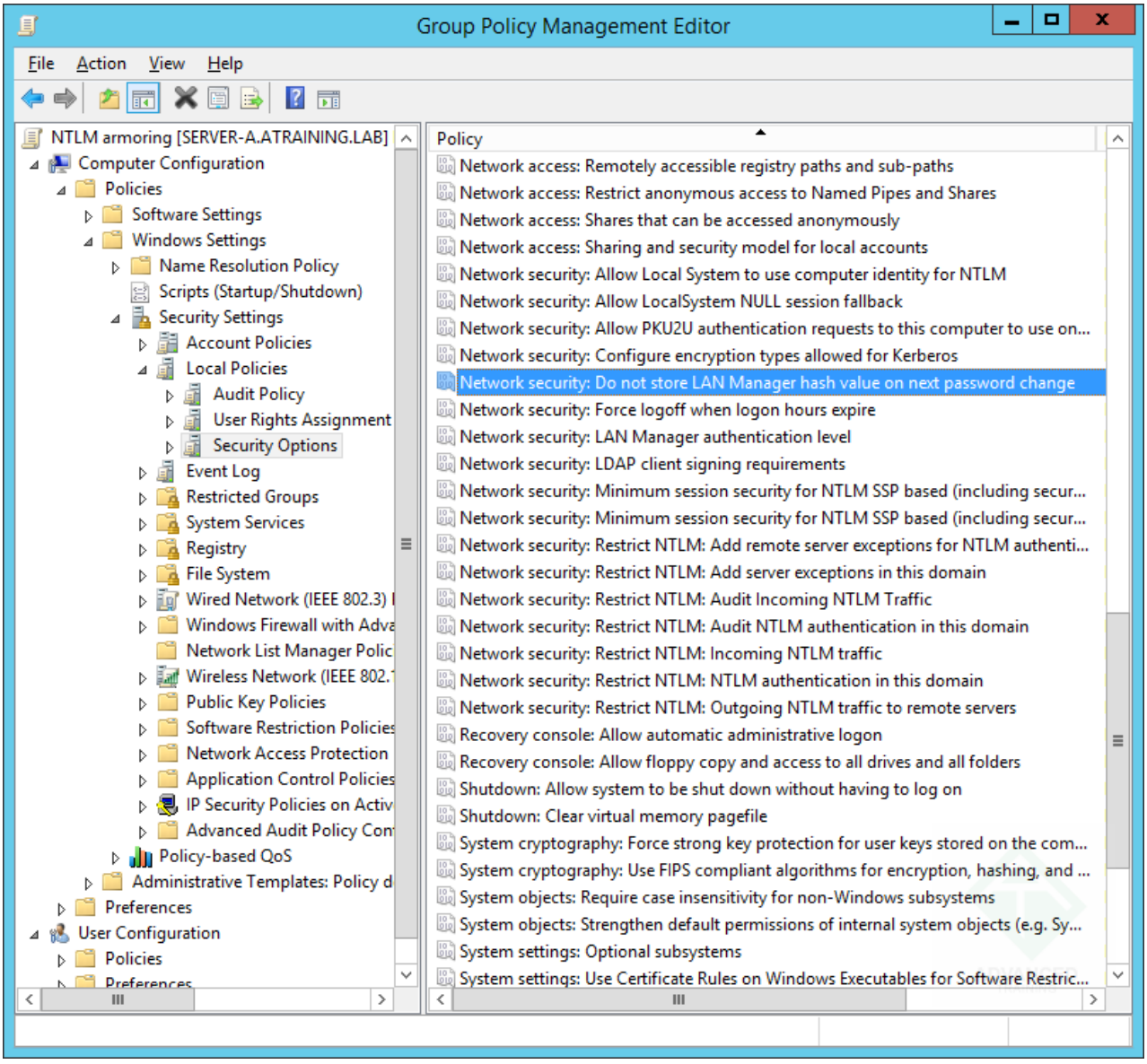
Защищаем протоколы семейства LanManager - NTLMv1 / NTLMv2

atraining.ru/lm-ntlm-ntlmv2-armoring

2015-07-25T02:58:47+08:00



Привет.

Протоколы семейства Lan Manager существуют очень давно – и, хотя, казалось бы, с появлением в Windows 2000 поддержки krbv5, должны были бы уходить на покой, остаются широко применяемыми. Причин этому несколько – во-первых последняя версия – NTLMv2 – широко распространена практически везде (выступая под ликом MS-CHAPv2 что в аутентификации для Wifi через EAP-MSCHAPv2, что для 802.1x, что внутри PEAP для различных применений, например VPN-подключений), во-вторых она является “подстраховкой” в случае не-срабатывания Kerberos в домене

Active Directory, да и некоторые операции – например, работу с локальными учётными записями на доменных машинах – Kerberos обрабатывать не умеет, поэтому там всегда используется кто-то из LM.

Полностью искоренить Lan Manager внутри домена возможно, но для начала стОит навести порядок в использовании существующих диалектов Lan Manager. Важнее ведь не сделать в сети одиночные “островки безопасности”, а поднять нижнюю планку – чтобы небезопасные и откровенно устаревшие варианты LM не использовались, а используемый NTLMv2 был бы безопасен настолько, насколько возможно.

Давайте займёмся этим. Я предполагаю, что вы владеете материалом на уровне нашего бесплатного курса Microsoft 20410D – ну, а если нет, то самое время с ним познакомиться.

Безопасное использование протоколов семейства Lan Manager (NTLMv2 в частности)

- Прощаемся с LM
- Возможные побочные следствия отключения LM
- Полное отключение LM
- Выключаем хранение LM-хэшей
- Удаление хранящихся LM-хэшей
- Выключение отправки LM-запроса и LM-ответа по сети
- Препятствование генерации LM-хэшей
- Прощаемся с NTLMv1
- То, что будет иметь потенциальные проблемы от отключения NTLMv1
- PPP-соединения и NTLMv1
- Защита RPC-запросов
- Защита RDP Gateway от использования NTLMv1
- Включаем Extended Protection for Authentication
- Используем группу Protected Users
- Настраиваем NTLMv2

Прощаемся с LM

Классический Lan Manager появился давно, в составе пачки технологий Microsoft’овского сетевого стека. Там был и Computer Browser, нужный для формирования списка сетевого окружения (значок с этим названием на десктопе был ещё недавно), и недопротокол L3 под названием NetBEUI, и транспортный враппер NetBIOS, имевший внутри IBM’овский вариант SMB и p2p-сервис разрешения имён, и серверная служба разрешения имён NBNS (обычно её называют WINS).

Весь этот зоопарк был нужен в те времена, когда единых стандартов особо не было, и каждый вендор пилил свой стек – поэтому работало всё это счастье поверх любого сетевого протокола – что IP, что IPX, что AppleTalk – и, в общем, нормально работало.

Однако, в классическом Lan Manager на уровне проектирования заложена куча проблем, которые не решаются и в данный момент являются критичными для применения. Посмотрим, например, на алгоритм вычисления LM-хэша от пароля.

1. Пароль обрезается до 14 ANSI-байтов (в то время вопросы нац.алфавитов особо остро не стояли), т.е. лишняя часть просто отбрасывается – если же пароль короче, он добивается 0x00
2. Все строчные латинские буквы (a-z) превращаются в прописные (A-Z)
3. Пароль делится на две части, с каждой из которых последующие действия производятся абсолютно независимо
4. Берётся классический алгоритм DES, с 64х битным ключом (56 бит из которых реально влияют на процесс шифрования), и каждая половинка становися ключом DES. Чтобы получить 64 бита из 7 входящих ASCII-символов (они по 8 бит), используется простая схема – предполагается, что символы эти – 7ми битовые, и просто у каждого байта старший бит зануляется. То есть была битовая строчка из 56 бит (7 символов по 8 бит), сделали 8 символов по 7 бит.
5. Теперь каждый из этих ключей шифрует константу – ASCII-строку “KGS!@#$%”, в результате чего получается два выходных блока DES по 64 бита. Их записывают подряд и получают LM-хэш длиной в 128 бит.

Хорошо заметно, что ослаблений исходных данных – куча. Это и обрезание до короткой длины, и разделение на две части (что неимоверно снижает простоту криптоанализа – перебрать два раза по 2^64 гораздо веселее, чем один раз 2^128) и уменьшение количества возможных символов на 26 (буквы-то только заглавные будут), и шифрование константы, и разрежение ключевой информации предсказуемыми нулевыми битами.

Скажу даже проще – 99% “страшных уязвимостей винды где пароль можно за час подобрать” – это когда используется старый LM, пароль, допустим, из 10 символов, его подбирают по схеме “вначале 3, это мгновенно, а потом прикинем какие 7 первых”.

Поэтому всё плохо и LM мы будем выключать.

Возможные побочные следствия отключения LM

Протокол LM устарел вместе с семейством Windows 9x – начиная с Windows NT 3.1 появился NTLM, про который мы поговорим чуть позже. Однако, даже если у вас в сети до сих пор есть ОС Microsoft на базе 9x, то вопрос решаем – в состав ISO-образа Windows Server 2000 / 2003 входит утилита dsclient, которая может устанавливаться на Windows 95 / 98 / NT Workstation (на Windows ME не будет – в силу “домашнего” позиционирования) и добавляет несколько полезных функций для

взаимодействия с Active Directory, одна из которых – поддержка NTLMv2. То есть при правильной настройке пользователи, заходящие с, допустим, Windows 98 SE, будут корректно аутентифицироваться в домене с DC на базе Windows Server 2012 R2. Я, конечно, надеюсь, что это не ваш случай, но по сути – LM необходимо выключать в любом случае – всё, что после Windows NT 3.1, точно умеет NTLM.

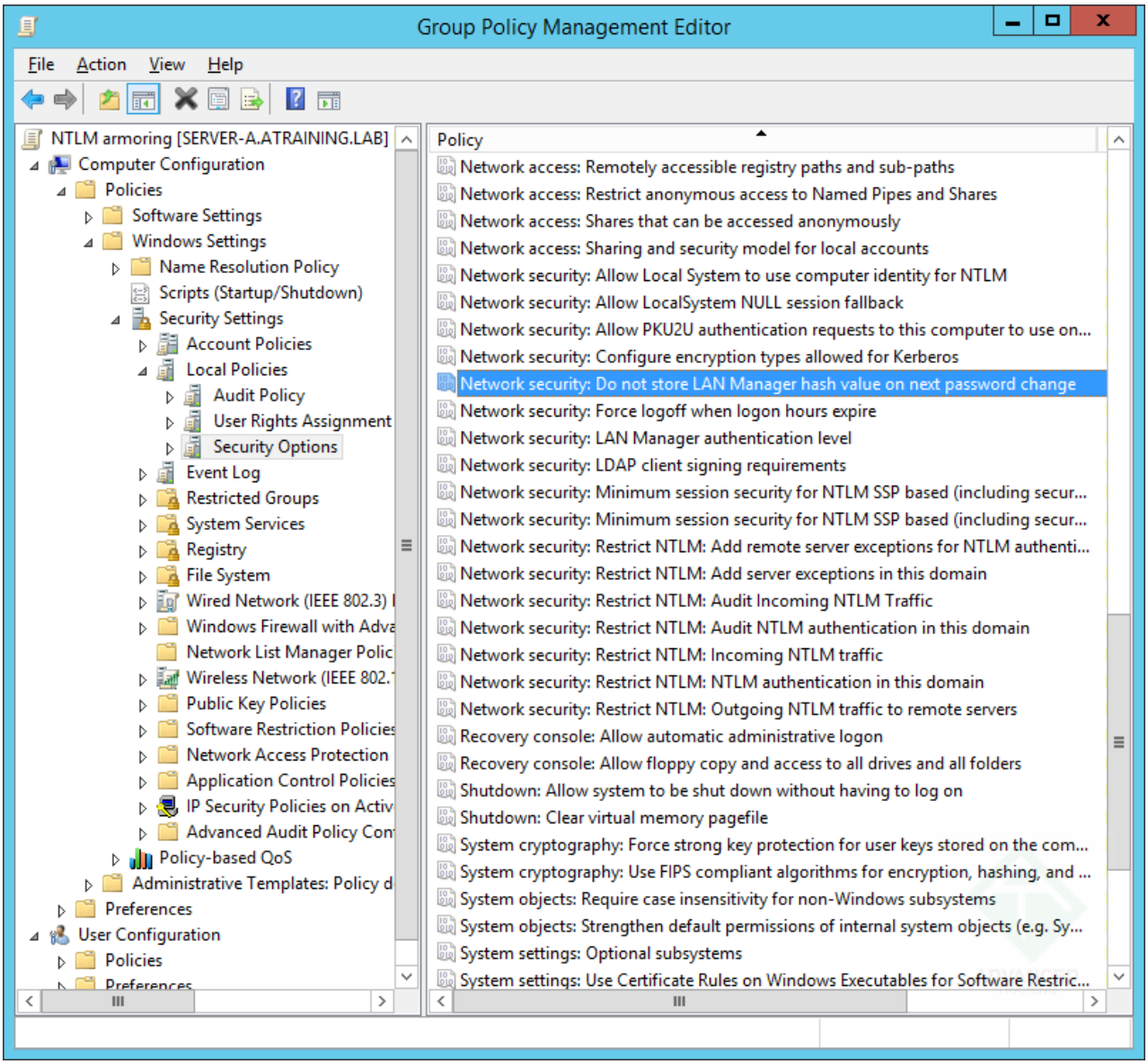
Приступим.

Полное отключение LM

Отключение будет состоять из выключения отправки/приёма протокола LM аутентифицирующими системами, а также выключения хранения LM-хэшей. Да, и ещё мы почистим домен от них.

Выключаем хранение LM-хэшей

Это достаточно несложно – у вас в групповых политиках есть стандартная настройка, которая на хостах, начиная с Windows Server 2000 SP2 запрещает генерацию LM-хэша. Если под эту настройку подпадает рабочая станция или обычный сервер – они не пишут LM-хэш в SAM, если DC – то не пишет в Active Directory. Выглядит настройка вот так (я завёл отдельный объект групповой политики про усиление защиты NTLM и буду всё делать в нём):



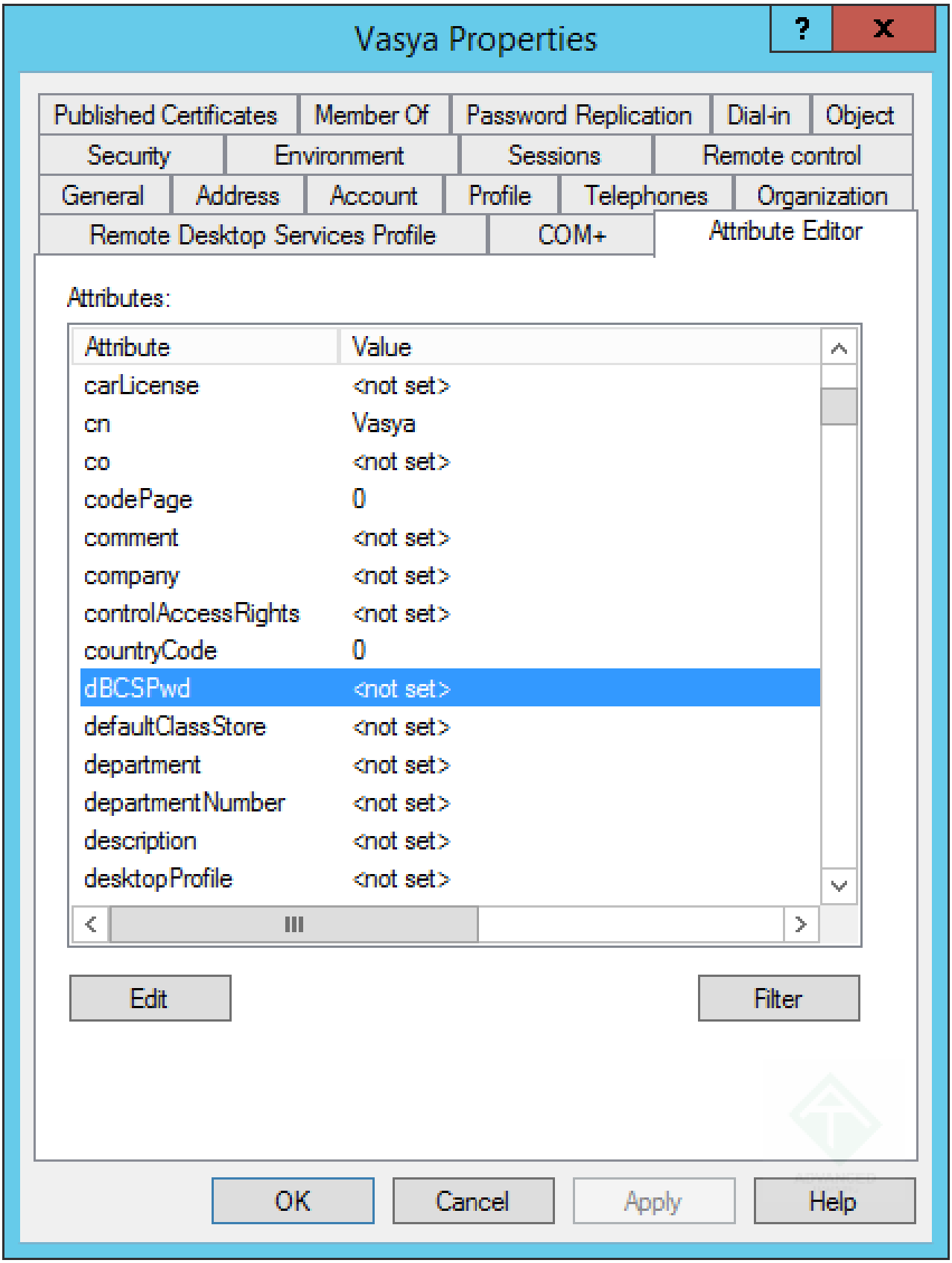
Выключаем хранение LM-хэша
(кликните для увеличения до 802 px на 740 px)

После применения этой настройки новые LM-хэши у подпавших объектов (это будут user и computer) образовываться не будут. У сервисных учётных записей – MSA / gMSA – LM-хэши и так не образовываются, т.к. у них пароль в 240 символов автоматически генерится. У трастовых записей LM-хэши создаваться не будут в том случае, если Вы не используете NT-тресты (если используете – то там будут; так что используйте внутри shortcut-тресты, а наружу forest-тресты – там krbv5rev6, никакого LM).

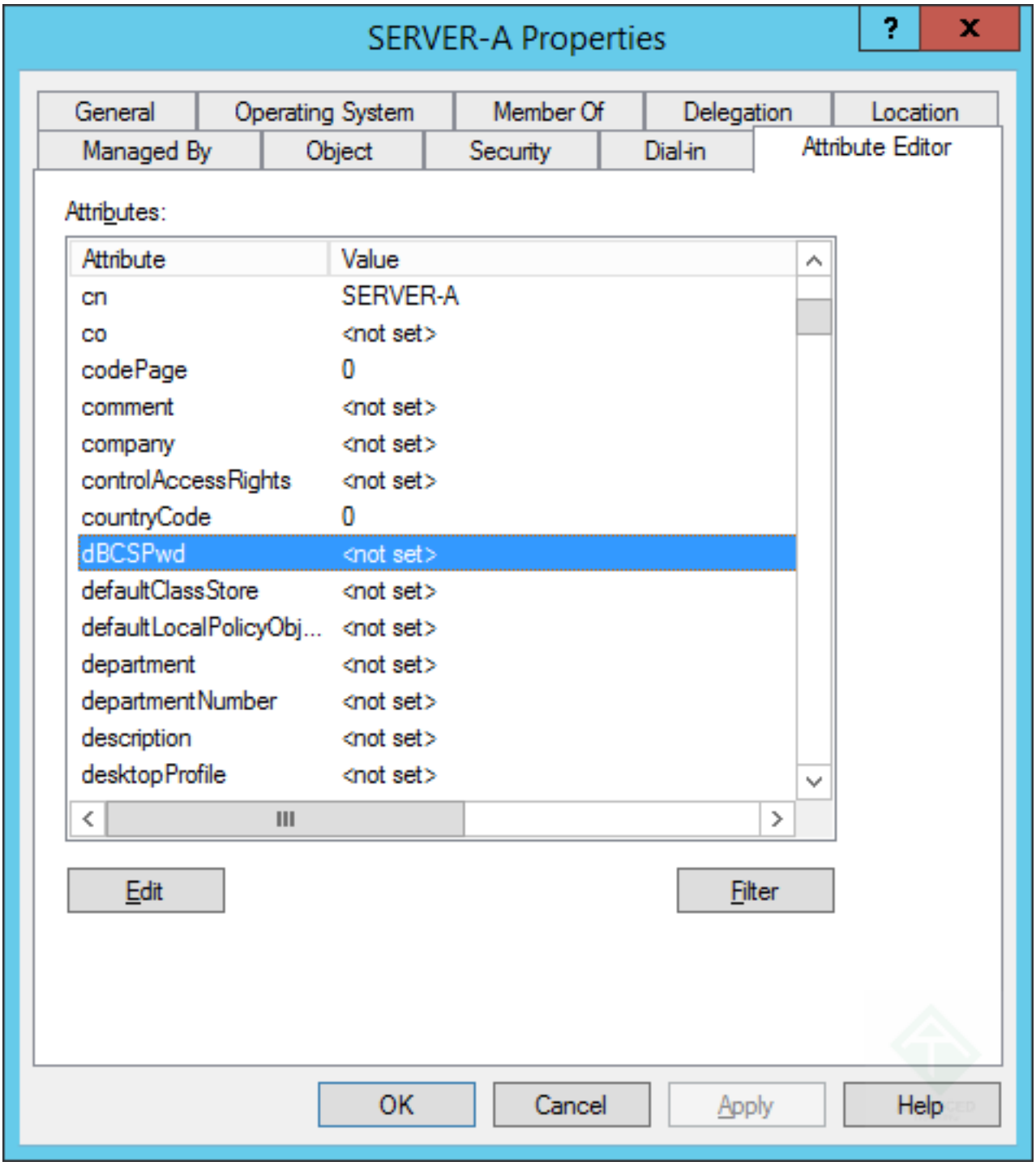
И, в принципе, если заставить всех пользователей пересбросить пароли, и дождаться, пока учётные записи компьютеров тоже регенерят свои учётки (время смены пароля у учётных записей рядовых компьютеров и контроллеров домена задаётся раздельно через политики Active Directory), вопрос будет закрыт. Но это не всегда возможно – давайте копнём поглубже.

Удаление хранящихся LM-хэшей

В домене Active Directory LM-хэш хранится в атрибуте `dBCSPwd`, который есть у security principal'ов – и у пользователя:



Атрибут, в котором хранится LM-хэш у user - dBCSPwd
(кликните для увеличения до 425 px на 563 px)
и у компьютера:



Атрибут, в котором хранится LM-хэш у computer - dBCSPwd
(кликните для увеличения до 477 px на 535 px)

Кстати, у объекта computer в явном виде в схеме его нет – это потому что объект computer порождён от пользователя (да, компьютер в Active Directory – киборг) и наследует от него этот атрибут.

Второй атрибут, который нам интересен – это `lmPwdHistory`, в нём хранится история LM-хэшей. Этот атрибут multivalued, до 25 строчек – вы ведь можете установить глубину хранения истории паролей до 24х штук, помните? Он также неиндексируемый и закрытый от чужих глаз.

Атрибуты выглядят как (not set) – Active Directory охраняет их от нас, т.к. атрибуты эти серьёзные, и даже если мы раздадим себе на них права, и подключимся по TLS, то нам все равно ничего не покажут. Это, в принципе, логично, но всё же возможность удалить содержимое есть.

Вкратце это будет выглядеть так (я опишу сие крайне рамочно, т.к. reverse engineering Windows Server официально считается делом неправильным).

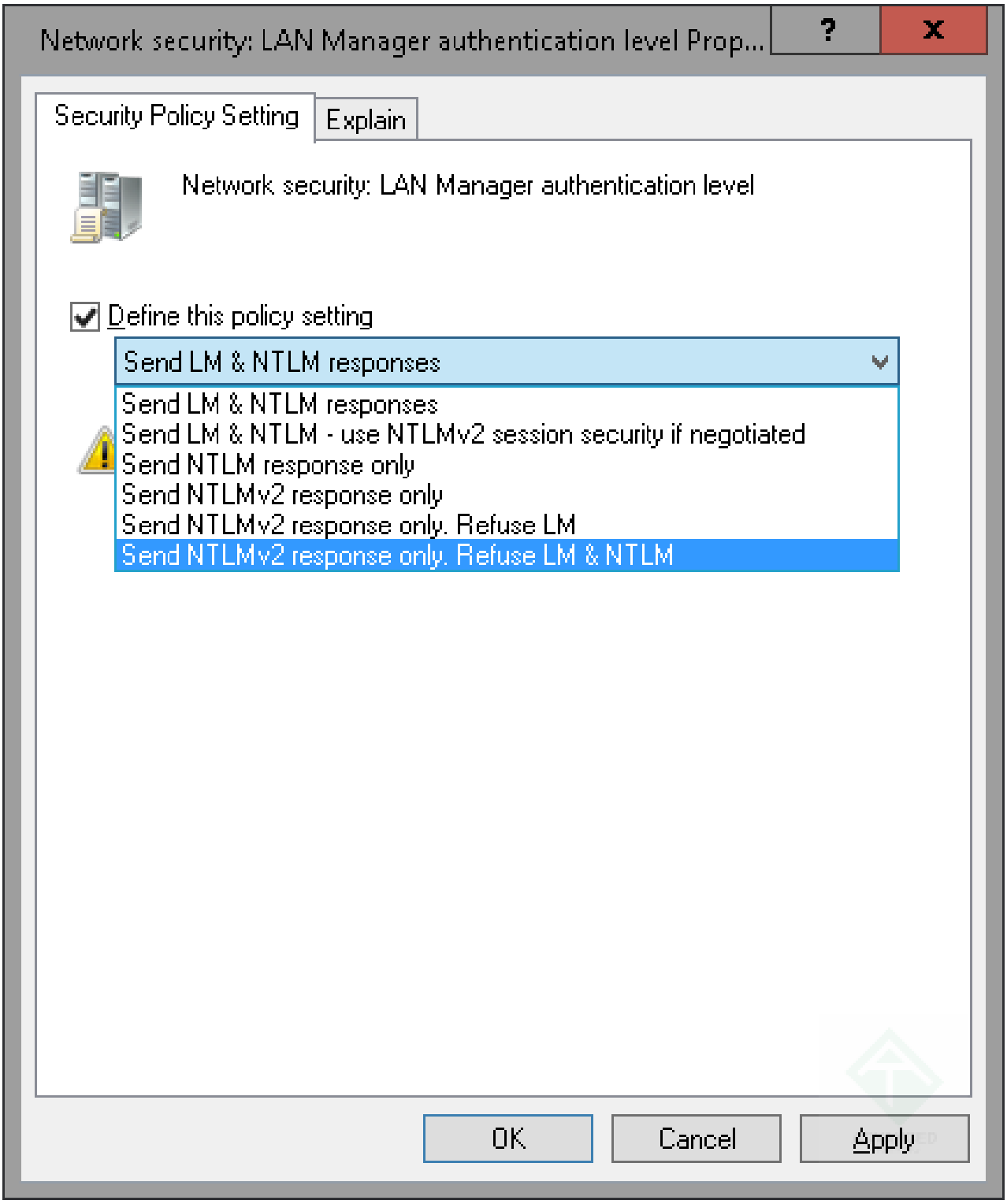
Вы открываете файл ntds.dit – в нём хранится вся ваша локальная информация про объекты Active Directory – используя для этого, например, libsedb. В структуре ntds.dit видите таблицу с предсказуемым названием `datatable` – это как раз объекты и атрибуты. Замечу, что хранится всё достаточно линейно – у каждого объекта потенциально может быть каждый атрибут (даже если он не добавлен в схеме в класс этого объекта). Находите атрибуты с типом “k” и номерами 589879 и 589984 – это будут как раз LM hash и LM hash history. Атрибуты зашифрованы ключом PEK, который зашифрован bootkey’ем (который хранится в реестре) – но нам не расшифровывать надо, а удалять – и можете заменять данные хэши на пустые записи. После этого, конечно, надо будет у данных атрибутов поднять номер ревизии, чтобы при репликации они затёрли значения на других DC домена, ну и после этого ещё и заново проверить целостность ntds.dit через, например, ntdsutil – а то после ручной правки с чексуммами будут проблемы.

Данный метод, конечно, не рекомендуется к выполнению, и приводится исключительно как демонстрация того, что всё же очистить LM-хэши можно.

Теперь перейдём к сетевой части.

Выключение отправки LM-запроса и LM-ответа по сети

Данный параметр будет нужен, чтобы в сетевом трафике никогда не появлялись значения LM-хэшей. Для этого есть стандартная и древняя (с Windows 2000 Server) настройка, которая в общем-то будет содержать в себе достаточно интересную логику использования (по разному обрабатываться на обычных системах и на DC), но в нашем случае, чтобы сэкономить время изложения, мы сразу включим её на самый максимум:



Выключение приёма и отправки слабых диалектов Lan Manager
(кликните для увеличения до 431 px на 514 px)

Такая настройка (она, если что, лежит в Computer Configuration \ Windows Settings \ Security Settings \ Local Policies \ Security Options) будет гарантировать то, что LM-хэши не будут ни отправляться, ни приниматься ни одной системой, подпадающей под эту политику.

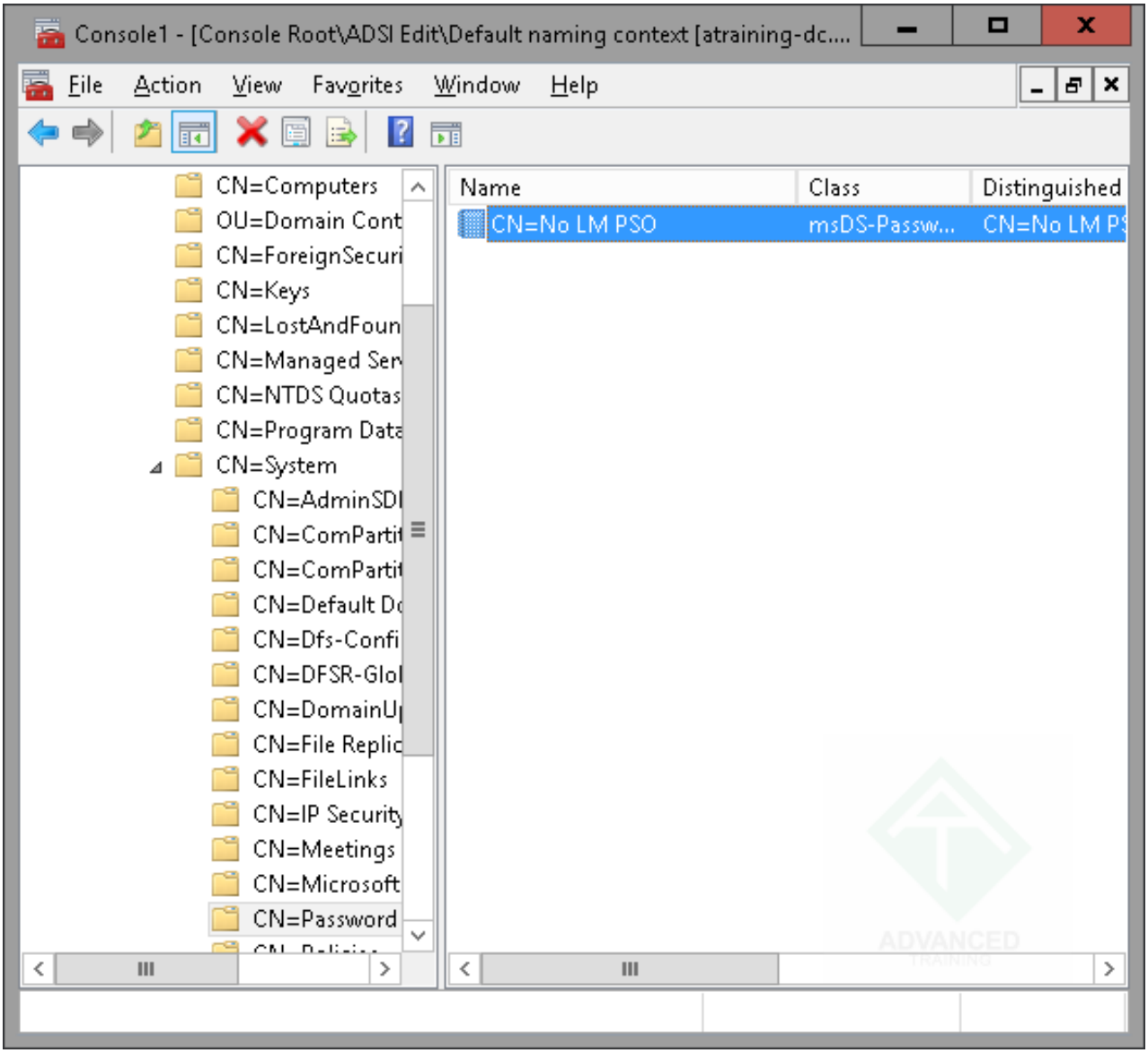
Хочу подчеркнуть, что это никак не приведёт к проблемам в работе современных систем – даже Windows 9x сможет корректно работать с доменом, где есть такая настройка – потому что поддержку NTLMv2 добавить, напомню, можно установкой

dsclient, штатной операцией. Чуть позже мы разберёмся с NTLM подробнее, но сейчас данную настройку будет иметь смысл применить сразу же.

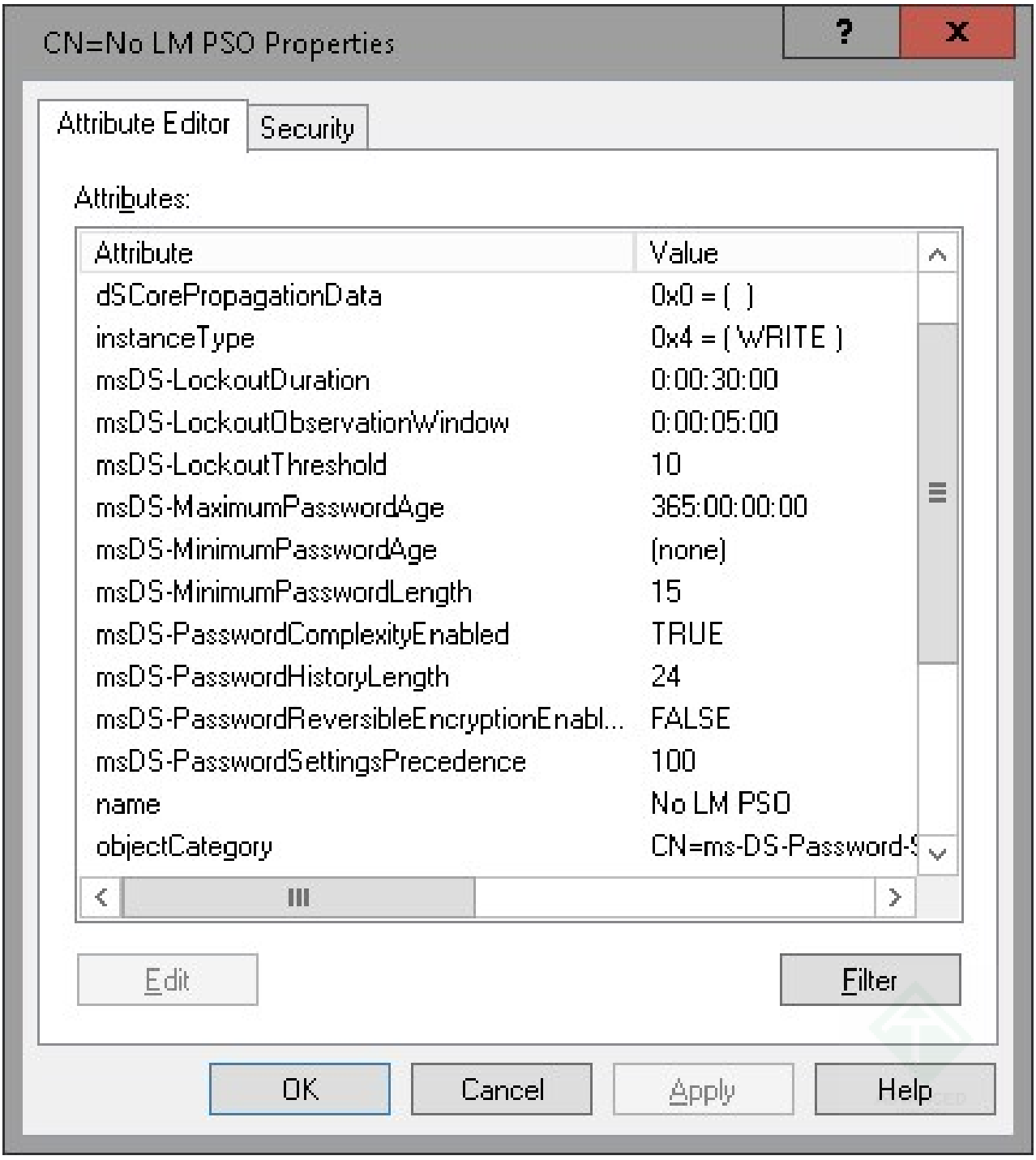
Препятствование генерации LM-хэшей

Это также надо учесть при работе. Самый простой вариант – это генерация паролей с длиной более 14ти символов. LM не сможет создать хэш для такого пароля и вы получите предупреждение – что введённый пароль не сможет быть применён на некоторых старых системах. На самом деле оно бессмысленное – даже на Windows 95 вы сможете установить directory services client (dsclient.exe с ISO-образа Windows 2000 Server / Windows Server 2003) и спокойно использовать пароли любой длины.

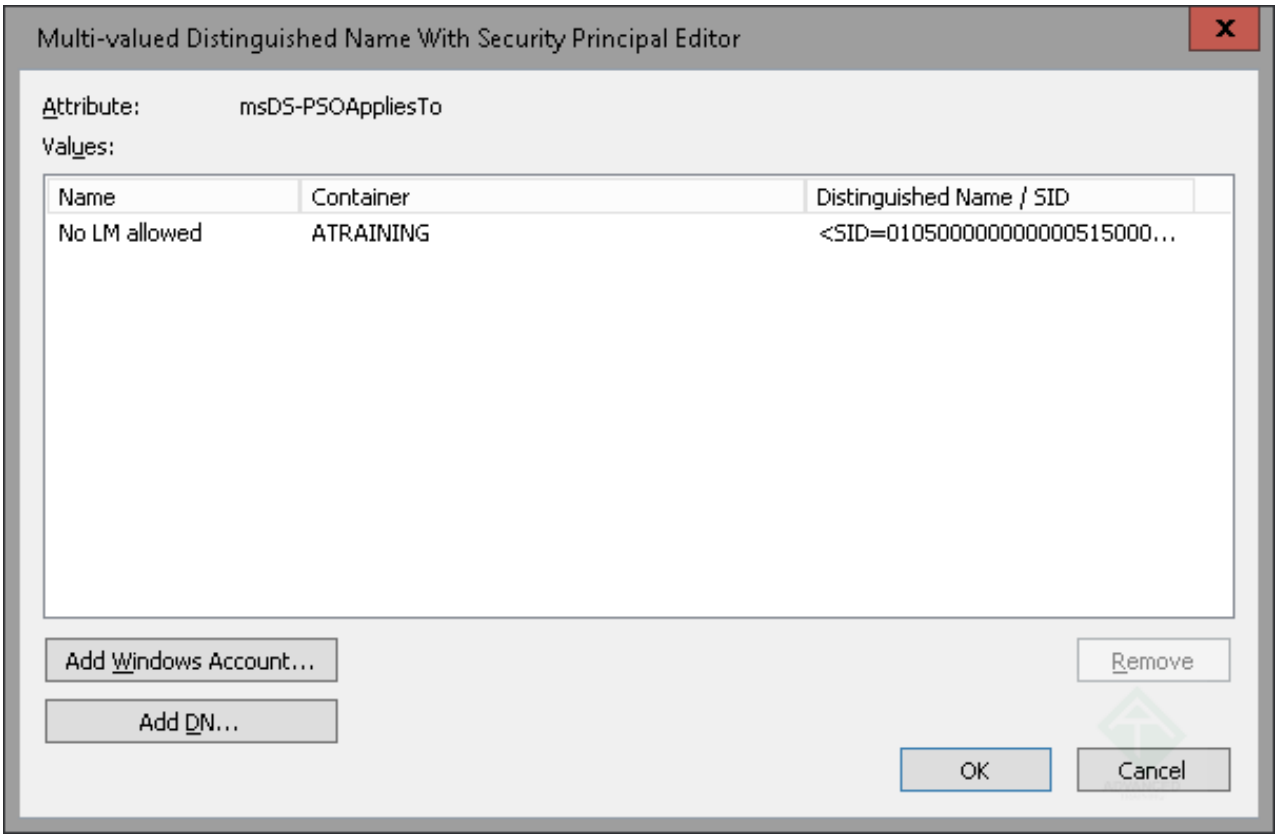
Я бы рекомендовал использовать Fine Grain Password Policies / Password Settings Object для того, чтобы ограничить использование потенциально пригодных к генерации LM-хэша паролей у определённых критичных групп пользователей – например, у администраторов домена. В этом случае всё просто – создаёте новый PSO и привязываете его к целевой группе с названием вида “те, кому запрещены пароли менее 15 символов”:



[Создание PSO для ограничения минимальной длины паролей в 15 символов (кликните для увеличения до 590 px на 539 px)]



[Настройки PSO для ограничения минимальной длины паролей в 15 символов (кликните для увеличения до 414 px на 462 px)]



Привязка PSO для ограничения минимальной длины паролей в 15 символов к доменной группе No LM allowed
(кликните для увеличения до 604 px на 393 px)
Я привязал его [объект] к группе No LM allowed – теперь можно будет просто включить нужных участников из нужных групп домена (например, из Enterprise Admins) в эту группу и на них начнёт действовать политика “пароли не менее 15 символов” – а для остальных продолжит действовать общедоменная политика.

Прощаемся с NTLMv1

Сразу предупреждаю, что с NTLMv1 мы особо много настраивать ничего не будем. Причина проста – это в 2015м году уже устаревший протокол, хранит хэши он так же, как и NTLMv2, поэтому нам надо будет разве что внимательно посмотреть, какие сервисы у нас могут быть не совместимы с NTLMv1, и перейти к NTLMv2. Этот протокол есть начиная с Windows NT 3.1 – поэтому он, как и LM, древний.

Вкратце же по части изменений – NTLMv1 будет гораздо лучше, чем LM. Пароль теперь хэшируется целиком, а в не две отдельные части, нет обрезания пароля до 14 символов и перевода латинских букв в верхний регистр. Поэтому хранящийся NTLM-хэш уже в порядке и будет использоваться и NTLMv1, и NTLMv2, и kerberos – зачистка хранилищ от этого хэша не нужна.

То, что будет иметь потенциальные проблемы от отключения NTLMv1

Первым делом – это клиенты RIS (Remote Installation Services) версий до Windows XP SP1. Они не умеют NTLMv2 и не будут уметь уже никогда. Если вы хотите создать базовый образ Windows XP RTM, и запустите на этой ОС RIS-клиента, чтобы “слить” образ на RIS-сервер, который будет принимать только NTLMv2, вы получите ошибку – клиент не умеет NTLMv2. Выход прост – не впадать в некрофилию и не пользоваться этой версией ОС.

Аналогичная проблема будет у PXE-загрузчика RIS. То есть, если вы загрузитесь с сети, получив по DHCP предложение от Windows Server’овского RIS, и на сервере будет только NTLMv2, вы получите проблемы с аутентификацией, которую вам надо пройти до выбора того образа, который будет заливаться на машину. Выход простой – не пользуйтесь древним сервисом RIS, современный WDS лучше во всех отношениях.

Windows Server 2003 тоже будет иметь проблему, но по сути малозначимую – он будет писать в логах, обрабатывая входящую аутентификацию с NTLMv2, что увидел NTLMv1 (см. KB 2701704). Это не сильно страшно, когда DC такое пишет, просто не пугайтесь, когда отключите NTLMv1 и такое увидите. Впоследствии мы будем искоренять NTLMv2 в домене по-полной, но это уже в следующей статье – поэтому в правильном домене такие сообщения вы должны будете видеть исключительно редко и только в случае какого-то нештатного поведения krbv5rev6.

Проблемы также будут у кучи опенсорсных поделок, которые до сих пор то ли поддерживают, то ли почти поддерживают, то ли почти совсем полностью поддерживают NTLMv2 – их авторы до сих пор решают, насколько всё это свободно и в рамках опенсорсной религии, поэтому в технической статье данные аспекты бытия небольших религиозных сект “истинно свободных по 147 критериям полной, сертифицированной FSF и утверждённой лидерами секты свободы байтовых строк” не затрагиваются.

В остальном, если проблема наблюдается у устройств, которые умеют подключаться к shares (различные принтеры-сканеры), обычно она решается обновлением ПО.

ОК, теперь выключим NTLMv1 везде, где ещё не выключили.

PPP-соединения и NTLMv1

Здесь всё достаточно просто – нам надо выключить использование MS-CHAPv1 везде, где можно, и оставить только MS-CHAPv2. В общем, это и нетрудно – внутри EAP’овских схем авторизации и так используется только MS-CHAPv2, а в случае обычной авторизации (когда есть выбор между PAP / SPAP / CHAP / MS-CHAPv1 / MS-CHAPv2) оставлять что-то, помимо MS-CHAPv2, нет смысла даже в тестовых средах. Выключите MS-CHAPv1 и со стороны сервера, и со стороны клиентов PPTP / L2TP / SSTP.

Защита RPC-запросов

Помимо аутентификации учётных записей, которые хотят куда-либо получить доступ, в домене Active Directory – да и на всех системах NT – работает аутентификация внутри запроса на удалённый вызов процедуры. Это DCE/RPC и DCOM – поэтому там тоже надо сказать, что использовать NTLMv1 не нужно. Это несложно сделать, существует две специальных политики – одна настраивает ситуацию “когда с этого хоста запрашивается RPC-вызов”, вторая – “когда к этому хосту приходит RPC-запрос”:

[Выключаем NTLMv1 на RPC-вызовах
(кликните для увеличения до 431 px на 514 px)]

В этом поле на самом деле 4 настройки, но нам показывают две, а настроить надо для достижения нашей цели вообще одну.

Очень важно, чтобы данная настройка применялась на все хосты, потому что до NT 6.0 она не является дефолтной, т.е. надо, чтобы XP и 2003и системы явно поняли, что надо использовать NTLMv2 для RPC.

Защита RDP Gateway от использования NTLMv1

Для RDP Gateway-серверов, начиная с Windows Server 2012 (не R2), существует способ явно отказывать клиентам, которые пытаются использовать NTLMv1 для подключения через шлюз служб терминалов. Этот способ – задание в ключе реестра `HKLM \ Software \ Microsoft \ Windows NT \ CurrentVersion \ TerminalServerGateway \ Config \ Core` параметра `EnforceChannelBinding` (DWORD32) в единицу. В случае применения клиентом NTLMv1 механизм channel binding не сработает и подключающиеся, выбравшие во вкладке RDP-клиента настройку “авторизоваться на шлюзе при помощи NTLM” будут использовать только NTLMv2.

Это, на самом деле, частный случай защиты NTLM от уязвимости “после установки TCP-сессии больше не проверяем всё, что дальше” – полновесное включение выглядит так:

Включаем Extended Protection for Authentication

Механизм EPA / Channel Binding Tokens появился как метод борьбы с патологической уязвимостью NTLM всех версий – возможностью replay-атаки. Т.е. NTLM до этого механизма аутентифицирует сессию по её сетевым параметрам – и в сценарии “из-под одного и того же адреса и порта прокси-сервера выходит несколько мультиплексированных сессий или одна сессия обрывается, а другая сразу начинает работать”, NTLM не понимает, что это уже разные сессии. Технически механизм представляет из себя генерацию (со стороны клиента) и проверку (со стороны сервера) двух дополнительных полей в сообщении NTLM_AUTHENTICATE – `MsvChannelBindings` and `MsvAvTargetName`, которые образно называются Channel Binding Token’ом.

Механизм появляется в Windows 7 / Windows Server 2008 R2, но патч, реализующий его, есть и для XP / 2003.

Ключевые пункты, которые надо для себя зафиксировать:

1. EPA работает только для NTLMv2 (а также digest и Kerberos) – для NTLMv1 он не реализован технически, ни на проверке со стороны сервера, ни на отправке со стороны клиента
2. Поддержка EPA не-Microsoft’овским софтом – вопрос индивидуальный, и всё необходимо проверять

Механизм будет включаться следующим образом – устанавливаться нужный патч (я предполагаю, что у вас в домене Active Directory все системы уже со всеми установленными security-патчами – речь про KB 973811 от 2009 года), и явно включаться проверка данных параметров со стороны сервера.

Включаем EPA для NTLM

Если речь про NTLMv2-аутентификацию в домене, то включается так:

Ключ реестра `HKEY_LOCAL_MACHINE \ System \ CurrentControlSet \ Control \ LSA \` дополняется DWORD32-значением `SuppressExtendedProtection`, которое ставится в ноль. Логика простая – это значение указывает, при работе каких протоколов подавлять генерацию CBT. Если в единицу поставить первый бит (т.е. `SuppressExtendedProtection` = 1), то CBT не будут генериться при работе NTLMv2, если второй – то при работе Kerberos. Соответственно, установка `SuppressExtendedProtection` в тройку выключает механизм и для NTLM, и для Kerberos, а в нуль – включает.

Включаем EPA для Telnet Server

Да, даже для такой некрофилии есть способ улучшить проверку подлинности. Telnet server в Windows Server умеет использовать NTLM изначально (чтобы не передавать учётные данные открытым текстом), и в нём тоже можно закрутить гайки, чтобы использовался именно NTLMv2 и с EPA. Это несложно – надо взять ключ реестра `HKEY_LOCAL_MACHINE \ SOFTWARE \ Microsoft \ TelnetServer \ 1.0 \`, создать в нём DWORD32 `ExtendedProtection` и поставить туда единицу, если хочется, чтобы CBT анализировались, и двойку – если хочется, чтобы требовались.

Включаем EPA для SMB-хранилищ

Для того, чтобы при доступе к обычной “файловой шаре” использовался исключительно NTLMv2, нам надо будет чуток поправить настройки сервиса LanmanServer. Мы зайдём в ключ реестра `HKEY_LOCAL_MACHINE \ SYSTEM \ CurrentControlSet \ Services \ LanmanServer \ Parameters \` и создадим там DWORD32-параметр `SmbServerNameHardeningLevel`. Нам надо будет выставить этот параметр как минимум в единицу – чтобы данный сервер принимал от клиентов EPA-запросы, и в двойку, если мы хотим жёстко ограничить работу с данным сервером только протоколом NTLMv2 (безусловно, из Lan Manager – семейства – в случае работы с Kerberos CBT будут присылаться без доп.настроек).

В случае включения этой настройки гарантированно станут недоступны общие папки, расположенные на не-доменных хостах Windows XP – но, думается, это не то чтобы минус, а даже скорее плюс – крайне сомнительно, что в современном домене на подобный сценарий может быть что-то завязано.

Включаем EPA для HTTP

Для включения проверки CBT у входящих аутентификаций поверх HTTP (это может быть и digest, и NTLM), нужно будет взять IIS версии 7.5 или старше, найти в его конфигурации (system.webServer / security / authentication / windowsAuthentication) параметр `< extendedProtection >`, в котором будет элемент `tokenChecking`, который надо установить в единицу.

Учтите, данный параметр весьма сильно влияет на вопросы совместимости. Убедитесь, что для начала на всех ваших серверных системах есть его поддержка и включен приём/обработка данного параметра, а после – что все сторонние продукты, использующие, например, NTLM-аутентификацию для прокси, поддерживают именно NTLMv2 и умеют создавать/читать CBT. Включение EPA с гарантией похоронит NTLMv1.

Убедитесь, что подключающиеся клиенты отправляют CBT – данная настройка реально улучшит безопасность, но повлияет на множество механизмов – например, на подключения Outlook с использованием RPC over HTTPS и модификаций данного механизма (который до сих пор использует NTLM). Если поставить у IIS вышеупомянутый параметр `tokenChecking` в двойку, а не единицу, CBT будут не accepted, а required, что приведёт, допустим, к гарантированному отбою для обычной Windows XP, где не установлен нужный патч, не включён механизм EPA в явном виде, и не включена отправка NTLMv2 (см. настройку LM Compatibility в части статьи про LM).

В общем, теперь перейдём к настройке NTLMv2 – который мы будем закапывать уже в следующей статье про Kerberos. Ещё раз подчеркну – упомянутые выше техники позволяют практически полностью исключить работу NTLMv1 в домене даже на базе XP / 2003 (за исключением разве что работы поверх HTTP – всё ж IIS 7.5 будет только с Windows Server 2008 R2) и перейти на более безопасный NTLMv2.

Используем группу Protected Users

Начиная с Windows Server 2012 R2, появляется дополнительная возможность по решению проблем с NTLM. Эта возможность реализуется через новую группу с названием Protected Users.

Данная группа создаётся при ситуации, когда FSMO-роль PDC-эмулятора первый раз переживает процесс включения на DC с версией Windows Server 2012 R2 или более высокой. У группы фиксированный RID, равный 525, и DC проверяет наличие в домене такой группы. Тип группы, так как она адресно предназначена для консолидации специфично защищённых учётных записей пользователей – не BUILTIN, а Global.

Есть ошибочное предположение, что для работы этого механизма уровень домена (или даже леса) должен быть также 2012R2+. Это не так, умение специфически обрабатывать наличие Protected Users в списке SID'ов у security principal'а – свойство сервиса lsass на конкретной системе. На ранних версиях Windows lsass

будет просто добавлять SID группы Protected Users в маркер, да и всё, не зная о том, что этот SID ‘знаковый’ и что его наличие нужно для применения к обрабатываемому security principal’у дополнительных ограничений.

Вы можете научить LSASS старых версий Windows работать с Protected Users, установив обновление KB 2871997.

Что же добавляет членство учётной записи пользователя в Protected Users?

Данный механизм нужен, чтобы упрощённо применять сразу несколько ограничений, в частности:

- Использовать NTLM для входа (блокируется хранение результата работы функции NTOWF).
- Кэшировать пароли в форматах plaintext и Digest (т.е. то самое ‘reversible encryption’ и CHAP работать не будут).
- Использовать до-NT6-методы-шифрования-kerberos-tickets. То есть все виды DES от ‘классического’ Kerberos и RC4-128.
- Осуществлять ‘позднее’ обновление TGT – т.е. обновлять во второй половине срока использования (при стандартных 8 часах это даёт ограничение в 4 часа).
- Проверять каждый запрос TGT, то есть вообще не кэшировать результат аутентификации Kerberos.
- Использовать любое делегирование – включая ограниченное, которое ‘constrained delegation’. Даже если в политике явно включено Allow delegating default credentials, например.

Так как наша статья про NTLM, то лишь отметим, что механизм выключения NTLM – именно выключения, полного запрета – появляется в NT 6.1, т.е. Windows 7 и Windows Server 2008 R2, и может применяться при любом уровне домена и/или леса Active Directory, а также вне домена. Поэтому Protected Users – это не какая-то Особая Технологичная Продвинутая Штука, а упрощённый способ применения “пачки” отдельных мер безопасности, работавших и на предыдущих, до-Windows 2012 R2 версиях Windows. Применение подхода вида “Появилась группа с надписью Защищённые – если туда добавить учётку, она Защитится” – это, конечно, финиш с точки зрения уровня логики и подхода к технической стороне вопроса, поэтому гораздо лучше изучить то, что именно применяется к участникам данной группы. Потому что отсутствие кэширования Kerberos, к примеру, явно скажется на скорости при ряде операций внутри домена (типа доступа к множеству сетевых файлов), а запрет на разные виды кэширования отключит часть возможностей по использованию RDP-подключений к доменным серверам со вне-доменных рабочих станций. Плюс в Protected Users нельзя добавлять учётные записи компьютеров и MSA/gMSA, т.к. оные хранят закэшированными все учётные данные для подтверждения своей подлинности при подключении к контроллеру домена.

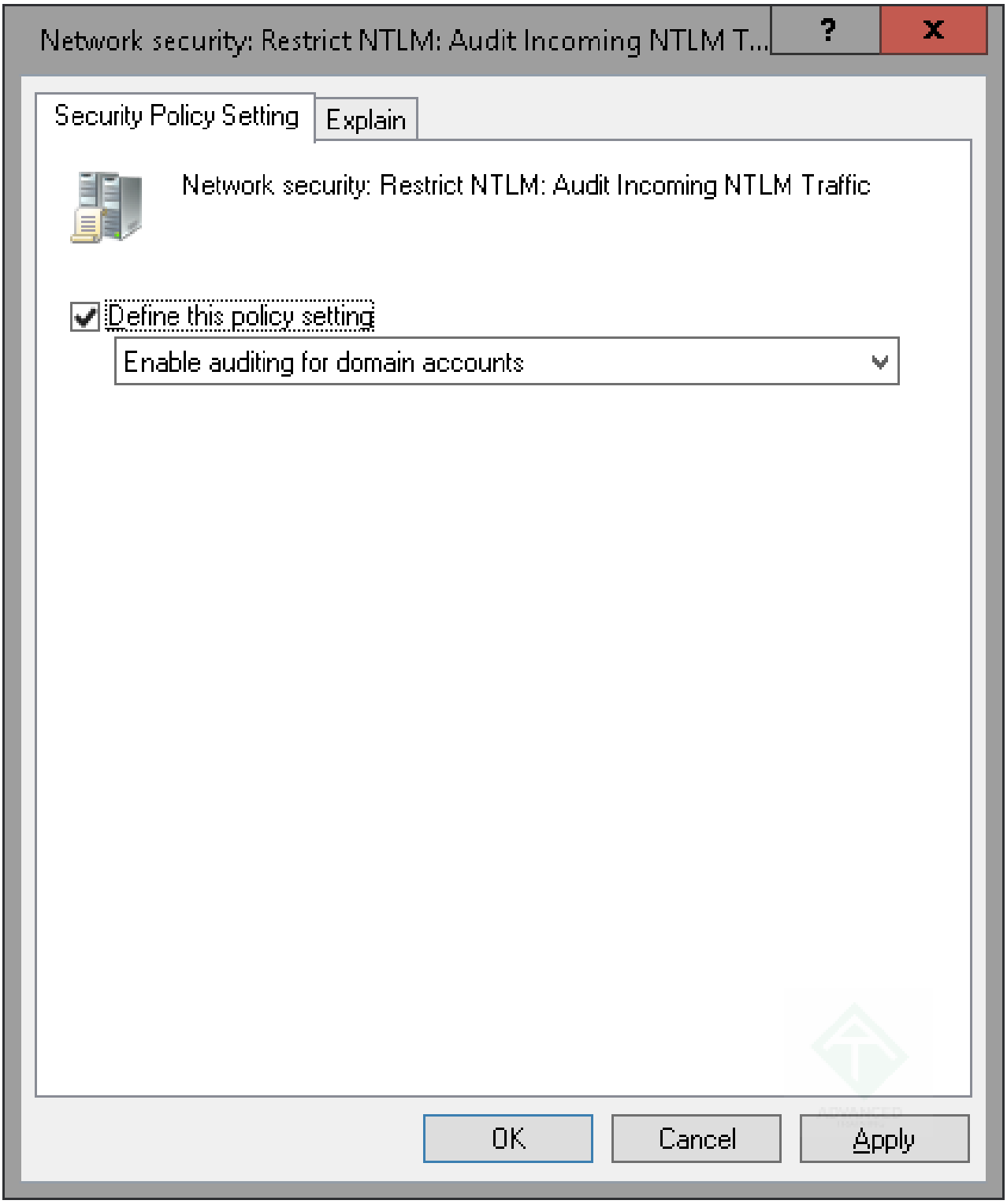
Поэтому что вера в волшебную кнопку “Всё Сделать Хорошо”, что пропагандистам подобного подхода “легко и быстро”, должна быть очень осторожной.

Настраиваем NTLMv2

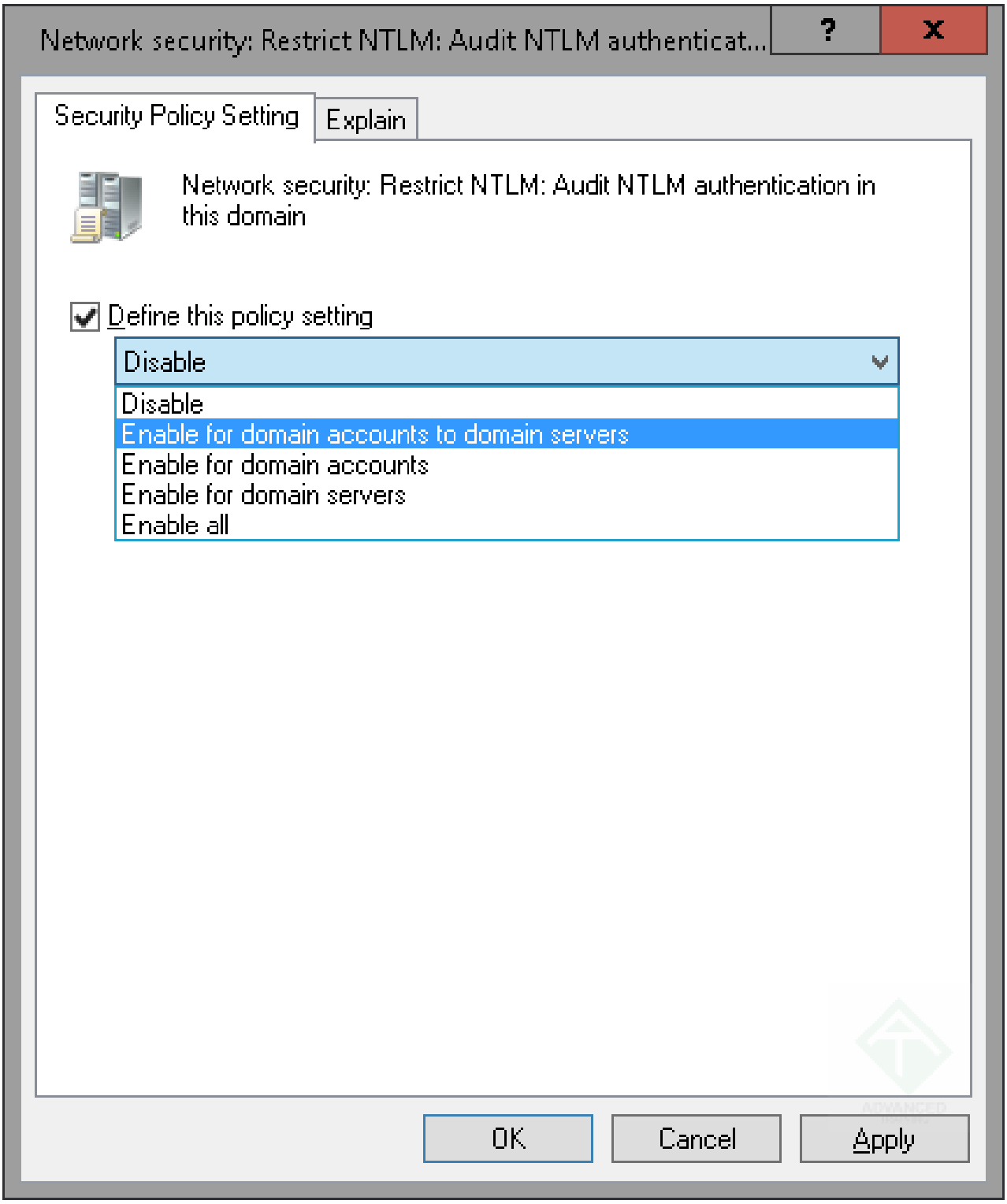
Настройка NTLMv2, после всех проведённых выше действий, будет уже иметь несколько другую цель – максимальную прозрачность сценариев применения NTLMv2 в домене и выход на последующую минимизацию использования NTLMv2, чтобы везде (в идеале) использовался Kerberos, а fallback to NTLM было бы редчайшим и вынужденным событием.

В современных доменах, администрирующихся по принципу “Next-Next-Finish” и “вродь пашет”, увы, характерной картиной является ситуация, что, несмотря на поддержку протокола Kerberos уже 16й год, Lan Manager используется более чем активно. Притом администраторы обычно не могут внятно объяснить, в каких сценариях и когда используется какой из протоколов – а в ряде случаев вообще наблюдается вполне официальная позиция вида “ставьте SharePoint 2013 с NTLM, а то с керберосом там мутно всё очень” ну и “у нас распоследняя версия Windows Server куплена и пропатчена, поэтому всё работает само идеально, а если что-то не так, это проблемы Microsoft’а”. Это, конечно, ~~полный пиз~~ очень плохо.

Поэтому мы включим аудит априори нехороших применений NTLM – т.е. в тех случаях, когда он зачем-то используется для аутентификации доменных учётных записей в Active Directory. Настройки для этого будут присутствовать, начиная с Windows Server 2008 R2, в той же самой вкладке групповой политики, в которой были предыдущие:



[Включаем аудит NTLMv2 в Active Directory](#)
[(кликните для увеличения до 431 px на 514 px)](#)
и



[Включаем аудит использования NTLMv2 в Active Directory
(кликните для увеличения до 431 px на 514 px)]

Данные настройки, применившись, начнут записывать в журналы операций ситуации, когда доменные учётные записи зачем-то аутентифицировались по NTLM, а не по Kerberos, что даст пищу для размышлений и траблшутинга работы Kerberos.

В общем-то на этом фазу укрепления Lan Manager можно закрывать и переходить к улучшению работы Kerberos – чтобы NTLM был нужен как можно реже и только в ситуациях, когда без него никак (например, в том же RPC over HTTPS или при

использовании локальных учётных записей для доступа к общим папкам и ресурсам).

Напоследок

Несмотря на то, что многие ассоциируют Lan Manager с чем-то древним и уже не используемым, используется он достаточно активно – поэтому его надо и настраивать, и оптимизировать, и защищать, и грамотно использовать только в нужных случаях, а не оставлять всё на самотёк. Такой подход порождает множество проблем в безопасности – ведь ломать-то будут не то, что админ штучно настроил, а самое слабое место. А если в сети ходит невнятная каша из протоколов аутентификации – то задача нарушителя сильно упрощается.

Надеюсь, что данная статья поможет вам в задаче улучшения безопасности корпоративной сети.

До новых встреч!