11 незаменимых средств управления Active Directory

https://fadmin.ru/article/11-nezamenimyh-sredstv-upravleniya-active-directory

Те, кому приходилось иметь дело с такими вещами, как таблица Excel, перечисляющая 200 новых сотрудников, начинающих работать со следующей недели, или учетные записи пользователей, настроенные неверно, потому что кто-то в службе поддержки щелкнул то, чего щелкать не следовало, а также те, кому интересен более простой способ управления Active Directory®, помимо открытия папок «Пользователи» и «Компьютеры» каждый раз, могут воспользоваться одним из бесплатных средств администрирования.Некоторые из них встроены прямо в операционную систему Windows®, некоторые поставляются в пакете Resource Kit или в наборе средств поддержки Windows, а некоторые являются бесплатной продукцией сторонних производителей. Что это за удобные средства и где их можно достать? Давайте выясним.

Начнем со встроенных средств командной строки в Windows Server® 2003, позволяющих создавать, удалять, модифицировать и искать объекты в Active Directory.

Contents

CSVDE....1
LDIFDE....2
Dsadd....4
Dsmod....4
Dsrm....5
Dsmove....5
Dsget и Dsquery....5
Некоторые находки от сторонних производителей....7
Adfind и Admod....7
Oldcmp....8

CSVDE

Средство CSVDE позволяет импортировать новые объекты в Active Directory, используя исходный CSV-файл; оно также дает возможность экспортировать существующие объекты в файл CSV. CSVDE нельзя использовать для изменения существующих объектов; при использовании этого средства в режиме импорта можно лишь создавать новые объекты.

Экспорт списка существующих объектов с помощью CSVDE довольно прост. Ниже показано, как экспортировать объекты Active Directory в файл под названием ad.csv:

```
csvde –f ad.csv
```

Параметр –f указывает, что за ним следует имя выходного файла. Но следует понимать, что, в зависимости от среды, этот базовый синтаксис может привести к выводу огромного и неудобного файла. Чтобы ограничить средство экспортом лишь объектов внутри определенного структурного подразделения (OU), команду можно изменить следующим образом:

```
csvde –f UsersOU.csv –d ou=Users,dc=contoso,dc=com
```

Предположим далее, что мне необходимо экспортировать лишь объекты пользователя в мой файл CSV. В таком случае можно добавить параметр –r, позволяющий указать фильтр протокола LDAP для данного поиска, который ограничит число экспортируемых атрибутов (заметьте, что все нижеследующее является одной строкой):

```
csvde –f UsersOnly.csv –d ou=Users,dc=contoso,dc=com –r
“(&(objectcategory=person)(objectclass=user))” –l
DN,objectClass,description
```

Параметр –i позволяет импортировать объекты в Active Directory из исходного файла CSV. Однако создание объектов пользователя с помощью CSVDE имеет один важный недостаток: с помощью этого средства нельзя устанавливать пароли пользователей, поэтому я бы не стала использовать CSVDE для создания объектов пользователей.

LDIFDE

Active Directory предоставляет второе встроенное средство для пакетных операций пользователей, именуемое LDIFDE и обладающее более широкими и гибкими возможностями, чем CSVDE. Помимо создания новых объектов, LDIFDE позволяет модифицировать и удалять существующие объекты и даже расширять схему Active Directory. Платой за гибкость LDIFDE является то, что необходимый входной файл (файл LDIF) с расширением .ldf использует более сложный формат, чем простой файл CSV. (Немного поработав, можно также настраивать пароли пользователей, но об этом чуть позже.)

Начнем с простого примера — экспорта пользователей в структурном подразделении в файл LDF (отметьте, что все нижеследующее является одной строкой):

```
ldifde -f users.ldf -s DC1.contoso.com -d “ou=UsersOU,dc=contoso,dc=com”
–r “(&(objectcategory=person)(objectclass=user))”
```

Как и в случае большинства средств командной строки, полное описание параметров LDIFDE можно получить, запустив команду LDIFDE /? . На Рис. 1 показаны те, что я использовала здесь. (Заметьте, что параметры для команд CSVDE и LDIFDE одинаковы.)

Figure 1 Параметры LDIFDE

Параметр	Описание
-d	Указывает путь LDAP, к которому LDIFDE следует подключиться для выполнения операции.
-f	Указывает имя файла, который следует использовать, в данном случае, для вывода результатов экспорта.
-r	Указывает фильтр LDAP для использования при экспорте.
-s	Указывает контроллер домена (DC) к которому следует подключиться для выполнения операции; если не вводить этот параметр, LDIFDE подключится к локальному DC (или DC, проверившему подлинность, если средство используется с рабочей станции).

По-настоящему возможности LDIFDE раскрываются при создании объектов и управлении ими. Однако перед этим необходимо создать входной файл. Нижеследующий код создает две новых учетных записи пользователя — afuller и rking; для создания входного файла введите текст в блокноте (или другом редакторе открытого текста) и сохраните его как NewUsers.ldf:

```
dn: CN=afuller, OU=UsersOU, DC=contoso, DC=com
changetype: add
cn: afuller
objectClass: user
samAccountName: afuller

dn: CN=rking, OU=UsersOU, DC=contoso, DC=com
changetype: add
cn: rking
objectClass: user
samAccountName: rking
```

После того как создание файла завершено, запустите следующую команду:

```
ldifde –i –f NewUsers.ldf –s DC1.contoso.com
```

Единственный новый параметр здесь — это -i, который, как несложно догадаться, указывает, что выполняется операция импорта, а не экспорта.

При модификации или удалении существующих объектов синтаксис команды LDIFDE не меняется; вместо этого изменяется содержимое файла LDF. Для изменения поля описания учетных записей пользователей создайте текстовый файл, именуемый ModifyUsers.ldf, такой как показано на Рис. 2.

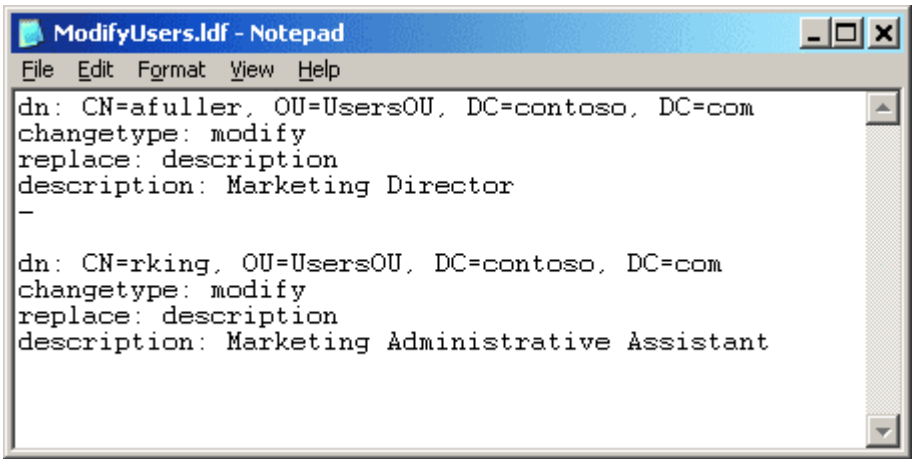

Рис. 2 Файл LDF ModifyUsers

Изменения импортируются путем запуска того же синтаксиса команды LDIFDE, что и раньше, с указанием нового файла LDF после параметры -f. Формат LDF для удаления объектов еще проще; для удаления пользователей, с которыми вы работали, создайте файл, именуемый DeleteUsers.ldf, и введите следующее:

```
dn: CN=afuller OU=UsersOU, DC=contoso, DC=com
changetype: delete

dn: CN=rking, OU=UsersOU, DC=contoso, DC=com
changetype: delete
```

Отметьте, что, в отличие от CSVDE, LDIFDE может настраивать пароли пользователей. Однако перед настройкой атрибута unicodePWD для учетной записи пользователя необходимо настроить шифрование SSL/TLS на контроллерах домена.

Вдобавок, LDIFDE может создавать и модифицировать любые объекты Active Directory, а не только учетные записи пользователей. Например, нижеследующий файл LDF создаст новое расширение схемы, именуемое EmployeeID-example, в схеме леса contoso.com:

```
dn: cn=EmployeeID-example,cn=Schema,
cn=Configuration,dc=contoso,dc=com
changetype: add
adminDisplayName: EmployeeID-Example
attributeID: 1.2.3.4.5.6.6.6.7
attributeSyntax: 2.5.5.6
cn: Employee-ID
instanceType: 4
isSingleValued: True
lDAPDisplayName: employeeID-example
```

Поскольку в файлах LDIFDE используется стандартный отраслевой формат файла LDAP, приложения от сторонних производителей, которым необходимо модифицировать схему Active Directory, часто поставляют файлы LDF, с помощью которых можно изучить и одобрить изменения, прежде чем применять их к производственной среде.

Помимо средств для операций пакетного импорта и экспорта, в состав Windows Server 2003 входит встроенный набор средств, позволяющий создавать, удалять и изменять различные объекты Active Directory, а также выполнять запросы к объектам, отвечающим определенным критериям. (Следует отметить, что данные средства, dsadd, dsrm, dsget, and dsquery, не поддерживаются Active Directory в Windows 2000.)

Dsadd

Dsadd используется для создания экземпляра класса объектов Active Directory в определенном разделе каталога. В число данных классов входят «пользователи», «компьютеры», «контакты», «группы», «структурные подразделения» и «квоты». У dsadd имеется общий синтаксис следующего вида:

```
dsadd <ObjectType> <ObjectDistinguishedName> attributes
```

Замечу, что каждый создаваемый тип объектов требует особого набора параметров, соотносящихся с атрибутами, доступными для этого типа. Эта команда создает один объект пользователя с различными заполненными атрибутами (отметьте, что все нижеследующее является одной строкой):

```
dsadd user cn=afuller,ou=IT,dc=contoso,dc=com
–samID afuller –fn Andrew –ln Fuller –pwd *
-memberOf cn=IT,ou=Groups,dc=contoso,dc=com “cn=Help Desk,ou=Groups,
dc=contoso,dc=com”
–desc “Marketing Director”
```

Параметр -memberOf требует полного различающегося имени (DN) каждой группы, к которой следует добавить пользователя, если его нужно добавить в несколько групп, можно добавить несколько DN, разделенных пробелами.

Если элемент, скажем DN группы «Служба поддержки», содержит пробел, этот элемент надо поместить в двойные кавычки. Если элемент, скажем структурное подразделение IT\EMEA, содержит обратную косую черту, эту черту нужно ввести дважды: IT\\EMEA. (Эти требования относятся ко всем средствам ds*.)

При использовании параметра -pwd * последует запрос на ввод пароля для пользователя в командной строке. Пароль можно указать внутри самой команды (-pwd P@ssword1), но тогда он будет отображен открытым текстом на экране или в любом текстовом файле либо файле сценария, в который вставлена команда.

Аналогично, можно создать объект группы и структурное подразделение при помощи следующих двух команд:

```
dsadd computer cn=WKS1,ou=Workstations,dc=contoso,dc=com
dsadd ou “ou=Training OU,dc=contoso,dc=com”
```

Dsmod

Dsmod используется для изменения существующих объектов, а работают с ним почти так же, как с dsadd, используя различные подменю и синтаксис, зависящие от типа изменяемого объекта. Нижеследующая команда dsmod изменяет пароль пользователя и модифицирует его учетную запись так, чтобы при следующем входе в систему ему был выдан запрос на смену пароля:

```
dsmod user “cn=afuller,ou=IT,dc=contoso,dc=com” –pwd P@ssw0rd1
–mustchpwd yes
```

Чтобы увидеть, насколько похожи эти параметры, взгляните на синтаксис dsadd, используемый для создания пользователя с теми же настроенными атрибутами:

```
dsadd user “cn=afuller,ou=IT,dc=contoso,dc=com” –pwd P@ssw0rd1
–mustchpwd yes
```

Очевидно, что, зная параметры для создания объектов при помощи dsadd, можно использовать их же для изменения пользователей при помощи dsmod.

Dsrm

Противоположностью dsadd является dsrm; как несложно вообразить, это средство используется для удаления объектов из командной строки. Базовый синтаксис dsrm достаточно прямолинеен: просто введите dsrm, а за ним — различающееся имя объекта, который следует удалить, примерно так:

```
dsrm cn=WKS1,ou=Workstations,dc=contoso,dc=com
```

По умолчанию dsrm выдаст запрос «Вы действительно хотите удалить этот объект?». Введите Y и нажмите кнопку Enter. Этот запрос можно отключить с помощью параметра –noprompt, но, очевидно, что в таком случае исчезнет шанс подтвердить перед удалением, что объект выбран верно,. Два дополнительных параметра могут быть полезны при удалении объекта-контейнера, то есть структурного подразделения, которое потенциально может содержать другие объекты. Следующая команда удаляет структурное подразделение TrainingOU и все содержащиеся в нем объекты:

```
dsrm ou=TrainingOU,dc=contoso,dc=com –subtree
```

А эта удаляет все дочерние объекты в TrainingOU, но не трогает само структурное подразделение:

```
dsrm ou=TrainingOU,dc=contoso,dc=com –subtree
–exclude
```

Dsmove

Для перемещения или переименования объекта в Active Directory используется средство dsmove, но следует отметить, что его можно использовать лишь для перемещения объектов внутри домена. Для переноса объектов между доменами или лесами используйте средство переноса Active Directory Migration Tool (ADMT), бесплатно загружаемое с веб-узла Майкрософт. Dsmove полагается на два параметра, которые можно использовать отдельно или вместе. Даная команда изменяет фамилию в учетной записи пользователя Steve Conn:

```
dsmove “cn=Conn, Steve,ou=IT,dc=contoso,dc=com”
–newname “Steve Conn”
```

Данная команда перемещает учетную запись Steve из структурного подразделения IT в подразделение Training:

```
dsmove “cn=Conn, Steve,ou=IT,dc=contoso,dc=com” –newparent
ou=Training,dc=contoso,dc=com
```

Переименование и перенос можно произвести в рамках одной операции, указав оба параметра разом:

```
dsmove “cn=Conn, Steve,ou=IT,dc=contoso,dc=com” –newname
“Steve Conn” –newparent ou=Training,dc=contoso,dc=com
```

Dsget и Dsquery

В состав набора средств командной строки ds* также входят два средства, используемые для запросов информации Active Directory, а не для создания или изменения объектов.

Dsget получает на входе различающееся имя (DN) объекта и выдает значение указанного атрибута или атрибутов. Dsget использует те же подменю, что dsadd и dsmod — «пользователь», «компьютер», «контакт», «группа», «структурное подразделение» и «квота».

Чтобы получить имя учетной записи SAM и код безопасности (SID) учетной записи пользователя, введите следующую команду (отметьте, что все нижеследующее является одной строкой):

```
dsget user cn=afuller,ou=IT,dc=contoso,dc=com
–samAccountName –sid
```

Результаты будут подобны показанным на Рис. 3.

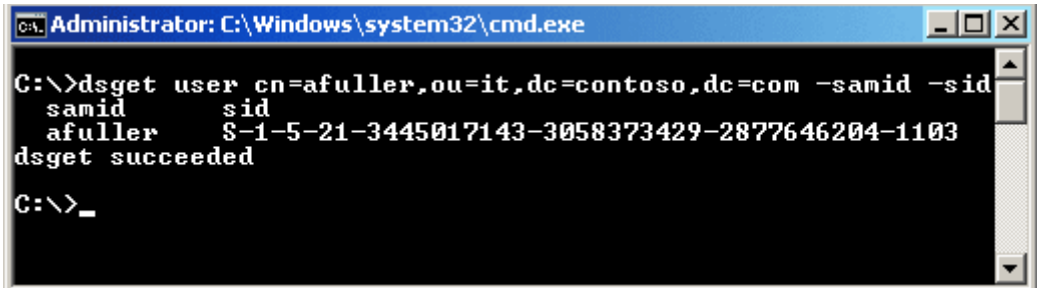

Рис. 3 Работа dsget

Dsquery возвращает список объектов Active Directory, отвечающих указанным критериям. Следующие параметры можно указать вне зависимости от используемого подменю:

```
dsquery <ObjectType> <StartNode> -s <Search Scope> -o <OutputFormat>
```

Dsquery может использовать следующие подменю, каждое со своим синтаксисом, для ObjectType: «компьютер», «контакт», «подсеть», «группа», «структурное подразделение», «веб-узел», «сервер» (следует отметить, что подменю сервера извлекает данные о контроллерах домена, а не о серверах в вашей среде), «пользователь», «квота» и «раздел». А если один из данных типов запросов не является тем, чем нужно, можно использовать подменю *, позволяющее ввести запрос LDAP свободной формы.

StartNode указывает местонахождение дерева Active Directory, в котором начнется поиск. Можно использовать конкретное DN, такое как ou=IT,dc=contoso,dc=com, или один из следующих описателей краткого пути: domainroot, начинающийся с корня определенного домена, или forestroot, начинающийся с корня корневого домена леса, используя сервер глобального каталога для выполнения поиска.

Наконец, параметр области поиска указывает, как средство dsquery должно производить поиск в дереве Active Directory. Опросы поддерева (вариант по умолчанию) обращаются к указанному StartNode и всем его дочерним объектам, одноуровневые опросы обращаются только к непосредственным дочерним объектам StartNode, и базовые опросы обращаются только к объекту StartNode.

Для лучшего понимания областей поиска представьте структурное подразделение (OU), содержащее как объекты пользователя, так и дочернее OU, которое также содержит дополнительные объекты. При использовании поддерева в качестве ообласти будет запрошено OU, все пользовательские объекты внутри него, дочернее OU и его содержимое. При одноуровневой области будет запрошены только пользователи, содержащие в OU, но не дочернее OU и его содержимое. При базовом запросе будет запрошено только само OU без запроса содержащихся в ней объектов.

Наконец, можно использовать выходной формат, чтобы контролировать форматирование результатов dsquery. По умолчанию dsquery возвращает различающиеся имена всех объектов, совпадающих с запросом, примерно так:

```
“cn=afuller,ou=Training,dc=contoso,dc=com”
“cn=rking,ou=ITTraining,ou=Training,dc=contoso,dc=com”
```

Чтобы запросить все объекты пользователей, содержащиеся в структурном подразделении IT и его дочерних OU, используйте следующее:

```
dsquery user ou=IT,dc=contoso,dc=com
```

Запрос можно сделать еще более точным, добавляя дополнительные параметры, такие как -disabled, возвращающий только отключенные учетные записи пользователей; -inactive x, возвращающий только пользователей, не подключавшихся в течении x или более недель; или -stalepwd x, возвращающий только пользователей, которые не меняли свои пароли в течении x или более дней.

В зависимости от числа объектов в каталоге может возникнуть необходимость указать параметр -limit x при запуске запроса. По умолчанию dsquery возвращает до 100 объектов, совпадающих с параметрами запроса; но

можно указать и большее число, такое как -limit 500, или использовать -limit 0, чтобы dsquery возвратило все совпадающие объекты.

Можно также использовать другие подменю для выполнения полезных запросов других типов объектов. Рассмотрим следующий запрос, возвращающий каждую подсеть, определенную в «Active Directory — узлы и службы», и входящую в пространство адресов 10.1.x.x:

```
dsquery subnet –name 10.1.*
```

А следующую команду можно использовать для возвращения каждой подсети, находящейся на веб-узле Corp:

```
dsquery subnet –site Corp
```

С помощью очередного подменю можно быстро определить, сколько контроллеров домена в лесу настроено для работы серверами глобального каталога:

```
dsquery server –forest –isgc
```

Можно также использовать данный синтаксис, чтобы упростить определение контроллера домена в определенном домене, содержащего роль FSMO эмулятора основного контроллера домена (PDC):

```
dsquery server –hasfsmo pdc
```

Как и в случае с другими командами ds*, включающими подменю, все параметры, доступные в конкретном подменю dsquery, можно просмотреть, войдя в командную строку и введя dsquery user /?, dsquery computer /?, dsquery subnet /?, и так далее.

Дополнительным хитрым приемом является передача исходящих данных dsquery по конвейеру в другое средство, такое как dsmod, при помощи знака | (SHIFT+обратная косая черта при английской раскладке клавиатуры). К примеру, компания переименовала отдел из «Подготовка» во «Внутреннее развитие», и теперь нужно обновить поле описания каждого относящегося к этому отделу пользователя. Одной командной строкой можно запросить все объекты пользователей, имеющие поле описания «Подготовка», и затем заменить это поле описания для всего пакета следующим образом:

```
dsquery user –description “Training” | dsmod
-description “Internal Development”
```

Некоторые находки от сторонних производителей

Поскольку Active Directory основана на стандартах LDAP, в ней можно создавать запросы и вносить изменения при помощи любого инструмента, понимающего LDAP. Многие сторонние поставщики выпустили платные средства для помощи в администрировании Active Directory, но порой можно найти и настоящие сокровища, которые распространяются бесплатно. Это, в частности, можно сказать про коллекцию, созданную обладателем звания MVP по службам каталогов Джо Ричардсом (Joe Richards) и доступной для загрузки на joeware.net/freetools. В ней можно найти многочисленные средства, служащие для решения различных задач. К трем из них я возвращаюсь постоянно — это adfind, admod и oldcmp.

Adfind и Admod

Adfind и admod подобны dsquery и dsmod; adfind является средством запроса с помощью командной строки для Active Directory, а admod может создавать, удалять или изменять объекты Active Directory.

В отличие от средств ds*, имеющих несколько подменю и различные параметры в зависимости от типа объекта, adfind и admod пользуются единым синтаксисом вне зависимости от типа выполняемого запроса или изменения. Базовый синтаксис для adfind:

```
adfind –b <Search Base> -s <Search Scope> -f <Search Filter>
attributesDesired
```

Запрос различающегося имени и описания всех объектов компьютеров в домене будет выглядеть как:

```
adfind –b dc=contoso,dc=com –s subtree –f (objectclass=computer) dn
description
```

Запрос всех объектов пользователей будет выглядеть как:

```
adfind –b dc=contoso,dc=com –s subtree –f “(&(objectcategory=person)
(objectclass=user))” dn description
```

Отметьте, что за исключением запроса содержимого LDAP, синтаксис не менялся.

Работая с adfind, можно найти несколько сокращенных вариантов записи параметров, которые избавляют от лишней работы по вводу. Например, параметр -default может заменить -b dc=contoso,dc=com в предыдущем примере и провести поиск по всему домену; -gc ищет, основываясь на сборке мусора (GC), и возвращает всех пользователей в вашем лесу Active Directory. Параметр -rb также можно использовать для установки относительной базы для поиска; если, скажем, необходимо найти структурное подразделение «Подготовка» в домене phl.east.us.contoso.com, то можно заметно сэкономить время, просто указав –default –rb ou=Training, вместо –b ou=Training, dc=phl,dc=east,dc=us,dc=contoso,dc=com.

Adfind может выполнять ряд функций расширенного поиска, которыми сложно управлять из командной строки без него, включая показанные на Рис. 4.

Figure 4 Параметры Adfind

Параметр	Описание
-showdel	Запрашивает контейнер удаленных объектов на предмет объектов-захоронений.
-bit	Запрашивает по знакам операции над битами, такие как атрибут user-AccountControl.
-asq	Выполняет запрос по атрибутам. Эта функция (которую нельзя воспроизвести в dsquery) может взять атрибут определенного объекта и провести запрос по нему.
-dsq	Передает результаты запроса adfind по конвейеру в dsmod или одно из прочих средств ds*.

Пример, использующий параметр -asq, будет запрашивать «Покажи мне членство в группах членов HelpDesk» следующим образом:

```
adfind –default –rb cn=HelpDesk,ou=IT –asq member memberOf
```

Admod, как следует из названия программы, используется для изменения объектов в Active Directory. Как и в случае adfind, в нем нет специализированных подменю со своими синтаксисами, которые надо запоминать; admod использует один и тот же синтаксис вне зависимости от типа обрабатываемого объекта. Admod также можно использовать для добавления, перемещения, переименования удаления и даже восстановления объектов путем простого добавления соответствующего параметра, скажем -add, -rm, -move, -undel. И точно так же, как в dsquery и dsmod, знак | можно использовать для передачи данных запроса adfind по конвейеру в admod.

Обратите внимание, что выполнение восстановления с помощью admod заключается в простой операции восстановления объекта-захоронения, в котором большинство атрибутов объекта уже удалено. Для полного восстановления объекта со всеми атрибутами потребуется провести принудительное восстановление объекта.

Oldcmp

Есть еще одно средство из коллекции программ Джо, которое я считаю незаменимой частью своего набора средств автоматизации: oldcmp, ищущее в базе данных Active Directory учетные данные компьютеров, которые не использовались в течение указанного числа недель, и способное проводить следующие действия:

• создавать отчеты об учетных записях без каких-либо действий в их отношении;
• отключать неиспользуемые учетные записи компьютеров;
• перемещать учетные записи компьютеров в иное, заранее указанное, структурное подразделение;
• полностью удалять учетные записи компьютеров.

Отмечу, что поскольку oldcmp может устроить серьезный разгром в каталоге, он снабжен несколькими встроенными функциями безопасности. Он не удаляет учетные записи, которые не были отключены ранее (если в командной строке вы не сказали: «Нет, я действительно хочу это сделать!»). Он не изменяет более 10 объектов за раз (если, опять же, обратное не указано особо), и он никогда не будет ничего делать с учетной записью компьютера контроллера домена.

К настоящему моменту Джо обновил oldcmp, так что он может выполнять подобные функции также и на учетных записях пользователей, которые не использовались в течение указанного отрезка времени.

Для небольшой среды Active Directory или среды, где работа идет лишь с одним-двумя дополнениями или изменениями за раз, средств с графическим интерфейсом, таких как «Active Directory — пользователи и компьютеры», может быть достаточно для повседневного администрирования, но при необходимости каждодневно добавлять или изменять большое количество объектов или простом желании найти более рациональное решение для задач администрирования переход на командную строку может намного ускорить процесс создания, изменения и удаления объектов в Active Directory. Как было показано выше, существует набор гибких и мощных бесплатных средств — как встроенных в Windows, так и распространяемых членами сообщества Active Directory. Любое из них способно намного повысить производительность работы администратора Active Directory, вместе же они становятся еще более важными для его повседневной работы.