Тюнинг RDP может значительно расширить функционал работы и ускорить доступ к терминальным серверам. Разбираем, как именно.

atraining.ru/windows-rdp-tuning

2013-10-05T07:03:54+08:00

Привет. Вчера, общаясь с Иваном Никитиным, получил дельный совет осветить работу и настройку протокола RDP. Мысль дельная, дальше – подробнее.

Введение

Протокол RDP – удобное, эффективное и практичное средство для удалённого доступа как для целей администрирования, так и для повседневной работы. Учитывая, что его реализации есть практически везде (различные платформы и ОС), и их много, нужно хорошо представлять его возможности. По крайней мере, это будет нужно по ряду причин:

- Зачастую вместо RDP используется другое решение (VNC, Citrix ICA) по простой причине – предполагается, что “встроенный RDP минимальный и ничего не умеет”.
- Во многих решениях, связанных с модными сейчас облачными технологиями (перевод офисов на “тонкие клиенты”, да и просто организация терминальных серверов), бытует мнение что “RDP плохой потому что встроенный”.
- Есть стандартный миф про то, что “RDP нельзя без VPN наружу выставлять, ломанут” (миф имеет под собой обоснование, но уже давно не актуален).
- Ну, раз уж про мифы заговорили – бытует мнение, что “Перейдя с RDP на Citrix трафик в пару раз падает”. Ведь цитрикс – это дорого, следовательно как минимум на 157% круче.

Все эти мифы – ерунда и смесь устаревших “дельных советов”, актуальных во времена NT 4.0, а так же откровенных вымыслов, не имеющих никаких причин к существованию. Так как IT – это точная наука, надо разобраться. Хорошо настроеный протокол RDP новых версий, с учётом всех новых функциональных возможностей, является достаточно хорошим и надёжным инструментом для организации удалённого доступа. Поэтому мы займёмся:

- Кратким упоминанием про версии RDP
- Настройкой режима защиты RDP-сессии
- Настройкой шифрования для RDP
- Привязкой к конкретному адаптеру и порту
    - Меняем стандартный порт на нужный
    - Делаем раздельные настройки RDP для нескольких сетевых адаптеров

- Включением NLA
  - Как включается NLA со стороны RDP-сервера
  - NLA и Windows XP
  - Как включить CredSSP в XP
- Выбором правильного сертификата для RDP
- Блокированием подключений по RDP учётным записям с пустым паролем
- Настройка ACL для подключения по RDP
- Оптимизацией скорости RDP
  - Отключаем редирект неиспользуемых устройств
  - Настраиваем общую логику оптимизации визуальных данных RDP
- Оптимизацией сжатия RDP
  - Настраиваем общее сжатие RDP
  - Настраиваем сжатие аудиопотока RDP
- Оптимизацией соотношения потоков данных RDP
- Включением Require secure RPC communication для RDP

Приступим.

Версии протокола RDP

Протокол имеет достаточно длительную историю, начиная с NT 4.0. Исторические детали мы оставим в стороне по простой причине – на данный момент имеет смысл говорить только про версию RDP 7.0, которая есть в Windows Vista SP1 / Windows Server 2008 и бесплатно добавляема в Windows XP установкой SP3 и обновлённого клиента RDP (находится по ссылке на KB 969084). Я предполагаю, что у Вас как минимум Windows XP, и что Вы поставили/можете поставить последний Service Pack и не трачу Ваше время на обсуждение преимуществ RDP в Windows 2000 SP2 перед NT 4.0 SP5.

Настройка режима защиты RDP-сессии

В принципе, это самая простая часть задачи. Суть в следующем. В различных версиях RDP применяется два основных механизма защиты сессии – встроенный в RDP и “заворачивание” сессии в TLS. Встроенный является недостаточно безопасным, и рекомендация “RDP можно наружу только в VPN” – про него. Поэтому всегда включайте поддержку TLS. Это тот минимум, с которого Вы должны начать. Ограничениями будут разве что версия сервера не ниже Windows Server 2003 SP1 и клиент RDP 5.2 и выше, но, думается, это в конце 2011 года вполне решаемо.

Как включить RDP over TLS

Вариантов, как всегда, несколько. Первый – включение через групповую политику. Для этого надо зайти в целевой объект групповой политики (ну или локально на своей домашней рабочей станции запустить `gpedit.msc`) и там последовательно выбрать “Computer Configuration” -> “Administrative Templates” -> “Windows

Components” -> “Remote Desktop Session Host” -> “Security” и там включить параметр Require use of specific security layer for remote connections, выбрав в нём `SSL (TLS 1.0) only`. Можно выбрать и более мягкий `Negotiate`, но я бы не рекомендовал, т.к. на данный момент это банально ниже приемлемого уровня безопасности. Как человек, создававший private cloud’ы с достаточно высоким уровнем безопасности, я могу сказать, что смысл выносить особо ценные данные в датацентр под Лондоном и ходить туда дефолтным RDP – нулевой и является поиском неприятностей. Можно и проще – откройте оснастку Remote Desktop Session Host Configuration (найдёте в mmc или готовую в меню Administrative Tools -> Remote Desktop Connections), выберите из списка Connections нужное подключение (обычно оно одно и называется RDP-Tcp), и откройте Properties, после – вкладку General и там выбрать нужный Security Layer. Для работы TLS необходим цифровой сертификат (как минимум – со стороны сервера). Обычно он уже есть (генерится автоматически), убедитесь в его наличии, про то, как сделать его хорошим, поговорим после. Пока надо, чтобы он просто был, иначе подключиться не получится.

Настраиваем шифрование для RDP

Для конфигурирования будет доступно 4 варианта шифрования. Рассмотрим каждый из них.

Режим RDP Low Encryption

Самый “никакой” режим. Наследие страшных времён и версий RDP 5.x. Может согласовать шифрование на базе 56ти битового DES или 40ка битового RC2, что на текущий момент является несерьёзным. Не нужен и опасен. Например, если включить его, то не включится TLS, т.к. TLS уже откажется согласовывать такие слабые шифры, которые предлагает этот вариант.

Режим RDP Client Compatible Encryption

Второй “никакой” режим. Наследие страшных времён и версий RDP 5.x. Попробует до 128 бит RC4, но сразу согласится на DES/RC2. Не нужен и опасен. Тоже не совместим с TLS.

Режим RDP High Encryption

Минимально допустимый режим. Потребует хотя бы 128ми битовый RC4. Работает со всеми серверами, начиная с Windows 2000 Server w/HEP.

Режим RDP FIPS140-1 Encryption

То, что нужно. Будет поддерживать современные симметричные алгоритмы и в явном виде не будет поддерживать RC2, RC4, одиночный DES, а также будет заставлять использовать для вычисления целостности (Message Authentication Code – MAC) алгоритм SHA-1, а не MD5. Включайте этот вариант всегда, найти сервер,

который не умеет 3DES, AES или SHA-1 практически нереально. Где делается эта настройка? Откройте оснастку Remote Desktop Session Host Configuration (найдёте в mmc или готовую в меню Administrative Tools -> Remote Desktop Connections), выберите из списка Connections нужное подключение (обычно оно одно и называется RDP-Tcp), и откройте Properties, после – вкладку General и там выберите нужный Encryption Level.

Привязываем RDP к конкретному адаптеру и порту

Для того, чтобы сервер работал безопасно и предсказуемо (например, не начинал принимать подключения с нового, свежедобавленного сетевого адаптера), необходимо в явном виде указать, на каких интерфейсах служба RDP-сервера должна принимать подключения. Плюс, достаточно часто бывает полезным переключить порт, на котором сервер слушает подключения. Конечно, можно это сделать и публикуя сервер с RDP через какой-нибудь шлюз, но можно и без этого. Такие, казалось бы, базовые действия в реальности ощутимо снизят процент дураков-скрипткиддисов, которые очередной “мощной тулзой” проверяют wellknown-порты.

Как привязать службу RDP к конкретному сетевому адаптеру или сделать несколько RDP с разными настройками для разных адаптеров

Откройте оснастку Remote Desktop Session Host Configuration (найдёте в mmc или готовую в меню Administrative Tools -> Remote Desktop Connections), выберите из списка Connections нужное подключение (обычно оно одно и называется RDP-Tcp), и откройте Properties, после – вкладку Network Interfaces. В ней Вы сможете выбрать один конкретный интерфейс, на котором надо ожидать подключения, плюс ограничить количество параллельных сессий. Если у Вас много интерфейсов, и Вам надо, допустим, чтобы можно было подключаться через 2 из 5 доступных, то Вам надо будет привязать существующий по-умолчанию RDP-Tcp к одному адаптеру, после зайти в меню Action и там выбрать Create New Connection. Подключение может слушать либо на всех интерфейсах, либо на одном, и в случае, когда надо, чтобы оно слушало на N интерфейсах, придётся создать N подключений. Соответственно, если у Вас есть задача “Чтобы на одном интерфейсе RDP слушал на одном порту, а на другом – на другом”, она решаема так же – отвязываете дефолтный `RDP-Tcp` от всех адаптеров и привязываете к конкретному, после – создаёте новое RDP-подключение и тоже привязываете к нужному сетевому интерфейсу.

Как привязать службу RDP к не-дефолтному порту

Порт по умолчанию – 3389 TCP. Кстати, не забудьте разрешить его в пакетном фильтре. Ну а если хотите другой – надо зайти в ключ реестра `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Terminal Server\WinStations\RDP-Tcp` и поправить в нём значение `PortNumber`. Учитывайте,

что отслеживание конфликтов в плане занятости портов – на Вашей совести, сам он, обнаружив, что назначенный Вами порт занят, “перепрыгнуть” никуда не сможет.

Включаем NLA – Network Level Authentication

Функция NLA появляется в NT 6.0, а позже добавляется возможность её частичного использования в предыдущей версии ОС путём установки SP3 для XP. Суть данной функции достаточно проста. В версиях RDP до 6.0 при подключении по RDP клиенту до аутентификации надо показать окно входа – т.е. вначале показать, а потом уже он попробует зайти в систему. Это создаёт простую уязвимость – сервер можно перегрузить пачкой запросов “а дай-ка мне попробовать новую сессию начать”, и он будет вынужден на все запросы отвечать созданием сессии и ожиданием входа пользователя. Фактически, это возможность DoS. Как с этим можно бороться? Логично, что надо придумать схему, целью которой будет как можно раньше запросить у клиента учётные данные. Оптимально – чтобы было что-то типа как kerberos в домене. Это и было сделано. NLA решает две задачи:

- Клиент аутентифицируется до инициации терминальной сессии.
- Появляется возможность передать данные локального клиентского SSP на сервер, т.е. начинает работать Single Sign-On.

Реализуется это через новый провайдер безопасности – CredSSP. Почитать его техническую спецификацию можно тут, ну, а говоря проще, надо всегда включать данную функцию. Конечно, учитывая, что для её работы нужно, чтобы:

- Клиентская ОС (та, с которой идёт подключение) была Windows XP SP3 и выше.
- Серверная ОС (та, к которой будет подключение) была Windows Server 2008 и выше.

Примечание: Несмотря на то, что ядро Windows Server 2003 новее, чем в XP (5.2 против 5.1), для Windows XP есть обновление, добавляющее поддержку NLA, а для Windows Server 2003 – нет. То есть, если Вы даже будете подключаться с самой максимально доступной версии – Windows Server 2003 R2 SP2 со всеми патчами, Вы не сможете подключиться к серверу, требующему NLA, и быть сервером, поддерживающим NLA. Увы.

Как включается NLA со стороны RDP-сервера

Лучше всего включить NLA на всех серверах через групповую политику. Для этого надо зайти в целевой объект групповой политики и там последовательно выбрать “Computer Configuration” -> “Administrative Templates” -> “Windows Components” -> “Remote Desktop Session Host” -> “Security” и там включить параметр Require user authentication for remote connections by using Network Layer Authentication. Можно включить и локально. Это делается путём вызова подменю Properties (стандартное подменю у Computer) и выбора там вкладки Remote, в которой будет выбор из трёх вариантов – запрещать подключения по RDP к данному хосту, разрешать

подключения по любому RDP, разрешать только с NLA. Всегда включайте вариант с NLA, это в первую очередь защищает сервер.

NLA и Windows XP

В случае, если у Вас Windows XP, то Вы также можете воспользоваться данной функцией. Распространённое утверждение “Для NLA нужна как минимум виста, это Microsoft сделал чтобы апгрейдились” ложно. В Service Pack 3 добавляется реализация CredSSP, позволяющая делегировать клиентские credentials’ы, которыми обладает местный SSP, на сервер. Т.е., говоря проще, это специально сделано, чтобы с Windows XP можно было подключаться на системы с NT 6.0+. На саму Windows XP SP3 с данной функцией подключаться не получится, поддержка NLA будет частичной (поэтому RDP сервер с поддержкой подключения клиентов с использованием NLA из Windows XP сделать штатными способами не получится, Windows XP будет только NLA-совместимым клиентом).
Примечание: NLA появляется с NT 6.0, и является частью пачки технологий, называемых RDP 6.0. 3й сервиспак для XP приносит не просто RDP 6.0, а возможность установки RDP 7.0, что является достаточно позитивным (например, в RDP 7.0 есть, в отличии от 6.0, EasyPrint, bidirectional audio и некоторые другие штуки, которые превращают RDP-клиента на Windows XP со всеми накрутками в достаточно практичную систему). Это к слову о плохом Microsoft, который так жутко всех заставлял апгрейдиться с Windows XP на плохую-преплохую висту, что аж в бесплатном сервиспаке для продукта 2001 года вшивал более новую подсистему RDP, чем та, которая шла в висте, вышедшей в 2006.

Включать данный функционал нужно в явном виде, так как несмотря на то, что Service Pack 3 добавляет приносит новую dll криптопровайдера, он её не включает.

Как включить CredSSP в XP

Ещё раз – данная операция проводится строго **после** установки Service Pack 3 на Windows XP и в контексте нашего разговора нужна для того, чтобы было возможно подключение к другим серверам по RDP 6.1 с использованием NLA. Шаг первый – расширяем перечень Security Packages. Для этого мы откроем ключ реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa` и найдём в нём значение `Security Packages`. Нажмём правую кнопку и выберем “Modify” (не Modify Binary Data, а просто Modify). Там будет список вида “название package на каждой строке”. Нам надо добавить туда `tspkg`. Остальное надо оставить. Место добавления некритично. Второй шаг – подцепляем библиотеку. Ключ будет другим: `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders` В нём надо будет найти значение `SecurityProviders` (заметьте, как и в предыдущем случае, это не subkey, а значение), и модифицировать его по аналогии, только добавив `credssp.dll`. Остальное в списке, опять же, трогать не надо. Теперь редактор реестра можно закрыть. После этих операций систему надо будет обязательно перезагрузить, т.к. криптопровайдеры – штука такая, которая на ходу точно не подцепится, и это скорее хорошо, чем плохо.

Выбираем правильный сертификат для RDP

Если у Вас есть возможность пользоваться не-дефолтным сертификатом для RDP, то лучше пользоваться именно им. Это не повлияет на безопасность сессии как таковой, но повлияет на безопасность и удобство подключения. В сертификате, который оптимально использовать, должны быть следующие момент:

- Имя (в subject или SAN), посимвольно совпадающее с тем именем, которое вводит клиент, подключающийся к серверу.
- Нормальное расширение CDP, указывающее на рабочий CRL (желательно хотя бы на два – OCSP и статический).
- Желательный размер ключа – 2048 бит. Можно и больше, но помните об ограничениях CAPI2 в XP/2003.
- Не экспериментируйте с алгоритмами подписи/хэширования, если Вам нужны подключения со стороны XP/2003. Чуть больше информации про это в статье про настройку TLS, но вкратце – выберите SHA-1, этого вполне достаточно.

Чуть подробнее остановлюсь на выпуске специального сертификата для RDP-сервера.

Делаем всё красиво – специальный шаблон сертификата для RDP-серверов

Идеально будет, если сертификат для RDP сделать не на основе обычного шаблона (типа Web Server) и иметь в поле `Application Policy` (которое в сертификате будет более привычно называться Enchanced Key Usage – EKU) стандартные значения `Client Authentication` и `Server Authentication`, а добавить свой шаблон, в котором будет единственное, специальное, не добавляемое стандартными способами значение применения – `Remote Desktop Authentication`. Это значение Application Policy придётся создать вручную, его OID'ом будет `1.3.6.1.4.1.311.54.1.2`, ну а после – уже можно сделать новый шаблон сертификата, на основании которого и выпустить сертификат, адресно "заточеный" под RDP Server. Чтобы полностью автоматизировать эту операцию, сделайте у нового шаблона предсказуемое название – например, "RDPServerCert" – и зайдите в объект групповой политики, а там откройте Computer Configuration -> Policies -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host -> Security. Выберите параметр `Server Authentication Certificate Template` и включите его, а в поле значения введите название – мы сделали RDPServerCert. Теперь все доменные хосты, подпадающие под эту политику, будут в случае включения на них RDP сами идти к Certification Authority, запрашивать в случае отсутствия себе сертификат на основе указанного шаблона, и автоматически делать его дефолтным для защиты подключений по RDP. Просто, удобно, эффективно.

Блокируем подключения по RDP учётным записям с пустым паролем

Мелочь, а забывать про неё не нужно. Для блокировки подключения учёток без паролей к RDP надо зайти в настройку объекта групповой политики: Computer Configuration -> Windows Settings -> Security Settings -> Local Policies -> Security Options и установить “Accounts: Limit local account use of blank passwords to console logon only” в Enabled. Не поленитесь проверить, что это так и есть.

Настройка ACL для подключения по RDP

По умолчанию для подключения к RDP-серверу необходимо иметь явное разрешение `User Access` или `Guest Access`. Это разрешение есть у локальных групп `Administrators` и `Remote Desktop Users`. Лучше всего использовать для управления доступом к RDP-серверу группу Remote Desktop Users, добавляя в неё нужные доменные группы, а не отдельных пользователей. Модицифируйте содержимое вкладки `Security` в настройках `Properties` у `RDP-Tcp` только в крайних случаях, лучше всего – добавляя группу “*имя хоста* RDP Blocked”, которой явно запрещен доступ по RDP к указанному узлу.

Оптимизация скорости RDP

Оптимизация скорости RDP – достаточно обширная тема, поэтому я разделю её на части. В этой будут те способы, которые будут уменьшать нагрузку на протокол до сжатия и до оптимизации сетевого уровня.

Цветность (битовая глубина)

В RDP 7.0 и выше доступны варианты 32,16 и 8 бит. Если речь идёт о работе, то для неё будет достаточно 16 бит. Это ощутимо снизит нагрузку на канал, притом иногда больше, чем в 2 раза, что удивительно, но факт. 8 бит, конечно, тоже можно, но уж больно страшно оно будет выглядеть. 16 бит же вполне приемлемы.
Примечание: В Windows Server 2008 R2 подключения с 8 битами уже не доступны.

Включите на сервере параметр Limit Maximum Color Depth, либо сделайте аналогичное действие в настройках RDP client.

Отключите ClearType

Когда у Вас выключен ClearType, протокол RDP передаёт не картинку, а команды по отрисовке символов. Когда включен – рендерит картинку со стороны сервера, сжимает и пересылает клиенту. Это с гарантией в разы менее эффективно, поэтому отключение ClearType значительно ускорит процесс работы и уменьшит время отклика. Сами удивитесь, насколько. Это можно сделать как на уровне настроек клиента, так и на стороне сервера (параметр `Do not allow font smoothing` в разделе Remote Session Enviroment в Computer Configuration -> Policies -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host).

Уберите wallpaper

Параметр `Enforce removal of RD Wallpaper` в разделе Remote Session Enviroment в Computer Configuration -> Policies -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host резко улучшит ситуацию с перерисовкой экрана терминальной сессии. Пользователи без котиков на десктопе выживают нормально, проверено.

Включаем и настраиваем кэширование изображений

Если на клиенте есть достаточно оперативной памяти, то имеет смысл включить и настроить кэширование битмапов. Это позволит выиграть до 20-50% полосы пропускания. Для установки надо будет зайти в ключ `HKEY_CURRENT_USER\SOFTWARE\Microsoft\Terminal Server Client\` и создать там параметры `BitmapPersistCacheSize` и `BitmapCacheSize`, оба типа DWORD 32. Параметр `BitmapPersistCacheSize` обозначает размер в килобайтах дискового кэша. Значение по умолчанию – 10. Имеет смысл увеличить этот параметр хотя бы до 1000. Параметр `BitmapCacheSize` обозначает размер в килобайтах кэша в RAM. Значение по умолчанию – 1500. Имеет смысл увеличить этот параметр хотя бы до 5000. Это будет всего 5 мегабайт на клиентскую сессию, при современных масштабах оперативной памяти это несущественно, и даже если приведёт к выигрышу 10% производительности, уже себя окупит. Кстати, этот же параметр можно поправить и в .rdp-файле; если сохранить своё RDP-подключение, а после открыть файл блокнотом, то среди параметров можно добавить что-то вида `bitmapcachesize:i:5000`, где 5000 – это 5МБ кэша.

Отключаем Desktop Composition

Desktop Composition привносит всякие “красивости” типа Aero и его друзей и ощутимо кушает полосу пропускания. Для работы это не нужно и вредно. Параметр `Allow desktop composition for RDP Sessions` в разделе Remote Session Enviroment в Computer Configuration -> Policies -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host необходимо выставить в параметр Disabled.

Оптимизируем параметры Desktop Window Manager

Параметры, находящиеся в разделе `Remote Session Enviroment` в `Computer Configuration -> Policies -> Administrative Templates -> Windows Components -> Desktop Window Manager`, будут управлять “красивым” отображением плавно выезжающих меню и подобного. Их три – `Do not allow window animations`, `Do not allow desktop compositions` и `Do not allow Flip3D invocation`. Все их надо переключить в режим Enabled, т.е. по сути – отключить все эти функции.

Отключаем редирект неиспользуемых устройств

Если у Вас не планируется подключение определённых классов устройств (например, COM и LPT-портов), или аудио, имеет смысл отключить возможность их перенаправления со стороны сервера. Чтобы клиенты с дефолтными настройками RDP Client не тратили время подключения на согласование неиспользуемого функционала. Это делается там же, где и остальные настройки сервера, в Properties у RDP-Tcp, вкладка Client Settings (там же, где мы делали настройки с глубиной цвета), раздел Redirection.

Настраиваем общую логику оптимизации визуальных данных RDP

Параметр, называющийся `Optimize visual experience for RDP sessions`, находящийся в разделе `Remote Session Enviroment` в `Computer Configuration -> Policies -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host -> Remote Session Enviroment`, будет управлять тем, как RDP будет воспринимает визуальные данные – как мультимедийные или как текстовые. Это, грубо говоря, “подсказка” алгоритму сжатия, как грамотнее себя вести. Соответственно, для работы надо будет выставить этот параметр в `Text`, а если хочется много красивых flash-баннеров, HTML5 и просматривать видеоклипы – лучше вариант `Rich Multimedia`.

Оптимизация сжатия RDP

Сжатие в RDP прошло долгий путь развития. По RDP 5.2 включительно была подсистема сжатия (“компрессор”), имеющий внутреннее название “Version 1” – самый простой и лёгкий вариант с точки зрения загрузки процессора клиента, но самый плохой с точки зрения нагрузки сети трафиком. В RDP 6.0 сделали “Version 2”, который был незначительно, но улучшен по параметру эффективности сжатия. Нам интересен “Version 3”, который работает только при подключении к серверам Windows Server 2008 и старше. Он сжимает лучше всех, а затраты процессорного времени с учётом мощностей современных компьютеров несущественны. Выигрыш при включении V3 может, судя по тестам, достигать 60% и, в общем-то, и без тестов ощутимо заметен на глаз.

Как включить оптимальное сжатие в RDP

Это – клиентская настройка. Откройте в нужном объекте групповой политики Computer Configuration -> Policies -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host -> Remote Session Enviroment, выберите там параметр `Set compression algoritm for RDP data`, включите его и выберите значение `Optimize to use less network bandwidth`. Примечание: У многих возникает вопрос, зачем в списке есть параметр “отключить сжатие”. Это нужно в случае, когда Ваши RDP-сессии сжимает внешнее устройство, оптимизирующее WAN-подключения, что-то вида Cisco WAAS. В других случаях, конечно, отключать сжатие смысла нет.

Настройка сжатия звукового потока

RDP 7.0 приносит отличную возможность регулировать качество сжатия входящего звукового потока (т.е. звука, который идёт с сервера на клиента). Это достаточно полезно – например, если идёт работа на терминальном сервере, то кроме всяких служебных звуков вида “пришло сообщение в ICQ” другие особо как не планируются. Нет смысла передавать с сервера несжатый звук CD-качества, если для работы это не нужно. Соответственно, нужно настроить уровень сжатия звукового потока. Данный параметр будет называться `Limit audio playback quality` и находиться в разделе `Device and Resource Redirection` в `Computer Configuration -> Policies -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host`. Вариантов будет три:

- `High` – звук будет идти без сжатия. Вообще. То есть, он будет подпадать под общее сжатие протокола RDP, но специфическое сжатие звука (с потерей качества) производиться не будет.
- `Medium` – сжатие будет адаптироваться под канал так, чтобы не увеличивать задержку при передаче данных.
- `Dynamic` – сжатие будет динамически адаптироваться под канал так, чтобы задержка не превышала 150ms.

Выберите подходящий. Как понятно, для офисной работы лучше выбрать `Dynamic`.

Оптимизация соотношения потоков данных в RDP

Трафик RDP-сессии не является чем-то монолитным. Наоборот, он достаточно чётко разделён на потоки данных перенаправляемых устройств (например, копирования файла с локального хоста на терминальный сервер), аудиопоток, поток команд примитивов отрисовки (RDP старается передавать команды примитивов отрисовки, и передаёт битмапы в крайнем случае), а также потоки устройств ввода (мышка и клавиатура). На взаимное соотношение этих потоков и логику его (соотношения) вычисления (этакий локальный QoS) можно влиять. Для этого надо со стороны сервера зайти в ключ реестра `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\TermDD` и создать там для начала (если их там нет) четыре ключа:

- FlowControlDisable
- FlowControlDisplayBandwidth
- FlowControlChannelBandwidth
- FlowControlChargePostCompression

Тип у всех – DWORD 32. Функционал у ключей будет следующим. Ключ `FlowControlDisable` будет определять, используется ли приоритезация вообще. Если задать единицу, то приоритезация будет выключена, если нуль – включена. Включите её. Ключи `FlowControlDisplayBandwidth` и `FlowControlChannelBandwidth` будут определять взаимное соотношение двух потоков данных:

- Поток взаимодействия с пользователем (изображение+устройства ввода)
- Прочие данные (блочные устройства, буфер обмена и всё остальное)

Сами значения этих ключей не критичны; критично то, как они соотносятся. То есть, если Вы сделаете `FlowControlDisplayBandwidth` равным единице, а `FlowControlChannelBandwidth` – четырём, то соотношение будет 1:4, и на поток взаимодействия с пользователем будет выделяться 20% полосы пропускания, а на остальное – 80%. Если сделаете 15 и 60 – результат будет идентичным, так как соотношение то же самое. Ключ `FlowControlChargePostCompression` будет определять, когда считается это соотношение – до сжатия или после. Нуль – это до сжатия, единица – после. Я рекомендую для использования вида “наш удалённый сервак далеко и к нему все по RDP подключаются и в офисе и 1С работают” ставить соотношение 1:1 и считать его после сжатия. По опыту это может реально помочь в ситуации “печать большого документа с терминального сервера на локальный принтер”. Но это не догма – пробуйте, главный инструмент – знание, как это считается и работает – у Вас уже есть.

Включаем Require secure RPC communication для RDP

Данный параметр действует аналогично настройкам для Secure RPC, которые есть в разделе Security групповой политики и действуют на всю систему, только настраивается проще. Включив этот параметр Вы сделаете обязательным для всех клиентских RPC-запросов шифрование (в зависимости от настроек системы “нижняя планка” шифрования будет разной – RC4/DES или, в случае включения FIPS-140 – 3DES/AES) и использование как минимум NTLMv2 для аутентификации удалённого вызова процедур. Всегда включайте этот параметр. Есть миф про то, что он не работает во внедоменной среде. Это не так, и усиление защиты RPC никому не помешает.

Это – серверная настройка. Откройте в нужном объекте групповой политики Computer Configuration -> Policies -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host -> Security, выберите там параметр `Require secure RPC communication` и включите его.

Заключение

~~Так как все уже давно вытащили сервера на внешние площадки за бугром, то этот материал является~~ Я надеюсь, что данный материал будет Вам полезен для оптимизации и защиты RDP. Если я что-то пропустил – прошу в комментарии.