Adresářové služby a LDAP

Pátek, 14.09.2007 14:11 | Samuraj - Petr Bouška | Microsoft admin

Tento článek pouze stručně popisuje adresářové služby, hlavně se zaměřením na Active Directory, a dále protokol pro komunikaci s adresářovými službami LDAP. V praxi se často zaměňují termíny adresář, adresářová služba a LDAP a článek vysvětluje vzájemné vazby mezi nimi. Jedná se o teoretický úvod k praktickému použití LDAPu v PHP.

Adresář

Adresář (directory) je databáze (hierarchická struktura), v které jsou uloženy informace o pojmenovaných objektech na síti, které jsou organizovány a sdružovány do skupin. Příkladem je počítač, tiskárna, služba, doména či uživatelský účet. ***Adresář*** se liší od ***relační databáze***, je navržen pro časté čtení a vyhledávání a pouze občasný záznam. V adresáři můžeme uchovávat data různých typů, jako text, digitální certifikát či obrázek. Můžeme omezovat přístup k záznamům pomocí ACL (Access Control List).

Adresářová služba

Adresářová služba (directory service - DS) je aplikace či skupina aplikací, které ukládají a organizují informace o uživatelích a zdrojích v počítačové síti. Adresářová služba přistupuje k ***adresáři***, občas označovanému jako ***adresářové repository*** (directory repository). Adresářová služba také funguje jako centrální ***autentizační autorita***, která umožňuje (bezpečnou) autentizaci zdrojů (uživatelů, služeb, počítačů).

Adresářová služba zprostředkovává informace z adresáře administrátorům, uživatelům, aplikacím apod. Adresářová služba vytváří fyzickou síťovou topologii a protokoly tak, aby vše bylo transparentní pro uživatele. Adresářová služba může být součástí operačního systému nebo také aplikace (např. elektronická pošta). Adresářové služby využívá většina síťových operačních systémů.

Adresářové služby jsou velmi vhodné pro určité účely, které jsou ve stylu kartotéky, tedy správa uživatelů, telefonních čísel a podobných dat. Stejně hojně se využívá vlastnosti pro autentizaci. Naopak oproti klasickým relačním databázím mají některé nevýhody, jako kontrola dat (referenční integrita apod.), složité dotazy, obsáhlé modifikace.

Rozšířeným příkladem adresářové služby je ***Active Directory*** (AD) firmy Microsoft. AD využívá LDAP (stejně jako většina adresářových služeb) a vyhovuje standardu LDAP v3 popsaném v RFC 3377.

LDAP

LDAP je zkratka pro ***Lightweight Directory Access Protocol***, což je aplikační protokol pro dotazování a modifikaci adresářových služeb nad TCP/IP. V 80tých letech vznikla skupina standardů **X.500** (DAP, DSP, DISP, DOP), které pokrývají adresářové služby. Zjednodušením standardu X.500 a zaměřením na TCP/IP vznikl LDAP.

LDAP používá ***LDAP Data Interchange Format*** (LDIF), což je standardizovaný textový formát pro výměnu dat. Data jsou při přenosu kódována pomocí ***Lightweight Basic Encoding Rules*** (LBER), to však není z důvodu bezpečnosti, ale nehomogenity prostředí, proto je velmi jednoduché data dekódovat.

LDAP je popsán pomocí čtyř modelů

- ***informační model - schéma*** - popisuje strukturu informací (atributy) v adresáři
- ***jmenný model*** - popisuje, jak jsou informace organizovány a odkazovány
- ***funkční model*** - popisuje, co může být uděláno s informacemi
- ***bezpečnostní model*** - jak jsou informace chráněny

Informační model - schéma

Informace v adresáři jsou uloženy ve stromové struktuře, která se označuje jako ***Directory Information Tree*** (DIT). Kořenem adresářového stromu je ***rootDSE***, která obsahuje globální informace o adresáři a nemá žádné jméno ani třídu. Informační model je založen na záznamech, které obsahují informace o nějakém objektu (něčem konkrétním), jako je uživatel či počítač. V Active Directory se ***LDAP záznamům*** říká **objekt**. Objekty jsou složeny ze skupiny ***atributů***, které mají vždy typ a jednu nebo více hodnot.

Implementace informačního modelu se označuje jako ***schéma***, což je sada objektů, které definují strukturu a obsah každého objektu, který může být vytvořen v adresářové službě. Schéma tedy definuje všechny možné třídy objektů a atributy. Výchozí schémata pro určitý adresář je možno rozšiřovat, příkladem je doplnění Exchange serveru do AD, které rozšíří schéma dalšími atributy potřebnými pro poštovní služby.

Třídy objektů (object classes) jsou kategorie objektů, které mohou být vytvořeny v adresáři. V LDAPu se používá označení ***objectClass*** a může se jednat například o *user*, *computer*, *organizationalUnit*, *domain*, *container*, *group*. Třídy objektů jsou zařazeny do jedné ze tří kategorií *Structural*, *Abstract* a *Auxiliary*.

> ***Pozn.:*** *Objekt může být zařazen do více tříd naráz. Příkladem je* ***uživatel*** *(user) v AD, který je zařazen do tříd top, person, organizationalPerson a user. Nebo* ***počítač*** *(computer) je v top, person, organizationalPerson, user, computer.*

Protože bývá objekt zařazen do několika *objectClass*, tak je hledání méně efektivní a nemusí být přesné (například při hledání podle objectClass=user se naleznou i počítače). Proto můžeme použít hledání podle ***kategorie objektů***, tedy atributu ***objectCategory***. Na rozdíl od *objectClass* má ***objectCategory*** pouze jednu hodnotu (je to classSchema objekt), což by mělo odkazovat na nejvíce specifickou třídu v hiearchii tříd objektů. Může se jednat třeba o *user*, *computer*, *group*, *organizationalUnit*.

> ***Pozn.:*** *Každý objekt classSchema má atribut defaultObjectCategory, což je pro většinu tříd sama třída. Když při hledání zadáme objectCategory = X, tak X je ldapDisplayName řídy. LDAP automaticky expanduje tento výraz na objectCategory = defaultObjectCategory třídy X. Atribut objectCategory se ukládá ve tvaru distinguished name a LDAP provádí automaticky konverzi do tohoto tvaru.*

Atributy objektů (object attributes) jsou charakteristiky (vlastnosti) objektů. Atribut může obsahovat jednu nebo více hodnot, například *jméno, příjmení, e-mail*. Určité atributy patří k určité třídě objektů a schéma také definuje, které hodnoty musí být vyplněny a které jsou volitelné. Schéma také určuje, jaké typy hodnot může atribut nabývat, například *textový řetězec, celé číslo*.

Podle toho, kde se nachází objekt ve stromové struktuře, se jedná buď o ***list*** (leaf object, nemá žádné potomky) nebo o ***kontejner*** (container object). Kontejner v sobě může obsahovat jeden či více objektů.

Následující tabulka ukazuje několik běžných atributů používaných v AD.

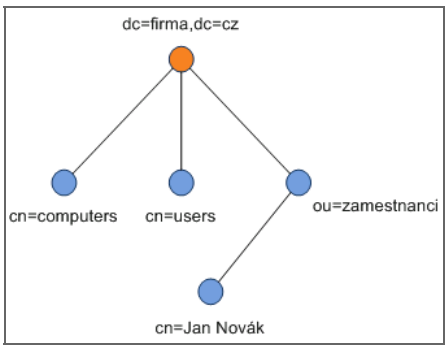
jméno	popis
sAMAccountName	SAM Account Name, přihlašovací uživatelské jméno, které podporuje starší systémy
sAMAccountType	typ účtu
userPrincipalName	UPN, přihlašovací jméno uživatelského účtu ve tvaru *<user>@<DNS-domain-name>*
displayName	jméno, které využívají aplikace (třeba Exchange)
givenName	křestní jméno
sn	surname - příjmení
description	popis
mail	adresa elektronické pošty
company	jméno společnosti
department	oddělení ve firmě
location	umístění
streetAddress	ulice
memberOf	seznam skupin, kterých je členem

Jmenný model

Distinguished Name - DN

Pro identifikaci objektů se používá ***Distinguished Name (DN)***, což je jednoznačný identifikátor objektu a obsahuje úplnou cestu k záznamu (pozici ve stromě). DN se skládá ze jména objektu a jmen jednotlivých kontejnerů a domén, které obsahují objekt, oddělených čárkou. Jednotlivé položky obsahují název atributu a přiřazenou hodnotu atributu, třeba `ou=zamestnanci`.

Následující obrázek ukazuje příklad části adresáře AD pro doménu `firma.cz`. V kontejneru (organizační jednotce) *zaměstnanci* je umístěn uživatel *Jan Novák*, pro kterého je `DN = cn=Jan Novák,ou=zamestnanci,dc=firma,dc=cz`.



Občas nepotřebujeme specifikovat celou cestu k objektu a můžeme pak použít ***Relative Distinguished Name (RDN)***, které je relativní a jednoznačné v daném kontejneru. Jedná se o poslední část DN, pro našeho uživatele je `RDN = cn=Jan Novák`.

Další možnosti identifikace

Běžnou identifikací objektu v LDAPu a také v AD je použití DN, ale existují i další metody speciálně pro Active Directory.

Můžeme použít **OID** (Object Identifiers), což je hierarchický, unikátní identifikátor, složený s dekadických číslic oddělených tečkou. Jedná se o stejný identifikátor jako u SNMP. Je běžný u X.500.

V ***Active Directory*** má každý objekt přiřazen jednoznačné 128-bitové číslo, které se označuje **GUID** (globally unique identifier). Toto číslo je stálé a nemění se při přesunu objektu v rámci lesa.

Active Directory také používá obdobu DN, která se označuje jako ***AD canonical name*** a příkladem je zápis `firma.cz/zamestnanci/Jan Novák`.

Jmenné atributy (Naming Attributes)

Každá část **DN** je vyjádřena pomocí `typ_atributu=hodnota`. Typ atributu, který se používá k popisu RDN, se označuje jako ***jmenný atribut***. Každá třída má přiřazen jmenný atribut, například *User* (uživatel) má `cn`. Následující tabulka ukazuje jmenné atributy pro LDAP a jejich ekvivalent pro AD.

LDAP atribut	jméno	AD atribut	jméno
CN	Common Name	CN	Common Name
OU	Organization Unit	OU	Organization Unit
O	Organization	DC	Domain Component
C	Country	-	-

Funkční model

Funkční model LDAPu definuje, co se může provádět s informacemi v adresáři. Jedná se o 9 operací, které jsou zařazeny do 3 funkčních oblastí.

oblast	operace	popis

autentizace (Authentication)	bind	inicializuje spojení, vyjednává o metodě autentizace, autentizuje
	unbind	ukončí session
	abandon	klient žádá o ukončení posílání výsledků na poslední dotaz
dotazování (Interrogation)	search	výběr dat z určitého regionu pomocí filtru
	compare	porovná hodnotu atributu se zadanou hodnotou
aktualizace (Update)	add	vytvoří nový objekt
	modify	upraví atributy záznamu (vytvořit, smazat, upravit)
	modify RDN	slouží k přesunutí objektu v rámci stromu adresáře
	delete	smazání záznamu

> ***Pozn.:*** *Active Directory podporuje několik dalších operací, které nejsou definovány v RFC.*

Dotazy pomocí LDAPu

Operace `search` má řadu vstupních parametrů. Jedním z nich je ***výchozí bod hledání***, který určuje kontejner, od kterého níže se bude prohledávat. Dalším důležitým parametrem je ***filtr***.

Ve ***filtrech*** můžeme použít matematické operace ***shoda*** (atribut=hodnota), ***větší než*** (atribut>=hodnota), ***menší než*** (atribut<=hodnota) a ***přibližná shoda*** (atribut~=hodnota). Jednotlivé filtry můžeme kombinovat pomocí logických operací, používá se zápis `(operator(filter1)(filter2)(filter3))`. Logické operace jsou ***AND*** (&(filter1)(filter2)), ***OR*** (|(filter1)(filter2)) a ***NOT*** (!(filter1)(filter2)). Také můžeme použít zástupný znak `*` pro libovolné znaky.

Příklady vyhledávacích filtrů:

```
(objectCategory=*)                  // všechny objekty
(&(objectClass=user)(!(cn=susan)))  // všichni uživatelé mimo Susan
```

Bezpečnostní model

Poslední model LDAPu určuje, jak se přistupuje k datům z bezpečnostního hlediska. Souvisí s autentizačními službami z adresářových služeb.

Nástroj ldp.exe

Pokud se díváme do Active Directory, například pomocí ***Active Directory Users and Computers***, tak se záznamy standardně zobrazují pomocí AD canonical name. Pokud bychom chtěli vidět AD z pohledu LDAPu, tak můžeme použít nástroj od Microsoftu, který se nachází v ***Support Tools*** a jedná se o `ldp.exe`. Tento nástroj je zajímavý s řadou funkcí. Hodí se pro testování řady věcí, jako je například vyhledávání (a definování filtrů).

Odkazy

- **Použití adresářových služeb v informačních systémech** - Na českém internetu jsem narazil na velmi pěkný zdroj informací a to je diplomová práce Karla Beňáka
- **Využití LDAPu v praxi** - další zajímavý český zdroj
- **How Active Directory Searches Work** - velice pěkný článek popisující LDAP s přihlédnutím k AD a následné vyhledávání
- **Active Directory Architecture**
- **Default Active Directory Attributes in the Windows 2000 Schema**
- **Active Directory Attributes in Windows 2000 and Windows 2003**

Autor: **Petr Bouška**

Autentizace v LDAPu (AD)

Neděle, 28.10.2007 14:32 | Samuraj - Petr Bouška | Microsoft admin

Tento článek navazuje na předchozí díl, který se věnoval Adresářovým službám a LDAPu. Zde se stručně pojednává o bezpečnostním modelu LDAPu a autentizačních možnostech adresářových služeb a zaměřuje se na speciality Active Directory. Jinak řečeno, jsou zde popsány způsoby, jakými se ověřuje uživatel vůči doméně.

Autentizace

Autentizace je mechanismus, který ***ověřuje identitu uživatele*** (a nejen uživatele, ale i třeba počítače nebo služby). Měl by tedy potvrdit, že jsme osoba, za kterou se vydáváme. Většinou se používá znalost jména a hesla, v lepším případě se používá certifikát či biometrie. Jako *jednofaktorová* se označuje autentizace, kdy potřebujeme pouze něco vědět, tedy zadáváme heslo. *Dvoufaktorová* (případně vícefaktorová) je pak autentizace za pomoci čipové karty (certifikátu) a PINu, musíme něco mít a něco znát.

Bezpečnostní model LDAPu

V *LDAP v3* (RFC 2251) je specifikováno, že pro bezpečnostní služby je možno použít ***SASL mechanismus***. SASL znamená ***Simple Authentication and Security Layer*** a byl původně definován v RFC 2222 a nově v RFC 4422. V LDAP stromě v kořenovém záznamu ***rootDSE*** jsou v atributu *supportedSASLMechanisms* vyjmenovány podporované SASL bezpečnostní metody daného adresáře.

SASL

SASL je metoda pro přidání podpory pro autentizaci do protokolů na bázi připojení. SASL odděluje autentizační mechanismus od aplikačních protokolů, takže teoreticky aplikace, která podporuje SASL může použít libovolný autentizační mechanismus. SASL vyjednává o tom, který mechanismus je nejlepší použít (a který je možno použít). Jednotlivé mechanismy jsou pojmenovány řetězcem s max. 20 znaky a používají velká písmena, čísla a pomlčku s podtržítkem. SASL se často používá v kombinaci s ***TLS*** (Transport Layer Security).

> ***Pozn.:*** *Jednotlivé mechanismy jsou definovány v různých RFC.*

SASL mechanismy, které podporuje Active Directory:

- **PLAIN** - autentizace s jednoduchým heslem v čistém textu
- **ANONYMOUS** - neautentizovaný přístup, nezasílají se žádné autentizační informace
- **NTLM** - *NT LAN Manager*, používá se na samostatných systémech nebo pro zpětnou kompatibilitu. Používá šifrované heslo spolu s doménou a jménem. Jedná se o Microsoftí protokol, částečně zpětně kompatibilní s LANMAN a objevil se spolu s SMB protokolem. Dnes je NTLMv2 a upřednostňován je Kerberos, ale NTLM se stále používá v některých případech.
- **DIGEST-MD5** - tento mechanizmus umožňuje autentizaci pomocí hesla, ale po síti se posílá pouze hash hesla. Je podobný jako CRAM-MD5, ale navíc zajišťuje integritu a důvěrnost spojení.
- **GSSAPI** - *Generic Security Services Application Program Interface*, RFC 2078, poskytuje možnost autentizovat se pomocí ***Kerberos*** *v5 session*, také poskytuje mechanizmy pro zaručení integrity a důvěrnosti spojení. Nejedná se o autentizační metodu, ale o mechanismus, který vybere nejvhodnější metodu a předává autentizační informace této metodě (službě). Poskytuje všeobecné bezpečnostní služby, které mohou být podporovány řadou podřazených mechanismů a služeb. Podporuje využití ***SSO*** (single sign on), pak se nevyplní credentials.

> ***Pozn.:*** *Anonymní přístup, tedy bez zadání hesla, je v AD defaultně zakázaný.*

Další *SASL* mechanismy jsou třeba *OTP* (one-time password), *SKEY, EXTERNAL, CRAM-MD5, SRP, GSS-SPNEGO, SECURID.*

Active Directory používá ještě několik proprietálních mechanizmů pro autentizaci:

- **SSPI** - *Security Support Provider Interface* - kvůli zpětné kompatibilitě, dnes se nedoporučuje používat, jednalo se o obdobu GSS-API.
- **DPA** - *Distributed Password Authentication* - používá se v Microsoft Commercial Internet Systém, zašifrované heslo.

> ***Pozn.:*** *Bezpečnostní model LDAPu neobsahuje mechanismus pro řízení přístupu, ale AD používá* ***ACL*** *(access control lists) na adresářové objekty.*

Kerberos V5 (Kerberos Network Authentication Service (V5)) je popsán v RFC 1510. Jedná se o nejpoužívanější metodu pro Active Directory. Definuje autentizační proces, který poskytuje metody pro ověření identity, například pro pracovní stanici či uživatele. Pro autentizaci klienti používají ***Kerberos tickets*** (lístky), které reprezentují síťové *credentials* (pověření) pro klienta. Klient získá ticket od ***KDC*** (Kerberos Key Distribution Center) a ukazuje tento lístek, když se vytváří síťové spojení. Kerberos představuje identitu klienta pomocí jména domény, uživatelského jména a hesla.

LDAP bind

Pokud chceme pracovat s daty adresáře, tak se musíme nejprve ověřit (autentizovat vůči adresáři). Toto ověření se provádí LDAP operací `bind`. Pokud chceme pouze autentizovat klienta vůči adresáři (například Active Directory) pro účely další aplikace (a ne pro přístup do adresáře), tak se běžně využívá operace `bind` a po jejím použití se hned odpojíme (unbind). Jinou možností je použití ***RADIUS*** (Remote Authentication Dial In User Service) serveru.

> ***Pozn.:*** *Připomínám, že pokud se zadá pouze jméno a prázdné heslo, tak AD předpokládá, že se jedná o anonymní přístup a vrátí, že ověření proběhlo v pořádku.*

Pro autentizaci operací `bind` máme dvě hlavní možnosti:

- **Simple bind** - používá *simple authentication*, pro autentizaci se používá DN jméno uživatele a heslo v čistém textu.
- **SASL bind** - SASL může použít řadu typů SASL mechanismů a k tomu patřičné credentials, například protokol Kerberos nebo klientský certifikát pomocí TLS.

Pokud použijeme ***Simple bind***, tak může dojít k odchycení hesla, proto by se mělo používat vždy ve spolupráci se šifrovaným kanálem pomocí SSL nebo TLS.

Odkazy

- Directory Server Standards and Specifications - znění některých RFC
- Logon and Authentication Technologies - metody autentizace ve Windows

Autor: Petr Bouška

Související články:

Active Directory a protokol LDAP

Správa počítačové sítě ala Microsoft, to je Active Directory. Jedná se o velice rozsáhlou skupinu technologií a služeb. Základem jsou adresářové služby, adresáře a komunikační protokol LDAP.

- Adresářové služby a LDAP [14.09.2007 14:11]
- Autentizace v LDAPu (AD) [28.10.2007 14:32] ..... **právě čtete**
- Jak na LDAP a LDAPS v PHP pod Windows [02.11.2007 16:52]
- Autentizace uživatele vůči AD v PHP [06.11.2007 18:18]
- Rozšíření Active Directory Users And Computers o editaci employeeID [03.12.2007 14:25]
- Active Directory komponenty - domain, tree, forest, site [08.02.2008 19:01]
- Převod domény z Windows Server 2003 na Windows Server 2008 R2 [31.03.2010 08:47]
- Kerberos protokol a Single sign-on [15.09.2010 18:33]
- Kerberos SSO - nastavení Internet Exploreru a Firefoxu [16.09.2010 15:53]
- Kerberos SSO v PHP aplikaci s Apachem na Linuxu [22.09.2010 17:58]
- Active Directory - fotografie u uživatelů nejen pro Outlook 2010 [20.10.2010 09:55]
- Auditování AD DS objektů ve Windows Server 2008 [26.03.2011 19:23]
- Active Directory Recycle Bin [31.03.2011 15:48]
- Exchange Server, Outlook a certifikáty v GALu [27.12.2011 14:48]
- Migrace replikací SYSVOLu z FRS na DFSR [13.03.2012 15:47]
- Windows Server 2012 Active Directory [01.10.2012 17:48]
- Microsoft Certification Authority Auditing [13.01.2014 17:05]
- Kerberos autentizace a členství ve skupinách [16.01.2014 10:22]
- Kerberos část 6 - Kerberos SSO mezi doménami [23.04.2014 15:03]
- Kerberos část 7 - Troubleshooting Kerberos SSO [03.05.2014 11:30]
- Kerberos část 1 - Active Directory komponenty [13.06.2014 14:28]
- Kerberos část 2 - AD uživatelské účty a Service Principal Name [16.06.2014 08:51]

Jak na LDAP a LDAPS v PHP pod Windows

Pátek, 02.11.2007 16:52 | Samuraj - Petr Bouška | webdesign

Tento článek slouží jako upozornění na možnosti, které nám dává PHP. V PHP můžeme přistupovat do adresářových služeb, například Active Directory, což se hodí pro řadu aplikací (třeba na intranetu). Také uvádí postup, jak rozchodit využití LDAPS pod Windows.

Teoretický popis LDAPu je v článku Adresářové služby a LDAP. Pokud se na problematiku podíváme velmi přibližně, tak adresář je databáze a LDAP jsou funkce pro přístup k databázi. Proto v teoretickém použití LDAPu z PHP není problém. Pro PHP existuje rozšíření (extension) `php_ldap`, které implementuje operace LDAPu.

LDAP

Pro zprovoznění LDAPu v PHP na operačním systému Windows, potřebujeme pouze zapnout PHP extension `php_ldap`. To provedeme přidáním řádku do sekce *extension* v souboru `php.ini` (případně jeho odkomentováním).

```
extension=php_ldap.dll
```

Potom již můžeme jednoduše přistupovat k LDAPu. Popis funkcí je v manuálu LDAP Functions, je zde implementováno všech 9 standardních operací z ***LDAP v3***. Některé jsou rozděleny do více funkcí a je zde řada doplňujících funkcí pro jednodušší použití. Velice stručně se zmíním o základních funkcích.

ldap_connect()
Vytvoří spojení na zadaný LDAP server, vrací ID linku nebo false. Pokud nespecifikujeme, tak se pro LDAP použije standardní port 389.

ldap_set_option()
Nastaví hodnotu specifikované vlastnosti, třeba verze LDAPu.

ldap_bind()
Navázání k adresáři pomocí ID linky. Provádí se autentizace podle jména a hesla, pokud se nezadá, tak se pokusí o anonymní připojení. Většinou se tak chová i v případě, pokud se vyplní pouze jméno a ne heslo. Server pak často vrátí true, to se stane i v případě AD, i když anonymní připojení není povoleno (a další přístup k adresáři není funkční). Jméno by se mělo zadávat pomocí *DN* (CN=Uzivatel,OU=Users,DC=firma,DC=local) , ale pro AD funguje i zadání pouze *username* (uzivatel) nebo *UPN* (uzivatel@firma.local).

ldap_search()
Operace search z LDAPu. Zadáváme výchozí bod hledání a filtr, další parametry jsou volitelné. Vrací ID výsledku nebo false.

ldap_get_entries()
Jedna z možností jak získat záznamy vrácené funkcí search (a dalších). Vrací vícerozměrné pole, kde jsou jednotlivé záznamy a pro každý záznam jeho atributy.

ldap_add()
Vložení nového záznamu s daným DN a určenými atributy.

ldap_compare()
Porovnání atributu s hodnotou pro daný záznam.

ldap_close()
Korektní ukončení session, jedná se pouze o alias k ldap_unbind().

Ještě uvádím jednoduchý příklad, který se připojí k AD pod uživatelem a provede jednoduché hledání a výpis pár hodnot. Pro správné použití chybí ošetření řady událostí apod.

```
<?php
function ldapQuery($filter, $baseDN = "OU=firma,DC=firma,DC=local", $server = "192.168.0.1") {
  $ldapCon = ldap_connect($server);
  if($ldapCon) {
    ldap_set_option($ldapCon, LDAP_OPT_PROTOCOL_VERSION, 3);    // AD podporuje LDAPv3
    ldap_set_option($ldapCon, LDAP_OPT_REFERRALS, 0);           // pro AD se doporučuje vypnout referrals
    $res = ldap_bind($ldapCon, "CN=Uzivatel,OU=Users,DC=firma,DC=local", "Password");
    if($res === false) { echo "<p>Autentizace selhala.</p>";  ldap_close($ldapCon); return false; }
    $rec = ldap_search($ldapCon, $baseDN, $filter);
    echo "<p>Počet vyhledaných záznamů: " . ldap_count_entries($ldapCon, $rec) . "</p>";
    $info = ldap_get_entries($ldapCon, $rec);
    for($i=0; $i<$info["count"]; $i++) {
      echo "DN: ".iconv("UTF-8", "ISO-8859-2", $info[$i]["dn"])."<br />";
      echo "První CN: ".iconv("UTF-8", "ISO-8859-2", $info[$i]["cn"][0])."<br />";
      echo "První email: ".$info[$i]["mail"][0]."<br /><hr />";
    }
    ldap_close($ldapCon);
  } else { echo "<p>Nepodařilo se připojit k LDAP serveru.</p>"; ldap_close($ldapCon); return false; }
}
ldapQuery("(memberOf=CN=Skupina,OU=firma,DC=firma,DC=local)");
?>
```

Pozn.: *Adresářové služby používají kódování UTF-8, takže pokud na stránce máme nějaké jiné, musíme hodnoty převádět.*

Pozn.: *Pokud v AD používáme v názvech (v DN) diakritiku, tak v příkazech můžeme použít znaky bez diakritiky (a nemusíme pak řešit problém s kódováním).*

LDAPS

Při běžném použití LDAPu probíhá komunikace mezi webovým serverem a adresářovým serverem (budu uvažovat AD) v otevřené formě a při zachycení komunikace je vše čitelné. Největší nebezpečí je u autentizace operací `bind`.

Pokud je pro nás problém pouze autentizace, tak můžeme použít ***SASL metodu autentizace***. Pokud nám jde o zabezpečení veškerých dat, tak potřebujeme šifrovat veškerou komunikaci a to můžeme buď pomocí **SSL** nebo **TLS**. Já se budu věnovat **SSL**. LDAP, který běží nad SSL, se označuje jako **LDAPS**.

> ***Pozn.:*** *Pokud jsou ještě nějaká důležitá data (například právě autentizace uživatele do AD) posílána z klienta na webový server, tak je dobré šifrovat pomocí SSL i tuto komunikaci (HTTPS).*

Jak rozchodit LDAPS

1. LDAPS používá stejnou knihovnu jako LDAP, takže nejprve musíme zapnout extension `php_ldap.dll`
2. Protože chceme používat SSL, tak musíme rozchodit Apache se SSL, to je popsáno v článku **Apache server (EasyPHP) se SSL na Windows**.
3. Musíme mít zprovozněné LDAPS na adresářovém serveru, takže AD server musí mít certifikát.
4. Knihovna `php_ldap.dll` má v sobě napevno specifikovánu cestu ke svému konfiguračnímu souboru, která je `C:\openldap\sysconf\ldap.conf`. Musíme tedy vytvořit tyto adresáře a do nich soubor, do kterého vložíme následující řádek

```
TLS_REQCERT never    // nekontroluje se certifikát serveru
```

A to je vše, nyní již můžeme využívat šifrované LDAPS. Abychom použili LDAPS v PHP, tak stačí ve funkci `ldap_connect` použít adresu serveru `ldaps://192.168.0.1`, případně s určením standardního portu 636 `ldap_connect("192.168.0.1", 636)`.

Výše uvedený postup je zjednodušený, nepoužíváme certifikát klienta a ani nekontrolujeme certifikát serveru. Většina návodů na internetu obsahuje kroky, kde se stáhne certifikát kořenové autority, od které má adresářový server svůj certifikát. Ten se pomocí ***OpenSSL*** převede do formátu PEM a přidá se do konfiguračního souboru `ldap.conf` pomocí parametru `TLS_CACERT`.

> ***Pozn.:*** *Narazil jsem na problém, když startoval Apache a `php_ldap.dll` se snažilo načíst certifikát, tak celý server spadnul. Pomohlo přehrání knihovny `php_ldap.dll` z poslední verze PHP (php-4.4.7-Win32.zip).*

Ladění

Pro řešení problémů s LDAPem můžeme zapnout ***debugovací mód***. To provedeme zadáním následujícího příkazu do kódu ještě před `ldap_connect`.

```
ldap_set_option(NULL, LDAP_OPT_DEBUG_LEVEL, 7);
```

Následně se do logu `errors.log` Apache serveru zapisují podrobně všechny prováděné operace s LDAPem.

Například pokud používáme LDAPS a nemáme správně certifikát či nepoužíváme `TLS_REQCERT never` (třeba když vůbec neexistuje ldap.conf), tak se v logu objeví následující chyba a neproběhne bind.

```
TLS certificate verification: Error, unable to get local issuer certificate
TLS trace: SSL3 alert write:fatal:unknown CA
```

Pokud nezapneme debug, tak se nedozvíme, proč se bind nepodařil.

Autor: **Petr Bouška**

Související články:

Active Directory a protokol LDAP

Správa počítačové sítě ala Microsoft, to je Active Directory. Jedná se o velice rozsáhlou skupinu technologií a služeb. Základem jsou adresářové služby, adresáře a komunikační protokol LDAP.

- **Adresářové služby a LDAP** [14.09.2007 14:11]
- **Autentizace v LDAPu (AD)** [28.10.2007 14:32]
- **Jak na LDAP a LDAPS v PHP pod Windows** [02.11.2007 16:52] ..... **právě čtete**
- **Autentizace uživatele vůči AD v PHP** [06.11.2007 18:18]
- **Rozšíření Active Directory Users And Computers o editaci employeeID** [03.12.2007 14:25]
- **Active Directory komponenty - domain, tree, forest, site** [08.02.2008 19:01]
- **Převod domény z Windows Server 2003 na Windows Server 2008 R2** [31.03.2010 08:47]
- **Kerberos protokol a Single sign-on** [15.09.2010 18:33]
- **Kerberos SSO - nastavení Internet Exploreru a Firefoxu** [16.09.2010 15:53]
- **Kerberos SSO v PHP aplikaci s Apachem na Linuxu** [22.09.2010 17:58]
- **Active Directory - fotografie u uživatelů nejen pro Outlook 2010** [20.10.2010 09:55]
- **Auditování AD DS objektů ve Windows Server 2008** [26.03.2011 19:23]
- **Active Directory Recycle Bin** [31.03.2011 15:48]
- **Exchange Server, Outlook a certifikáty v GALu** [27.12.2011 14:48]
- **Migrace replikací SYSVOLu z FRS na DFSR** [13.03.2012 15:47]
- **Windows Server 2012 Active Directory** [01.10.2012 17:48]
- **Microsoft Certification Authority Auditing** [13.01.2014 17:05]
- **Kerberos autentizace a členství ve skupinách** [16.01.2014 10:22]
- **Kerberos část 6 - Kerberos SSO mezi doménami** [23.04.2014 15:03]
- **Kerberos část 7 - Troubleshooting Kerberos SSO** [03.05.2014 11:30]
- **Kerberos část 1 - Active Directory komponenty** [13.06.2014 14:28]
- **Kerberos část 2 - AD uživatelské účty a Service Principal Name** [16.06.2014 08:51]

Autentizace uživatele vůči AD v PHP

Úterý, 06.11.2007 18:18 | Samuraj - Petr Bouška | webdesign

Tento článek navazuje na minulé ukázky použití LDAPu v PHP. Jednou užitečnou vlastností, kterou nám přináší přístup přes LDAP k adresářovým službám, je možnost provedení autentizace uživatele. V praxi tak můžeme, třeba na intranetu, autentizovat uživatele jeho doménovým účtem proti Active Directory. Článek ukazuje jednoduchý příklad, jak to provést. Samozřejmě lepší by ještě bylo využití Single Sign On, ale to již není tak jednoduché.

Článek navazuje na teoretický úvod Adresářové služby a LDAP a Autentizace v LDAPu (AD). A praktické ukázky použití LDAPu v Jak na LDAP a LDAPS v PHP pod Windows. Věnuje se prakticky možnosti, jak autentizovat uživatele vůči Active Directory pomocí kódu v PHP.

Jak bylo zmíněno v článku o autentizaci v LDAPu, tak pro autentizaci uživatele můžeme v praxi použít právě protokol **LDAP** a jeho operaci `bind`. Jinou možností by byla nějaká externí služba, například ***RADIUS server***. Pomocí ***LDAP bind*** máme dvě možnosti, buďto ***simple bind*** nebo ***SASL bind***.

Simple bind

Nejjednodušší metoda, ale také nejméně bezpečná, je použití ***simple bind***. V tomto případě se veškerá (autentizační) data přenáší v plain textu, tedy čistém a nešifrovaném textu. Adresářovým službám zasíláme uživatelské jméno ve tvaru **DN** (Distinguished Name), tedy třeba `CN=Uzivatel,OU=Users,DC=firma,DC=local`. V praxi jsem ověřil, že pro *Active Directory* je možno použít jméno i v **UPN** tvaru (User Principal Name) , tedy `uzivatel@firma.local`, nebo ve tvaru `doména\uživatelské jméno`.

Protože se heslo posílá v čistém textu, tak se určitě doporučuje použít přenos přes šifrovaný kanál. Můžeme použít **SSL** nebo **TLS**. Já používám SSL a tedy ***LDAPS***.

> ***Pozn.:*** *Důležité upozornění je, že je třeba kontrolovat, zda uživatel nezadal prázdné heslo, protože AD by pak uvažovalo anonymní přihlášení a vrátilo by true.*

```
function authUserAD($username, $password, $DomainName="firma.local", $ldap_server="ldaps://192.168.0.1") {
  $auth_user = $username."@".$DomainName;
  if($connect = ldap_connect($ldap_server)){
    ldap_set_option($connect, LDAP_OPT_PROTOCOL_VERSION, 3);
    ldap_set_option($connect, LDAP_OPT_REFERRALS, 0);
    if(ldap_bind($connect, $auth_user, $password)) {
      ldap_close($connect);
      return(true);
    }
  }
  ldap_close($connect);
  return(false);
}
if(authUserAD("uzivatel", "Heslo")) echo "<p>Přihlášení je OK.</p>";
else echo "<p>Přihlášení se nezdařilo.</p>";
```

SASL bind

Podle poznámek u LDAP funkcí, PHP podporuje SASL. Je potřeba, aby bylo PHP zkompilováno s podporou SASL a vyžaduje knihovny libeay32.dll, ssleay32.dll (případně ještě libsasl.dll). PHP verze 5 obsahuje funkci (v podstatě nedokumentovanou) `ldap_sasl_bind`. Tu se mi však nepodařilo rozchodit, nezískal jsem pro Windows verzi PHP se SASL podporou.

Další možnosti

Jinou možností je využít nějaký modul pro ***Apache server*** (nebo použít IIS). Navíc tyto moduly většinou umožňují využít **SSO** (Single Sign On), pokud to podporuje klient.

- mod_ntlm
- mod_ntlm2
- mod_auth_ntlm_winbind
- mod_auth_sspi
- mod_auth_gssapi
- kerb_auth_mod
- mod_auth_kerb

Autor: Petr Bouška

Související články:

Rozšíření Active Directory Users And Computers o editaci employeeID

Upraveno 03.12.2007 14:25 | vytvořeno 03.12.2007 14:25 | **Samuraj - Petr Bouška** | ***Microsoft admin***

Tento článek se věnuje relativně jednoduchému postupu, jak rozšířit možnosti administračního nástroje Active Directory Users And Computers o editaci dalšího atributu. Jedná se o atribut, který je součástí schématu AD, ale nemá podporu v ADUC. V příkladu je uveden atribut employeeID (číslo zaměstnance), ale je možno obdobně aplikovat na libovolný jiný. V prvním kroku se vytváří VB skript, který přistupuje k hodnotě. V druhém kroku se rozšiřuje kontextové menu ADUC.

> ***Pozn.:*** *Toto řešení není žádná novinka a na internetu naleznete řadu anglických návodů, já jsem pouze dodal českou verzi.*

Adresář ***Active Directory*** obsahuje celou řadu atributů, některé byly přidány od verze Windows Server 2003. AD je kompatibilní s LDAP adresářem a proto obsahuje různé atributy definované pro LDAP, které však standardně nevyužívá. Pomocí Snap-In do MMC konzole ***Active Directory Users And Computers*** (dále jen ADUC) můžeme zobrazovat a editovat pouze některé atributy.

Pro přístup ke všem atributům můžeme využít nějaký nástroj (třeba `ldp.exe`) či programovací jazyk, které využívá protokolu LDAP. Nebo můžeme použít MS nástroj ***ADSIEdit***, který se nachází ve Windows Server 2000/2003 ***Support Tools*** (například Windows Server 2003 Service Pack 2 32-bit Support Tools).

Komponentu ADUC však můžeme sami upravovat a rozšiřovat. Jednou možností je úprava knihovny, příkladem je ***Additional Account Info*** přímo od MS (součást Windows 2003 Server Resource Kit, musí se zaregistrovat knihovna - `regsvr32 acctinfo.dll`), ale to není tak jednoduché. Druhou možností je využití *Visual Basic skriptu*, který spustíme z ***kontextového menu ADUC***.

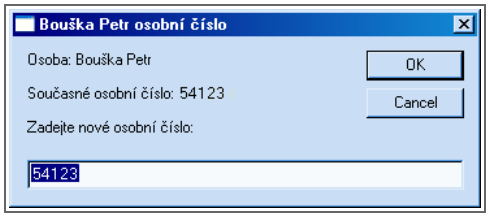
Následující příklad ukazuje, jak vytvořit skript pro zobrazení a editaci atributu `employeeID` (zaměstnanecké číslo). A jak rozšířit kontextové menu objektu `User`, aby se mohl spouštět tento skript.

Vytvoření skriptu pro atribut employeeID

Nejprve musíme vytvořit jednoduchý VB skript, který bude dělat veškerou práci se čtením, zobrazováním a zapisováním hodnot z AD. Toto je pouze jednodušší příklad, který by se mohl rozšířit a doplnit o nějaké kontroly.

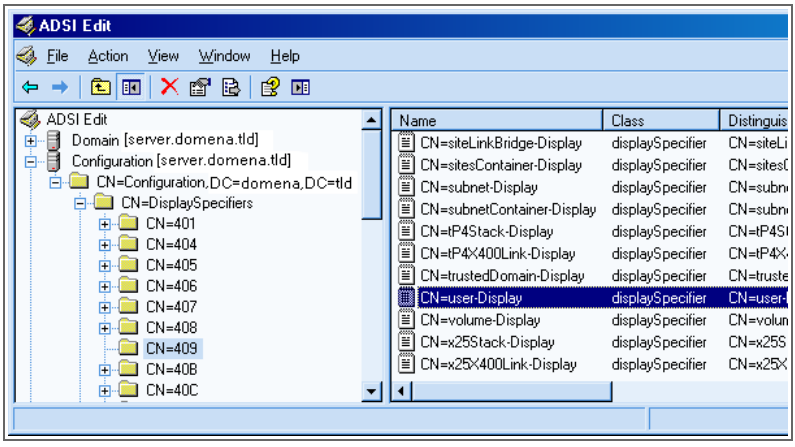
```
Set wshArgs = Wscript.Arguments
Set objAD = GetObject(wshArgs(0))
eID = objAD.employeeID
if eID = "" then eID = "prázdné"
res = InputBox("Osoba: " & objAD.cn & vbCrLf & vbCrLf & "Současné osobní číslo: " & eID & vbCrLf & vbCrLf & "Zadejte nové oso
if res <> "" then
  Err.Clear
  objAD.EmployeeID = res
  objAD.setinfo
  if Err Then MsgBox "Osobní číslo se nepodařilo uložit.", vbCritical, "Chyba !!!"
end if
```

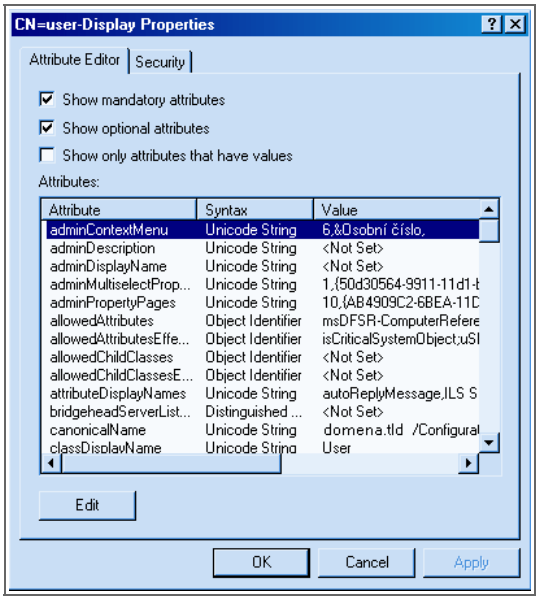
Tento skript uložíme například jako employeeID.vbs a umístíme jej na místo, kde bude přístupný všem, kteří jej budou potřebovat. Úprava kontextového menu se aplikuje na doménu, takže odkudkoliv někdo přistoupí pomocí ADUC na tuto hodnotu, musí mít přístup ke skriptu. Vhodným místem pro uložení skriptu je adresář `NETLOGON` na doménovém kontroleru.

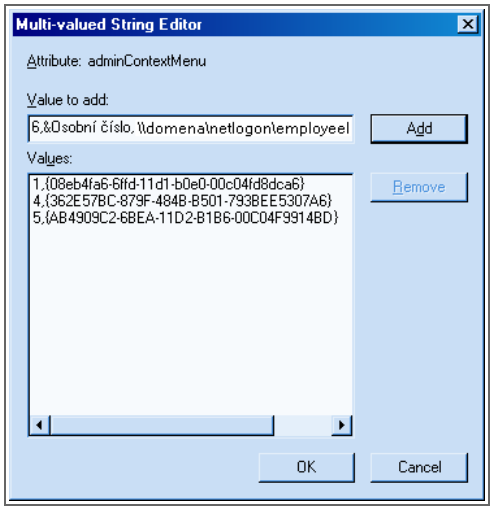


Přidání položky do kontextového menu

1. spustíme ADSIEdit.msc s potřebnými právy (Domain Admin či Enterprise Admin podle struktury)
2. otevřeme ***Configuration***, dále naši ***doménu*** a ***CN=DisplaySpecifiers***
3. podle jazyka vybereme další složku, ***CN=409*** pro angličtinu, ***CN=405*** pro češtinu
4. zde nalezneme položku ***CN=user-Display*** a dvojitým kliknutím otevřeme
   (takže celé DN záznamu je `CN=user-Display,CN=409,CN=DisplaySpecifiers,CN=Configuration,DC=domena,DC=tld`)
5. první atribut v seznamu by měl být ***adminContextMenu***, otevřeme jej dvojitým kliknutím
6. nyní vytvoříme položku do kontextového menu uživatele, první hodnotu zadáme některé volné číslo, druhá je název položky v kontextovém menu a poslední cesta ke skriptu, který se spustí
   Př: `6,&Osobní číslo,\\domain-server\netlogon\employeeID.vbs`







Zajímavé odkazy

- Step-by-Step Guide to Using Active Directory Schema and Display Specifiers - popis problematiky od MS
- Scripts and Files of the Inside Active Directory - řady VB skriptů pro inspiraci
- Scripts to manage Active Directory Users - další VB skripty
- VBScript Fundamentals for Windows Scripting - ADSI - popis psaní VB skriptů pro ADSI

Active Directory komponenty - domain, tree, forest, site

Pátek, 08.02.2008 19:01 | Samuraj - Petr Bouška | Microsoft admin

Mým původním plánem bylo napsat článek, který by byl určen začátečníkům v oblasti správy Microsoftího firemního prostředí a jednoduchou formou popsal základní struktury Active Directory (les, strom, doména a site), jejich vzájemné vztahy a použití. Ale když jsem článek psal, tak jsem doplňoval více a více informací, takže ač myslím, že je stále vhodný pro začátečníky, tak i zkušenější admin může nalézt vhodné informace. Stále je však velké množství podrobností, kterým se v článku nevěnuji, jako replikace či vztahy důvěry. Doplnil jsem ale popis globálního katalogu, operations masters rolí a něco o nasazení.

> *Část věcí, které řeší tento článek, jsem popsal nově v **Kerberos část 1 - Active Directory komponenty** a **Kerberos část 2 - AD uživatelské účty a Service Principal Name**.*

Active Directory

Úplně na začátku se pokusím o stručný (a neúplný) popis *Active Directory*. Termín **Active Directory** (dále pouze AD) je velice rozsáhlý a ukrývá se pod ním celé řešení správy počítačové sítě (počítače s OS Microsoft Windows) ve firemním prostředí podle Microsoftu. ***Active Directory***, ve své podstatě, je distribuovaná ***adresářová služba*** od MS a je součástí Windows Server 2000/2003.

> ***Pozn.:** Více o adresářových službách jsem psal v článku **Adresářové služby a LDAP**.*

Adresář obsahuje uložené soubory informací o ***objektech*** a jejich vzájemných vztazích. V počítačové síti se zde nachází velká řada objektů, jako servery, stanice, aplikace, databáze, uživatelé. Uživatelé musí být schopni nalézt a použít tyto informace. Správci musí mít možnost řídit, jak jsou tyto objekty používány. Je třeba, aby adresář byl *centralizovaný* (centralized), ale také dostatečně *výkonný/škálovatelný* (scalability).

AD v sobě zahrnuje řadu služeb. Jeho primární role je poskytování centrálních služeb pro autentizaci a autorizaci, tedy ***správa uživatelů*** (přesněji správa účtů, protože to může být i třeba počítač). Ale různé části poskytují mnoho dalších funkcí, například ***Group Policy*** umožňuje spravovat politiky jednotlivých počítačů (co je na nich povoleno) a instalovat hromadně (a vzdáleně) aplikace.

AD je silně provázáno s **DNS**, snad mohu říci, že některé části jsou na DNS založeny (rozhodně bez něj nefungují). AD používá stejnou hierarchickou strukturu jako DNS.

> ***Pozn.:** O DNS i ve vztahu k AD jsem psal v článku **DNS (Domain Name System) zaměřeno na Microsoft**.*

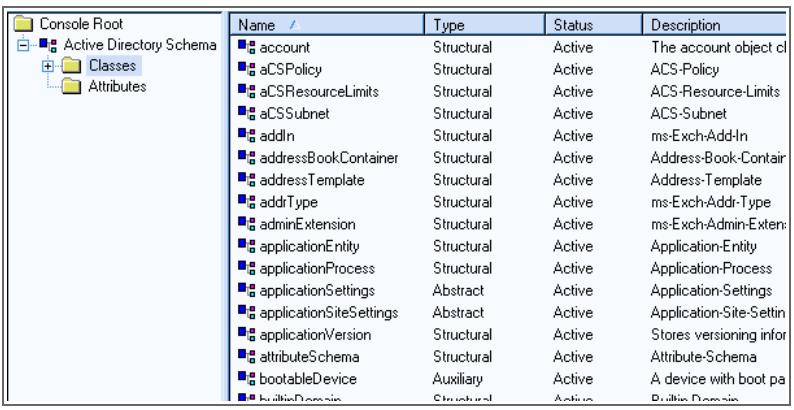
> ***Pozn.:** Pro správu AD slouží balík nástrojů, který můžete nainstalovat z instalačního CD Windows Server pomocí souboru `adminpack.msi`. Pokud chcete rozšířit informace o uživatelích, které poskytuje nástroj **Active Directory Users and Computers** (ADUC), nainstalujte **Windows 2003 Server Resource Kit** a pak spusťte `regsvr32 acctinfo.dll`.*

AD objekty a schéma

Data uložená v **AD** (informace o uživatelích, skupinách, apod.) jsou organizovány jako ***objekty***. Objekt je pojmenovaná skupina ***atributů***, které reprezentují ***síťový prostředek*** (resource). Některé objekty mohou obsahovat jiné objekty, to jsou ***kontejnery*** (container).

Active Directory schéma definuje objekty, které mohou být uloženy v AD. Schéma je seznam definicí, které určují ***druhy objektů*** a ***typy informací***, které mohou uchovávat. Ve schématu jsou dva typy objektů, ***schema classes*** (určuje možné objekty, je souhrnem atributů) a ***schema attributes*** (jednotlivé atributy pro objekt). Dohromady se tyto objekty nazývají ***metadata***.

> ***Pozn.:** Pro správu schématu slouží nástroj **Active Directory Schema**. Ten se standardně nenachází mezi **Administrative Tools**, ale je třeba zaregistrovat knihovnu `regsvr32 schmmgmt.dll` a pak přidat pomocí mmc konzole tento Snap-in (nejlépe následně uložit).*

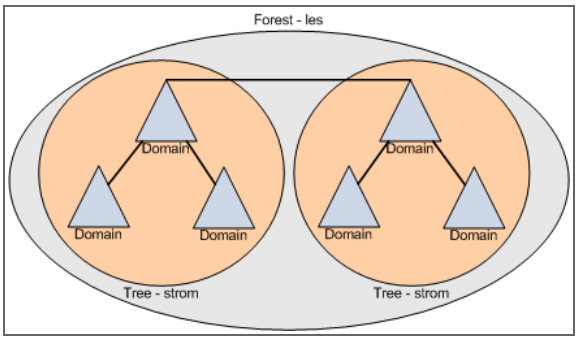


AD komponenty

Komponenty slouží k vytvoření *struktury adresáře* tak, aby odpovídala struktuře organizace a splňovala její potřeby. Některé komponenty reprezentují ***logickou*** a jiné ***fyzickou strukturu***. Tyto komponenty také pomáhají splnit požadavky uvedené výše (jako škálovatelnost).

Logická struktura Active Directory (organizace zdrojů) je tvořena pomocí ***lesa, stromů, domén*** a ***OU***. Na vrcholu struktury je **les - Forest**. Ten může obsahovat jeden nebo více **stromů - Trees**. Strom je tvořen jednou či více **doménami - domains**. Uvnitř domén již máme jednotlivé **organizační jednotky - OU** (Organizational Unit). Uvnitř **OU** se nachází jednotlivé ***objekty*** (počítače, uživatelé, tiskárny, apod.).

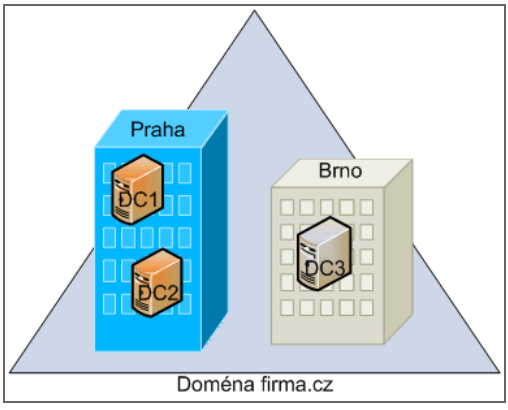
Pozn: Logické seskupování zdrojů slouží k tomu, abychom tyto objekty mohli nalézt podle názvu, místo znalosti jejich fyzického umístění.



Fyzická struktura Active Directory se vytváří pomocí ***doménových řadičů*** a ***site***. Nevím jaké správné české slovo použít pro anglické **site**, ale jedná se v podstatě o ***sítě/podsítě***. ***Site*** je dána určitým rozsahem adres a většinou se ***site*** rovná ***LAN***. Druhou fyzickou komponentou je ***doménový řadič*** - ***Domain Controller*** - **DC**, tedy přímo určitý server, na kterém se nachází část (nebo celá) **AD**.

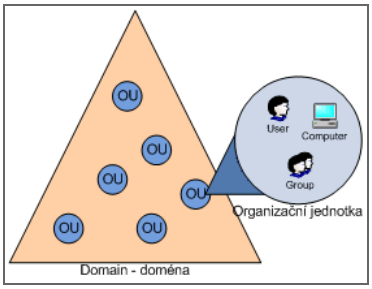
Doména - Domain

Doména (domain) je základním prvkem *logické struktury AD*. V doméně jsou přímo uloženy ***objekty*** (může se jednat o milióny), které do dané domény patří. ***AD*** je tvořena jednou nebo více doménami. Doména není omezena na fyzickou lokaci a může se rozprostírat přes všechny pobočky. Doména je bezpečnostní hranicí, přístup k doménovým objektům je řízen pomocí ACL, které má nastaveno oprávnění (permissions). Bezpečnostní nastavení a oprávnění nemohou přecházet mezi doménami.



Organizační jednotka OU

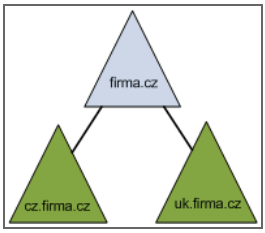
OU (Organizational Unit) je ***kontejner***, který se uvnitř domény používá k seskupování/organizování objektů do logických administračních skupin. **OU** je nejmenší jednotka, na kterou můžeme delegovat *administrační oprávnění*. OU můžeme zanořovat do sebe a vytvářet libovolnou ***hierarchickou strukturu***. Hierarchie OU je lokální uvnitř domény a neovlivňuje jiné domény. OU se většinou vytváří tak, že odráží strukturu organizace (tedy třeba podle divizí a oddělení). Podle potřeby můžeme uživatelské a počítačové účty umísťovat do stejných OU či vytvářet oddělenou strukturu.



> ***Pozn.:*** *Pro správu OU slouží nástroj* ***Active Directory Users and Computers***.

Strom - Tree

Strom je seskupení nebo hierarchická organizace jedné nebo více ***domén***. Vytvoří se tak, že k rodičovské doméně (parent domain), která se nazývá ***kořenová doména*** (Root Domain), přidáme podřízenou doménu(child domain). Domény ve stromě sdílí souvislý ***jmenný prostor*** (namespace), schéma a hierarchické spojení doménových jmen. Používá se DNS standard, takže doménové jméno potomka (child domain) vznikne použitím jeho relativního jména doplněné za tečkou jménem jeho rodičovské domény.



Les - Forest

Les je seskupení jednoho nebo více oddělených nezávislých ***stromů***. Všechny domény v lese sdílí stejné schéma, globální katalog a jsou propojeny implicitním dvoucestným vztahem důvěry (trust). Stromy v lese mají vlastní pojmenování - DNS jméno (oddělený jmenný prostor podle domén). Domény v lese pracují nezávisle, ale díky lesu je umožněna vzájemná komunikace přes celou organizaci (autorizace). ***Root domain*** je důležitá pro les, protože je standardně držitelem dvou speciálních rolí.

Site

Site je kombinace jednoho nebo více IP subnetů, které jsou spojeny spolehlivými a rychlými linkami. ***Site*** obsahuje ***doménové řadiče***. Pokud máme více ***lokálních sítí*** (lokalit, poboček) spojených pomocí ***WAN*** sítě, tak se většinou vytváří ***site*** pro každou LAN. Když se podíváme na logickou strukturu AD, tak se zde site nikde nezobrazují. V určení site figurují pouze **DC** (servery) a spoje a používají se k replikaci mezi nimi. ***Site*** mají význam hlavně pro ***replikace***.

> ***Pozn.:*** *Pro správu site slouží nástroj* ***Active Directory Sites and Services.***

Doménový řadič DC

Doménový řadič (Domain Controller, často se používá zkratka DC) je ***počítač*** (server), na kterém běží operační systém Windows Server 2003 (nebo 2000) a obsahuje repliku ***doménového adresáře*** (lokální doménovou databázi). V doméně může být více doménových řadičů a každý obsahuje úplnou repliku adresáře pro danou doménu. Na jednom řadiči může být pouze jedna doména. Doménový řadič také slouží k autentizaci uživatelů.

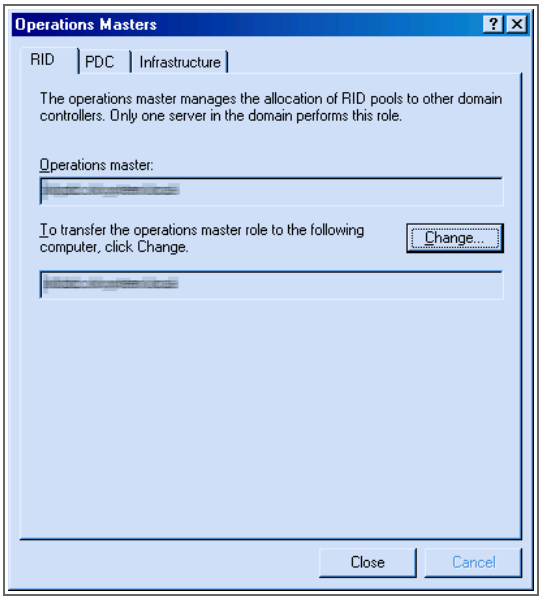
Pokud máme více DC a provedeme nějakou změnu v AD, tak ji provedeme na jednom z DC. Následně se provádí automatická ***replikace*** (periodicky plánovaná nebo v případě důležitých údajů okamžitá), aby ostatní DC měli stejný stav. Běžné replikace jsou typu ***multimaster***, všechny DC jsou si rovny a mají zapisovatelnou kopii adresáře. Přesto některé hodnoty nemohou být řešeny tímto způsobem a používá se model ***singlemaster***, kdy je jeden DC hlavní a na něm se provádí změny. Tyto role se označují jako ***Operations masters roles***. Při změnách stejné hodnoty na více místech se provádí detekce kolize pomocí vlastnosti atributu číslo verze.

Dvě role jsou unikátní pro celý ***les*** a označují se tedy jako ***Forest-wide roles***. Jedná se o ***Schema Master***, udržuje a spravuje celé schéma AD. A ***Domain Naming Master***, řídí přidávání a odebírání domén do lesa.

Další tři role jsou unikátní pro ***doménu***, tedy ***Domain-wide roles***. ***Relative Identifier (RID) Master*** přiděluje řadičům bloky čísel **RID**, která se použijí pro vytvoření **SID** (security identifier - unikátní identifikátor každého objektu v AD). ***Primary Domain Controller (PDC) Emulator*** slouží k emulaci Windows NT 4.0 PDC pro staré klienty. ***Infrastructure Master*** udržuje vazby mezi objekty z různých domén, využívá globální katalog.

Převod role ***Domain Naming Master*** se provádí pomocí **Active Directory Domains and Trusts**. Převod role ***Schema Master*** se provádí pomocí **Active Directory Schema**. Přes pravé tlačítko se připojíme k požadovanému DC a zvolíme položku **Operations Master**. Role ***RID Master***, ***PDC Emulator*** a ***Infrastructure Master*** se převádí pomocí **Active Directory Users and Computers**. Připojíme se k požadované doméně a pravým tlačítkem na ní zvolíme **Operations Master**.

Pro zjištění, který DC je držitelem které role, můžeme použít předchozí grafické nástroje nebo řádkový příkaz `dsquery server -hasfsmo {schema | name | infr | pdc | rid}`.



Globální katalog GC

Globální katalog (Global Catalog - GC) již neurčuje ani logickou ani fyzickou strukturu AD, ale má velmi důležitou roli, takže je zde také popsaný.

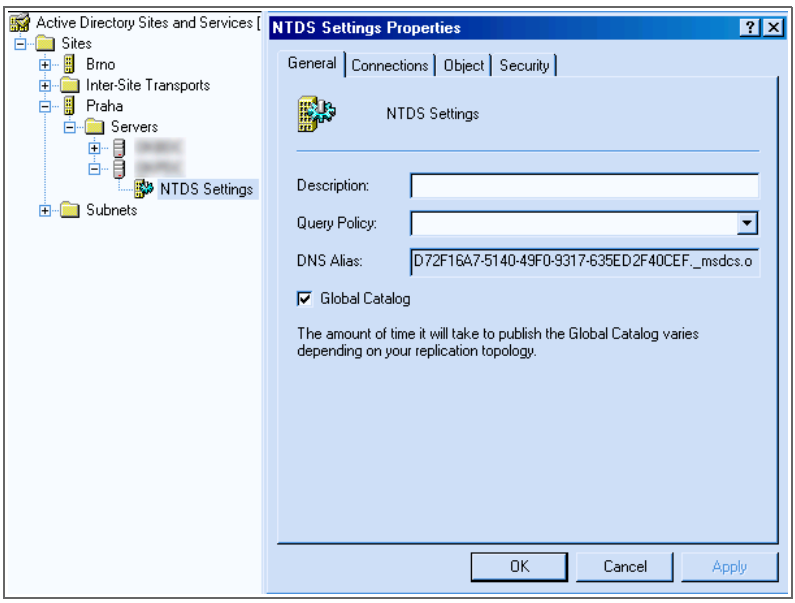
Pokud hledáme nějaký objekt v AD a tento objekt se nachází ve stejné doméně, tak se ptáme některého DC. Pokud však hledáme objekt z jiné domény (ale uvnitř stejného adresáře - stejné AD), tak potřebujeme nějakou službu,

která nám pomůže. V AD je touto službou ***Globální katalog***. To je centrální repository, které obsahuje vybrané informace o objektech z celého stromu či lesa.

DC, který obsahuje kopii globálního katalogu, se nazývá ***Global Catalog Server*** (jinak řečeno GC může být provozován pouze na DC). Globálních katalogů můžeme mít více a mezi nimi se provádí ***multimaster replikace***. GC se často umisťují do různých site, pak ale musíme pamatovat na provoz způsobený replikací (který může být značný).

GC tedy umožňuje nalézt informace z adresáře bez ohledu na to, v které doméně v lese se nachází. Jeho druhou funkcí je, že poskytuje informace o členství v ***univerzálních skupinách*** (universal group membership), které jsou potřeba při přihlašovacím procesu.

> ***Pozn.:*** *Pro nastavení **globálního katalogu** se používá nástroj **Active Directory Sites and Services**, kde se proklikáme přes odpovídající site na hledaný DC a pod ním klikneme pravým tlačítkem na **NTDS Settings** a zvolíme Properties.*



Pokud není k dispozici ***globální katalog*** při přihlašování, tak se uživatel může přihlásit pouze lokálně na počítač. Možností, jak tento problém obejít bez provozování GC, je zapnutí funkce **universal group membership caching** (UGMC) na danou site. V tomto případě si DC ukládá informace lokálně. Při prvním přihlášení uživatele se dotáže globálního katalogu a uloží vrácené hodnoty do keše, kde je uchovává a obnovuje. Při dalším přihlášení se použijí informace z této keše.

> ***Pozn.:*** *Zapnutí funkce **UGMC** se provádí pomocí **Active Directory Sites and Services**, kde klikneme na požadovanou site a pod ním klikneme pravým tlačítkem na **NTDS Site Settings** a zvolíme Properties.*

Nasazení v praxi

V této části se pokusím sepsat pár rad a doporučení o praktickém nasazování AD.

Dělení domén, stromů a lesů

V našich podmínkách často stačí ***jedna doména***. Při instalaci prvního **DC** se vytvoří i ***strom*** a ***les***. Více domén můžeme potřebovat, například když chceme *rozdílnou politiku na hesla* (protože se aplikuje na celou doménu).

Častější důvod na dělení struktury může být kvůli rozdělení ***administračních oprávnění***. Správu uživatelů a počítačů můžeme řešit na úrovni OU, ale pro řadu vyšších úkolů využijeme třeba domény.

> ***Pozn.:*** *Instalace DC se provádí jednoduše pomocí průvodce, který se vyvolá zadáním příkazu `dcpromo`. Jakýkoliv Windows 2003 Server můžeme povýšit na doménový řadič. V průvodci určíme, zda se jedná o novou doménu nebo přidáváme DC do existující. Stejně tak určíme les.*

> ***Pozn.:*** *Pokud instalujeme první DC, tak ten se automaticky stává Root Domain a získá všech pět rolí.*

Více ***stromů*** (tree) využijeme, pokud potřebujeme provozovat různé (oddělené) ***jmenné prostory*** (namespace). To může nastat například při akvizici nějaké jiné firmy, kdy potřebujeme sdílet oprávnění, ale chceme zachovat vlastní názvy.

Více ***lesů*** (forest) potřebujeme pouze ve speciálních případech. Je to tehdy, kdy potřebujeme absolutní oddělení z pohledu oprávnění a autonomie (izolace). Například máme provozní prostředí a testovací, které spolu nesmí mít žádný vztah. Protože *Enterprise admin* má přístup všude v rámci lesa, tak z důvodu úplného oddělení oprávnění musíme vytvořit více lesů. Jinak řečeno, více lesů použijeme, pokud potřebujeme dosáhnout izolace či autonomnosti jednotky uvnitř společnosti nebo pokud spojujeme vzdálené společnosti. Pokud potřebujeme oddělené schéma a globální katalog či ručně řídit vztahy důvěry (trust).

Dělení site

Dělení na ***sity*** je naprosto běžné a používané. Pokud máme více než jednu lokalitu/pobočku, tak je většinou výhodné, provozovat je jako samostatné site.

Umístění globálního katalogu

V umístění ***globálního katalogu*** jsou dvě zásadní otázky. První se týká umísťování do poboček. Obecně je dobré mít v každé pobočce GC, protože to urychluje přihlašování a nepřenáší se data pokaždé po WAN. Také v případě výpadku WAN linky zajistí GC, aby se uživatelé mohli přihlásit. Na druhou stranu se při replikaci přenáší větší objem dat a řada jich je (pro pobočku) zbytečná. Proto pro malé pobočky s pomalým připojením je vhodnější

použít **universal group membership caching.**

Pokud máme ***les*** pouze s ***jednou doménou***, tak se doporučuje mít všechny DC zároveň jako GC, protože GC obsahuje stejná data jako DC. Další doporučení je, pokud nemáme na všech DC globální katalog, tak by měli být ***Operations masters roles*** umístěny na server bez GC.

Na závěr pouze jeden odkaz na materiály Microsoftu **Domains and Forests Technical Reference**. Toto téma je samozřejmě pospáno na mnoha místech, ale dobrý článek není jednoduché nalézt.

Autor: **Petr Bouška**

Související články:

Active Directory a protokol LDAP

Správa počítačové sítě ala Microsoft, to je Active Directory. Jedná se o velice rozsáhlou skupinu technologií a služeb. Základem jsou adresářové služby, adresáře a komunikační protokol LDAP.

- **Adresářové služby a LDAP** [14.09.2007 14:11]
- **Autentizace v LDAPu (AD)** [28.10.2007 14:32]
- **Jak na LDAP a LDAPS v PHP pod Windows** [02.11.2007 16:52]
- **Autentizace uživatele vůči AD v PHP** [06.11.2007 18:18]
- **Rozšíření Active Directory Users And Computers o editaci employeeID** [03.12.2007 14:25]
- **Active Directory komponenty - domain, tree, forest, site** [08.02.2008 19:01] ..... **právě čtete**
- **Převod domény z Windows Server 2003 na Windows Server 2008 R2** [31.03.2010 08:47]
- **Kerberos protokol a Single sign-on** [15.09.2010 18:33]
- **Kerberos SSO - nastavení Internet Exploreru a Firefoxu** [16.09.2010 15:53]
- **Kerberos SSO v PHP aplikaci s Apachem na Linuxu** [22.09.2010 17:58]
- **Active Directory - fotografie u uživatelů nejen pro Outlook 2010** [20.10.2010 09:55]
- **Auditování AD DS objektů ve Windows Server 2008** [26.03.2011 19:23]
- **Active Directory Recycle Bin** [31.03.2011 15:48]
- **Exchange Server, Outlook a certifikáty v GALu** [27.12.2011 14:48]
- **Migrace replikací SYSVOLu z FRS na DFSR** [13.03.2012 15:47]
- **Windows Server 2012 Active Directory** [01.10.2012 17:48]
- **Microsoft Certification Authority Auditing** [13.01.2014 17:05]
- **Kerberos autentizace a členství ve skupinách** [16.01.2014 10:22]
- **Kerberos část 6 - Kerberos SSO mezi doménami** [23.04.2014 15:03]
- **Kerberos část 7 - Troubleshooting Kerberos SSO** [03.05.2014 11:30]
- **Kerberos část 1 - Active Directory komponenty** [13.06.2014 14:28]
- **Kerberos část 2 - AD uživatelské účty a Service Principal Name** [16.06.2014 08:51]

Převod domény z Windows Server 2003 na Windows Server 2008 R2

Středa, 31.03.2010 08:47 | **Samuraj - Petr Bouška** | ***Microsoft admin***

Tento článek obsahuje stručný popis kroků pro převod MS domény na nový funkční level. Nejedná se o žádný podrobný ani úplný návod, ale pouze o orientační informace, které jsem si sepsal, když jsem převod prováděl. Vycházíme z toho, že máme doménu Windows Server 2003 a pro zjednodušení popisu jeden řadič.

Podle mne nejjednodušší a přitom spolehlivá metoda spočívá v instalaci nového serveru, jeho povýšení na Domain Controller (DC) a převodu rolí a služeb. Tzv. in-place upgrade (upgrade OS serveru) se mi v minulosti několikrát neosvědčil, takže se o něj ani nepokouším. Navíc v některých případech musíme vyměnit HW, protože Windows Server 2008 R2 existuje pouze v 64 bitové verzi.

Jednotlivé kroky

- instalace serveru
- příprava lesa a domény
- povýšení serveru na DC
- přesun operations master rolí
- nastavení NTP
- převod služeb/rolí (podle potřeby)
  - DNS
  - WINS
  - DHCP
  - DFS
  - IAS (RADIUS)
  - Certification Authority
  - TS Licensing
- odstranění starého DC
- povýšení funkčního stupně domény a lesa

Instalace serveru

- Nový server HW
- standardní instalace Windows Server 2008 R2
- nastavíme pevnou IP adresu
- zařadíme server do domény (to není třeba, ale já to tak provádím)

Příprava lesa a domény

Dříve než přidáme první DC s Win2k8R2, tak musíme aktualizovat schéma ***Active Directory*** (AD).

- na současném DC, které je držitelem role ***Schema Master***, se přihlásíme účtem, který je ***Enterprise Admin***, připojíme instalační CD Win2k8R2 a v adresáři `/Support/adprep` spustíme soubor `Adprep32.exe /forestprep` , pokud náš současný DC je 32 bitový nebo `Adprep.exe /forestprep` pokud je 64 bitový
- a potom obdobně pustíme `adprep /domainprep /gpprep` (na držiteli role **infrastructure operations master**)
- pokud bychom chtěli použít ***RODC*** (DC pouze pro čtení), tak ještě `adprep /rodcprep`

Povýšení serveru na doménový řadič (Domain Cotroller)

Povýšení serveru na DC znamená instalaci ***Active Directory Domain Services*** (AD DS).

- přihlásíme se účtem doménového admina
- spustíme `dcpromo.exe` (můžeme použít i přidání role Active Directory Domain Services, ale tradice je tradice ;-))
- já zaškrtávám ***Advanced mode***
- volíme existující les i doménu, nemusíme měnit credentials, protože jsem jako domain admin
- nastavíme *site*, asi většinou ponecháme instalaci ***DNS serveru*** a ***Global Catalog*** (GC)
- volíme, zda se data replikují z nějakého DC nebo ze souboru, umístění databáze AD, logů a SYSVOLU (defaultně *C:\Windows\NTDS* a *C:\Windows\SYSVOL*)
- nastavíme heslo pro ***Directory Services Restore Mode Administrator*** a někde si ho bezpečně poznamenáme
- a dokončíme instalaci restartem

Na konci je dobré **zkontrolovat logy** na novém i starém DC.

Zjištění jaká Operations Master role kde běží

Při instalaci nebo dále kvůli přesunu musíme vědět, na jakém DC nám běží jaká ***Operations Master role***. Pro zjištění je nejjednodušší použít řádkový příkaz (další možnost je grafické rozhranní, postup jako při přesunu rolí).

- `dsquery server` - seznam DC serverů
- `dsquery server -hasfsmo schema` - vypíše, kdo má roli Schema Master
- `dsquery server -hasfsmo name` - vypíše, kdo má roli Domain Naming Master
- `dsquery server -hasfsmo infr` - vypíše, kdo má roli Infrastructure Master
- `dsquery server -hasfsmo pdc` - vypíše, kdo má roli PDC Emulator
- `dsquery server -hasfsmo rid` - vypíše, kdo má roli RID Master
- `dsquery server -isgc` - vypíše DC, které mají GC

Přesun Operations Masters Roles

Role unikátní pro celý les

Změnu musíme provádět pod účtem ***Enterprise Admin***.

Schema Master

Pomocí **Active Directory Schema**, klikneme pravým tlačítkem na root, pomocí *Change Active Directory Domain Controller* se připojíme k DC, kterému chceme předat roli, pomocí **Operations Master** můžeme změnit držitele role.

Domain Naming Master

Pomocí **Active Directory Domains and Trusts**, klikneme pravým tlačítkem na root, pomocí *Change Active Directory Domain Controller* se připojíme k DC, kterému chceme předat roli, pomocí **Operations Master** můžeme změnit držitele role.

Role unikátní pro doménu

Změnu musíme provádět minimálně pod účtem ***Domain Admin***.

Relative Identifier (RID) Master

Pomocí **Active Directory Users and Computers**, klikneme pravým tlačítkem na jméno domény, pomocí *Change Domain Controller* se připojíme k DC, kterému chceme předat roli, pomocí **Operations Master** můžeme změnit držitele role.

Primary Domain Controller (PDC) Emulator

Pomocí **Active Directory Users and Computers**, klikneme pravým tlačítkem na jméno domény, pomocí *Change Domain Controller* se připojíme k DC, kterému chceme předat roli, pomocí **Operations Master** můžeme změnit držitele role.

Infrastructure Master

Pomocí **Active Directory Users and Computers**, klikneme pravým tlačítkem na jméno domény, pomocí *Change Domain Controller* se připojíme k DC, kterému chceme předat roli, pomocí **Operations Master** můžeme změnit držitele role.

Nastavení Global Catalog (GC)

Pokud jsme při instalaci nenastavili DC jako GC, tak můžeme nastavit pomocí **Active Directory Sites and Services**, kde se proklikáme přes odpovídající site na hledaný DC a pod ním klikneme pravým tlačítkem na **NTDS Settings** a zvolíme Properties.

Nastavení NTP (Network Time Protocol) - synchronizace času

Počítače v doméně používají jako NTP server doménový řadič, ke kterému se autentizovali. Doménové řadiče používají držitele role PDC. Když instalujeme nový PDC, tak mu musíme nastavit, aby synchronizoval svůj čas z nějakého jiného zdroje, nejčastěji z časového serveru v internetu.

Služba ***Windows Time*** má nastavení uloženo v registrech, takže můžeme buď přímo upravit hodnoty v `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\W32Time\Parameters`. Nebo k tomu samému využít řádkový příkaz `w32tm`. Hlavní je ***NtpServer***, adrese serveru (třeba tik.cesnet.cz) a ***Type***, `NT5DS` znamená synchronizaci s DC a `NTP` se serverem.

Více informací v [How to configure an authoritative time server in Windows Server].

Přesun DNS serveru

Roli DNS jsme nainstalovali spolu s AD DS. Běžné zóny jsou součástí AD, takže se nám přenesou spolu s ***replikací AD***. Pouze pokud jsme použili nějaké zóny, které nejsou ***Active Directory-Integrated***, tak je musíme řešit.

Na serveru 2008 došlo ke změně v **Conditional Forwarders**. Dříve jsme je nalezli na záložce ve vlastnostech DNS serveru, nyní se nachází jako samostatná větev pod DNS serverem. Po instalaci jsem je musel znovu vytvořit.

Také jsem narazil na problém, kdy pro některé domény ***nevracel server IP adresu*** (například csas.cz). Do logu se zapisovala chyba ***Event 5504***, podle kterého jsem našel MS článek [Windows Server 2008 DNS Servers may fail to resolve queries for some top-level domains]. Uvedené řešení ovšem nefungovalo a jediné, co jsem na internetu našel, bylo **nepoužívat Root Hints**, ale přeposílat veškeré dotazy na jiný DNS server. Není to dobré řešení, ale jiné neznám.

> ***Pozn.:*** *K novému DNS serveru se nedá přihlásit pomocí Admin Packu ze Serveru 2003.*

Převod WINS (Windows Internet Name Service) serveru

WINS server (NetBIOS Name Service) na 2008kách stále je, ale je trochu schovaný. Není zde mezi ***Roles***, ale mezi **Features**. Převod na nový server je opět jednoduchý. Po instalaci nastavíme servery navzájem jako ***Replication Partners*** a provedeme replikaci. Vazbu pak můžeme zrušit a starý server odstranit.

Převod DHCP (Dynamic Host Configuration Protocol) serveru

Převod nastavení i databáze DHCP je možné provést pomocí exportu a importu. Ten se dá provést i pomocí grafické konsole DHCP serveru, ale tehdy se může objevit problém, takže se ***doporučuje použít*** řádkový příkaz `netsh`.

Export z původního DHCP

- otevřeme příkazový řádek
- `netsh`
- `dhcp`
- `server <IP adresa>`
- `export c:\dhcp`

Import na novém DHCP

- Instalace role DHCP - nastavení není důležité
- zkopírovat soubor s exportem
- zastavit DHCP
- smazat db, tzn. soubor `c:\windows\system32\DHCP\DHCP.mdb`
- nastartovat
- spustit `cmd` jako Admin
- `netsh`
- `dhcp`
- `server IP`
- `import c:\dhcp`
- restartovat DHCP

> ***Pozn.:*** *Novinkou DHCP na Serveru 2008 je podpora IPv6.*

Více informací v [How to move a DHCP database from a computer that is running Windows Server 2003 to Windows Server 2008].

Převod DFS (Distributed File System)

Služba **Distributed File System** je součástí role ***File Services***, takže nainstalujeme potřebné části. Potom pomocí ***DFS Management*** přidáme nový **Namespace Server** pro každý používaný ***Namespace***.

Převod IAS (RADIUS)

Na serveru 2008 již není ***Internet Authentication Service***, která zajišťovala služby RADIUS (Remote Authentication Dial In User Service) serveru, ale máme zde **Network Policy and Access Services** (NPS). Když používáme Windows Server 2008 R2, tak můžeme převod konfigurace IAS na NPS provést jednoduše exportem a importem nastavení. NPS na serveru 2008 (bez R2) neobsahoval možnost importu (ale MS udělal dodatečně nástroj pro příkazovou řádku).

Export nastavení z IAS (uloží konfiguraci do textového souboru):

```
netsh aaaa show config > c:\iasconfig.txt
```

Import do NPS můžeme provést přes grafické rozhranní nástroje ***Network Policy Server***, pravým tlačítkem na server a ***Import Configuration***. Nebo přes příkazový řádek (musíme spustit jako administrátor):

```
netsh nps import c:\iasconfig.txt
```

Nějaké informace v [How to import and to export IAS configuration information from one Windows 2000 Server-based computer to another Windows 2000 Server-based computer or to another Windows 2003 Server-based computer] a [NPS Fast Facts for Windows Server 2008 R2].

Převod Terminal Services Licensing server

Na doménovém řadiči můžeme mít instalovaný i ***licenční server pro Terminal Services***, na Serveru 2008 se TS nazývají **Remote Desktop Services**. Jeho vlastní převod je jednoduchý, nainstalujeme na nový server a zadáme licence. Ovšem licence musíme nejprve převést. Buď to můžeme provést přes telefon s Microsoft Clearinghouse centrem nebo jednodušeji přes webovou stránku (kterou jsem naštěstí nalezl) [https://activate.microsoft.com/].

Popis převodu v [Move TS Licensing to a New Server].

Převod Certification Authority

Nejvíce problematický je převod CA a rovnou řeknu, že po řadě neúspěšných pokusů jsem provedl čistou instalaci, rozběhl CA nanovo a teprve tehdy vše dobře fungovalo. Uvedu zde jen pár poznámek.

MS postup je v článku [AD CS Migration: Migrating the Certification Authority]. Možná je dnes tento článek v lepší podobě, protože, když jsem jej použil, tak se jednalo o beta verzi článku a nyní se již jedná o normální článek.

V postupu je uvedena i možnost, že se CA převádí na server s jiným jménem (dříve se vždy uvádělo, že je třeba zachovat jméno serveru), to je možná jeden z problémů. Záloha staré CA proběhla bez problémů, stejně tak jako odebrání.

> ***Pozn.:*** *Při odinstalování CA na serveru zůstane řada údajů. Takže když na serveru, kde již byla CA (například špatně převedená), chceme nainstalovat čistou CA, tak musíme nejdříve provést vyčištění. Stejně tak v AD zůstávají určité údaje o staré CA.*

V MS návodu se uvádí, že by se nejprve měla instalovat pouze samotná ***role service*** **Certification Authority** a pokud chceme nainstalovat i další služby, tak bychom to měli udělat později. V praxi jsem narazil na to, že když jsem chtěl dodatečně doinstalovat ***Certification Authority Web Enrollment***, tak jsem dostal chybu. Její řešení jsem našel jediné (a to přímo u MS), odinstalovat CA a nainstalovat znovu, ale vše dohromady.

Když jsem převedl CA na nový server a vyřešil různé "drobné" problémy, tak vše vypadalo, že je OK a funguje. Jenže když se někdo pokusil využít SmartCard logon (pro který hlavně CA využíváme), tak dostal podivnou chybu o CRL, jejíž řešení jsem nikde nenalezl.

Odstranění starého DC (demote domain controller)

Nejprve je potřeba převést Operations Masters role a Global Catalog, ale to jsme již provedli.

- odinstalujeme komponenty/služby, které jsme převedli - ***Add or Remove Programs*** - ***Add/Remove Windows Components***
- odstraníme AD - funkci doménového řadiče
  - spustíme `dcpromo`
  - zadáme heslo pro lokálního Admina
  - dojde k restartu
- odstraníme server z domény (Computer Name - Change - Member of Workgroup)

Více informací [Demote a domain controller].

Vyčištění DNS

Náš DC byl DNS serverem a při odstranění rolí se neodstranily všechny záznamy z DNS, kde byl mezi Name Servery.

- otevřeme DNS správu na novém serveru - ***Forward Lookup Zones*** - pravým tlačítkem postupně na všech zónách - ***Properties - Name Servers***
- smažeme starý záznam
- a to samé pro ***Reverse Lookup Zones***
- ještě můžeme zkontrolovat záznamy ve Forward Lookup Zones **_msdcs.naše.doména**, zde na několika místech mohl zůstat starý záznam a může působit problémy

Můžeme ručně smazat záznam počítače z AD.

V AD **Site and Services** - ***Sites*** - ***naše site*** - ***Servers*** můžeme smazat odstraněný DC.

Povýšení funkčního stupně (functional level)

Povýšení domény

- pomocí **Active Directory Users and Computers**

- pravým tlačítkem na ***doménu***
- volba ***Raise domain functional level***
- zvolíme ***Windows Server 2008 R2*** a klikneme Raise.

Povýšení lesa

- pomocí **Active Directory Domains and Trust**
- pravým tlačítkem na ***Active Directory Domains and Trust***
- volba ***Raise Forest Functional Level***
- zvolíme ***Windows Server 2008 R2*** a klikneme Raise.

Co nám přinese převod domény na vyšší stupeň

- Understanding Windows Server 2008 Active Directory Domain and Forest Functional Levels
- What's New in AD DS in Windows Server 2008
- What's New in Active Directory Domain Services
- How Active Directory Functional Levels Work
- Server 2008 R2: Active Directory Functional Levels

Autor: Petr Bouška

Související články:

Active Directory a protokol LDAP

Správa počítačové sítě ala Microsoft, to je Active Directory. Jedná se o velice rozsáhlou skupinu technologií a služeb. Základem jsou adresářové služby, adresáře a komunikační protokol LDAP.

- Adresářové služby a LDAP [14.09.2007 14:11]
- Autentizace v LDAPu (AD) [28.10.2007 14:32]
- Jak na LDAP a LDAPS v PHP pod Windows [02.11.2007 16:52]
- Autentizace uživatele vůči AD v PHP [06.11.2007 18:18]
- Rozšíření Active Directory Users And Computers o editaci employeeID [03.12.2007 14:25]
- Active Directory komponenty - domain, tree, forest, site [08.02.2008 19:01]
- Převod domény z Windows Server 2003 na Windows Server 2008 R2 [31.03.2010 08:47] ..... **právě čtete**
- Kerberos protokol a Single sign-on [15.09.2010 18:33]
- Kerberos SSO - nastavení Internet Exploreru a Firefoxu [16.09.2010 15:53]
- Kerberos SSO v PHP aplikaci s Apachem na Linuxu [22.09.2010 17:58]
- Active Directory - fotografie u uživatelů nejen pro Outlook 2010 [20.10.2010 09:55]
- Auditování AD DS objektů ve Windows Server 2008 [26.03.2011 19:23]
- Active Directory Recycle Bin [31.03.2011 15:48]
- Exchange Server, Outlook a certifikáty v GALu [27.12.2011 14:48]
- Migrace replikací SYSVOLu z FRS na DFSR [13.03.2012 15:47]
- Windows Server 2012 Active Directory [01.10.2012 17:48]
- Microsoft Certification Authority Auditing [13.01.2014 17:05]
- Kerberos autentizace a členství ve skupinách [16.01.2014 10:22]
- Kerberos část 6 - Kerberos SSO mezi doménami [23.04.2014 15:03]
- Kerberos část 7 - Troubleshooting Kerberos SSO [03.05.2014 11:30]
- Kerberos část 1 - Active Directory komponenty [13.06.2014 14:28]
- Kerberos část 2 - AD uživatelské účty a Service Principal Name [16.06.2014 08:51]

Kerberos protokol a Single sign-on

Upraveno 06.03.2014 21:16 | vytvořeno 15.09.2010 18:33 | Samuraj - Petr Bouška | *Microsoft admin*

Podíváme se na standardní protokol Kerberos, který je již dlouho používaný ve Windows a v posledních verzích se jedná o primární protokol pro autentizaci. Tento protokol je velmi bezpečný a ze svého principu podporuje Single sign-on. A právě SSO je to, co nás zajímá. Kerberos je obecně využívaný (nejen na Windows), ale v popisu budeme vycházet z MS implementace v doménovém prostředí.

> *Věcem, které popisuji v tomto článku, jsem se věnoval nově a detailněji a připravil rozsáhlý seriál Kerberos protokol se zaměřením na SSO v AD DS.*

Single sign-on (SSO) je metoda, která nám dovolí použít jedno přihlášení do více aplikací (pokud se ověřují u stejného zdroje, není nutné znovu vyplňovat stejné přihlašovací údaje). Ve Windows to většinou znamená, že se při přihlášení do počítače autentizujeme vůči Active Directory (doméně). Když je následně potřeba autentizace do dalšího systému, který využívá také účty v doméně (nebo používá mapování svých účtů na doménové), využijí se data, která již máme v systému, pro ověření a přihlášení do další aplikace (aniž by bylo třeba něco zadat).

Ve Windows se používá ***Integrated Windows Authentication*** (IWA), která používá protokoly SPNEGO, Kerberos a NTLMSSP. IWA je primárně spojován s webovými prohlížeči a dovolí nám využití SSO do webové aplikace (je třeba podpora v prohlížeči a na webovém serveru). ***SPNEGO*** (Simple and Protected GSSAPI Negotiation Mechanism, RFC 2478) vyjednává, zda se použije *Kerberos* (verze 5, RFC 4120, což je v MS prostředí nástupce NTLM) nebo *NTMLSSP* (NT LAN Manager - NTLMv1 nebo bezpečnější NTLMv2, ale dnes oboje nedoporučované), případně jiný protokol. **Kerberos** je bezpečný protokol, který využívá metodu klíčů (ticketů) a jemu se budeme dále věnovat.

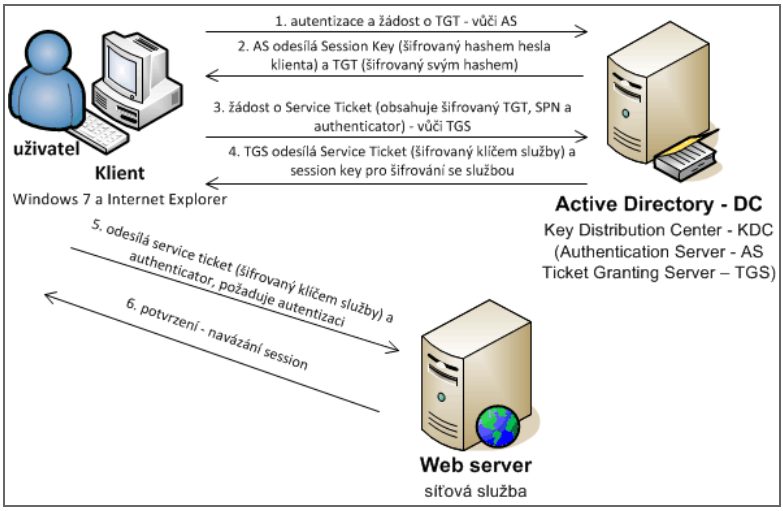
Kerberos a Single sign-on - jak funguje

Protokol Kerberos je síťový autentizační protokol, který používá silnou kryptografii pro bezpečné ověření klienta a serveru přes nezabezpečenou síť. Funguje na principu toho, že se klient neověřuje vůči serveru, kde chce získat nějakou službu, ale vůči prostředníkovi KDC. Centrální autentizační prvek zvyšuje bezpečnost a může poskytovat služby více aplikacím. Na druhou stranu se může jednat o single point of failure (takže musíme zajistit redundanci) a jeho ovládnutí ovlivní více systémů (takže jej musíme dobře zabezpečit na fyzické i síťové úrovni). KDC zná šifrovací klíče všech klientů (klienti, servery, aplikace) a ti se ověřují vůči němu, pro vzájemnou komunikaci mezi klienty jsou od KDC poskytnuty pouze nezbytné údaje.

Stručný popis Kerberos protokolu ve Windows:

- ***autentizace uživatele***
  - při první přihlášení uživatel zadá své přihlašovací údaje (budeme uvažovat jméno a heslo)
  - klient provede jednosměrnou hashovací funkci nad heslem a získat ***secret key***
  - klient odešle žádost na ***Authentication Server*** (AS) což je součást ***Key Distribution Center*** (KDC), ta obsahuje uživatelské údaje v otevřeném textu (neposílá heslo ani secret key)
- ***získání Ticket-Granting Ticket***
  - AS ověří, jestli uživatel existuje a vytvoří si jeho ***secret key*** (heslo má u uživatele v AD)
  - jako odpověď odešle klientovi ***session key*** (klíč pro budoucí šifrování komunikace mezi KDC a klientem), který je zašifrovaný pomocí *secret key klienta*
  - a také odešle unikátní klíč ***Ticket-Granting Ticket*** (TGT), který zašifruje svým klíčem (KDC secret key)
  - tím, že si klient dokáže rozšifrovat session key se potvrdilo, že je to opravdu on (aniž by bylo třeba odesílat heslo)
  - TGT dále slouží k identifikaci uživatele, obsahuje jméno klienta, adresu, dobu platnosti, session key
  - klient nedokáže TGT rozšifrovat (to může pouze KDC)
  - TGT má omezenou životnost (default 10 hodin), ale může probíhat automatická reautentizace, při každém přihlášení se vytváří nový TGT
  - TGT i session key se na klientovi ukládají do ticket cache
- ***získání Service Ticket***
  - když se nyní chceme přihlásit k nějaké službě v síti, tak aplikace na klientovi použije TGT a požádá KDC o ***service ticket***
  - klient posílá žádost o ***service ticket*** na ***Ticket Granting Service*** (TGS), další součást KDC, pomocí dvou zpráv
  - jedna zpráva obsahuje klientův TGT a ID služby - ***Service Principal Name*** (SPN), TGT je již šifrovaný pomocí *KDC secret key* (tak jsme jej získali)
  - druhá zpráva je ***authenticator***, skládá se z uživatelovy identifikace a časové známky, je šifrovaný pomocí *session key (klient a KDC)*
  - dohromady tyto zprávy tvoří credentials uživatele, které jsou platné pouze po dobu logon session, není tedy nutné se znovu ptát uživatele na přihlášení
  - TGS rozšifruje klientův TGT, z něj se dozví *session key* a pomocí něj rozšifruje druhou zprávu, tím ověřil klienta
  - pokud jsou údaje v pořádku, tak TGS odešle ***odpověď***, která obsahuje část klienta a serveru
  - v klientské části je ***session key*** pro komunikaci klienta a serveru (kde je služba), šifrovaný pomocí *session key (klient a KDC)*
  - v serverové části je ***service ticket*** - identifikace uživatele, jeho adresa, doba platnosti a session key (klient a server), šifrováno pomocí *secret key služby*
  - serverovou část nemůže klient dešifrovat a ani pozměnit
- ***ověření u služby (serveru)***
  - service ticket se potom použije k ***autentizaci klienta*** na serveru
  - na server (kam se chceme pomocí SSO přihlásit) se posílá ***service ticket***, stále šifrovaný pomocí *secret key služby*
  - a také ***authenticator***, který se skládá z uživatelovy identifikace a časové známky, je šifrovaný pomocí *session key (klient a server)*
  - server rozšifruje service ticket, z něho získá session key a pomocí něj rozšifruje authenticator
  - tak server získá důvěryhodné údaje o klientovi, když je vše v pořádku, tak odešle zašifrované potvrzení
  - potvrzení obsahuje časovou známku od klienta + 1, je zašifrované pomocí *session key (klient a server)*

- ***navázání session***
  - klient rozšifruje potvrzení a porovná časovou známku, pokud je vše v pořádku, došlo k úspěšné autentizaci
  - ověřil se tedy nejen klient na serveru, ale i server vůči klientovi



> ***Pozn.:*** *Obecně jsme používali dva typy klíčů.* ***Session key*** *pro šifrování komunikace mezi dvěma účastníky, znají jej obě strany.* ***Secret key*** *zná KDC a jedna strana, ověřuje autentičnost.*

Zajímavě napsaný popis Kerberosu naleznete na stránkách MS **Microsoft Kerberos**, bohužel v něm jsou špatně vidět jednotlivé vazby. Hodně informací přináší také PDF dokument **Recommended Practices for Deploying & Using Kerberos in Mixed Environments**.

Nově jsem našel velice hezký popis v článku **Explain like I'm 5: Kerberos**.

Praktický popis SSO k webové aplikaci s použitím Active Directory

Zkusíme si celý proces popsat z praktického a zjednodušeného pohledu. Řešil jsem řadu různých situací a potřeboval jsem přesně vědět, co se kam posílá. Ne vše jsem nalezl v nějaké dokumentaci, takže jsem další věci ověřil praxí a zkoumáním zachycené komunikace. Možná jsou mé závěry závislé na použitém prostředí, šlo o stanici s Windows, jako KDC doménový řadič AD DS, aplikační server Tomcat na Linuxu i Windows (v obou případech využívá keytab soubor).

Na začátku se ***přihlásíme do Windows*** (dojde k ověření vůči AD DS a získáme TGT), nyní pomocí webového prohlížeče otevřeme nějakou stránku/adresu (budeme ji označovat jako službu, třeba `www.firma.local`), která vyžaduje autentizaci a podporuje SSO. Klient teď musí kontaktovat ***KDC*** (ten běží na všech *doménových řadičích*), aby získal ***Service Ticket*** pro danou službu. Určitou roli hraje termín ***Kerberos Realm***, což v MS světě odpovídá ***doméně*** (DNS názvu, třeba `firma.local`), standardně se zapisuje velkými písmeny (tedy třeba `FIRMA.LOCAL`). Dříve jsem myslel (a některá literatura na to poukazovala), že se určuje Realm pro danou službu z jejího FQDN a podle toho se dotazuje KDC. Ale není to tak. Klient vždy ***kontaktuje svoje KDC***, tedy svůj doménový řadič, který (standardně) nalezl pomocí DNS SRV záznamu. Pro non-Windows Kerberos realms můžeme definovat příkazem `ksetup` a uloží se do registrů `HKLM\System\CurrentControlSet\Control\LSA\Kerberos\Domains`.

Když známe KDC, tak se sestaví žádost (`TGS-REQ`), která obsahuje uživatelův ***Realm*** (FIRMA.LOCAL), typ služby (HTTP) a její adresu (www.firma.local) = ***SPN*** (`HTTP/www.firma.local`) a další údaje. KDC vyhledá dle SPN službu (nejprve hledá ve svém realmu, pokud zde SPN nenalezne, tak hledá v existujících důvěryhodných doménách a případně předá klientovi odkaz na jiný řadič) a sestaví ***service ticket*** (který obsahuje údaje o uživateli), který je zašifrovaný pomocí klíče služby (takže s ním klient nic nenadělá) a odešle jako odpověď (`TGS-REP`). ***Service ticket*** posílá klient službě (aplikačnímu serveru), která se z něj dozví kdo je ověřený uživatel a nastaví jej jako přihlášeného. Klíč pro zašifrování service ticketu zná pouze služba (v mém případě využije keytab) a AD, takže není možné podvržení.

> ***Pozn.:*** *Důležitá je také otázka co přesně (jaký atribut) je* ***údaj o uživateli****. Nenašel jsem žádnou oficiální dokumentaci, kde by to bylo popsáno. Na řadě míst se zmiňuje, že jde o* ***User Principal Name*** *(UPN, atribut* `userPrincipalName`*).* ***Kerberos*** *ve své terminologii používá označení* ***Principal Name****, což je možná důvod zmatení s MS atributem UPN. Když zachytíme komunikaci Wiresharkem, tak se položka jmenuje* ***Client Name (Principal)****, ale obsahuje pouze jméno (bez domény).To by vypadalo spíše na atribut* `sAMAccountName`*. Udělal jsem praktické pokusy, kdy jsem měl jiné* `UPN` *a* `sAMAccountName` *a vždy jsem při SSO dostal* `sAMAccountName`*. Zdá se to i logické, protože atribut* `UPN` *je v AD nepovinný (oproti* `sAMAccountName`*) a řada firem nebo jen účtů jej nepoužívá. Takže jsem přesvědčen, že se při SSO používá* `sAMAccountName`*, ale aplikace získává uživatele ve formátu* `sAMAccountName@ClientRealm` *(tedy třeba administrator@FIRMA.LOCAL).*

Z výše uvedeného plyne, že ***server*** (služba) při ověřování ***nemusí komunikovat s AD***, stačí mu, že vlastní svůj klíč pro dešifrování. S AD komunikuje pouze klient, který se chce autentizovat na server. Získá zašifrované údaje, které server rozšifruje a tím se mu potvrdila autentičnost obsažených údajů. Celá bezpečnost tkví v šifrování, vždy se používá klíč, aby zprávu rozšifrovala pouze správná strana, a pro autentizaci se používají dočasné tickety. Na klientovi se tickety kešují a mohou se použít opakovaně.

> ***Pozn.:*** *Náhodou jsem narazil na zajímavou věc. Generoval jsem nový keytab s jinou DNS adresou serveru (šlo ale pořád o stejný server) a použil jsem stejné heslo jako dříve. Aniž bych keytab nahrál na Tomcat, tak jsem zkusil SSO pro novou DNS adresu a vše fungovalo. Očividně se na Tomcatu pouze rozšifroval service ticket pomocí starého secret key (odvozeného od hesla) obsaženého v keytabu a vůbec se neověřovalo SPN.*

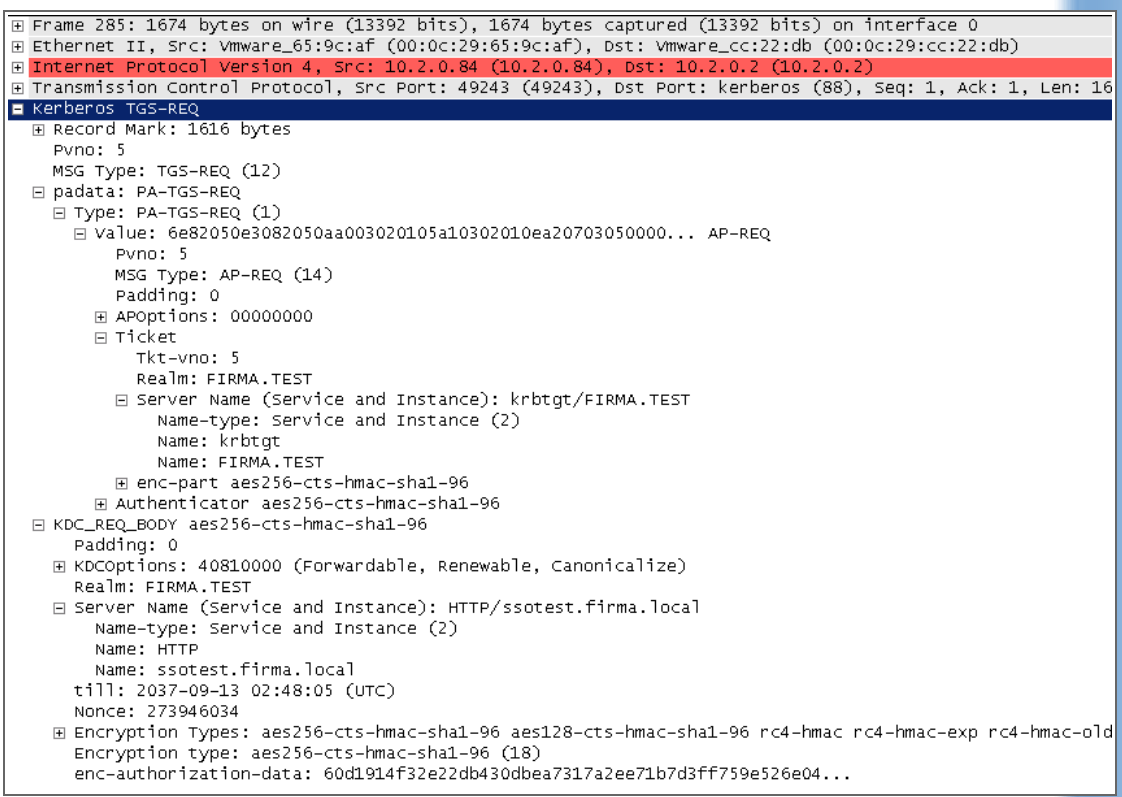
Pro různé testování se může hodit nástroj `klist`, který je součástí Windows a vypíše nám všechny tickety. Pomocí `klist purge` můžeme nakešované tickety smazat. Podobně se může hodit `setspn`, který mimo nastavování SPN na účty, může také vypsat existující SPN a jaký účet je má přiřazeny. V hledání můžeme využít zástupný znak `*`, takže například `setspn -Q HTTP/*`.

> ***Pozn.:*** *V praxi jsem řešil situaci, kdy byl aplikační server umístěn mimo lokální síť a doménu*

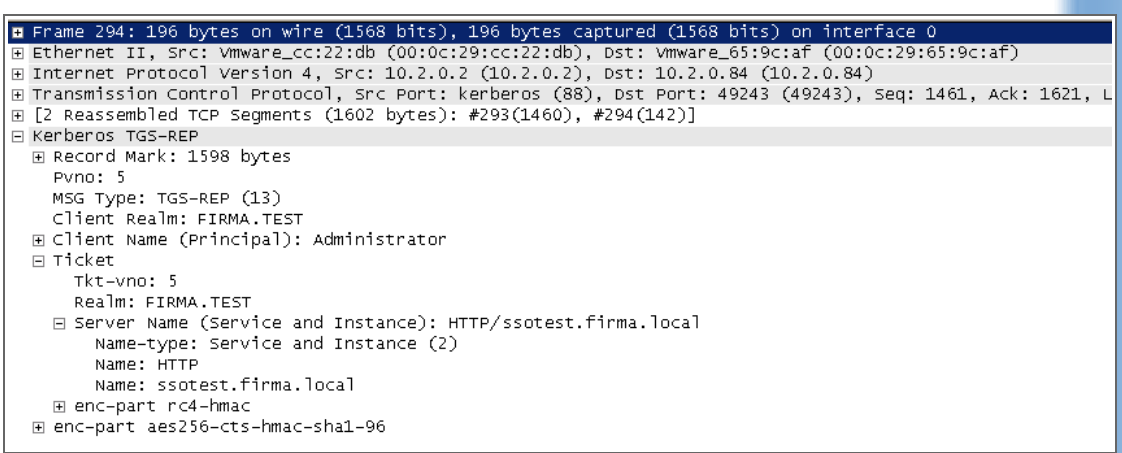
firma.local a požadavek byl přistupovat z vnitřní sítě přes veřejnou adresu server.firma.cz. Potvrdil se teoretický předpoklad, že je vše funkční, když nastavíme SPN HTTP/server.firma.cz na účet v doméně firma.local. Lokální klient odesílá Kerberos požadavek na svůj DC, který podle zadaného SPN nalezne účet a vytvoří Service Ticket, který aplikační server rozšifruje pomocí keytab souboru.

Kerberos pakety žádost a odpověď

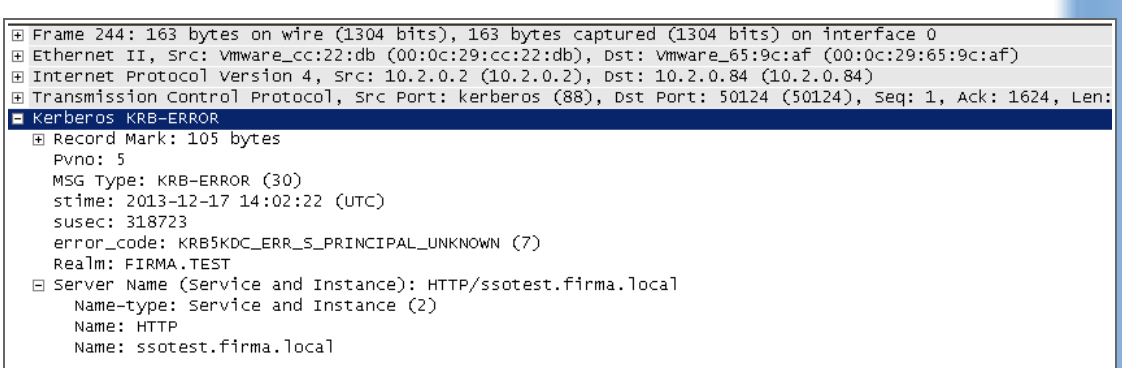
Mnoho informací se dá získat, pokud zachytíme komunikaci na klientovi a podíváme se na pakety, které si vyměňuje s KDC. Nejprve je vidět požadavek od klienta pro SPN `HTTP/ssotest.firma.local`. Uživatel na klientovi je `Administrator` přihlášený v doméně `firma.test`.



Dále vidíme korektní odpověď.



Poslední příklad ukazuje odpověď od DC, pokud nebylo nalezeno požadované SPN.



Související články:

Kerberos a Single Sign-On

Autentizační protokol, který hojně využívá (nejen) Microsoft. Články se věnují jednotnému přihlašování (SSO), v praxi jde hodně o využití Microsoft Active Directory Domain Services.

- **Kerberos protokol a Single sign-on** [15.09.2010 18:33] ..... **právě čtete**
- **Kerberos SSO - nastavení Internet Exploreru a Firefoxu** [16.09.2010 15:53]
- **Kerberos SSO v PHP aplikaci s Apachem na Linuxu** [22.09.2010 17:58]
- **Kerberos autentizace a členství ve skupinách** [16.01.2014 10:22]

Active Directory a protokol LDAP

Správa počítačové sítě ala Microsoft, to je Active Directory. Jedná se o velice rozsáhlou skupinu technologií a služeb. Základem jsou adresářové služby, adresáře a komunikační protokol LDAP.

- **Adresářové služby a LDAP** [14.09.2007 14:11]
- **Autentizace v LDAPu (AD)** [28.10.2007 14:32]
- **Jak na LDAP a LDAPS v PHP pod Windows** [02.11.2007 16:52]
- **Autentizace uživatele vůči AD v PHP** [06.11.2007 18:18]
- **Rozšíření Active Directory Users And Computers o editaci employeeID** [03.12.2007 14:25]
- **Active Directory komponenty - domain, tree, forest, site** [08.02.2008 19:01]
- **Převod domény z Windows Server 2003 na Windows Server 2008 R2** [31.03.2010 08:47]
- **Kerberos protokol a Single sign-on** [15.09.2010 18:33] ..... **právě čtete**
- **Kerberos SSO - nastavení Internet Exploreru a Firefoxu** [16.09.2010 15:53]
- **Kerberos SSO v PHP aplikaci s Apachem na Linuxu** [22.09.2010 17:58]
- **Active Directory - fotografie u uživatelů nejen pro Outlook 2010** [20.10.2010 09:55]
- **Auditování AD DS objektů ve Windows Server 2008** [26.03.2011 19:23]
- **Active Directory Recycle Bin** [31.03.2011 15:48]
- **Exchange Server, Outlook a certifikáty v GALu** [27.12.2011 14:48]
- **Migrace replikací SYSVOLu z FRS na DFSR** [13.03.2012 15:47]
- **Windows Server 2012 Active Directory** [01.10.2012 17:48]
- **Microsoft Certification Authority Auditing** [13.01.2014 17:05]
- **Kerberos autentizace a členství ve skupinách** [16.01.2014 10:22]
- **Kerberos část 6 - Kerberos SSO mezi doménami** [23.04.2014 15:03]
- **Kerberos část 7 - Troubleshooting Kerberos SSO** [03.05.2014 11:30]
- **Kerberos část 1 - Active Directory komponenty** [13.06.2014 14:28]
- **Kerberos část 2 - AD uživatelské účty a Service Principal Name** [16.06.2014 08:51]

Články

Kerberos SSO - nastavení Internet Exploreru a Firefoxu

Čtvrtek, 16.09.2010 15:53 | Samuraj - Petr Bouška | Microsoft admin

V článku se řeší nastavení klienta, aby bylo možno používat Single sign-on vůči aplikaci na serveru. Jako klienty bereme Microsoft Windows, které jsou zařazeny do domény, celá SSO autentizace probíhá vůči Active Directory a uvažujeme pouze protokol Kerberos. Samozřejmě naše serverová aplikace musí SSO podporovat. Hlavně se budeme věnovat webovým aplikacím a tedy konfiguraci webových prohlížečů a hromadné konfiguraci pomocí Group Policy.

> *Věcem, které popisuji v tomto článku, jsem se věnoval nově a detailněji a připravil rozsáhlý seriál **Kerberos protokol se zaměřením na SSO v AD DS**. S tímto článkem nejvíce souvisí díl **Kerberos část 10 - nastavení webových prohlížečů**.*

Pro provedení **Single sign-on** (SSO) budeme uvažovat bezpečný protokol Kerberos, jeho popis naleznete v předchozím článku **Kerberos protokol a Single sign-on**. MS klientské OS od verze Windows 2000 podporují protokol Kerberos nativně a dokonce jej upřednostňují před starším proprietárním protokolem Microsoftu NTLM (který se dnes nepovažuje za bezpečný). Windows také poskytují API pro aplikace, aby mohly využít tento protokol. ***Local Security Authority*** (LSA) zařizuje autentizaci uživatele a volá ***Security Support Provider Interface*** (SSPI), což je rozhraní které komunikuje s autentizačními protokoly (a jeho funkce může využít aplikace), například ***Kerberos SSP***.

Problémy ve Windows

V praxi, při nasazování SSO pro aplikace, jsme na klientovi narazili na dva problémy. Oba vznikají stále větším zabezpečením klientů, o které se MS snaží. Následuje jejich stručný popis a možné řešení.

Poskytnutí TGT a session key

Microsoft zvedl bezpečnost v OS tím, že omezil rozhraní pro získání ***TGT/session key*** páru z Kerberos autentizačního balíčku. Když máme na Windows klientovi aplikaci (napsanou třeba v Jave), která používá Kerberos autentizaci, tak může používat jedno ze dvou rozhraní. Buď ***SSPI***, které zajistí kompletní Kerberos služby nebo ***LsaCallAuthenticationPackage*** z Windows SDK, jehož funkce poskytují mechanismus pro získání ticketu z Kerberos cache. A právě při použití LsaCallAuthenticationPackage potřebujeme mít možnost získat ze systému i session key. Pro takové účely je možno jej povolit pomocí registrů. Jde o hodnotu **AllowTGTSessionKey**, která je defaultně `[dword] 0` (funkce vrací pouze TGT bez session key, takže nelze použít pro SSO) a my ji můžeme změnit na `1`. Ve Windows XP se tento klíč nachází v

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Kerberos\
```

Ve Windows 7, Vista, 2000, Server 2003, Server 2008, je to

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\Kerberos\Parameters\
```

Popis je v článku **Registry Key to Allow Session Keys to Be Sent in Kerberos Ticket-Granting-Ticket**. Změna se projeví až po restartu. Dalším poznatkem je, že aplikace musí běžet pod ***lokálním administrátorem*** (při použití UAC ve Windows 7 se musí manuálně spustit), jinak se jí nepodaří získat TGT. Vše by se zlepšilo, kdyby programátoři použili SSPI, jako je tomu v Internet Exploreru nebo ve Firefoxu.

Popsaný problém může být doprovázen chybou v Jave:

```
GSSException: No valid credentials provided (Mechanism level: No valid credentials provided (Mechanism level: Integrity check
```

Extended Protection for Authentication

Další zvýšení bezpečnosti přinesla technologie ***Extended Protection for Authentication*** (channel binding token - CBT). Pokud je s ní problém, je možno ji vypnout. Chyba se projeví, například když na serveru běží Java (bylo opraveno ve verzi JRE 6 update 21) a my se připojujeme z Internet Exploreru. Při SSO zaloguje chybu `ChannelBinding not provided!`. Popis se nachází v MS článcích **Extended Protection for Authentication Overview** a **Extended Protection for Authentication**. Problém se vyřeší následujícím nastavením registrů (i když hodnotu 2 MS neuvádí jako možnou), na Windows XP je třeba restart.

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\SuppressExtendedProtection=dword:00000002
```

SSO do webové aplikace

Obecný způsob, jak funguje SSO pro webové aplikace. Ve webovém prohlížeči otevřeme nějakou adresu (odešleme HTTP/Post/Get požadavek na web server). Zde je vyžadována autentizace a aplikace má nastaveno použití SSO, které podporuje web server. Takže web server odešle klientovi SPNEGO vyjednávání o autentizaci (HTTP 401 Authenticate/Negotiate). Prohlížeč přijme žádost a nyní záleží na tom, zdá má povoleno IWA a také webovou adresu serveru. Pokud ne, tak odmítne vyjednávání a je již na web serveru nebo spíše aplikaci, jak se zachová. Pokud má povoleno, tak se vyjednává protokol, my uvažujeme pouze Kerberos, takže web server musí podporovat tento. Pokud je vše v pořádku, tak webový prohlížeč využije sytémové knihovny SSPI (Security Support Provider Interface) a získá service ticket od KDC. Tento service ticket odešle jako odpověď na web server (authorization SPNEGO token v HTTP hlavičce). Server jej zpracuje, získá uživatelské ID a ověří jej, a když je vše v pořádku, tak předá aplikaci uživatelovi údaje.

Nastavení Internet Exploreru

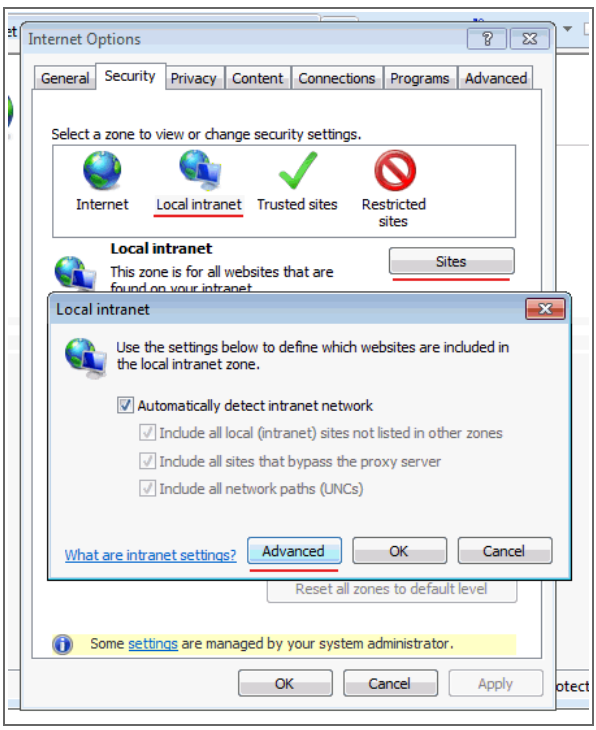
Internet Explorer (IE) jako MS aplikace samozřejmě podporuje ***Integrated Windows Authentication***. Tuto metodu je možno v nastavení zakázat, ale defaultně je povolena. Uplatňuje se pouze na internetové adresy, které patří do

skupiny ***Local Intranet***.

Takže požadavky, aby fungoval SSO v IE jsou:

- povolit ***Integrated Windows Authentication*** - defaultně povoleno (nastavení v menu *Tools - Internet Options - Advanced* - část *Security*)
- nastavit adresu serveru mezi ***Local Intranet*** - defaultně je povolena možnost *Automatically detect intranet network*, takže by se měly automaticky detekovat lokální adresy (*nastavení v menu Tools - Internet Options - Security - Local Intranet - Sites - Advanced*), ale bezpečnější je nastavit přímo adresu

Když zadáváme adresy mezi ***Local Intranet***, tak můžeme zadat přímo adresu serveru, třeba `www.domena.com` (pak se uplatňuje na *http* i *https* protokol) nebo můžeme specifikovat i protokol `https://www.domena.com` nebo můžeme zahrnutou celou doménu (a všechny podřízené adresy) pomocí `*.domena.com`. Místo DNS jmen můžeme použít i IP adresy (volitelné oktety nahradit hvězdičkou), ale v praxi jsem měl s tímto nastavením problémy.



Konfigurace pomocí Group Policy

Nastavení věcí okolo ***Local Intranet*** v IE můžeme provést pomocí ***Group Policy*** (GP) a aplikovat buď na uživatele, nebo na počítač. Doporučované je provést nastavení pro uživatele. Hodnoty se v GP nachází:

```
User Configuration - Policies - Administrative Templates - Windows Components -  Internet Explorer - Internet Control Panel -
```

Otevřeme položku **Site To Zone Assignment List**, povolíme tuto politiku a přidáme adresy, které chceme zařadit do zóny. Pro každou adresu je jeden řádek, do ***value name*** zadáme adresu (třeba **.domena.local*) a do ***value*** dáme číslo zóny, do které chceme zařadit adresu (1 - intranet, 2 - trusted, 3 - internet, 4 - restricted). Vedlejším efektem, který může být vítaný nebo naopak ne, je, že po nastavení této politiky se uživatelům zakáže přidávat (upravovat) adresy v zónách. Hodit se může i informace, kde se nastavení zón nachází v registrech (ale ne pokud jej nastavujeme politikou):

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\Domains\
```

Na stejném místě můžeme také zapnout politiky `Intranet Sites: Include all sites that bypass the proxy server`, `Intranet Sites: Include all network paths (UNC)` a `Intranet Sites: Include all local (intranet) sites not listed in other zones`.

Mozilla Firefox

Ve ***Firefoxu*** je použití ***Integrated Windows Authentication*** také možné již dlouhou dobu. Firefox využívá systémové knihovny SSPI (Security Support Provider Interface nebo na Linuxu GSS-API). Funkce je obecně zapnutá, ale je třeba nakonfigurovat adresy, pro které se použije. Pro všechny ostatní Firefox odmítá všechna vyjednávání od web serveru (mohlo by dojít ke zneužití, hlavně při použití NTLM, které odesílá hash hesla). Ruční konfigurace ve Firefoxu se provádí tak, že do adresního řádku zadáme text:

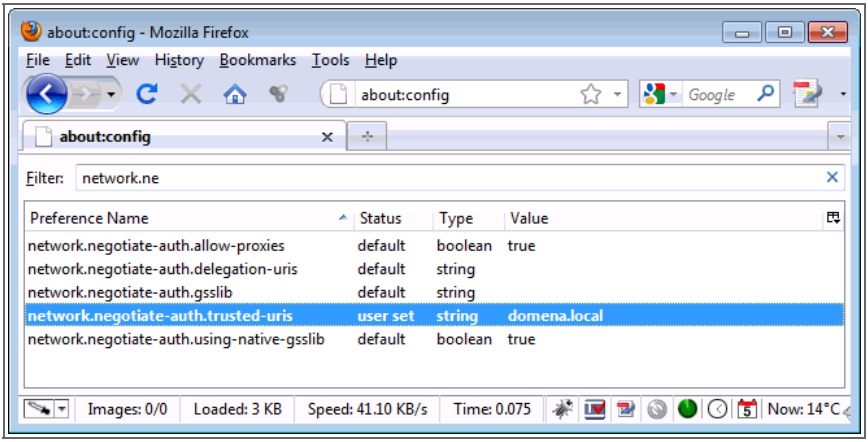
```
about:config
```

Po potvrzení varování, že změny konfigurace mohou být nebezpečné, vidíme jednotlivé konfigurační parametry. Do filtru zadáme proměnnou, kterou chceme měnit. V našem případě jsou k dispozici dvě

Pro Kerberos je to `network.negotiate-auth.trusted-uris`

Pro použití NTLM je `network.automatic-ntlm-auth.trusted-uris`

My používáme Kerberos, takže zadáme první hodnotu a po vyfiltrování ji rozklikneme. Nyní zadáváme textový řetězec. Do hodnoty můžeme zadat přímo adresu serveru, třeba `www.domena.com` (pak se uplatňuje na *http* i *https* protokol) nebo specifikovat i protokol `https://www.domena.com` případně můžeme zahrnutou celou doménu (a všechny podřízené adresy) pomocí `domena.com`. Více hodnot můžeme oddělit čárkou. Změna se projeví okamžitě.



Konfigurace pomocí Group Policy

Konfiguraci Firefoxu můžeme provádět pomocí JavaScriptových konfiguračních souborů. Těch je celá řada, ***uživatelské konfigurace*** se nachází pod uživatelským profilem. Například na Windows 7 je to `c:\Users\{profil}\AppData\Roaming\Mozilla\Firefox\Profiles\{firefox-profil}.default\`. Zde je soubor `prefs.js`, který se vytváří automaticky a neměl by se modifikovat. Můžeme vytvořit souboru `user.js`, kam zadáme nastavení pro uživatele. Jednodušší jsou ***globální konfigurace***, které se nachází v adresáři, kde je nainstalovaný Firefox. Defaultní cesta na 64 bitových Windows 7 je `c:\Program Files (x86)\Mozilla Firefox\defaults\pref\`. Zde můžeme vytvořit soubor `all.js` případně `all-{jméno firmy}.js`. Do souboru zadáme příkaz

```
pref("network.negotiate-auth.trusted-uris",  "seznam adres oddělených čárkou");
```

Když chceme provést hromadnou konfiguraci více stanic, tak můžeme opět použít Group Policy, i když trochu složitěji. Pomocí GP zajistíme zkopírování konfiguračního souboru. Budeme vycházet z toho, že máme k dispozici "novinku" ***Group Policy Preferences***, která nám celou věc usnadní.

Nejprve musíme vytvořit soubor s konfigurací, zmíněný `all.js`. Ten potom potřebujeme umístit do sdíleného úložiště na síti, kam bude mít počítač/uživatel při startu přístup. Jednou možností je adresář ***NETLOGON*** na doménovém řadiči (díky replikaci tedy na všech = na doménovém sharu).

Dále si musíme rozmyslet, jestli budeme politiku aplikovat na uživatele nebo počítač. Soubor budeme nahrávat do ***Program Files***, takže můžeme vytvořit politiku pro počítač, ale jde o to, jak se nám bude lépe aplikovat. Podle toho pak volíme část ***Computer Configuration*** nebo ***User Configuration*** v GP. V příkladu zvolíme uživatelskou část, potřebná konfigurace je v

```
User Configuration - Preferences - Windows Settings - Files
```

Klikneme do plochy pravým tlačítkem a zvolíme ***New - File***. Zvolíme metodu, dostatečná může být ***Create***. Zadáme zdrojovou cestu ke sdílenému souboru a cílové umístění. Budeme počítat pouze s defaultní lokalitou instalace, takže pro 64 bitové Windows 7 je to `C:\Program Files (x86)\Mozilla Firefox\defaults\pref\all.js`. Tím můžeme mít politiku hotovou.

Pokud ale máme v síti i 32 bitové OS a nebudeme řešit rozdělení pomocí aplikace GPO na kontejner, který obsahuje pouze počítače s 32 bitovým OS a jinou politiku aplikovat na 64 bitové OS (což zde nemůžeme, když jsme zvolili uživatelskou část konfigurace). Tak to můžeme vyřešit pomocí ***Item-level targeting***. Máme stále otevřené vlastnosti vytvoření souboru, přepneme se na záložku **Common**. Zde se nachází ***Item-level targeting***, zaškrtneme a klikneme na **Targeting**. Přes ***New Item*** zadáme parametr, který chceme porovnávat. Nabízela by se položka *Operating System*, ale v praxi mi nefungovala správně, takže zvolíme ***File Match*** a porovnávám, jestli existuje adresář `C:\Program Files (x86)\Mozilla Firefox\defaults\pref`. V tom případě se vytvoří soubor zde.

Potom doplníme druhý soubor s cestou `C:\Program Files\Mozilla Firefox\defaults\pref\all.js`, kde změníme pouze u porovnání ***Item Option*** z **Is** na **Is Not**.

Active Directory - fotografie u uživatelů nejen pro Outlook 2010

Středa, 20.10.2010 09:55 | Samuraj - Petr Bouška | Microsoft admin

Již od Windows Server 2000 jsou součástí MS Active Directory (AD) atributy pro uložení obrázku (fotografie) k uživatelskému účtu. Tyto údaje můžeme využít v některých aplikacích (třeba Instant Messaging), které načítají údaje o uživatelích z AD. A hodí se, pokud používáme adresář pro personální data. Nově jsou fotografie z AD podporovány v MS Outlook 2010. Vkládání obrázků do AD není složité, ale existují různé možnosti, které si zde popíšeme.

Fotografie uživatele v AD

V ***Active Directory*** existují dva atributy pro uložení fotografie/obrázku. Jmenují se `thumbnailPhoto` a `jpegPhoto`. První je určený pro náhledy (rozuměno malé obrázky), druhý pro normální obrázky. Ovšem ukládání větších obrázků do AD je třeba zvážit, kvůli nárůstu velikosti databáze. Microsoft dodnes nepřišel s GUI pro editaci těchto atributům, přitom ***Active Directory Users and Computers*** (ADUC) je ideálním kandidátem. Naštěstí máme dnes k dispozici PowerShell.

Před příchodem Windows Server 2008 byla jediná možnost ***editovat přímo atribut*** (třeba přes ADSIEdit) a zadat hexa data obrázku. Samozřejmě není problém napsat jednoduchou ***aplikaci*** (třeba v PHP), která atribut edituje pomocí LDAP příkazů. Jinou možností je ***rozšíření ADUC*** o možnost pohodlné editace v GUI. S příchodem Windows Server 2008, PowerShellu a modulu ActiveDirectory máme možnost využít ***PowerShellových příkazů***.

Outlook 2010 a fotky uživatelů

S příchodem ***Outlooku 2010***, se o fotografiích v AD začalo více hovořit, protože jsou konečně integrovány do pošty. Obecně Outlook 2010 používá řadu personálních údajů a umožňuje napojení na různé sociální a komunikační sítě a systémy (jako IM či IP telefonie).

Na všech místech, kde jsou údaje o ***uživateli*** (a také v hlavičce a konci emailu), se může zobrazit jeho ***fotografie***, pokud ji má Outlook k dispozici. Nejprve se Outlook dívá do lokálně uložených kontaktů, pokud tam fotografie není, tak se může podívat do Active Directory (samozřejmě v doménovém prostředí). Vlastní zprostředkování fotografií neprovádí Exchange Server, ale sám Outlook komunikuje s doménovým řadičem. Fotografie nejsou obsaženy v ***Global Address Listu*** (GAL) a nestanou se ani součástí odesílaných zpráv. Takže se žádným způsobem nedostanou mimo firmu.

Jak jsme si řekli, fotografie se stahují přímo z doménového řadiče (a Outlook je ani nijak dlouhodobě nekešuje, takže se pokaždé stahují znovu). Pokud si zachytíme probíhající komunikaci, tak jsou vidět šifrovaná spojení na DC, kdy se fotografie stahují. Zajímavé je, že když použijeme ***Outlook Anywhere*** (dříve RPC over HTTP), tak fotografie také fungují a stahují se přes to šifrované spojení s Exchange serverem (jak to funguje nevím).

Fotografie uživatele se hledá v atributu `thumbnailPhoto` u daného uživatele. Jak jsme řekli, tak ten je zde již od Windows Server 2000, ale pokud nemáte doménu ***Windows Server 2008***, tak vám fotografie fungovat nebudou. Protože se nejprve ověřuje hodnota atributu `mAPIID`, která musí být `35998`, což je v případě AD Windows Server 2008 (tak jsem to alespoň nalezl na internetu).

Pokud máme ***Exchange Server 2010***, tak máme k dispozici i nový PowerShell příkaz `Import-RecipientDataProperty` pro vkládání fotografií. Ale jak dále uvidíme, nepřináší nic nového oproti standardnímu příkazu z modulu ActiveDirectory. Přitom se mi podle diskuzí na internetu zdá, že tu jednoduchou metodu neznají ani lidé z MS týmu Exchange Serveru.

```
Import-RecipientDataProperty -Identity Ayla -Picture -FileData ([Byte[]]$(Get-Content -Path "M:\Employee Photos\AylaKol.jpg"
```

Atribut `thumbnailPhoto` se standardně nereplikuje do ***Global Catalog*** (GC) a ve všech návodech na internetu se uvádí, že je toto potřeba změnit (pomocí *Active Directory Schema* snap-inu). V praxi jsem ověřil, že to třeba není, ale asi pouze pokud máme jen jednu doménu. Dále se uvádí, že když nemáme Exchange 2010 a používáme v Outlooku ***Cached Exchange mode***, tak nám obrázky fungovat nebudou. Údajně až Exchange 2010 rozšiřuje GAL o odkazy na fotky v AD. Já používám Exchange Server 2007, Outlook 2010, Cached Exchange mode a vše mi funguje.

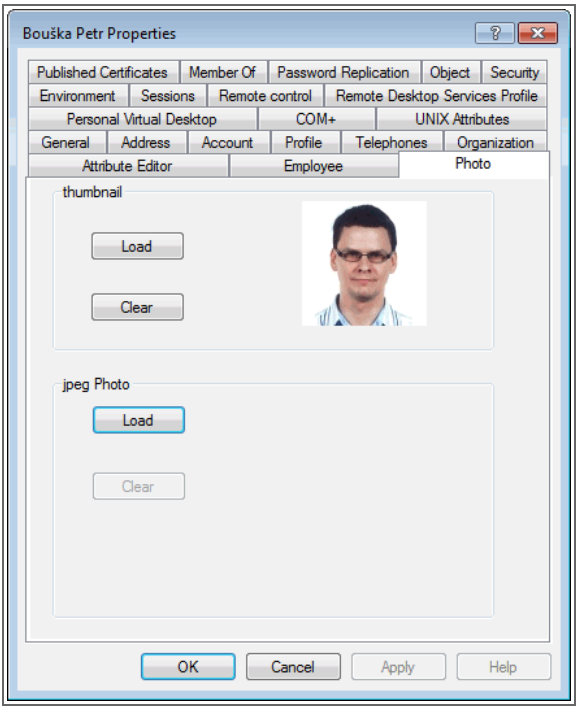
Ještě jedna zajímavá poznámka k tomuto tématu. Outlook 2010 obsahuje také funkci jednosměrné ***synchronizace údajů*** z Exchange GAL do kontaktů. Funguje to pouze pro již existující záznamy v kontaktech a porovnává se SMTP adresa. Takže můžeme vytvořit lokální kontakt pouze se zadanou emailovou adresou a ostatní údaje, včetně fotografie, se nám se synchronizují z Active Directory.

Parametry obrázku pro thumbnailPhoto

Obrázek, který chceme vložit do atributu `thumbnailPhoto`, musí splňovat několik podmínek (i když se často hovoří o tom, že by měl splňovat). ***Formát obrázku*** musí být **JPG** (někde se uvádí i GIF, netestoval jsem). Důležitá je jeho ***velikost***, která nesmí překročit **10 kB**. A doporučené ***rozměry*** obrázku jsou **96 x 96px**.

Editace thumbnailPhoto pomocí rozšíření ADUC

Rozšíření ***Active Directory Users and Computers*** o editaci fotografií k uživateli, si můžeme napsat sami (na internetu existuje řada návodů), ale jelikož ***Oli Dewdney*** napsal funkční knihovnu, tak můžeme použít tu. Stáhnout se dá z adresy **www.dewdney.co.uk/adext/adext.zip**, kde je v zipu vlastní knihovna a návod, jak ji zaregistrovat. Knihovnu jsem otestovat pod ***Windows 7 64 bit*** a funguje korektně a pohodlně. Provede za nás i to, že vkládaný obrázek zmenší a ořeže na potřebné rozměry (což vždy nemusí vypadat nejlépe). Editace obrázku se pak nachází na nové záložce ve vlastnostech uživatele. Takže je vše pohodlné, přehledné a funkční. Pouze je problém, když chceme na počátku editovat velké množství uživatelů, na to je vhodnější skript.



Editace thumbnailPhoto pomocí PowerShellu

Nemám nasazený Exchange Server 2010, tak jsem hledal, jestli neexistuje jiná pohodlná cesta vkládání obrázků a hlavně dávkovým způsobem. Našel jsem několik diskuzí a na první pohled složitých postupů, jak použít PowerShell, až jsem narazil na vynikající, jednoduchou metodu, popsanou v článku **How To: Use AD PowerShell to Manage Outlook 2010 User Photos with Previous Versions of Exchange**.

Použijí se jednoduché dva příkazy a vše funguje. Prvním načteme obrázek ze souboru do proměnné a druhým vložíme do patřičného atributu. Nijak se ovšem nekontroluje velikost obrázku apod., takže si je musíme připravit dopředu.

```
$photo = [byte[]](Get-Content c:\bouska.jpg -Encoding byte)
Set-ADUser bouska -Replace @{thumbnailPhoto=$photo}
```

Obdobným způsobem můžeme i obrázek od uživatele uložit do souboru.

```
$user = Get-ADUser bouska -Properties thumbnailphoto
$user.thumbnailphoto | Set-Content c:\bouska.jpg -Encoding byte
```

Editace thumbnailPhoto pomocí PHP

Pouze jako stručné nastínění možností uvedu příklad kódu, který zobrazuje fotografii z atributu `thumbnailPhoto` uživatele. Zapsání atributu je obdobné. Krátký úvod do použití LDAPu pod PHP jsem popsal v článku **Jak na LDAP a LDAPS v PHP pod Windows**.

```
<?php
function showPictures($user) {
  if($connect = @ldap_connect("ldaps://IP_domenoveho_radice")){
    ldap_set_option($ds, LDAP_OPT_PROTOCOL_VERSION, 3);
    ldap_set_option($ds, LDAP_OPT_REFERRALS, 0);
    if($bind = @ldap_bind($connect, "CN=Uzivatel,OU=Users,DC=firma,DC=local", "heslo")){
      $sr=ldap_search($connect, "OU=Users,DC=firma,DC=local", "(sAMAccountName={$user})");
      $entry = ldap_first_entry($connect, $sr);
      $info = ldap_get_values_len($connect, $entry, "thumbnailPhoto");
      echo $info[0];
      @ldap_close($connect);
      return(true);
    }
  }
  @ldap_close($connect);
  return(false);
}
Header("Content-type: image/jpeg");
Header("Content-disposition: inline; filename=jpeg_photo.jpg");
showPictures($_GET["user"]);
?>
```

Auditování AD DS objektů ve Windows Server 2008

Sobota, 26.03.2011 19:23 | Samuraj - Petr Bouška | Microsoft admin

Pokud chceme logovat operace (zaznamenávat události do logu) nad Active Directory Domain Services (AD DS) - prakticky jde o vytváření, měnění či mazání uživatelských a počítačových účtů a skupin, tak využijeme funkci auditování AD DS. Tento audit byl možný již dříve a to obdobným způsobem, ale Windows Server 2008 přináší rozšíření a zpřesnění. Nově můžeme auditovat pouze určitou podkategorii (vylepšené nastavení nabízí Windows Server 2008 R2), u změn se loguje i původní a nová hodnota (ne jen kdo a jaký atribut změnil) a také se změnily ID událostí.

Tento článek vyšel také na Microsoft TechNet Blogu CZ/SK pod názvem Auditování AD DS objektů ve Windows Server 2008 a v Microsoft sekci na Živě Auditování AD DS objektů ve Windows Server 2008.

Situace okolo auditování událostí nad AD DS není jednoduchá. Musíme si dobře rozmyslet, co chceme sledovat. Jestli nám jde o podezřelé operace, failed události nebo dohled nad vytvářením a mazáním účtů. Pomocí ***Directory Service Access*** (DS Access) a detailního nastavení SACL, můžeme sledovat tisíce údajů (a špatným nastavením si pouze zaplníme log). Hlavní zde tedy je dobře pochopit a nastavit filtrování pomocí SACL.

Navíc použití ***DS Access*** není jediná možnost, i když je Microsoftem všude prezentovaná. Máme ještě kategorii ***Account Management***. Tyto události můžeme použít na sledování změn lokálních účtů na stanici, ale stejně tak na doménovém řadiči pro doménové účty. Tato kategorie nám nenabízí tolik možností, ale její nastavení je jednodušší. A někdy doplňuje informace, které získáme z DS Access. Navíc myslím, že řadě lidí může tato kategorie stačit a vůbec nemusí použít pro logování DS Access.

Auditování kategorie DS Access

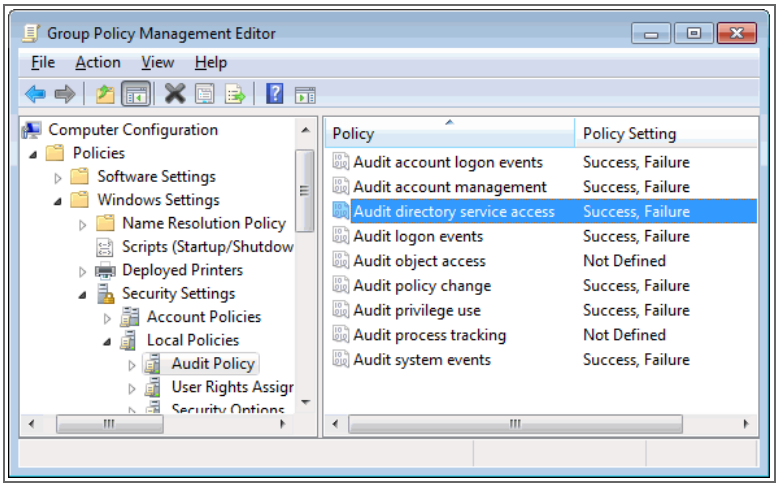
Abychom zapnuli auditování, tak musíme provést několik kroků:

1. **zapneme auditování** - máme dvě možnosti
   - ***globálně*** pomocí auditovací politiky
   - pro jednotlivé ***podkategorie*** - podpora od *Windows Server 2008*, opět dvě možnosti
     - pomocí příkazu **auditpol.exe**
     - pomocí nové **rozšířené auditovací politiky** - podpora od *Windows Server 2008 R2*
2. **vybereme detailně**, jaké operace pro jaké objekty chceme sledovat - pomocí **SACL**
3. (volitelně) **nastavíme výjimky** pro atributy - úpravou ***AD schema*** - provede se nastavením atributu **searchFlags** na hodnotu 256 u objektu, který nechceme logovat, této oblasti se zde nebudeme věnovat

Globální zapnutí auditování

Na DC aplikujeme politiku, která globálně zapíná auditování. Toto nastavení můžeme provést například v ***Default Domain Controllers Policy***.

- spustíme **Group Policy Management**
- otevřeme politiku, třeba **Default Domain Controllers Policy**, v **Group Policy Management Editor**
- proklikáme se do **Computer Configuration/Policies/Windows Settings/Security Settings/Local Policies/Audit Policy**
- otevřeme položku **Audit directory service access**
- zaškrtneme **Define these policy settings** a zvolíme, jestli chceme sledovat úspěšné a neúspěšné událost - **Success** a **Failed**



Zapnutí auditování pro podkategorie

Od Windows Server 2008 můžeme vybrat subkategorii, kterou chceme auditovat. Není to povinné, pokud nastavíme politiku z předchozího bodu, tak se nám zapnout všechny podkategorie k dané kategorii. Standardně (bez politiky) je na serveru zapnuto pouze auditování ***Directory Service Access*** na ***Success***. Jednotlivé podkategorie jsou:

- Directory Service Access
- Directory Service Changes

- Directory Service Replication
- Detailed Directory Service Replication

Nastavení pomocí auditpol.exe

Pokud nemáme ***Windows Server 2008 R2***, tak pro nastavení jednotlivých podkategorií nemůžeme použít Group Policy, ale musí nám stačit řádkový příkaz **auditpol.exe**. Konfiguraci musíme provádět na jednotlivých doménových řadičích a postupně na všech, kde chceme, aby se uplatnilo.

Můžeme si vypsat jednotlivé podkategorie (všechny nebo z nějaké kategorie)

```
Auditpol /list /subcategory:*
Auditpol /list /subcategory:"DS Access"
```

Můžeme provést zálohu nastavení

```
Auditpol /backup /file:C:\auditpolicy.csv
```

Můžeme zjistit aktuální nastavení (všech hodnot nebo z naší kategorie)

```
Auditpol /get /category:*
Auditpol /get /category:"DS Access"
```

A pak můžeme měnit nastavení (povolovat nebo zakazovat jednotlivé podkategorie)

```
Auditpol /set /subcategory:"directory service changes" /success:enable
Auditpol /set /subcategory:"Detailed Directory Service Replication" /success:disable /failure:disable
```

Většinu příkazů musíme spouštět na serveru a konfiguraci provádět na DC. Pokud se o něco pokusíme na stanici nebo ne pod administrátorským oprávněním (Run as administrator), tak dostaneme chybovou hlášku:

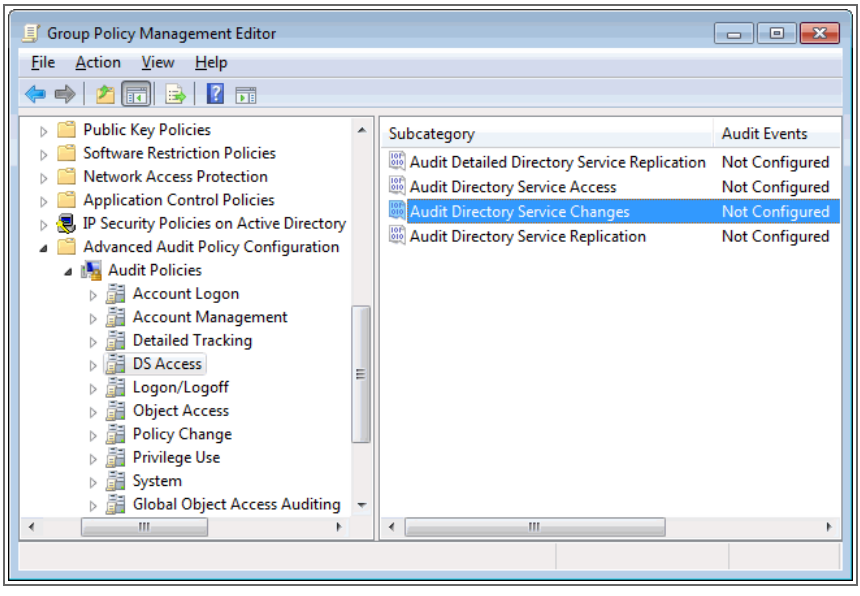
```
Auditpol /get /category:*
Error 0x00000522 occurred:
A required privilege is not held by the client.
```

```
C:\>Auditpol /get /category:"DS Access"
System audit policy
Category/Subcategory                      Setting
DS Access
  Directory Service Changes               Success and Failure
  Directory Service Replication           Success and Failure
  Detailed Directory Service Replication  No Auditing
  Directory Service Access                Success and Failure
```

Nastavení pomocí Group Policy

Windows Server 2008 R2 (a Windows 7) byl rozšířen o možnost konfigurovat audtitování na úrovni podkategorií pomocí Group Policy. Tyto politiky se nachází na trochu jiném místě ve skupině **Advanced Audit Policy Configuration**.

- spustíme **Group Policy Management**
- otevřeme politiku, třeba **Default Domain Controllers Policy**, v **Group Policy Management Editor**
- proklikáme se do **Computer Configuration/Policies/Windows Settings/Security Settings/Advanced Audit Policy Configuration/Audit Policy/DS Access**
- otevřeme položku **Audit directory service access**
- vybereme jednotlivé podkategorie a u nich nastavíme **Success** či **Failed**



Když nastavíme nějakou hodnotu z **Advanced Audit Policy Configuration**, tak se automaticky vypnout všechny hodnoty z **Local Policies/Audit Policy**. Přesto se doporučuje ještě nastavit politiku, která toto zařizuje:

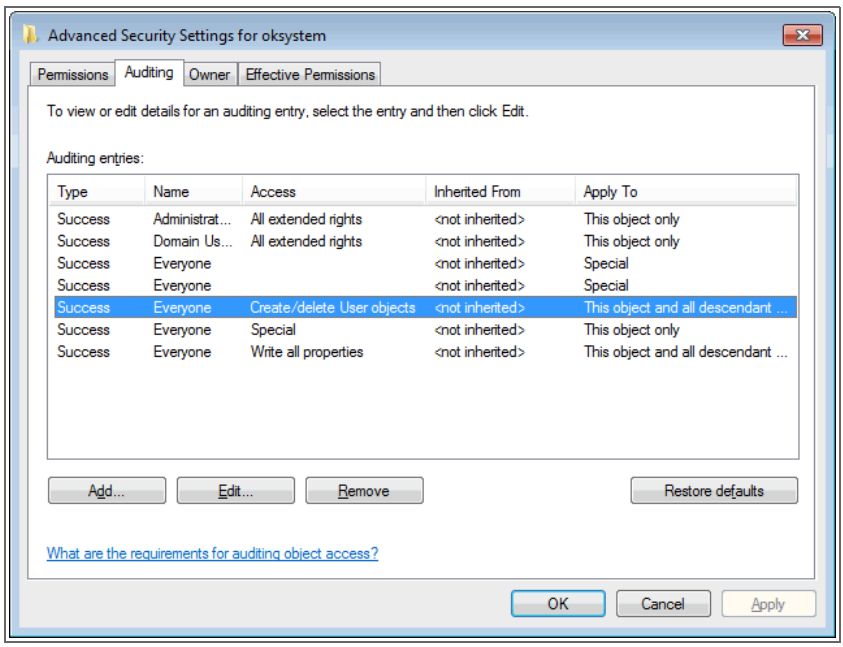
- proklikáme se do **Computer Configuration/Policies/Windows Settings/Security Settings/ Local Policies/Security Options**
- otevřeme položku **Audit: Force audit policy subcategory settings (Windows Vista or later) to override audit policy category settings**
- a zapneme - **Enabled**

Určení, jaké objekty chceme sledovat - SACL

Nastavení oprávnění ***System Access Control List*** - **SACL**, umožňuje logovat přístupy k zabezpečeným objektům,

určuje jaká operace a kým provedená se loguje. Dokud nenastavíme SACL odpovídajícím způsobem, tak se nám nebudou vytvářet žádné záznamy!

- pomocí **ADUC** (Active Directory Users and Computers) vybereme objekt (nejčastěji OU nebo celou doménu)
- klineme pravým tlačítkem a zvolíme **Properties**
- přepneme se na záložku **Security**
- klikneme na tlačítko **Advanced**
- přepneme se na záložku **Auditing**



Zde vidíme již nastavené SACL. Přidání nového:

- klikneme na tlačítko **Add**
- vybereme, pro koho chceme logování provádět, standardně **Authenticated Users** nebo **Everyone**
- na záložce **Object** můžeme nastavovat operace, které chceme sledovat, například **Create User objects** a zatrháváme, jestli chceme sledovat **Failed** nebo **Successful**
- na záložce **Properties** můžeme specifikovat přímo operaci Read/Write
- na obou záložkách můžeme určit, na jaké objekty chceme aplikovat (pak se zobrazí pouze možné nabídky), v **Apply onto**, například **Descendant User objects**

Jakou operaci a na co aplikovanou (Apply onto) musíme dobře zvážit. Například vytvoření nového objektu (uživatele) musíme sledovat na nadřazeném objektu (tedy asi OU). Změnu nějaké hodnoty musíme sledovat na daném objektu jako write property.

Události z kategorie DS Access a jejich ID

Audit generuje události do **Security logu**. Následuje seznam událostí v jednotlivých podkategoriích, oficiálně tento seznam naleznete na webu Microsoftu **DS Access**.

Audit Directory Service Changes

- **Event ID 5136** - A directory service object was modified.
- **Event ID 5137** - A directory service object was created.
- **Event ID 5138** - A directory service object was undeleted.
- **Event ID 5139** - A directory service object was moved.
- **Event ID 5141** - A directory service object was deleted.

Audit Directory Service Access

- **Event ID 4662** - An operation was performed on an object.

Audit Directory Service Replication

- **Event ID 4932** - Synchronization of a replica of an Active Directory naming context has begun.
- **Event ID 4933** - Synchronization of a replica of an Active Directory naming context has ended.

Audit Detailed Directory Service Replication

- **Event ID 928** - An Active Directory replica source naming context was established.
- **Event ID 4929** - An Active Directory replica source naming context was removed.
- **Event ID 4930** - An Active Directory replica source naming context was modified.
- **Event ID 4931** - An Active Directory replica destination naming context was modified.
- **Event ID 4934** - Attributes of an Active Directory object were replicated.
- **Event ID 4935** - Replication failure begins.
- **Event ID 4936** - Replication failure ends.
- **Event ID 4937** - A lingering object was removed from a replica.

Auditování kategorie Account Management

Nastavení auditování kategorie ***Account Management*** je mnohem jednodušší. Spočívá pouze v zapnutí auditování dané kategorie. A stejně jako u ***DS Access*** to můžeme provést dvěma způsoby:

- ***globálně*** pomocí auditovací politiky
- pro jednotlivé ***podkategorie*** - podpora od *Windows Server 2008*, opět dvě možnosti
  - pomocí příkazu **auditpol.exe**
  - pomocí nové **rozšířené auditovací politiky** - podpora od *Windows Server 2008 R2*

Globální zapnutí auditování

Auditování povolíme stejně jako u kategorie DC Access, to znamená nejlépe přes Group Policy. Pouze tentokrát vybereme položku **Audit account management**.

Zapnutí auditování pro podkategorie

Opět můžeme vybrat pouze některé podkategorie pomocí příkazu příkaz **auditpol.exe** nebo pomocí politiky **Advanced Audit Policy Configuration**. Kategorie ***Account Management*** obsahuje podkategorie:

- Audit Application Group Management
- Audit Computer Account Management
- Audit Distribution Group Management
- Audit Other Account Management Events
- Audit Security Group Management
- Audit User Account Management

Události z kategorie Account Management a jejich ID

Audit generuje události opět do **Security logu**. Následuje seznam událostí v jednotlivých podkategoriích, oficiálně tento seznam naleznete na webu Microsoftu **Account Management**.

Audit User Account Management

- **Event ID 4720** - A user account was created
- **Event ID 4722** - A user account was enabled
- **Event ID 4723** - An attempt was made to change an account's password
- **Event ID 4724** - An attempt was made to reset an account's password
- **Event ID 4725** - A user account was disabled
- **Event ID 4726** - A user account was deleted
- **Event ID 4738** - A user account was changed
- **Event ID 4740** - A user account was locked out
- **Event ID 4765** - SID History was added to an account
- **Event ID 4766** - An attempt to add SID History to an account failed
- **Event ID 4767** - A user account was unlocked
- **Event ID 4780** - The ACL was set on accounts which are members of administrators groups
- **Event ID 4781** - The name of an account was changed
- **Event ID 4794** - An attempt was made to set the Directory Services Restore Mode
- **Event ID 5376** - Credential Manager credentials were backed up
- **Event ID 5377** - Credential Manager credentials were restored from a backup

Audit Computer Account Management

- **Event ID 4741** - A computer account was created
- **Event ID 4742** - A computer account was changed
- **Event ID 4743** - A computer account was deleted

Audit Application Group Management

- **Event ID 4783** - A basic application group was created
- **Event ID 4784** - A basic application group was changed
- **Event ID 4785** - A member was added to a basic application group
- **Event ID 4786** - A member was removed from a basic application group
- **Event ID 4787** - A non-member was added to a basic application group
- **Event ID 4788** - A non-member was removed from a basic application group
- **Event ID 4789** - A basic application group was deleted
- **Event ID 4790** - An LDAP query group was created

Audit Distribution Group Management

- **Event ID 4744** - A security-disabled local group was created
- **Event ID 4745** - A security-disabled local group was changed
- **Event ID 4746** - A member was added to a security-disabled local group
- **Event ID 4747** - A member was removed from a security-disabled local group
- **Event ID 4748** - A security-disabled local group was deleted
- **Event ID 4749** - A security-disabled global group was created
- **Event ID 4750** - A security-disabled global group was changed
- **Event ID 4751** - A member was added to a security-disabled global group
- **Event ID 4752** - A member was removed from a security-disabled global group
- **Event ID 4753** - A security-disabled global group was deleted
- **Event ID 4759** - A security-disabled universal group was created
- **Event ID 4760** - A security-disabled universal group was changed
- **Event ID 4761** - A member was added to a security-disabled universal group
- **Event ID 4762** - A member was removed from a security-disabled universal group

Audit Other Account Management Events

- **Event ID 4782** - The password hash for an account was accessed
- **Event ID 4793** - The Password Policy Checking API was called

Audit Security Group Management

- **Event ID 4727** - A security-enabled global group was created
- **Event ID 4728** - A member was added to a security-enabled global group
- **Event ID 4729** - A member was removed from a security-enabled global group
- **Event ID 4730** - A security-enabled global group was deleted
- **Event ID 4731** - A security-enabled local group was created
- **Event ID 4732** - A member was added to a security-enabled local group
- **Event ID 4733** - A member was removed from a security-enabled local group
- **Event ID 4734** - A security-enabled local group was deleted
- **Event ID 4735** - A security-enabled local group was changed
- **Event ID 4737** - A security-enabled global group was changed
- **Event ID 4754** - A security-enabled universal group was created
- **Event ID 4755** - A security-enabled universal group was changed
- **Event ID 4756** - A member was added to a security-enabled universal group
- **Event ID 4757** - A member was removed from a security-enabled universal group
- **Event ID 4758** - A security-enabled universal group was deleted
- **Event ID 4764** - A group's type was changed

Zprávy v Event Logu

Pro nalezení záznamů, které nás zajímají, z velkého množství logů v ***Security logu***, můžeme použít filtrování.

- otevřeme ***Event Viewer***
- doklikáme se na **Security**
- klikneme pravým tlačítkem a zvolíme ***Filter Current log***
- v kolonce ***Event sources*** vybereme **Microsoft Windows security audting.**
- do kolonky ***Task category*** vybereme všechny kategorie, které nás zajímají, třeba:
  - Directory Service Changes
  - Directory Service Access
  - User Account Management
  - Computer Account Management

Obecně se samozřejmě počítá s tím, že si události posíláme na nějaký server, který je dále zpracovává. Může se jednat o standardní ***Syslog server*** (do Windows ale musíme doplnit Syslog klienta, aby se logy odesílaly) nebo třeba Microsoft ***SCOM*** (System Center Operations Manager).

Příklad zpráv

Malá ukázka, co se zaloguje, pokud je zapnuto ***DS Access*** i ***Account Management*** pro události ***Success*** i ***Failed***. **SACL** je nastaveno na celou doménu pro uživatele ***Everyone*** přístup ***Create/Delete User objects***. Zajímavé je, že není zalogována událost 5141.

Vytvoření uživatele

```
12:46:10 4662 An operation was performed on an object. / Object Access Create Child
12:46:10 4720 A user account was created.
12:46:10 4724 An attempt was made to reset an account's password.
12:46:10 4738 A user account was changed.
12:46:10 4722 A user account was enabled.
12:46:19 5137 A directory service object was created.
```

Smazání uživatele

```
13:01:52 4726 A user account was deleted.
```

Odkazy

- AD DS Auditing Step-by-Step Guide
- Auditpol
- Windows Server 2008: Auditing Active Directory

Exchange Server, Outlook a certifikáty v GALu

Úterý, 27.12.2011 14:48 | Samuraj - Petr Bouška | Microsoft admin

Článek se věnuje ukládání šifrovacích certifikátů do Active Directory. Jak k nim potom přistupují klienti jako je Outlook nebo OWA. Jaké mohou nastat problémy, když přejdeme na novou certifikační autoritu, a klientům se z AD nabízí stále staré certifikáty. A také různým PowerShell příkazům, pomocí kterých můžeme s certifikáty v AD pracovat.

Pokud používáme ***Microsoft Exchange server*** a ***Outlook*** a využíváme ***šifrování emailových zpráv***, tak je dobré umístit šifrovací certifikáty jednotlivých uživatelů do ***Active Directory*** (AD). Potom jsou tyto certifikáty (zjednodušeně řečeno) automaticky distribuovány všem uživatelům jako součást ***GALu*** (Global Address List). Takže si uživatelé mohou navzájem posílat šifrované zprávy.

Publikování certifikátu do AD

Certifikáty se v AD ukládají do atributu u objektu uživatele. Umístit certifikát k uživateli (Publish certificate in AD) můžeme několika způsoby:

- **při vydání certifikátu** - pokud využíváme MS CA, tak na šabloně můžeme nastavit, aby se certifikát automaticky publikoval do AD
- **pomocí Outlooku** - uživatelé si mohou vypublikovat nastavený šifrovací certifikát sami v Outlooku (na stejném místě, kde se certifikát nastavuje pro použití), v Outlooku 2010 to je *File - Options - Trust Center - Trust Center Settings* - tlačítko *Publish to GAL*
- **pomocí ADUC** - když si zobrazíme vlastnosti uživatelského účtu pomocí *Active Directory Users and Computers* (ADUC) a máme zapnuté zobrazení *Advanced Features*, tak vidíme záložku *Published Certificates*, zde můžeme přidávat i mazat certifikáty
- **další možnosti** - pomocí nějakých příkazů jako *certutil.exe*, *PowerShell* případně i s rozšířením od firmy *Quest* pro AD

Problém s nabízeným certifikátem

Všechno by se zdálo jednoduché a jasné, hlavně dokud nám vše pěkně funguje. Jak už to ale u Microsoftu bývá, tak je skutečnost o dost složitější. Nedávno jsem narazil na problém, kdy jsme přešli na novou certifikační autoritu. Takže uživatelům bylo třeba vydat nové certifikáty, ty se vypublikovaly do GALu (automaticky pomocí CA). Jenže když se takovému uživateli poslal email, tak se pořád šifroval jeho starým certifikátem. Vymazal jsem tedy v AD staré certifikáty, ale to nepomohlo. Vymazal jsem všechny certifikáty, ale pořad Outlook nabízel starý certifikát. Vyzkoušel jsem všechny různé možností, které jsem popisoval v článku **Šifrovaní emailů v MS Outlook a řešení problémů**. Ale stále nic, až jsem nakonec situaci vyřešil, když jsem se podíval na detaily uživatelského objektu pomocí PowerShell příkazu **Get-ADUser**. Potom se mi podařilo dohledat i dokumentaci, kde jsem se dozvěděl, proč celá situace nastala.

Jak se certifikáty v AD ukládají

Jde o to, že certifikáty se ukládají do atributu `userCertificate`, to je často zmiňovaný atribut. Certifikáty jsou zde uloženy ve formátu ***DER Encoded X.509 v3*** a tyto certifikáty se zobrazují v ADUC na záložce ***Published Certificates***. Jenže certifikáty se také ukládají do dalšího atributu `userSMIMECertificate`, kde jsou pro změnu uloženy ve formátu ***PKCS #7*** a to včetně celého řetězu certifikátů autorit.

Potom, když chceme použít certifikát v nějakém klientovi, tak záleží na klientovi, jaký atribut použije. Například ***MS Outlook*** se podívá do atributu ***userSMIMECertificate*** a pokud certifikát nenalezne, tak potom do atributu ***userCertificate***. A aby to nebylo jednoduché, tak ***Outlook Web Access*** (OWA) to dělá přesně opačně.

Takže když smažeme staré certifikáty u uživatele pomocí ADUC, tak Outlook je bude stále používat (a ještě nastane ta zajímavá situace, že přes OWA se použije ten nový certifikát). Pokud použijeme v Oulooku funkci ***Publish to GAL***, tak se certifikáty umístí do ***obou atributů***. A samozřejmě se doplňují a ne přepisují. Ale pokud využijeme publikování certifikátu do AD při ***vystavení certifikátu***, tak se uloží pouze do atributu ***userCertificate***.

Microsoft sám uvádí ve svém článku, používejte pouze jeden z těchto dvou atributů a určitě nechcete používat *userSMIMECertificate*. Doporučuje také zabránit uživatelům v publikování certifikátů z Outlook (pak se totiž použije tento atribut). Trochu zvláštní tvrzení ...

Mazání certifikátů v AD

Takže v řadě praktických situací potřebujeme smazat certifikáty z atributu **userSMIMECertificate**. Pro ruční operaci můžeme použít ***ADUC*** (z RSAT), kde ve vlastnostech uživatele máme záložku ***Attribute Editor***. Tam jednoduše nalezneme libovolný atribut a můžeme jeho hodnotu smazat. Pokud to potřebujeme udělat pro celou firmu, tak to není moc praktické a jako jasná volba by se stavěl ***PowerShell***.

V ***PowerShellu*** (rozšířeném o MS modul AD) můžeme použít třeba následující příkaz, který nám zobrazí informace o certifikátech. Ten ale bohužel opět pracuje jen s atributem ***userCertificate***.

```
Import-Module ActiveDirectory
(Get-ADUser username -Properties Certificates).Certificates
```

Můžeme se ale podívat přímo na atributy.

```
Get-ADUser username -Properties userCertificate, userSMIMECertificate | FL Name, userCertificate, userSMIMECertificate
```

A obsah atributu ***userSMIMECertificate*** můžeme **smazat** níže uvedeným příkazem. Druhý řádek ukazuje možnost smazání pro všechny uživatele v daném kontejneru.

```
Set-ADUser username -Clear userSMIMECertificate
Get-ADUser -Filter * -SearchBase "OU=Firma,DC=firma,DC=local" | Set-ADUser -Clear userSMIMECertificate
```

Outlook 2010 - zablokování tlačítka Publish to GAL

Zrušení tlačítka ***Publish to GAL*** můžeme provést jednoduše a elegantně pomocí ***Group Policy***. Potřebujeme k tomu, ale šablony pro Office 2010 (nebo odpovídající verzi), toto téma jsem popisoval v článku **MS Outlook a konfigurace pomocí Group Policy**.

Nastavení se nachází v cestě `User Configuration\Administrative Templates\Microsoft Outlook 2010\Security\Cryptography` pod položkou `Do not display 'Publish to GAL' button`.

Odstranění neplatných certifikátů z AD

Když se již bavíme o tématu certifikátů v GALu, tak určitě všichni po nějaké době narazíme na to, že nám Exchange server bude logovat varování, že se u nějakého uživatele nachází certifikát nevalidní nebo s prošlou dobou platnosti a že nebude přidán do GALu. Abychom se těmto varováním vyhnuli, tak by se hodilo automatizovaně promazávat neplatné certifikáty. Pokud to pro nás neudělá certifikační autorita, tak si musíme pomoci nějakým skriptem.

Určitě by se dal v PowerShellu s modulem pro AD napsat nějaký skript, který by využil následující informace, ale nebylo by to nic jednoduchého.

```
(Get-ADUser username -Properties Certificates).Certificates[0].GetExpirationDateString()
(Get-ADUser username -Properties Certificates).Certificates | foreach {New-Object System.Security.Cryptography.X509Certificat
```

Naštěstí je tu firma **Quest** a jejich rozšíření PowerShellu **ActiveRoles Management Shell for Active Directory**. Bohužel ale i jejich příkaz pracuje pouze s atributem ***userCertificate***. Následuje několik jednoduchých příkladů, jak smazat prošlý certifikát uživatele, nevalidní, pokud je v názvu vydavatele slovo Firma nebo všechny certifikáty všech uživatelů.

```
Add-PSSnapin Quest.ActiveRoles.ADManagement


Get-QADUser username | Remove-QADCertificate -Expired
Get-QADUser username | Remove-QADCertificate -Valid:$false
Get-QADUser username | Remove-QADCertificate -IssuerDN '*Firma*'
Get-QADUser | Remove-QADCertificate
```

Migrace replikací SYSVOLu z FRS na DFSR

Úterý, 13.03.2012 15:47 | Samuraj - Petr Bouška | *Microsoft admin*

SYSVOL je důležitý sdílený adresář, který se nachází na všech doménových kontrolerech (DC). A právě proto, aby se nacházel stejný na všech DC, tak se využívají replikace. Od Windows 2000 Server jsou replikace řešeny pomocí File Replication Service (FRS). Při přechodu na doménu Windows Server 2008 se standardně stále používá FRS, ale je možno celkem jednoduše přejít na novější a lepší Distributed File System Replication (DFSR). Jak na to si stručně popíšeme zde.

> ***Pozn.:*** *Pokud instalujeme doménu nově s Windows Server 2008, tak se rovnou využije DFSR.*

Složka SYSVOL

Složka **SYSVOL** (System Volume) obsahuje veřejné soubory, ke kterým mají přístup klienti na síti. Jde hlavně o ***Group Policy*** (skupinové politiky, mohou zde být i šablony) a ***skripty*** a Default User profile (dříve složka NETLOGON). Oproti Active Directory databázi, kde se replikují změny (používá RPC nebo SMTP), se u Sysvolu (pomocí FRS) replikují celé soubory.

Využití DFSR přináší větší výkon, plánování replikací, možnost využít mmc konzole DFS Management a další. Více jsme si o technologii DFS povídali v článku **Popis DFS a migrace na Windows Server 2008 mode**. Obecně je přechod na DFSR silně doporučovaný.

Složka ***Sysvol*** se nachází na každém ***DC*** standardně v cestě `C:\Windows\SYSVOL\sysvol`. Zde je adresář se jménem domény, což je ovšem ve skutečnosti ***Junction Point*** na složku `C:\Windows\SYSVOL\domain`. Dále jsou podsložky `Policies` (kde jsou GPO a GPT) a `Scripts` (ta je z historických důvodů sdílena jako NETLOGON).

Migrace na DFSR

Abychom mohli přejít z ***FRS na DFSR***, tak musíme mít doménu ve funkčním stupni ***Windows Server 2008***. Na všech DC musíme mít nainstalovánu roli ***DFS Replication*** (Roles - File Services - Distributed File System - DFS Replication). Potom můžeme na libovolném DC využít příkaz **dfsrmig.exe**, který je součástí Windows Server 2008 a Windows Server 2008 R2. Na Windows Server 2008 se doporučuje mít nainstalovaný SP2, který vylepšuje replikace (já upřednostňuji rovnou verzi R2).

Než se pustíme do přepínání replikací, tak určitě doporučuji zkontrolovat, že nám aktuální replikace probíhají bez chyb.

Migrace je docela jednoduchá, celou ji provedeme pomocí jednoho příkazu, a probíhá v několika krocích. Jde o to, že se vedle složky **SYSVOL** vytvoří druhá složka **SYSVOL_DFSR**, která se replikuje pomocí DFSR. Jednotlivé stavy migrace jsou:

- **0 - Start** - využívá se pouze FRS
- **1 - Prepared** - vytvoří se kopie složky SYSVOL a ta se začne replikovat pomocí DFSR, ale klienti pořád přistupují na tu původní
- **2 - Redirected** - přesměruje se share na novou složku SYSVOL_DFSR, takže klienti se připojují k nově replikované složce, ale pořád běží obě replikace
- **3 - Eliminated** - odstraní se původní složka, všechny záznamy a vypne se služba FRS, z tohoto stavu se již není možno vrátit zpět

```
C:\>dfsrmig /GetGlobalState

DFSR migration has not yet initialized. To start migration please set global state to desired value.

C:\>dfsrmig /SetGlobalState 1

Current DFSR global state: 'Start'
New DFSR global state: 'Prepared'

Migration will proceed to 'Prepared' state. DFSR service will copy the contents of SYSVOL to SYSVOL_DFSR folder.

If any DC is unable to start migration then try manual polling.
OR Run with option /CreateGlobalObjects.
Migration can start anytime between 15 min to 1 hour.
Succeeded.

C:\>dfsrmig /GetGlobalState

Current DFSR global state: 'Prepared'
Succeeded.
```

Dále postupně, vždy po odzkoušení funkčnosti, přejdeme přes ***stav 2***, kdy se pořád můžeme vracet do stavu 0, až ke konečnému ***stavu 3***. V tomto stavu dostáváme následující informaci, ta je stejná, jako v případě, kdy jsme čistě instalovali 2008 doménu.

```
C:\>dfsrmig /GetGlobalState

Current DFSR global state: 'Eliminated'
Succeeded.
```

Pro podrobnější popis celé problematiky, doporučuji anglické články **SYSVOL Migration Series: Part 1 - Introduction to the SYSVOL migration process** nebo **Introduction to Administering DFS-Replicated SYSVOL**.

Autor: Petr Bouška

Windows Server 2012 Active Directory

Pondělí, 01.10.2012 17:48 | Samuraj - Petr Bouška | Microsoft admin

Nový Windows Server 2012 přináší i novou verzi Active Directory Domain Services. Jednou z novinek je i způsob instalace, kdy se již nepoužívá dcpromo, ale PowerShell nebo Server Manager.

Novinky v Active Directory Domain Services na serveru 2012

Nové Domain Services přináší úpravu či rozšíření stávajících vlastností a také pár novinek. Oficiální popis je v článku What's New in Active Directory Domain Services. Některé body jsou dle mého názoru spíše marketingové, ale některé stojí za povšimnutí. Tento článek se nevěnuje popisu novinek, takže jen letem světem vybrané body.

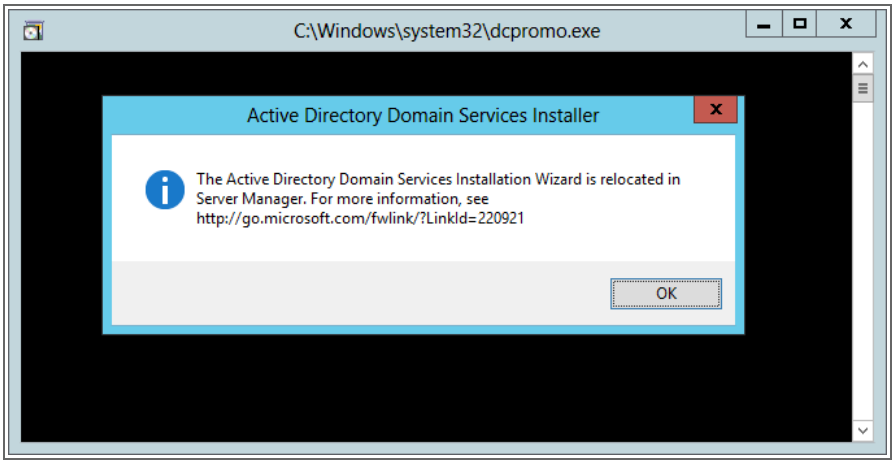
- ***podpora virtualizace*** - virtualizovaný DC nyní podporuje i snapshoty, klonování a podobné funkce
- ***povýšení serveru na DC*** - údajně jednodušší, ale novinkou je, že jej můžeme provést vzdáleně
- ***GUI pro AD Recycle Bin*** - doteď jsme museli použít PowerShell nebo nástroj třetích stran
- ***GUI pro Fine-grained Password Policy*** - zjednodušení konfigurace
- ***nové PowerShell cmdlety*** - 59 nových příkazů hlavně pro Active Directory Sites and Services, Active Directory replication, Dynamic Access Control a Domain Controller cloning a 9 pro nasazení AD DS
- ***Windows PowerShell History Viewer*** - zobrazení historie PowerShell příkazů, které byly použity v Active Directory Administrative Center
- ***Dynamic Access Control*** - nová metoda pro řízení přístupu
- ***Active Directory Activation Services*** - náhrada KMS (Key Management Services) serveru bohužel pouze pro Windows 8
- ***Active Directory Federation Services*** (AD FS) - jsou součástí serveru jako jeho role
- ***Group Managed Service Accounts*** (gMSA) - rozšíření Managed Service Accounts (MSAs)
- ***rozšíření Kerberosu*** - Kerberos Constrained Delegation across domains a Flexible Authentication Secure Tunneling (FAST)

Instalace Active Directory

V tomto článku využijeme čistě nainstalovaný (pouze s minimální konfigurací) Windows Server 2012, jak jsme si připravili v Windows Server 2012 RTM - instalace. A vytvoříme novou testovací doménu `firma.it` v novém lese. Celý proces je jednoduchý a dle mého názoru byl jednoduchý i dříve. Takže jde spíše o referenci jednotlivých kroků.

Oficiální popis nalezneme u Microsoftu v článku Active Directory Domain Services a je zde i popis instalace Active Directory Installing AD DS by using Server Manager.

Podle slov MS je instalace AD DS dnes jednodušší a rychlejší než kdy dříve. Mně připadá, že to příliš rozdílné není. Každopádně již nemůžeme využít `dcpromo.exe`, to nám vrátí chybu.



Instalace probíhá ve dvou krocích

- přidáme roli ***Active Directory Domain Services***
- povýšíme server na DC ***Promote this server to a domain controller***

Oba kroky můžeme provést pomocí

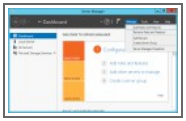
- ***Windows PowerShell***
- ***Server Manager***

a můžeme je provádět i vzdáleně nebo hromadně na více serverech najednou. Zdejší screenshoty jsou z instalace pomocí GUI, tedy využití Server Manageru, přímo na serveru.

Přidání role AD DS

- spustíme ***Server Manager***
- (například) v pravém horním rohu klikneme na ***Manage*** a ***Add roles and features***
- při instalaci volíme roli ***Active Directory Domain Services*** (automaticky se nabídnou Features), můžeme rovnou přidat i některé další, třeba DNS (to budeme potřebovat, ale automaticky jej nainstaluje další průvodce)
- při zobrazení výsledků průvodce jej můžeme zavřít, instalace probíhá na pozadí a informace si můžeme

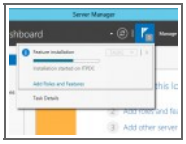
zobrazit kliknutím na vlajku v pravém horním rohu Server Manageru



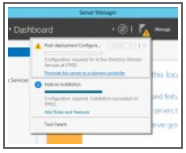


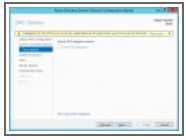
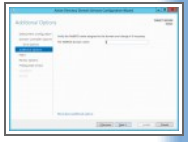


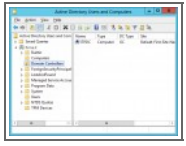
Povýšení serveru na doménový řadič

- po dokončení přidání role se nám v Server Manageru u vlajky zobrazí vykřičník, kliknutím na ikonu se dozvíme, že je třeba provést ***Post-deployment Configuration***
- klikneme na ***Promote this server to a domain controller***
- zde vytváříme nový les, tedy novou kořenovou doménu, kterou nazveme `firma.it`
- když se dokončí instalace, tak se automaticky provede restart serveru







Kerberos část 1 - Active Directory komponenty

Pátek, 13.06.2014 14:28 | Samuraj - Petr Bouška | Microsoft admin

Vítejte u prvního dílu seriálu, který se věnuje protokolu Kerberos se zaměřením na Single Sign-On (SSO) v prostředí Microsoft Active Directory. Dnešní díl se přímo Kerberos protokolu nevěnuje, ale popíšeme si základní termíny Active Directory, které potřebujeme znát, a s kterými Kerberos autentizace souvisí (když ji používáme v doménovém prostředí). Stručně zmíníme AD komponenty, protože struktura souvisí s Kerberos Realm. Dále si popíšeme, jak klient hledá doménový řadič (což je zároveň Kerberos autentizační server).

V dalších dílech si popíšeme obecně vlastnosti a funkce Kerberos protokolu. Podíváme se na to, co vlastně znamená SSO a jaké jsou možnosti. Rozebereme si jednotlivé komponenty a prvky, které Kerberos protokol používá. Popíšeme si proces Kerberos autentizace (tedy SSO) od zjednodušeného, přes podrobnější, až k detailnímu popisu. Podíváme se na fungování SSO autentizace mezi různými doménami. A zmíníme možnosti využití Kerberos SSO pro webové aplikace.

> *Tento seriál vychází také (v lehce upravené podobě) v rámci Microsoft TechNet Blog CZ/SK v Microsoft sekci na Živě. První díl Kerberos část 1 - Microsoft Active Directory.*

Microsoft AD DS

Samozřejmě si tu nepopíšeme vše (a s určitou znalostí AD DS počítáme), nějaké další věci jsem popsal již před lety v článcích Adresářové služby a LDAP, Active Directory komponenty - domain, tree, forest, site a DNS (Domain Name System) zaměřeno na Microsoft. Odkaz na více materiálů se nachází na konci článku.

Active Directory Domain Services

Active Directory (AD) je adresářová služba od Microsoftu. ***Active Directory Domain Controller*** (DC či doménový řadič nebo pouze řadič) je server na kterém běží ***Active Directory Domain Services*** (AD DS, tento název se používá od Windows Server 2008, předtím se používal pouze Active Directory). AD DS poskytuje distribuovanou databázi, která v hierarchické struktuře obsahuje síťové objekty (jako je uživatel, počítač, skupina). Zajišťuje také autentizaci a autorizaci uživatelů a počítačů v síti. Využívá se LDAP protokol, rozšířený Kerberos verze 5 a DNS (Domain Name System).

Pro správu AD DS od Windows Server 2008 se využívá balíček ***Remote Server Administration Tools (RSAT)***, který je součástí serverových OS a na klientské jej můžeme doinstalovat.

Doména (Domain)

Doména (Domain) je logická skupina počítačů, které sdílí společnou AD databázi. Když vytvoříme doménu, tak se automaticky vytvoří strom a les. Nejjednodušší je mít vše pouze jednou. V praxi to ale vždy nelze, takže musím vytvářet složitější struktury. Doména se ve schématech znázorňuje jako trojúhelník, prvky uvnitř jsou členy dané domény.

Jméno Active Directory ***domény*** je běžně plné ***DNS*** (Domain Name System) jméno (příklad `firma.local`). Může obsahovat písmena, číslice, pomlčku a tečku (pouze pro oddělení komponent), maximální délka je 64 znaků. Každá doména má ale také (z důvodu kompatibility) *pre-Windows 2000 jméno*, které se označuje jako ***NetBIOS doménové jméno*** (příklad `firma`). To může být maximálně 15 znaků dlouhé a nesmí obsahovat tečku a řadu dalších znaků.

Zobrazit doménová jména si můžeme například pomocí nástroje ***Active Directory Domains and Trusts***. V okně vidíme seznam domén a jejich ***DNS jména***, pokud na doménu klikneme pravým tlačítkem a zvolíme *Properties*, tak uvidíme ***NetBIOS jméno*** a také stupeň lesa a domény.

Strom (Tree) a hierarchie domén

V rámci ***stromu*** (Tree) existuje hlavní ***kořenová doména*** (Tree Root Domain), která má svůj ***jmenný prostor*** (namespace, třeba firma.local). Pod kořenovou doménou můžeme vytvořit další domény, které se označují ***Child Domain***, ty obsahují jmenný prostor nadřazené domény (Parent Domain) a svoje relativní jméno (tedy třeba test.firma.local). Každá doména má vlastní AD databázi, která se nachází na všech doménových řadičích dané domény. V rámci stromu může existovat pouze jedna kořenová doména, pokud chceme druhou (hlavně kvůli novému namespace), tak musíme vytvořit nový strom. Strom se ve schématech znázorňuje jako ovál či kruh.

Les (Forest), schéma a vztahy důvěry

Jeden nebo více stromů se nachází v ***lese*** (Forest). Les sdílí společné AD schéma (Active Directory Schema). ***Schéma*** definuje AD databázi, co v ní může být uloženo a jakou to má strukturu. Každá doména má vlastní databázi (obsah), ale stejnou strukturu (schéma). V rámci lesa existuje jedna ***kořenová doména*** (Forest Root Domain), jde o prvně vytvořenou doménu v daném lese. Ta obsahuje skupiny *Enterprise Admins* a *Schema Admins*. Les se ve schématech znázorňuje jako obdélník či kruh.

Když chceme zjistit, jaká doména je ***Forest Root***, tak můžeme spustit nástroj ***Active Directory Domains and Trusts***, který také použijeme pro správu ***vztahů důvěry***. Zvolíme ***Action - Change forest*** a zde vidíme aktuální kořenovou doménu.

Všechny domény v rámci lesa mají automaticky vytvořenou ***důvěru*** (Trust). Jde o ***Two-way Transitive Trust***, to znamená, že vztah důvěry je obousměrný a není omezen na dvě přímo propojené domény, ale může se rozšířit dále (pokud má druhá doména důvěru s další doménou, tak této třetí doméně důvěřuje i ta první). Vytváří se důvěra mezi kořenovými doménami stromů v rámci lesa, ta se označuje ***Tree-Root Trust***. A mezi nadřízenými doménami v rámci stromu, to je ***Parent-Child Trust***. Ručně můžeme vytvořit důvěru mezi různými lesy (navazuje se mezi Forest Root Domain), to je ***Forest Trust***, a dále *Shortcut Trust*, *Realm Trust* a *External Trust*.

Pro správu schématu slouží nástroj ***Active Directory Schema***. Ten se standardně nenachází mezi ***Administrative Tools (ani když jsme nainstalovali RSAT, který potřebujeme)***, ale je třeba zaregistrovat knihovnu `regsvr32 schmmgmt.dll`. Potom můžeme použít ***mmc konzoli*** a přidat Snap-in ***Active Directory Schema*** (nejlépe následně uložit).

Global Catalog

V rámci lesa je důležitá služba ***Global Catalog*** (GC). Ta obsahuje index pro všechny objekty v rámci lesa (jde o částečnou kopii adresářových oblastí pouze pro čtení), ke každému má pouze základní informace. V každé doméně musí být alespoň jeden GC server (a ten je zároveň DC). Doménový řadič má údaje pouze o objektech, které patří do jeho domény. Pokud chceme přistoupit k prvku z jiné domény, tak se právě využije *Global Catalog*, který nám poskytne informace, na jakém doménovém řadiči se tento objekt nachází.

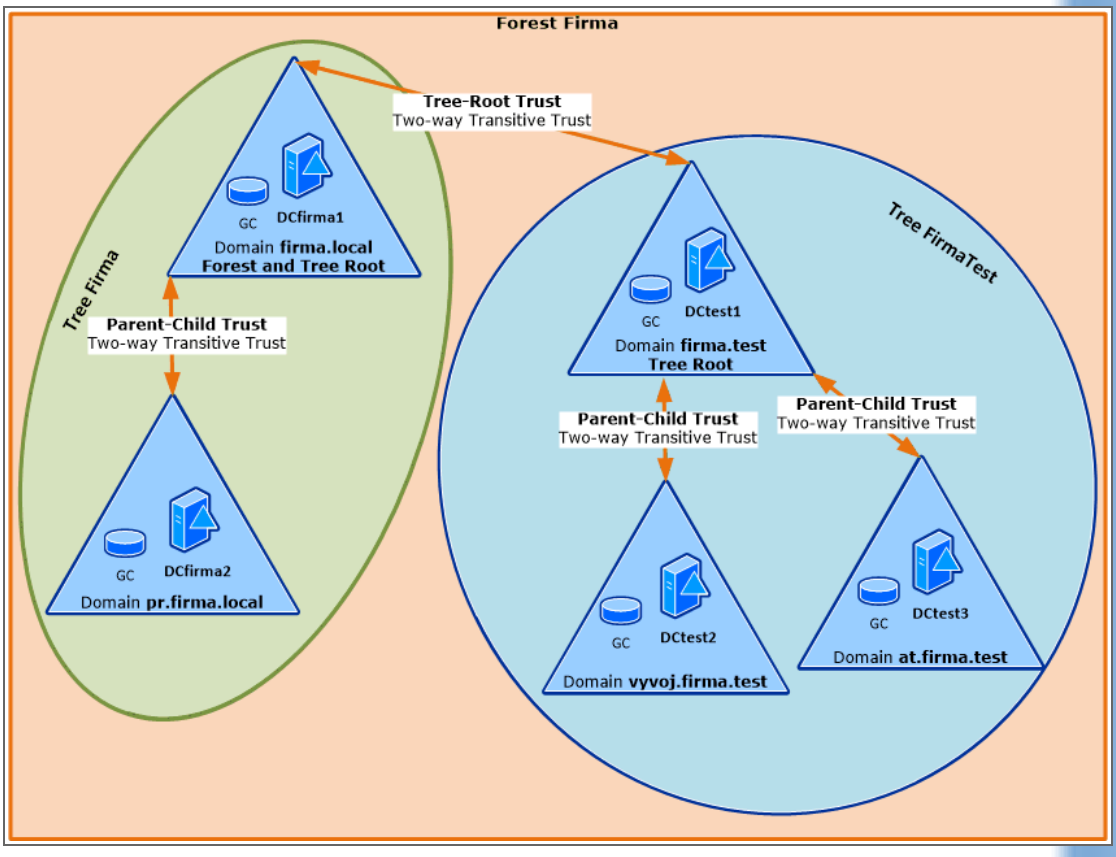
Pro nastavení ***globálního katalogu*** se používá nástroj ***Active Directory Sites and Services***, kde se proklikáme přes odpovídající *Site* na hledaný DC a pod ním klikneme pravým tlačítkem na ***NTDS Settings*** a zvolíme *Properties*.

Schéma doménových vazeb

Následující obrázek jednoduše zobrazuje výše popsané. Je zde prostředí s ***jedním lesem***, ***dvěma stromy*** a celkem ***pěti doménami***. První byl instalován řadič `DCfirma1`, čímž vznikla doména `firma.local` (a strom i les) a stala se kořenovou pro strom i celý les. K této doméně byla přidána podřazená doména `pr.firma.local`. Byl také vytvořen druhý les, který má kořenovou doménu `firma.test`. V tomto lese jsou dvě podřazené domény.

Automaticky se vytvořily vztahy důvěry, takže z domény `pr.firma.local` můžeme komunikovat s objekty ve `vyvoj.firma.test` (musí to být samozřejmě zařízeno síťově, zde řešíme logickou část). Komunikace postupuje po vztazích důvěry, nejprve na kořenovou doménu stromu `firma.local`, přes kořenový vztah lesa na `firma.test` až na `vyvoj.firma.test`.

> ***Pozn.:*** *Ve schématech je pro každou doménu znázorněn pouze jeden doménový řadič. V praxi bývají většinou minimálně dva.*



Nalezení doménového řadiče potažmo KDC

Když se uživatel přihlašuje v rámci domény, tak musí nalézt vhodný ***doménový řadič*** (DC). V případě Kerberos autentizace kontaktuje službu *Authentication Service* (AS) na *Key Distribution Center* (KDC), tato služba běží na každém DC.

Řadič se hledá pomocí ***DNS SRV záznamu*** (případně NetBIOS) pro danou doménu. Obecně jde o záznam `_ldap._tcp.`DnsDomainName (třeba *ldap. tcp.firma.local*), ale Microsoft používá speciální záznamy, aby hledal pouze Windows LDAP servery, `_ldap._tcp.dc._msdcs.`DnsDomainName (třeba *_ldap._tcp.dc._msdcs.firma.local*) Klient získá seznam všech dostupných řadičů a odešle na ně LDAP dotaz, oni odpoví základními informacemi. Klient použije první DC, který odpověděl a pomocí své IP adresy a subnetu zjistí, do jaké patří *Site*. Pomocí DNS SRV záznamu `_ldap._tcp.`SiteName`._sites.dc._msdcs.`DnsDomainName (třeba *_ldap._tcp.Praha._sites.dc._msdcs.firma.local*) zjistí seznam všech DC v dané Site. Na všechny odešle LDAP dotaz, první který odpoví, použije pro autentizaci. Výsledek je, že se upřednostňují řadiče ve stejné Site jako klient, pokud jich je více, tak se volí pomocí Round Robin.

Pro některé situace je třeba nalézt ***Global Catalog*** (GC), to se provádí obdobně pomocí záznamu `_ldap._tcp.gc._msdcs.`DnsForestName (třeba *_ldap._tcp.gc._msdcs.firma.local*). Existují i další záznamy pro GC. Identicky existuje několik záznamů pro Kerberos KDC, některé obecné a jiné speciálně pro Microsoft servery. Pro MS je hlavní `_kerberos._tcp.dc._msdcs.`DnsDomainName (třeba *_kerberos._tcp.dc._msdcs.firma.local*).

> ***Pozn.:*** *Všechny tyto SRV záznamy si registruje NetLogon služba na DC při svém startu.*

Odkazy

- Active Directory
  - What Are Domains and Forests?
  - How the Global Catalog Works
  - How Domain and Forest Trusts Work

Kerberos část 2 - AD uživatelské účty a Service Principal Name

Pondělí, 16.06.2014 08:51 | Samuraj - Petr Bouška | Microsoft admin

Druhá část seriálu o protokolu Kerberos, se zaměřením na Single Sign-On (SSO) v prostředí Microsoft Active Directory, naváže na první díl a nebude se věnovat Kerberosu, ale věcem okolo Active Directory Domain Services. Docela podrobně se podíváme na přihlašovací jména uživatelských účtů, tedy User Principal Name (UPN) a sAMAccountName. V závěru ještě popíšeme jména instancí služeb (Service Principal Name).

Uživatelské účty

Uživatelský účet je objekt v Active Directory a jedná se o ***Security Principal***, takže umožňuje autentizaci a autorizaci. Pro správu účtů můžeme použít ***Active Directory Users and Computers*** (ADUC). Kerberos slouží k autentizaci uživatelů, takže je důležité jaké uživatelské jméno z AD používá.

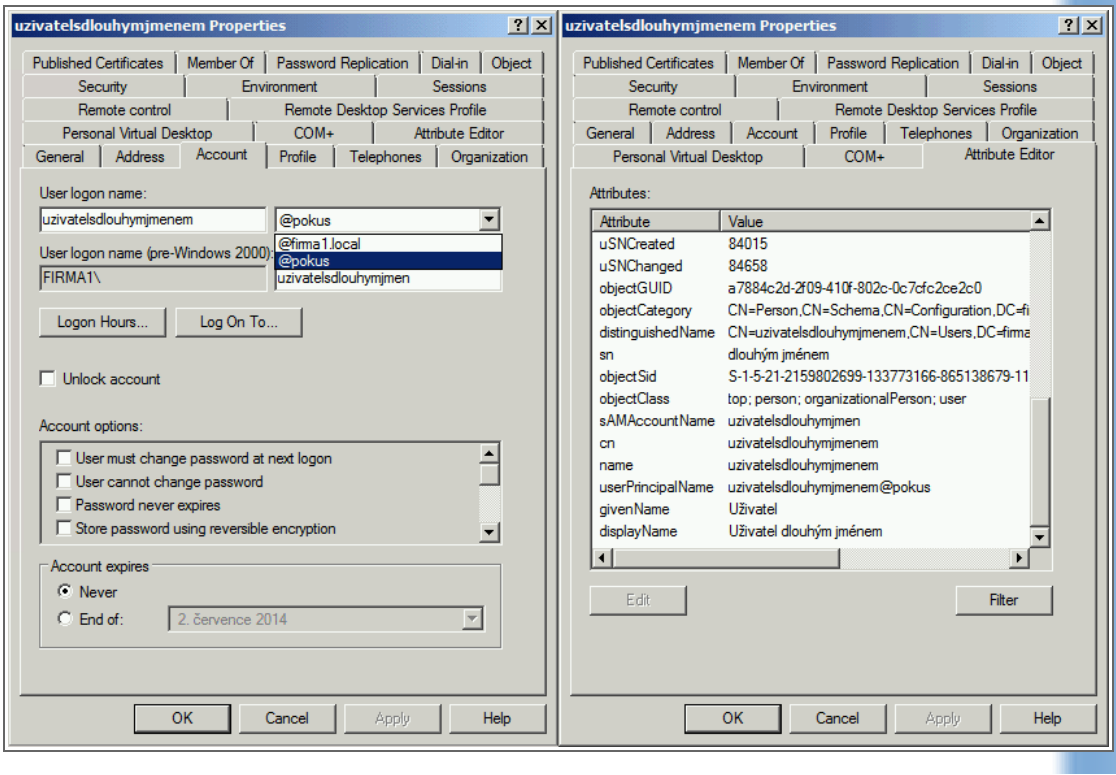
Každý objekt v AD má několik ***různých jmen*** (Naming Attributes - možností jak na objekt odkazovat či jej identifikovat). Pro provázání se používají ID, takže ostatní jména můžeme měnit, aniž by se ovlivnili různé vazby. Možné formy jména objektu, spolu s těmi, které jsou speciálně pro uživatelský účet:

- ***Relative Distinguished Name (RDN)*** - relativní LDAP jméno v rámci kontejneru (Organizational Unit), u uživatelů jde o ***Common Name (CN)***, atribut `cn`, příklad `bouska`
- ***Distinguished Name (DN)*** - globální unikátní LDAP jméno, obsahuje RDN a umístění objektu v rámci AD hierarchie, atribut `distinguishedName`, příklad `CN=bouska,CN=Users,DC=firma,DC=local`
- ***Canonical Name*** - obdoba DN v jiné notaci, jde o konstruovaný atribut (vytváří se při dotazu) `canonicalName`, příklad `firma.local/Users/bouska`
- ***Name*** - je stejné jako ***Common Name*** (CN), atribut `name`
- ***Security Identifier (SID)*** - unikátní identifikátor uživatele, používá se pro Kerberos a zařazování do skupin, atribut `objectSid`
- ***Globally Unique Identifier (GUID)*** - unikátní identifikátor objektu, atribut `objectGUID`

Active Directory používá v ***DN názvu*** tři klíčová slova dle LDAP normy, jde o ***CN - Common Name***, ***OU - Organizational Unit*** a ***DC - Domain Component***. Oproti tomu ***Canonical Name*** využívá DNS formát.

Uživatelský účet má dva možné typy ***přihlašovacího jména*** (Logon Name), jsou to zároveň také *Naming Attributes*:

- ***sAMAccountName*** - *Security Accounts Manager (SAM) Account Name*
  - označuje se také jako ***Pre-Windows 2000 Name***
  - používá se zápis `Domain\sAMAccountName`, který se označuje jako ***NetBIOS Logon Name***
  - maximální délka je 20 znaků
  - je unikátní v rámci domény a povinný
  - ukládá se do atributu `sAMAccountName`
- ***User Principal Name (UPN)***
  - označuje se jako *Internet-style login name* a je založen na RFC 822
  - skládá se ze dvou částí, ***UPN předpony*** (UPN prefix) - uživatelské přihlašovací jméno (User Logon Name) a ***UPN přípony*** (UPN suffix) - DNS doménové jméno, spojených pomocí znaku `@`, příklad `logon.name@firma.local`
  - maximální délka je 64 znaků
  - je unikátní v rámci lesa a nepovinný
  - ukládá se do atributu `userPrincipalName`



User Principal Name (UPN)

Protože věci okolo UPN nejsou úplně jednoduché, tak se na ně podíváme podrobněji.

Microsoft začal ***UPN*** používat pro doménové uživatele v době ***Windows 2000***. Mělo se jednat o ***primární přihlašovací jméno***, které by měl uživatel používat. Jeho forma je kratší než *Distinguished Name* a MS představa byla, že bude stejné jako emailová adresa (což v praxi může zjednodušit práci útočníkovi, který nemusí hádat jméno, ale pouze heslo), takže bude pro uživatele jednoduché pro pamatování. Oproti DN také není závislé na přesunu do jiného kontejneru ani přejmenování domény (přípona nemusí odpovídat doméně). Přesto bych řekl, že i v kombinaci Windows Server 2012 a Windows 8 se hojně využívá ***NetBIOS Logon Name***.

Jako ***UPN přípona*** se defaultně nabízí DNS jméno aktuální a kořenové domény, ale můžeme přidat alternativní doménové jméno (v podstatě libovolný řetězec v DNS formátu), které vytvoří administrátor v rámci lesa. Správa se provádí pomocí ***Active Directory Domains and Trusts***, kde klikneme pravým tlačítkem na stejně pojmenovanou položku a zvolíme ***Properties***.

V souvislosti se ***sAMAccountName*** Microsoft doporučuje používat začátek UPN shodný. Ale protože je zde omezení na 20 znaků, tak v řadě případů se třeba v UPN používá jméno a příjmení, ale v sAMAccountName první písmeno jména a příjmení.

Microsoft v některých materiálech uvádí, že uživatelský účet má vždy ***sAMAccountName*** i ***User Principal Name***. Jiné potvrzují to, co můžeme ověřit v praxi, že povinný je pouze atribut `sAMAccountName` a nikoliv `userPrincipalName`. Přesto je tu zajímavá informace, kterou jsem nalezl pouze v neoficiálních materiálech. A to, že ***UPN existuje vždy***, i když není atribut `userPrincipalName` vyplněn. Někde mluví o defaultním UPN, někde o ***implicitním UPN***, ale ve výsledku je to stejné (a možná ani nejde o UPN, ale prostě o jiný zápis sAMAccountName). Ať máme nebo nemáme vyplněné UPN, tak můžeme použít `username@DnsDomainName`, kde ***username*** je *sAMAccountName* a ***DnsDomainName*** je *DNS jméno domény*, kde se účet nachází. Pokud atribut `userPrincipalName` vyplníme, tak tím definujeme další ***explicitní UPN***, kde můžeme použít libovolné uživatelské jméno a příponu (tu musíme zaregistrovat v rámci lesa).

Co je ***uživatelské jméno*** je hodně důležité pro Kerberos autentizaci. Kerberos užívá termín ***Client Name*** (Principal Name) a ***Realm***, často narazíme i na použití *User Principal Name*. V AD doméně je *Realm* rovno *DNS doméně* a praktickými pokusy jsem ověřil, že *Client Name* je rovno *sAMAccountName*. Ve výsledku to odpovídá implicitnímu UPN.

Abych výše uvedené ověřil v praxi, tak jsem provedl následující testy. V **doméně** s ***DNS jménem*** `firma1.local` a ***NetBIOS jménem*** `firma1` jsem vytvořil ***UPN příponu*** `pokus` a uživatele s následujícími jmény:

```
distinguishedName CN=uzivatelsdlouhymjmenem,CN=Users,DC=firma1,DC=local
sAMAccountName    uzivatelsdlouhymjmen
cn                uzivatelsdlouhymjmenem
name              uzivatelsdlouhymjmenem
userPrincipalName uzivatelsdlouhymjmenem@pokus
```

Pak jsem se zkoušel přihlásit. Následující jména prošla:

- `FIRMA1\uzivatelsdlouhymjmen` - NetBIOS logon name
- `uzivatelsdlouhymjmenem@pokus` - zadané UPN
- `uzivatelsdlouhymjmen@firma1.local` - implicitní UPN
- `firma1.local\uzivatelsdlouhymjmen` - použití DNS jména domény v NetBIOS přihlášení

Přihlášení se nepodařilo:

- `FIRMA1\uzivatelsdlouhymjmenem`
- `uzivatelsdlouhymjmenem@firma1.local`
- `firma1.local\uzivatelsdlouhymjmenem`

V popisu u MS jsem se o přihlašování dočetl následující. Když použijeme pro ***přihlášení sAMAccountName***, tak zároveň ***určujeme doménu***, v které se účet nachází (případně se použije aktuální). Při ***použití UPN*** není jasná doména (zadáváme pouze příponu, která nemusí odpovídat doméně), takže se nejprve hledá v ***aktuální doméně***, pokud se UPN nenalezne, tak se použije ***globální katalog*** a hledá se doména, kde se účet nachází.

Při pokusech s různým přihlášením jsem také ***zachytával komunikaci***. Aby bylo vše zajímavější, tak jsem se přihlašoval z jiné důvěryhodné domény (šlo o *firma2.local*). Když se klient snaží přihlásit, tedy ***získat TGT***, tak v žádosti posílá přihlašovací údaje, které jsme zadali (viz. výše uvedené příklady). Pokud úspěšně získáme tiket, tak je v něm vždy ***DNS doména*** a uživatelské jméno ***sAMAccountName***.

Pokud zadáme tvar `doména\jméno`, tak se hledá ***jméno*** (porovnává se s atributem *sAMAccountName*) v rámci domény ***doména*** (jedno jestli jde o NetBIOS nebo DNS). Tedy v **KRG-AS-REQ** paketu se použije ***Client Name = jméno***, ***Realm = doména***.

Pokud zadáme `jméno@doména` (tedy UPN), tak se nejprve hledá v ***lokální doméně***, použije se ***Client Name = jméno@doména*** (porovnává se s atributem *userPrincipalName*), ***Realm = DNS jméno aktuální domény***. Pokud se nenalezne, tak DC hledá pomocí ***globálního katalogu***, jestli UPN existuje v jiné doméně nebo přípona odpovídá jiné doméně. Pokud ano, tak vrátí odkaz na doménu. Klient provede nový dotaz s jiným ***Realm*** a zaslaný na DC z odkazu.

Informace o uživateli na stanici

Na stanici můžeme použít určité řádkové příkazy, abychom získali informace o uživateli. První příkaz je dotaz na libovolného ***uživatele v doméně***:

```
C:\>net user bouska /domain
The request will be processed at a domain controller for domain firma.local.

User name                    bouska
Full Name                    Bouška Petr
Comment                      Ing. Petr Bouška
User's comment
Country/region code          (null)
Account active               Yes
Account expires              Never

Password last set            1.3.2014 14:54:14
Password expires             Never
Password changeable          1.3.2014 14:54:14
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   19.5.2014 16:24:41

Logon hours allowed          All

Local Group Memberships
Global Group memberships     *G Jabber IM          *Domain Users
The command completed successfully.
```

Druhá možnost je řada různých parametrů známého příkazu `whoami`, který zobrazí ***aktuálně přihlášeného uživatele***.

```
C:\>whoami /upn
bouska@firma.local
C:\>whoami /fqdn
CN=Bouška Petr,OU=IS,OU=Firma,DC=firma,DC=local
C:\>whoami /logonid
S-1-5-5-0-1835268
C:\>whoami /user /fo list
USER INFORMATION
----------------
User Name: ok\bouska
SID:       S-1-5-21-2200232112-3866456066-1594429224-1223
```

Hodně informací se můžeme dozvědět i z jednoduchého výpisu ***systémových proměnných***. Je zde vidět ***jméno počítače***, ***autentizačního severu*** (DC), ***domény*** (NetBIOS i DNS) či ***uživatele***. Ukázkový kousek výpisu:

```
C:\>set
COMPUTERNAME=BOUSKAP
LOGONSERVER=\\DC
USERDNSDOMAIN=FIRMA.LOCAL
USERDOMAIN=FIRMA
USERNAME=bouska
USERPROFILE=C:\Users\bouska
windir=C:\WINDOWS
```

Service Principal Name (SPN)

Service Principal Name (SPN) se využívají v Active Directory, kde jsou uloženy v atributu `servicePrincipalName` u účtu. Jde o jeden ze základních prvků fungování Kerberos autentizace, spolu se službou DNS. ***SPN je unikátní identifikátor služby běžící na serveru***. Jinak řečeno, je to jméno, pomocí kterého klient jednoznačně identifikuje instanci služby v rámci lesa. Každá služba, kde chceme využít Kerberos autentizaci, musí mít SPN registrované na účtu, který používá k přihlášení. Jedna služba může mít více rozdílných SPN, stejně tak k jednomu účtu může být přiřazeno více SPN.

SPN syntaxe je obecně `ServiceClass/Host:Port/ServiceName`, ale nejčastěji jde pouze o `ServiceClass/Host`. První část je název, který ***identifikuje obecně třídu služby***, příkladem je *ldap, GC, HOST, DNS, HTTP, TERMSRV, MSSQLSvc*. Pro webové servery, ať jde o protokol HTTP nebo HTTPS, se používá `HTTP`. Druhá část je ***jméno počítače***, kde služba běží. Většinou se používá ***FQDN*** (Fully Qualified Domain Name), ale pokud ke službě přistupujeme přes ***NetBIOS jméno***, tak musíme použít to. Stejně tak v některých případech potřebujeme přidat DNS alias. Naštěstí SPN můžeme registrovat více. Příkladem SPN pro webový server je `HTTP/www.firma.local`.

Ve Windows (od Windows 7 / Server 2008) máme k dispozici nástroj pro příkazový řádek **setspn**, který slouží k ***prohlížení, nastavování a mazání SPN*** v Active Directory. Pro určité operace potřebujeme patřičné oprávnění. Můžeme tak ověřit existenci určitého SPN, či vyhledat všechna SPN pro danou službu či server. Několik příkladů hledání pomocí setspn:

```
setspn -Q HTTP/server.firma.local
setspn -Q HTTP/*
setspn -Q */server.firma.local
setspn -Q */*
```

Pokud chceme hledat v rámci celého lesa, tak musíme přidat přepínač F.

```
setspn -F -Q HTTP/server.firma.local
```

Další příklad ukazuje registraci nového SPN pro webový server k účtu webserver.

```
setspn -S HTTP/www.firma.local firma\webserver
```

Group Policy Preferences, GPO, GPMC, GPME

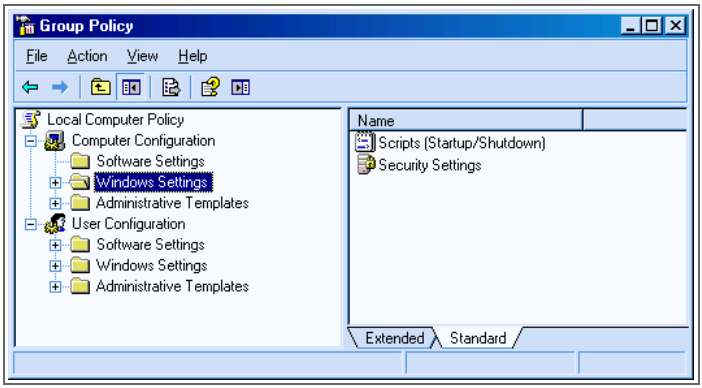
Pondělí, 16.03.2009 18:09 | Samuraj - Petr Bouška | Microsoft admin

Asi každý správce MS domény využívá Group Policy. Myslím, že se jedná o výbornou vlastnost, i když její vznik se již datuje řadu let nazpátek. Nedávno, s příchodem Windows Server 2008, rozšířil MS Group Policy o další možnosti a vlastnosti zvané Group Policy Preferences. Stručný popis těchto novinek přináší tento článek. Poznámka na okraj, od té doby co znám Group Policy, je používám i lokálně na domácím PC. Je to jednodušší než nastavovat různé vlastnosti třeba přes registry.

Group Policy Object Editor (GPedit)

Na ***Microsoft Windows 2000/XP*** a vyšších můžeme editovat ***Group Policy*** pro lokální počítač pomocí nástroje ***Group Policy Object Editor***, který je součástí systému. Jedná se o ***Snap-in*** do ***Microsoft Management Console*** - **MMC**, který můžeme spustit buď přímo zadáním `gpedit.msc` nebo pomocí `mmc.exe` (Microsoft Management Console), kde přidáme Snap-in ***Group Policy Object Editor***. Pokud využijeme `mmc`, tak si můžeme vybrat, zda se chceme připojit k *lokálnímu počítači*, k *jinému počítači v síti* či *doménové politice*, tedy k ***Group Policy Object*** (GPO).

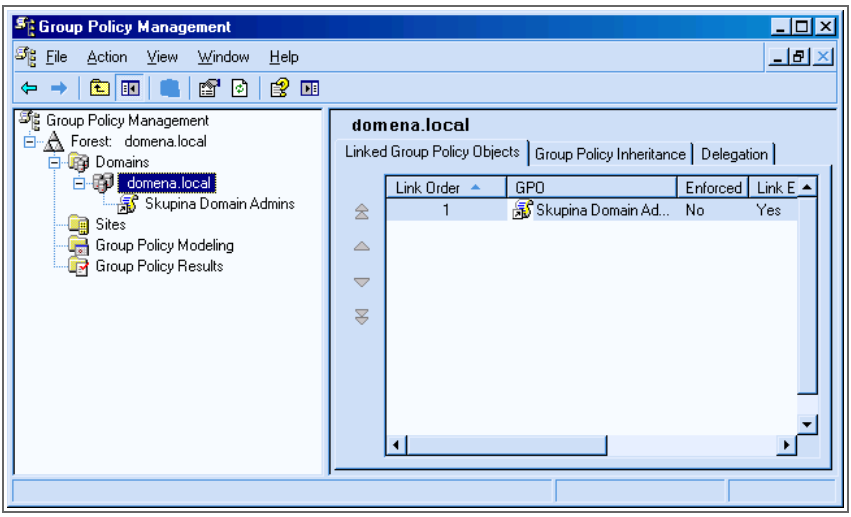
> ***Malá rada pro začátečníky.*** *MMC je velice dobrý nástroj, můžeme si do něj načíst všechny důležité snap-iny pro správu domény (já jich mám 17), a uložit. Potom otevíráme pouze jeden program (s právy doménového admina) a máme přístupné všechny potřebné nástroje.*



> ***Narazil*** *jsem na situaci, kdy jsem lokálně spustil* `gpedit.msc` *a pod kontejnery* ***Administrative Templates*** *chyběla většina voleb. Jde o to, že tento kontejner je modulární a můžeme do něj přidávat připravené šablony (přípona* `adm`*). Některé se otevírají automaticky, ale mohou se smazat. Takže je stačí opět přidat, například přidání System:*
>
> - *klikneme pravým tlačítkem na* ***Computer Configuration - Administrative Templates***
> - *vybereme* ***Add/Remove Templates***
> - *klikneme na* ***Add***
> - *vybereme soubor* `system.adm (C:\windows\inf)`

Group Policy Management Console (GPMC)

Spolu s ***Windows Server 2003*** vydal MS nový nástroj pro správu Group Policy, jedná se o **Group Policy Management Console**. Na serveru 2003 se tento nástroj nachází přímo v systému, pokud chceme doménu spravovat z Windows XP stanice (a může se jednat i o doménu WIndows Server 2000), tak jej můžeme stáhnout z MS - **Group Policy Management Console with SP1**. Spustit jej můžeme opět přímo zadáním `gpmc.msc` nebo přidáním Snap-inu Group Policy Management do **mmc**.



GPMC slouží ke správě ***doménových GPO***. Obsahuje seznam všech objektů skupinových politik v doméně, strukturu organizačních jednotek (OU) domény, kde můžeme linkovat a povolovat/vypínat jednotlivé GPO, podporuje zálohu/obnovu a import GPO. Navíc obsahuje nástroj ***Group Policy Results***, který nabízí obdobu příkazu `gpresult.exe`. Tedy pro vybraný počítač a uživatele zobrazí, jak se aplikovaly politiky. A nástroj ***Group Policy Modeling***, který simuluje aplikaci politik (z množstvím nastavení). **GPMC** nám zobrazí pouze hodnoty z ***GPO***, které byly změněny, takže jednoduše nalezneme, co daná politika provádí. Pokud chceme editovat GPO, tak se otevře `GPEDIT`, v kterém se otevře daná politika.

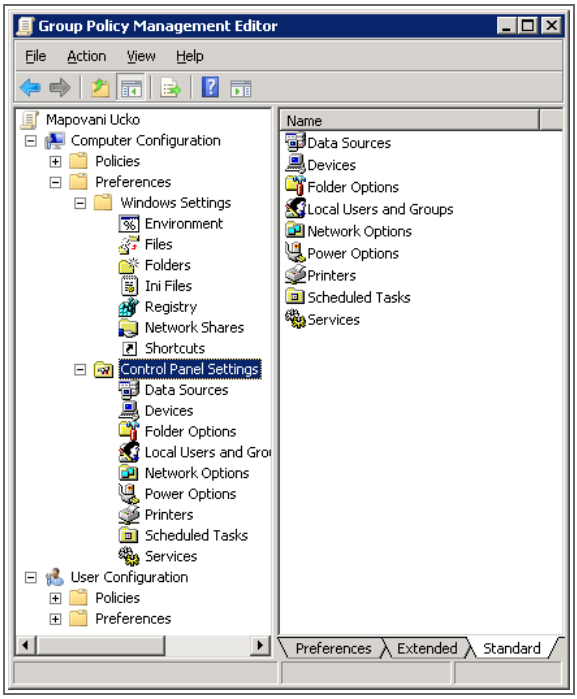
Group Policy Preferences (GPP)

S příchodem ***Windows Server 2008*** došlo k dalšímu rozšíření Group Policy, jedná se o ***Group Policy Preferences***. Abychom mohli **GPP konfigurovat**, tak potřebujeme mít nainstalován **Remote Server Administration Tools** (RSAT). Ten je součástí serveru 2008 a je možné jej doinstalovat do **Windows Vista SP1 x86** a **x64**. RSAT obsahuje upravenou verzi ***Group Policy Management Console*** spolu s **Group Policy Management Editor**.

Abychom mohli **GPP používat**, tak stačí mít doménu ***Windows Server 2003***. Nepotřebujeme žádný server s Windows Server 2008, ale potřebujeme alespoň jednu stanici s Windows Vista, z které budeme politiky vytvářet. Standardně se budou GPP uplatňovat pouze na server 2008, ale pro **Windows XP SP2 x86** a **x64**, **Windows Vista x86** a **x64** a **Windows Server 2003 SP1 x86** a **x64** existuje ***Group Policy Preference Client Side Extensions*** (CSE). CSE můžeme nainstalovat manuálně nebo jako volitelný update z Windows Update `kb943729`.

Pokud s RSAT otevřeme **GPMC** a na nějaké politice zvolíme ***edit***, tak se neotevře GPedit, ale nový ***Group Policy Management Editor*** (dá se spustit přímo jako `gpme.msc`). Ten pod ***Computer*** a ***User Configuration*** obsahuje nové rozdělení na **Policies** a **Preferences**.

Group Policy Preferences se ukládají stejně jako ***Policies*** (dohromady jako GPO) do složky `SYSVOL` na doménových řadičích. Ale konfigurace GPP je uložena v XML souborech, takže klient musí obsahovat XML parser.



GPP přináší více než 20 nových rozšíření GP. Navíc je zde i rozdílný způsob aplikace. ***Politiky***, které jsme nastavili, se vynutí na stanici a není možné je změnit (volba zašedne). Kdežto ***preference*** můžeme upravit a buď se po nějaké době (***group policy refresh interval***, default 90 minut) opět přenastaví nebo zvolíme možnost, že se uplatní pouze jednou (možnost ***Apply once and do not reapply***). Také je zde vlastnost zvaná **Item-level targeting**. Ta nám dovoluje velice detailně určit, na jaké počítače se má daná GPP uplatnit. V podmínkách můžeme použít řadu hodnot, jako MAC adresu, LDAP dotaz, WMI, volné místo na disku, atd. Pokud se aplikovala ***politika*** a počítač/uživatel přestal být v rozsahu aplikace této politiky, tak se všechna nastavení odstranila. U **GPP** je defaultní chování, že hodnoty zůstanou nastaveny stále (ale můžeme změnit zaškrtnutím ***Remove this item when it is no longer applied***).

> ***Pozn.:*** ***Policies*** *se označují jako* ***registry-based settings****, jejich nastavení se provede pomocí registrů.* ***Preferences*** *oproti tomu mohou konfigurovat více než jen registry.*

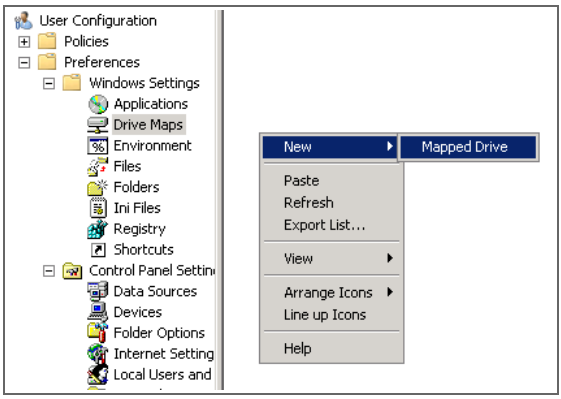
Některé z možností, které přináší GPP jsou: mapování síťových disků, úpravy lokáních uživatelů a skupin, kopírování souborů, vytváření zástupců, práce s registry, nastavení proměnných prostředí, konfigurace VPN spojení, nastavování vlastností složek a řada dalších.

Níže uvádím dva příklady použití GPP na operace, které se dříve prováděli komplikovaněji.

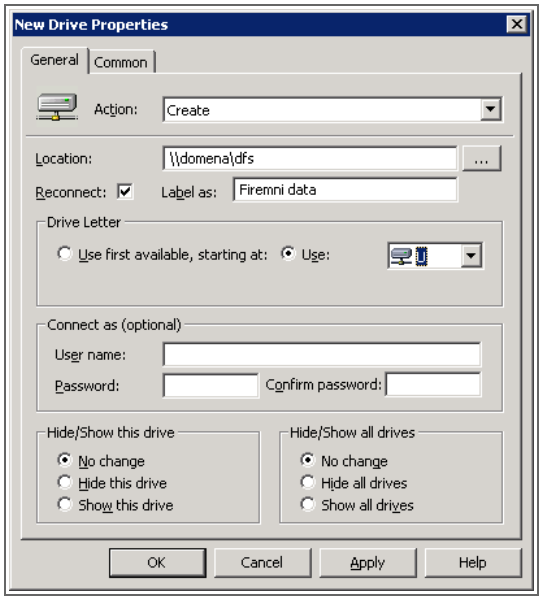
Mapování síťového disku

Myslím, že je běžnou operací ve firemním prostředí, mapovat všem (nebo určitým skupinám) uživatelům, nějaké společné datové úložiště jako síťový disk. Donedávna to bylo třeba provádět pomocí login skriptu. Dnes máme GPP, které to řeší elegantněji.

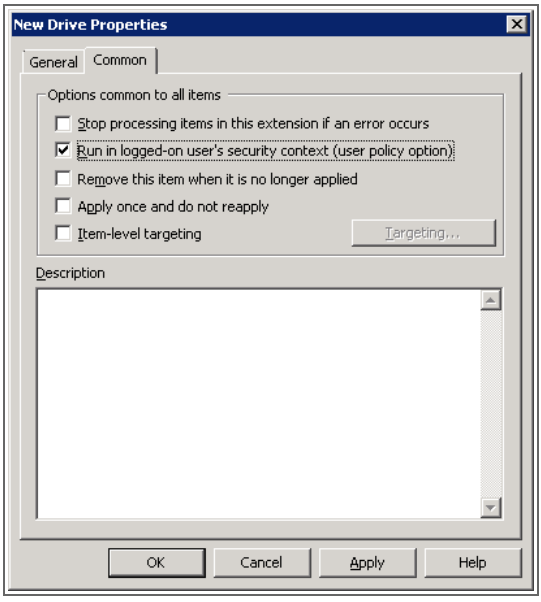
- spustíme ***Group Policy Management*** (Start > Administrative Tools)
- doklikáme se na kontejner (OU), na který chceme nastavení aplikovat
- klikneme na něj pravým tlačítkem a zvolíme ***Create a GPO in this domain, and Link it here*** …
- zadáme jméno a klikneme **OK**, tím se nám vytvořil **GPO**
- klikneme na něj pravým tlačítkem a zvolíme ***Edit***
- GPO se otevře v ***Group Policy Management Editor***
- rozklikneme ***User Configuration > Preferences > Windows Settings > Drive Maps***
- klikneme pravým tlačítkem a zvolíme ***New > Mapped Drive***



- zvolíme parametry pro mapovaný disk, v praxi se mi osvědčila akce ***Create*** (zkoušel jsem Replace, která by měla v případě existence mapovaného disku, jej nejprve zrušit, ale nějak se mi nedařilo)



- na záložce ***Common*** zaškrtneme ***Run in logged-on user's security context (user policy option)***, aby se disk připojoval s právy aktuálně přihlášeného uživatele

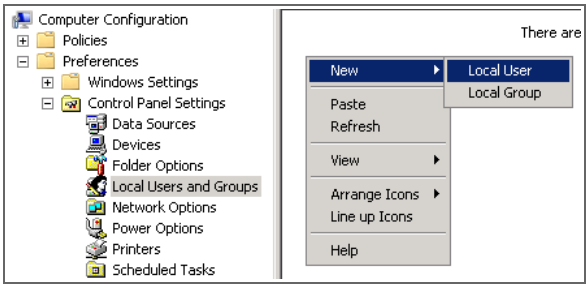


- pozavíráme okna a máme hotovo

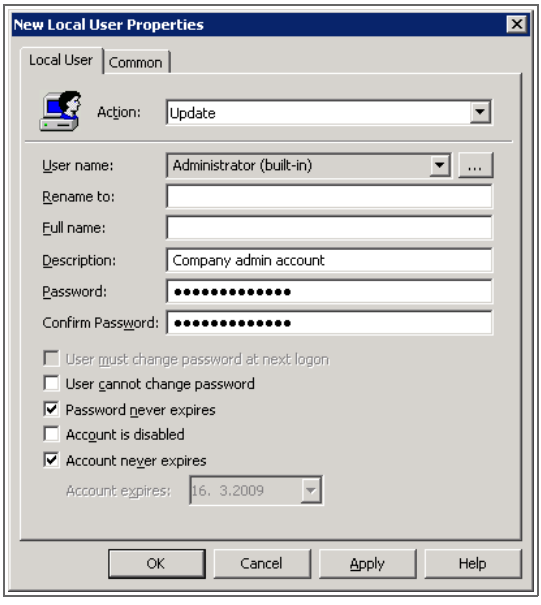
Změna hesla lokálního administrátora

Druhou zajímavou úlohou, kterou můžeme řešit pomocí GPP a dříve to byl problém, je změna parametrů lokálního účtu (hlavně heslo).

- spustíme ***Group Policy Management*** (Start > Administrative Tools)
- doklikáme se na kontejner (OU), na který chceme nastavení aplikovat
- klikneme na něj pravým tlačítkem a zvolíme ***Create a GPO in this domain, and Link it here*** …
- zadáme jméno a klikneme **OK**, tím se nám vytvořil **GPO**
- klikneme na něj pravým tlačítkem a zvolíme ***Edit***
- GPO se otevře v ***Group Policy Management Editor***
- rozklikneme ***Computer Configuration > Preferences > Control Panel Settings > Local Users and Groups***
- klikneme pravým tlačítkem a zvolíme ***New >Local User***



- vyplníme údaje o upravovaném uživateli, v našem případě zvolíme akci ***Update***, protože chceme upravovat již existující účet (to je důležité, protože zůstává zachováno původní SID a tudíž i oprávnění)
- protože chceme upravit účet lokálního administrátora, tak je vybereme z nabídky (ostatní zadáme ručně), můžeme jej přejmenovat (což je dobré bezpečnostní opatření) a nastavit další parametry



- pozavíráme okna a máme hotovo

Samozřejmě je zde otázka bezpečnosti takovéhoto nastavení. Dobrá zpráva je, že hesla jsou v GPP uložena

šifrována pomocí ***256 bitové AES enkrypce***. Ale nachází se takto uložena v XML souboru, který může kdokoliv získat.

Autor: **Petr Bouška**

Související články:

Group Policy

Základem centrální správy počítačů v doméně jsou určitě skupinové politiky (Group Policy). Pomocí nich můžeme řídit nastavení, bezpečnost a chování pracovních stanic a serverů.

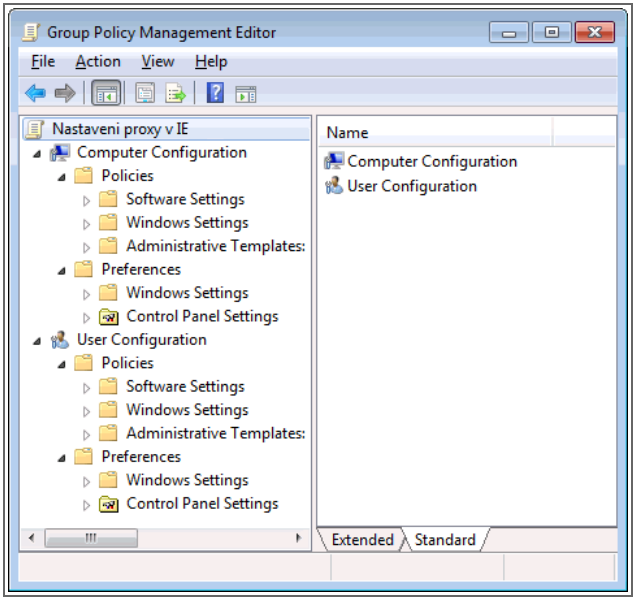
- **Nastavení přihlašování do domény pomocí Smart Card** [12.07.2006 16:04]
- **Group Policy Preferences, GPO, GPMC, GPME** [16.03.2009 18:09] ..... **právě čtete**
- **Group Policy - řízení aplikace politik** [12.12.2010 17:42]
- **Group Policy - Administrative Templates** [19.01.2011 18:16]
- **Group Policy - Politiky hesel a zamykání účtů** [21.02.2011 18:46]
- **MS Outlook a konfigurace pomocí Group Policy** [23.03.2011 16:30]
- **Auditování AD DS objektů ve Windows Server 2008** [26.03.2011 19:23]
- **Microsoft Certification Authority Auditing** [13.01.2014 17:05]

Group Policy - řízení aplikace politik

Neděle, 12.12.2010 17:42 | Samuraj - Petr Bouška | Microsoft admin

Group Policy (skupinové politiky) slouží k centrální správě počítačů s pomocí Active Directory. Hlavně se tedy využijí pro počítače zařazené do domény. Můžeme ale využít i lokální politiky (Local Group Policy), které nabízí o něco omezenější funkčnost, ale fungují i na samostatné počítače. Nově máme nejen Group Policy, ale také Group Policy Preferences, které mají trochu odlišné chování. Zde se podíváme na to, jak se Group Policy aplikují na objekty (uživatelé a počítače), jak je můžeme filtrovat, využít Loopback processing nebo hledat chyby při aplikování.

Nastavení a vytváření **Group Policy** dnes provádíme nejčastěji pomocí ***Group Policy Management Console*** (GPMC, ten budeme také používat v celém článku), dříve (nebo pro lokální politiky) se používal *Group Policy Object Editor* (GPedit). Nastavení se ukládají do ***Group Policy Object*** (GPO). Group Policy mají několik pevně daných částí (podle verze OS) a pak možná rozšíření pomocí *Administrative Templates*. Nejčastěji upravují registry, ale také mohou měnit bezpečnostní nastavení, instalovat SW, nastavovat Internet Explorer, apod. Politiky mají dvě hlavní části ***Computer Configuration*** a ***User Configuration***.



Group Policy nejčastěji fungují na principu úprav registrů na klientské stanici či serveru. Část ***Computer Configuration*** se týká nastavení pro počítač, tyto nastavení se mohou aplikovat na počítačové objekty v Active Directory (uplatňuje se na vybraný počítač a nezáleží na přihlášeném uživateli). Upravují větev registrů `HKEY LOCAL MACHINE` (HKLM), například `HKLM\Software\Policies\Microsoft\Windows`, `HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\System`.

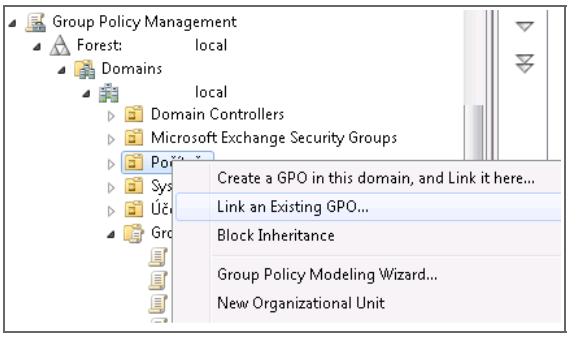
Část ***User Configuration*** se týká nastavení pro uživatele, politiku s tímto nastavením můžeme aplikovat na uživatelské účty v Active Directory (uplatňuje se na vybraného uživatele a nezáleží na jakém počítači). Upravuje větev registrů `HKEY CURRENT USER` (HKCU), například `HKCU\Software\Policies\Microsoft\Windows`, `HKCU\Software\Microsoft\Windows\CurrentVersion\Policies`.

Politiky aplikované na počítač se standardně uplatňují při ***startu počítače***, politiky aplikované na uživatele probíhají při ***přihlášení uživatele***. Obě se pak aplikují při periodické obnově Group Policy (to je standardně každých 90 minut + náhodný posun o až 30 minut). Aplikaci politik také můžeme vynutit ručně pomocí příkazu

```
gpupdate /force
```

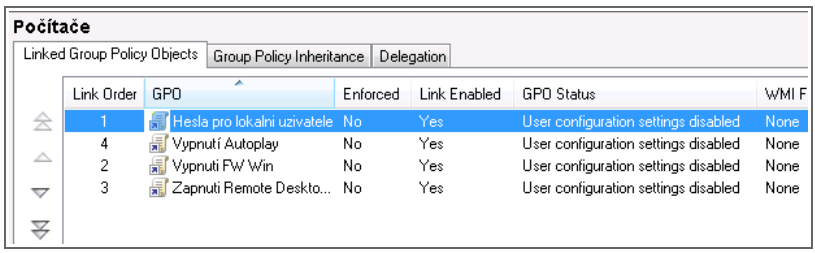
Aplikace politik

Group Policy se aplikují tak, že je spojíme s nějakým kontejnerem (link to). Jedná se o Active Directory kontejnery ***site***, ***doména*** (domain) či ***organizační jednotka*** (OU - Organization Unit).



Při aplikaci politik se uplatňuje ***dědění*** (inheriting), dané hierarchickou strukturou AD, a ***souhrnný účinek*** (cumulative). To znamená, že se politika, aplikovaná na OU, projeví na všech počítačích a uživatelích, kteří se nachází v této a ve vnořených OU. Když je více politik, tak se jejich účinek spojuje dohromady. Samozřejmě, pokud politika obsahuje pouze část pro počítač, tak se sice na uživatele aplikuje, ale nemá žádný efekt. Navíc záleží na tom, jestli počítač nebo uživatel má na danou politiku práva (tak se využívá filtrování).

Politiky se ***zpracovávají postupně***, později zpracovaná politika může přepsat nastavení předchozí. Postupuje se v pořadí lokální GPO, site, doména, OU, poslední je OU nejblíže k objektu. Politiky na jednom kontejneru se zpracovávají v pořadí, v jakém jsou nastaveny na záložce *Linked Group Policy Objects*.

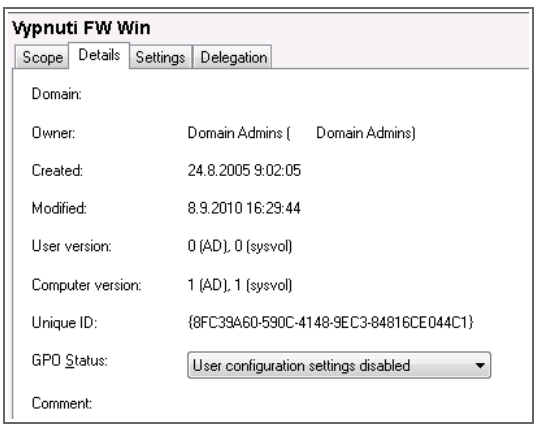


Vypnutí části politiky

I když politika obsahuje pouze jednu část (počítačovou nebo uživatelskou), tak se standardně zpracovává na všech objektech (uživatelé a počítače), které se nachází v rozsahu aplikace. Zpracování každé politiky, i prázdné, zatěžuje cílový počítač. Takže je vhodné u politik, kde využíváme pouze jednu část, tu druhou vypnout. Také je vhodné minimalizovat počet politik, tím že je sloučíme do jedné.

Na politice se přepneme na záložku ***Details*** a zde je položka ***GPO Status***. Můžeme nastavit jednu ze čtyř hodnot.

- ***Enabled*** - vše zapnuto
- ***All settings disabled*** - vše vypnuto
- ***Computer configuration settigs disabled*** - aplikuje se pouze uživatelská část
- ***User configuration settigs disabled*** - aplikuje se pouze počítačová část



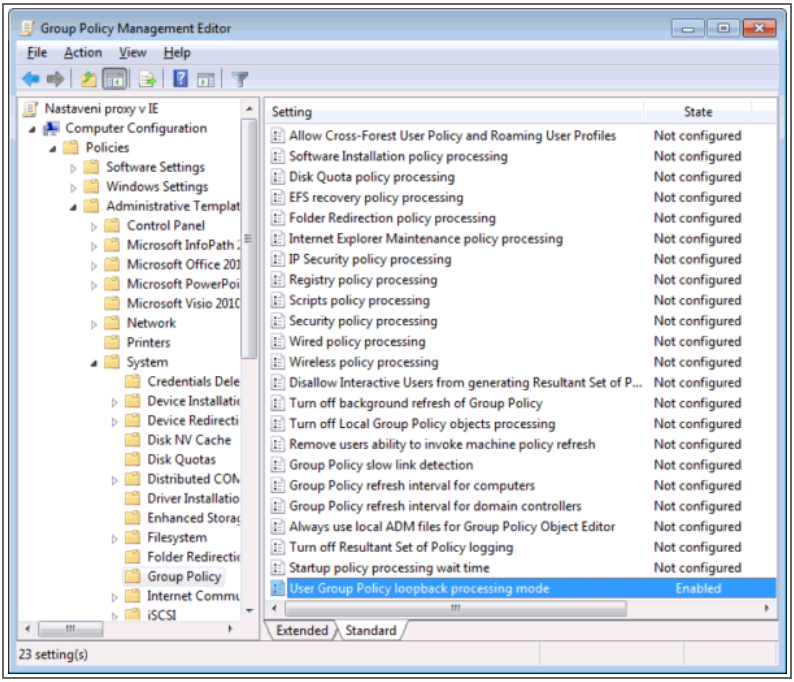
Loopback processing

Občas se může stát, že potřebujeme aplikovat nějaké ***uživatelské nastavení***, ale přiřadit jej ***počítači***. Aby jakýkoliv uživatel, který se přihlásí na tento počítač, měl dané nastavení. Ale pokud se připojí jinde, tak i nastavení může být jiné. To se týká například terminálových serverů, notebook vs. pracovní stanice, apod.

Abychom toho dosáhli, tak můžeme využít Group Policy v režimu zpracování ***Loopback***. Může být buď v módu ***Replace***, kdy nahrazuje nastavení, které je aplikované pro uživatele, nebo ***Merge***, kdy se obě nastavení spojí. Nastavení se provede v dané politice. Hodnota ***Computer Configuration - Administrative Templates - System - Group Policy - User Group Policy loopback processing mode***. Nastavíme ***Enabled*** a vybereme Mode ***Replace*** nebo ***Merge***.

Takto příjemné mi to připadalo, když jsem si o této metodě četl. Bohužel v praxi člověk zjistí, že je toto řešení téměř nepoužitelné. Nejedná se totiž o nastavení dané politiky, abychom mohli zařídit, že to jedno určité nastavení podle počítače se provede na uživateli. Ale jde o globální nastavení, kterým nastavíme chování počítačů, na které se aplikuje. Takže se potom uplatní na všechny politiky.

Znamená to tedy, když máme nastavený mód ***Replace***, tak se nepoužijí žádné politiky, které jsou aplikované jen na uživatele. Musíme uživatelské politiky přiřazovat počítačům. Můžeme politiky nastavit třeba pro celou doménu, ale přijdeme pak o cílení politik na určité OU s uživateli (což nám asi vždy bude vadit). Nebo můžeme použít mód ***Merge***, tehdy se uplatňují politiky, které jsou aplikované na uživatele, i politiky na počítač. Jenže tím se prodlouží doba zpracovávání politik, mohou vzniknout neočekávané situace a těžce se pak řeší troubleshooting.



Omezování aplikace politiky

Filtrování aplikace politiky

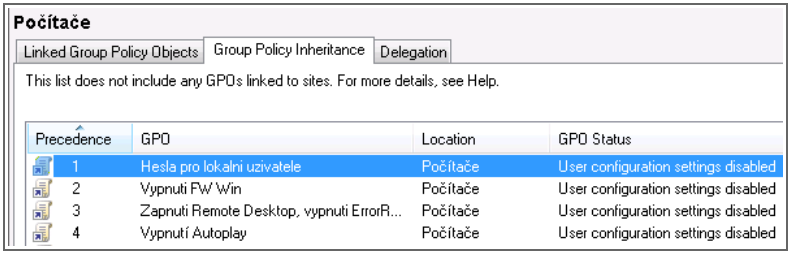
Pro dynamické řízení, na jaké počítače a uživatele se má politika aplikovat, můžeme použít ***WMI Filtering*** nebo ***Security Filtering***. Pomocí ***WMI*** (Windows Management Instrumentation) můžeme vytvořit filtry, které zjistí různé údaje o počítači a politika se pak aplikuje pouze na ty, které splňují danou podmínku. Příkladem je třeba určitá verze OS. ***Security*** filtrování zase povolí politiku pouze těm, kteří jsou členem určité bezpečnostní skupiny nebo určité počítače či uživatelé.

Zrušení dědění

Jiná možnost jak řídit aplikaci politiky je, že pro určitý kontejner můžeme zrušit dědění. Klikneme pravým tlačítkem na objekt a zvolíme ***Block Inheritance***, tím se přestanou zpracovávat všechny nadřazené politiky mimo vynucených.

Jinou možností je naopak vynutit aplikaci politiky, aby například nemohla být přepsána (a ani blokována). Klikneme pravým tlačítkem na politiku a zvolíme ***Enforced***.

Jaké politiky jsou na objektu uplatňovány, se můžeme podívat na záložce ***Group Policy Inheritance***. Vidíme zde i pořadí zpracování.



Kontrola aplikování politik

Součástí ***Group Policy Management Console*** jsou dva nástroje, které můžeme použít při řešení problémů. ***Group Policy Results*** nám pro vybraný počítač a uživatele vrátí online informace o aplikovaných politikách, včetně chyb. Informace získává přímo z DC a daného počítače. Jedná se o velice užitečný nástroj. Druhý nástroj je ***Group Policy Modeling***, který provede simulaci aplikace politik a zobrazí výsledek.

Group Policy - Administrative Templates

Středa, 19.01.2011 18:16 | Samuraj - Petr Bouška | Microsoft admin

Group Policy se skládají ze dvou hlavních částí, konfigurace počítače (Computer Configuration) a konfigurace uživatele (User Configuration). Obě části mají rozdílné jednotlivé položky (možnosti nastavení), ale mají společné kategorie. To je Software Settings, Windows Settings a Administrative Templates. První dvě jsou pevně dány, kdežto Administrative Templates se načítají z konfiguračních souborů (ADM a ADMX/ADML) a můžeme je i sami rozšiřovat. Obsahují politiky založené na registrech. V tomto článku se budeme věnovat právě oblasti Administrative Templates a k jakým změnám došlo při příchodu Windows Server 2008.

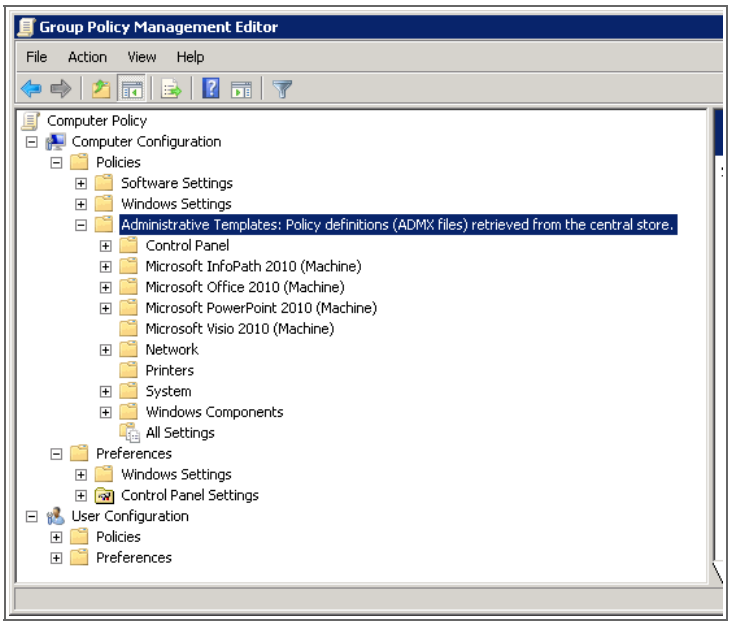
*Tento článek vyšel také jako součást **Microsoft TechNet Flash zpravodaje** pod názvem **Group Policy Administrative Templates ve Windows Server 2008**.*

Soubory Administrative Templates (AT)

Administrative Templates soubory (máme ADM a ADMX svázané s ADML, když budu mluvit obecně, budu říkat ADM/ADMX soubory), obsahují šablony, které dovolí konfigurovat nějakou vlastnost pomocí změn v registrech. Když vytvářím ***Group Policy Object*** (GPO, jednoduše politiku) pomocí ***Group Policy Object Editor*** (GPOE) či ***Group Policy Management Console***, tedy ***Group Policy Management Editor*** (GPME), tak se ADM/ADMX soubory načtou a v editoru se zobrazí určité položky. Když nějakou položku změním, tak se uvnitř politiky uloží do souboru `registry.pol`. Ten se pak aplikuje na počítače/uživatele. Šablony jsou potřeba pouze pro vytváření politiky, ale ne již pro vlastní aplikaci na klientovi.

Základní ADM/ADMX soubory se distribuují spolu s operačním systémem. ADM/ADMX soubory v sobě obsahují nejen nastavení pro aktuální OS, ale pro všechny platformy podporované v dané době (to znamená, že na Windows XP máme i informace k serverovým nebo starším OS). Také nové defaultní ADMX soubory obsahují vše, co bylo ve standardních ADM souborech.

Na internetu můžeme stáhnout různé rozšiřující ADM/ADMX soubory, například pro MS Office. Případně i ty ADM/ADMX soubory, které jsou součástí různých verzí Windows. Není ani problém si vytvořit vlastní ADM/ADMX soubor s nějakým nastavením.



Soubor ADM

Původní formát souborů pro šablony od Windows 2000 po 2003/XP. Jedná se o textový soubor, který je jazykově závislý (pro každý jazyk máme nový soubor stejného názvu, takže konfigurace politiky se může provádět pouze v jednom jazyce). ADM soubory můžeme použít i v novějších verzích OS (Windows 7), ale standardní soubory, distribuované s OS, jsou již pouze ADMX a tyto soubory mají přednost před ADM.

Pokud používáme ADM soubory při vytváření politiky, tak se tyto ***soubory ukládají do vytvořené politiky*** (do Sysvol adresáře v doméně). Takže politika má velikost několik MB místo pár kB.

Soubor ADMX

Jedná se o náhradu ADM souborů, která přišla s Windows Vista a Windows Server 2008. ADMX je soubor v XML formátu, který je jazykově neutrální. K určitému ADMX souboru potřebujeme odpovídající ADML soubor. Tím je podporována multijazyčnost (více správců může editovat jednu politiku, každý v jiném jazyce). ADMX/ADML soubory se navíc neukládají do vlastní politiky, takže ta je výrazně menší. Volitelně můžeme použít ***Central Store*** (více dále).

Soubor ADML

Doplněk ADMX souboru, který obsahuje jazykové informace (texty pro politiku v určitém jazyce). Ve složce s ADMX soubory se vytváří podadresáře pro určité jazyky (jako `en-US`, `cs-CZ` či `de-DE`) a do nich se ukládají ADML soubory se stejným názvem jako ADMX.

Načítání ADM/ADMX souborů

ADMX soubory

V administračních nástrojích (GPOE, GPME) se ***ADMX soubory*** automaticky hledají ve dvou cestách. V ***Central Store***, tedy síťovém úložišti v doméně, jehož adresa je `\\FQDN\SYSVOL\FQDN\policies\PolicyDefinitions`. Pokud se zde nenalezne, tak se hledá v ***lokálním úložišti*** v `c:\Windows\PolicyDefinitions`. Pokud Central Store existuje a je dostupný, tak se lokální soubory nepoužijí.

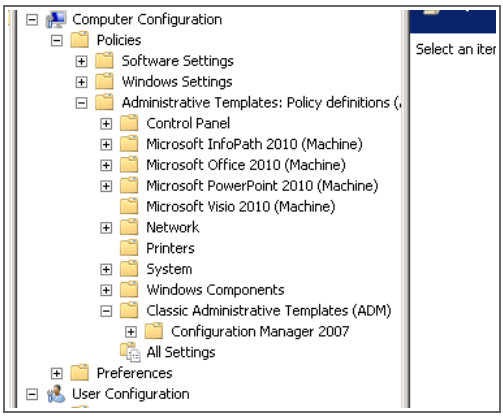
Vytvoření Central Store

Pokud chceme vytvořit **Central Store**, tak stačí založit daný adresář na jednom doménovém řadiči (provede se replikace na ostatní) a zkopírovat do něj ADMX a ADML soubory například z Windows Server 2008 R2. Vždy je potřeba ***udržovat nejnovější verzi*** těchto souborů, aby obsahoval všechna nastavení (například po instalaci Service Pack). V současnosti nejnovější verze je pro *Windows Server 2008 R2/Windows 7*.

ADM soubory

Naproti tomu ***ADM soubory*** jsou standard ně (pokud je máme v systému) uloženy v `c:\Windows\inf`. Při vytváření politiky (ve Windows, kde ještě nejsou ADMX) se načítají některé defaultní a můžeme je ručně přidávat nebo ubírat. Klikneme pravým tlačítkem na ***Administrative Templates*** v editoru a zvolíme **Add/Remove Templates.** Defaultní ADM soubory jsou `conf.adm`, `inetres.adm`, `system.adm`, `wmplayer.adm`, `wuau.adm`. Těchto 5 souborů má dohromady skoro 4 MB a vkládají se do každé vytvořené politiky.

Pokud používáme systém s ADMX soubory, tak můžeme stále přidat ADM soubory, ty se zobrazí pod skupinou `Classic Administrative Templates (ADM)`.



Uložení GPO

Politiky (GPO) se v doméně ukládají (na doménové řadiče) do cesty `\\FQDN\SYSVOL\FQDN\policies`. Zde je adresář pro každou politiku, který má název `{GPO GUID}` (ID politiky v závorkách). Pokud je to politika vytvořená pomocí ADM souboru, tak se zde nachází podadresář `Adm`, který obsahuje všechny ADM soubory, které byly při vytváření přidány. Takže je odsud můžeme i zkopírovat, pokud je potřebujeme. Ale určitě je efektivnější používat nové ADMX soubory uložené v Central Store (jednotlivé politiky jsou pak mnohem menší), to znamená vytvářet a spravovat politiky v nových verzích Windows.

Převod politiky s ADM soubory na novou

Jak jsme řekli, je lepší vytvářet nové politiky v novějších verzích OS (třeba Windows 7) a v nich provádět i jejich správu (nové položky bychom ve starších OS neviděli a rovnou by se do GPO přidali ADM soubory). Staré politiky můžeme editovat v nových OS, ale tím se ADM soubory neodstraní (i když pokud nejsou speciální, tak je ani při editaci nevyužijeme – pokud existuje odpovídající ADMX, tak se použije ten).

Čistým řešením, jak se zbavit ADM souborů, je vytvořit GPO znovu. Lákala by možnost importovat nastavení ze staré politiky do nové, bohužel při importu se vloží i ADM soubory. Přesto se někdy možnost importu může hodit, ta zkopíruje nastavení (a pouze to, ne práva, delegaci apod.) ze zazálohované politiky. Zálohu můžeme provést jako jeden krok průvodce. Import provedeme pomocí ***Group Policy Management Console***, proklikáme se na ***Group Policy Objects*** a klikneme pravým tlačítkem na politiku, do které chceme importovat nastavení. Z menu zvolíme ***Import Settings*** a projdeme průvodce.

> ***Pozn.:*** *Zálohu všech politik můžeme provést tak, že klikneme pravým tlačítkem na Group Policy Objects v Group Policy Management Console a zvolíme* ***Back Up All****.*

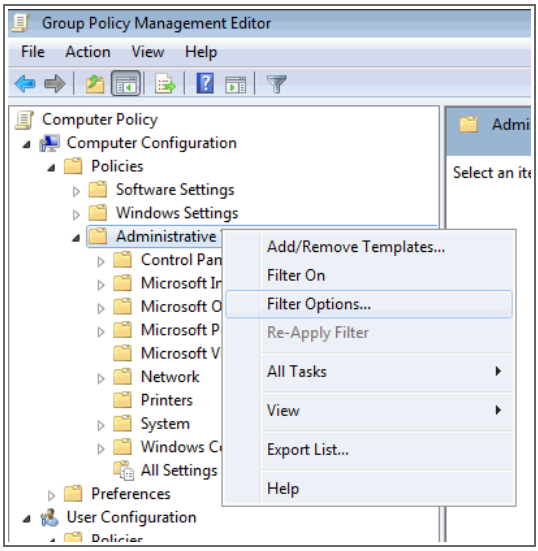
Možností, jak odstranit ADM soubory z GPO, je otevřít politiku v novém OS. Kliknout pravým tlačítkem na ***Administrative Templates*** a zvolit **Add/Remove Templates** a odstranit všechny připojené soubory. Když se pak podíváme do adresáře politiky, tak je adresář Adm prázdný. A když se podíváme na politiku, třeba v ***Group Policy Management Console***, vidíme, že nastavení zůstalo.

Konverze ADM souboru do ADMX

Pro převod starých ADM šablon na nové ADMX můžeme použít nástroj **ADMX Migrator** . Ten můžeme využít i k editaci ADMX souboru nebo k vytvoření nového.

Vyhledávání v Administrative Templates

V minulosti byl vždy velký problém cokoliv v nastavení politik nalézt. Položek je zde několik tisíc a chyběla základní možnost vyhledávání. Dnes máme naštěstí k dispozici dvě pomocné operace, zobrazení všech položek na jednom místě a filtrování.



Všechna nastavení

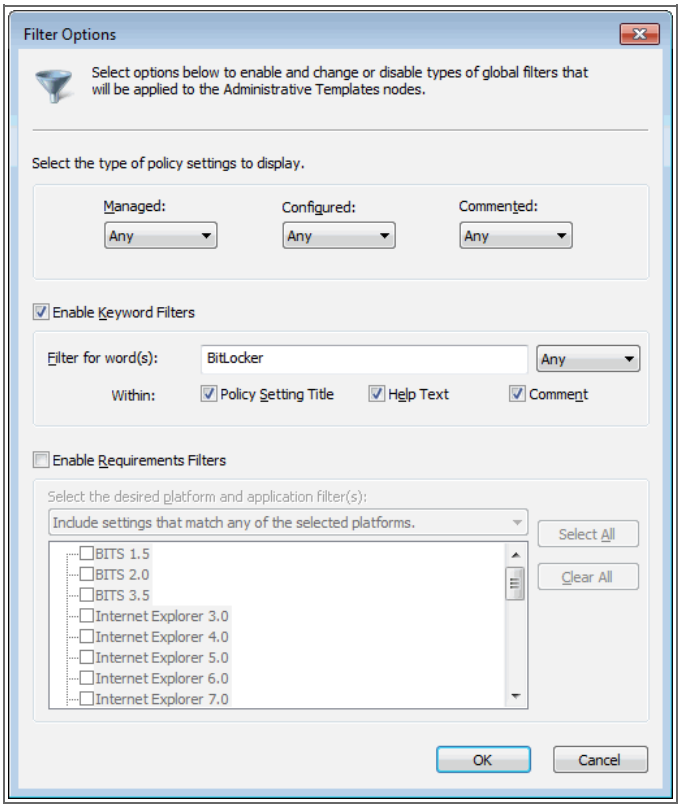
Nově je v administračních nástrojích pro editaci GPO (GPOE, GPME), pod položkou ***Administrative Templates***, k dispozici složka **All Settings**, která obsahuje všechna nastavení ze všech načtených ADM/ADMX souborů. Takže můžeme vyhledávat položky podle jména.

Filtrování Administrative Templates

Druhou novinkou editovacích nástrojů je možnost filtrování nastavení z ADMX souborů. Když klikneme pravým tlačítkem na ***Administrative Templates*** nebo nějakou podsložku, tak máme v kontextovém menu k dispozici položky **Filter On** a **Filter Options**.

Když zvolíme **Filter Options**, tak se zobrazí dialog, kde můžeme zadat parametry filtrování. Můžeme si vypsat položky, které jsou ***nastavené*** (Configured), u kterých je ***komentář*** (Commented), které jsou určeny pro určitou ***platformu*** (Requirements Filters). A asi hlavní je ***filtrování podle klíčových slov*** (Keyword Filters), kde zadáme jedno nebo více slov a určíme jeho výskyt v názvu politiky, popisu či komentáři. Po kliknutí na OK se rovnou filtrování zapne (Filter On).

> ***Pozn.:*** *Pokud zadáte klíčové slovo a nedostanete žádný výsledek, tak to může být způsobeno bugem v MS nástroji. Je potřeba přepnout klávesnici na anglickou a znovu zadat parametry filtrování (Filter Options).*



Odkazy

- Group Policy popis MS
- Administrative Templates (ADMX) for Windows Server 2008 R2 and Windows 7
- Administrative Templates (ADMX) for Windows Server 2008

Group Policy - Politiky hesel a zamykání účtů

Pondělí, 21.02.2011 18:46 | Samuraj - Petr Bouška | Microsoft admin

V každé síti je třeba řídit bezpečnost. Jedním z hlavních útoků je prolomení hesla a proto musíme mít definovánu politiku, jak mají být hesla silná a co se děje při zadání špatných hesel. V Microsoft prostředí na to využijeme Group Policy, které nám dovolí definovat a vynutit parametry hesel. Hesla můžeme definovat pro lokální účty a v doménovém prostředí také pro doménové účty. Windows Server 2008 navíc přináší možnost definovat více rozdílných politik pro doménové účty.

Tento článek vyšel také na Microsoft TechNet Blogu CZ/SK pod názvem Politiky hesel a zamykání účtů pomocí Group Policy.

Anglicky se tato oblast označuje jako **Password and Account Lockout Policy**. ***Politika hesel*** (Password Policy) nám dovoluje specifikovat parametry hesla, jako je *minimální počet znaků*, zda musí být heslo *komplexní*, zda se musí hesla v *pravidelném intervalu měnit* (minimální a maximální stáří hesla), jestli a kolik *starších hesel se pamatuje* (ty se pak nedají nastavit znovu).

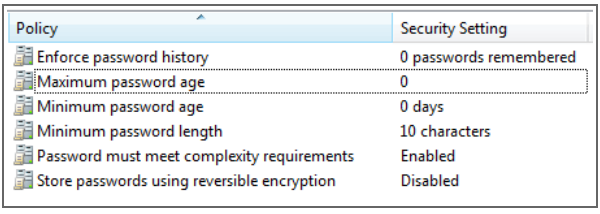
Komplexní heslo v tomto případě znamená, že nesmí obsahovat část uživatelova jména a musí obsahovat znaky minimálně ze tří skupin (ze čtyř možných), což jsou:

- velké písmeno (A-Z)
- malé písmeno (a-z)
- číslice (0-9)
- speciální (ne-alfanumerický) znak (třeba *#,.@%)

V Group Policy se nastavení politiky hesel nachází v cestě `Computer Configuration/Windows Settings/Security Settings/Account Policies/Password Policy`. Nastavení v této politice se uplatňuje při změně hesla. To znamená, že když změníme nastavení politiky, tak se pro uživatele nic nemění, dokud si nebudou měnit heslo, pak musí dodržet tyto parametry.

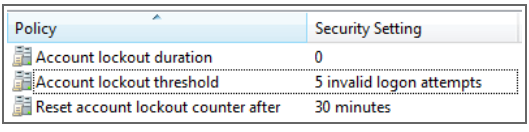
Změna hesla může být vynucena intervalem (maximální stáří hesla) nebo nastavením na účtu ***User must change password at next logon***. Pokud je na uživatelském účtu nastaveno ***Password never expires***, tak se pro tento účet neuplatňuje stáří hesla (není povinnost měnit heslo, i když je nastavena politika). Není možno nastavit současně *User must change password at next logon* a *Password never expires*. Pokud tedy nastavíme politiku hesel, a chceme mít hesla podle ní, tak máme jedinou možnost. Nastavíme na účty atribut *User must change password at next logon* a vypneme (pro jistotu) *Password never expires*. To můžeme udělat jednoduše pomocí ***PowerShellu***, například:

```
Get-ADUser -Filter * -SearchBase "OU=Firma,DC=firma,DC=local" | Set-ADUser -PasswordNeverExpires $false
Get-ADUser -Filter * -SearchBase "OU=Firma,DC=firma,DC=local" | Set-ADUser -ChangePasswordAtLogon $true
```



Policy	Security Setting
Enforce password history	0 passwords remembered
Maximum password age	0
Minimum password age	0 days
Minimum password length	10 characters
Password must meet complexity requirements	Enabled
Store passwords using reversible encryption	Disabled

Politika zamykání účtů (Account Lockout Policy) nám určuje jestli, a po kolika chybně zadaných heslech, se *účet zamkne*, jestli se po zadané době *automaticky odemkne* a po jaké době se *resetuje počítadlo* chybných hesel. V Group Policy se nastavení politiky hesel nachází v cestě `Computer Configuration/Windows Settings/Security Settings/Account Policies/Account Lockout Policy`.



Policy	Security Setting
Account lockout duration	0
Account lockout threshold	5 invalid logon attempts
Reset account lockout counter after	30 minutes

Politika pro doménové účty před Windows Server 2008

Pokud máme doménu nižšího stupně než ***Windows Server 2008***, tak můžeme mít pouze ***jednu politiku hesel pro jednu doménu***. To znamená, že pro všechny účty v doméně platí stejná pravidla. Obecně je to dáno tím, že může existovat pouze jedna politika hesel pro každou databázi účtů. Active Directory je jedna databáze, každý počítač pak má jednu svoji lokální databázi pro lokální účty. Takže pro každý počítač může být jiná politika pro lokální účty a pak může být pouze jedna společná politika pro doménu a doménové účty v ní. Obejít se to dá pouze složitě, třeba vytvořením více domén.

Standardně se při instalaci prvního DC vytvoří dvě GPO:

- ***Default Domain Policy GPO*** - připojena na kořen domény, hlavní nastavení pro doménu včetně politiky hesel
- ***Default Domain Controller Policy GPO*** - připojena k OU Domain Controllers, úvodní bezpečnostní nastavení pro DC

Pozn.: *Doporučené řešení je neměnit nastavení těchto dvou GPO, ale vytvořit nové GPO na stejném místě a dát jim vyšší prioritu. Protože se tak můžeme rychle vrátit k originálnímu nastavení v případě problémů (stačí vypnout novou GPO).*

Doménová politika hesel je standardně součástí ***Default Domain Policy***, ale to není naprosto nutné. Můžeme použít libovolnou politiku (vytvořit novou), kde definujeme pravidla politiky hesel pro doménové účty, ale tuto politiku musíme připojit na ***kořenový kontejner domény***. A její hodnoty nesmí přepisovat jiná politika na tomto místě, jinak řečeno nastavit této politice nejvyšší prioritu.

Politika pro lokální účty

Politiku hesel pro lokální účty na počítači můžeme definovat buďto lokálně pomocí ***Local Group Polici Editor*** nebo centrálně doménovou politikou. Když vytvoříme politiku hesel a připojíme ji na kontejner s počítači, tak se toto nastavení neuplatní pro doménové účty, ale pro lokální účty na daných počítačích.

Další politiky pro doménové účty na Windows Server 2008

Pokud máme funkční stupeň domény ***Windows Server 2008***, tak můžeme použít ***Fine-Grained Password Policy (FGPP)***, které nám dovolí definovat více politik hesel pro různé skupiny doménových účtů. Základní politiku hesel (defaultní) ale vytváříme pořád stejně a linkujeme ji na kořen domény.

Pokud chceme vytvořit další politiku hesel a aplikovat ji na určité účty, tak musíme použít FGPP. AD schéma je rozšířeno o dva objekty ***Password Settings Container*** a ***Password Setting***, které se v tomto případě využijí. Nejde tedy použít standardní metody a nástroj *Group Policy Management Console*, ale musíme buď využít ***ADSI Edit*** (adsiedit.msc) nebo ***PowerShell*** a modul Active Directory (mnou upřednostňovaná metoda).

Postup je pak takový, že nejprve vytvoříme ***Password Settings Object*** (PSO) uvnitř *Password Settings Containeru*, kterému nastavíme parametry politiky. Potom PSO ***aplikujeme na uživatele*** nebo bezpečnostní skupinu (global security group). Nemůžeme PSO aplikovat na počítač nebo organizační jednotku (OU).

Abychom si to shrnuli. Microsoft konečně doplnil možnost vytvořit více politik hesel v doméně. Bohužel to ale neudělal standardním způsobem, takže další politiky musíme vytvářet úplně jiným postupem. Navíc změnil i to, že standardní politiky aplikujeme na OU (či domény nebo site), ale PSO se aplikuje na bezpečnostní skupinu nebo uživatele (neříkám, že je to špatné).

Vytvoření PSO pomocí ADSI Edit

Pouze stručná informace, jak můžeme využít ADSI Edit.

- pomocí `adsiedit.msc` se připojíme k doméně (pod účtem s oprávněními k AD - Domain Admin) k ***Default naming context***
- rozbalíme `CN=Password Settings Container,CN=System,DC=firma,DC=local`
- klikneme pravým tlačítkem a z kontextového menu zvolíme ***New - Object***
- vybereme `msDS-PasswordSettings` a ***Next***
- nyní projdeme jednoduchého průvodce, kde vyplníme 11 povinných atributů

MS popis, který obsahuje detaily o jednotlivých atributech je v článku **AD DS Fine-Grained Password and Account Lockout Policy Step-by-Step Guide**.

Přiřazení PSO pomocí ADUC

Vytvořené PSO objekty vidíme i pomocí ***Active Directory Users and Computers*** (ADUC), pokud máme nastaveno v menu **View - Advanced Features**. A právě v ADUC budeme přiřazovat politiky uživatelům nebo skupinám (můžeme to samozřejmě udělat i v ADSI Edit).

- rozbalíme naši doménu, kontejner *System* a *Password Settings Container*
- otevřeme PSO (dvojklik)
- přepneme se na záložku **Attribute Editor**
- najdeme hodnotu `msDS-PSOAppliesTo` a zvolíme ***Edit***
- zde přidáme požadované uživatele nebo skupiny

Pokud chceme zjistit jestli, a případně jaká, PSO se aplikuje na uživatele, tak si otevřeme uživatelský účet v ADUC a přepneme se na záložku ***Attribute Editor***. Pokud nemáme zobrazené konstruované atributy, tak klikneme na ***Filter*** a zaškrtneme ***Constructed***. Potom najdeme atribut `msDS-ResultantPSO`, který obsahuje název aplikované (vítězné) politiky (PSO). Také se můžeme podívat na atribut `msDS-PSOApplied`, kde je seznam všech aplikovaných PSO (tento atribut vidíme i u skupin).

Vytvoření a přiřazení PSO pomocí PowerShellu

Záleží na vkusu, ale jako lepší metoda se mi zdá využití ***PowerShellu***. Potřebujeme k tomu modul ***ActiveDirectory***, který je součástí ***Remote Server Administration Tools***.

Pro vytvoření PSO využijeme cmdlet `New-ADFineGrainedPasswordPolicy`. Jednotlivé parametry jsou pochopitelné již názvu. Je tu však důležitý rozdíl oproti standardní politice a to, že hodnotu `MaxPasswordAge` nemůžeme nastavit na `0`, aby platnost hesla nikdy nevypršela. Pokud v cmdletu nulu zadáme, tak dostaneme chybu, která mi nepřipadá moc informativní.

```
New-ADFineGrainedPasswordPolicy : The modification was not permitted for security reasons
```

Řešením je buď na účtech nastavit ***Password never expires*** nebo nastavit `MaxPasswordAge` na maximální hodnotu `10675199.00:00:00`, která je opravdu vysoká. Následují dva příklady vytvoření PSO.

```
New-ADFineGrainedPasswordPolicy -Name "JmenoPSO" -DisplayName "Domain Users PSO" -Description "The Domain Users Password Poli
New-ADFineGrainedPasswordPolicy -Name "PSO2" -Precedence 10 -LockoutDuration "0.12:00:00" -LockoutObservationWindow "0.00:15:
```

Ještě jsem narazil na problém, že někdy mi PowerShell nefungoval na doménovém řadiči a dostával jsem chybu.

```
New-ADFineGrainedPasswordPolicy : Access is denied
```

Pomocí PowerShellu můžeme také přiřadit PSO k uživateli/skupině jednoduchým příkazem.

```
Add-ADFineGrainedPasswordPolicySubject JmenoPSO -Subjects 'Domain Users'
```

Změna hodnot v PSO.

```
Set-ADFineGrainedPasswordPolicy "JmenoPSO" -MinPasswordLength 8 -PasswordHistoryCount 24
```

Smazání PSO.

```
Remove-ADFineGrainedPasswordPolicy -Identity JmenoPSO
```

Zrušení přiřazení PSO k uživateli/skupině.

```
Remove-ADFineGrainedPasswordPolicySubject JmenoPSO -Subjects User1
```

Vypsání seznamu PSO.

```
Get-ADFineGrainedPasswordPolicy -Filter 'Name -like "*"' | FT Name,Precedence,MinPasswordLength -AutoSize
```

Zobrazení detailů k jedné PSO.

```
Get-ADFineGrainedPasswordPolicy JmenoPSO -Properties *
```

Jaká PSO se aplikuje na určitého uživatele.

```
Get-ADUserResultantPasswordPolicy User1
```

Zajímavý je i následující příkaz, který zobrazí defaultní politiku hesel pro aktuálního (přihlášeného) uživatele.

```
Get-ADDefaultDomainPasswordPolicy
```


Autor: **Petr Bouška**

Související články:

Group Policy

Základem centrální správy počítačů v doméně jsou určitě skupinové politiky (Group Policy). Pomocí nich můžeme řídit nastavení, bezpečnost a chování pracovních stanic a serverů.

- **Nastavení přihlašování do domény pomocí Smart Card** [12.07.2006 16:04]
- **Group Policy Preferences, GPO, GPMC, GPME** [16.03.2009 18:09]
- **Group Policy - řízení aplikace politik** [12.12.2010 17:42]
- **Group Policy - Administrative Templates** [19.01.2011 18:16]
- **Group Policy - Politiky hesel a zamykání účtů** [21.02.2011 18:46] ..... **právě čtete**
- **MS Outlook a konfigurace pomocí Group Policy** [23.03.2011 16:30]
- **Auditování AD DS objektů ve Windows Server 2008** [26.03.2011 19:23]
- **Microsoft Certification Authority Auditing** [13.01.2014 17:05]

MS Outlook a konfigurace pomocí Group Policy

Středa, 23.03.2011 16:30 | Samuraj - Petr Bouška | Microsoft admin

Článek rozebírá některé možnosti, jak centrálně konfigurovat aplikace z balíku MS Office se zvláštním zaměřením na Outlook (primárně se jedná o verzi 2010, ale většinou to není podstatné). Začít můžeme modifikací instalace (zde pouze zmínka), hlavní část provedeme pomocí Group Policy Administrative Templates pro Office, některé požadavky vyřešíme změnou registrů a pro složitější příklad z praxe využijeme PowerShell.

Office Customization Tool

Pro základní nastavení balíku MS Office je vhodný nástroj ***Office Customization Tool***, který je součástí instalace a spustíme jej voláním `setup.exe /admin`. Pomocí něj vytvoříme konfigurační soubor ***Setup customization (.msp)***, který se použije při instalaci. Tomu se zde nevěnujeme, takže pouze odkaz na informace MS Office Customization Tool in Office 2010.

Administrative Templates pro MS Office

Pokud chceme použít **Group Policy** pro konfiguraci aplikací z balíku ***MS Office***, tak musíme stáhnout ***šablony*** (Administrative Templates) pro odpovídající verzi. Popis a odkaz ke stažení, pro verzi Office 2010, je na adrese Office 2010 Administrative Template files (ADM, ADMX, ADML) and Office Customization Tool. Stáhneme samorozbalovací archiv, například `AdminTemplates_64.exe`, který rozbalíme někde na disk. Součástí jsou jak starší šablony **ADM**, tak novější **ADMX** (pro Windows Server 2008 a Windows 7).

> ***Pozn.:*** *Šablony slouží pouze k vytvoření politik, takže pokud máme administrátorskou stanici s Windows 7 (nebo třeba DC s Server 2008), můžeme použít ADMX šablony, i když v síti máme počítače s Windows XP.*

ADMX šablony zkopírujeme do ***Central Store*** v doméně `\\FQDN\SYSVOL\FQDN\policies\PolicyDefinitions` (FQDN je plné jméno domény), pokud jej používáme. Nebo lokálně do cesty `c:\Windows\PolicyDefinitions`. Pokud musíme použít ***ADM***, tak šablony načteme při vytváření politiky.

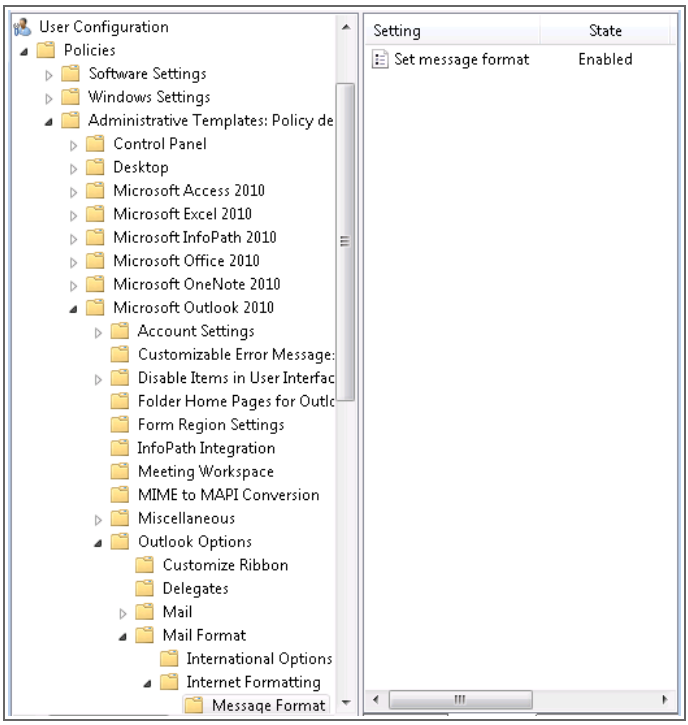
Popis celé problematiky od MS je v článku Enforce settings by using Group Policy in Office 2010. Seznam vlastností, které můžeme řídit pro Outlook 2010, je v článku Determine which features to enable or customize in Outlook 2010.

Politiky pro Outlook pomocí šablony

Pomocí ***Group Policy*** můžeme řídit řadu vlastností aplikace ***Oulook***, ale rozhodně ne všechny. To, co nemůžeme nastavit, často MS odpovídá, že bychom nastavovat neměli, ale praxe si žádá opak. Například můžeme nastavit výchozí formát emailových zpráv, třeba na HTML, ale nemůžeme definovat font. Navíc nezakážeme, aby si uživatelé změnili parametry v nové zprávě.

Nastavení výchozího formátu emailu

Když chceme vytvořit ***politiku***, která nastaví ***výchozí formát emailu na HTML***, a nedovolí uživatelům tento výchozí formát změnit (jak je pro politiky běžné), tak standardním způsobem vytvoříme novou politiku. Nastavení můžeme provádět pouze pro uživatele. Doklikáme se na položku `User Configuration/Policies/Administrative Templates/Microsoft Outlook 2010/Outlook Options/Mail Format/Internet Formatting/Message Format`. Zde je jediná položka `Set message format`, kterou zapneme a nastavíme na `HTML`. Politiku pak přilinkujeme na kontejner s uživateli, kterým chceme nastavení provést.



Politiky pro Outlook pomocí registrů

Hodnoty, které nemůžeme nastavit pomocí šablony, můžeme pořád ještě nastavit přímým zápisem do registrů (šablona nedělá téměř nic jiného). V tomto případě zůstane uživatelům možnost, danou hodnotu změnit, ale při dalším aplikování politiky se opět přepíše.

Nastavení výchozího fontu pro email

Takže například, pokud chceme nastavit ***výchozí font pro emaily***, můžeme použít nastavení ***registrů***. Nejlepší je nastavit na počítači požadované hodnoty a pak se podívat do registrů, kde je toto nastavení uložené `HKEY_CURRENT_USER\Software\Microsoft\Office\14.0\Common\MailSettings`. Potom vytvoříme politiku (preferences) a v cestě `User Configuration/Preferences/Windows Settings/Registry` zadáme `Update` jednotlivých klíčů v registrech. Vybereme ***Hive*** `HKEY_CURRENT_USER`, ***Key Path*** `Software\Microsoft\Office\14.0\Common\MailSettings` a ***Value name*** jednotlivé položky. Hodnoty fontů zadáváme jako `REG_BINARY`, musí být ve správném formátu (například provedeme export z registrů a z hodnot odstraníme čárky).

Name	C	Act...	Hive	Key	Value Na...	Type	Value Data
Template	1	Up...	HKEY...	Softwar...	Template	REG_SZ	
NewTheme	2	Up...	HKEY...	Softwar...	NewThe...	REG_BINARY	
ComposeFontSimple	3	Up...	HKEY...	Softwar...	Compos...	REG_BINARY	3c0000001f0000f80000...
ReplyFontSimple	4	Up...	HKEY...	Softwar...	ReplyFon...	REG_BINARY	3c0000001f0000f80000...
TextFontSimple	5	Up...	HKEY...	Softwar...	TextFont...	REG_BINARY	3c0000001f0000f80000...
ComposeFontComplex	6	Up...	HKEY...	Softwar...	Compos...	REG_BINARY	3c68746d6c3e0d0a0d0...
ReplyFontComplex	7	Up...	HKEY...	Softwar...	ReplyFon...	REG_BINARY	3c68746d6c3e0d0a0d0...
TextFontComplex	8	Up...	HKEY...	Softwar...	TextFont...	REG_BINARY	3c68746d6c3e0d0a0d0...

Outlook, Group Policy a Powershell

Pořád máme některá nastavení, která se nám výše uvedenými postupy nepovede nastavit.

Nastavení sdílení kalendáře

Jedním běžným příkladem je ***nastavení sdílení kalendáře*** pro celou firmu. Uživatel si to sice nastavuje v Outlooku, ale ve skutečnosti jde o nastavení na *Exchange serveru* a účtu uživatele. Na *Exchange Server 2010 již* máme *PowerShell* příkaz pro toto nastavení (`Set-MailboxFolderPermission`), ale na *Exchange Server 2007* ne, takže můžeme použít ***PFDavAdmin*** pro ruční nastavení. Jediný automatizovaný způsob, který jsem nalezl je pomocí **Exchange Web Services a PowerShell**.

Generování podpisů do Outlooku

Jinou častou položkou, kterou bychom mohli chtít automatizovaně konfigurovat, je ***podpis uživatele*** v Outlooku, tak aby odpovídal firemním předpisům. To je také značně složitý úkol, jediné docela funkční řešení, které jsem nalezl, je pomocí **PowerShellu**, který spouštíme pod uživatelem. Ukázkový skript nalezneme na internetu na několika místech, například **Outlook signature based on user information from Active Directory**. Tento skript je dobrý začátek, já jej z 50% upravil a je docela funkční.

Podpisy v Outlooku jsou běžné soubory ve formát **RTF**, **HTM** a **TXT**, které jsou uložené v cestě (na Windows 7) `c:\Users\{username}\AppData\Roaming\Microsoft\Signatures`. Skript dělá to, že do počítače zkopíruje připravenou šablonu ve Wordu, pak využije *COM objekt* knihovny Wordu (`Microsoft.Office.Interop.Word.dll`) a pomocí jeho funkcí nahradí některá slova údaji, které načte z ***Active Directory*** (musíme tedy mít potřebné hodnoty u jednotlivých uživatelských účtů). Potom soubor uloží do požadovaných formátů a nastaví výchozí podpis.

> ***Pozn.:*** *Na některých počítačích jsem narazil na problém, že se automaticky nenačetla knihovna a musel jsem ji nejprve načíst [reflection.assembly]::LoadFile. Někde se objevil problém s funkcí $MSWord.ActiveDocument.SaveAs, kde se standardně parametry předávají jako [ref] a já je musel předat přímo.*

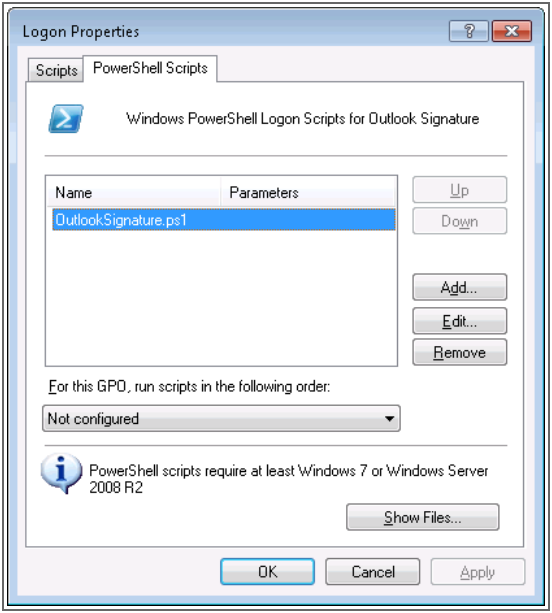
Volání tohoto ***PowerShell skriptu*** můžeme provádět pomocí ***Group Policy*** jako ***Logon*** nebo ***Logoff skript***, cesta

v Group Policy je `User Configuration/Policies/Windows Settings/Scripts/Logon`. Skript můžeme nakopírovat přímo do adresáře GPO. Pro Windows 7 můžeme využít rovnou novou záložku ***PowerShell Scripts***. Pro starší systémy musíme využít ***standardní skript***, vytvoříme **cmd** soubor a do něj zadáme volání *powershellu*. Pokud máme PowerShell skript v adresáři politiky, tak volání vypadá nějak takto:

```
powershell ". '\\firma.local\SYSVOL\firma.local\Policies\{F9815FA8-A285-4995-81AA-3432BB5A5C16}\User\Scripts\Logon\OutlookSig
```

> ***Pozn.:*** *Na stanicích musíme mít povolené spouštění PowerShell skriptů.*

Aby se skript zbytečně nespouštěl stále znovu (a nebrzdil nám start, i když běží na pozadí), tak se na začátku skriptu zapisuje hodnota (datum) do registrů a porovnává se, jestli nedošlo ke změně souboru. Pokud nedošlo, tak se provádění skriptu ukončí.



Auditování AD DS objektů ve Windows Server 2008

Sobota, 26.03.2011 19:23 | Samuraj - Petr Bouška | Microsoft admin

Pokud chceme logovat operace (zaznamenávat události do logu) nad Active Directory Domain Services (AD DS) - prakticky jde o vytváření, měnění či mazání uživatelských a počítačových účtů a skupin, tak využijeme funkci auditování AD DS. Tento audit byl možný již dříve a to obdobným způsobem, ale Windows Server 2008 přináší rozšíření a zpřesnění. Nově můžeme auditovat pouze určitou podkategorii (vylepšené nastavení nabízí Windows Server 2008 R2), u změn se loguje i původní a nová hodnota (ne jen kdo a jaký atribut změnil) a také se změnily ID událostí.

Tento článek vyšel také na Microsoft TechNet Blogu CZ/SK pod názvem Auditování AD DS objektů ve Windows Server 2008 a v Microsoft sekci na Živě Auditování AD DS objektů ve Windows Server 2008.

Situace okolo auditování událostí nad AD DS není jednoduchá. Musíme si dobře rozmyslet, co chceme sledovat. Jestli nám jde o podezřelé operace, failed události nebo dohled nad vytvářením a mazáním účtů. Pomocí ***Directory Service Access*** (DS Access) a detailního nastavení SACL, můžeme sledovat tisíce údajů (a špatným nastavením si pouze zaplníme log). Hlavní zde tedy je dobře pochopit a nastavit filtrování pomocí SACL.

Navíc použití ***DS Access*** není jediná možnost, i když je Microsoftem všude prezentovaná. Máme ještě kategorii ***Account Management***. Tyto události můžeme použít na sledování změn lokálních účtů na stanici, ale stejně tak na doménovém řadiči pro doménové účty. Tato kategorie nám nenabízí tolik možností, ale její nastavení je jednodušší. A někdy doplňuje informace, které získáme z DS Access. Navíc myslím, že řadě lidí může tato kategorie stačit a vůbec nemusí použít pro logování DS Access.

Auditování kategorie DS Access

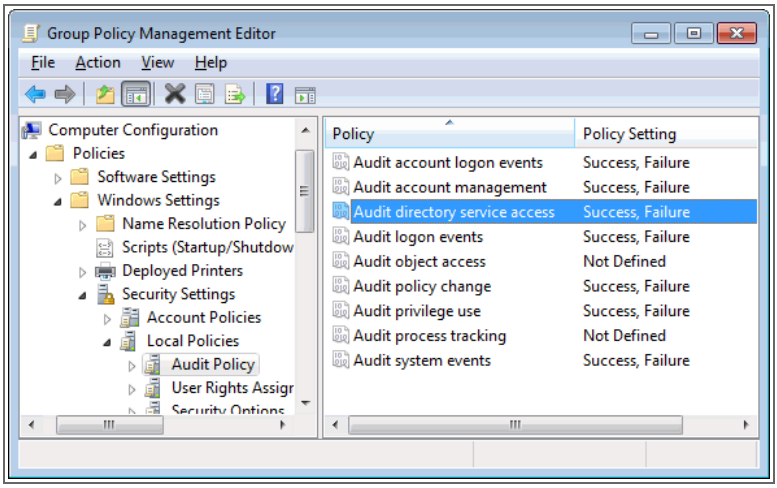
Abychom zapnuli auditování, tak musíme provést několik kroků:

1. **zapneme auditování** - máme dvě možnosti
   - ***globálně*** pomocí auditovací politiky
   - pro jednotlivé ***podkategorie*** - podpora od *Windows Server 2008*, opět dvě možnosti
     - pomocí příkazu **auditpol.exe**
     - pomocí nové **rozšířené auditovací politiky** - podpora od *Windows Server 2008 R2*
2. **vybereme detailně**, jaké operace pro jaké objekty chceme sledovat - pomocí **SACL**
3. (volitelně) **nastavíme výjimky** pro atributy - úpravou ***AD schema*** - provede se nastavením atributu **searchFlags** na hodnotu 256 u objektu, který nechceme logovat, této oblasti se zde nebudeme věnovat

Globální zapnutí auditování

Na DC aplikujeme politiku, která globálně zapíná auditování. Toto nastavení můžeme provést například v ***Default Domain Controllers Policy***.

- spustíme **Group Policy Management**
- otevřeme politiku, třeba **Default Domain Controllers Policy**, v **Group Policy Management Editor**
- proklikáme se do **Computer Configuration/Policies/Windows Settings/Security Settings/Local Policies/Audit Policy**
- otevřeme položku **Audit directory service access**
- zaškrtneme **Define these policy settings** a zvolíme, jestli chceme sledovat úspěšné a neúspěšné událost - **Success** a **Failed**



Zapnutí auditování pro podkategorie

Od Windows Server 2008 můžeme vybrat subkategorii, kterou chceme auditovat. Není to povinné, pokud nastavíme politiku z předchozího bodu, tak se nám zapnout všechny podkategorie k dané kategorii. Standardně (bez politiky) je na serveru zapnuto pouze auditování ***Directory Service Access*** na ***Success***. Jednotlivé podkategorie jsou:

- Directory Service Access
- Directory Service Changes

- Directory Service Replication
- Detailed Directory Service Replication

Nastavení pomocí auditpol.exe

Pokud nemáme ***Windows Server 2008 R2***, tak pro nastavení jednotlivých podkategorií nemůžeme použít Group Policy, ale musí nám stačit řádkový příkaz **auditpol.exe**. Konfiguraci musíme provádět na jednotlivých doménových řadičích a postupně na všech, kde chceme, aby se uplatnilo.

Můžeme si vypsat jednotlivé podkategorie (všechny nebo z nějaké kategorie)

```
Auditpol /list /subcategory:*
Auditpol /list /subcategory:"DS Access"
```

Můžeme provést zálohu nastavení

```
Auditpol /backup /file:C:\auditpolicy.csv
```

Můžeme zjistit aktuální nastavení (všech hodnot nebo z naší kategorie)

```
Auditpol /get /category:*
Auditpol /get /category:"DS Access"
```

A pak můžeme měnit nastavení (povolovat nebo zakazovat jednotlivé podkategorie)

```
Auditpol /set /subcategory:"directory service changes" /success:enable
Auditpol /set /subcategory:"Detailed Directory Service Replication" /success:disable /failure:disable
```

Většinu příkazů musíme spouštět na serveru a konfiguraci provádět na DC. Pokud se o něco pokusíme na stanici nebo ne pod administrátorským oprávněním (Run as administrator), tak dostaneme chybovou hlášku:

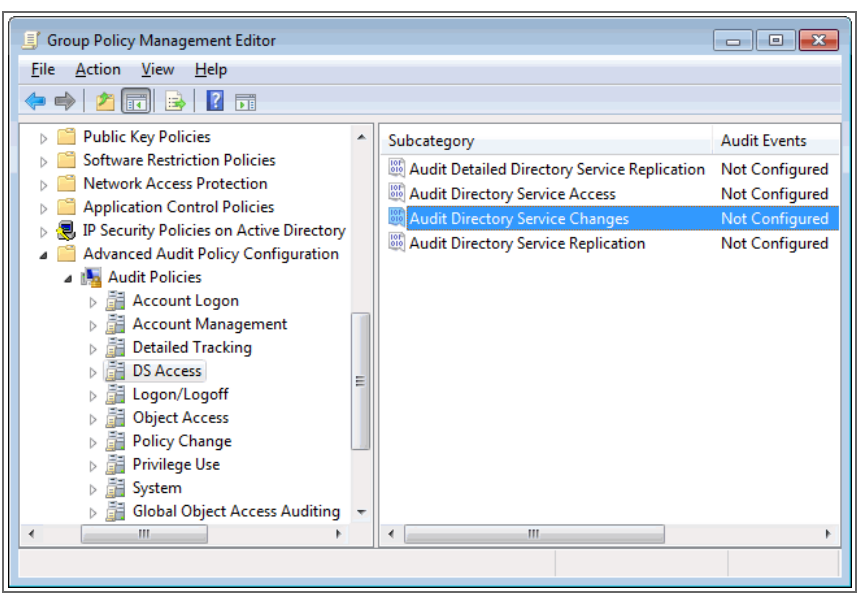
```
Auditpol /get /category:*
Error 0x00000522 occurred:
A required privilege is not held by the client.
```

```
C:\>Auditpol /get /category:"DS Access"
System audit policy
Category/Subcategory                      Setting
DS Access
  Directory Service Changes               Success and Failure
  Directory Service Replication           Success and Failure
  Detailed Directory Service Replication  No Auditing
  Directory Service Access                Success and Failure
```

Nastavení pomocí Group Policy

Windows Server 2008 R2 (a Windows 7) byl rozšířen o možnost konfigurovat audtitování na úrovni podkategorií pomocí Group Policy. Tyto politiky se nachází na trochu jiném místě ve skupině **Advanced Audit Policy Configuration**.

- spustíme **Group Policy Management**
- otevřeme politiku, třeba **Default Domain Controllers Policy**, v **Group Policy Management Editor**
- proklikáme se do **Computer Configuration/Policies/Windows Settings/Security Settings/Advanced Audit Policy Configuration/Audit Policy/DS Access**
- otevřeme položku **Audit directory service access**
- vybereme jednotlivé podkategorie a u nich nastavíme **Success** či **Failed**



Když nastavíme nějakou hodnotu z **Advanced Audit Policy Configuration**, tak se automaticky vypnout všechny hodnoty z **Local Policies/Audit Policy**. Přesto se doporučuje ještě nastavit politiku, která toto zařizuje:

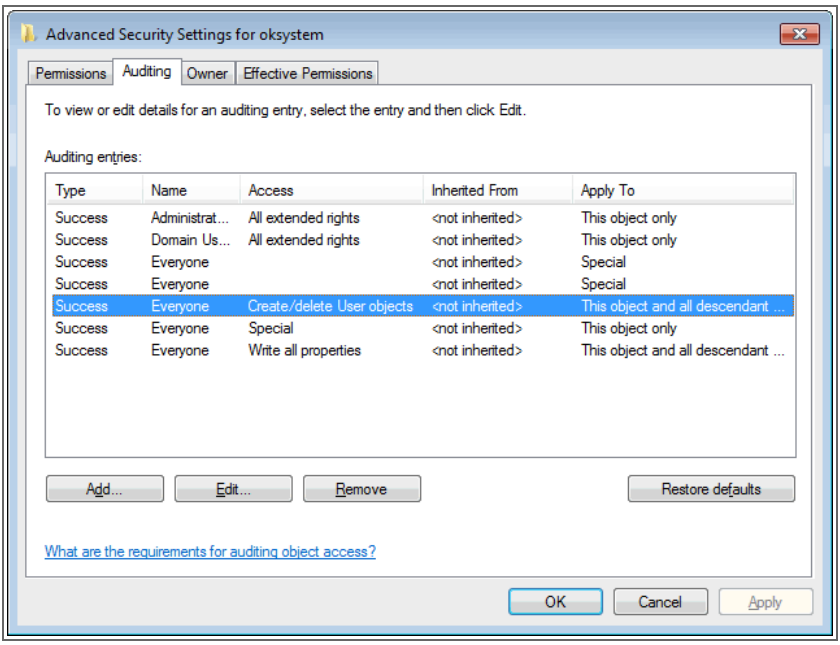
- proklikáme se do **Computer Configuration/Policies/Windows Settings/Security Settings/ Local Policies/Security Options**
- otevřeme položku **Audit: Force audit policy subcategory settings (Windows Vista or later) to override audit policy category settings**
- a zapneme - **Enabled**

Určení, jaké objekty chceme sledovat - SACL

Nastavení oprávnění ***System Access Control List*** - **SACL**, umožňuje logovat přístupy k zabezpečeným objektům,

určuje jaká operace a kým provedená se loguje. Dokud nenastavíme SACL odpovídajícím způsobem, tak se nám nebudou vytvářet žádné záznamy!

- pomocí **ADUC** (Active Directory Users and Computers) vybereme objekt (nejčastěji OU nebo celou doménu)
- klineme pravým tlačítkem a zvolíme **Properties**
- přepneme se na záložku **Security**
- klikneme na tlačítko **Advanced**
- přepneme se na záložku **Auditing**



Zde vidíme již nastavené SACL. Přidání nového:

- klikneme na tlačítko **Add**
- vybereme, pro koho chceme logování provádět, standardně **Authenticated Users** nebo **Everyone**
- na záložce **Object** můžeme nastavovat operace, které chceme sledovat, například **Create User objects** a zatrháváme, jestli chceme sledovat **Failed** nebo **Successful**
- na záložce **Properties** můžeme specifikovat přímo operaci Read/Write
- na obou záložkách můžeme určit, na jaké objekty chceme aplikovat (pak se zobrazí pouze možné nabídky), v **Apply onto**, například **Descendant User objects**

Jakou operaci a na co aplikovanou (Apply onto) musíme dobře zvážit. Například vytvoření nového objektu (uživatele) musíme sledovat na nadřazeném objektu (tedy asi OU). Změnu nějaké hodnoty musíme sledovat na daném objektu jako write property.

Události z kategorie DS Access a jejich ID

Audit generuje události do **Security logu**. Následuje seznam událostí v jednotlivých podkategoriích, oficiálně tento seznam naleznete na webu Microsoftu **DS Access**.

Audit Directory Service Changes

- **Event ID 5136** - A directory service object was modified.
- **Event ID 5137** - A directory service object was created.
- **Event ID 5138** - A directory service object was undeleted.
- **Event ID 5139** - A directory service object was moved.
- **Event ID 5141** - A directory service object was deleted.

Audit Directory Service Access

- **Event ID 4662** - An operation was performed on an object.

Audit Directory Service Replication

- **Event ID 4932** - Synchronization of a replica of an Active Directory naming context has begun.
- **Event ID 4933** - Synchronization of a replica of an Active Directory naming context has ended.

Audit Detailed Directory Service Replication

- **Event ID 928** - An Active Directory replica source naming context was established.
- **Event ID 4929** - An Active Directory replica source naming context was removed.
- **Event ID 4930** - An Active Directory replica source naming context was modified.
- **Event ID 4931** - An Active Directory replica destination naming context was modified.
- **Event ID 4934** - Attributes of an Active Directory object were replicated.
- **Event ID 4935** - Replication failure begins.
- **Event ID 4936** - Replication failure ends.
- **Event ID 4937** - A lingering object was removed from a replica.

Auditování kategorie Account Management

Nastavení auditování kategorie ***Account Management*** je mnohem jednodušší. Spočívá pouze v zapnutí auditování dané kategorie. A stejně jako u ***DS Access*** to můžeme provést dvěma způsoby:

- ***globálně*** pomocí auditovací politiky
- pro jednotlivé ***podkategorie*** - podpora od *Windows Server 2008*, opět dvě možnosti
  - pomocí příkazu **auditpol.exe**
  - pomocí nové **rozšířené auditovací politiky** - podpora od *Windows Server 2008 R2*

Globální zapnutí auditování

Auditování povolíme stejně jako u kategorie DC Access, to znamená nejlépe přes Group Policy. Pouze tentokrát vybereme položku **Audit account management.**

Zapnutí auditování pro podkategorie

Opět můžeme vybrat pouze některé podkategorie pomocí příkazu příkaz **auditpol.exe** nebo pomocí politiky **Advanced Audit Policy Configuration**. Kategorie ***Account Management*** obsahuje podkategorie:

- Audit Application Group Management
- Audit Computer Account Management
- Audit Distribution Group Management
- Audit Other Account Management Events
- Audit Security Group Management
- Audit User Account Management

Události z kategorie Account Management a jejich ID

Audit generuje události opět do **Security logu.** Následuje seznam událostí v jednotlivých podkategoriích, oficiálně tento seznam naleznete na webu Microsoftu Account Management.

Audit User Account Management

- **Event ID 4720** - A user account was created
- **Event ID 4722** - A user account was enabled
- **Event ID 4723** - An attempt was made to change an account's password
- **Event ID 4724** - An attempt was made to reset an account's password
- **Event ID 4725** - A user account was disabled
- **Event ID 4726** - A user account was deleted
- **Event ID 4738** - A user account was changed
- **Event ID 4740** - A user account was locked out
- **Event ID 4765** - SID History was added to an account
- **Event ID 4766** - An attempt to add SID History to an account failed
- **Event ID 4767** - A user account was unlocked
- **Event ID 4780** - The ACL was set on accounts which are members of administrators groups
- **Event ID 4781** - The name of an account was changed
- **Event ID 4794** - An attempt was made to set the Directory Services Restore Mode
- **Event ID 5376** - Credential Manager credentials were backed up
- **Event ID 5377** - Credential Manager credentials were restored from a backup

Audit Computer Account Management

- **Event ID 4741** - A computer account was created
- **Event ID 4742** - A computer account was changed
- **Event ID 4743** - A computer account was deleted

Audit Application Group Management

- **Event ID 4783** - A basic application group was created
- **Event ID 4784** - A basic application group was changed
- **Event ID 4785** - A member was added to a basic application group
- **Event ID 4786** - A member was removed from a basic application group
- **Event ID 4787** - A non-member was added to a basic application group
- **Event ID 4788** - A non-member was removed from a basic application group
- **Event ID 4789** - A basic application group was deleted
- **Event ID 4790** - An LDAP query group was created

Audit Distribution Group Management

- **Event ID 4744** - A security-disabled local group was created
- **Event ID 4745** - A security-disabled local group was changed
- **Event ID 4746** - A member was added to a security-disabled local group
- **Event ID 4747** - A member was removed from a security-disabled local group
- **Event ID 4748** - A security-disabled local group was deleted
- **Event ID 4749** - A security-disabled global group was created
- **Event ID 4750** - A security-disabled global group was changed
- **Event ID 4751** - A member was added to a security-disabled global group
- **Event ID 4752** - A member was removed from a security-disabled global group
- **Event ID 4753** - A security-disabled global group was deleted
- **Event ID 4759** - A security-disabled universal group was created
- **Event ID 4760** - A security-disabled universal group was changed
- **Event ID 4761** - A member was added to a security-disabled universal group
- **Event ID 4762** - A member was removed from a security-disabled universal group

Audit Other Account Management Events

- **Event ID 4782** - The password hash for an account was accessed
- **Event ID 4793** - The Password Policy Checking API was called

Audit Security Group Management

- **Event ID 4727** - A security-enabled global group was created
- **Event ID 4728** - A member was added to a security-enabled global group
- **Event ID 4729** - A member was removed from a security-enabled global group
- **Event ID 4730** - A security-enabled global group was deleted
- **Event ID 4731** - A security-enabled local group was created
- **Event ID 4732** - A member was added to a security-enabled local group
- **Event ID 4733** - A member was removed from a security-enabled local group
- **Event ID 4734** - A security-enabled local group was deleted
- **Event ID 4735** - A security-enabled local group was changed
- **Event ID 4737** - A security-enabled global group was changed
- **Event ID 4754** - A security-enabled universal group was created
- **Event ID 4755** - A security-enabled universal group was changed
- **Event ID 4756** - A member was added to a security-enabled universal group
- **Event ID 4757** - A member was removed from a security-enabled universal group
- **Event ID 4758** - A security-enabled universal group was deleted
- **Event ID 4764** - A group's type was changed

Zprávy v Event Logu

Pro nalezení záznamů, které nás zajímají, z velkého množství logů v ***Security logu***, můžeme použít filtrování.

- otevřeme ***Event Viewer***
- doklikáme se na **Security**
- klikneme pravým tlačítkem a zvolíme ***Filter Current log***
- v kolonce ***Event sources*** vybereme **Microsoft Windows security audting**.
- do kolonky ***Task category*** vybereme všechny kategorie, které nás zajímají, třeba:
  - Directory Service Changes
  - Directory Service Access
  - User Account Management
  - Computer Account Management

Obecně se samozřejmě počítá s tím, že si události posíláme na nějaký server, který je dále zpracovává. Může se jednat o standardní ***Syslog server*** (do Windows ale musíme doplnit Syslog klienta, aby se logy odesílaly) nebo třeba Microsoft ***SCOM*** (System Center Operations Manager).

Příklad zpráv

Malá ukázka, co se zaloguje, pokud je zapnuto ***DS Access*** i ***Account Management*** pro události ***Success*** i ***Failed***. **SACL** je nastaveno na celou doménu pro uživatele ***Everyone*** přístup ***Create/Delete User objects***. Zajímavé je, že není zalogována událost 5141.

Vytvoření uživatele

12:46:10 4662 An operation was performed on an object. / Object Access Create Child
12:46:10 4720 A user account was created.
12:46:10 4724 An attempt was made to reset an account's password.
12:46:10 4738 A user account was changed.
12:46:10 4722 A user account was enabled.
12:46:19 5137 A directory service object was created.

Smazání uživatele

13:01:52 4726 A user account was deleted.

Odkazy

- AD DS Auditing Step-by-Step Guide
- Auditpol
- Windows Server 2008: Auditing Active Directory

Microsoft Certification Authority Auditing

Pondělí, 13.01.2014 17:05 | Samuraj - Petr Bouška | Microsoft admin

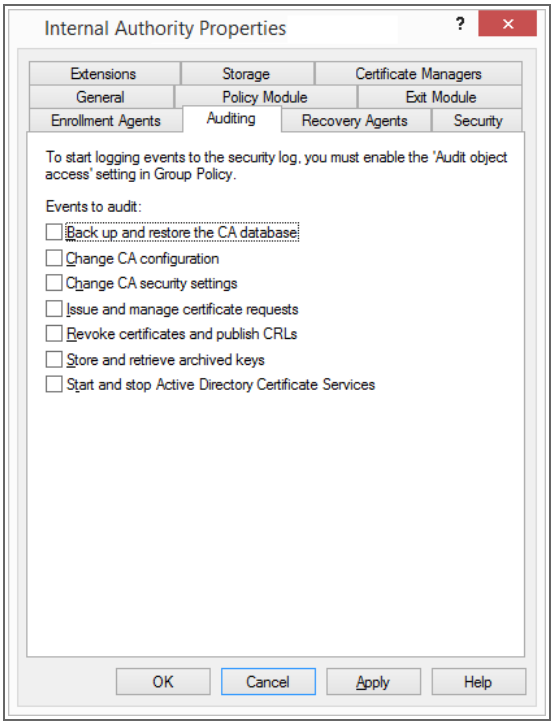
Když provozujeme certifikační autoritu od MS, což se hodí skoro pro každou firmu, tak je dobré nastavit určité logování operací. Do té doby můžeme sledovat informace, třeba o selhání vystavení certifikátu, pouze v konzoli certifikační autority. Článek je pouze pro přehled, stejné informace nalezneme přímo u MS.

Logování událostí CA

Pokud zapneme auditování na ***Certification Authority*** (CA), tak se nám budou události zapisovat do standardního Windows ***Event Logu Security***. Zapnutí logování se provádí ve dvou krocích. Nejprve zvolíme, jaké události chceme sledovat, poté musíme zapnout vlastní logování.

Nastavení auditování událostí CA

- otevřeme ***Certification Authority snap-in*** pro MMC
- klikneme pravým tlačítkem na název naší CA a zvolíme ***Properties***
- přepneme se na záložku ***Auditing***
- zde zvolíme události, které chceme zaznamenávat



Následně je třeba ***restartovat*** CA, třeba opět v kontextovém menu autority ***All Tasks - Stop Service*** a následně ***Start Service***.

Zapnutí auditování

I když jsme výše nastavili události, které chceme zaznamenávat, tak logování nezačne. Musíme jej totiž nastavit pomocí auditovací politiky, kterou konfigurujeme v ***lokální*** nebo ***doménové Group Policy***. Ukážeme si příklad na ***Domain Group Policy***, kde musíme zvolit nějakou GPO, která je aplikována na server, kde běží CA.

- spustíme ***Group Policy Management***
- vytvoříme novou, nebo otevřeme existující, politiku v ***Group Policy Management Editor***
- proklikáme se do ***Computer Configuration/Policies/Windows Settings/Security Settings/Local Policies/Audit Policy***, kde zapneme **Audit object access**
- nebo lépe, na novějších Windows (od Windows Server 2008 R2), se proklikáme do ***Computer Configuration/Policies/Windows Settings/Security Settings/Advanced Audit Policy Configuration/Audit Policy/Object Access***, zde nastavíme **Audit Certification Services** (většinou ***Success*** i ***Failure***)

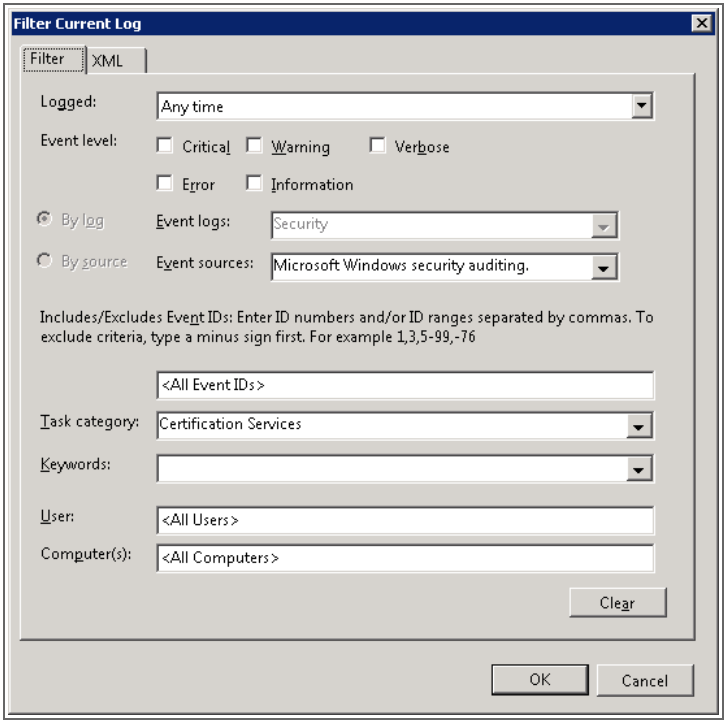
Nyní je třeba počkat nebo ručně vynutit aplikování politiky a události na CA by se měli začít logovat.

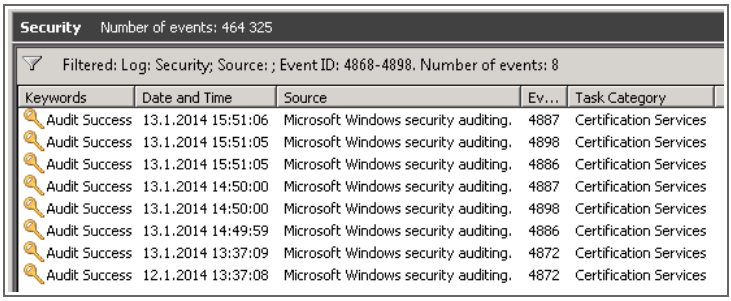
Zobrazení událostí

Události se ukládají do ***Security*** logu, který většinou obsahuje velké množství různých událostí. Proto si musíme umět vybrat, co nás zajímá. Seznam ***Event ID***, pro autoritu běžící na Windows Server 2008 R2, je v článku **Audit Certification Services**. Jde o rozsah ***Event ID*** od `4868` do `4898`, například ID `4887` je vystavení certifikátu a ID `4888` odmítnutí vystavení certifikátu.

Jednoduchým řešením je ***nastavení filtru***. Klikneme pravým tlačítkem na ***Security log*** a zvolíme ***Filter Current Log***. Můžeme nastavit Event ID, která nás zajímají (třeba celý rozsah 4868 - 4898) nebo zvolit ***Event sources*** na ***Microsoft Windows security auditing***, a k tomu ***Task category*** na ***Certification Services***. Pak se nám zobrazí

všechny události, které se týkají CA.





Oprávnění na CA

Malá poznámka z jiné oblasti. ***Certifikační autorita*** je brána jako důležitý bezpečnostní prvek. Oprávnění můžeme řídit mnoha způsoby, kde máme řadu rolí jako *Certificate Manager*, *CA administrator*, *Enrollment Agent* a další. Můžeme využít ***Role-Based Administration***.

Hlavní oprávnění se nastavují jednoduše přímo na autoritě. Pokud třeba chceme dát nějakému uživateli oprávnění zobrazit všechny parametry CA (bez toho, že by mohl něco měnit), tak to můžeme provést následně. Pro přístup pak uživatel využije ***Certification Authority snap-in***, který si může nainstalovat na stanici z ***Remote Server Administration Tools*** (RSAT).

- otevřeme ***Certification Authority snap-in*** pro MMC
- klikneme pravým tlačítkem na název naší CA a zvolíme ***Properties***
- přepneme se na záložku ***Security***
- zde přidáme uživatele (nebo lépe skupinu) a nastavíme oprávnění ***Read***

PowerShell - Active Directory

Neděle, 14.11.2010 13:13 | Samuraj - Petr Bouška | Microsoft admin

Tak už i na mne dolehl PowerShell, ne že bych se mu bránil, ale hodně věcí udělám radši v GUI. Pro různé hromadné úpravy či výpisy určitých hodnot, se jedná o velice užitečného pomocníka. Myslím ale, že PowerShell není vytvořen zrovna User Friendly, takže pokud něco nepoužíváme pravidelně, není možné si to zapamatovat (neobsahuje takovou jednoduchou a chytrou nápovědu jako Cisco IOS). Proto jsem začal vytvářen tento seznam užitečných příkazů (a jejich variací), abych měl v případě potřeby, kam nahlédnout.

Na hodně věcí můžeme použít ***Active Directory Users and Computers*** (ADUC), a v něm třeba Queries, nebo nový ***Active Directory Administrative Center***. **PowerShell** a ***modul ActiveDirectory*** toho však umí více, můžeme některé operace skriptovat, a také dostaneme výsledky, s kterými můžeme dále pracovat (třeba v Excelu). Pokud používám PowerShell přímo, tak nejraději z ***PowerShell ISE***, které je součástí Windows 7.

Abychom mohli použít ActiveDirectory modul na stanici, tak musíme mít nainstalovaný ***Active Directory Domain Services*** (AD DS and AD LDS Tools, což je součást RSAT - Remote Server Administration Tools) a v něm povolenu vlastnost ***Active Directory Module for Windows PowerShell***. Také musíme být alespoň na jednom doménovém řadiči nainstalovaná služba ***Windows Server 2008 R2 Active Directory Web Services***.

Obecné vlastnosti

Použití AD příkazů

Pokud použijeme ***PowerShell ISE***, tak po každém spuštění musíme nejprve importovat modul ActiveDirectory, abychom měli k dispozici příkazy pro Active Directory.

```
Import-Module ActiveDirectory
```

Procházení Active Directory

AD můžeme procházet stejně, jako souborový systém. Nejprve se přepneme do AD a pak používáme příkazy `dir`, `cd` apod.

```
PS C:\> cd ad:
PS AD:\> dir

Name                ObjectClass         DistinguishedName
----                -----------         -----------------
firma               domainDNS           DC=firma,DC=local
Configuration       configuration       CN=Configuration,...
Schema              dMD                 CN=Schema,CN=Conf...
ForestDnsZones      domainDNS           DC=ForestDnsZones...
DomainDnsZones      domainDNS           DC=DomainDnsZones...

PS AD:\> cd 'DC=firma,DC=local'
PS AD:\DC=firma,DC=local>
```

Informace o aktuální doméně

```
Get-ADDomain
```

Informace o doménových řadičích

```
$dc = Get-ADDomainController -Filter *
$dc | FT Name,Site,OperationMasterRoles,OperatingSystem,IsGlobalCatalog -AutoSize
```

Uživatelé (user account) - vytváření a nastavování hodnot

Založení uživatele

Pokud máme Exchange server, tak se nám spíš hodí použít příkaz, který vytvoří zároveň účet i mailbox. Uživatele vytvoříme s defaultním heslem `123456` a určujeme kontejner, v kterém se účet vytvoří.

```
$pass = ConvertTo-SecureString -AsPlainText -Force -String '123456'
New-ADUser bouska -Name 'Bouška Petr' -SamAccountName 'bouska' -Path 'ou=Firma,dc=firma,dc=local' -UserPrincipalName 'bouska@
```

Nastavení uživatelům, že musí při příštím přihlášení změnit heslo

Aby šlo nastavit `ChangePasswordAtLogon`, tak nesmí být nastaveno `PasswordNeverExpires`.

```
Get-ADUser -Filter * -SearchBase "OU=Firma,DC=firma,DC=local" | Set-ADUser -PasswordNeverExpires $false
Get-ADUser -Filter * -SearchBase "OU=Firma,DC=firma,DC=local" | Set-ADUser -ChangePasswordAtLogon $true
```

Uživatelé (user account) - výpis účtů a hodnot

Zobrazení všech vlastností uživatele

```
Get-ADUser bouska -Properties *
```

Seznam uživatelů a počet špatně zadaných hesel

```
Get-ADUser -Filter * -Properties badPwdCount | FT Name,badPwdCount,Enabled
```

Seznam uživatelů, kteří mají nastaveno ChangePasswordAtLogon

Když chceme zobrazit seznam uživatelů, kteří mají nastaven parametr `ChangePasswordAtLogon` (to můžeme nastavit pomocí PowerShellu nebo ADUC), tak zjistíme, že žádný takový LDAP atribut neexistuje, takže takové uživatele nemůžeme zobrazit. Ono se totiž děje to, když nastavíme `ChangePasswordAtLogon`, tak se nastaví parametr `pwdLastSet` na hodnotu `0` a `PasswordExpired` na `true`. Takže uživatele můžeme zobrazit podle jedné z těchto hodnot.

```
Get-ADUser -Filter {pwdLastSet -eq 0} -Properties pwdLastSet | FT Name
```

Pro zobrazení podle atributu `PasswordExpired` by měl fungovat následující příkaz, ale mě v praxi nefunguje filtrování pouze hodnot `true`. Takže jsem to obešel druhým příkazem, kdy na začátku seznamu jsou hodnoty `true`.

```
Get-ADUser -Filter {PasswordExpired -eq $true} -Properties PasswordExpired  | FT Name, PasswordExpired
Get-ADUser -Filter * -Properties PasswordExpired | Sort-Object PasswordExpired -Descending | FT name,PasswordExpired -AutoSiz
```

Seznam disablovaných uživatelských účtů

Dvě možnosti pomocí různého příkazu.

```
Get-ADUser -Filter {Enabled -eq $false} | FT name
Search-ADAccount -AccountDisabled -UsersOnly | Sort-Object Name | FT Name,DistinguishedName -AutoSize
```

Seznam zamčených uživatelských účtů

```
Search-ADAccount -LockedOut -UsersOnly | Sort-Object Name | FT Name,DistinguishedName -AutoSize
```

Seznam uživatelů, kteří se nepřihlásili za posledních 30 dní

```
Search-ADAccount -AccountInactive -TimeSpan "30" -UsersOnly | Sort-Object LastLogonDate | FT Name,LastLogonDate,Enabled,Locke
```

Skupiny (group)

Seznam skupin, do kterých je zařazen uživatel

```
Get-ADPrincipalGroupMembership -Identity bouska
```

Zařazení uživatele do skupiny

```
Add-ADPrincipalGroupMembership -Identity bouska -MemberOf 'G Nějaká skupina'
```

Seznam uživatelů ve skupině

```
Get-ADGroupMember -Identity 'skupina 1' | FT name
```

Seznam uživatelů a počet skupin, kterých jsou členem

```
Get-ADUser -Filter * -SearchBase "OU=firma,DC=firma,DC=local" -ResultPageSize 10 | ForEach-Object {
  $groupc = (Get-ADPrincipalGroupMembership $_.SamAccountName | Measure-Object).Count
  Write-Host $_.SamAccountName $groupc
}
```

Výpis skupin dle filtru a seznam jejich členů

```
Get-ADGroup -Filter { name -like "Group *" } | ForEach-Object { $_.SamAccountName
   Get-ADGroupMember $_.SamAccountName | ForEach-Object { Write-Host "   " $_.name }
}
```

Počítače (computer account) - výpis účtů a hodnot

Počítače s OS Windows 7

```
Get-ADComputer -Filter {OperatingSystem -like "Windows 7*"}  -Properties OperatingSystem, CanonicalName, whenCreated | Sort-O
```

Počítače s disablovaným účtem

```
Search-ADAccount -ComputersOnly -AccountDisabled | Sort-Object Name | FT Name,DistinguishedName,LastLogonDate -AutoSize
```

Počítače, které se dlouho nepřihlásili

Buď můžeme požít kolik posledních dní se nepřihlásili, pomocí `TimeSpan`, nebo od kdy se nepřihlásili, pomocí `DateTime`. V obou případech mi ale připadá, že datum úplně nesedí, a zobrazují se záznamy o 14 dní starší.

Doplněno. Tato situace je trochu komplikovanější a podrobně jsem ji popsal v článku **Powershell - poslední přihlášení počítače či uživatele**.

```
Search-ADAccount -ComputersOnly -AccountInactive -TimeSpan "30" | Sort-Object LastLogonDate | FT Name,LastLogonDate,Enabled,L
Search-ADAccount -ComputersOnly -AccountInactive -DateTime "1.11.2010" | Sort-Object LastLogonDate | FT Name,LastLogonDate,En
```

V některých případech můžeme potřebovat zobrazit i jiné hodnoty, tak se může hodit alternativa pomocí `Get-ADComputer` a filtru.

```
$logonDate = New-Object System.DateTime(2010,9,19)
Get-ADComputer -Filter { lastLogon -le $logonDate } | FT
Get-ADComputer -Filter { lastLogon -le $logonDate } -Properties LastLogonTimeStamp, OperatingSystem  | Select-Object Name, Di
```

Počítače, které si dlouho nezměnili heslo

Počítače, které si změnili heslo naposled před 45 dny. Počítače v doméně si musí pravidelně měnit heslo (defaultně každých 30 dní), pokud tak neučiní, znamená to, že jsou neaktivní.

```
$lastSetdate = [DateTime]::Now - [TimeSpan]::Parse("45")
Get-ADComputer -Filter {PasswordLastSet -le $lastSetdate} -Properties passwordLastSet -ResultSetSize $null | FT samaccountnam
```


Autor: **Petr Bouška**

Související články:

PowerShell

Články týkající se Microsoft skriptovacího jazyku PowerShell, který se používá ve všech nových verzích MS OS a aplikací.

- **PowerShell - Active Directory** [14.11.2010 13:13] ..... **právě čtete**
- **PowerShell - Exchange server** [19.11.2010 08:45]
- **PowerShell - Windows** [12.12.2010 18:00]
- **PowerShell - Ovládání aplikací a další použití** [09.01.2011 18:59]
- **Exchange Web Services a PowerShell** [13.01.2011 11:00]
- **Active Directory Recycle Bin** [31.03.2011 15:48]
- **Automatizovaná kontrola prostředí pomocí PowerShellu** [13.06.2011 17:35]
- **Powershell - poslední přihlášení počítače či uživatele** [06.09.2011 14:44]
- **Exchange Server, Outlook a certifikáty v GALu** [27.12.2011 14:48]

PowerShell - Exchange server

Upraveno 07.08.2013 16:10 | vytvořeno 19.11.2010 08:45 | **Samuraj - Petr Bouška** | ***Microsoft admin***

Na rozdíl od Active Directory se pro MS Exchange Server 2007 musel používat PowerShell pro řadu operací již od začátku. Některé operace byly časem doplněny do GUI Exchange Management Console, ale některé můžeme provést pouze přes PowerShell. Takže opět můj malý tahák užitečných PowerShell příkazů pro Exchange server (zatím ve verzi 2007 a 2010).

Obecné vlastnosti

Pokud chceme používat *PowerShell ISE*, a ne *Exchange Management Shell*, tak nejprve musíme načíst příkazy Exchange serveru.

```
Add-PSSnapin Microsoft.Exchange.Management.PowerShell.Admin
```

Pro Exchange Server 2010 můžeme použít.

```
Add-PSSnapin Microsoft.Exchange.Management.PowerShell.E2010
```

Jiná možnost je připojit se vzdáleně na Exchange Powershell.

```
$user = Get-Credential
$ExchSession = New-PSSession -ConfigurationName Microsoft.Exchange -ConnectionURI http://mail.firma.local/powershell -Credent
Import-PSSession $ExchSession
# příkazy Exchange
Remove-PsSession $ExchSession
```

Uživatelé a mailboxy (schránky)

Založení uživatele s mailboxem

Uživatele vytvoříme s defaultním heslem `123456` a určujeme kontejner, v kterém se účet vytvoří.

```
$pass = ConvertTo-SecureString -AsPlainText -Force -String '123456'
New-Mailbox -Name 'Bouška Petr' -Alias 'bouska' -OrganizationalUnit 'firma.local/Firma -UserPrincipalName 'bouska@firma.local
```

Informace o mailboxu

```
Get-Mailbox -Identity bouska | FL
```

Nastavení AutoAccept pro události v kalendáři

Například u zasedací místnosti chceme, aby automaticky schvalovala rezervace. Důležitý parametr je `BookingWindowInDays`, který určuje, jak dlouhou (opakující se) událost můžeme zadat.

```
Set-MailboxCalendarSettings -Identity zasedacka -BookingWindowInDays 360 -AutomateProcessing AutoAccept -AddOrganizerToSubjec
```

Oprávnění na mailbox

Zjištění oprávnění

```
Get-MailboxPermission -Identity zasedacka | FT -AutoSize
```

Nastavení oprávnění

```
Add-MailboxPermission -Identity zasedacka -User bouska -AccessRights FullAccess -InheritanceType All
```

Odebrání oprávnění

```
Remove-MailboxPermission -Identity zasedacka -User bouska -AccessRights FullAccess -InheritanceType All
```

Informace o kalendáři

```
Get-MailboxCalendarSettings -Identity zasedacka | FL
```

Nastavení delegace pro kalendář

```
Set-MailboxCalendarSettings -Identity zasedacka -ResourceDelegates bouska -ForwardRequestsToDelegates $true
```

Nastavení oprávnění pro kalendář

```
Add-MailboxFolderPermission -Identity či-kalendář:\Calendar -User komu-dáme-práva -AccessRights Reviewer
```

Seznam databází

```
Get-MailboxDatabase
```

Seznam mailboxů

```
Get-Mailbox
Get-Mailbox -Database SERVER\VIP
```

Seznam velikostí mailboxů

```
Get-MailboxStatistics -Database SERVER\VIP
Get-MailboxStatistics -Server SERVER -Identity bouska
Get-MailboxStatistics | select DisplayName,TotalItemSize,ItemCount,LastLogonTime | sort TotalItemSize
```

Vypnutí POP3 a IMAP

```
Set-CASMailbox -Identity bouska -PopEnabled $false -imapEnabled $false
```

Seznam mailboxů, které mají zapnutý POP3 či IMAP

```
Get-CASMailbox -Filter {popEnabled -eq "True"}
Get-CASMailbox -Filter {imapEnabled -eq "True"}
```

Nastavení IMAP a POP3 pro mailbox

Uživatelé mají defaultně povolený IMAP i POP3, pokud tomu chceme zabránit, tak můžeme pouze vypnout služby. Pokud ale chceme mít pro pár vybraných účtů povoleno třeba IMAP a ostatním to vypnout, tak to jednoduše nelze. Jedno řešení je těm speciálním účtům nastavit rozšířený atribut třeba na hodnotu IMAP a všem účtům, které toto nastavení nemají vypnout.

Seznam uživatelů, kteří mají nastavený `customAttribute1` (v ADUC u uživatele záložka Attribute Editor, položka `extensionAttribute1`)

```
Get-Mailbox -Filter {customAttribute1 -eq "IMAP"}
```

Vypnutí POP3 pro všechny uživatele, kteří jej mají zapnuto

```
Get-CASMailbox -Filter {popEnabled -eq "True"} | Set-CASMailbox -PopEnabled $false
```

Vypnutí IMAP u všech, kteří nemají nastavený atribut

```
Get-CASMailbox -Filter {imapEnabled -eq "True"} | get-Mailbox | where{$_.customAttribute1 -ne "IMAP"} | Set-CASMailbox -imapE
```

Změna limitu na velikost pravidel v Rules and Alerts

Uživatel může mít ve své schránce vytvořeno pouze omezené množství aktivních pravidel (Rules and Alerts). Toto omezení je dánou společnou velikostí, kterou tato pravidla zabírají, a na Exchange 2007 je to 64 kB.

```
Set-Mailbox -Identity bouska -RulesQuota 128kB
```

Veřejné složky (Public Folders)

Zjištění uživatelských oprávnění na Public Folder

```
Get-PublicFolderClientPermission -Server SERVER -Identity "\složky\hotline"
```

Nastavení uživatelských oprávnění na Public Folder

```
Add-PublicFolderClientPermission -Identity "\složky\hotline" -AccessRights Owner -User bouska
```

Zjištění oprávnění na AD objektu

```
Get-ADPermission -Identity hotline | FT -AutoSize
```

Nastavení SendAs oprávnění

```
Add-ADPermission -Identity hotline -User bouska -ExtendedRights Send-as
```

Seznam velikostí Public Folderů

```
Get-PublicFolderStatistics -Server SERVER -ResultSize Unlimited | Sort-Object TotalItemSize | FT Name,FolderPath,ItemCount,To
```

Další statistiky

```
Get-PublicFolderStatistics -Server SERVER -ResultSize Unlimited | FT Name, FolderPath, ItemCount, @{label='TotalItemSize(KB)'
Get-PublicFolderStatistics -Server SERVER -ResultSize Unlimited | ForEach-Object {$_.TotalItemSize.Value.ToMB()} | Measure-Ob
Get-PublicFolderStatistics -Server SERVER -ResultSize Unlimited | Measure-Object -Sum -Property ItemCount
```

Seznam Public Folderů, jak velký je jejich soubor a kolik obsahuje prázdného místa

Hodnota AvailableNewMailboxSpace nám ukazuje, kolik místa můžeme uvolnit, když provedeme Offline Defrag.

```
Get-PublicFolderDatabase -Status | FT Name, DatabaseSize, AvailableNewMailboxSpace
```

Seznamy adres (Address List)

Seznam všech Address List

```
Get-AddressList
```

Přejmenování Address Listu

Změna jména a zobrazovaného jména seznamu. Jméno můžeme změnit i přes GUI, ale nezměníme tam DisplayName, takže uživatelé v Outlooku stále uvidí staré jméno.

```
Set-AddressList -Identity "All Users " -Name "Zaměstnanci"
Set-AddressList -Identity "Zaměstnanci" -DisplayName "Zaměstnanci"
```

Zprávy a logy (email)

Administrator Audit Log

Od Exchange 2010 SP1 se ukládá záznam o každém administrátorském zásahu do speciálního arbitration mailboxu. Informace z něj můžeme získat pomocí dvou cmdletů nebo ECP.

Jednoduše můžeme zobrazit použití určitého cmdletu.

```
Search-AdminAuditLog -Cmdlets New-Mailbox
```

Nebo si výsledek dotazu můžeme nechat zaslat na email v podobě XML.

```
New-AdminAuditLogSearch -Name "Mailbox change log" -Cmdlets New-Mailbox, Set-Mailbox -StatusMailRecipients bouska@oksystem.cz
```

Nalezení určitých zpráv za danou dobu

Potřebujeme zjistit, jestli za poslední den přišla nějaká zpráva na určitou adresu, nebo chceme vypsat všechny příchozí zprávy od nějaké odesílatele, podobných situací je mnoho. Použijeme cmdlet `Get-MessageTrackingLog`, který, jak již název napovídá, prochází message tracking log a vyhledává určité záznamy. Specifikovat můžeme kategorii (jako Receive, Send), odesílatele, příjemce, časové rozmezí, předmět (subject - pokud jej logujeme), apod.

```
Get-MessageTrackingLog -server MAIL1 -EventID "RECEIVE" -Recipients "samuraj@samuraj-cz.com" -Start "01/12/2010 00:00:00" -En
Get-MessageTrackingLog -server MAIL1 -EventID "RECEIVE" -Sender "samuraj@samuraj-cz.com" -Recipients "samuraj@samuraj-cz.com"
```

Pokud chceme zjistit, jestli za poslední hodinu přišla nějaká zpráva, můžeme například použít.

```
$time = Get-Date
$mails = Get-MessageTrackingLog -server OKMAIL1 -EventID "RECEIVE" -Sender "samuraj@samuraj-cz.com" -Recipients "samuraj@samu
($mails | Measure-Object).Count
```

Statistiky za poslední dobu

Může se hodit zjistit různé statistické údaje z odesílané a přijímané pošty. Takže třeba top 10 adres z jakých byla odesílána pošta za posledních 12 hodin.

```
Get-MessageTrackingLog -Start (Get-Date).AddHours(-12) -End (Get-Date) -ResultSize unlimited | group-object -Property Sender
```

Nebo podobně, počty zpráv odeslaných z adresa za poslední hodinu.

```
Get-MessageTrackingLog -Start (Get-Date).AddHours(-1) -End (Get-Date) -ResultSize unlimited | group-object -Property Sender |
```

Další podobný příklad je počet zpráv pro příjemce za dané období.

```
Get-MessageTrackingLog -Start "8/6/2013 08:00:00" -End "8/6/2013 10:00:00" -ResultSize unlimited | Select-Object -ExpandPrope
```

A nebo seznam zpráv větších než 1MB za poslední hodinu.

```
Get-MessageTrackingLog -Start (Get-Date).AddHours(-1) -End (Get-Date) -ResultSize unlimited  | Select-Object sender,recipient
```

Testy a kontroly

Test funkčnosti ActiveSync pro určitého uživatele (musíme zadat jeho heslo).

```
Test-ActiveSyncConnectivity -MailboxCredential (get-credential user@domain)
```

Test synchronizace s Edge serverem a ověří, jestli má aktuální data.

```
Test-EdgeSynchronization
```

Ověřuje, jestli funguje odesílání a doručování emailů, může testovat i mezi definovanými mailbox servery a ověřovat zpoždění.

```
Test-Mailflow
```

Přihlásí se k definované nebo systémové schránce, tím ověří funkčnost MAPI, LDAP a Exchange store.

```
Test-MAPIconnectivity
```

Ověřuje funkci autodiscover.

```
Test-OutlookWebServices | FT -AutoSize
```

Ověřuje funkčnost OWA pro určitého uživatele (musíme zadat jeho heslo).

```
Test-OwaConnectivity -MailboxCredential (get-credential user@domain)
```

Ověřuje funkčnost POP3 pro určitého uživatele (musíme zadat jeho heslo)..

```
Test-PopConnectivity -MailboxCredential (get-credential user@domain)
```

Testuje jestli daná IP adresa je legitimní pro odesílání mailů z dané domény.

```
Test-SenderId -IPAddress 1.2.3.4 -PurportedResponsibleDomain domain.tld
```

Vypíše seznam všech Exchange služeb a otestuje zda běží a jsou nastaveny na autostart.

```
Test-ServiceHealth
```

Získá data o Exchange systému a podle Best Practices vypíše doporučení.

```
Test-SystemHealth
```

PowerShell - Windows

Neděle, 12.12.2010 18:00 | Samuraj - Petr Bouška | Microsoft admin

PowerShel využijeme pro spoustu oblastí. Jednou z nich je samozřejmě operační systém. Zjišťování údajů, konfigurace apod. Pro spouštění příkazů nejčastěji používám PowerShell ISE, které je součástí Windows 7.

> *Protože je stále málo času, tak v době zveřejnění tohoto článku neobsahuje příliš příkazů, ale časem budu doplňovat. Na různých místech mám poznámky, v kterých chci udělat pořádek. Určitě je tu i příležitost pro vás, podělit se o zajímavý příkaz v komentářích.*

Obecné

Enumeration constants

Pro použití je více možností.

```
[Microsoft.Office.Interop.Word.WdSaveFormat]::wdFormatPDF
[Enum]::Parse([Microsoft.Office.Interop.Word.WdSaveFormat], "wdFormatRTF")
```

Vypsání hodnot.

```
[Microsoft.Office.Interop.Word.WdSaveFormat] | Get-Member -Static -MemberType Property
```

Formátování výstupu

Hlavní dvě formátování jsou do tabulky `Format-Table` alias `FT` a pomocí seznamu `Format-List` alias `FL`. Výstup předáváme do formátovací funkce pomocí `|` (pipe). Za funkci můžeme vypsat seznam sloupců, které chceme zobrazit. Také můžeme použít `-AutoSize` pro přizpůsobení šířky sloupců u tabulky.

```
Get-Process | FT ProcessName, CPU -AutoSize
```

Řazení výstupu

Pomocí `Sort-Object` můžeme řadit výstup, který zobrazujeme. Pokud chceme zároveň formátovat výstup, tak musíme nejprve řadit.

```
Get-Process | Sort-Object -Descending CPU
Get-Process | Sort-Object -Descending CPU | FT name,cpu -AutoSize
```

Zjištění typu proměnné

```
$x = 15
$x.GetType().FullName
System.Int32
```

Zjištění metod a vlastností objektu

```
$objekt | Get-Member
```

Statistiky objektu, počet prvků

```
$objekt | Measure-Object
($objekt | Measure-Object).Count
```

Registry

Registry můžeme procházet stejně, jako souborový systém. Podstromy jsou mapovány na disky, `HKEY_LOCAL_MACHINE` na `HKLM:` a `HKEY_CURRENT_USER` na `HKCU:`.

```
PS C:\>cd HKCU:
PS HKCU:\> dir
PS HKCU:\> cd Software
PS HKCU:\Software> dir
```

Pro čtení hodnot se používá cmdlet `Get-ItemProperty` (pro aktuální `Get-ItemProperty .`), pro zápis `Set-ItemProperty`, pro mazání `Remove-ItemProperty`.

Systémové proměnné prostředí - Environment Variables

Výpis všech proměnných.

```
Get-ChildItem Env:
```

Výpis jedné hodnoty.

```
$Env:COMPUTERNAME
```

Vytvoření nové hodnoty.

```
$Env:test = "Pokus"
```

Záznamy do EventLogu

Seznam logů, které jsou v systému.

```
Get-EventLog -list
```

Vytvoření záznamu do logu. Jde vytvořit záznam pouze pro existující `Source` a důležité je i `EventId`.

```
Write-EventLog -LogName "Application" -Source "PerfOS" -EventId 2011 -EntryType Error -Message "Test"
```

Výpis nejnovějších 5 záznamů daného typu a ID z logu System.

```
Get-EventLog -LogName system -EntryType warning -InstanceId 129 -Newest 5
```

Otevření webové adresy (URL)

Otevření okna prohlížeče a v něm dané adresy.

```
(New-Object -com Shell.Application).Open("http://www.samuraj-cz.com")
Start-Process -FilePath "http://www.samuraj-cz.com"
```

Pokud nechceme otevírat prohlížeč, ale chceme pouze zavolat URL, či zpracovávat obsah, můžeme použít následující kód, který vrátí obsah stránky (můžeme jej uložit do proměnné).

```
(new-object net.webclient).DownloadString("http://www.samuraj-cz.com")
```

Práce s časem

```
$time = Get-Date
(Get-Date).AddHours(-1)
(Get-Date).Day
```

Odeslání emailu

```
$smtp = new-object Net.Mail.SmtpClient("mail.domena.tld")
$smtp.Send(New-Object System.Net.Mail.MailMessage('prijemce@domena.tld','odesilatel@domena.tld','předmět','text'))
```

Nebo nativními prostředky

```
Send-MailMessage -From odesilatel@domena.tld -To prijemce@domena.tld -Subject "předmět" -SmtpServer server -Attachments soubo
```

Souborový systém

Práce se soubory a adresáři

Zjištění, jestli existuje daný adresář nebo soubor.

```
Test-Path C:\Scripts\test
```

Vytvoření adresáře.

```
New-Item C:\Scripts\test -type directory
```

Vytvoření souboru.

```
New-Item C:\Scripts\pokus.txt -type file
```

Vyhledání složek, kde je nastaveno oprávnění pro AD skupinu

Jednoduché hledání, které prochází zadanou cestu (včetně vnořených složek) a vypisuje ty složky, na které je nastavena zadaná AD skupina. V tomto případě, pokud existuje skupina, jejíž jméno začíná stejně, tak nalezne i tuto (nejde o přesnou shodu).

```
$StrGroup = "DL Skupina"
$Folder = "C:\1"
Get-ChildItem $Folder -Recurse | where { $_.Psiscontainer } | Get-Acl | where {$_.AccessToString -match $StrGroup} | select p
```

Operační systém

Zjištění verze OS

Pro řadu úkonů zatím nemáme nativní cmdlet, takže můžeme využít širokých možností WMI (pokud jej máme zapnuté).

Vrátí verzi OS, info o SP, architekturu (32 bit nebo 64 bit).

```
Get-WmiObject Win32_OperatingSystem -ComputerName pocitac | FL Caption,ServicePackMajorVersion,OSArchitecture
```

> ***Pozn.:*** *Místo příkazu* ***Get-WmiObject*** *můžeme použít jeho alias* ***gwmi.***

Windows služby

Seznam všech služeb a barevné odlišení běžících.

```
Get-Service | Sort-Object status,displayname |
ForEach-Object { if($_.status -eq "running") {
  Write-Host $_.status `t $_.name `t $_.displayname -ForegroundColor "green"
} elseif( $_.status -eq "stopped" ) {
  Write-Host $_.status `t $_.name `t $_.displayname -ForegroundColor "red"
} else {
  Write-Host $_.status `t $_.name `t $_.displayname
} }
```

Hardware

Zjištění informací o připojeném disku

V příkladu zjišťujeme výrobce, model, velikost a ID USB flash disku. To samé můžeme použít i pro pevné disky. Volání provádíme pomocí PowerShellu, ale ve skutečnosti se používá WMI (Windows Management Instrumentation). Je možno volat lokálně i na vzdálený počítač (pokud k němu máme práva).

```
gwmi Win32_DiskDrive | where-object {$_.InterfaceType -like  "USB"} | fl Model,Size,PNPDeviceID
gwmi Win32_DiskDrive -ComputerName pocitac | where-object  {$_.InterfaceType -like "USB"} | fl Model,Size,PNPDeviceID
```

PowerShell - Ovládání aplikací a další použití

Neděle, 09.01.2011 18:59 | **Samuraj - Petr Bouška** | ***Microsoft admin***

Další článek o užitečném nástroji (někdy i nezbytnosti) Microsoft PowerShell. Tentokrát si uvedeme nějaké speciální příklady použití, kdy můžeme řídit aplikace jako je MS Word a Outlook nebo přistupovat k Active Directory. PowerShell totiž umožňuje přistupovat ke COM objektům nebo k třídám z .NET Framework, tedy k programátorským rozhraním různých aplikací.

Microsoft Word

Musíme mít nainstalovánu aplikaci MS Word. Potom můžeme použít `ProgID` od `COM objektu` Wordu a vytvořit objekt. Následně můžeme provádět řadu operací, jako otevírat dokumenty, upravovat obsah, nastavovat parametry (a to i některé pro Outlook). Vše se může dít skrytě nebo veřejně (v příkladu pomocí `visible`).

```
$MSWord = New-Object -ComObject Word.Application
$MSWord.Visible = $true
# můžeme otevřít existující soubor nebo vytvořit nový
# $MSWord.Documents.Open("d:\test.docx")
$WDoc = $MSWord.Documents.Add()
$WPar = $WDoc.Paragraphs.Add()
$WPar.Range.Text = "Test"
$WDoc.SaveAs([ref]"d:\test.docx")
$MSWord.Quit()
```

Microsoft Outlook

Některé nastavení Outlooku můžeme provést přes objekt `Word.Application`. V příkladu je nastavení podpisu pro nové správy (musí existovat podpis s tímto jménem).

```
$MSWord = New-Object -ComObject Word.Application
$MSWord.EmailOptions.EmailSignature.NewMessageSignature = "podpis"
$MSWord.Quit()
```

Pro některé funkce máme objekt `Outlook.Application`. Příklad přistoupí ke kalendáři a vrátí první záznam (standardně je vyžadováno potvrzení přístupu v aplikaci Outlook).

```
$outlook = New-Object -ComObject Outlook.Application
$mapi = $outlook.GetNamespace("MAPI")
$calendar = $mapi.GetDefaultFolder("olFolderCalendar")
$calendar.Items.GetFirst()
```

Internet Explorer

Stejným způsobem můžeme pracovat s Internet Explorerem.

```
$ie = New-Object -ComObject InternetExplorer.Application
$ie.Visible = $true
$ie.Navigate("www.samuraj-cz.com")
$ie.Quit()
```

Shell

Objekt `Shell.Application` nám zařídí přístup do systému (shellu). Uvedené příklady provedou - otevření Windows Explorer se zadanou složkou, otevření adresy v přiřazeném prohlížeči, minimalizuje všechna otevřená okna, zobrazí dialog pro vypnutí systému.

```
$shell = New-Object -ComObject Shell.Application
$shell.Explore("c:\")
$shell.Open("http://www.samuraj-cz.com")
$shell.MinimizeAll()
$shell.ShutdownWindows()
```

Active Directory

Pro pohodlný přístup k AD můžeme použít modul ***Active Directory Module for Windows PowerShell***. To ale znamená, že na počítači, kde chcem spustit skript, musí být nainstalovaný. V některých případech to nemůžeme zajistit, ale pokud na stanicích máme .NET Framework, tak můžeme použít jeho možností. Ten obsahuje ***System.DirectoryServices Namespace***, kde jsou třídy `System.DirectoryServices.DirectorySearcher` a `System.DirectoryServices.DirectoryEntry`, které slouží k přístupu k AD DS.

V příkladu vytvoříme instanci třídy `System.DirectoryServices.DirectorySearcher`, zadáme dotaz na vyhledání uživatele (aktuálně přihlášeného do systému, využijeme proměnnou prostředí) a zobrazíme jeho jméno a email. Použití cmdletu z Active Directory modulu je jednodušší, ale i tato metoda je funkční.

```
$Searcher = New-Object System.DirectoryServices.DirectorySearcher
$Searcher.Filter = "(&(objectCategory=User)(samAccountName="+$Env:username+"))"
$ADUser = $Searcher.FindOne().GetDirectoryEntry()
$ADUser.DisplayName
$ADUser.mail
```

Druhý příklad ukazuje vypsání seznamu doménových kontrolerů. A možností je samozřejmě mnohem více.

```
$domain = [System.DirectoryServices.ActiveDirectory.Domain]::getcurrentdomain()
$domain.FindAllDomainControllers() | FT Name,SiteName,Roles -AutoSize
```


Autor: **Petr Bouška**

Související články:

PowerShell

Články týkající se Microsoft skriptovacího jazyku PowerShell, který se používá ve všech nových verzích MS OS a aplikací.

- **PowerShell - Active Directory** [14.11.2010 13:13]
- **PowerShell - Exchange server** [19.11.2010 08:45]
- **PowerShell - Windows** [12.12.2010 18:00]
- **PowerShell - Ovládání aplikací a další použití** [09.01.2011 18:59] ..... **právě čtete**
- **Exchange Web Services a PowerShell** [13.01.2011 11:00]
- **Active Directory Recycle Bin** [31.03.2011 15:48]
- **Automatizovaná kontrola prostředí pomocí PowerShellu** [13.06.2011 17:35]
- **Powershell - poslední přihlášení počítače či uživatele** [06.09.2011 14:44]
- **Exchange Server, Outlook a certifikáty v GALu** [27.12.2011 14:48]

Články

Exchange Web Services a PowerShell

Čtvrtek, 13.01.2011 11:00 | **Samuraj - Petr Bouška** | ***Microsoft admin***

Pro práci s Exchange serverem máme k dispozici řadu cmdletů v PowerShellu a pro některé funkce GUI. Ale někdy se může hodit i trochu jiný způsob. Můžeme použít Exchange Web Services Managed API, což je .NET rozhranní k EWS, které využívá Web Services SOAP protokol a Autodiscover. A stejně jako jiné .NET rozrhaní jej můžeme použít rovnou z PowerShellu.

Od verze Exchange 2007 SP1 obsahuje poštovní server ***Exchange Web Services*** (EWS), tedy rozhraní pro přístup k položkám v Exchange data store. Pro programování nebo skriptování je k dispozici **Exchange Web Services Managed API**, což je knihovna, která dovolí využít EWS a to i třeba z PowerShellu. Můžete to být zajímavý nástroj, pouze se mi zdá podivný způsob, jak je umožněno přistupovat ke schránkám uživatelů. I když jste Exchange Admin, tak nemáte oprávnění pomocí EWS získat nějaké informace, ale musí se použít **Impersonation** (popis pro Exchange 2007), což je zjednodušeně řečeno nastavení speciálního oprávnění na účty uživatelů.

Tento článek je pouze malým navedením, jak EWS v PowerShellu použít.

Nejprve musíme načíst knihovnu EWS.

```
Import-Module -Name "C:\Program Files\Microsoft\Exchange\Web Services\1.1\Microsoft.Exchange.WebServices.dll"
```

Vytvoříme objekt služby Exchange, důležité je specifikovat verzi Exchange (jinak dostaneme chybu při bind, zadání s SP2 mi nefungovalo).

```
$exchService = New-Object Microsoft.Exchange.WebServices.Data.ExchangeService([Microsoft.Exchange.WebServices.Data.ExchangeVe
```

Musíme zadat adresu CAS serveru, buď ručně nebo se určí automaticky pomocí Autodiscover (pro určitého uživatele).

```
$exchService.Url = "https://mail.firma.local/EWS/Exchange.asmx"
$exchService.AutodiscoverUrl("bouska@firma.cz")
```

Další důležitý krok je, k jaké schránce a pod jakými právy budeme přistupovat, máme tři možnosti. U prvních dvou zadáváme uživatele, pod kterým budeme přistupovat, a defaultně se přistupuje k jeho schránce. První možnost, využijí se credentials aktuálního uživatele.

```
$exchService.UseDefaultCredentials = $true
```

Nebo zadáme credentials nějakého určitého uživatele, musí se zadat i jeho heslo.

```
$exchService.Credentials = New-Object Microsoft.Exchange.WebServices.Data.WebCredentials($user, "heslo", "domain")
```

Poslední možnost je využít impersonalizace, tehdy přistupujeme k účtu zadanému v této funkci. Standardně pod uživatelem, který spustil skript, a ten musí mít práva pro Impersonation.

```
$exchService.ImpersonatedUserId = New-Object Microsoft.Exchange.WebServices.Data.ImpersonatedUserId([Microsoft.Exchange.WebSe
```

Nyní se již můžeme připojit k nějaké složce (folder) uživatele. V příkladu se připojíme ke kalendáři.

```
$calendar = [Microsoft.Exchange.WebServices.Data.Folder]::Bind($exchService, [Microsoft.Exchange.WebServices.Data.WellKnownFo
```

Zde můžeme například zobrazit práva na kalendáři. Druhý řádek zobrazí informace o uživateli, který je přiřazen první položce.

```
$calendar.Permissions
$calendar.Permissions[0].UserId
```

Práva můžeme i nastavit (pokud pro daného uživatele již práva nastavena jsou, dostaneme chybu). Na konci je třeba provést aktualizaci, aby se změna provedla.

```
$FolderPermission = New-Object Microsoft.Exchange.WebServices.Data.FolderPermission("nekdo-jiny@firma.cz", [Microsoft.Exchang
$calendar.Permissions.Add($FolderPermission)
$calendar.Update()
```

Active Directory Recycle Bin

Čtvrtek, 31.03.2011 15:48 | Samuraj - Petr Bouška | *Microsoft admin*

Novinkou na doméně (lese) ve funkčním stupni Windows Server 2008 R2 je koš (Recycle Bin). Myslím, že je to užitečná vlastnost, i když jsem ji za rok, co je zapnutá, použil pouze jednou. Rozhodně ničemu neškodí. Dělá to, co každý tuší. Smazaný objekt v Active Directory Domain Services (AD DS) se přesune do koše a jej možné je obnovit do původního stavu.

Bez funkce ***Recycle Bin*** se při smazání objektu ještě úplně neodstraní, ale stane se z něj ***tombstone*** (náhrobek) a po dobu ***tombstone lifetime*** (od Windows Server 2003 SP1 je to default 180 dní) je možné jej obnovit metodou ***tombstone reanimation***. Problém je, že ztratíme některé atributy, například členství ve skupinách.

Funkce ***Active Directory Recycle Bin*** je defaultně vypnutá, pokud ji zapneme, tak se po smazání objekt přesune do kontejneru ***Deleted Objects*** a odtud může být obnoven, do stejného stavu jako před smazáním, po dobu ***deleted object lifetime*** (standardně 180 dní). Po vypršení tohoto času se z ***deleted object*** stává ***recycled object*** (obdoba tombstone) a ještě další dobu (standardně také 180 dní) může být obnoven bez některých atributů. Pak se teprve fyzicky smaže.

Zapnutí AD Recycle Bin

Podmínkou tedy je mít funkční stupeň lesa **Windows Server 2008 R2**. Jakmile tuto funkci zapneme, tak ji již nelze vypnout. Zapnutí musíme provádět pod účtem s právy ***Enterprise Admins***. Nejjednodušší je zapnutí této funkce pomocí ***cmdletu z PowerShell modulu ActiveDirectory***. Nejprve je formát příkazu a potom příklad použití pro doménu **firma.local**.

```
Enable-ADOptionalFeature -Identity <ADOptionalFeature> -Scope <ADOptionalFeatureScope> -Target <ADEntity>
Enable-ADOptionalFeature –Identity "CN=Recycle Bin Feature,CN=Optional Features,CN=Directory Service,CN=Windows NT,CN=Services,CN=Configuration,DC
```

Nastavení doby, kdy je možné objekt obnovit

Pokud chceme změnit výchozí hodnotu `msDS-deletedObjectLifetime`, tak můžeme opět použít ***PowerShell*** (v příkladu doména firma.local a doba 60 dní).

```
Set-ADObject -Identity "CN=Directory Service,CN=Windows NT,CN=Services,CN=Configuration,DC=firma,DC=local" –Partition "CN=Configuration,DC=firma,DC=
```

Obnovení smazaného objektu

Obnovení smazaného objektu není úplně jednoduché, ale určitě není těžké a máme několik možností. Můžeme použít nástroj `ldp.exe` nebo můžeme použít ***cmdlet z PowerShellu*** (modul ActiveDirectory) a nebo můžeme použít jednoduchou grafickou aplikaci **ADRecycleBin** od firmy ***Overall Solutions***, která je poskytovaná zdarma.

Obnovení pomocí PowerShellu

Operace musíme provádět pod účtem s právy ***Domain Admins***. Nejprve musíme nalézt smazaný objekt, to můžeme pomocí příkazu `Get-ADObject`. Například seznam všech smazaných uživatelů zobrazíme:

```
Get-ADObject -Filter {(deleted -eq $true) -and (ObjectClass -eq "user")} -Properties sAMAccountName -IncludeDeletedObjects | FT Name, sAMAccountName, Del
```

Určitého uživatele můžeme vybrat třeba podle uživatelského jména:

```
Get-ADObject -Filter {sAMAccountName -eq "novak"} –IncludeDeletedObjects
```

A pro obnovení využijeme cmdlet `Restore-ADObject`:

```
Get-ADObject -Filter {sAMAccountName -eq "novak"} -IncludeDeletedObjects | Restore-ADObject
```

Podobně můžeme obnovit účet počítače:

```
Get-ADObject -Filter {sAMAccountName -eq "pcname$"} -IncludeDeletedObjects | Restore-ADObject
```

Obnovení pomocí ADRecycleBin

Aplikaci můžeme stáhnout z adresy ADRecycleBin, kde je i její popis. Použití je opravdu jednoduché a funkční. Musíme ji spustit pod účtem ***doménového admina***. Vybereme, jaké objekty chceme zobrazit a klikneme na ***Load Deleted Objects***, nyní se nám zobrazí seznam objektů. Zaškrtneme ty, které chceme obnovit a klikneme na ***Restore Checked Objects***. V AD již uvidíme opět naše objekty.

Na závěr již jen odkaz na podrobný popis od MS v článku Active Directory Recycle Bin Step-by-Step Guide.

Automatizovaná kontrola prostředí pomocí PowerShellu

Pondělí, 13.06.2011 17:35 | Samuraj - Petr Bouška | Microsoft admin

aneb Pár tipů jak využít PowerShell. Každá síť potřebuje dohled (monitoring, kontrolu) a čím je větší, tím je složitější provádět různé kontroly ručně. Existuje řada systémů, které tyto kontroly budou provádět za nás, ale ty komplexní většinou stojí nemalé peníze. Přitom se dá vše postavit zdarma s menším či větším úsilím. Důležité je si uvědomit, co nás v naší síti zajímá, a na co chceme být upozorněni. Pak můžeme vytvářet jednoúčelové skripty, které nám zajistí potřebný přehled.

Tento článek jsem napsal pro Microsoft TechNet Blog CZ/SK, kde vyšel jako Automatizovaná kontrola prostředí pomocí PowerShellu a zároveň byl publikován v Microsoft sekci na Živě Automatizovaná kontrola prostředí pomocí PowerShellu.

Základem monitoringu, ale stále zůstává ***odesílání důležitých událostí*** ze serverů na jedno centrální místo, kde se budou kontrolovat a vybrané události odesílat třeba na email. Tak můžeme kontrolovat chybné pokusy o přihlášení, zamčení účtu, vytvoření nějakého objektu v Active Directory, restart serveru, apod.

Druhou nezbytnou oblastí je nějaký ***standardní monitoring***, který pravidelně kontroluje dostupnost serverů a vybraných služeb. Tak zjistíme kompletní pád serveru, nedostupnost služby (jako například smtp protokol na poštovním serveru), nedostatek místa na disku, dlouhodobé vytížení procesoru, apod.

Dnes se ale podíváme na něco jiného. Můžeme vytvořit jednoduché (samozřejmě i složité) ***skripty v PowerShellu***, které se automaticky spustí každý den (či jinou definovanou periodu) a odešlou nám určitý ***report***. Potom krátkým pohledem zjistíme stav v určité oblasti. Podobně můžeme vytvořit řadu ***pravidelných kontrol***, které nám pošlou zprávu, pokud je něco v nestandardním stavu. V dalších řádcích si ukážeme pár praktických příkladů, jak toto využít na Microsoftím prostředí.

Úvod ke skriptům v PowerShellu

Nebudeme se zde věnovat základům PowerShellu, ale pokud je neznáte, tak se nemusíte obávat. Použití není nikterak složité a možná zjistíte, že je to opravdu užitečný nástroj. Ani dále v článku nebudeme rozebírat jednotlivé příkazy PowerShellu, ale ukážeme si pár jednoduchých praktických příkladů. Ty budou vždy provádět jednu činnost a z jejich zápisu určitě pochopíme význam jednotlivých částí.

Spouštění PowerShell skriptů

Standardně v operačních systémech není povoleno ***spouštět PowerShell skripty***, jde o bezpečnostní nastavení zvané Execution Policy. Toto nastavení můžeme v systému změnit pomocí *PowerShell příkazu* nebo pomocí *Group Policy* (toto nastavení znemožní změnu příkazem). Aktuální nastavení zjistíme příkazem:

```
Get-ExecutionPolicy
```

Nemáme zde prostor vysvětlovat možnosti a bezpečnostní dopady. Takže jednoduše, abychom mohli spustit lokální nepodepsaný skript, tak musíme zadat příkaz:

```
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned
```

Pokud chceme spouštění povolit pro celou skupinu serverů či na celou firmu, tak je výhodnější použít ***Group Policy***. Nastavení se nachází v položce:

```
Computer Configuration/Policies/Administrative Templates/Windows Components/Windows PowerShell/Turn on Script Execution
```

Abychom tuto položku měli k dispozici, tak musíme mít v systému (nebo v Central Store) nahraný soubor `PowerShellExecutionPolicy.admx`, který obsahuje tuto šablonu. Ten se nachází pouze na Windows Server 2008 R2 a ne třeba ve Windows 7 (ale je možno jej zkopírovat ze serveru nebo stáhnout z webu MS).

Načítání modulů a snap-in

V následujících příkladech budeme pracovat s ***ActiveDirectory*** a k tomu využijeme ***PowerShell modul*** od Microsoftu, který je součástí (mimo jiné) Windows Server 2008 R2. A také s ***Exchange Server 2007***, který obsahuje ***Snap-In pro PowerShell***. Abychom mohli použít ***cmdlety*** z modulů a snapinů v čistém PowerShellu (powershell.exe), tak je musíme nejprve zavést (načíst). Informace o načtených modulech a snapinech získáme pomocí příkazů:

```
Get-Module
Get-PSSnapin
```

Načtení ***ActiveDirectory modulu*** provedeme příkazem:

```
Import-Module ActiveDirectory
```

Načtení ***Exchange Server 2007 Snap-Inu*** provedeme příkazem:

```
Add-PSSnapin Microsoft.Exchange.Management.PowerShell.Admin
```

Plánování skriptů

Standardně se ***PowerShell Script*** ukládá do souboru s příponou **ps1**. Spustit jej můžeme pomocí příkazu (důležité je zadání cesty, i když to lze samozřejmě i jinak):

```
powershell c:\Skripty\skript.ps1
```

Pro automatické spouštění využijeme ***plánovač úloh*** z Windows (Task Scheduler). Ve ***Windows Server 2008*** přidáme na záložce *Action* událost *Start a program* a jako script dáme `powershell`, jako argument dáme `c:\Skripty\skript.ps1`. Na záložce *Triggers* přidáme plánovaný úkol a zadáme požadovanou periodu spouštění. Na záložce *General* nastavíme, pod jakým účtem skript poběží. Na ***Windows Server 2003*** je situace obdobná, pouze do políčka *Run* zadáme celý příkaz `powershell c:\Skripty\skript.ps1`.

Kontrola účtů v Active Directory

Nejen z bezpečnostního hlediska je dobré mít v Active Directory pořádek. Přitom některé informace se dozvíme pouze položením určitého dotazu. To můžeme automatizovat skriptem, spouštět pravidelně každou noc a na email si odeslat výsledek.

> ***Pozn.:*** *V prvním příkladu popíšeme celý skript, v dalších již budeme vynechávat načtení modulu na začátku a odeslání emailu na konci. Skript vždy uložíme do nějakého souboru s příponou ps1 na doménovém řadiči a budeme automatizovaně spouštět pomocí Task Scheduler pod účtem s dostatečným oprávněním (stačí čtení AD). Samozřejmě můžeme použít více skriptů najednou a odeslat jeden společný email.*

Neaktivní uživatelé

Například nás může zajímat seznam uživatelů, kteří se nepřihlásili nějakou definovanou dobu (zde 30 dní) do domény. Tím můžeme odhalit, že jsme nějaký účet zapomněli smazat nebo již přestal být potřeba.

Ve skriptu nejprve načteme modul. Druhý příkaz získá seznam uživatelských účtů, které se za posledních 30 dní nepřihlásili, ten seřadí podle abecedy, vrátí pouze vybrané sloupce a uloží do souboru. Poslední příkaz odešle informační email a do přílohy vloží vygenerovaný soubor.

```
Import-Module ActiveDirectory
Search-ADAccount -AccountInactive -UsersOnly -TimeSpan "30" | Sort-Object Name | FT Name,LastLogonDate,Enabled,LockedOut -Aut
Send-MailMessage -From ad@firma.cz -To monitoring@firma.cz -Subject "Kontrola uživatelů v AD" -SmtpServer mail.firma.local -A
```

Neaktivní počítače

Stejným způsobem můžeme nalézt počítače, které se 30 dní nepřihlásili.

```
Search-ADAccount -AccountInactive -ComputersOnly -TimeSpan "30" | Sort-Object Name | FT Name,LastLogonDate,Enabled,LockedOut
```

Zakázané účty uživatelů a počítačů

Můžeme vytvářet seznam počítačových a uživatelských účtů, které jsou zakázané (disabled). Tuto informaci asi nepotřebujeme každý den, ale můžeme ji pro přehled přidat k předchozím.

```
Search-ADAccount -AccountDisabled -UsersOnly | Sort-Object Name | FT Name -AutoSize | Out-File -FilePath "c:\Skripty\disabled

Search-ADAccount -AccountDisabled -ComputersOnly | Sort-Object Name | FT Name -AutoSize | Out-File -FilePath "c:\Skripty\disa
```

Zamčené uživatelské účty

Pokud máme politiku na zamykání účtů, tak nás může zajímat seznam zamčených účtů. Uživatelé by se asi ozvali, ale například se nemusíme dozvědět, že se zamknul nějaký servisní účet.

```
Search-ADAccount -LockedOut -UsersOnly | Sort-Object Name | FT Name -AutoSize | Out-File -FilePath "c:\Skripty\locked-users.t
```

Nové počítačové účty

Můžeme vytvářet report, který bude obsahovat seznam nových, smazaných nebo přejmenovaných počítačových účtů. Tyto informace můžeme získat i z auditování AD DS, ale takto dostaneme přehledný report.

Nejprve načteme předchozí stav (takže při prvním spuštění nám celý skript nezafunguje). Potom získáme seznam všech počítačů a seřazený jej uložíme do souboru. Tento seznam znovu načteme do jiné proměnné (kvůli typu a následnému porovnání). Porovnáme tyto dva seznamy a jejich rozdíl uložíme do souboru (tento soubor pak můžeme odeslat jako email). Ve výsledku vidíme nové záznamy (indicator =>) a chybějící záznamy (indicator <=), pokud došlo ke změně jména, tak bude zobrazeno jako odstraněný a nově vložený.

```
$yesterday = Get-Content c:\skripty\computers.txt
Get-ADComputer -Filter * | Sort-Object Name | FT Name -AutoSize | Out-File -FilePath c:\skripty\computers.txt
$today = Get-Content c:\skripty\computers.txt
Compare-Object $yesterday $today | FT -AutoSize | Out-File -FilePath c:\skripty\computers-changes.txt
```

Kontrola poštovních schránek na Exchange serveru

Tak jako informace z domény nás může zajímat řada informací na poštovním serveru, hlavní se týkají poštovních schránek (mailboxů).

> ***Pozn.:*** *Opět první skript bude kompletní a dále jen hlavní část. Skript bude v souboru s příponou ps1 a plánovat jej budeme přímo na Exchange serveru (nyní musí mít účet práva na Exchange).*

Poštovní schránky s překročeným limitem

Na velikost poštovní schránky se může nastavit několik limitů, při prvním se uživateli posílá upozornění, ale jinak může standardně pracovat. Pokud nás zajímá seznam schránek, které překročili tento limit, tak můžeme použít následující skript.

Nejprve načteme snapin. Potom vezmeme seznam mailboxů, odstraníme z něj ty, které mají velikost pod limitem, seřadíme podle velikosti a vybrané sloupce uložíme do souboru. Ten pak odešleme emailem.

```
Add-PSSnapin Microsoft.Exchange.Management.PowerShell.Admin
Get-MailboxStatistics | sort TotalItemSize -Descending | where { $_.StorageLimitStatus -ne "BelowLimit" } | FT DisplayName,To
Send-MailMessage -From exchange@firma.cz -To monitoring@firma.cz -Subject "Kontrola velikosti poštovních schránek" -SmtpServe
```

Veřejné složky s překročeným limitem

Obdobně můžeme kontrolovat velikost veřejných složek (Public Folders) a vytvořit seznam těch, které jsou větší než zadaná hodnota (zde cca 500MB). Jen je třeba si uvědomit, že podsložky ve veřejných složkách se počítají samostatně.

```
Get-PublicFolderStatistics | Sort TotalItemSize -Descending | where { $_.TotalItemSize -gt 500000000 } | FT Name,FolderPath,I
```

Přehled mobilních zařízení

Pokud používáme Exchange ActiveSync, tak se mobilní zařízení, která se serverem synchronizujeme, spárují s účtem a na serveru se o nich vytvoří záznam. K dispozici máme příkaz, který nám napíše několik informací. Uživatelé si mohou připojit různé přístroje a nás může zajímat, že přibylo nové zařízení či obecně přehled o nich.

Můžeme opět naplánovat úlohu, která zjistí seznam všech zařízení, a vytvoří rozdíl oproti předchozímu dni, takže vidíme změny.

```
$yesterday = Get-Content c:\skripty\ActiveSync.txt
Get-Mailbox | ForEach { Get-ActiveSyncDeviceStatistics -Mailbox:$_.Identity } | FT Identity,DeviceType,DeviceUserAgent,Device
$today = Get-Content c:\skripty\ActiveSync.txt
Compare-Object $yesterday $today | Out-File -FilePath c:\skripty\ActiveSync-differences.txt -Width 400
```

Kontrola komunikace

V řadě situací můžeme využít test, jestli přišel nějaký určitý email. Například máme vzdálený systém, který není přímo dostupný z internetu, ale může odesílat emaily. Necháme tedy jednou za hodinu posílat emailový keepalive a máme hrubou kontrolu, že systém žije.

```
$mails = Get-MessageTrackingLog -EventID "RECEIVE" -Sender "server@nekde.cz" -Recipients "monitor@firma.cz" -Start (Get-Date)
if($mails -eq $null) { Send-MailMessage -From dohled@firma.cz -To monitoring@firma.cz -Subject "Kontrolní email ze serveru" -
```

Systémové kontroly

Ještě se podíváme na dva příklady, které nespadají do předchozích kategorií. Navíc se jedná o malinko složitější skripty.

Kontrola zálohy

Pokud provádíme nějaké jednoduché zálohy, například dump z linuxového serveru, který v taru kopírujeme na souborový server s Windows, tak můžeme testovat, jestli se v danou dobu soubor vytvořil a jestli není jeho velikost nulová.

Do proměnné `$files` si uložíme seznam souborů (bez adresářů), které byly změněny (tedy i nově vznikly) za poslední den v adresáři `c:\backup` (mohli bychom hledat pouze soubory určitého jména). Potom porovnáme, jestli je v seznamu alespoň jeden soubor (nemůžeme použít jednodušší `$files.Count`, protože nefunguje na prázdný seznam). Pokud nějaké soubory existují, tak ověříme, jestli nemají nulovou velikost (můžeme testovat zadanou minimální velikost). Pokud došlo k problému, tak odešleme informaci emailem.

```
$message = ""
$files = Get-ChildItem c:\backup | Where-Object { $_.LastWriteTime -gt (Get-Date).AddDays(-1) -and ! $_.PSIsContainer }
if(($files | Measure-Object).Count -eq 0) {
   $message = "Za poslední den nevznikl žádný soubor."
} else {
   $files | ForEach-Object { if($_.Length -eq 0) { $message += "Soubor " + $_.name + " má nulovou velikost. " } }
}
if($message -ne "") { Send-MailMessage -From zalohovani@firma.cz -To monitoring@firma.cz -Subject "Zálohování selhalo" -SmtpS
```

Změny na DHCP serveru

Sice je řada možností, jak zabezpečit připojení "cizího" zařízení do sítě, ale dokonalé zabezpečení je většinou v praxi nemožné. Pro zvýšení našeho přehledu o zařízeních v síti si můžeme posílat report zařízení, která dostala IP adresu od DHCP. Není to myšleno jako metoda odhalení nepovolených zařízení v síti, také bychom mohli využít auditování DHCP serveru, je to pouze ukázka možností.

Ani v poslední verzi Windows serveru neobsahuje PowerShell cmdlety pro ovládání DHCP serveru. Nic nám ale nebrání využít z PowerShellu klasický příkaz `netsh` (v neinteraktivním režimu) a zpracovat jeho výsledky. Takže si vypíšeme seznam aktivních DHCP Scope, a z nich vybereme prvních 15 znaků (což je síťová IP adresa) a uložíme do proměnné. Potom pro každou adresu scopu vypíšeme seznam pronajatých adres (lease). Výsledek uložíme do souboru. Porovnáváme stav z předchozího dne s aktuálním a výsledkem jsou změny.

```
$leases_yesterday = Get-Content c:\skripty\dhcp.txt
$scopes = netsh dhcp server \\pdc show scope | Where-Object { $_.Contains("Active")  } | ForEach-Object { $_.Substring(0, 15)
$leases = $scopes | ForEach-Object { netsh dhcp server \\pdc scope $_.Trim() show clients 1 } | Where-Object { $_.Contains("-
$leases | Out-File -FilePath c:\skripty\dhcp.txt
Compare-Object $leases_yesterday $leases | FT -AutoSize | Out-File -FilePath d:\dhcp-changes.txt
```

Závěr

Pokud jste dočetli až sem, tak vám děkuji a doufám, že pro vás měl článek nějaký přínos. Jeho cílem bylo ukázat, že můžeme jednoduše získat přehled o tom, co se děje v našem prostředí. A nebude nás to stát příliš času ani peněz. Potom také, že PowerShell není špatná ani složitá technologie. Hlavní je rozmyslet, o čem v síti chceme být informováni. Sestavení skriptu již nemusí být složité a hodně nám pomůže Google. Samozřejmě příklady v článku jsou docela triviální, daly by se vylepšit, aby měly větší inteligenci a to by obnášelo trochu více kódu a složitosti.

Autor: Petr Bouška

Powershell - poslední přihlášení počítače či uživatele

Úterý, 06.09.2011 14:44 | Samuraj - Petr Bouška | Microsoft admin

V určitých situacích se nám může hodit, nalézt k počítačům či uživatelům v doméně informaci, kdy se naposledy přihlásili. Jednou z běžných situací je, že hledáme nepoužívané účty. Nebo naopak chceme nalézt, jak dlouho již počítač běží. Popíšeme si zde nějaké možnosti pomocí PowerShellu. První situace je relativně jednoduchá (pokud akceptujeme nějaké problémy) a informace získáme z Active Directory. V druhém případě již musíme přistupovat na jednotlivé počítače.

Když se podíváme na ***atributy počítačového účtu*** (které můžeme získat pomocí PowerShellu), tak vidíme řadu zajímavých hodnot.

```
Import-Module ActiveDirectory
Get-ADComputer -Identity <jmeno-pocitace> -Properties *
```

Počítačový nebo uživatelský účet v ***Active Directory*** má atribut `lastLogonTimestamp`. Tato hodnota říká, kdy se ***naposledy přihlásil*** počítač nebo uživatel do domény. Bohužel, ale tato hodnota není přesná pro (defaultně) hodnoty ***kratší 14 dnů***. Je to z principu fungování, vždy při přihlášení se hodnota přečte a zkontroluje, jestli není starší než `current_time - msDS-LogonTimeSyncInterval`, pokud ano, tak se aktualizuje. Defaultně je `msDS-LogonTimeSyncInterval` 14 dní mínus náhodné číslo do 5 dnů. Hodnota `lastLogonTimestamp` je replikovaná mezi doménovými řadiči.

Hodnota `msDS-LogonTimeSyncInterval` je nastavena na doméně a můžeme si ji prohlédnout/změnit pomocí ***Active Directory Users and Computers*** (pravým tlačítkem na jméno domény - *Properties - Attribute Editor*). Pokud je zde hodnota `not-set`, tak to znamená, že je to defaultních 14 dní. Pokud zadáme hodnotu menší než 5, tak se neodečítá náhodná hodnota, pokud zadáme 0, tak se vypne používání `lastLogontimeStamp`. Vše je takto nastaveno proto, aby se nereplikovalo příliš mnoho dat. V menších prostředích to ale není problém.

Pokud použijeme PowerShell příkaz `Get-ADComputer` (zajímavé je, že tento příkaz nám vrací i IP adresu, což není atribut počítačového účtu, patrně se berou údaje z DNS), tak můžeme vrátit hodnotu atributu `lastLogonTimestamp` typu `System.Int64` nebo převedenou podle aktuálního časového pásma na datum z atributu `LastLogonDate` (typ `System.DateTime`).

```
Get-ADComputer -Identity <jmeno-pocitace> -Properties * | FT Name, LastLogonDate, LastLogon, lastLogonTimestamp, IPv4Address
```

Pak máme ještě atribut `lastLogon`, ten obsahuje ***skutečný čas posledního přihlášení***, ale není replikovaný, takže bychom se museli zeptat všech DC a vzít nejnovější hodnotu. Je uložen také jako číslo a můžeme jej převést na datum pomocí funkce.

```
[DateTime]::FromFileTime([Int64]::Parse((Get-ADComputer -Identity <jmeno-pocitace> -Properties lastLogon).lastLogon))
```

Pokud chceme najít účty, které se ***delší dobu nepřihlásili***, tak nám většinou stačí hodnota `lastLogonTimestamp`, pokud nám stačí hodnoty starší 14 dní. Například následujícím způsobem můžeme hledat počítače, které se nepřihlásili déle než 30 dní.

```
$ldate = [DateTime]::Now - [TimeSpan]::Parse("30")
Get-ADComputer -Filter { lastLogonTimestamp -le $ldate } -Properties * | FT Name, LastLogonDate, Enabled, LockedOut -AutoSize
```

Pokud bychom dělali podrobnější pokusy, tak můžeme narazit na to, že některé účty se nám ve výstupu nezobrazí. Jsou to takové, které se nikdy nepřihlásili a jejich hodnota `lastLogonTimestamp` je prázdná, což asi nastane spíše u uživatelských účtů. Nevím proč, ale ve filtru nelze porovnávat `null` hodnotu, takže musíme předchozí příkaz trochu přepsat.

```
$ldate = ([DateTime]::Now - [TimeSpan]::Parse("30")).ToFileTime()
Get-ADComputer -Filter * -Properties * | Where-Object { ($_.lastLogonTimestamp -le $ldate) -or ($_.lastLogonTimestamp -eq $nu
```

Máme tady ještě jednodušší možnost pomocí příkazu `Search-ADAccount`, ale tento příkaz nám nevrátí všechny atributy.

```
Search-ADAccount -ComputersOnly -AccountInactive -TimeSpan "30" | FT Name, LastLogonDate, Enabled, LockedOut -AutoSize
```

Pokud chceme najít čas, kdy ***naposledy počítač nabootoval***, tak nemůžeme použít ani hodnotu `lastLogonTimestamp` ani `lastLogon`, protože k autentizaci počítače dochází i za jeho běhu. Hodnota, kdy naposledy nastartoval počítač, se asi nachází pouze na něm. Můžeme ji získat třeba pomocí ***WMI***.

```
[System.DateTime]::ParseExact((Get-WmiObject -Class Win32_OperatingSystem -ComputerName <jmeno-pocitace>).LastBootUpTime.spli
```

Můžeme pak například udělat test, který prochází počítače z daného kontejneru v AD, otestuje, zda je počítač zapnutý, a pokud ano, tak zjistí datum posledního startu.

```
Get-ADComputer -Filter * -SearchBase "OU=Pocitace,DC=firma,DC=local" |
  ForEach-Object {
    $str = $_.Name + ", "
    if(Test-Connection -ComputerName $_.Name -Count 1 -Quiet) {
      $str += "running, "
      $str += [System.DateTime]::ParseExact((Get-WmiObject -Class Win32_OperatingSystem -ComputerName $_.Name).LastBootUpTime
    }
    $str
  }
```


Autor: Petr Bouška

Související články:

PowerShell

Články týkající se Microsoft skriptovacího jazyku PowerShell, který se používá ve všech nových verzích MS OS a aplikací.

- PowerShell - Active Directory [14.11.2010 13:13]
- PowerShell - Exchange server [19.11.2010 08:45]
- PowerShell - Windows [12.12.2010 18:00]
- PowerShell - Ovládání aplikací a další použití [09.01.2011 18:59]
- Exchange Web Services a PowerShell [13.01.2011 11:00]
- Active Directory Recycle Bin [31.03.2011 15:48]
- Automatizovaná kontrola prostředí pomocí PowerShellu [13.06.2011 17:35]
- Powershell - poslední přihlášení počítače či uživatele [06.09.2011 14:44] ..... **právě čtete**
- Exchange Server, Outlook a certifikáty v GALu [27.12.2011 14:48]

Exchange Server, Outlook a certifikáty v GALu

Úterý, 27.12.2011 14:48 | Samuraj - Petr Bouška | Microsoft admin

Článek se věnuje ukládání šifrovacích certifikátů do Active Directory. Jak k nim potom přistupují klienti jako je Outlook nebo OWA. Jaké mohou nastat problémy, když přejdeme na novou certifikační autoritu, a klientům se z AD nabízí stále staré certifikáty. A také různým PowerShell příkazům, pomocí kterých můžeme s certifikáty v AD pracovat.

Pokud používáme ***Microsoft Exchange server*** a ***Outlook*** a využíváme ***šifrování emailových zpráv***, tak je dobré umístit šifrovací certifikáty jednotlivých uživatelů do ***Active Directory*** (AD). Potom jsou tyto certifikáty (zjednodušeně řečeno) automaticky distribuovány všem uživatelům jako součást ***GALu*** (Global Address List). Takže si uživatelé mohou navzájem posílat šifrované zprávy.

Publikování certifikátu do AD

Certifikáty se v AD ukládají do atributu u objektu uživatele. Umístit certifikát k uživateli (Publish certificate in AD) můžeme několika způsoby:

- **při vydání certifikátu** - pokud využíváme MS CA, tak na šabloně můžeme nastavit, aby se certifikát automaticky publikoval do AD
- **pomocí Outlooku** - uživatelé si mohou vypublikovat nastavený šifrovací certifikát sami v Outlooku (na stejném místě, kde se certifikát nastavuje pro použití), v Outlooku 2010 to je *File - Options - Trust Center - Trust Center Settings* - tlačítko *Publish to GAL*
- **pomocí ADUC** - když si zobrazíme vlastnosti uživatelského účtu pomocí *Active Directory Users and Computers* (ADUC) a máme zapnuté zobrazení *Advanced Features*, tak vidíme záložku *Published Certificates*, zde můžeme přidávat i mazat certifikáty
- **další možnosti** - pomocí nějakých příkazů jako *certutil.exe*, *PowerShell* případně i s rozšířením od firmy *Quest* pro AD

Problém s nabízeným certifikátem

Všechno by se zdálo jednoduché a jasné, hlavně dokud nám vše pěkně funguje. Jak už to ale u Microsoftu bývá, tak je skutečnost o dost složitější. Nedávno jsem narazil na problém, kdy jsme přešli na novou certifikační autoritu. Takže uživatelům bylo třeba vydat nové certifikáty, ty se vypublikovaly do GALu (automaticky pomocí CA). Jenže když se takovému uživateli poslal email, tak se pořád šifroval jeho starým certifikátem. Vymazal jsem tedy v AD staré certifikáty, ale to nepomohlo. Vymazal jsem všechny certifikáty, ale pořad Outlook nabízel starý certifikát. Vyzkoušel jsem všechny různé možností, které jsem popisoval v článku Šifrovaní emailů v MS Outlook a řešení problémů. Ale stále nic, až jsem nakonec situaci vyřešil, když jsem se podíval na detaily uživatelského objektu pomocí PowerShell příkazu Get-ADUser. Potom se mi podařilo dohledat i dokumentaci, kde jsem se dozvěděl, proč celá situace nastala.

Jak se certifikáty v AD ukládají

Jde o to, že certifikáty se ukládají do atributu `userCertificate`, to je často zmiňovaný atribut. Certifikáty jsou zde uloženy ve formátu ***DER Encoded X.509 v3*** a tyto certifikáty se zobrazují v ADUC na záložce ***Published Certificates***. Jenže certifikáty se také ukládají do dalšího atributu `userSMIMECertificate`, kde jsou pro změnu uloženy ve formátu ***PKCS #7*** a to včetně celého řetězu certifikátů autorit.

Potom, když chceme použít certifikát v nějakém klientovi, tak záleží na klientovi, jaký atribut použije. Například ***MS Outlook*** se podívá do atributu ***userSMIMECertificate*** a pokud certifikát nenalezne, tak potom do atributu ***userCertificate***. A aby to nebylo jednoduché, tak ***Outlook Web Access*** (OWA) to dělá přesně opačně.

Takže když smažeme staré certifikáty u uživatele pomocí ADUC, tak Outlook je bude stále používat (a ještě nastane ta zajímavá situace, že přes OWA se použije ten nový certifikát). Pokud použijeme v Oulooku funkci ***Publish to GAL***, tak se certifikáty umístí do ***obou atributů***. A samozřejmě se doplňují a ne přepisují. Ale pokud využijeme publikování certifikátu do AD při ***vystavení certifikátu***, tak se uloží pouze do atributu ***userCertificate***.

Microsoft sám uvádí ve svém článku, používejte pouze jeden z těchto dvou atributů a určitě nechcete používat *userSMIMECertificate*. Doporučuje také zabránit uživatelům v publikování certifikátů z Outlook (pak se totiž použije tento atribut). Trochu zvláštní tvrzení ...

Mazání certifikátů v AD

Takže v řadě praktických situací potřebujeme smazat certifikáty z atributu **userSMIMECertificate**. Pro ruční operaci můžeme použít ***ADUC*** (z RSAT), kde ve vlastnostech uživatele máme záložku ***Attribute Editor***. Tam jednoduše nalezneme libovolný atribut a můžeme jeho hodnotu smazat. Pokud to potřebujeme udělat pro celou firmu, tak to není moc praktické a jako jasná volba by se stavěl ***PowerShell***.

V ***PowerShellu*** (rozšířeném o MS modul AD) můžeme použít třeba následující příkaz, který nám zobrazí informace o certifikátech. Ten ale bohužel opět pracuje jen s atributem ***userCertificate***.

```
Import-Module ActiveDirectory
(Get-ADUser username -Properties Certificates).Certificates
```

Můžeme se ale podívat přímo na atributy.

```
Get-ADUser username -Properties userCertificate, userSMIMECertificate | FL Name, userCertificate, userSMIMECertificate
```

A obsah atributu ***userSMIMECertificate*** můžeme **smazat** níže uvedeným příkazem. Druhý řádek ukazuje možnost smazání pro všechny uživatele v daném kontejneru.

```
Set-ADUser username -Clear userSMIMECertificate
Get-ADUser -Filter * -SearchBase "OU=Firma,DC=firma,DC=local" | Set-ADUser -Clear userSMIMECertificate
```

Outlook 2010 - zablokování tlačítka Publish to GAL

Zrušení tlačítka ***Publish to GAL*** můžeme provést jednoduše a elegantně pomocí ***Group Policy***. Potřebujeme k tomu, ale šablony pro Office 2010 (nebo odpovídající verzi), toto téma jsem popisoval v článku **MS Outlook a konfigurace pomocí Group Policy**.

Nastavení se nachází v cestě `User Configuration\Administrative Templates\Microsoft Outlook 2010\Security\Cryptography` pod položkou `Do not display 'Publish to GAL' button`.

Odstranění neplatných certifikátů z AD

Když se již bavíme o tématu certifikátů v GALu, tak určitě všichni po nějaké době narazíme na to, že nám Exchange server bude logovat varování, že se u nějakého uživatele nachází certifikát nevalidní nebo s prošlou dobou platnosti a že nebude přidán do GALu. Abychom se těmto varováním vyhnuli, tak by se hodilo automatizovaně promazávat neplatné certifikáty. Pokud to pro nás neudělá certifikační autorita, tak si musíme pomoci nějakým skriptem.

Určitě by se dal v PowerShellu s modulem pro AD napsat nějaký skript, který by využil následující informace, ale nebylo by to nic jednoduchého.

```
(Get-ADUser username -Properties Certificates).Certificates[0].GetExpirationDateString()
(Get-ADUser username -Properties Certificates).Certificates | foreach {New-Object System.Security.Cryptography.X509Certificat
```

Naštěstí je tu firma **Quest** a jejich rozšíření PowerShellu **ActiveRoles Management Shell for Active Directory**. Bohužel ale i jejich příkaz pracuje pouze s atributem ***userCertificate***. Následuje několik jednoduchých příkladů, jak smazat prošlý certifikát uživatele, nevalidní, pokud je v názvu vydavatele slovo Firma nebo všechny certifikáty všech uživatelů.

```
Add-PSSnapin Quest.ActiveRoles.ADManagement


Get-QADUser username | Remove-QADCertificate -Expired
Get-QADUser username | Remove-QADCertificate -Valid:$false
Get-QADUser username | Remove-QADCertificate -IssuerDN '*Firma*'
Get-QADUser | Remove-QADCertificate
```