БЕЗОПАСНОСТЬ ПОДКЛЮЧЕНИЯ СЕТЕВЫХ ВИДЕОКАМЕР

Р.А.БРЫЛУНОВ
Пермский государственный национальный исследовательский университет, 614990, Пермь, Букирева, 15

Видеонаблюдение – это процесс фиксирования, передачи и отображения видеоинформации. Оно является достаточно значимой частью систем обеспечения безопасности и позволяет осуществлять визуальный контроль и автоматический анализ изображений. Исторически, первыми появились аналоговые системы видеонаблюдения, но, по мере развития технологий, происходил постепенный переход от аналогового сигнала к цифровому. Результатом такого развития сегодня является сетевое видеонаблюдение, в котором для передачи информации используются компьютерные сети.

Главная функция сетевых видеокамер – передача потока видеоинформации по сети на некоторое конечное устройство, где будет производиться просмотр, запись или анализ полученной информации. Для того, что бы начать получать информацию от камеры, необходимо установить с ней соединение. Чаще всего, на современных сетевых видеокамерах имеется собственный веб-сервер, поэтому одним из способов установления соединения является подключение к камере с помощью веб-браузера. В этом случае обычно используется протокол HTTP. Кроме того получить видеопоток можно с использованием специализированных протоколов. Одним из таких протоколов является протокол транспортного уровня RTP (Real-time Transport Protocol). Но протокол RTP осуществляет только передачу потока, а для управления передачей в связке с ним должен использоваться некоторый протокол управления. Стандартом для протокола RTP является протокол RTCP, но большую популярность так же имеет протокол RTSP. Протокол RTP может базироваться либо на протоколе TCP, либо на протоколе UDP, но, т.к. потеря видеоинформации нежелательна, предпочтение отдается RTP на базе TCP. Перечисленные протоколы не предполагают какой-либо защиты кроме аутентификации клиента (того, кто подключается к камере), т.е. оставляют возможными атаки прослушивания и подмены потока видеоинформации.

С целью увеличения безопасности, большинство сетевых камер поддерживают контроль доступа с разделением прав пользователей. Самым простым способом обеспечения контроля является парольная

аутентификация. Этот способ реализован практически во всех камерах, и именно он используется чаще всего. Аутентификации при этом подвергается клиент (т.е. ПО видеонаблюдения), сама же сетевая камера не проверяется, что позволяет подменять или прослушивать поток, отдаваемый видеокамерой, или даже подменить само устройство. Существует вариант аутентификации сетевых видеокамер с использованием стандарта IEEE 802.1X, но такой подход не позволяет защититься от атак типа «человек посередине» (man in the middle) или прослушивания. Другим вариантом является протокол PPPoE, позволяющий осуществлять аутентификацию и шифрование данных, но камеры обычно поддерживают только возможность аутентификации.

Существуют и более защищенные варианты подключения к камере. Одним из самых популярных протоколов, используемых для создания защищенного соединения, является протокол HTTPS. В самом простом варианте он позволяет производить аутентификацию сервера (т.е. в нашем случае сетевой видеокамеры) с помощью цифрового сертификата. Кроме того может использоваться шифрование данных и контроль целостности для защиты. Протокол HTTPS поддерживается достаточно большим количеством сетевых видеокамер, но он обычно используется только при подключении с помощью веб-браузера, и не используется в системах видеонаблюдения.

Для протоколов RTP/RTCP существует защищенная реализация - протоколы SRTP и SRTCP соответственно, но данная связка защищенных протоколов практически не используется в сетевых видеокамерах.

Еще одним вариантом повышения безопасности подключения является создание защищенного VPN канала. В этом случае сетевая камера выступает в качестве VPN клиента и устанавливает подключение к серверу VPN. После установления подключения, между камерой и сервером формируется защищенный канал, вся информация, по которому передается в зашифрованном виде. Проверке при этом подвергаются как сервер, так и клиент. Кроме того, такой подход позволяет избежать проблем с подключением к сетевой камере в случае использования механизма преобразования адресов (NAT) на стороне камеры.

Главной проблемой в реализации идеи использования VPN является отсутствие поддержки технологии VPN в большинстве камер. Эту проблему можно решить, добавив между сетевой видеокамерой и сервером видеонаблюдения устройство, которое будет выполнять функцию клиента VPN. При этом необходимо максимально сократить расстояние между этим устройством и камерой, вплоть до расположения

в одном внешнем корпусе. Такой подход позволит использовать простые камеры без поддержки VPN, или модернизировать существующую структуру сетевого видеонаблюдения, защитив при этом все подключения к камерам. Пример использования подобного устройства представлен на рис. 1.

Рис. 1. *Схема сети в случае использования выделенного клиента VPN*

Может сложиться такая ситуация, что в системе видеонаблюдения используются камеры большого разрешения, или просто большое количество камер. При этом поток информации, создаваемый этими камерами, может оказаться очень большим, и для его централизованного дешифрования может понадобиться очень большое количество вычислительных ресурсов. В этом случае можно добавить несколько устройств на границе контролируемой зоны, которые будут выполнять функции сервера VPN, осуществляя только шифрование/дешифрование информации, и снимая нагрузку с основных серверов. Вариант с использованием двух выделенных устройств для создания VPN-туннеля представлен на рис. 2.

Рис. 2. *Схема сети в случае использования выделенного клиента и выделенного сервера VPN*

Сегодня все большее распространение приобретают микропроцессоры

на архитектуре ARM. Они отличаются небольшим энергопотреблением и тепловыделением, обладая при этом достаточной вычислительной мощностью. Одной из целей данной работы являлась оценка возможности использования устройства на основе ARM-микропроцессора в качестве шифратора/дешифратора информации, передаваемой по сети, т.е. проверка применимости в качестве устройства, описанного выше.

В качестве тестового образца использовался микрокомпьютер Raspberry Pi Model B, обладающий процессором на базе архитектуры ARM 11 с тактовой частотой 700МГц и 512 МБ оперативной памяти. На микрокомпьютер была установлена ОС Raspbian – специальная версиея Debian Linux.

Тестовый стенд состоял из клиента VPN, роль которого выполнял микрокомпьютер Raspberry Pi, и сервера VPN, в качестве которого использовался ПК с установленной ОС Ubuntu 12.10 x32. Работу виртуальной частной сети обеспечивало ПО OpenVPN, являющееся свободной реализацией с открытым исходным кодом. Сервер и клиент VPN в ходе эксперимента были соединены напрямую сетевым кабелем.

Целью эксперимента было определение максимальной скорости, которую может поддерживать RaspberryPi в качестве клиента VPN, а так же вносимой при этом задержки. Задержка, создаваемая при передаче по сети, является достаточно важным фактором в сетевом видеонаблюдении, т.к. она влияет на разницу во времени между тем, что происходит на самом деле в поле зрения камеры, и тем, что отображается на экране оператора видеонаблюдения или в архиве.

Во время тестов были проверены 2 серверные конфигурации, в которых использовался протокол транспортного уровня TCP и UDP для сравнения. В обоих случаях использовался алгоритм шифрования AES-128-CBC и ip-туннель. Для измерения скорости сети использовалась утилита *iperf*, а для измерения задержки утилита *ping*.

В ходе тестов были получены следующие результаты (представлены усредненные результаты):

1. Скорость передачи по сети:
   - Протокол TCP – 13,5 Мбит/с;
   - Протокол UDP – 14,1 Мбит/с.
2. Увеличение задержки при передаче по сети, вносимое VPN:
   - Протокол TCP – 18,2 мс;
   - Протокол UDP – 15,7 мс.

Результаты измерения оказались достаточно закономерными: шифрование вносит некоторую дополнительную задержку, при этом задержка, добавляемая при использовании протокола TCP, превышает задержку при использовании протокола UDP. Максимальная скорость выше для протокола UDP. С учетом факта, что для передачи потока видеоинформации от сетевых видеокамер чаще всего используется протокол TCP, использование VPN-туннеля с использованием протокола UDP является предпочтительным.

Величина потока видеоинформации, создаваемого камерой с разрешением 1 Mpix при использовании формата MJPEG составляет около 12 Мбит/с, а при использовании формата H.264 около 2-3 Мбит/с. Поэтому можно сделать вывод, что вычислительных ресурсов Raspberry Pi достаточно для использования в качестве шифратора/дешифратора в паре с такой камерой. Величина добавляемой задержки так же не является критической. Стоит так же учесть, что вычислительные ресурсы Raspberry Pi все же не очень велики и микрокомпьютер не содержит модулей аппаратного ускорения шифрования, а значит можно достичь и более высоких скоростей и меньших задержек.

Подводя итоги, получаем следующее:

1. Чаще всего безопасность подключения сетевых видеокамер находится на достаточно низком уровне;
2. Одним из простых способов повышения безопасности может быть использование VPN;
3. Вычислительных ресурсов небольшого ARM микрокомпьютера достаточно для выполнения функций шифрования/дешифрования и организации VPN.

СПИСОК ЛИТЕРАТУРЫ

1. *Нильссон Ф.* Энциклопедия сетевого видеонаблюдения. Пер. с англ. – М.: ООО «Ай-Эс-Эс Пресс», 2011 – 404 с.;
2. RPi Easy SD Card Setup [http://elinux.org/RPi_Easy_SD_Card_Setup]/ Embedded Linux Wiki – режим доступа http://elinux.org/Main_Page свободный;
3. OpenVPN [https://help.ubuntu.com/11.10/serverguide/openvpn.html]/ Official Ubuntu Documentation – режим доступа https://help.ubuntu.com/ свободный;
4. Тестирование скорости локальной сети Ethernet [http://bloggik.net/index.php/articles/networks/29-others/112-ethernet-test-speed]/ Bloggik.net – режим доступа http://bloggik.net/index.php/home свободный.