функций обеспечения гарантированной передачи, связанных с реализацией взаимодействия с блоком памяти большой емкости и исполнения протокола гарантированной передачи, представляется чрезмерно сложной для непосредственной реализации на ПЛИС. Учитывая асинхронный характер процесса гарантированной передачи, связанный с необходимостью хранения данных неопределенное время, поддержания сложных структур данных, на практике используемый алгоритм может быть реализован с применением универсального процессора, на котором уже с помощью загружаемой программы будет реализован требуемый алгоритм обработки. Для связи этого процессора с модулем ПЛИС целесообразно использовать интерфейс Ethernet 10 Гбит.

Предложенная авторами архитектура однонаправленного шлюза приведена на рис. 1.

Поток данных, циркулирующих в разрабатываемом шлюзе, представляет собой два независимых подпотока:

- подпоток данных, передаваемых без гарантии передачи (потоковая информация);
- подпоток данных с гарантированной передачей (файловая информация).

Структура потока данных приведена на рис. 2.

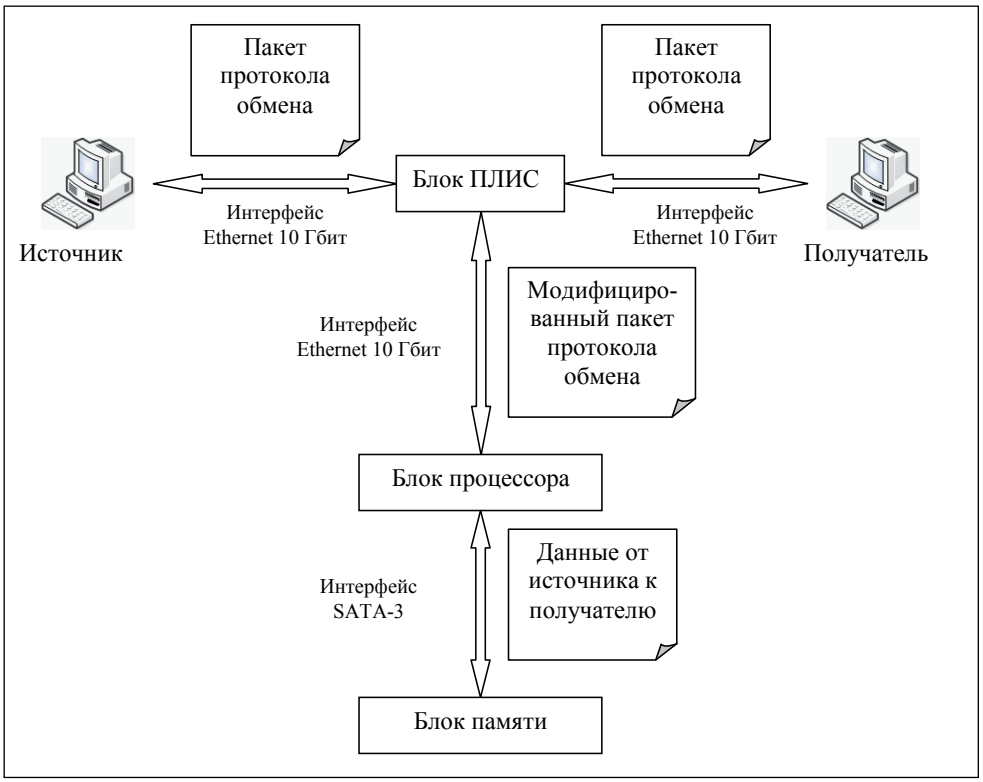

Рис. 1. Архитектура высокоскоростного однонаправленного шлюза

Потоковые данные, поступающие от источника, без изменений транслируются шлюзом получателю с контролем поступающих пакетов на соответствие протоколу обмена.

Фрагменты файлов, поступающие от источника, под управлением микрокода ПЛИС передаются в блок процессора, где сохраняются в блоке памяти, из которого передаются получателю. В процессе обработки выполняется контроль целостности данных и полноты передачи файлов.

Сетевые пакеты, поступающие от получателя, контролируются на соответствие протоколу обмена. Пакеты, соответствующие протоколу обмена, передаются в блок процессора — это подтверждение о получении. Другие пакеты игнорируются.

Для ведения журналов программными компонентами используется энергонезависимая память (жесткий диск, флэш-накопитель и т. п.) источника и получателя, шлюз также ведет журналы работы и использует свой блок энергонезависимой памяти.

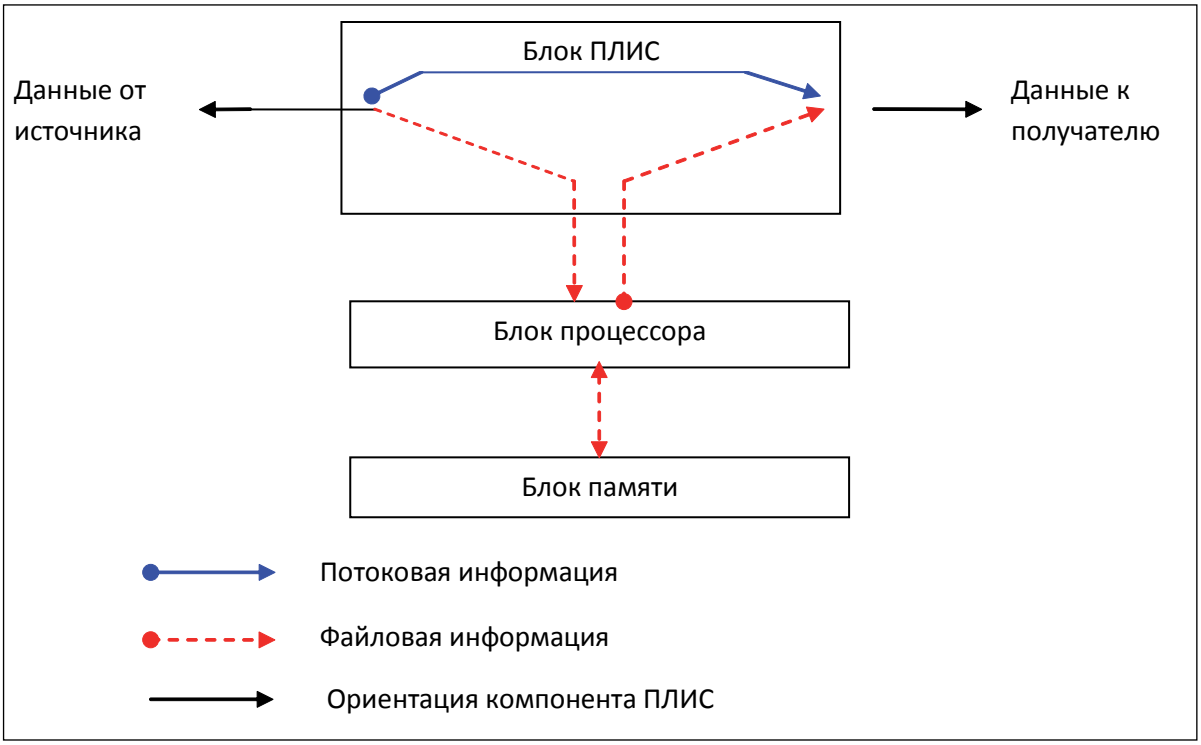

Рис. 2. Структура потоков данных в однонаправленном шлюзе

2. Моделирование взаимодействия участников обмена с однонаправленным шлюзом

Разработанный макет, позволяющий изучить взаимодействие участников обмена с высокоскоростным шлюзом однонаправленной гарантированной передачи данных, представляет собой два клиентских приложения: программный компонент отправки (ПКО) и программный компонент приема (ПКП).

Функции ПКО — фрагментация и инкапсуляция исходных данных в пакеты протокола обмена и их передача от источника к шлюзу.

Функции ПКП — прием, дефрагментация и деинкапсуляция исходных данных из пакетов протокола обмена.

При непосредственной передаче данных от ПКО, размещенного на средствах вычислительной техники (СВТ) источника, к ПКП, размещенному на СВТ получателя, моделируется взаимодействие источника и шлюза. Поскольку взаимодействие в тракте передачи «Шлюз — Получатель» полностью совпадает с взаимодействием в тракте «Источник — Шлюз», достаточно подтвердить эффективность протокола обмена в одном из них.

В силу того что, как правило, конечным клиентом шлюза является обычный персональный компьютер, решено использовать сетевой интерфейс стандарта Ethernet 1 Гбит в качестве интерфейса взаимодействия ПКО и ПКП.

3. Методика тестирования макета однонаправленного шлюза

3.1. Описание программно-аппаратного стенда

Для проведения нагрузочного тестирования предполагается использовать следующую аппаратную конфигурацию СВТ источника и получателя:

- материнская плата с поддержкой сокета Intel s1156;
- процессор Intel® Core™ i7-920 Processor (8M Cache, 2.66 ГГц);
- оперативная память 16 ГБайт (4 x 4 ГБайт) Hynix Original HMT351U6BFR8C-H9N0 (двухканальный режим);
- встроенный сетевой контроллер стандарта 1 Гбит Ethernet;
- монитор, стандартные устройства ввода.

Для подключения СВТ источника и получателя к сетевым интерфейсам шлюза стандарта 10 Гбит Ethernet предполагается использование дополнительного оборудования в составе:
- коммутатор Cisco Catalyst 4900M.

Для проведения нагрузочного тестирования предполагается использовать следующий комплект программного обеспечения, установленного на СВТ источника и получателя:
- операционная система Windows 7 Professional SP1 64-bit;
- набор драйверов и системного программного обеспечения, входящего в комплект используемых аппаратных средств;
- программный компонент отправки (СВТ источника);
- программный компонент приема (СВТ получателя).

В качестве дополнительного и вспомогательного программного обеспечения предполагается использование следующих программных продуктов:
- многофункциональный файловый менеджер Total Commander (версии 8.01 64-bit);
- утилита создания RAM-диска SoftPerfect RAM-Disk (версии 3.3.3 64-bit);
- утилита оценки пропускной способности локальной сети Iperf (версии 2.0.5).

Тестирование программного макета проводилось при подключении источника и получателя согласно схеме, приведенной на рис. 3.

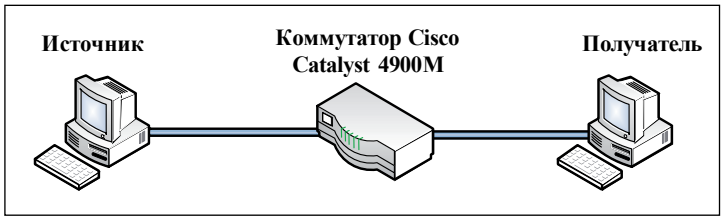

Рис. 3. Схема подключения источника и получателя

3.2. Методика оценки пропускной способности локальной сети

Чтобы оценить пропускную способность локальной сети, на СВТ получателя необходимо запустить на терминале утилиту Iperf в режиме работы сервера. Для этого необходимо выполнить на терминале команду:

Iperf –f m –s.

Затем на СВТ источника следует запустить на терминале утилиту Iperf в режиме работы клиента. Для этого необходимо выполнить на терминале команду:

Iperf –f m –P 1 –c <IP-адрес Получателя>.

IP-адреса источника и получателя при подготовке стенда устанавливаются вручную администратором или автоматически по протоколу DHCP. По завершении оценки фиксируются полученные значения, затем проводится повторная оценка с количеством потоков передачи, равным двум. Для этого на СВТ источника необходимо выполнить на терминале команду:

Iperf –f m –P 2 –c <IP-адрес Получателя>.

Количество потоков передачи следует увеличивать до тех пор, пока не перестанет увеличиваться результат оценки пропускной способности сети. В качестве оценки пропускной способности сети следует принять максимальный из полученных результатов.

3.3. Методика оценки скорости потоковой передачи данных

Для оценки скорости потоковой передачи данных на стороне получателя запускается ПКП с параметрами работы в потоковом режиме. Для этого необходимо выполнить на терминале команду:

pkp.exe –T: 0 –N: 0 –S: <Порт Источника> –R: <Порт Получателя> –IP: <IP-адрес Источника> –Location: <Путь для сохранения файла>.

При запуске ПКП с параметром –T:0 запускается процесс ожидания входных данных файла, имитирующего последовательность пакетов потоковой информации, при этом подтверждение о приеме не отправляется.

Для инициирования процесса передачи файла, имитирующего последовательность пакетов потоковой информации, на стороне источника запускается ПКО с параметрами работы в потоковом режиме. Для этого следует выполнить на терминале команду:

pko.exe –T: 0 –N: 0 –S: <Порт Источника> –R: <Порт Получателя> –IP: <IP-адрес Получателя> –File: <Путь к файлу>.

При запуске ПКО с параметром –T:0 запускается процесс передачи указанного файла в режиме потоковой передачи – программа не ожидает от получателя подтверждения о приеме предыдущих пакетов, а продолжает передачу оставшихся данных файла.

По окончании процесса передачи ПКО уведомляет пользователя о завершении процесса и сообщает среднюю скорость отправки пакетов, затем завершает свою работу. ПКП также сигнализирует о завершении процесса и сообщает объем полученной информации, продолжительность передачи и среднюю скорость приема данных.

3.4. Методика оценки скорости передачи данных с подтверждением

Для оценки скорости передачи данных с подтверждением на стороне получателя запускается ПКП с параметрами работы в режиме подтверждения при приеме. Для этого необходимо выполнить на терминале команду:

pkp.exe –T: 1 –N: 0 –S: <Порт Источника> –R: <Порт Получателя> –IP: <IP-адрес Источника> –Location: <Путь для сохранения файла>.

При запуске ПКП с параметром –T:1 запускается процесс ожидания входного инициализационного пакета, содержащего служебную информацию о передаваемом файле. В этом режиме работы ПКП осуществляет квитирование каждой группы принятых пакетов: пакет-подтверждение содержит информацию о номерах принятых пакетов. Источник должен повторить отправление неподтвержденных пакетов.

Для начала процесса передачи файла с подтверждением на стороне источника запускается ПКО с параметрами работы в режиме подтверждения передачи. Для этого следует выполнить на терминале команду:

pko.exe –T: 1 –N: 0 –S: <Порт Источника> –R: <Порт Получателя> –IP: <IP-адрес Получателя> –File: <Путь к файлу>.

При запуске ПКО с параметром –T:1 запускается процесс передачи указанного файла в режиме с подтверждением передачи – приложение циклично отправляет группу пакетов до подтверждения получателем приема всей группы пакетов или ее части. Источник повторяет неподтвержденные пакеты.

По окончании процесса передачи на стороне источника ПКО выводит информационное сообщение о средней скорости передачи файла и завершает свою работу. На стороне получателя ПКП выводит информационное сообщение о средней скорости приема файла и ожидает инициализационного пакета, свидетельствующего о начале процесса передачи следующего файла.

3.5. Методика проверки целостности передаваемых данных с подтверждением

Для проверки целостности передаваемых данных перед началом процесса передачи на стороне источника вычисляются контрольные суммы передаваемого файла согласно алгоритмам MD5 [3] и SHA-1 [4].

По окончании процесса передачи на стороне получателя вычисляются контрольные суммы принятого файла согласно алгоритмам MD5 и SHA-1, затем сравниваются со значениями, ранее вычисленными источником.

3.6. Методика оценки количества утраченных или искаженных байт при передаче в потоковом режиме

Оценка количества утраченных в процессе передачи байт данных определяется сравнением количества отправленных байт и фактически принятых ПКП, что отражается в журнале работы. На основании полученной информации вычисляется процент потерь.

Если процент потерь равен нулю, уместно проведение анализа искажения байт в процессе передачи. Для этого на СВТ получателя переносится исходный файл способом, отличным от передачи с использованием ПКО и ПКП. С помощью утилиты Total Commander проводится сравнение оригинального и принятого файлов путем вызова через контекстное меню программы команды «Файлы\Сравнить по содержимому».

По окончании сравнения за количество искаженных байт принимается значение параметра «Обнаружено отличий». Сообщение «Эти файлы одинаковы» свидетельствует о передаче данных без потерь и искажений.

3.7. Методика оценки избыточности сетевого трафика при передаче данных с подтверждением

Оценка количества повторно отправленных пакетов при передаче данных с подтверждением проводится методом анализа журналов работы ПКО на стороне источника.

По окончании процесса приема-передачи ПКО информирует пользователя о количестве реально отправленных пакетов и минимальном количестве пакетов, на которые может быть фрагментирован файл для передачи по протоколу однонаправленного обмена.

Пусть M — минимальное количество пакетов, на которые может быть фрагментирован файл, K — число реально отправленных ПКО пакетов. Избыточность данных при передаче данных с подтверждением R определяется по формуле:

$$R = 100\frac{K - M}{K} \qquad (1)$$

Вычисленное значение характеризует процент избыточного сетевого трафика в составе общего трафика, переданного источником.

3.8. Общие рекомендации по проведению тестирования

Для достижения максимального результата скорости передачи рекомендуется выполнить следующие действия:

- приостановить работу фоновых сетевых служб и программ, не оказывающих влияние на передачу и прием данных по сети;
- расположить отправляемый ПКО файл и целевой каталог ПКП для сохранения данных на виртуальном RAM-диске, который может быть создан с использованием утилиты SoftPerfect RAM-Disk.

Использование RAM-диска может существенно улучшить результаты тестирования, поскольку в этом случае исключаются задержки, возникающие при чтении и записи данных на жесткие диски источника и получателя соответственно.

4. Критерии оценки эффективности протокола однонаправленного обмена и методика их расчета

Под эффективностью протокола однонаправленного обмена в дальнейшем будем понимать продуктивность использования ресурсов локальной вычислительной сети в процессе приема-передачи информации.

4.1. Эффективность использования пропускной способности сети при потоковой передаче данных

Пусть S_s — скорость потоковой передачи (Мбит/с), B — пропускная способность локальной сети (Мбит/с), измеренные в соответствии с рассмотренной выше методикой тестирования.

Тогда эффективность использования пропускной способности локальной вычислительной сети (ЛВС) в процессе потоковой передачи данных E_s вычисляется по следующей формуле:

$$E_s = 100\frac{S_s}{B} \tag{2}$$

Получаемое значение показывает степень использования пропускной способности ЛВС при передаче данных в потоковом режиме, выраженную в процентах.

4.2. Эффективность использования пропускной способности ЛВС при передаче данных с подтверждением

Пусть S_c — скорость передачи данных с подтверждением (Мбит/с), B — пропускная способность локальной сети (Мбит/с), измеренные в соответствии с рассмотренной выше методикой тестирования.

Тогда эффективность использования пропускной способности ЛВС в процессе передачи данных с подтверждением E_c вычисляется по следующей формуле:

$$E_c = 100\frac{S_c}{B} \tag{3}$$

Получаемое значение показывает степень использования пропускной способности ЛВС при передаче данных в режиме с подтверждением, выраженную в процентах.

4.3. Показатель достоверности передачи данных в режиме с подтверждением

Пусть T_c — количество достоверно переданных файлов в режиме передачи с подтверждением, измеренное в соответствии с рассмотренной выше методикой тестирования, F_c — общее количество передаваемых файлов в режиме передачи с подтверждением.

Тогда показатель достоверности передачи данных в режиме с подтверждением A_c вычисляется по следующей формуле:

$$A_c = 100\frac{T_c}{F_c} \tag{4}$$

Получаемое значение показывает долю достоверно переданных файлов от общего числа файлов, передаваемых в режиме с подтверждением.

4.4. Показатель полноты передачи данных в потоковом режиме

Пусть P_r — количество принятых ПКП пакетов в потоковом режиме передачи, P_s — количество отправленных ПКО пакетов в потоковом режиме передачи.

Тогда показатель полноты передачи данных в потоковом режиме A_s рассчитывается по следующей формуле:

$$A_s = 100\frac{P_r}{P_s} \tag{5}$$

Значения величин P_r и P_s устанавливаются согласно приведенной выше методике тестирования.

Значение показателя полноты передачи данных в потоковом режиме показывает долю принятых ПКП пакетов от общего числа отправленных ПКО в потоковом режиме.

5. Результаты тестирования программного макета и анализ эффективности протокола обмена

В настоящем разделе приведены результаты нагрузочного тестирования разработанного программного макета и дан анализ показателей эффективности протокола обмена.

5.1. Результаты тестирования программного макета и показатели эффективности протокола обмена

Опишем результаты тестирования программного макета, проведенного в соответствии с рассмотренной выше методикой. В процессе выполнения ряда экспериментов получено, что пропускная способность локальной сети составляет 108,12 Мбайт/с. В таблицах 1 и 2 приведены основные результаты, полученные при тестировании программного макета однонаправленного шлюза в потоковом режиме передачи данных и в режиме с подтверждением соответственно.

Таблица 1. Результаты тестирования программного макета в потоковом режиме передачи данных

Размер файла (Гбайт)	Скорость передачи (Мбайт/с)	Средняя скорость передачи (Мбайт/с)	Процент потерь (%)
1	101,23	100,86	0,05
2	101,98		0,08
4	102,23		0,07
6	100,79		0,12
8	98,60		0,12
10	100,31		0,14

Таблица 2. Результаты тестирования программного макета в режиме передачи данных с подтверждением

Размер файла (Гбайт)	Скорость передачи (Мбайт/с)	Средняя скорость передачи (Мбайт/с)	Процент избыточного сетевого трафика (%)
1	101,15	102,76	0,04
2	103,40		0,03
4	101,94		0,04
6	101,39		0,04
8	106,12		0,01
10	102,54		0,03

В таблице 3 приведены результаты расчетов значений критериев эффективности на основании результатов тестирования.

Таблица 3. Результаты расчетов значений критериев эффективности протокола обмена

Размер файла (Гбайт)	E_s	E_c	A_s
1	93,62	93,55	99,95
2	94,32	95,63	99,92
4	94,55	94,28	99,93
6	93,22	93,77	99,88
8	91,19	98,15	99,88
10	92,77	94,84	99,86

Для шести переданных файлов указанных в таблицах размеров показатель достоверности передачи данных в режиме с подтверждением $A_c = 100$, то есть достоверно были переданы все файлы.

5.2. Анализ результатов тестирования и показателей эффективности

Средняя скорость передачи данных в потоковом режиме и в режиме с подтверждением практически идентична и составляет более 800 Мбит/с, что почти на порядок превышает показатели используемых на сегодняшний день решений.

Ненулевой процент потерь в потоковом режиме передачи при всех размерах передаваемого файла обусловлен, в первую очередь, предельным режимом работы программного макета и сетевого стека операционной системы. Доля потерь до 0,15–0,20 % является приемлемой для передачи видеоинформации в реальном масштабе времени, что доказывает возможность практического применения предложенного протокола однонаправленного обмена.

В режиме подтверждения передачи были получены результаты стопроцентной достоверности передаваемых данных и низкий процент избыточного трафика (0,04%), не превышающий уровень потерь при передаче данных в потоковом режиме. Полученный результат свидетельствует о высокой эффективности протокола обмена в режиме с подтверждением передачи.

Эффективность использования пропускной способности сети в обоих режимах работы не опускается ниже уровня в 91 %, что вполне приемлемо.

На основании полученных результатов тестирования и расчета показателей эффективности протокола обмена можно сделать положительное заключение о возможности его практического применения при использовании высокоскоростного однонаправленного шлюза.

Заключение

Для снижения уровня потерь и повышения показателей эффективности протокола обмена предлагается выполнить снижение максимальной скорости передачи в пределах нескольких процентов. Это может существенно улучшить показатель потерь в потоковом режиме, поскольку в этом случае снижается нагрузка на сетевой стек операционной системы и процесс передачи данных становится более устойчивым к случайным труднопрогнозируемым задержкам в сети.

СПИСОК ЛИТЕРАТУРЫ:

1. *Архангельская А. В., Архангельский В. Г., Калмыков В. В.* Шлюз однонаправленной гарантированной передачи данных // Проблемы информационной безопасности. Компьютерные системы. 2013. № 4. С. 27–39.
2. *Архангельский В. Г., Лосев С. А.* О противодействии новым угрозам при использовании однонаправленных шлюзов // Информатизация и связь. 2012. № 8. С. 194–196.
3. RFC1321 — The MD5 Message-Digest Algorithm. URL: http://tools.ietf.org/html/rfc1321 (дата обращения: 14.05.2014).
4. RFC3174 — US Secure Hash Algorithm 1 (SHA1). URL: http://tools.ietf.org/html/rfc3174 (дата обращения: 14.05.2014).

REFERENCES:

1. *Arkhangelskaya A. V., Arkhangelskiy V. G., Kalmykov V. V.* The secure one-way data transfer // Information security problems. Computer systems. 2013. № 4. P. 27–39.
2. *Arkhangelskiy V. G., Losev S. A.* About counteractions to new threats when using unidirectional gateway // Informatization and communication. 2012. № 8. P. 194–196.
3. RFC1321 — The MD5 Message-Digest Algorithm. URL: http://tools.ietf.org/html/rfc1321 (date of request 14.05.2014).
4. RFC3174 — US Secure Hash Algorithm 1 (SHA1). URL: http://tools.ietf.org/html/rfc3174 (date of request 14.05.2014).