Administrowanie systemami Windows

- Wykład

Administrator - zadania (I)

- Instalacja sprzętu
- Instalacja oprogramowania
  - System operacyjny,
  - Programy użytkowe,
  - Aplikacje internetowe,
  - Serwery usług.
- Planowanie
  - Konta użytkowników (np. loginy, grupy),
  - Alokacja zasobów (systemy plików, składy danych, urządzenia usługowe),
  - Usługi systemów,
  - Struktura sprzętu, nazewnictwo.

Administrator - zadania (II)

- Kopie bezpieczeństwa i odtwarzanie danych z kopii
- Kontrola startu i zatrzymania systemu
- Konfiguracja sieci
- Zarządzanie uprawnieniami użytkowników
- Monitorowanie wydajności systemu
- Projektowanie rozwoju systemu (szacowanie potrzeb, propozycje zakupów)
- Szkolenia użytkowników

Struktura administracji systemu

- Zależna od wielkości organizacji w której pracuje system:
  - Jeden komputer (personal use)
    - użytkownik = administrator
  - Mała firma
    - jeden z użytkowników to administrator
  - Średnia firma
    - część użytkowników to administratorzy, często przypisani do określonych obiektów wymagających administrowania , posiadający koordynatora
  - Duża firma
    - administratorzy to odrębna grupa z własną hierarchią

Zapewnienie bezpieczeństwa

- Długoterminowa polityka
- Jasny, sztywny i szczegółowy system uprawnień i procedur postępowania nie tolerujący żadnych wyjątków
- Kontrola skuteczności, natychmiastowe lecz udokumentowane wdrożenia koniecznych zmian polityki
- Angażowanie do analiz wszystkich działów organizacji (nie tylko IT)

Identyfikacja zagrożeń ze strony ludzi

- Bezpośrednio zagrożenie mogą powodować:
  - Dzieci, studenci, uczniowie - dla zabawy,
  - Konkurencja,
  - Pracownicy własnej firmy - zaniedbania, korupcja, szantaż z zewnątrz,
  - Przestępcy - kierowani chęcią zysku (przestępstwa bankowe, kradzieże danych itp.)

Źródła wzrostu zagrożeń

- Rosnąca przepustowość sieci - utrudnienie monitorowania treści
- Rosnąca anonimowość - utrudnienie monitorowania aktorów użytkujących zasoby sieciowe
- Rosnąca liczba użytkowników - razem z nią potencjalnych sprawców wykroczeń

Techniczne powody zagrożeń

- Wyścig dostawców - wypuszczanie na rynek niedostatecznie przetestowanych produktów
- Open source - powszechne udostępnienie kodu usług do poszukiwania słabych ogniw systemu
- Rosnąca ilość usług i protokołów - utrudnienie kontroli dozwolonych dla użytkowników operacji

Konsekwencje zagrożeń wobec danych (I)

- W konsekwencji włamania (przestępstwo), przypadku (błąd człowieka) czy awarii sprzętu może nastąpić:
  - Skasowanie danych - trwała utrata bieżących i aktualnych danych
  - Przecieki - wypływ danych zastrzeżonych na zewnątrz organizacji (na jej szkodę)
  - Przekłamanie danych:
    - Jawne - organizacja wie, że doszło do przkłamania danych lecz nie jest w stanie przywrócić ich poprawnej treści
    - Utajnione - organizacja nie wie, że dane w systemie zostały sfałszowane

Konsekwencje zagrożeń wobec danych (II)

- Trwała utrata danych:
  - W konsekwencji dodatkowe koszty odtworzenia danych lub w skrajnym przypadku paraliż prac organizacji.
- Przekłamanie danych
  - W konsekwencji dodatkowe koszty lokalizacji błędu i korekty danych a w przypadku przekłamania utajnionego koszty np. błędnych decyzji organizacji.
- Przecieki danych
  - W konsekwencji koszty prawne (odszkodowania), marketingowe (utrata zaufania klientów, technologiczne (gdy na wycieku korzysta konkurencja).

Mechanizmy ochrony zasobów

- Identyfikacja użytkowników
- Kopie bezpieczeństwa
- Ograniczone usługi (szczególnie świadczone zdalnie)
- Uprawnienia do zasobów
- Audyty

Jednostkowe obiekty administrowania

- Niezależna maszyna (komputer) - odizolowana lub pracujące jako klient usług sieciowych
- Komputer świadczący usługi sieciowe - serwer
- Struktura sieci lokalna
- Sieć rozległa - pomosty pomiędzy sieciami lokalnym, wyjście na świat z sieci danej organizacji

Cechy dobrego administratora

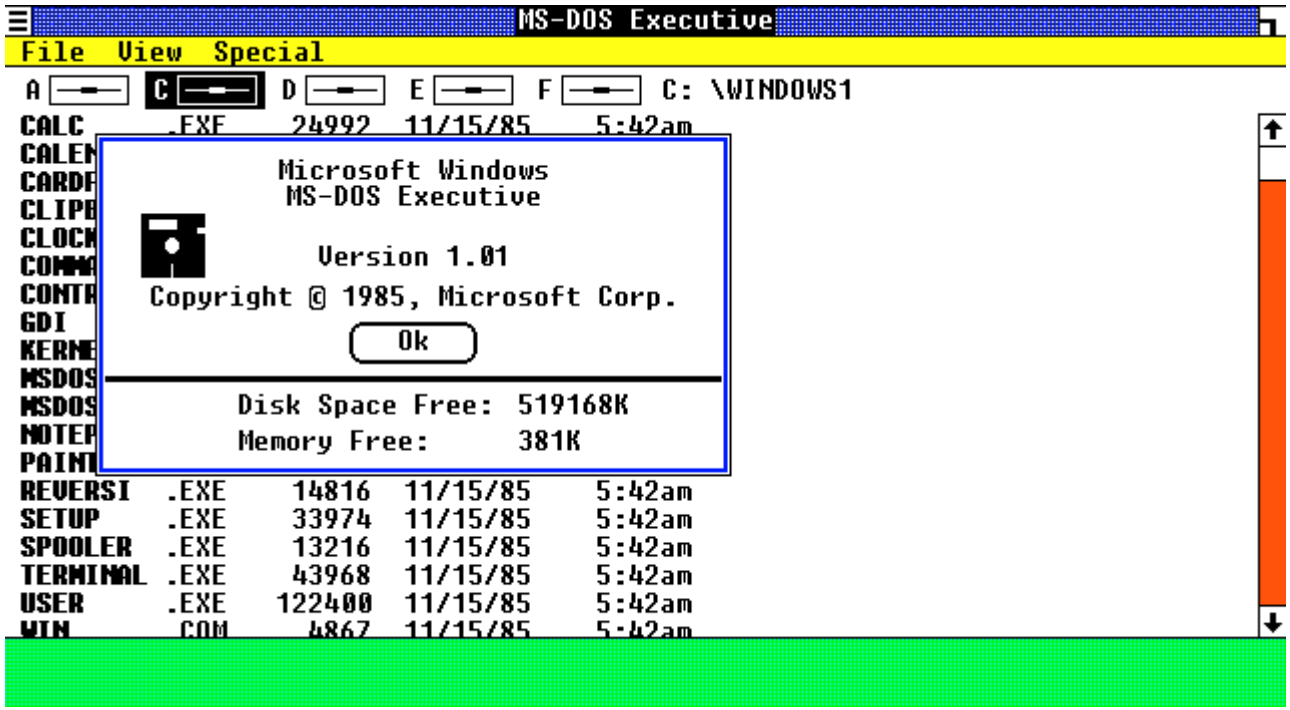
- Wiedza o systemie
- Zaufanie organizacji zatrudniającej
- Dyspozycyjność
- Cierpliwość
- Znajomość języka angielskiego, także w zagadnieniach technicznych

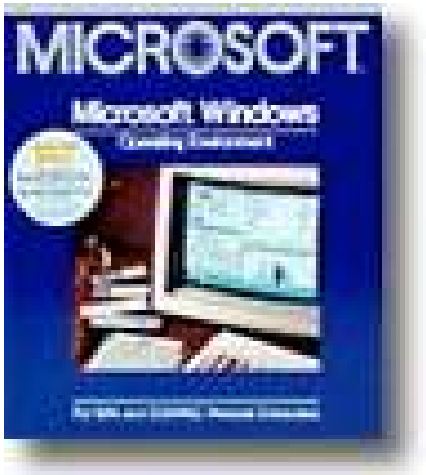
Materiały dla administratora

- Internet
  - Tematyczne grupy dyskusyjne
  - Strony producentów oprogramowania
  - Projekty opensource
  - Kursy w formie publikowanych w sieci materiałów
  - Banki przykładów i zasobów systemowych (typu TechNet)
- Instrukcje/manuale
  - Drukowane, dołączone do systemu
  - Elektroniczne (np. man, winhelp)
- Szkolenia i bezpośrednia wymiana informacji
- Książki, czasopisma

Systemy Windows (I)

- Windows 1.0 - 1985
- Windows 2.0 - 1987



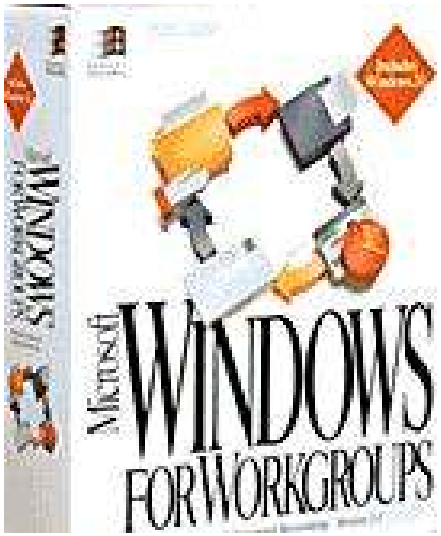


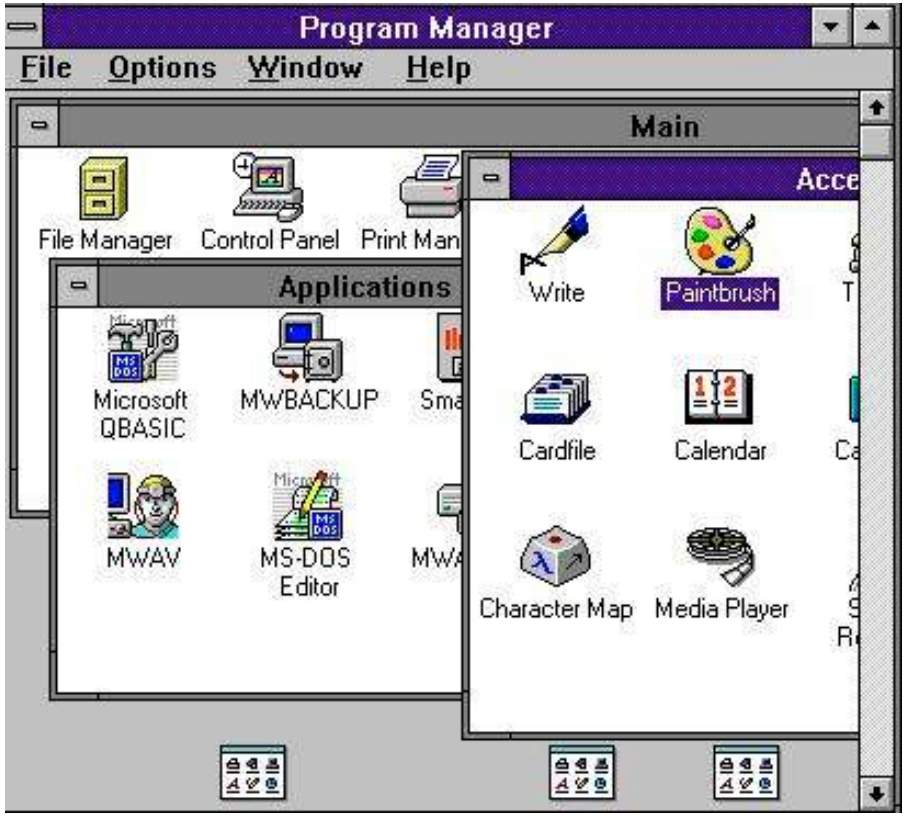
Systemy Windows (II)

- Windows 286

Systemy Windows (III)

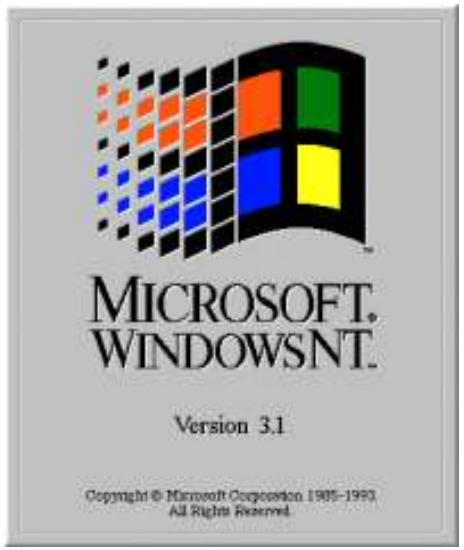
- Windows 3.0 - 1990, później 3.1, 3.11, Windows for Workgroups

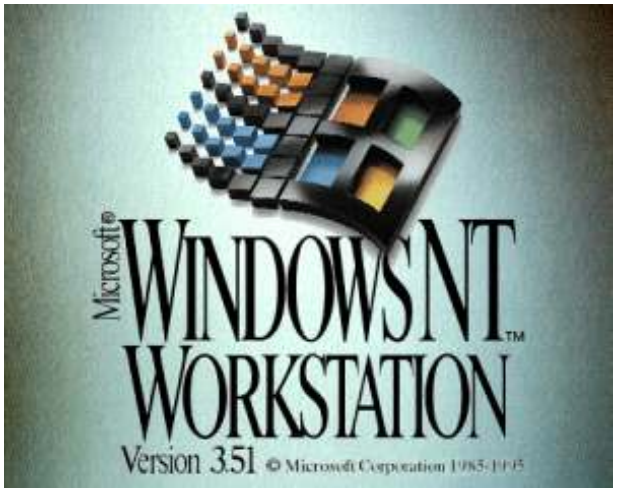


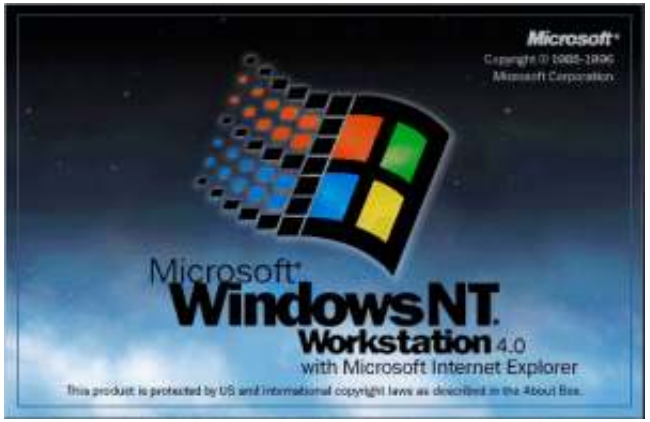


Systemy Windows (IV)

- Windows NT - od 1993
  - Wersje 3.1, 3.51, 4.0, 5.0
  - Przeznaczenie na serwery usług sieciowych
  - Interfejs graficzny podąża za linią 3.1/95

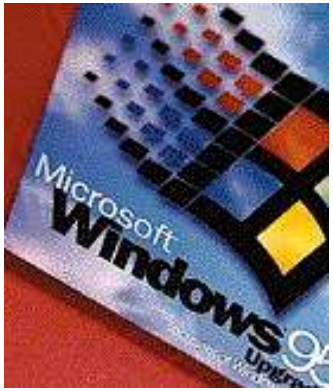


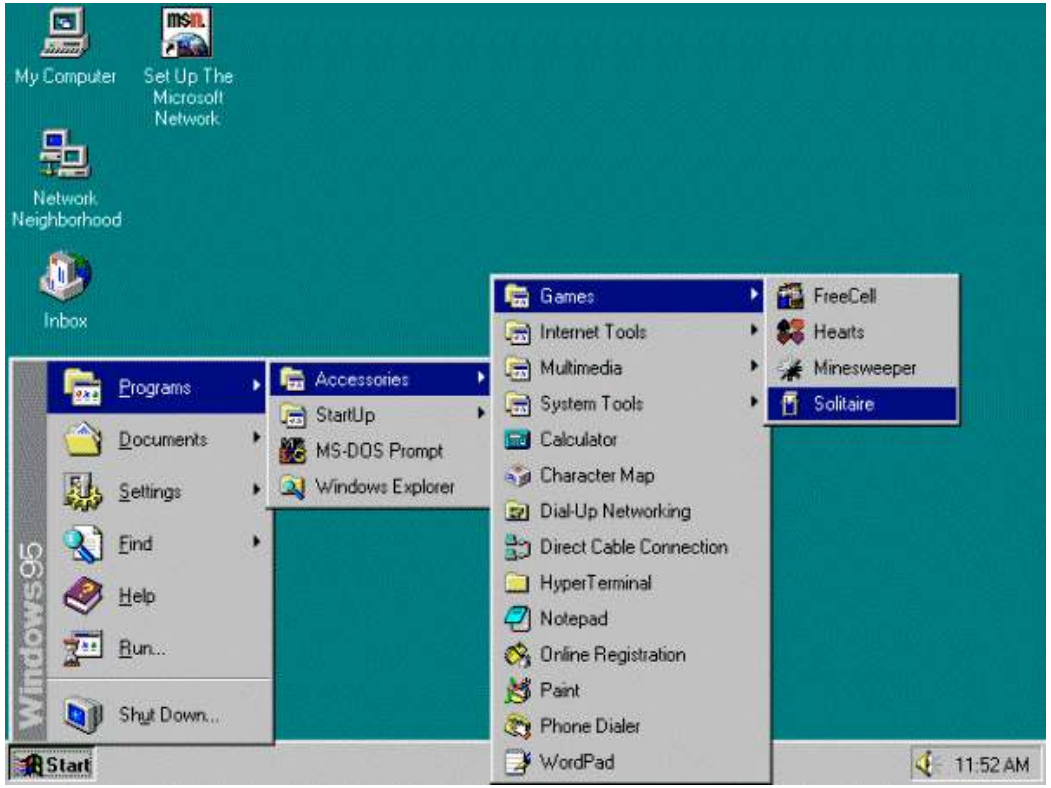




Systemy Windows (V)

- Windows 95/98 - od 1995





Systemy Windows (VI)

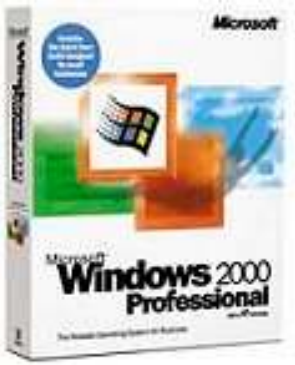
- Windows CE
  - Pierwsza edycja na drobne urządzenia przenośne
  - Wersja 1.0 - 1996
  - Interfejs użytkownika analogiczny do Win95, oczywiście znacznie okrojony

Systemy Windows (VII)

- Windows ME
  - Rok 2000
  - *Millenium Edition*
  - Skierowany do domowych użytkowników
  - Rozbudowana platforma multimediów

Systemy Windows (VIII)

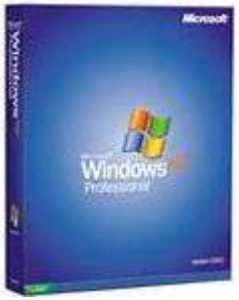
- Windows 2000 Professional
  - Inaczej: Windows NT 5.0 w nowej oprawie
  - Rozbudowana platforma internetowa
  - Wirtualne sieci prywatne
  - Zarządzanie aplikacjami za pośrednictwem Active Directory
  - Instalacje sieciowe aplikacji



Systemy Windows (IX)

- Windows 2000 Server
  - Wersje: Server, Advanced Server, Datacenter Server
  - Zaprojektowany z myślą o udostępnianiu plików, usług wydruku i aplikacji ogólnego zastosowania
  - Zaawansowana funkcjonalność: grupowanie przetwarzania (*Clustering*) przy przetwarzaniu symetrycznym,równoważenie obciążenia sieciowego, sprzętowego i programowego (*Load Ballancing*)

Systemy Windows (X)



- Windows XP
  - Kontynuacja linii *Windows 9x/Me* oraz *NT/2000*
  - Obsługa domen umożliwiające korzystanie podłączanie komputerów do serwera pełniącego funkcję *kontrolera domeny*
  - Wbudowany serwer *IIS (*ang. *Internet Information Services*),
  - System bezpieczeństwa plików bazujący na Win2000 i rozbudowany o procedury automatycznego raportowania błędów
  - Szyfrowany system plików (ang. *Encryptet File System*) - kodowanie każdego pliku przy użyciu losowo generowanego klucza
  - Zdalny dostęp do systemu

Systemy Windows (XI)

- Windows XP Home Edition
  - Okrojona wersja Win XP Pro
  - Brak obsługi *Zdalnego dostępu* do komputera,
  - Brak możliwości szyfrowania plików i folderów z wykorzystaniem mechanizmu *EFS*,
  - Brak obsługi domen — komputer z tym systemem nie może być klientem serwera *Active Directory*,
  - Brak możliwości zainstalowania kilku kart sieciowych,
  - Obsługa maksymalnie 5 równoczesnych połączeń,
  - Obsługa tylko jednego procesora

Systemy Windows (XII)

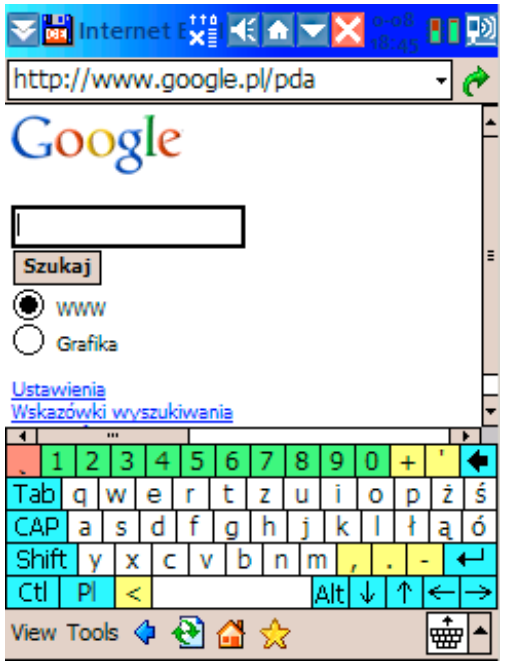
- Windows 2003 Server
  - Wersja: Web Edition, Standard Edition, Enterprise Edition
  - Kontynuacja linii 2000 Server
  - Nowości:
    - Dalsza rozbudowa platformy .NET i symetrycznego wieloprocesorowego przetwarzania danych
    - Usługi metakatalogowe Microsoft (MMS) - synchronizacja informacji o tożsamości i zasobach w całej organizacji.

Systemy Windows (XIII)

- Windows Mobile 2002, 2003 for Pocket PC
  - Następca Windows CE
  - Kompilacje na różne procesory
  - Funkcjonalność zbliżona do systemów desktop
  - Pocket Office, IE, WMPlayer
  - Porty większości popularnych aplikacji pod Windows

Systemy Windows (XIV)

- Windows Mobile 2005 for Pocket PC
  - Wersja z poprawkami
  - Lepsza ochrona systemu (antywirusowa!)
  - Ulepszone standardowe aplikacje - scheduler, mailbox itp.
  - Aktualizacje mobilne systemu
  - Wyrzucenie z RAM'u systemu plików
  - Obracany ekran



Systemy Windows (XV)

- Windows Vista (pierwotnie: Longhorn)
  - Okna renderowane na akceleratorze 3D (interfejs pod nazwą Aero)
  - Całkowicie nowe API do programowania aplikacji (WinFX)
  - Powszechne zastosowanie XML'a w formatowaniu dokumentów, konfiguracji, zarządzaniu systemem
  - Nakładka na system NTFS - system archiwizacji i indeksowania plików (WinFS). Na razie nie pojawił się.

Systemy Windows (XVI)

- Windows Vista (Beta1):
  - Silnik graficzny DCE (Desktop Composite Engine) zarządzający oknami
  - Planowane Wersje:
    - Starter Edition,
    - Home Basic Edition,
    - Home Basic Premium Edition,
    - Small Business Edition,
    - Enterprise Edition,
    - Ultimate Edition (Limited)

Nowości Vista (Beta2)

- Okienkowy i wielozadaniowy interfejs instalatora (DVD), a w nim:
  - cmd
  - Startup Repair - automatyczne odtwarzanie boot-recordu
  - System Restore - automatyczne odtwrzanie systemu
  - Windows Backup Disaster Recovery - odtwarzanie partycji z kopii zapasowej
  - Memory Diagnostics Tool

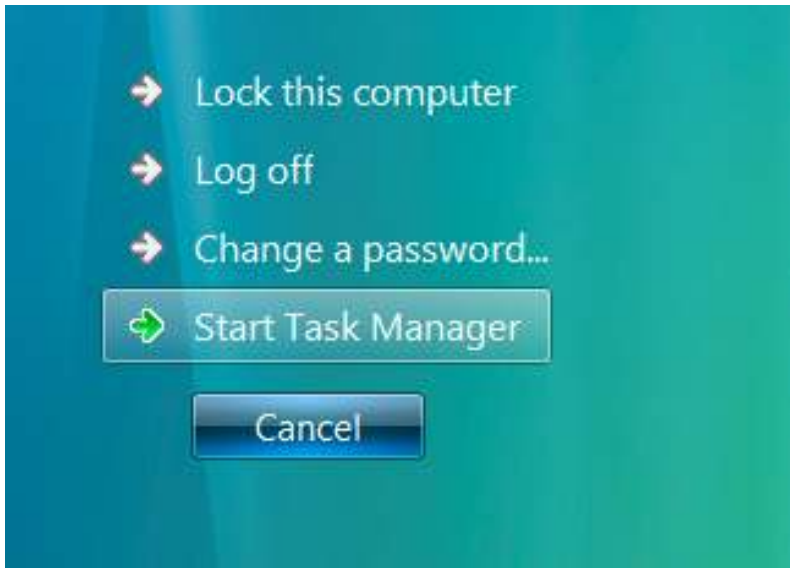
Nowości Vista

- Czas instalacji: 2:30, z czego zależne od sprzętu jest około 25 min
- Problemy z odświeżaniem kursora myszy i akceleratorem 2D
- Struktura katalogów:
  - Windows
  - Users, w nim: Administrator, Public, katalogi użytkowników
  - Documents and Settings
  - Program Files
  - Program Data
  - Boot
  - System Volume Information

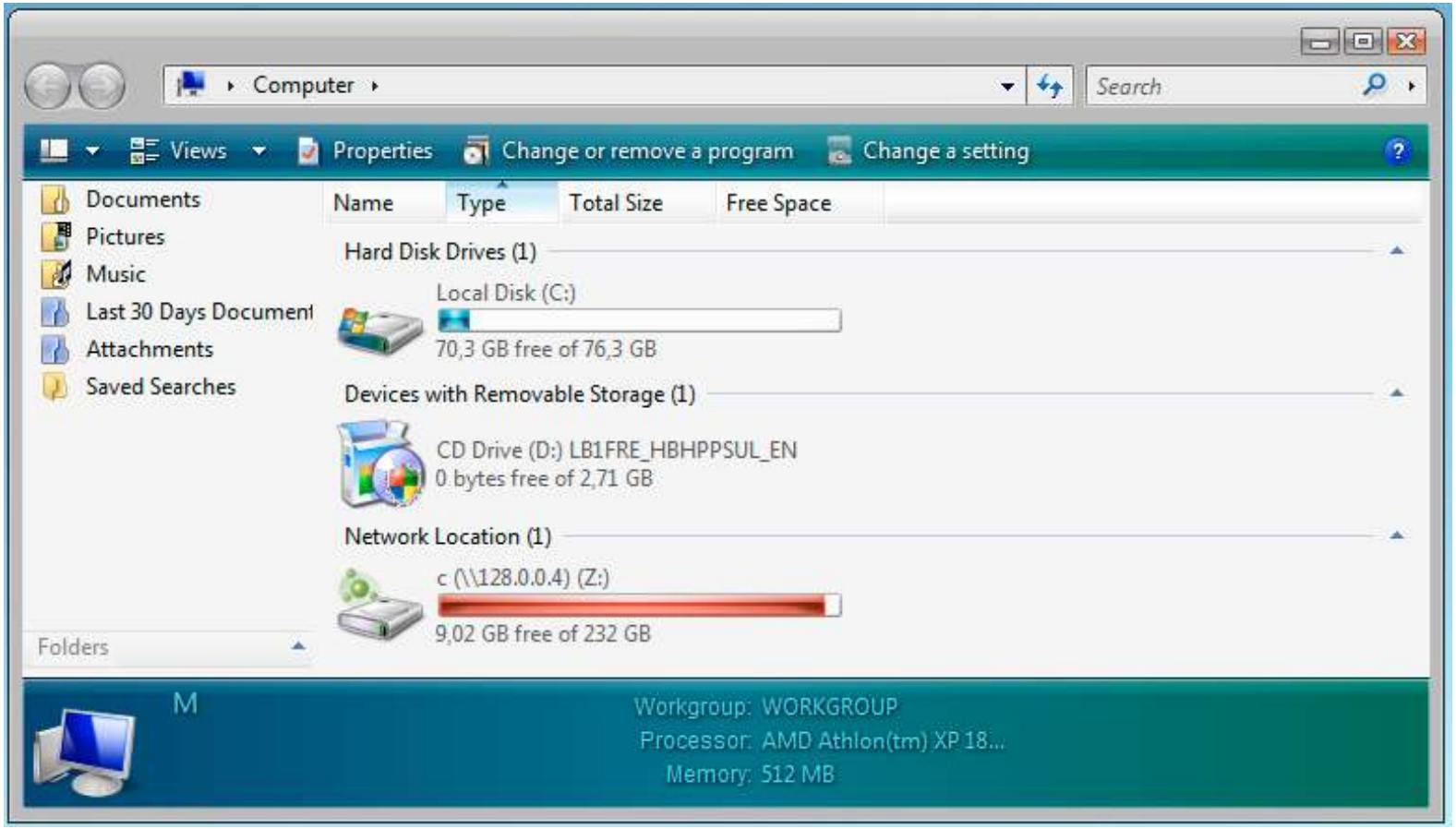
Nowości - Vista

- Wersja instalacyjna: 3.2 GB (DVD)
- System po zainstalowaniu: 5.9 GB
- Czas startu: 2:40 (Athlon 2 GHz, SATA 100)
- Czas startu po Hibernacji: 1:30 (nie zaalokowane 512 MB)

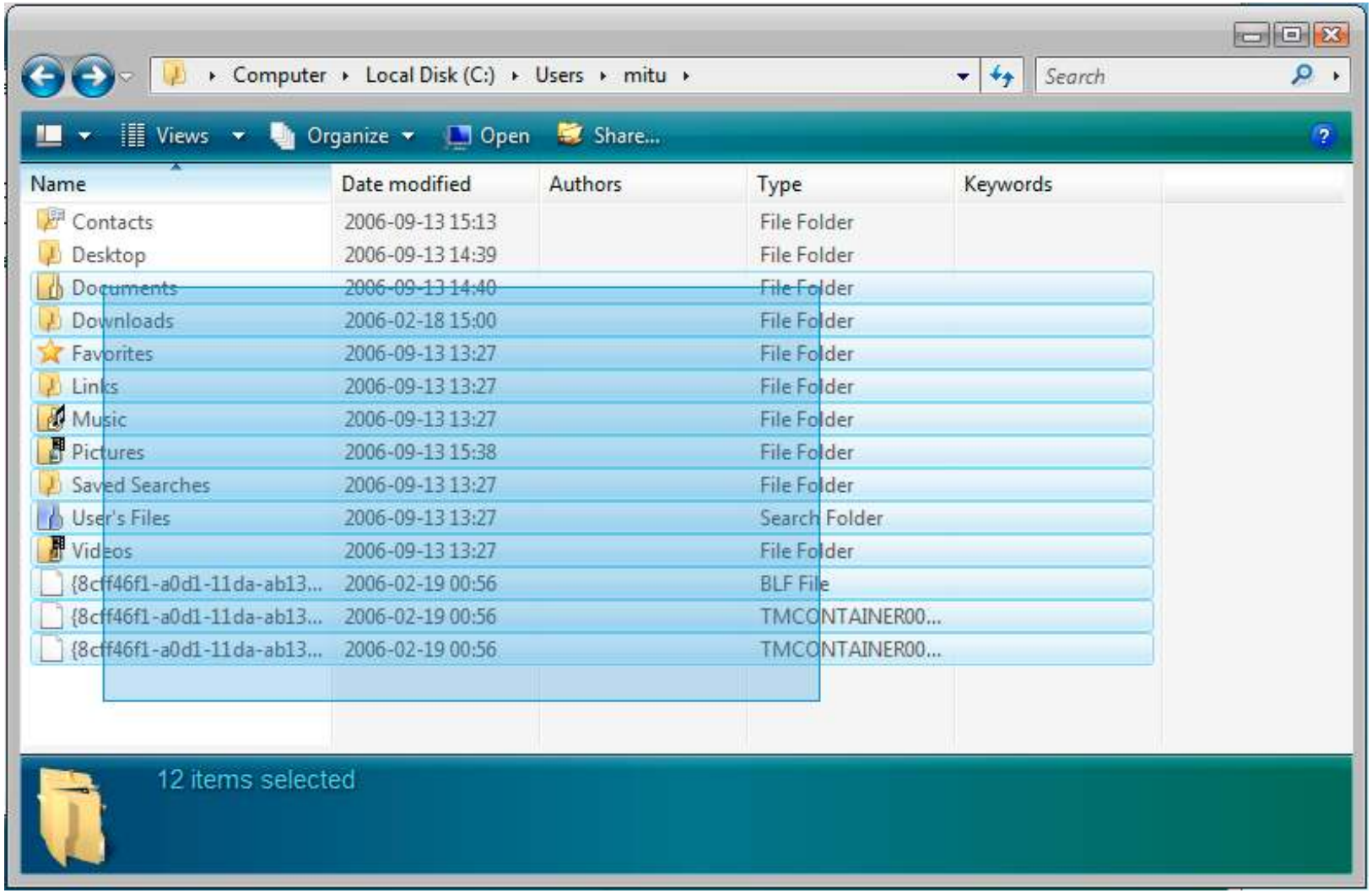
Nowości Vista - Start i zakończenie



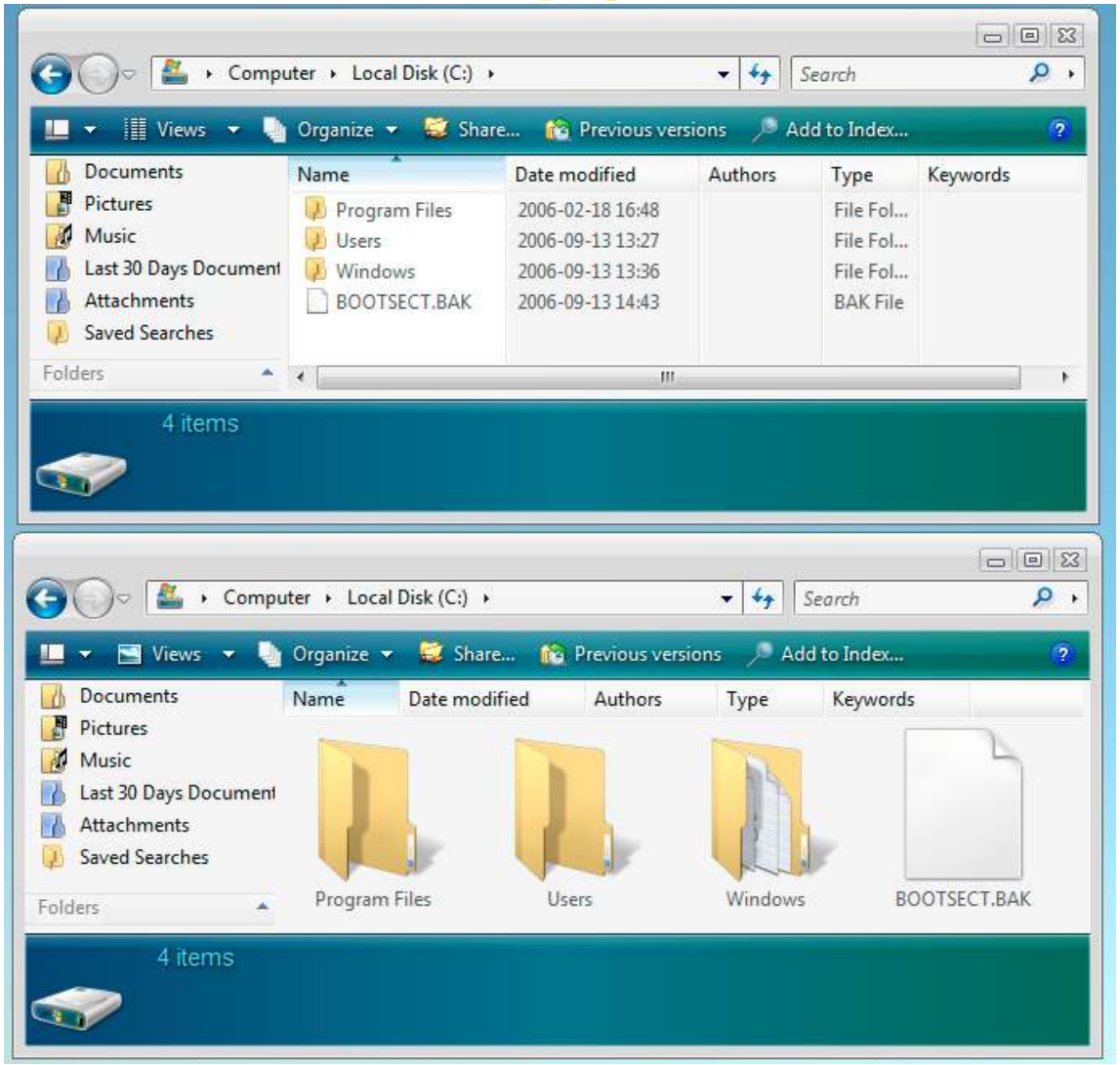
Nowości Vista - Foldery (I)



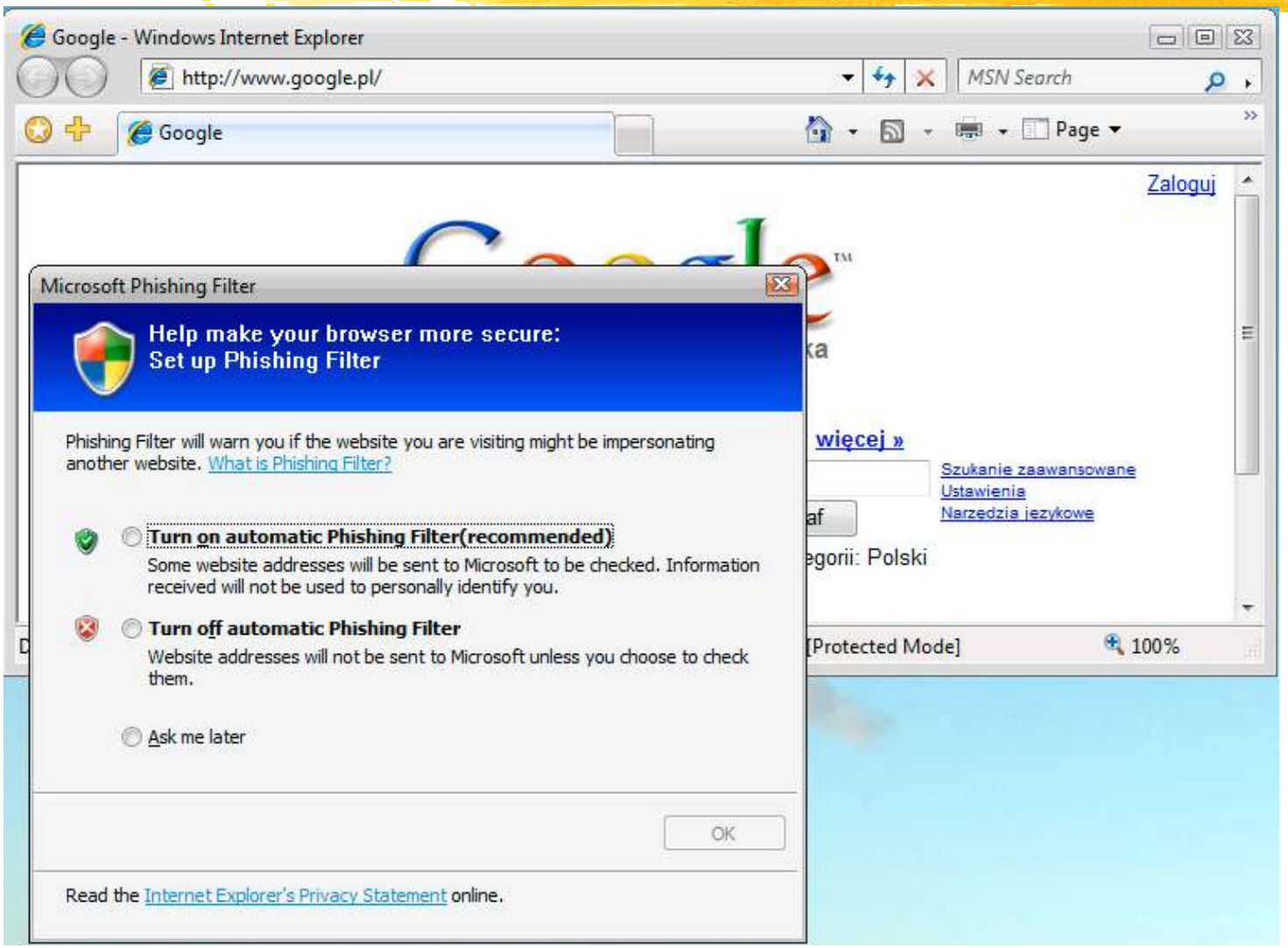
Nowości Vista - Foldery (II)



Nowości Vista - Foldery (III)

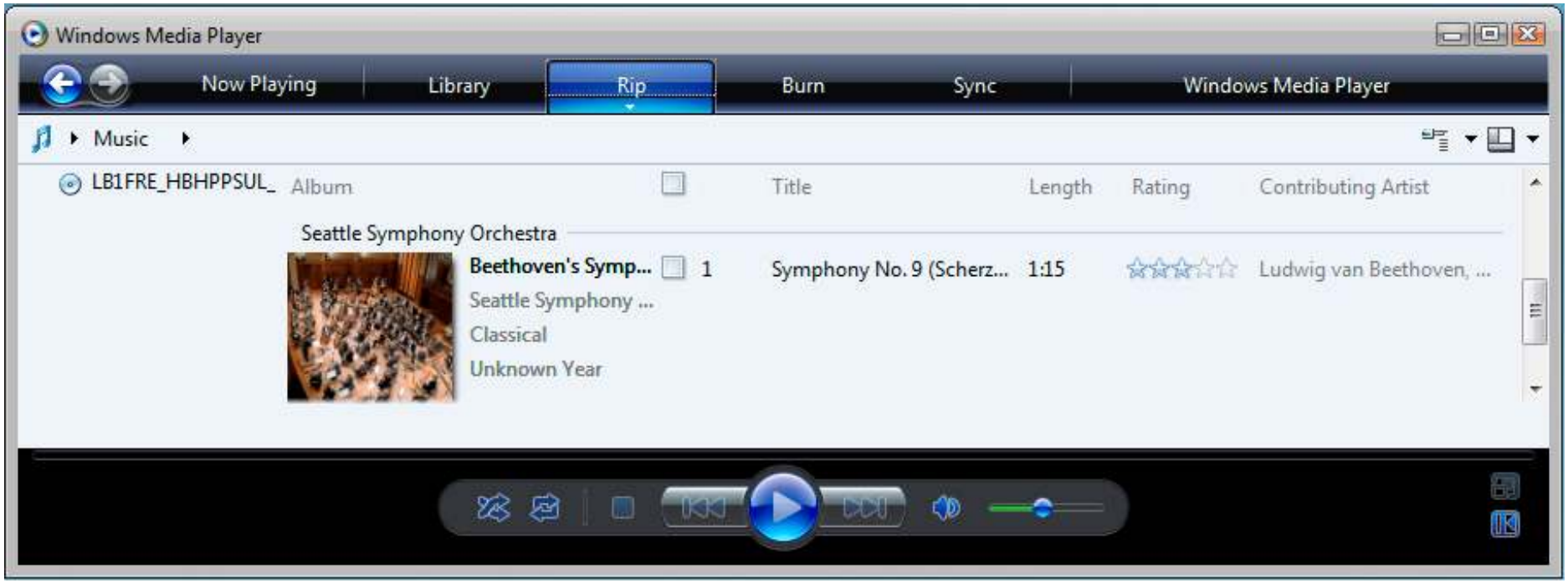


Nowości Vista - IE

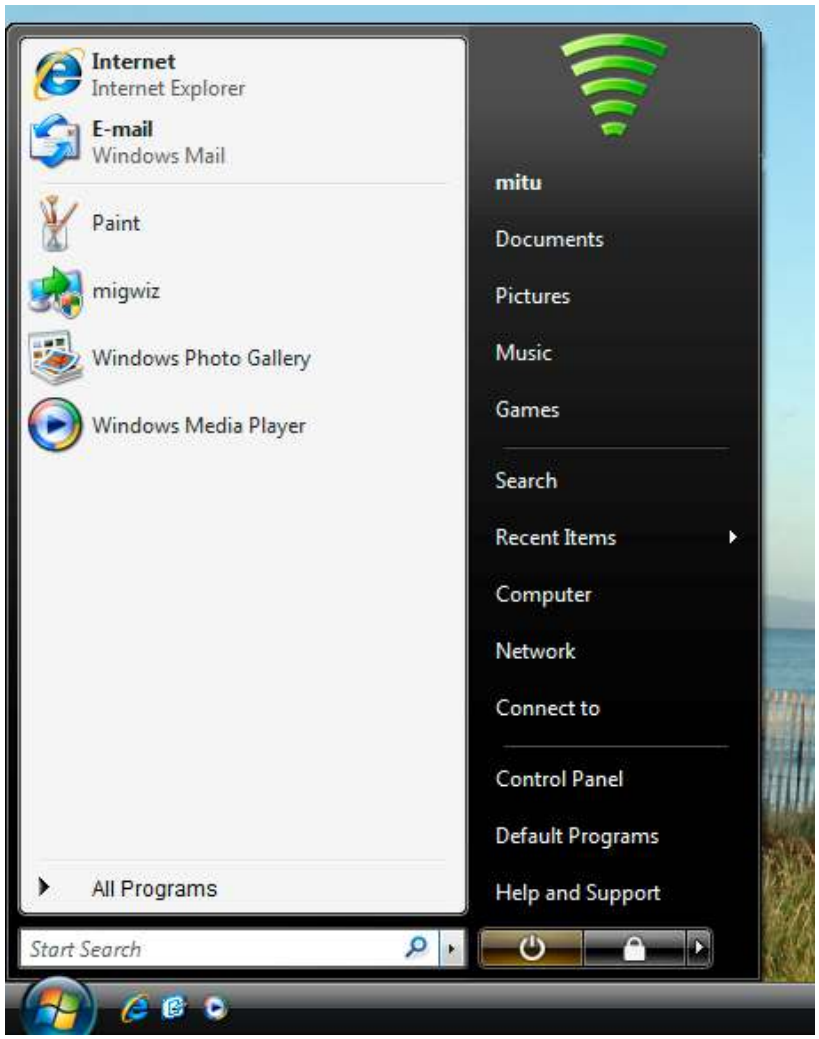


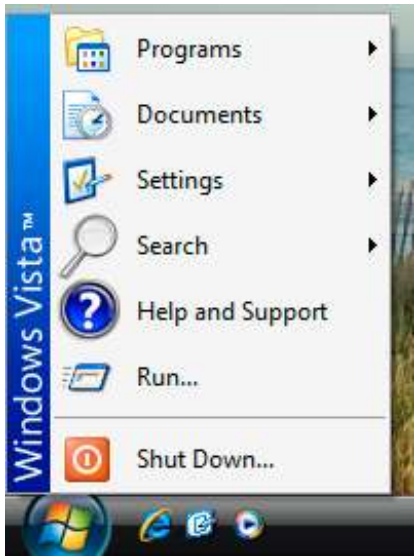
Nowości Vista - WMP

- Ripper mediów (np. z CD)
- Synchronizacja z urządzeniami przenośnymi
- Wbudowany CD/DVD burner



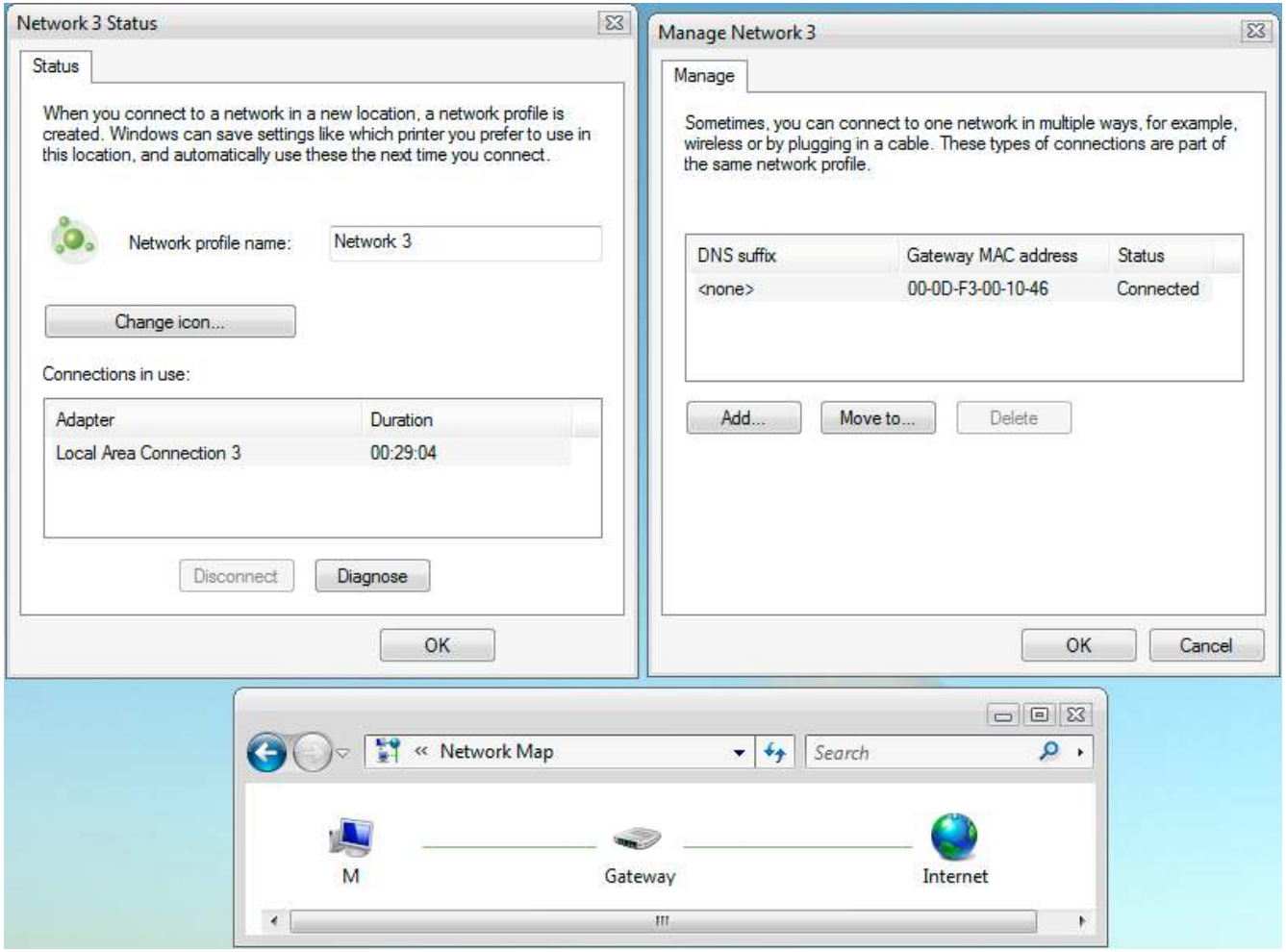
Nowości Vista - Menu



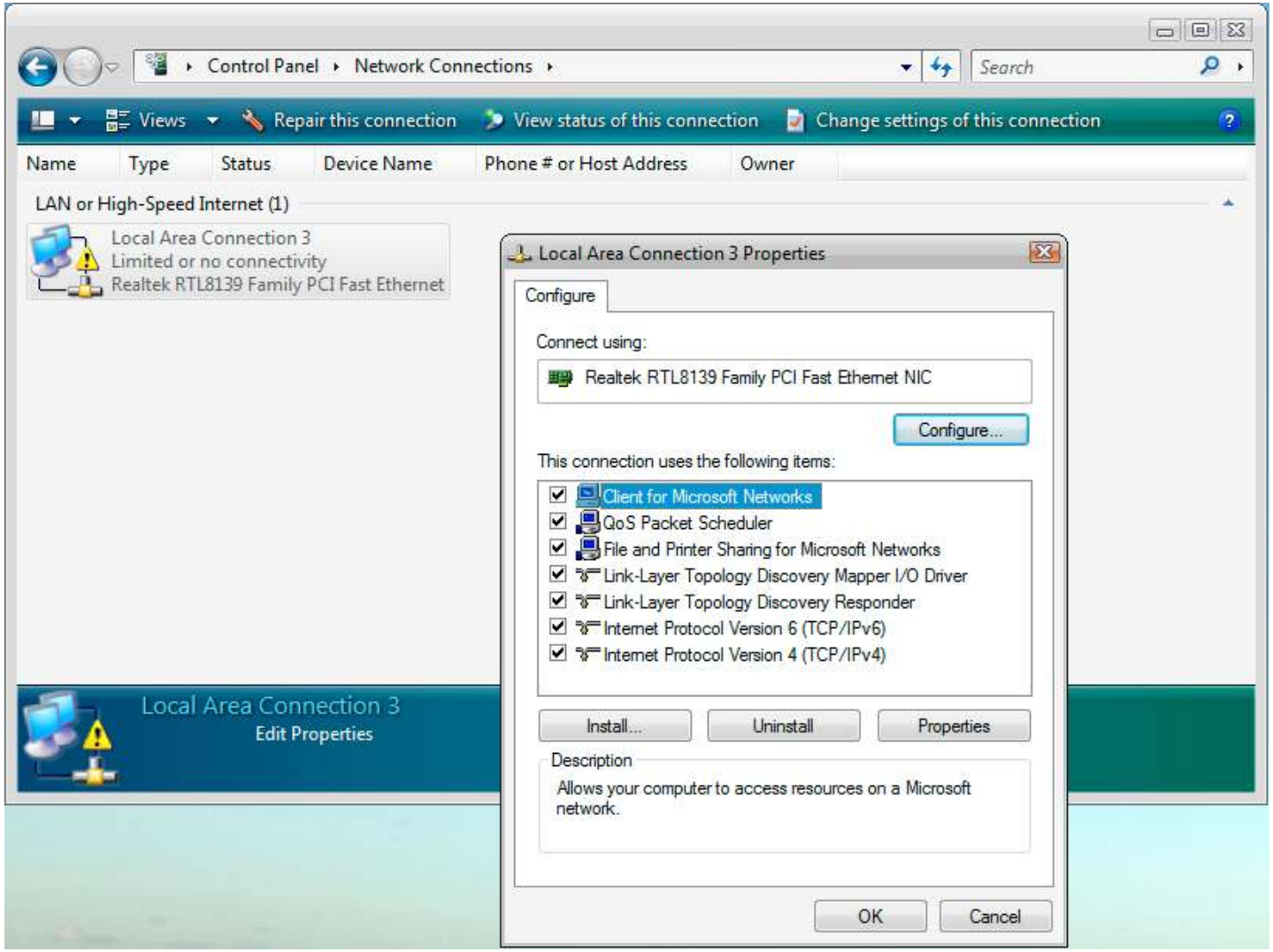


Nowości Vista - Net (I)

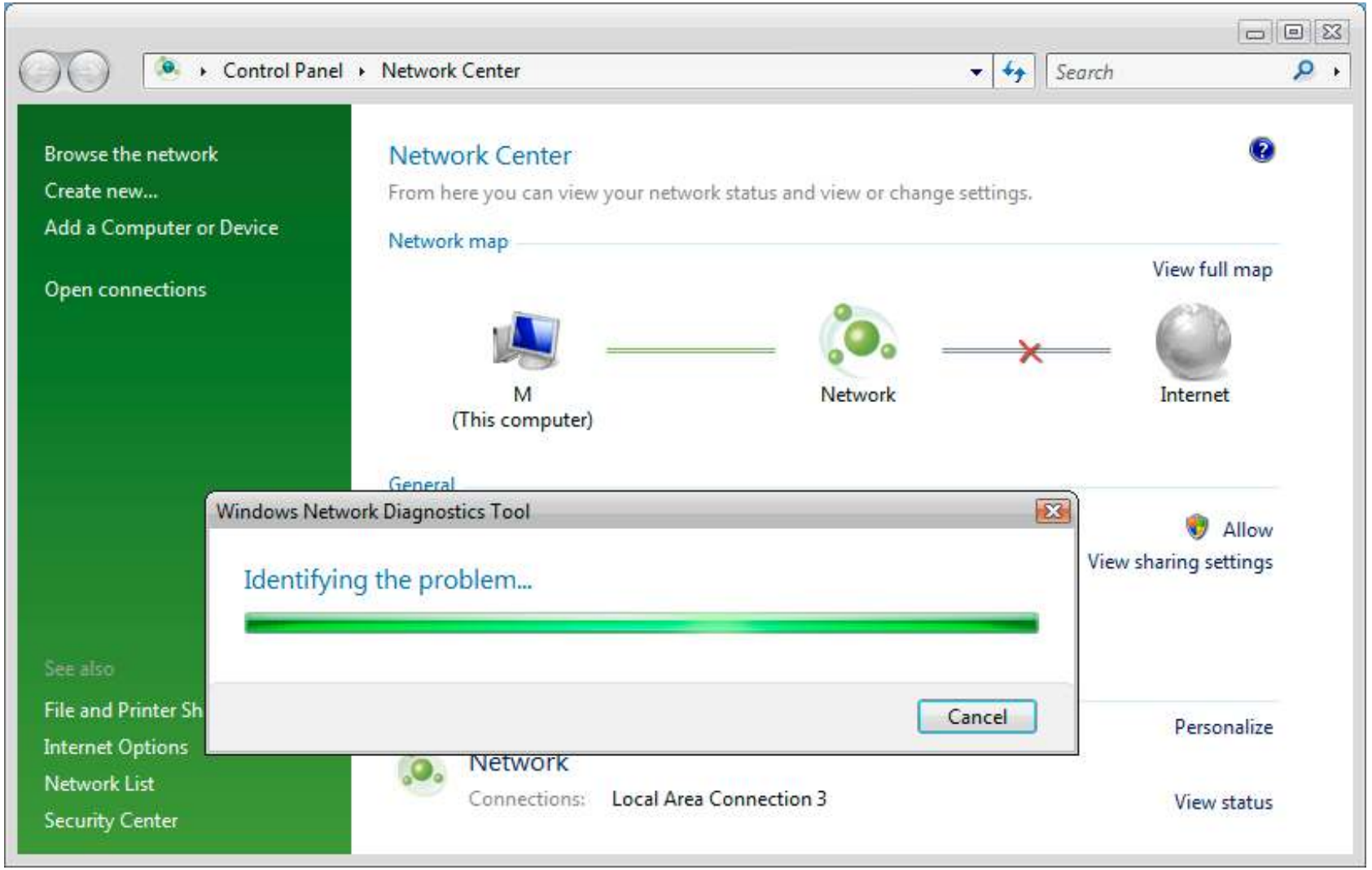
- Uporządkowany problem drogi dla połączenia sieciowego.



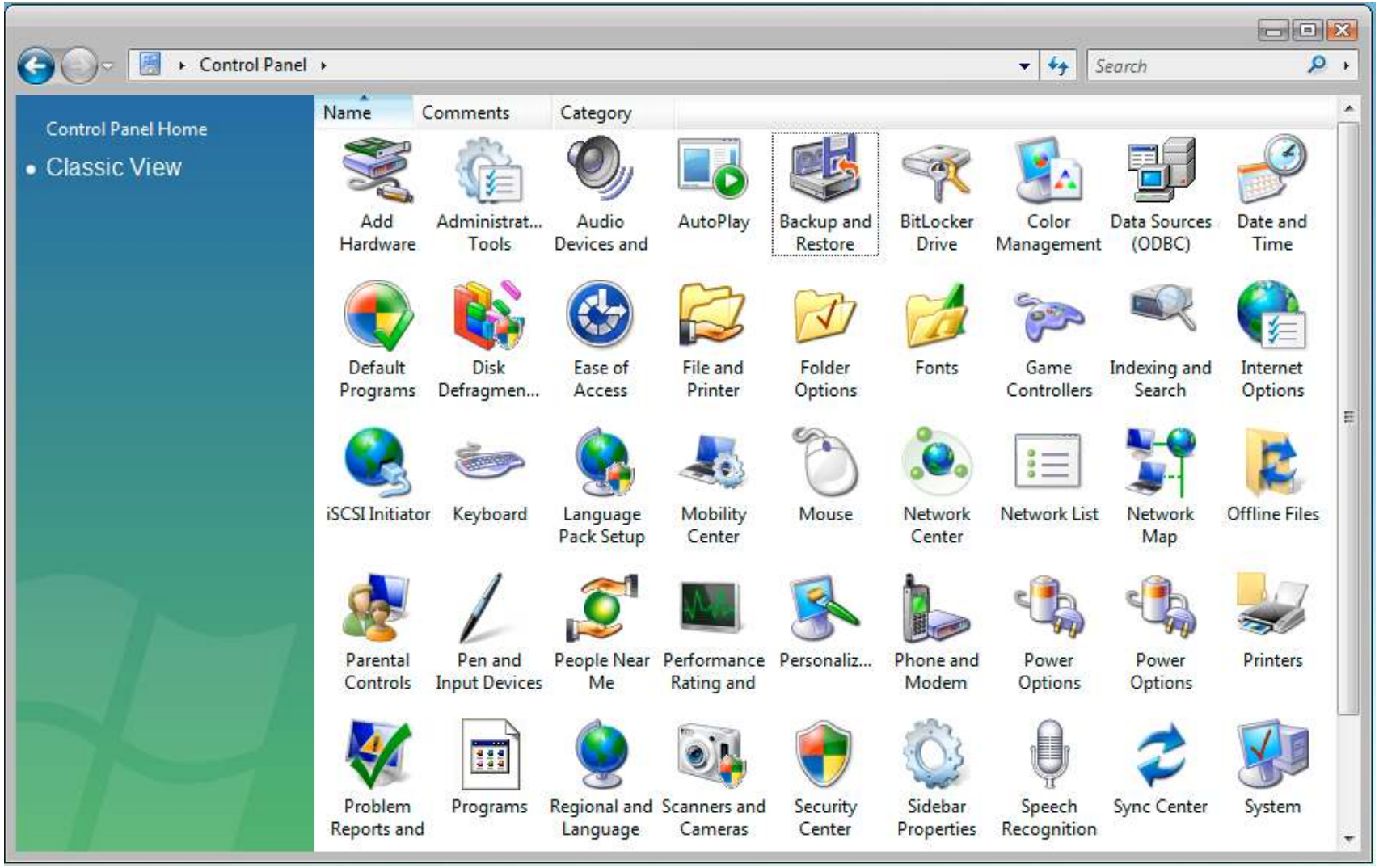
Nowości Vista - Net (II)



Nowości Vista - Net (III)

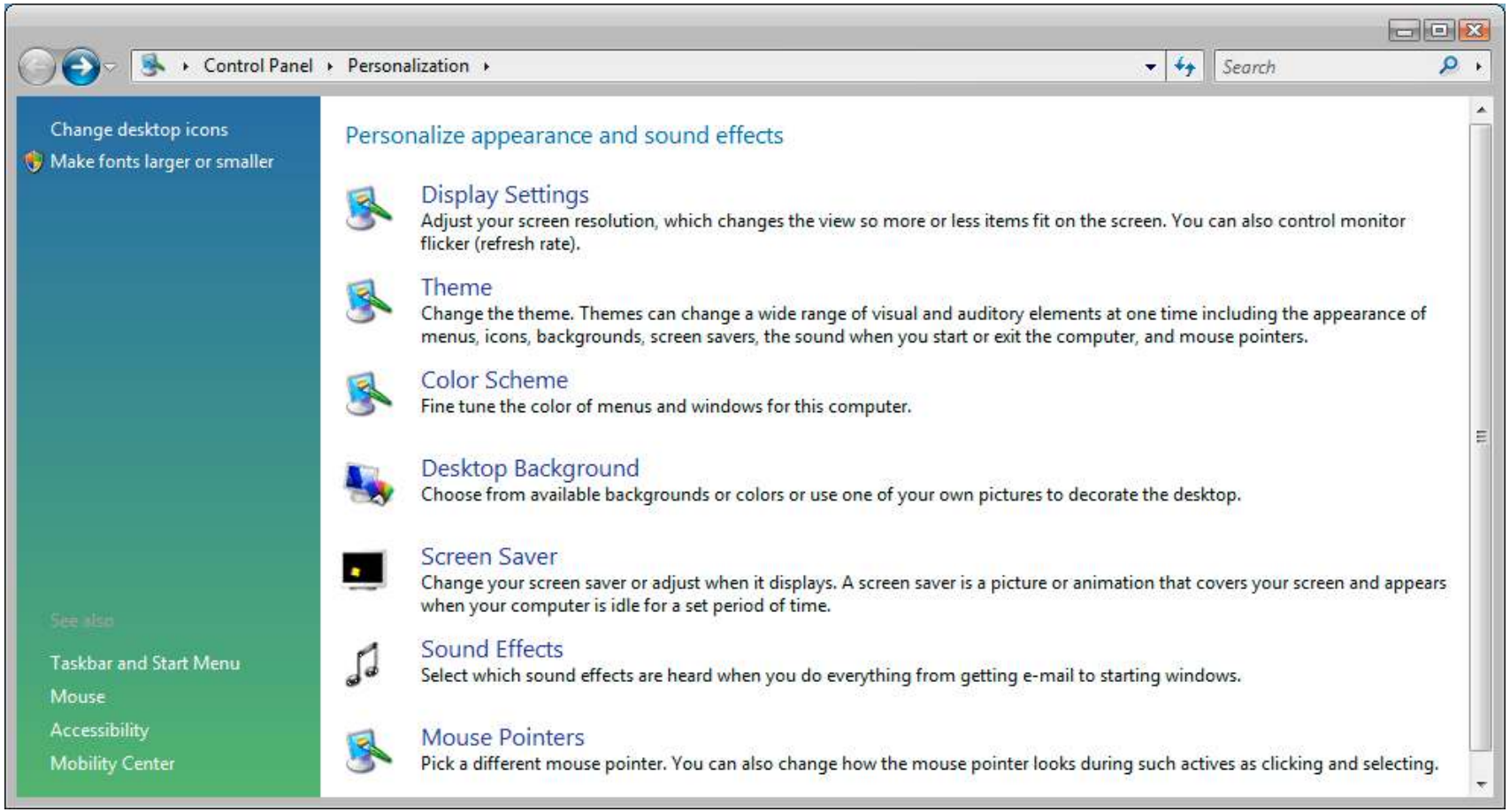


Nowości Vista - Control Panel



Nowości Vista - Personalization

- Uporządkowanie ustawień grafiki, ergonomii etc. w jednym panelu

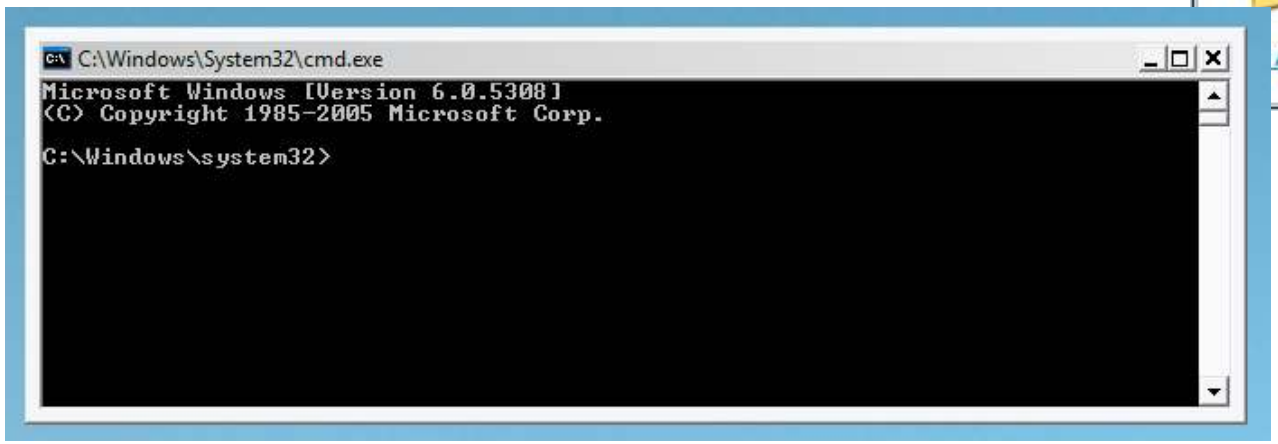


Nowości Vista - Aero

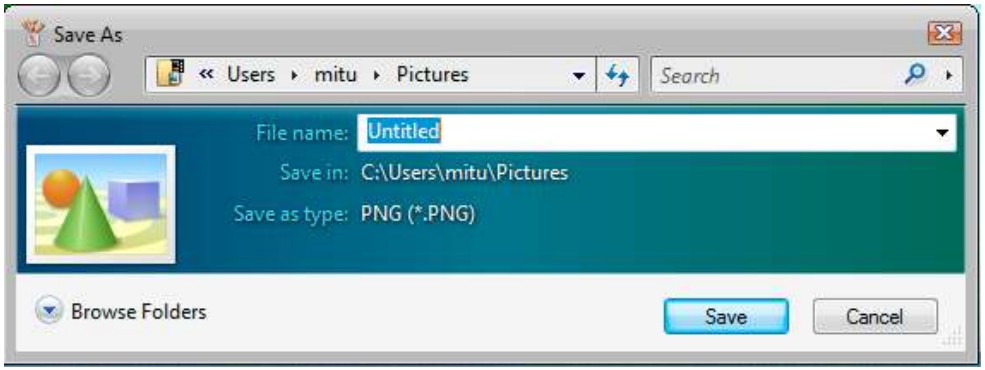
- Vista Glass Look
- Filp-3D Window Changer

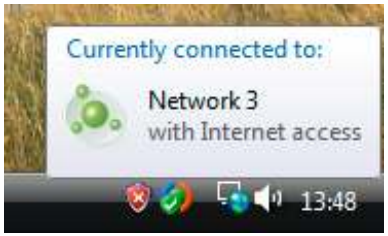
Nowości Vista - Drobiazgi (I)

- Unifikacja adresów internetowych, ścieżek i komend uruchamianych

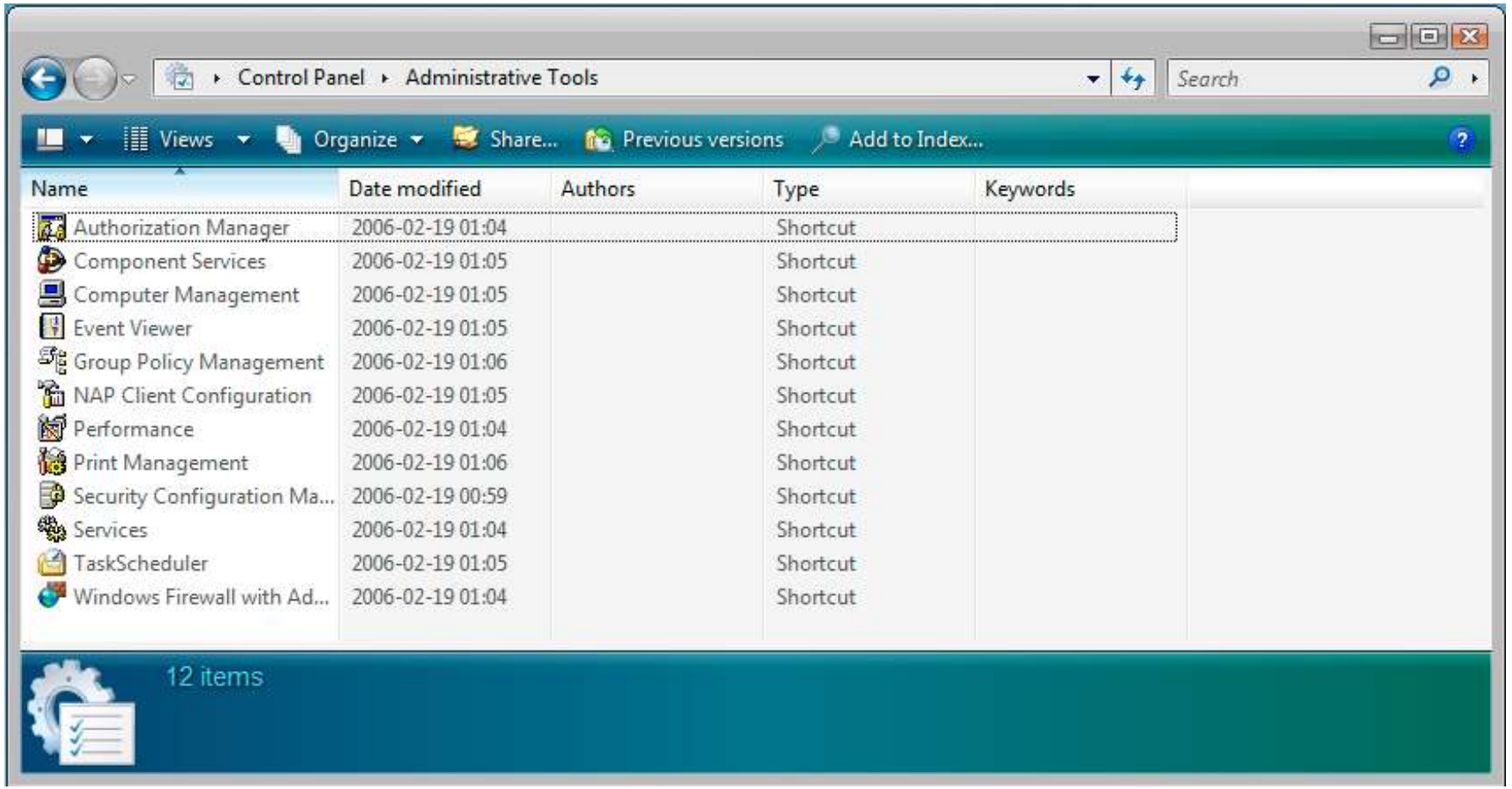


Nowości Vista - Drobiazgi (II)



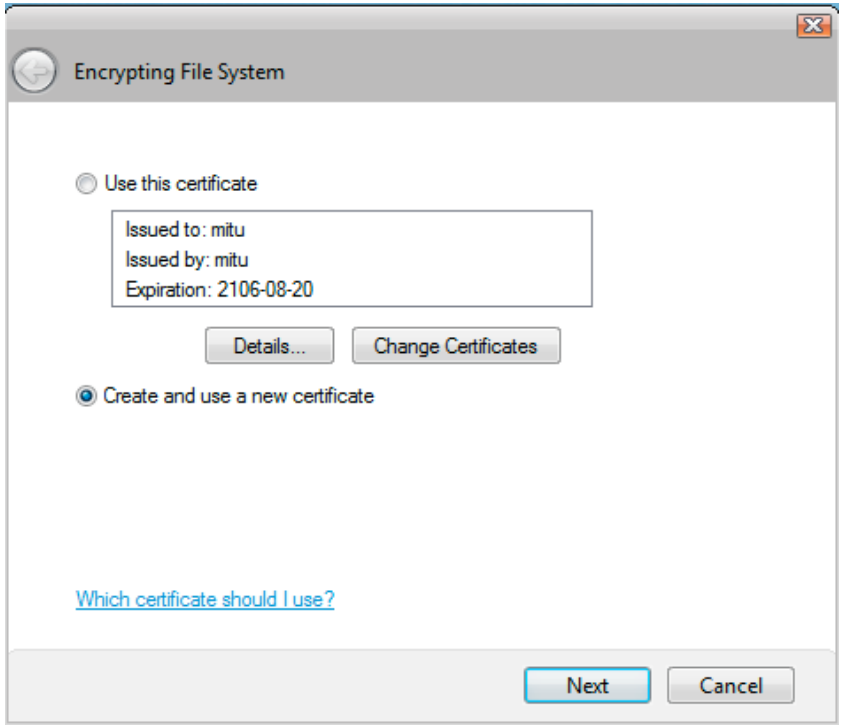


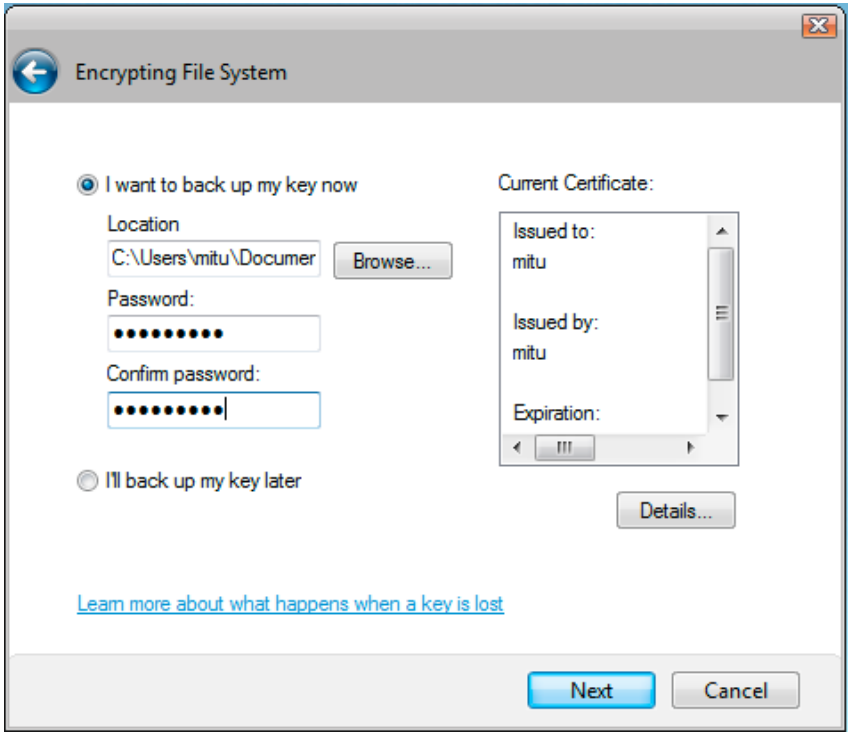
Nowości Vista - Administrowanie (I)



Nowości Vista - Certyfikaty (I)

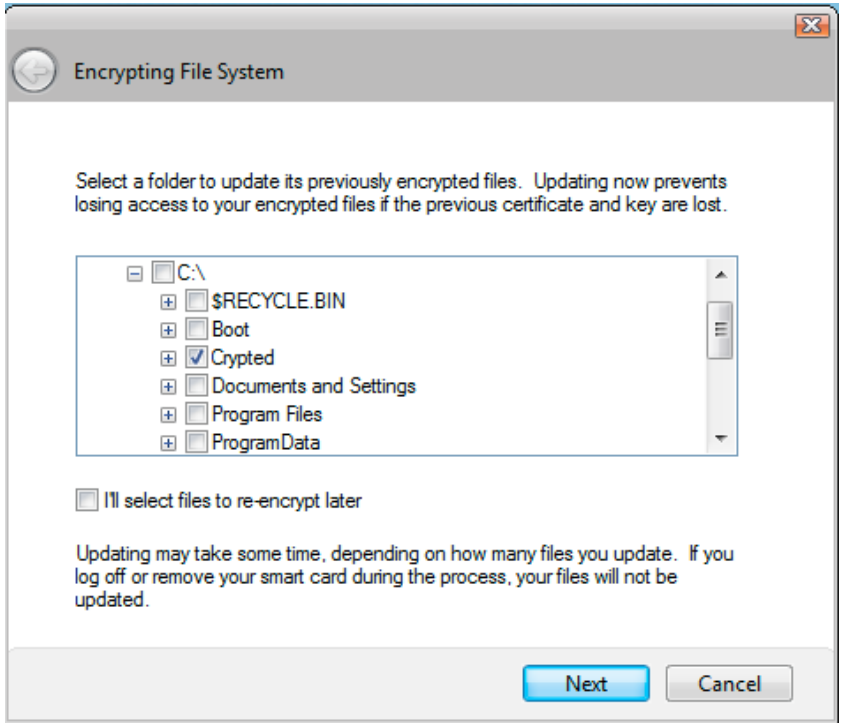
- Porządek w zarządzaniu własnymi certyfikatami szyfrowania plików.

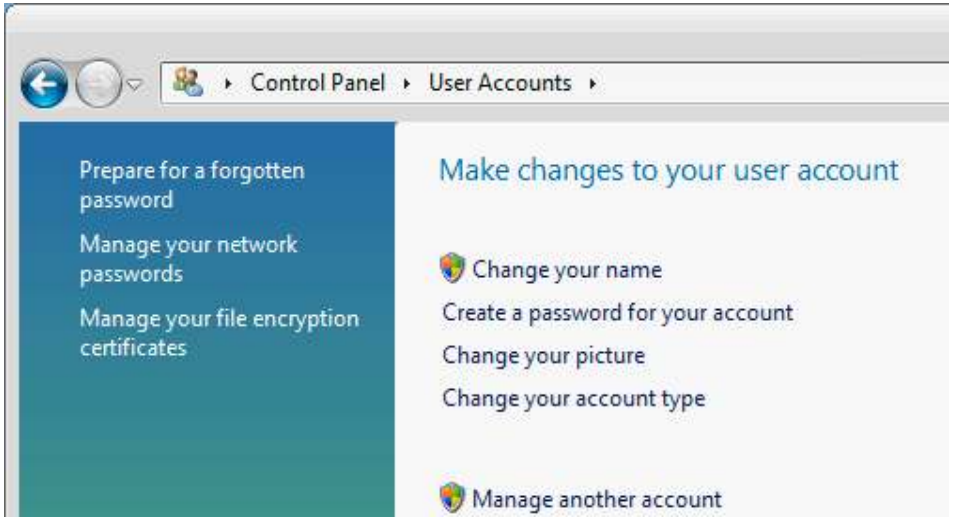




Nowości Vista - Certyfikaty (II)

- Spójny panel zarządzania ustawieniami własnego konta





Proces instalacji systemu operacyjnego (I)

- Planowanie instalacji:
  - Partycjonowanie dysku
  - Rodzaj systemu plików
  - Wybór pakietów instalacji
- Uruchomienie instalacji
- Konfiguracja instalacji
- Instalacja aktualnych łat (ServicePack)

Proces instalacji systemu operacyjnego (II)

- Legalność instalacji systemu
- Kontrola nośników pod instalację - zabezpieczenie danych poprzednio umieszczonych na tych nośnikach, kontrola przydatności nośników do instalacji (dostateczna pojemność, wydajność itp.)
- Źródło instalacji:
  - dyskietki
  - taśmy
  - obrazy HDD
  - CD, DVD
  - sieć

Dystrybucja oprogramowania

- DSP - “Delivery Service Partner” - oprogramowanie dystrybuowane z przeznaczeniem na konkretny typ sprzętu (np. produkowany w danym kraju),
- OEM - “Original Equipment Manufacturer” - oprogramowanie sprzedane bezpośrednio producentowi danego sprzętu z zastrzeżeniem użytkowania wyłącznie na tym (nowym) sprzęcie (z dokładnością do konkretnego egzemplarza),
- BOX - oprogramowanie dystrybuowane indywidualnie, bez prawnych wymogów sprzętowych co do jego przeznaczenia,
- Instalacje Internetowe.

Konfiguracja nośników przed instalacją systemu

- Partycjonowanie
  - Każdy system powinien być na osobnej partycji
  - Narzędzia
    - fdisk
    - format
- Boot manager/boot loader:
  - lilo, managery Windows (wariant zależny od wersji OS).

Opcja RAID przy nośnikach systemowych

- *Redundant Array of Independent Disks*
- Cele dla RAID:
  - Zwiększanie niezawodności
  - Zwiększenie wydajności nośników
  - Powiększenie przestrzeni dostępnej jako jedna całość
- Polega na stosowaniu wielu nośników systemowych (wiele dysków) zawierających informacje składające się na ten sam logiczny dysk
- Najbardziej popularne: RAID-0 i RAID-1

RAID-0 (Striping)

- Polega na zapisywaniu dysku logicznego na wielu dyskach fizycznych dzieląc informacje na poszczególne dyski według klucza
- Rozwiązanie software'owe lub hardware'owe
- Zalety:
  - łatwa implementacja
  - znaczne przyspieszenie operacji dyskowych
- Wady:
  - pogorszenie się bezpieczeństwa danych - w przypadku awarii jednego dysku tracimy całość informacji

RAID-1 (Mirroring)

- Polega na duplikowaniu całych dysków logicznych na bliźniaczych dyskach fizycznych prowadząc równolegle identyczne kopie dysków logicznych podczas pracy systemu
- Rozwiązanie software'owe lub hardware'owe
- Zalety:
  - łatwa implementacja
  - znaczne poprawienie bezpieczeństwa danych
  - znaczne przyspieszenie operacji odczytu
- Wady:
  - duża chłonność nośnika (1:2, 1:3 itp.)

Systemy plików

- Rodzaje systemów plików:
  - alokacja ciągła
  - lista odsyłaczowa (np. FAT)
  - alokacja indeksowana (systemy UNIX)
- W Windows:
  - FAT (ang. File Allocation Table),
  - FAT32,
  - NTFS (ang. New Technology File System)
  - WFS - hmm, może kiedyś...

FAT

- Właściwa nazwa - FAT16,
- MS-DOS i późniejsze
- Maksymalna liczba klastrów - 65536,
- Ograniczenie wielkości partycji do 2 GB
- Brak zabezpieczeń lokalnych na poziomie pliku i folderu

FAT 32

- Windows 95 (nie NT 4.0) i późniejsze,
- Używa tylko 28 bitów adresu klastra - pozwalając teoretycznie na opisanie 268.435.438 klastrów,
- W praktyce - z powodu ograniczeń ScanDisk - 4.177.920 klastrów
- Ograniczenie wielkości partycji do 2 TB (w Windows ograniczono wielkość pojedynczej partycji do 32 GB /WinXP, Win2000/, lub do 124,55 GB - z uwagi na kompatybilność narzędzi),
- Powiększony boot rekord - 3 sektory dyskowe, migrujący katalog główny dysku

NTFS (I)

- NTFS -*New Technology File System*
- Wersje:
  - Windows NT 4.0 (wersja NTFS 4) - bazuje na kodowaniu informacji dodatkowych o plikach w rejestrze systemu
  - Windows 2000 (wersja NTFS 5)
  - Windows Server 2003, Windows XP, Windows Vista (wersja NTFS 6)
- Wielkość klastra 0,5 - 4 KB,
- Maksymalny rozmiar partycji: teoretycznie 2^{64} klastrów, w implementacjach 2^{32} klastrów.

NTFS (II)

- Zabezpieczenia lokalne na poziomie plików — uprawnienia dostępu można nakładać na foldery oraz pojedyncze pliki. Uprawnienia NTFS obowiązują podczas korzystania z zasobów przez sieć, jak również lokalnie,
- Wspomaganie kompresji plików — kompresja “w locie”, niewymagająca zewnętrznych programów kompresujących,
- Przydziały dysku (ang. Disk Quota),
- System szyfrowania plików (EFS) (ang. Encrypted File System),
- Pliki rozrzedzone NTFS,
- Funkcjonalność w zakresie śledzenia zmian.

Systemy plików - ważne uwagi (I)

- Woluminy FAT mniejsze od 16 megabajtów (MB) są formatowane jako FAT12.
- Woluminy FAT16 większe niż 2 gigabajty (GB) nie są dostępne dla komputerów używających systemu MS-DOS, Windows 95, Windows 98 i wielu innych.
- Mimo, że woluminy FAT32 mogą teoretycznie osiągać rozmiar aż do 2 terabajów (TB), to w systemie Windows 2000 i XP istnieje ograniczenie rozmiaru formatowanego woluminu FAT32 do 32 gigabajtów. System Windows 2000 może jednak odczytywać i zapisywać większe woluminy FAT32, które zostały sformatowane w innych systemach operacyjnych.

Systemy plików - ważne uwagi (II)

- Implementacja systemu FAT32 w systemie Windows 2000 i XP ogranicza maksymalną liczbę klastrów woluminu FAT32, który może zostać zainstalowany, do 4.177.918. Jest to maksymalna ilość klastrów w woluminie FAT32, która może zostać sformatowana w systemie Windows 98.
- Woluminy NTFS mogą teoretycznie osiągać rozmiar aż do 16 eksabajtów (EB), ale w praktyce są ograniczone do 2 terabajtów.
- Użytkownik może określić rozmiar klastra podczas formatowania woluminu NTFS. Kompresja NTFS nie jest jednak obsługiwana w przypadku klastrów większych niż 4 kilobajty (KB).

(->lab1) Konsola MMC (I)

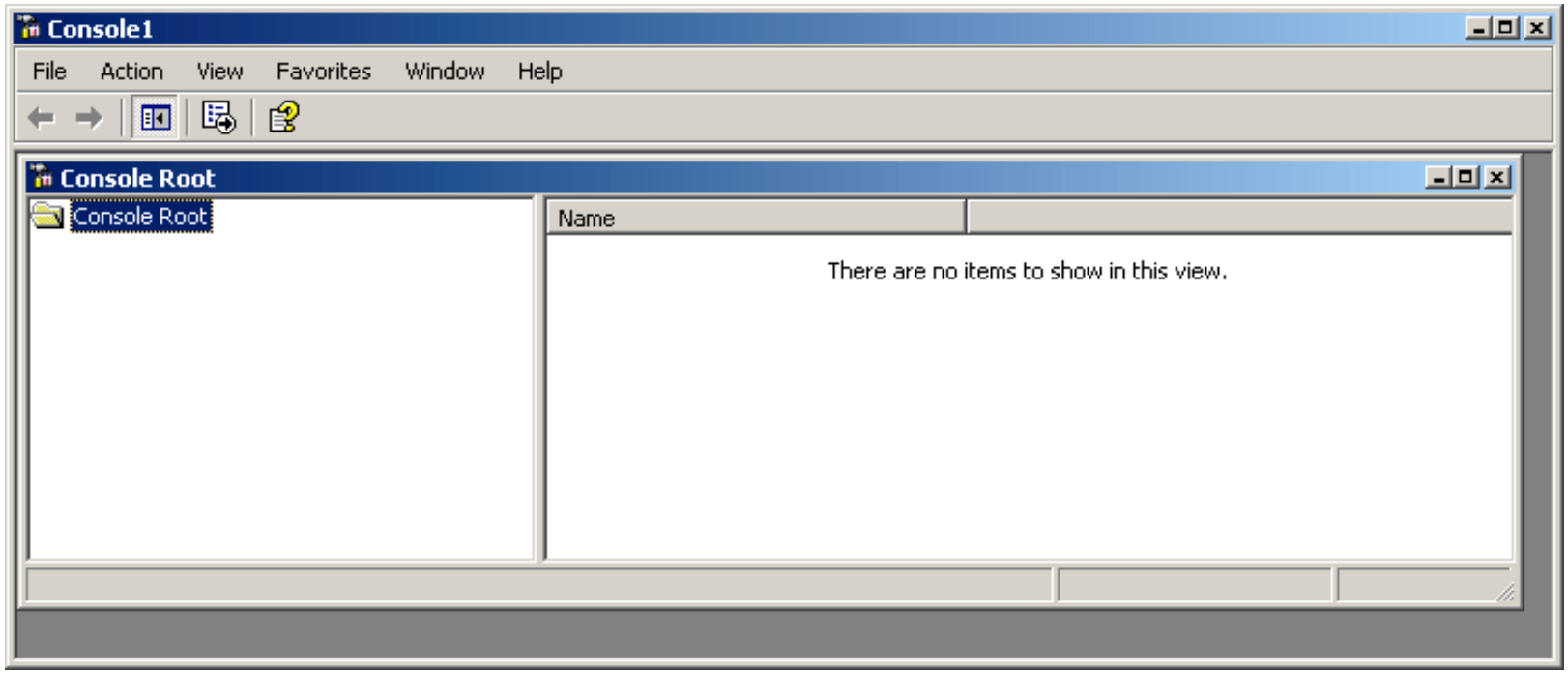
- *Microsoft Management Console*
- Jednolite środowisko, do którego można dodawać programy administracyjne w postaci tzw. *przystawek*
- Możliwość tworzenia własnych przystawek
- Przystawki zapisywane w plikach *.msc

Konsola MMC (II)

- Ważniejsze przystawki predefiniowane (WinXP):
  - compmgmt.msc - zarządzanie komputerem
  - certmgr.msc - zarządzanie certyfikatami
  - devmgmt.msc - menedżer urządzeń
  - dfrg.msc - defragmentator dysków
  - diskmgmt.msc - zarządzanie dyskami
  - eventvwr.msc - dziennik zdarzeń
  - eventvwr.msc - zarządzanie zasobami dzielonymi
  - perfmon.msc - analizator wydajności
  - secpol.msc - zabezpieczenia
  - services.msc - zarządzanie usługami

Konsola MMC (III)

- Pusta konsola MMC. Uruchomienie poleceniem: *mmc /a*

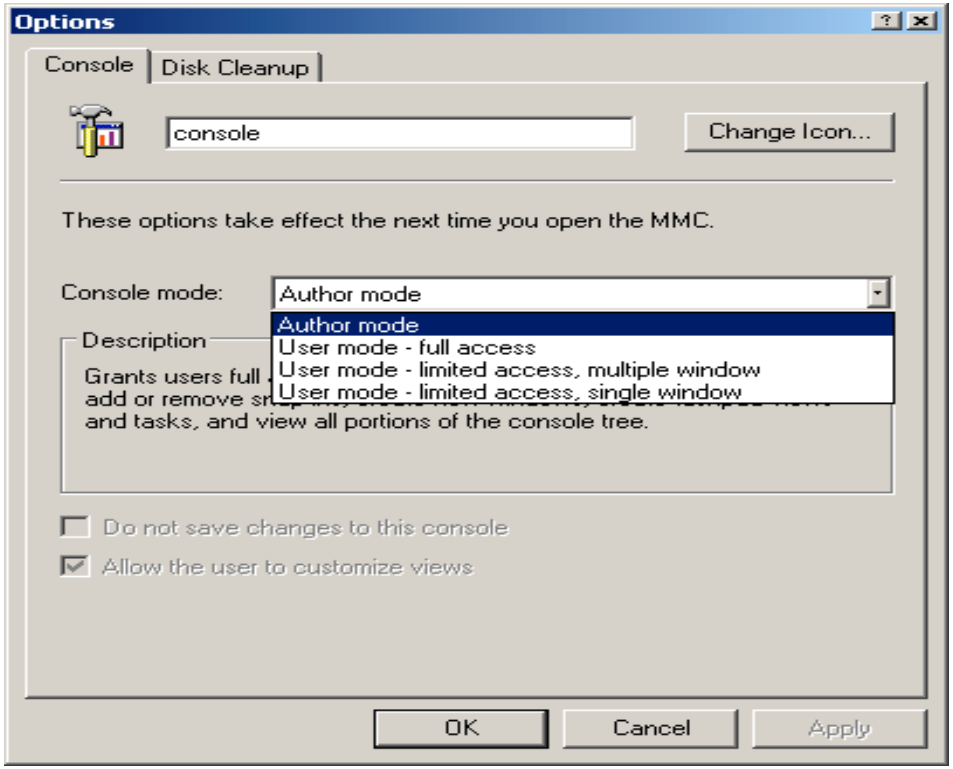


Konsola MMC - tryby pracy

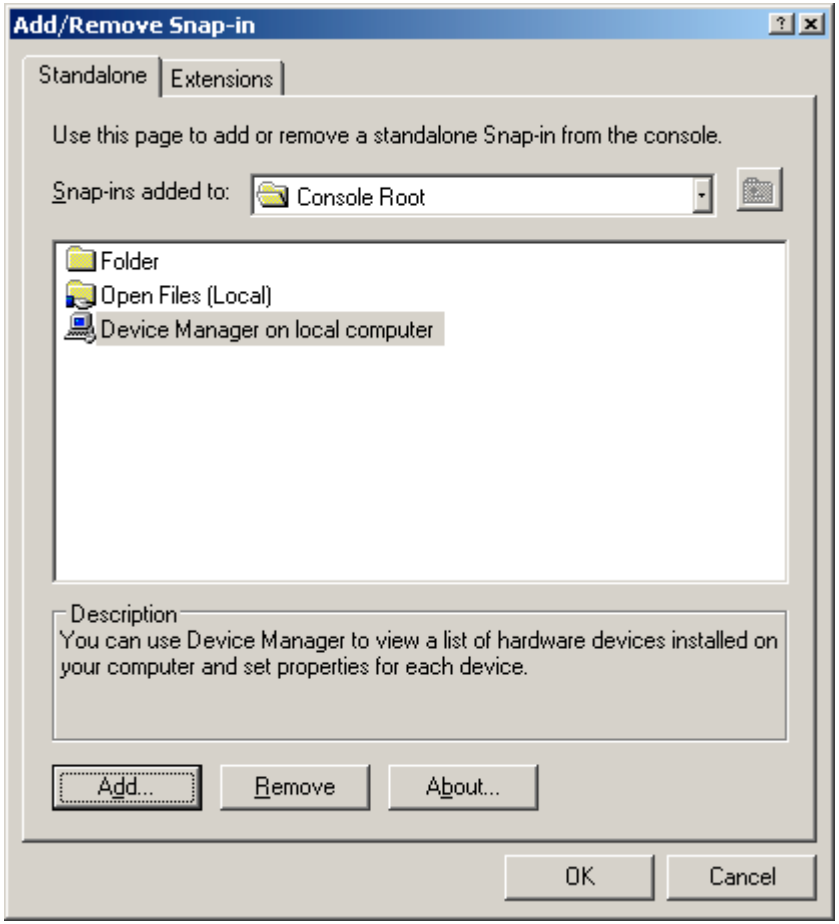
- *Tryb autorski* - zezwala użytkownikowi na pełny dostęp do funkcji konsoli MMC. Użytkownik może dodawać przystawki, tworzyć nowe okna, edytować parametry pracy przystawek.
- *Tryb użytkownika – pełny dostęp*
  Daje użytkownikowi pełny dostęp do konsoli, lecz nie pozwala na dodawanie i usuwanie przystawek oraz zmianę właściwości. Wszystkie pozostałe funkcje są włączone.
- *Tryb użytkownika – ograniczony dostęp, wiele okien*
  Pozwala na dostęp wyłącznie do tych obszarów drzewa konsoli, które były widoczne w momencie zapisywania konsoli, tworzenie nowych okien. Możliwość zamykania istniejących okien jest niedostępna.
- *Tryb użytkownika – ograniczony dostęp, jedno okno*
  Ustawienie najbardziej restrykcyjne, różni się od poprzedniego dodatkowym ograniczeniem w postaci braku możliwości tworzenia dodatkowych okien.

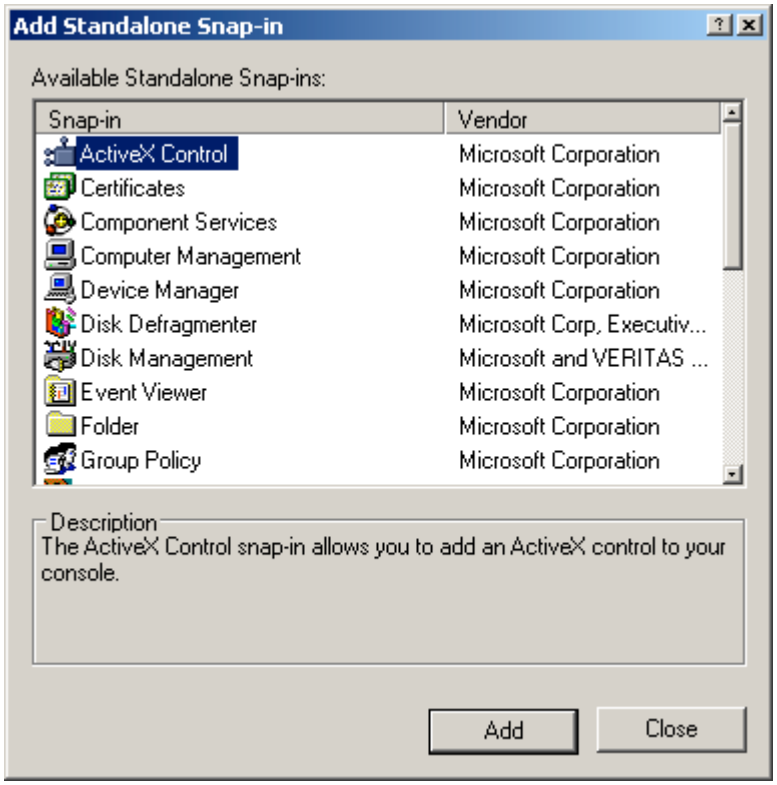
Konsola MMC - wybór trybu pracy

- Zapisanie konsoli z niskimi uprawnieniami uniemożliwia późniejszą zmianę tych uprawnień!



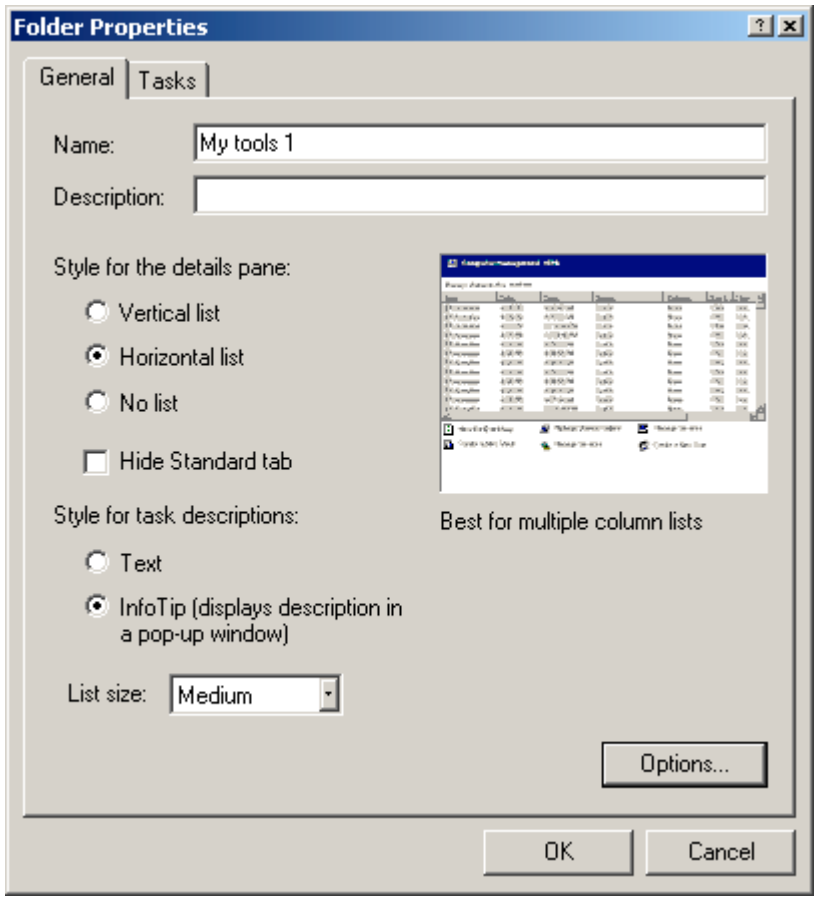
Konsola MMC - dodawanie standardowych przystawek





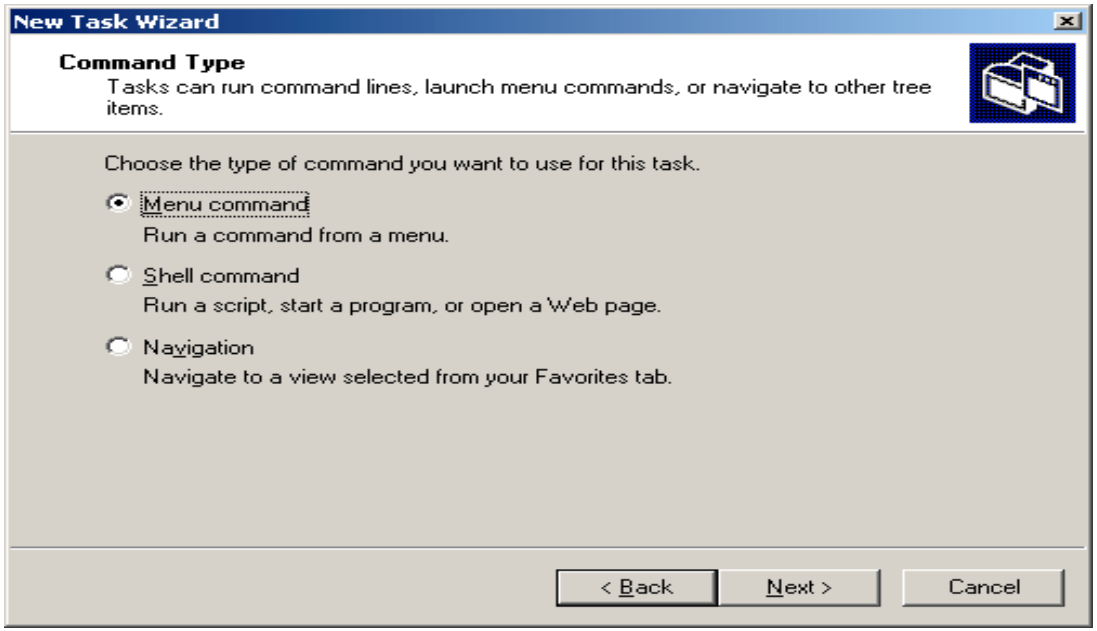
Konsola MMC - widoki bloku zadań

- W bloku zadań instalujemy definicje zadań możliwych do wykonania z poziomu konsoli



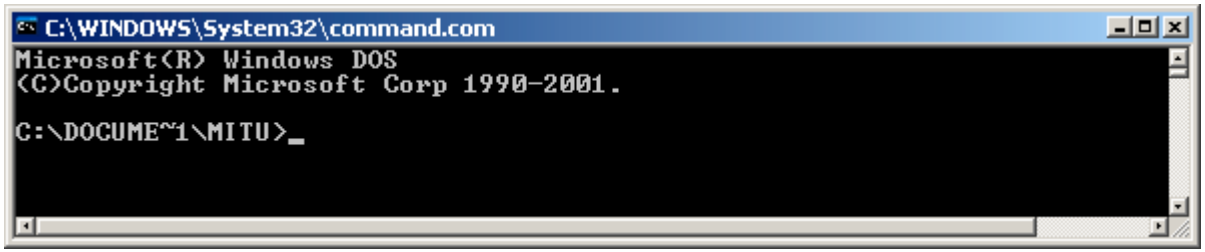
Konsola MMC - zadania użytkownika

- Możliwe jest dodanie zadania będącego akcją interfejsu MMC, wywołaniem zewnętrznego programu lub pochodzącego z puli konsoli MMC dla danego systemu.

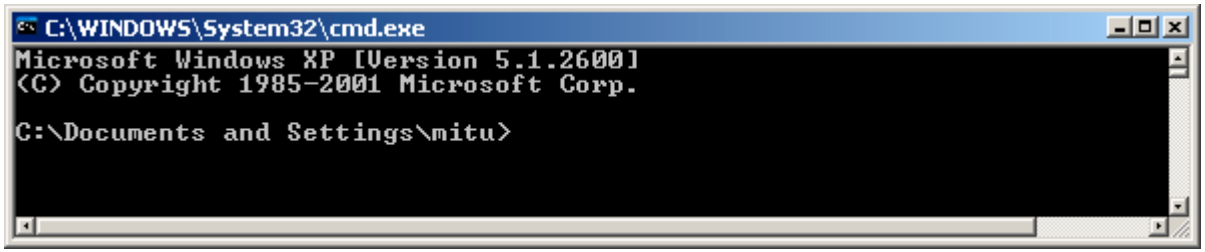


Shell komend Windows - wersja historyczna i obecne

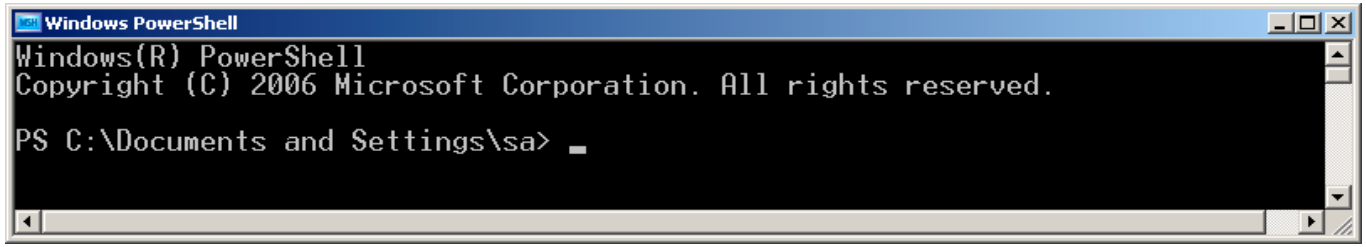
- command.com



- cmd.exe



- Monad Shell
- PowerShell



Shell komend Windows - funkcjonalność w Win32

- Wbudowana historia komend z numerowaniem i dostępem z menu (F9,F8,F7)
- Obsługa długich nazw, ścieżek BNC
- Obsługa sieci
- Automatyczne uzupełnienie nazw plików i katalogów
- Ustawienia pobierane z rejestru:

  HKEY_CURRENT_USER|Software|Microsoft|Command Processor

Shell komend Windows - autouzupełnienie

- Uruchomienie opcji następuje poprzez wywołanie:
  - cmd.exe /F:ON - dla plików i katalogów
- Po włączeniu rozwinięcie do folderu uzyskujemy przez naciśnięcie Ctrl-D, do pliku przez Ctrl-F
- Standardowe ustawienia przełączników są zdefiniowane w rejestrze

Shell Windows - sterowanie komendami (I)

- Separator komend w linii: &&, przykładowo:
  - cd C: && dir - spowoduje przejście na C: i wyświetlenie zawartości
- Przekierowanie strumieni (pipe): |, przykładowo:
  - dir | find „exe”
  - Przy przekierowaniach strumieni istotne są parametry podane do cmd: /A - strumień ANSI, /U - strumień UNICODE

Shell Windows - sterowanie komendami (II)

- Przekierowanie strumieni do plików:
  - >> - dodanie do zawartości
  - > - zapisanie zawartości
  - Przykład:
    - dir >> plik.txt - dopisze wygenerowany przez komendę tekst do pliku
    - dir > plik.txt - zapisze wygenerowany przez komendę tekst w pliku likwidując poprzednią zawartość

Windows Shell - przetwarzanie łańcuchów

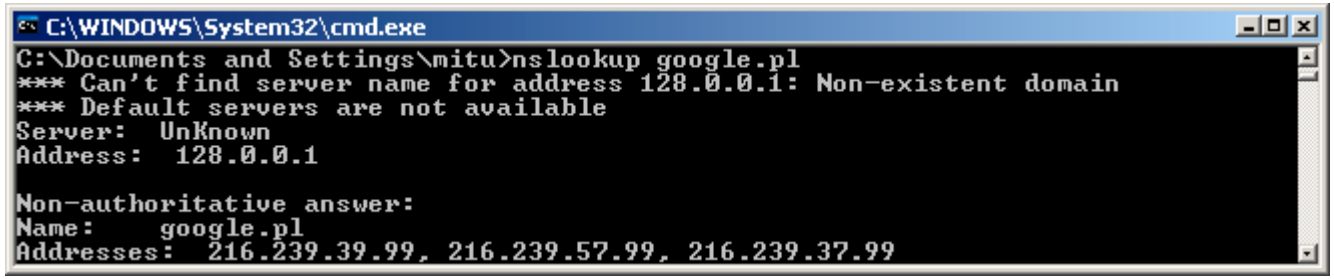
- FIND - przeszukuje łańcuchy testowe
- FINDSTR - wersja zaawansowana operująca na wyrażeniach regularnych i wielu plikach
- SORT - sortuje kolekcje łańcuchów testowych
- COMP - porównuje kolekcje łańcuchów testowych typując różniące się

Shell Windows - inne ustawienia

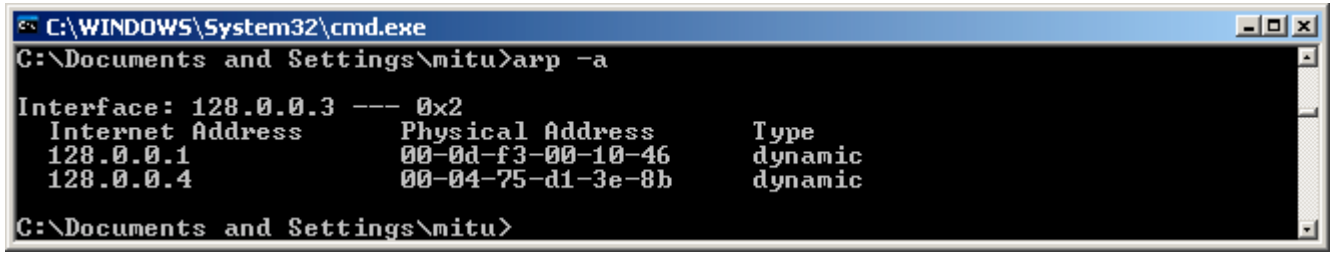
- Echo ekranowe - włączenie i wyłączenie echa oprócz komendy echo [off|on] uzyskujemy parametrem cmd /Q wyłączającym echo
- Kolory konsoli - komenda color ?? lub parametr cmd /T:??, gdzie ?? To dwie cyfry hex odpowiadające kolejno za kolor tla i tekstu. Kolor zapisany jest w kluczu: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Command Processor\DefaultColor

Tekstowe narzędzia do diagnostyki sieci (I)

- nslookup - klient DNS

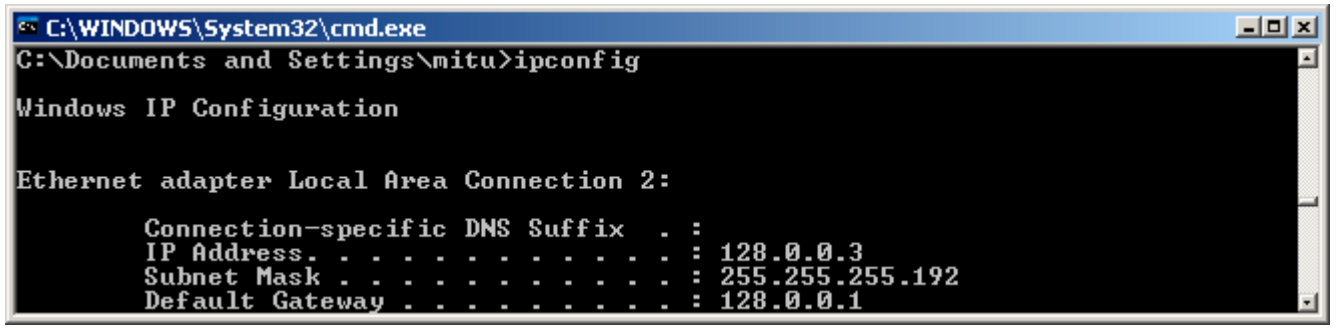


- arp - zarządzania tablicami ARP

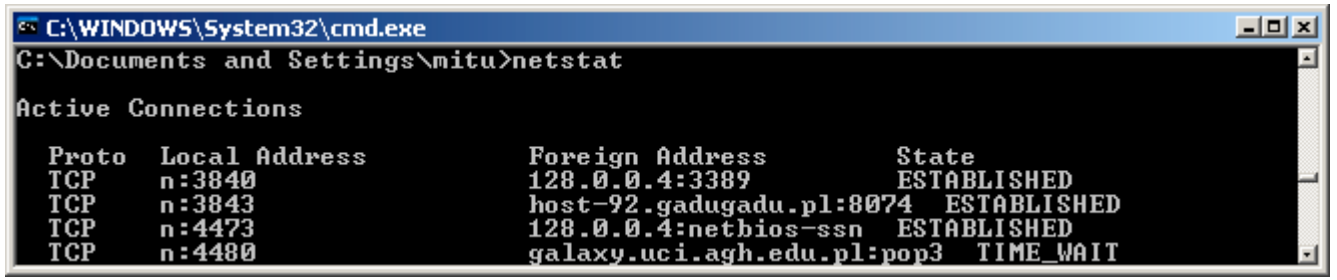


Tekstowe narzędzia do diagnostyki sieci (II)

- getmac - pobiera adresy MAC interfejsów sieciowych
- ipconfig - przedstawia konfigurację sieci

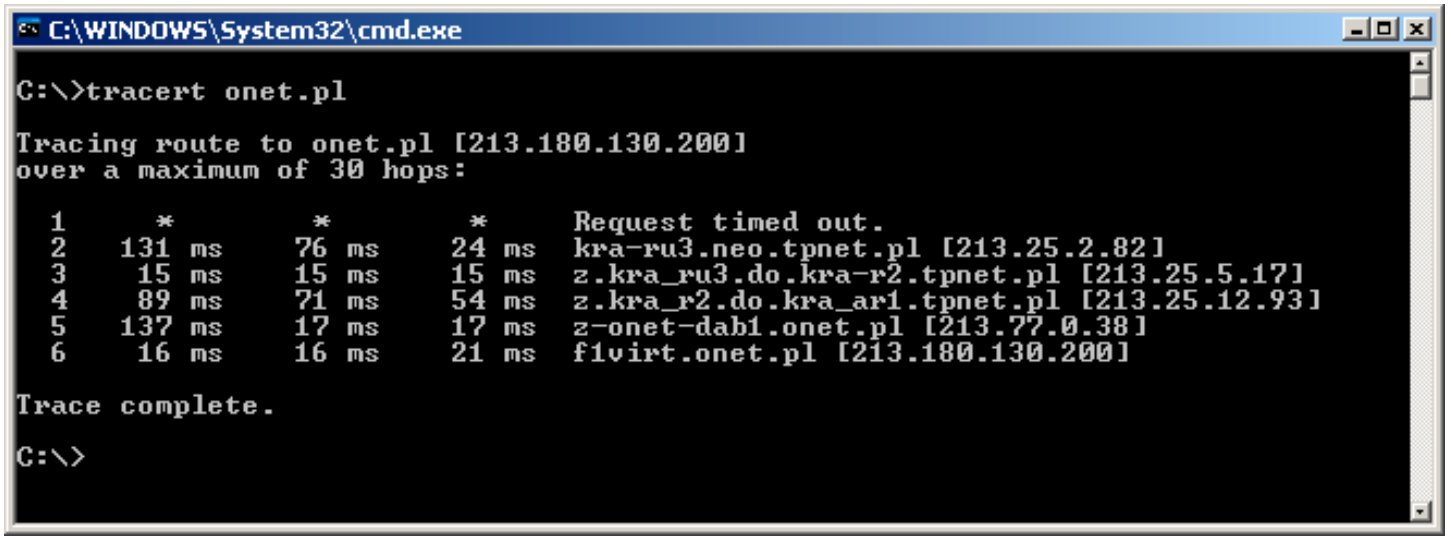


- netstat - aktualne połączenia sieciowe



Tekstowe narzędzia do diagnostyki sieci (III)

- Traceroute, śledzenie hosta. Przykład:
  - *tracert.exe onet.pl*



Konfiguracja interfejsów sieciowych z linii komend (I)

- Narzędzie: netsh
- Przykłady:

```
netsh interface ip set address name="Połączenie
Lokalne 4" source=static addr=10.0.0.1
mask=255.0.0.0 gateway=10.0.0.2 2
```

- ustawienie parametrów IP

```
netsh interface ip set address name="Połączenie
Lokalne 4" source=dhcp
```

- ustawienie użycia DHCP

Konfiguracja interfejsów sieciowych z linii komend (II)

- Przykłady dla netsh (CD):
  - netsh interface ip dump
    - pobranie ustawień konfiguracyjnych (w postaci skryptu). Skrypt taki można następnie wykorzystać do ponownego konfigurowania,przykładowo:

    netsh interface ip dump >> plik.conf
    - zapisanie ustawień w pliku
    netsh exec plik.conf
    - (późniejsze) użycie ustawień z pliku

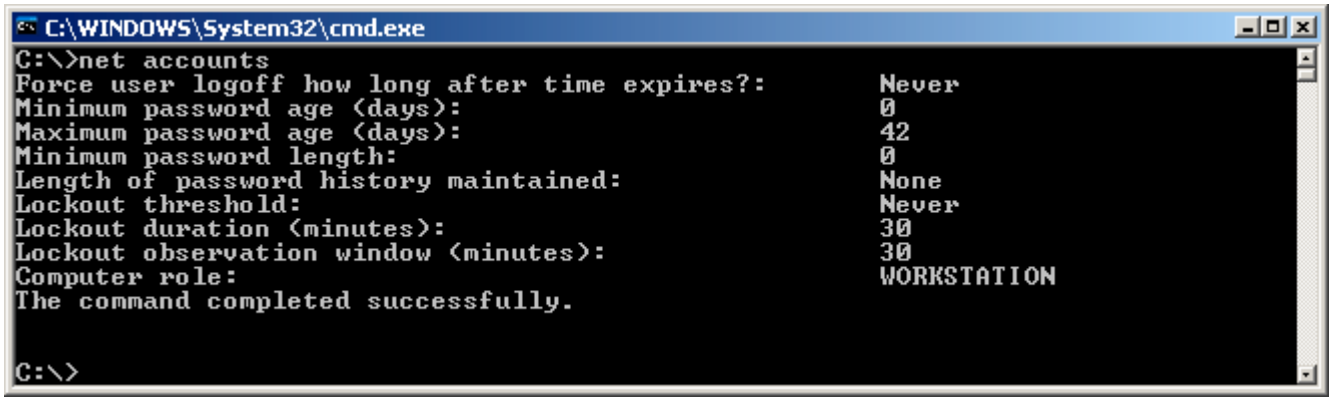
Program NET (I)

- Zestaw narzędzi sieciowych i funkcji konfigurujących
- Program interpretuje komendy, podane jako pierwszy parametr
- Bardziej rozbudowana alternatywa dla interfejsu okienkowego

Program NET (II)

- Zarządzanie ustawieniami kont :

  net accounts



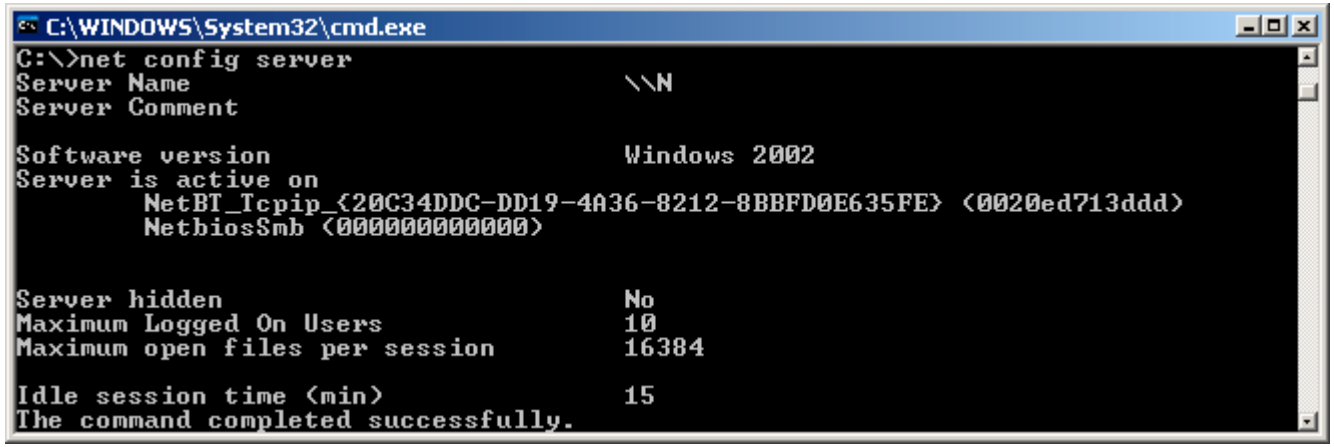
- Zarządzanie komputerami w domenie:

  net computer \\newcomp /add

Program NET (III)

- Kontrola usług (np. konfiguracja, zatrzymanie, wznowienie):

  net config server

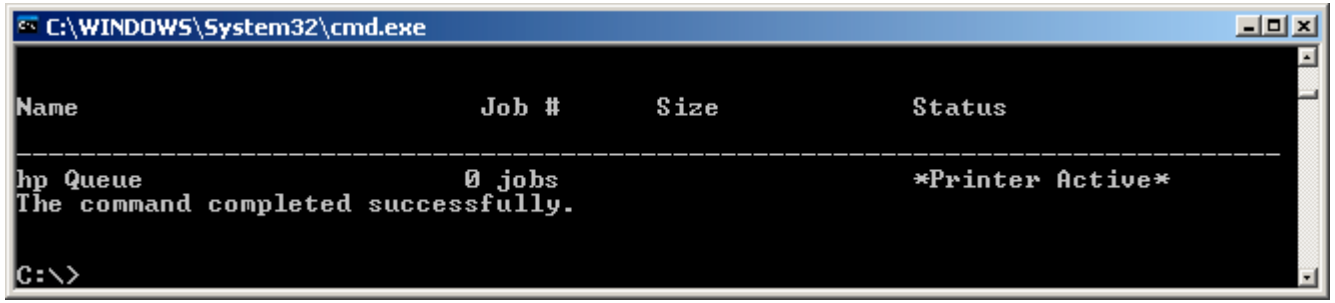


  net pause, net stop, net continue, net start

Program NET (IV)

- Zarządzanie wydrukiem. Przeglądanie, kasowanie, wstrzymywanie zadań wydruku identyfikowanych poprzez numery

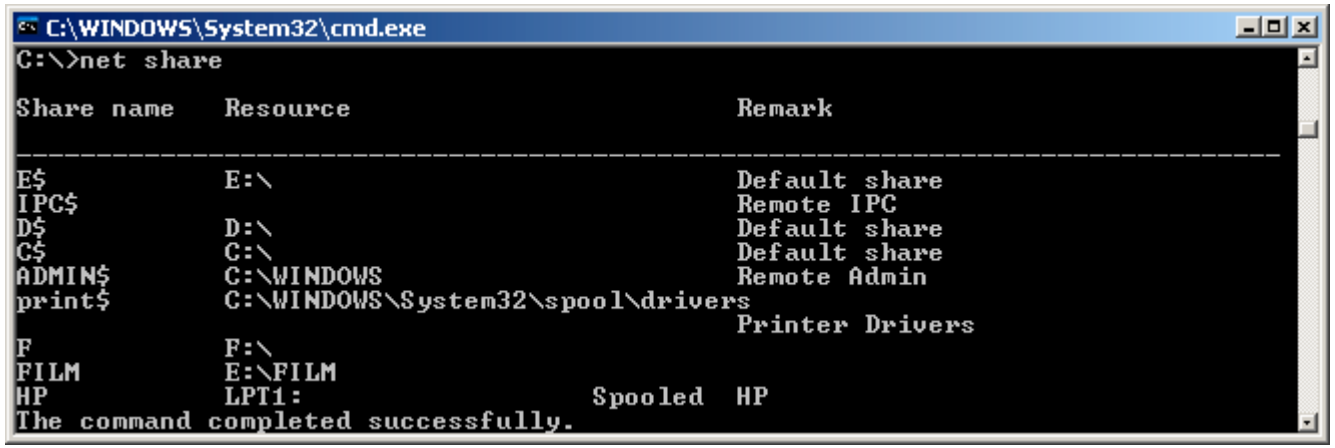
  net print \\komputer\drukarka



Program NET (V)

- Zarządzanie udostępnianiem udziałów sieciowych

  net share
  net share G=c:\tmp
  net share G /DELETE

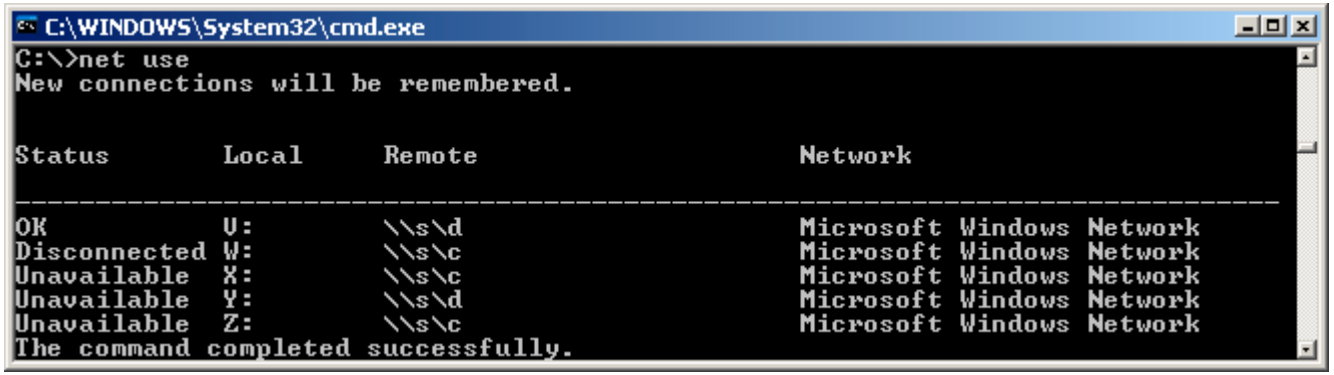


Program NET (VI)

- Zarządzanie użytkowaniem udziałów sieciowych. Mapowanie na identyfikatory lokalne

  net use * \\kompS\shareC ”pass" /user:user1

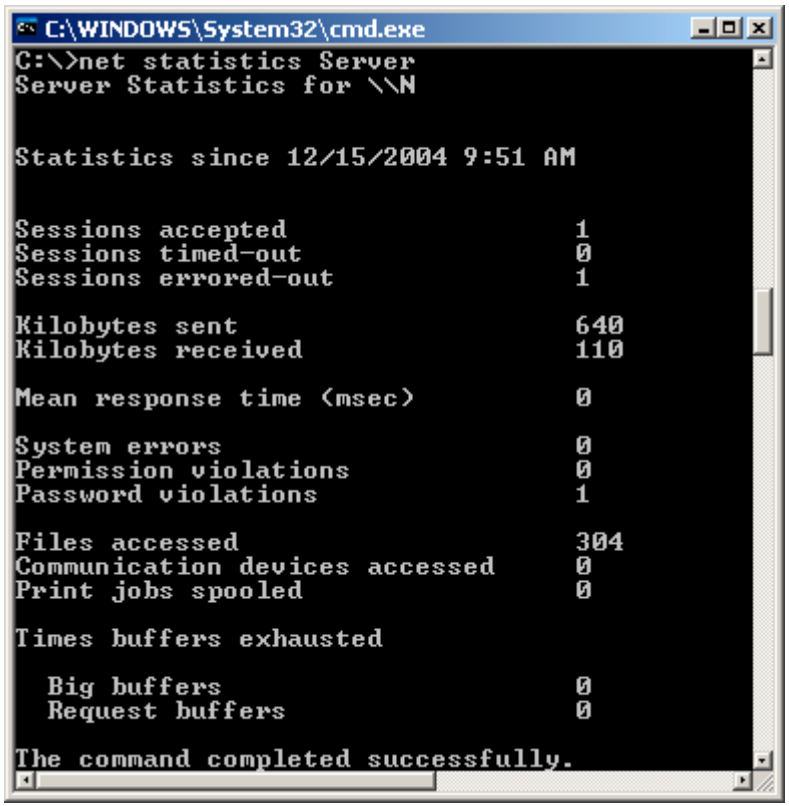
  net use * /DELETE



Program NET (VII)

- Statystyki dla niektórych usług sieciowych

  net statistics Server



Program NET (VIII)

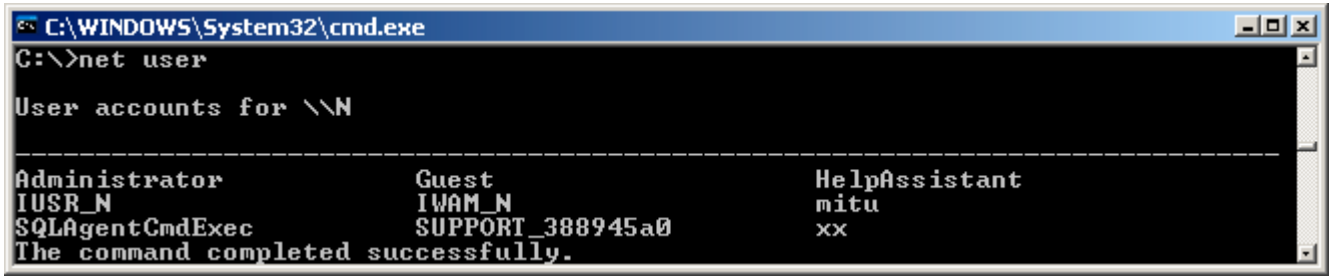
- Przesyłanie komunikatów do innych maszyn, domeny, użytkowników posiadających otwarte sesje:

  net send * „Komunikat”
  net send domain:domena „Komunikat”
  net send komputer „Komunikat”

Program NET (X)

- Zarządzanie kontami: wyświetlanie, dodawanie, usuwanie, zmiana haseł

  net user
  net user nowy /ADD
  net user uzytkownik nowe_haslo
  net user nowy /DELETE

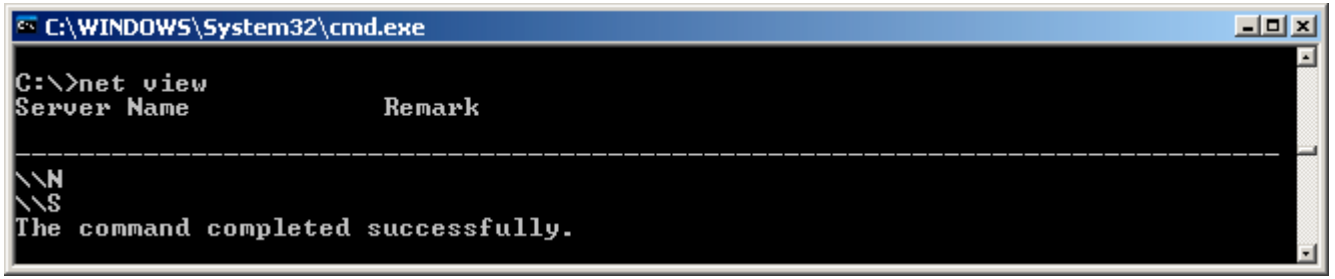


Program NET (XI)

- Lokalizowanie maszyn i ich zasobów w obrębie sieci lokalnej:

  net view

  net view \\komp

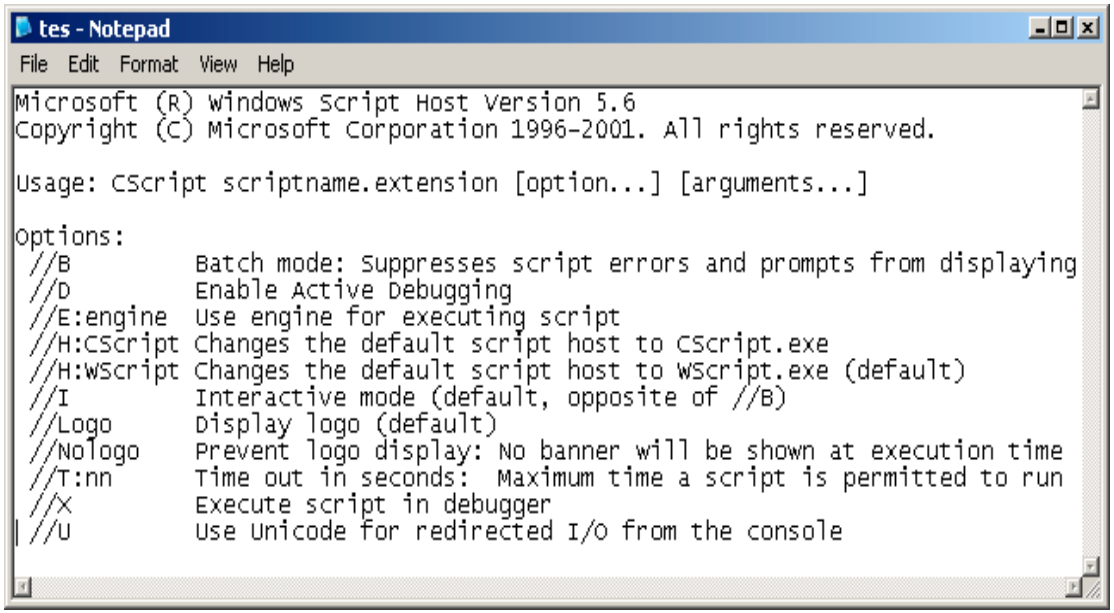


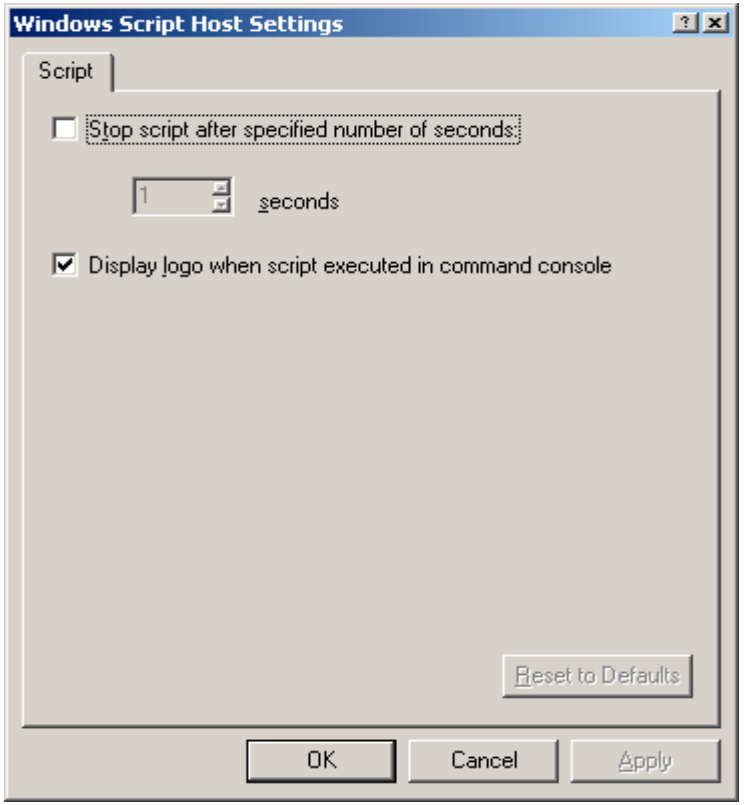
Skrypty WSH (I)

- Host skryptów windows (*Windows Script Host*)
- Alternatywne języki. Standardowo JScript, VBScript
- Standardowe pliki skryptów: .js, .vbs
- Alternatywne interfejsy hosta skryptów - tekstowy i graficzny. Jeden z nich domyślnie wybrany.
- Tryb debugowania skryptów

Skrypty WSH (II)

- Wywołanie:
  - host tekstowy cscript.exe plik_skryptu
  - host wscript.exe plik_skryptu

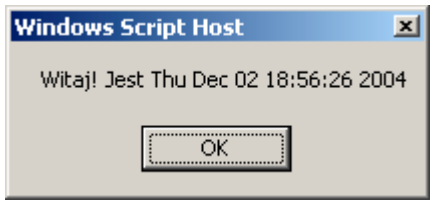


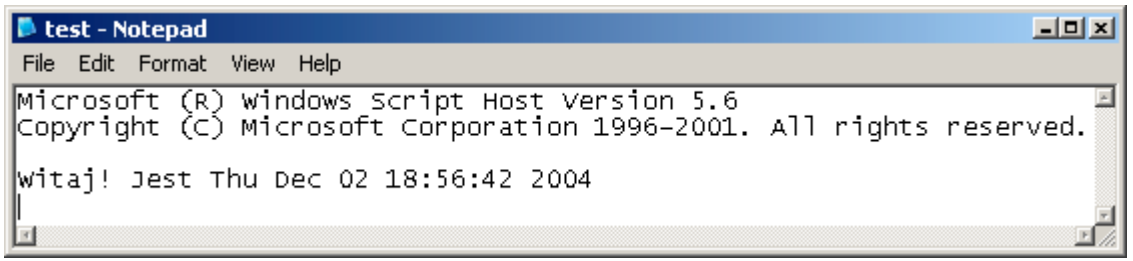


Skrypty WSH - najprostszy przykład

- Jscript:

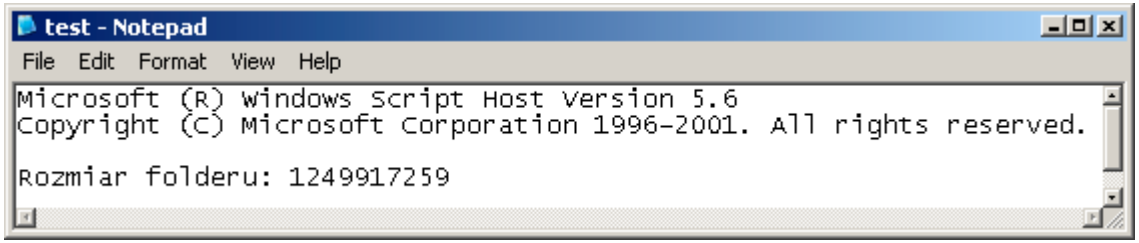
```
WScript.Echo('Witaj! Jest ' +  Date());
WScript.Sleep 2000
```





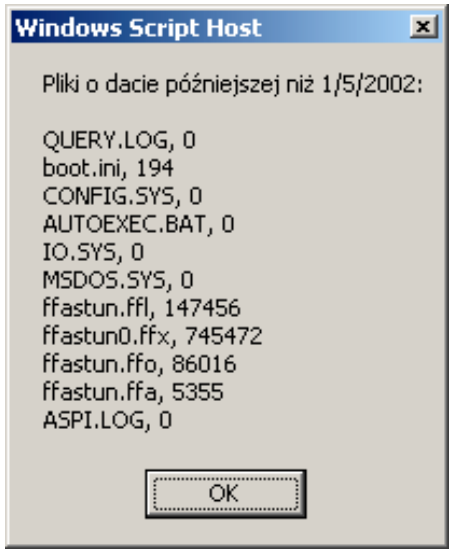
Skrypty WSH - przykłady (I)

- fso = new ActiveXObject('Scripting.FileSystemObject');
- var katalog = fso.GetFolder ('c:\\windows');
- WScript.Echo('Rozmiar folderu: ' + katalog.Size);



Skrypty WSH - przykłady (II)

```
var fso, pliki, data, buf = '';
fso = new ActiveXObject('Scripting.FileSystemObject');
var katalog = fso.GetFolder('c:\\');
pliki = new Enumerator(katalog.files);
data = new Date(2002, 5, 1);
for (;!pliki.atEnd(); pliki.moveNext())
if (pliki.item().DateLastModified > data)
buf += pliki.item().Name + ', ' + pliki.item().Size + '\n';
WScript.Echo('Pliki o dacie późniejszej niż ' + data.getDate() + '/' +
   data.getMonth() + '/' +
   data.getYear() + ':\n\n' + buf);
```



Skrypty WSH - przykłady (III)

- Modyfikacje plików:
  - Tworzenie:

```
Set objFSO = CreateObject("Scripting.FileSystemObject")
Set objFile = objFSO.CreateTextFile("C:\NowyPlik.txt")
```

  - Tworzenie pliku tymczasowego - z unikatową nazwą i złożoną ścieżką:

```
Set objFSO = CreateObject("Scripting.FileSystemObject")
strPath = "C:\katalog"
strFileName = objFSO.GetTempName
strFullName = objFSO.BuildPath(strPath, strFileName)
Set objFile = objFSO.CreateTextFile(strFullName) objFile.Close
```

  - Kasowanie:

```
(..)
objFSO.DeleteFile(strFullName)
```

Skrypty WSH - przykłady (IV)

- Współpraca z aplikacjami:
  - var x1 = new ActiveXObject ('InternetExplorer.Application');
  - x1.Navigate('http://www.wste.edu.pl/');
  - x1.Visible = true;
- Word.Application, Access.Application i inne

Skrypty WSH - przykłady (V)

- Nawigacja IE z formularzami (dostęp do wyników):

```
Set objExplorer = WScript.CreateObject ("InternetExplorer.Application", "IE_")
objExplorer.Navigate "file:///c:\form.html"
objExplorer.Visible = 1
objExplorer.ToolBar = 0
objExplorer.StatusBar = 0
objExplorer.Width=400
objExplorer.Height = 250
objExplorer.Left = 0
objExplorer.Top = 0
Do While (objExplorer.Document.Body.All.OKClicked.Value = "")
    Wscript.Sleep 250
Loop
strPassword = objExplorer.Document.Body.All.Password.Value
strText = objExplorer.Document.Body.All.Text.Value
objExplorer.Quit
Wscript.Sleep 250
Wscript.Echo "Nazwa: " & strText
Wscript.Echo "Haslo: " &  strPassword
```

Skrypty WSH - przykłady (VI)

- Nawigacja IE z formularzami (cd):
- <SCRIPT LANGUAGE="VBScript">
- Sub RunScript
- OKClicked.Value = "OK"
- End Sub
- </SCRIPT>
- <BODY>
- Nazwa:<input type="text" name="Text" size="40"></p>
- Hasło:<input type="password" name="Password" size="40"></p>
- <input type="hidden" name="OKClicked" size = "20">
- <input id=runbutton class="button" type="button" value=" OK "
- name="ok_button" onClick="RunScript">
- </BODY>

Skrypty WSH - przykłady (VII)

- Dostęp do aplikacji MS Word:

```
Set objWord = CreateObject("Word.Application")
objWord.Caption = ”Dokument testowy"
objWord.Visible = True
Set objDoc = objWord.Documents.Add()
Set objSelection = objWord.Selection
objSelection.Font.Name = "Arial"
objSelection.Font.Size = "18"
objSelection.TypeText "Spis procesow systemu, data: " & Date() & " " & Time()
objSelection.TypeParagraph()
objSelection.TypeParagraph()
objSelection.Font.Size = "10"
set list = GetObject("winmgmts:").execquery("select * from win32_process")
For Each item in list
        objSelection.TypeText "" & item.Name
        objSelection.TypeParagraph()
Next
objDoc.SaveAs("C:\bb.doc")
objWord.Quit
```

Skrypty WSH - przykłady (VIII)

- Dostęp do aplikacji MS Excel - operowanie na plikach:
  - Set objExcel = CreateObject("Excel.Application")
  - objExcel.Visible = True
  - Set objWorkbook = objExcel.Workbooks.Open("C:\aa.xls")
  - objExcel.Cells(7, 7).Value = "Tresc ze sktyptu"
  - objExcel.ActiveWorkbook.SaveAs("C:\ab.xls")

Skrypty WSH - przykłady (IX)

- Dostęp do aplikacji MS Excel - dostęp do komórek:
  - Set objExcel = CreateObject("Excel.Application")
  - objExcel.Visible = True
  - objExcel.Workbooks.Add
  - objExcel.Cells(5, 5).Value = "Tresc ze sktyptu"
  - objExcel.Cells(5, 5).Font.Bold = TRUE
  - objExcel.Cells(5, 5).Interior.ColorIndex = 30
  - objExcel.Cells(5, 5).Font.ColorIndex = 2
- Dostęp bezpośredni:
  - (..)
  - objExcel.Workbooks("aa.xls").Sheets("Arkusz1").Range("B10").Value

Skrypty WSH - przykłady (X)

- Dostęp do komend shell'a
  - var shl, folder, plik;
  - shl = new ActiveXObject('Shell.Application');
  - shl.Explore('C:\\');
  - shl.ControlPanelItem('desk.cpl');

  - Set objShell = Wscript.CreateObject("WScript.Shell")
  - objShell.Run "calc.exe"

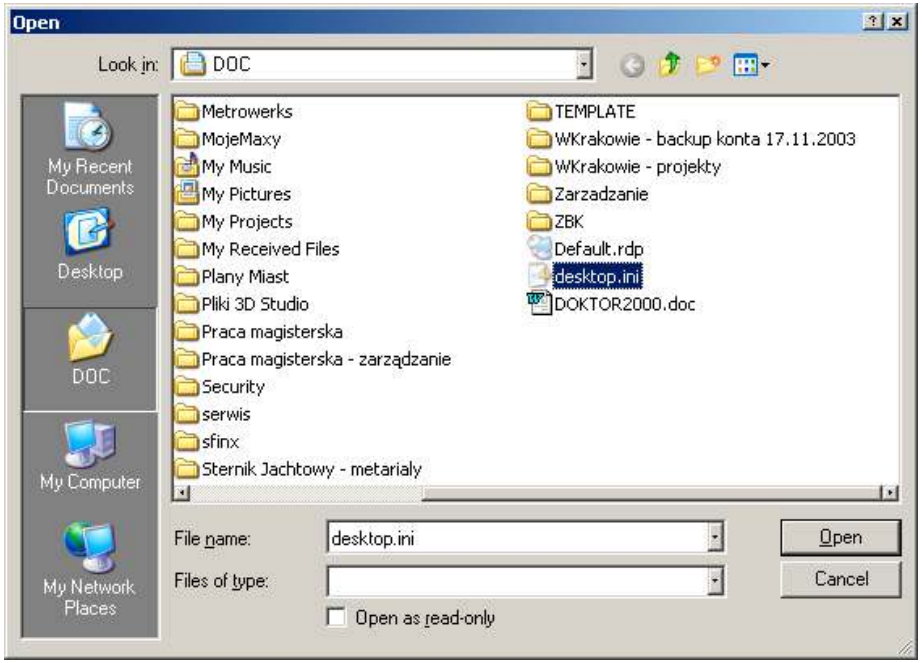
Skrypty WSH - przykłady (XI)

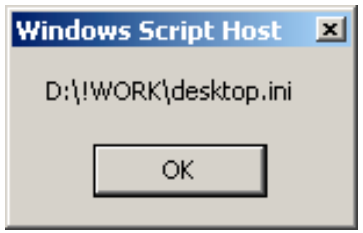
- Wzajemne wywoływanie skryptów i przekazywanie parametrów

```
var fso,
    argumenty = WScript.Arguments;
if (argumenty.Count() == 2) {
fso = new ActiveXObject('Scripting.FileSystemObject');
if (fso.FileExists(argumenty.item(0))) {
    fso.CopyFile(argumenty.item(0), argumenty.item(1), false)
    } else {
    WScript.Echo('Plik nie istnieje');
    }
} else {
    WScript.Echo('Złe argumenty!');
}
```

Skrypty WSH - przykłady (XII)

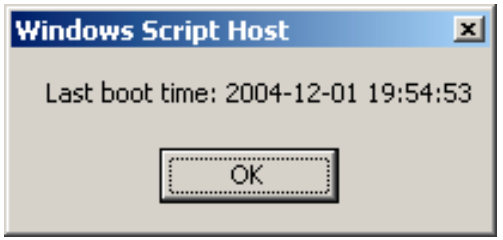
- Posługiwanie się interfejsem okienkowym - pobranie pliku od użytkownika:
  - set oUA = CreateObject("UserAccounts.CommonDialog")
  - oUA.ShowOpen
  - wscript.echo oUA.FileName





Skrypty WSH - przykłady (XIII)

- Dostęp do informacji systemowych;
  - set dateTime = CreateObject("WbemScripting.SWbemDateTime")
  - set OSobjs = GetObject("winmgmts:").InstancesOf("Win32_OperatingSystem")
  - for each obj in OSobjs
  - dateTime.Value = obj.LastBootUpTime
  - wscript.echo "Last boot time:",dateTime.GetVarDate
  - next



Skrypty WSH - przykłady (XIV)

- Zastosowania sieciowe:
  - set WshNetwork = WScript.CreateObject("WScript.Network")
  - WScript.Echo "User Name: " & WshNetwork.UserName
  - WScript.Echo "Computer Name: " & WshNetwork.ComputerName
  - WScript.Echo "Domain Name: " & WshNetwork.UserDomain
  - WshNetwork.MapNetworkDrive "S:", "\\server\share"

Skrypty WSH - przykłady (XV)

- Raportowanie zdarzeń ze skryptu:
  - set WshShell = CreateObject("WScript.Shell")
  - WshShell.LogEvent 0, ” Zadanie wykonano poprawnie"

Typy zdarzenia:

0 SUCCESS,

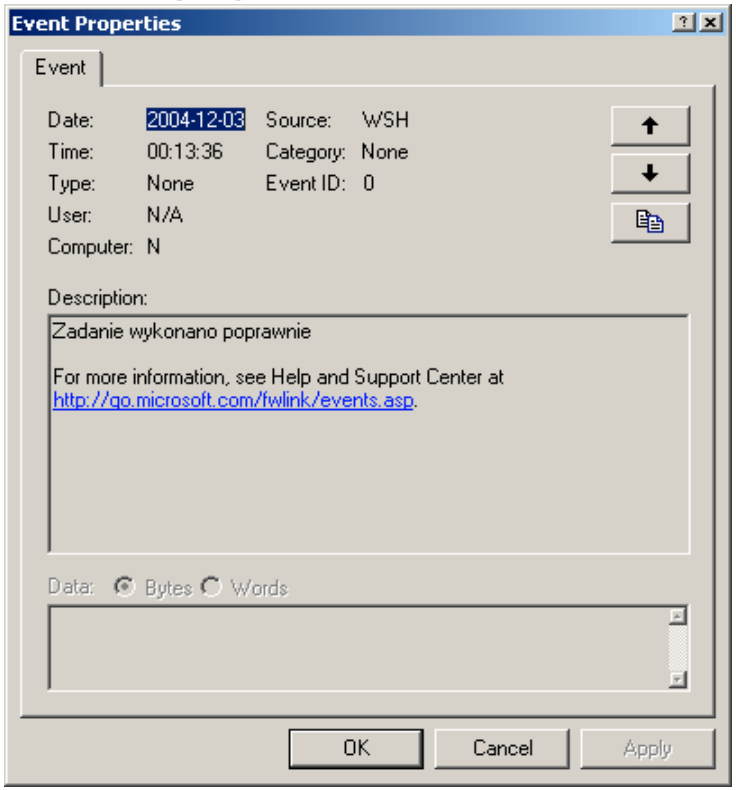
1 ERROR,

2 WARNING,

4 INFORMATION,

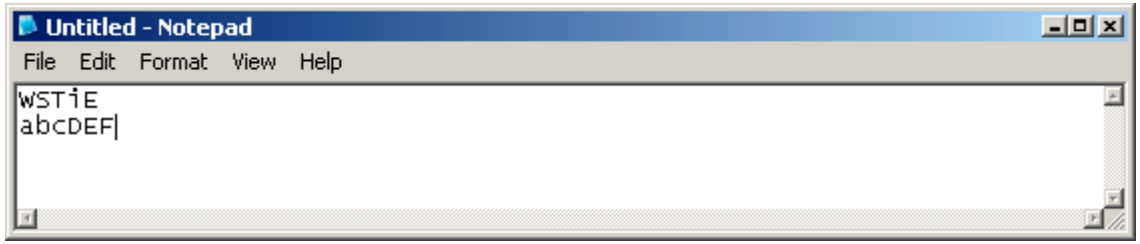
8 AUDIT_SUCCESS,

16 AUDIT_FAILURE



Skrypty WSH - przykłady (XVI)

- Sterowanie aplikacjami poprzez symulowanie komunikatów z konsoli systemowej:
  - Set WshShell = WScript.CreateObject("WScript.Shell")
  - WshShell.Run "notepad.exe"
  - WshShell.AppActivate "Notepad"
  - WshShell.SendKeys "WSTiE"
  - WshShell.SendKeys "{ENTER}"
  - WshShell.SendKeys "abc"
  - WshShell.SendKeys "{CAPSLOCK}"
  - WshShell.SendKeys "def"



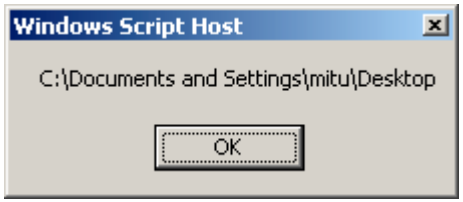
Skrypty WSH - przykłady (XVII)

- Wykorzystanie IE jako interfejsu graficznego

```
Set objExplorer = WScript.CreateObject("InternetExplorer.Application")
objExplorer.Navigate "about:blank"
objExplorer.ToolBar = 0
objExplorer.StatusBar = 0
objExplorer.Width=400
objExplorer.Height = 200
objExplorer.Left = 0
objExplorer.Top = 0
Do While (objExplorer.Busy)
    Wscript.Sleep 50 '* oczekiwanie na gotowosc IE
Loop
objExplorer.Visible = 1
objExplorer.Document.Body.InnerHTML = "Uzycie IE”
Wscript.Sleep 800
objExplorer.Document.Body.InnerHTML = "do prezenacji interfejsu graficznego”
Wscript.Sleep 2000
objExplorer.Quit
```

Skrypty WSH - przykłady (XVIII)

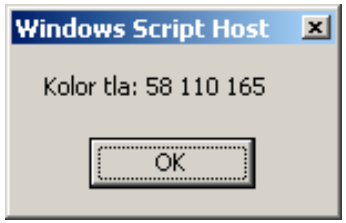
- Dostęp do folderów specjalnych
  - Set WshShell = WScript.CreateObject("WScript.Shell")
  - WScript.Echo WshShell.SpecialFolders("AllUsersPrograms")
  - WScript.Echo WshShell.SpecialFolders(”Desktop")



- Dostęp do folderów systemu
  - set WshShell = CreateObject("WScript.Shell")
  - WScript.Echo WshShell.ExpandEnvironmentStrings("%SystemRoot%")
  - WScript.Echo WshShell.ExpandEnvironmentStrings("%WinDir%")

Skrypty WSH - przykłady (XIX)

- Dostęp do rejestru systemowego:
  - Option Explicit
  - Dim WSHShell, RegKey, Tlo
  - Set WSHShell = CreateObject("WScript.Shell")
  - RegKey = "HKCU\Control Panel\Colors\"
  - Tlo = WSHShell.RegRead(RegKey & "Background")
  - WScript.echo "Kolor tla: " & Tlo
  - WSHShell.RegWrite regkey & "Background", "15 15 15"



Skrypty WSH - przykłady (XX)

- Tworzenie podprocesów z pobraniem produkowanego przez nie wyjścia:
  - Set objShell = WScript.CreateObject("WScript.Shell")
  - Set objExecObject = objShell.Exec("cmd /c dir")
  - Do While Not objExecObject.StdOut.AtEndOfStream
  - strText = objExecObject.StdOut.ReadLine()
  - Wscript.Echo strText
  - Loop

Skrypty WSH - przykłady (XXI)

- Dostęp do baz danych ze skryptu - odczyt z plikowej bazy danych:

```
Const adOpenStatic = 3
Const adLockOptimistic = 3
Set objConnection = CreateObject("ADODB.Connection")
Set objRecordSet = CreateObject("ADODB.Recordset")
objConnection.Open _
    "Provider = Microsoft.Jet.OLEDB.4.0; " & "Data Source = firma.mdb"
objRecordSet.Open "SELECT * FROM KLIENCI",objConnection,
adOpenStatic , adLockOptimistic
Do Until objRecordset.EOF
    Wscript.Echo objRecordset.Fields.Item("MIEJSCOWOSC") & _
        vbTab & objRecordset.Fields.Item("NIP")
    objRecordset.MoveNext
Loop
objRecordSet.Close
objConnection.Close
```

Skrypty WSH - przykłady (XXII)

- Dostęp do baz danych ze skryptu - zapis do plikowej bazy danych:

```
Const adOpenStatic = 3
Const adLockOptimistic = 3
Set objConnection = CreateObject("ADODB.Connection")
Set objRecordSet = CreateObject("ADODB.Recordset")
objConnection.Open _
    "Provider = Microsoft.Jet.OLEDB.4.0; " & "Data Source = firma.mdb"
objRecordSet.Open "SELECT * FROM KLIENCI",objConnection,
adOpenStatic , adLockOptimistic
objRecordSet.AddNew
objRecordSet("NAZWA") = "ATI”
objRecordSet("MIEJSCOWOSC") = "Krakow”
objRecordSet("NIP") = "593-45-65-124”
objRecordSet.Update
objRecordSet.Close
objConnection.Close
```

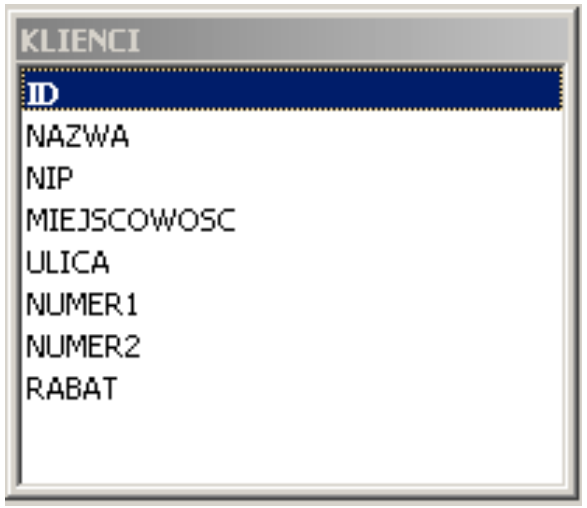
Skrypty WSH - przykłady (XXIII)

- Dostęp do baz danych ze skryptu - dostęp do MS SQL Server poprzez ADO:

```
Const adOpenStatic = 3
Const adLockOptimistic = 3
Set objConnection = CreateObject("ADODB.Connection")
Set objRecordSet = CreateObject("ADODB.Recordset")
objConnection.Open "Provider=SQLOLEDB;User ID=sa;Password=;Initial
Catalog=firma;Data Source=localhost"
objRecordSet.Open "SELECT * FROM KLIENCI",objConnection,
adOpenStatic , adLockOptimistic
objRecordSet.AddNew
objRecordSet("NAZWA") = "ATI”
objRecordSet("MIEJSCOWOSC") = "Krakow”
objRecordSet("NIP") = "593-45-65-124”
objRecordSet.Update
objRecordSet.Close
objConnection.Close
```

Skrypty WSH - przykłady (XXIV)

- Baza danych dla trzech poprzednich przykładów:
- W tabeli „FIRMA” znajdują się między innymi kolumny „MIEJSCOWOSC”, „NAZWA”, „NIP”:



Skrypty WSH - przykład WMI (I)

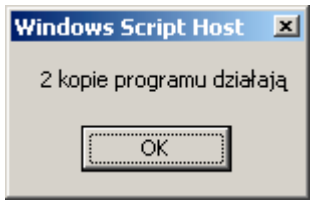
- WMI (*Windows Management Instrumentation*)- mechanizm pełniący rolę interface'u pośredniczącego miedzy użytkownikiem a częściami systemy operacyjnego.
- API, za pomocą którego możemy odwoływać się do obiektów systemowych otrzymując o nich informacje, lub wywoływać funkcje - np. informacje o dyskach /całkowita pojemność, filesystem, pozostałe wolne miejsce/, shutdown komputera i dziesiątki innych
- Możliwość realizowania zapytań do systemów zdalnych !

Skrypty WSH - przykład WMI (II)

- Użycie obiektów WMI:
  - strDate = "20031102000000.000000+000"
  - strComputer = "."
  - Set objWMIService = GetObject _
  - ("winmgmts:\\" & strComputer & "\root\cimv2")
  - Set colFiles = objWMIService.ExecQuery _
  - ("Select * From CIM_DataFile Where CreationDate < '" & strDate & "'")
  - For Each objFile in colFiles
  - Wscript.Echo objFile.Name
  - Next

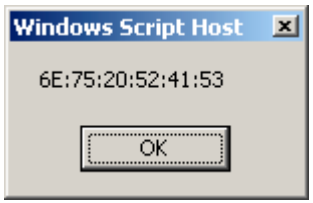
Skrypty WSH - przykład WMI (III)

- Kontrola funkcjonowania aplikacji:
  - set list = getobject("winmgmts:")_
  - .execquery("select * from win32_process where name='Notepad.exe'")
  - If list.count > 0 Then
  - Wscript.Echo list.count & " kopie programu działają"
  - Else
  - Wscript.Echo ”Aplikacja nie działa"
  - End If



Skrypty WSH - przykład WMI (IV)

- Operacje niskiego poziomu (np. pobranie adresu MAC adapterów sieciowych):
  - Set NetAdapterSet = GetObject("winmgmts:{impersonationLevel=impersonate}").ExecQuery("select * from Win32_NetworkAdapter")
    for each NetAdapter in NetAdapterSet
    wscript.echo NetAdapter.MACAddress
    next



Windows PowerShell (I)

- Powłoka Windows (następca Monad)
- Wymagana platforma .NET v 2.0
- Nowa interaktywna linia komend oraz technologia skryptowa systemu Windows oparta na zadaniach
- Przeznaczenie: zarówno wersje desktop jak i server

Windows PowerShell (II)

- Cechy:
  - Dostosowywanie komend i tworzenie swoich własnych narzędzi linii poleceń
  - Duże możliwości manipulacji obiektami,
  - Rozszerzalność komend, która umożliwia administratorom szybkie pisanie własnych skryptów
  - Bezpośrednia kontrola danych systemowych
  - Spore możliwości formatowania wyjścia
  - Obsługa obecnych skryptów i narzędzi linii komend.

Windows PowerShell - Cmdlet (I)

- Cmdlet („command-let”) - analogia komendy. Fragment funkcjonalności PowerShell, uruchamiany jego poleceniem. Nie jest powiązany z programem (np. *.exe) jak dawniej.
- Przykład: Get-Process
- Object Pipeline: Przekazywanie obiektów analogicznie do dawnego przekazywania strumieniu z komendy do komendy. Część Cmdlet’ów jest dostosowanych do występowania tylko jako odbiorcy obiektów (na przynajmniej drugiej pozycji w linii komendy). Znak łączący jak dawniej: |

Windows PowerShell - Cmdlet (II)

- Konwencja nazw Cmdlet'ów:
  czynność-cel
- Gdzie czynność to np.: Get, Set, Write, Test, Update, Clear, Convert, Export, New, Import, Rename itp.
- Cmdlet może posiadać parametry, typowane po znaku „-" z wartościami typowanymi po spacji (notacja Unixowa), np.:
  Get-Process -ProcessName Far

Windows PowerShell - Podstawy (I)

- Zmienne dostępne są po symbolu „$”, np.:
  $env:path, $env:username
- Część zmiennych jest predefiniowana, resztę można definiować samemu
- Komendy mają skrótowe aliasy, np.:
  Get-Process = gps
  Copy-Item = cpi
- Zintegrowaną pomoc dostarcza Cmdlet Get-Help
- Stos lokalizacji, umożliwiający zapamiętywanie miejsc w drzewie katalogów:
  - Push-Location = położenie na stosie
  - Pop-Location = pobranie ze stosu

Windows PowerShell - Podstawy (I)

- Możliwa jest zmiana lokalizacji nie tylko w obrębie drzew katalogów, np.:

  cd Microsoft.PowerShell.Core\Registry\HKCU

- Odwołanie do bieżącego katalogu następuje poprzez zastosowanie przedrostka ./ (podobnie jak w Unixach)
- Cmdlety pomagające poruszać się po strukturze komend, klas, typów, funkcji itp.
  - Get-Member (np. 1 | Get-Member poda szczegóły klasy System.Int32 kolekcji .NET)
  - Get-ChildItem (np. dir | Get-ChildItem spowoduje rekurencyjne wykonanie dir)

Windows PowerShell - Bezpieczeństwo

- PowerShell posiada kilka wariantów polityki bezpieczeństwa, między którymi można go przełączać. Dotyczą one przede wszystkim uruchamiania skryptów. Skrypty można podpisywać certyfikatami - przykład procedury dostępny pod komendą:

  Get-Help about_signing

- Warianty:
  - Restricted (bez skryptów)
  - AllSigned (uruchomić można tylko skrypt podpisany)
  - RemoteSigned (wszystkie skrypty pochodzące z sieci muszą być podpisane)
  - Unrestricted (każdy zkrypt można uruchomić

Windows PowerShell - Bezpieczeństwo i skrypty

- Wariant standardowy: Restricted
- Aby uruchomić jakikolwiek skrypt trzeba zmienić wariant komendą Set-ExecutionPolicy <wariant>, np.:
  Set-ExecutionPolicy RemoteSigned
- Skrypty zapisywane są w pliku z domyślnym rozszerzeniem: *.ps1
- Aby uruchomić skrypt z pliku „test.ps1”:
  - Komenda Windows:
    c:\ PowerShell -command ./test.ps1
  - Komenda PowerShell (tryb interakcyjny):
    PS c:\> ./test.ps1

Konsola WMIC (I)

- WMIC - Windows Management Instrumentation Command-line
- Korzystanie z WMI przy pomocy wiersza poleceń. Podstawowe funkcje, głównie wyświetlanie informacji
- Dostęp z poziomu konsoli do kolekcji WMI
- Możliwość użycia pseudo SQL'a do filtrowania wyników
- Konsola WMIC dostępna jest tylko w wersjach typu Professional oraz server

Konsola WMIC (II)

- Proste życie WMIC (wyświetlanie danych):
  WMIC *<przelaczniki>* /locale:ms_409 *<alias>*
  gdzie <alias> to synonim kolekcji WMI
  przykładowo:
  COMPUTERSYSTEM, DESKTOP, DISKDRIVE, ENVIRONMENT, GROUP, LOGON, NETUSE, NETCLIENT, OS, PRINTER, PROCESS, REGISTRY, SERVER, SERVICE, SHARE, SOFTWAREELEMENT, STARTUP, SYSACCOUNT, SYSDRIVER, VOLUME i wiele innych
- Uwaga: opcja /locale:ms_409 jest wymagana gdy wersja językowa systemu jest inna niż English U.S.

Konsola WMIC (III)

- Komunikacja z maszyną zdalną:
  WMIC /NODE:128.0.0.5 /USER:xxx
  /PASSWORD:pass /locale:ms_409 <alias>
- Przełączniki /NODE, /USER, /PASSWORD umożliwiają autoryzację na maszynie zdalnej

Konsola WMIC (IV)

- Zlecenia WMIC:
  - Po <alias> można zdefiniować zlecenie dot. kolekcji (zależnie od jej typu):
    - CALL - Wykonuje metody,
    - CREATE - Tworzy nową instancję i ustawia dla niej wartości właściwości. Zlecenie to nie służy do tworzenia nowych klas,
    - DELETE - Usuwa bieżącą instancję lub zbiór instancji. Za pomocą tego zlecenia można usuwać klasy,
    - GET - Pobiera informacje na temat określonych właściwości,
    - LIST - Pokazuje dane (domyślne),
    - SET - Operacje konfigurowania właściwości.

Konsola WMIC (V)

- Jak uzyskać informacje o możliwościach danej kolekcji:
- Przykładowo (dla kolekcji OS):
  - WMIC /locale:ms_409 OS CALL /? - produkuje listę komend (w przykładzie dla zlecenia CALL) wraz z możliwymi parametrami

Konsola WMIC (VI)

- Przykłady:
  - WMIC /NODE:128.0.0.4 /USER:xxx /PASSWORD:pass /locale:ms_409 STARTUP WHERE (Caption="desktop") - pobranie listy programów automatycznie uruchamianych podczas startu w filtrowaniem (Caption="desktop")
  - WMIC /locale:ms_409 OS GET bootdevice, csname - pobranie z OS wartości bootdevice, csname
  - WMIC /locale:ms_409 SERVICE where caption='TELNET' CALL STARTSERVICE - uruchomienie usługi telnet
  - WMIC /NODE:128.0.0.4 /USER:user /PASSWORD:s /locale:ms_409 PROCESS CALL Create "NET USER xxx /ACTIVE:no” - zablokowanie konta użytkownika xxx na zdalnej maszynie (pośrednio: uruchomienie zdalnego procesu!)
  - WMIC /locale:ms_409 OS GET SerialNumber - pobranie numeru seryjnego OS

Konsola WMIC (VII)

- Przykłady (CD):
  - WMIC /locale:ms_409 process where „name=‘explorer.exe‘“ call terminate
  - WMIC /locale:ms_409 process call create „calc.exe“
  - WMIC /locale:ms_409 process call create „c:\\temp\\test.bat“
  - WMIC /locale:ms_409 service get Name, DisplayName
  - WMIC /locale:ms_409 os call reboot
  - WMIC /locale:ms_409 baseboard get Product - wersja płyty głównej
  - WMIC /locale:ms_409 datafile where „name=‘c:\\boot.ini‘“ list full - pełna informacja o pliku
  - WMIC /locale:ms_409 Product get name - spis oprogramowania
  - WMIC /output:c:\temp.html /locale:ms_409 service list brief /format:hform - zapis wyników do pliku

Wybrane komendy systemowe - cacls (I)

- Sterowanie uprawnieniami do plików i folderów
- Użycie: CACLS ścieżka <opcje>
- Opcje:
  - /T - z podfolderami
  - /G *uzytkownik:uprawnienie* - przyznanie uprawnień
  - /P *uzytkownik:uprawnienie* - zmiana uprawnień
  - /D *uzytkownik* - cofnięcie wszystkich uprawnień
  - /E - modyfikuje tylko typowanego użytkownika
- Rodzaje uprawnień:
  - N (None), R (Read), W (Write), C (Change = read+write), F (Full control)

Wybrane komendy systemowe - cacls (II)

- Przykłady:
  - CACLS plik.txt /E /G ”user1":F - przyznanie pełnych uprawnień
  - CACLS plik.txt /E /G ”user2":R - przyznanie uprawnień do odczytu
  - CACLS plik.txt /E /P ”user1":F - cofnięcie uprawnień
- Uwaga na opcję /E - brak tej opcji spowoduje ustawienie praw wg komendy i jednoczesne usunięcie wszystkich innych ustawień praw do danego pliku

Wybrane komendy systemowe - cacls (III)

- Uprawnienia poprzez wyłączenie - wykonujemy kolejno:

    1. cacls plik.txt /E /P user:N - cofnięcie wszystkiego
    2. cacls plik.txt /E /G user:R - zezwolenie na odczyt

  - i osiągamy możliwość zezwolenia użytkownikowi na odczyt zabraniając mu wszystkiego innego przy jednoczesnym pełnym zezwoleniu dla innych. Jest to tzw uprawnienie specjalne typu „odmowa” (deny)

Wybrane komendy systemowe - cacls (IV)

- Uprawnienia poprzez wyłączenie - efekt działania poprzednich komend:

cacls *
C:\plik.txt KOMP1\user:(DENY)(dostęp specjalny:)
DELETE
WRITE_DAC
WRITE_OWNER
FILE_WRITE_DATA
FILE_APPEND_DATA
FILE_WRITE_EA
FILE_DELETE_CHILD
FILE_WRITE_ATTRIBUTES
BUILTIN\Administratorzy:F
ZARZĄDZANIE NT\SYSTEM:F
KOMP1\User:F
BUILTIN\Użytkownicy:R

Wybrane komendy systemowe - openfiles

- Analiza (lokalna lub zdalna) otwartych plików udostępnionych.
- Możliwość odłączania plików (parametr /disconnect)
- Możliwość zarządzania zdalną maszyną (parametr /s *komputer, /u domena|użytkownik)*

```
D:\>openfiles /query /v /s s

Hostname        ID       Accessed By  Type       #Locks     Open Mode       Open File (Path)

=============== ======== ========================== ========== =============== ===============
s               7071     SA           Windows    0          Read            C:
s               7077     SA           Windows    0          Write + Read    \PIPE\srvsvc
```

Wybrane komendy systemowe - MSIexec

- Instalator pakietów i elementów Windows
- Możliwość przeprowadzania instalacji pakietów uszkodzonych lub wybrakowanych
- Możliwość przeprowadzenia weryfikacji wcześniejszej instalacji pakietu
- Możliwość weryfikacji pakietów, odzyskiwania danych i naprawy uszkodzonych pakietów
- Możliwość prowadzenia dedykowanych instalacji pakietów na podstawie plików transformacji (*.mst)

Wybrane komendy systemowe - start (I)

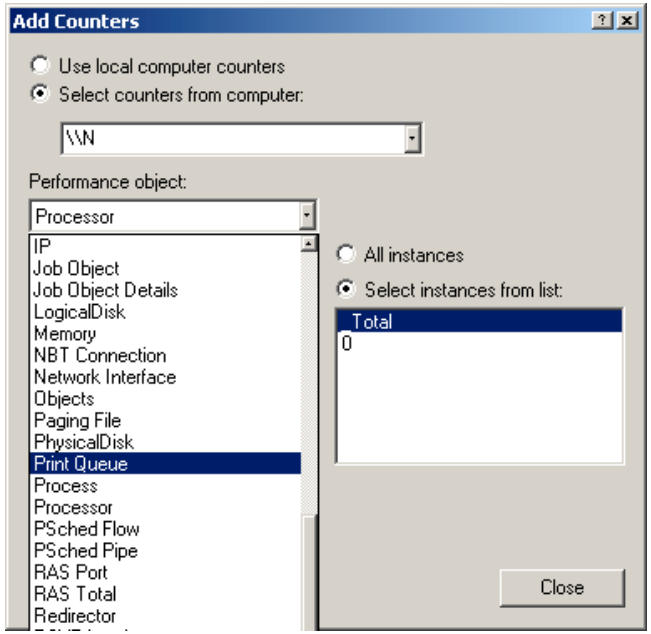
- Uruchomienie programu lub powielenie wiersza poleceń (bez parametrów). Parametry:
  - /d*ścieżka* - określa katalog startowy.
  - /i - przekazuje środowisko startowe programu cmd.exe do nowego okna wiersza polecenia.
  - /min - uruchamia nowe zminimalizowane okno wiersza polecenia.
  - /max - uruchamia nowe zmaksymalizowane okno wiersza polecenia.
  - /separate - uruchamia programy 16-bitowe w oddzielnym obszarze w pamięci.
  - /shared - uruchamia programy 16-bitowe we współużytkowanym obszarze pamięci.

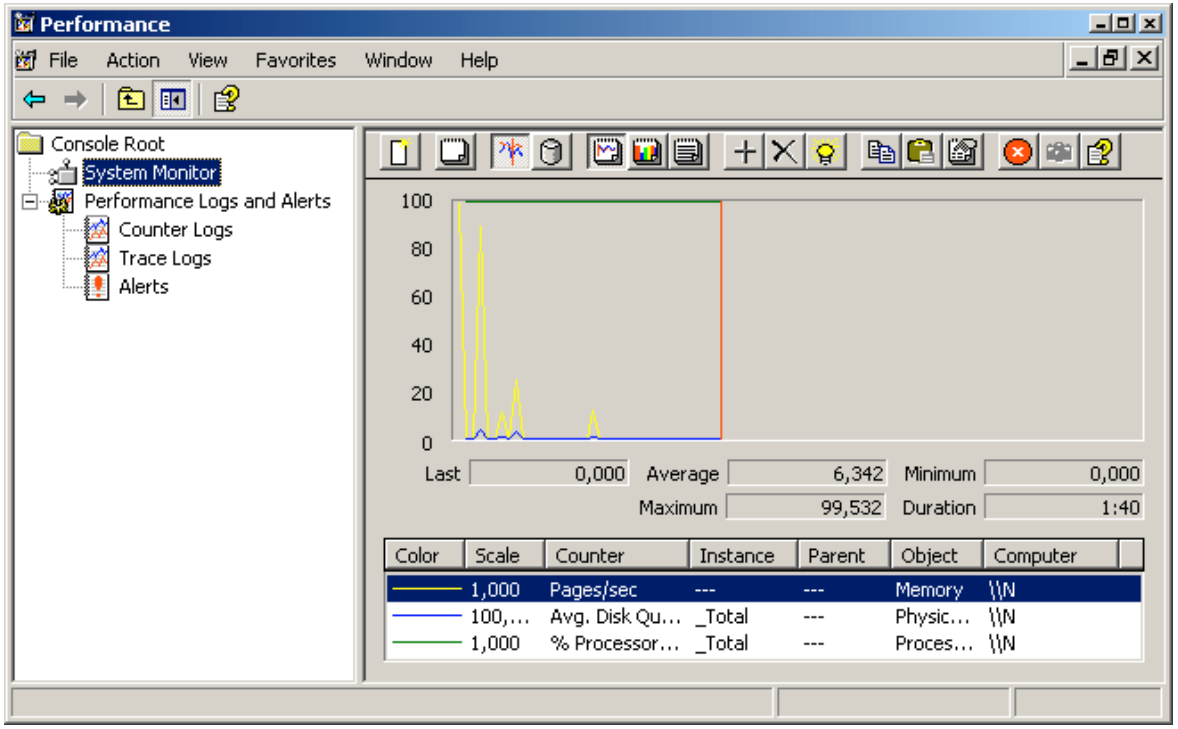
Wybrane komendy systemowe - start (II)

- Parametry (cd):
  - /low - uruchamia aplikację w klasie priorytetu bezczynności.
  - /normal - uruchamia aplikację w klasie normalnego priorytetu.
  - /high - uruchamia aplikację w klasie wysokiego priorytetu.
  - /realtime - uruchamia aplikację w klasie priorytetu czasu rzeczywistego.
  - /abovenormal, /belownormal - uruchamia aplikację w klasie priorytetu powyżej lub poniżej normalnego.
  - /wait - uruchamia aplikację i oczekuje na zakończenie aplikacji.

Wybrane komendy systemowe - perfmon

- Analizator wydajności systemu. Obecnie - okienkowa wersja graficzna.





Wybrane komendy systemowe - pathping

- Połączenie ping i tracert - tworzy analizy statystyczne strat pakietów na poszczególnych bramkach w dłuższym okresie czasu

```
Tracing route to galaxy.agh.edu.pl [149.156.96.9]
over a maximum of 30 hops:
  0  n [128.0.0.3]
  1     *        *        *
Computing statistics for 25 seconds...
            Source to Here   This Node/Link
Hop  RTT    Lost/Sent = Pct  Lost/Sent = Pct  Address
  0                                           n [128.0.0.3]
                                 100/ 100 =100%   |
  1  ---     100/ 100 =100%     0/ 100 =  0%  n [0.0.0.0]

Trace complete.
```

Wybrane komendy systemowe - SFC

- Kontroler plików systemowych
- Polecenie skanuje i weryfikuje wersje wszystkich chronionych plików systemowych
- Ważniejsze parametry:
  - /scannow - natychmiast skanuje wszystkie chronione pliki systemowe.
  - /scanboot - skanuje wszystkie chronione pliki systemowe przy każdym ponownym uruchamianiu komputera.
  - /revert - przywraca domyślny tryb skanowania.

Wybrane komendy systemowe - assoc, ftype

- Komendy zarządzają przypisaniami typów plików.
- ftype przypisuje (lub wyświetla) komendę, jaka będzie stosowana wobec danego pliku
- assoc = przypisuje lub wyświetla przypisanie rozszerzenia do nazwy danego pliku
- Aby kasować przypisania należy postawić znak spacji po ‘=‘

Wybrane komendy systemowe - fsutil

- Zestaw narzędzi do zarządzania systemami plików (FAT, NTFS)
- Parametry:
  - behavior - ustawienia trybu,
  - dirty - zarządzania flagą autocheck
  - hardlink - zarządzanie linkami do plików
  - quota - kwoty dyskowe
  - sparse - zarządzanie plikami rozrzedzonymi

Wybrane komendy systemowe - compact

- Zarządzanie kompresją NTFS
- Parametry:
  - /c - kompresuje określony katalog lub plik.
  - /u - dekompresuje określony katalog lub plik.

```
D:\s>compact

 Listing D:\s\
 New files added to this directory will not be compressed.


Of 0 files within 1 directories
0 are compressed and 0 are not compressed.
0 total bytes of data are stored in 0 bytes.
The compression ratio is 1,0 to 1.
```

Wybrane komendy systemowe - cipher

- Zarządzanie szyfrowaniem NTFS
- Parametry:
  - /e - szyfruje określony katalog.
  - /u - deszyfruje określony katalog.

```
D:\s>cipher /e

 Listing D:\FILM\s\
 New files added to this directory will be encrypted.
```

Program Control (I)

- Umożliwienie dostępu z linii komend do wszelkich graficznych paneli konfiguracyjnych systemu.
- Jako parametr pobiera pliki *.cpl - będące składnikami „Panelu Sterowania”, „Ustawień Pulpitu” itp., oraz komponenty z HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Control Panel\Cpls , będące dodatkami do Panelu Sterowania.
- Parametry wywołania (separator parametrów to przecinek):
  - control.exe plik.cpl @nr_okna, nr_zakładki

Program Control (II)

- Moduły *.cpl:
  - access.cpl - Accessibility controls
  - appwiz.cpl - Add/Remove Programs
  - desk.cpl - Display properties
  - hdwwiz.cpl - Add hardware
  - inetcpl.cpl - Configure Internet Explorer and Internet properties
  - intl.cpl - Regional settings
  - joy.cpl - Game controllers
  - main.cpl - Mouse properties and settings
  - main.cpl,@1 - Keyboard properties
  - mmsys.cpl - Sounds and Audio
  - ncpa.cpl - Network properties

Program Control (III)

- Moduły *.cpl (CD):
  - nusrmgr.cpl - User accounts
  - powercfg.cpl - Power configuration
  - sysdm.cpl - System properties
  - timedate.cpl - Date and time properties
- Przykłady wywołań:
  - control.exe desk.cpl - wyświetlenie ustawień pulpitu
  - control.exe sysdm.cpl,,2 - wyświetlenie zakładki 2 („Sprzęt”) „Ustawień Systemu”

Korzystanie ze zmiennych systemowych (I)

- Nazwa zmiennej zawarta wewnątrz symboli ‘%’
- Możliwość użytkowania w wywołaniach z cmd, skryptach lub jako parametry programów
- Wprowadzanie modyfikacji - poleceniem **set**
- Ważniejsze zmienne:
  - %ALLUSERSPROFILE% - Zwraca lokalizację profilu wszystkich użytkowników.
  - %APPDATA% - Zwraca lokalizację, w której aplikacje domyślnie przechowują dane.
  - %CD% - Zwraca ciąg bieżącego katalogu.
  - %COMPUTERNAME% - Zwraca nazwę komputera.
  - %COMSPEC% - Zwraca dokładną ścieżkę do pliku wykonywalnego powłoki poleceń.

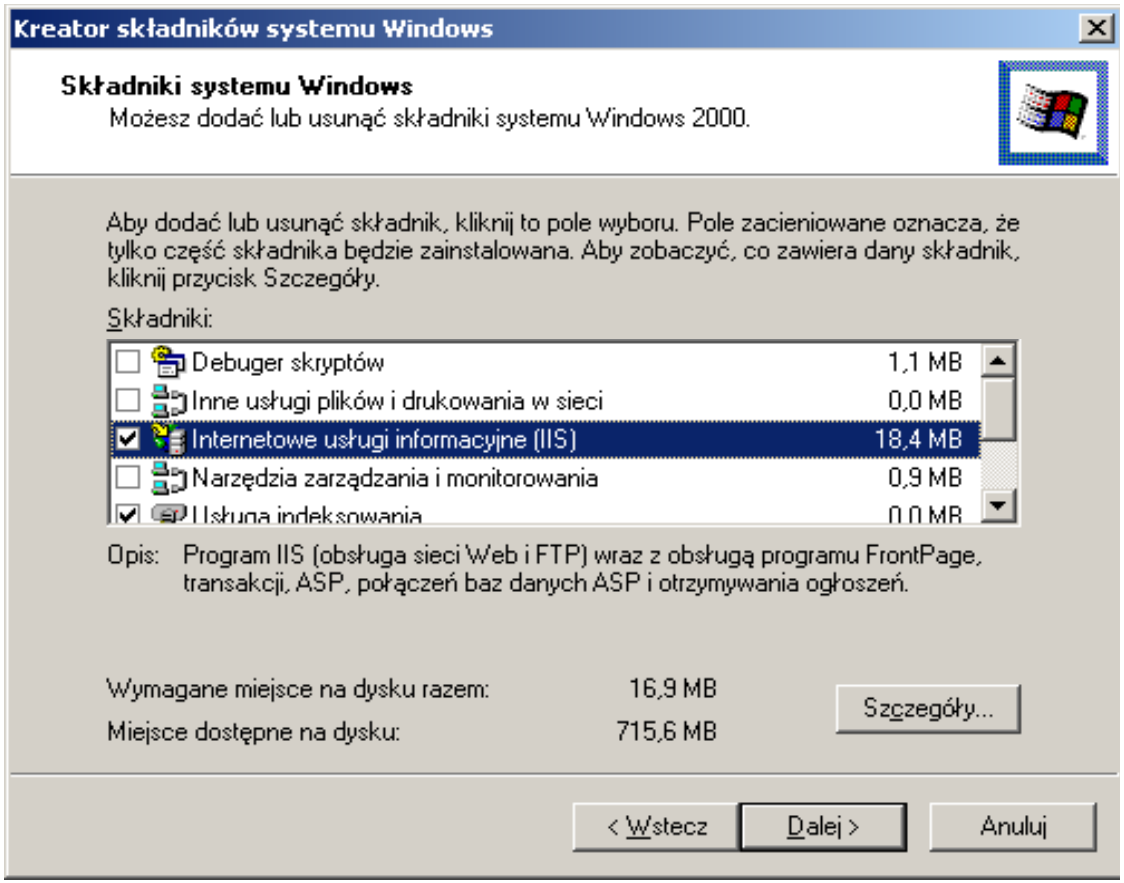
Korzystanie ze zmiennych systemowych (II)

- Ważniejsze zmienne (cd):
  - %DATE% - Zwraca bieżącą datę.
  - %OS% - Zwraca nazwę systemu operacyjnego.
  - %PATH% - Określa ścieżkę wyszukiwania plików wykonywalnych.
  - %PATHEXT% - Zwraca listę rozszerzeń nazw plików rozpoznawanych jako wykonywalne przez system operacyjny.
  - %PROMPT% - Zwraca ustawienia wiersza polecenia bieżącego interpretera.
  - %RANDOM% - Zwraca losowy numer dziesiętny z zakresu od 0 do 32 767.
  - %TEMP% i %TMP% - Zwraca domyślne katalogi tymczasowe używane przez aplikacje dostępne dla użytkowników, którzy są aktualnie zalogowani.

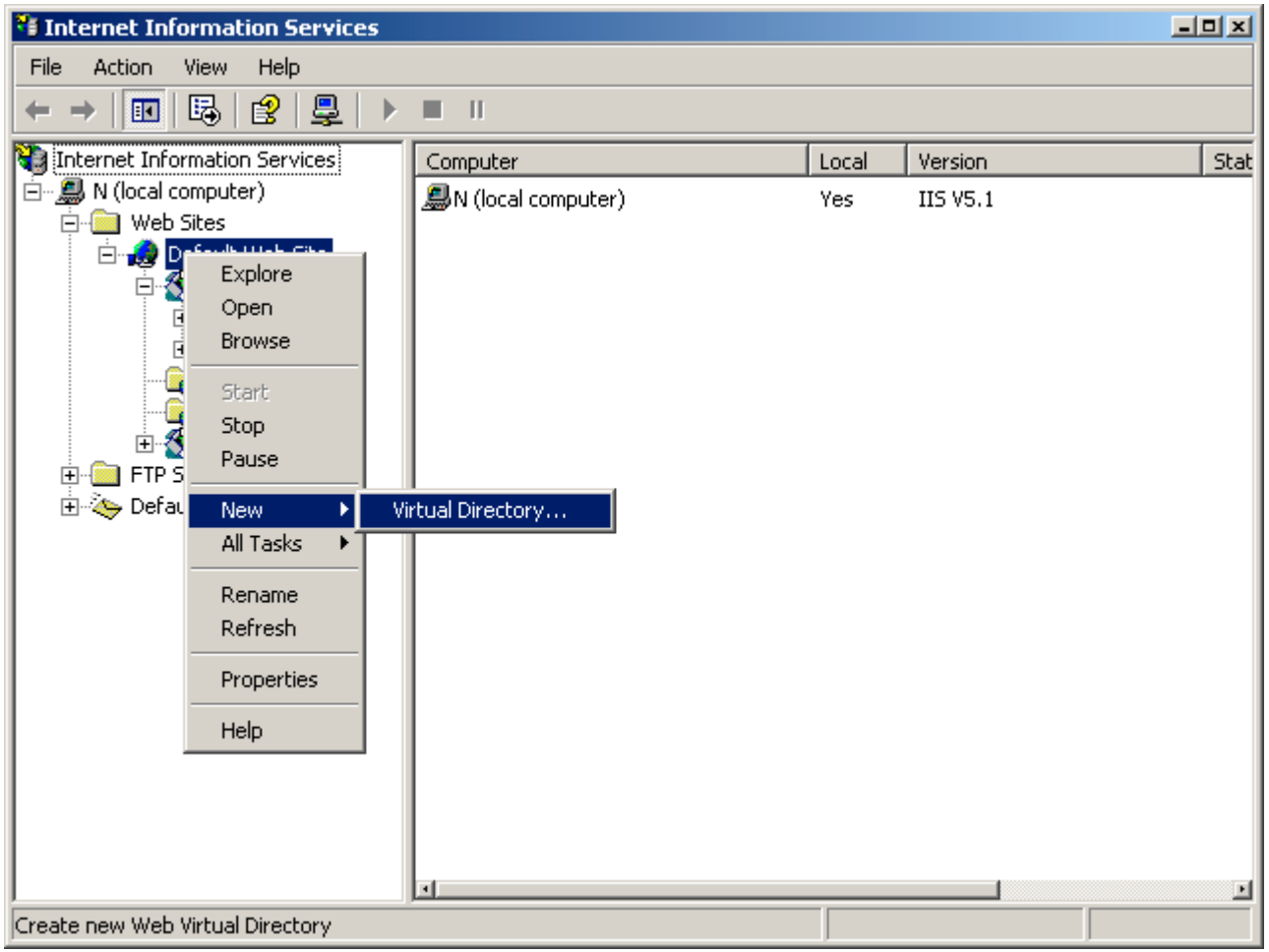
Korzystanie ze zmiennych systemowych (III)

- Ważniejsze zmienne (cd):
  - %TIME% - Zwraca bieżącą godzinę.
  - %USERNAME% - Zwraca nazwę aktualnie zalogowanego użytkownika.
  - %USERPROFILE% - Zwraca lokalizację profilu bieżącego użytkownika.
  - %WINDIR% - Zwraca lokalizację katalogu systemu operacyjnego.

Instalacja IIS



Konfiguracja IIS - katalogi wirtualne (I)

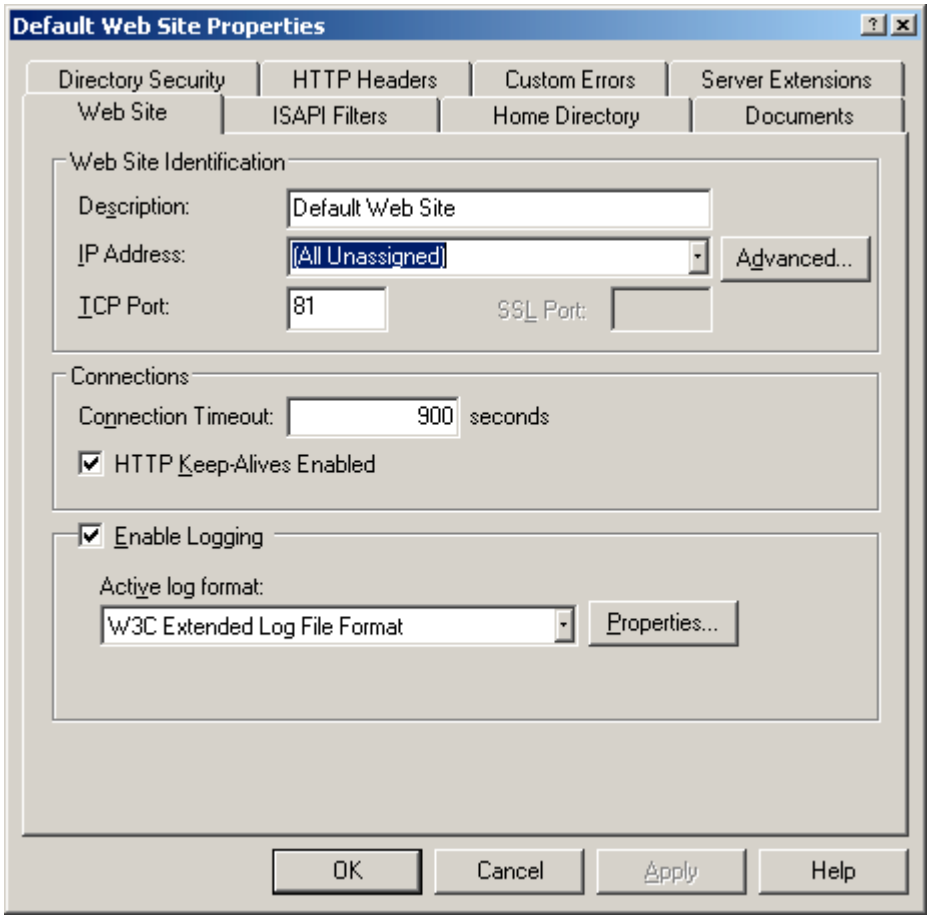


Konfiguracja IIS - katalogi wirtualne (II)

- Dodanie nowego katalogu wirtualnego wymaga podania ścieżki fizycznej do katalogu skryptów oraz aliasu tej ścieżki używanego w URL
- W ramach katalogu obowiązuje drzewo podkatalogów analogicznie do katalogu wskazywanego ścieżką fizyczną
- W katalogach wirtualnych możemy umieszczać kolejne katalogi wirtualne

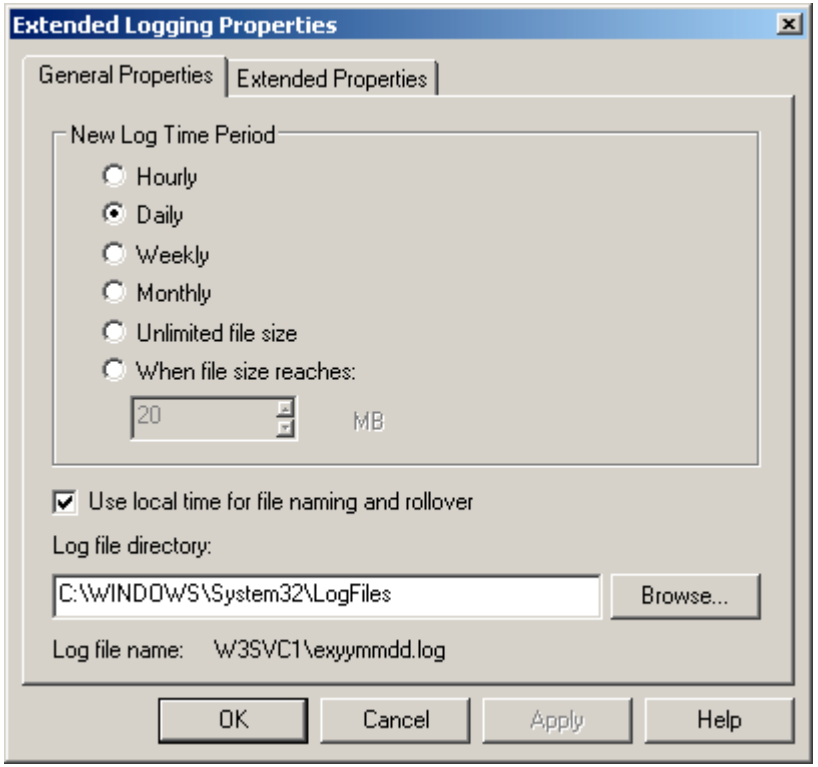
IIS - konfiguracja witryny głównej (I)

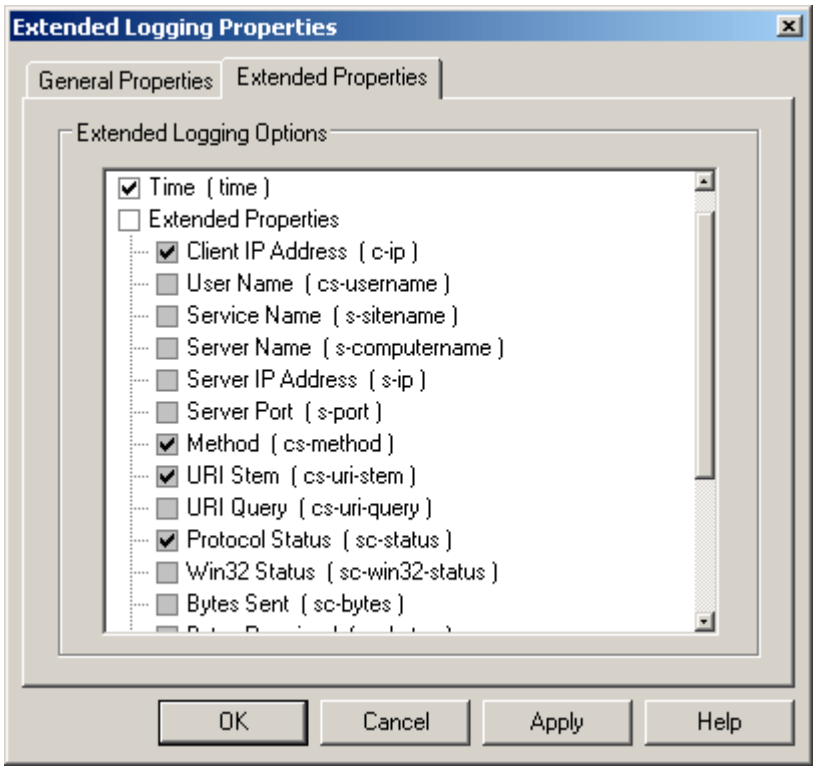
- IP i port HTTP, Timeout, ustawienia audytu



IIS - konfiguracja witryny głównej (II)

- Audyt zdarzeń - logi serwera HTTP



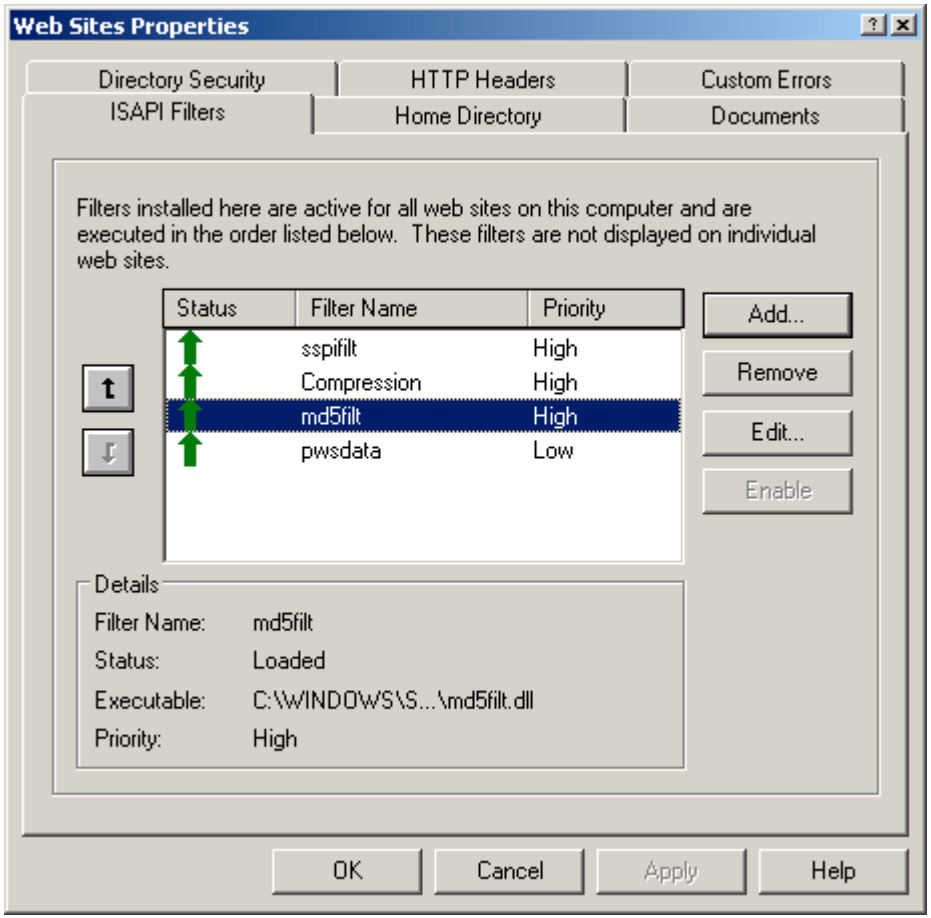


IIS - konfiguracja witryny głównej, przykład logów HTTP

- #Version: 1.0
- #Date: 2004-10-06 19:44:35
- #Fields: time c-ip s-ip cs-method cs-uri-query sc-status sc-bytes cs-bytes time-taken cs-host cs(User-Agent)
- 19:44:35 128.0.0.3 128.0.0.3 GET - 200 628 451 100 128.0.0.3:81 Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1)
- 19:44:38 128.0.0.3 128.0.0.3 GET - 200 628 290 80 128.0.0.3:81 Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1)
- 19:44:40 128.0.0.3 128.0.0.3 GET - 302 301 419 0 128.0.0.3:81 Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1)
- 19:44:40 128.0.0.3 128.0.0.3 GET - 200 1089 420 0 128.0.0.3:81 Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1)

IIS - konfiguracja witryny głównej (III)

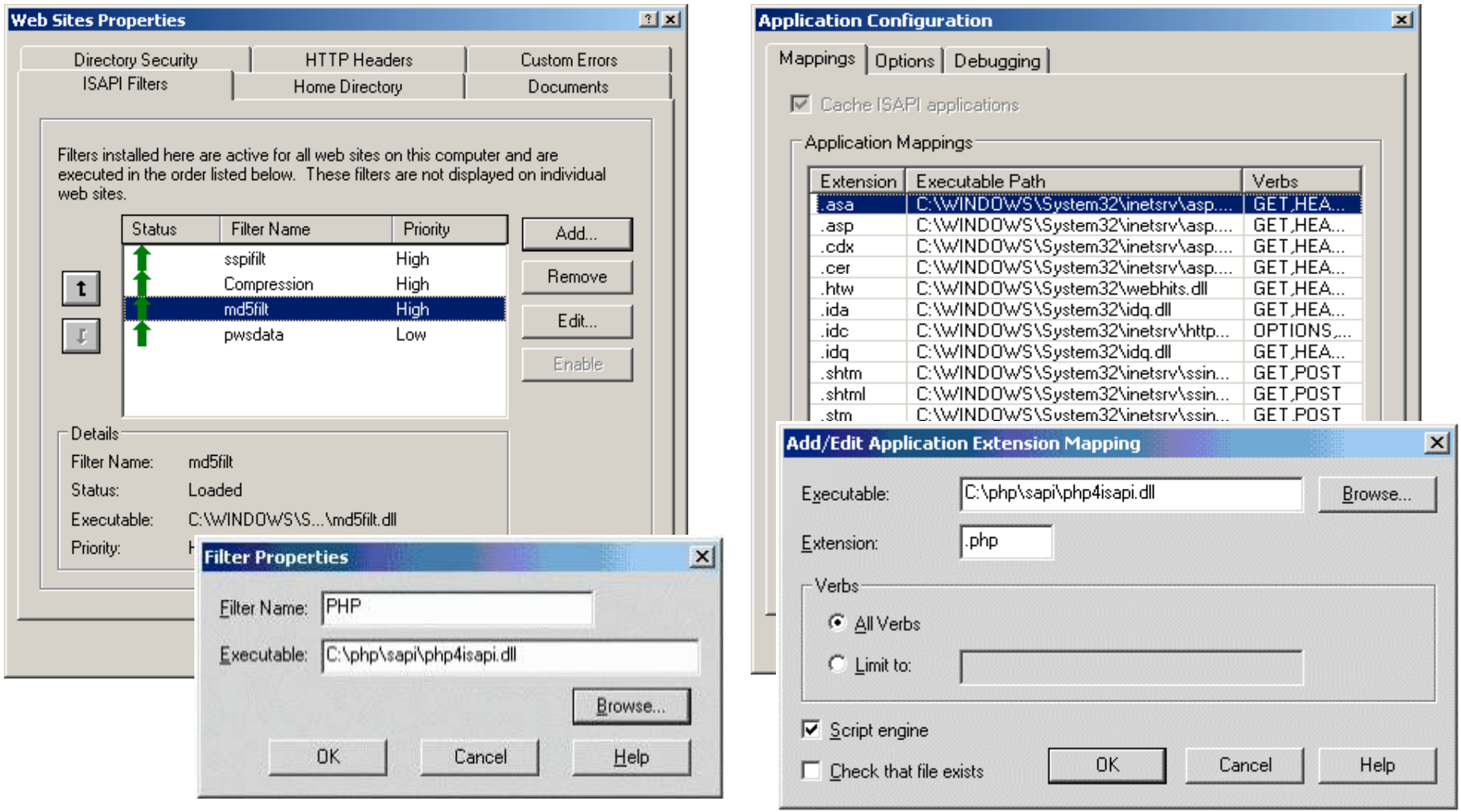
- Filtry obsługujące zdarzenia podczas przetwarzania zapytań i wtyczki przetwarzania.



IIS - przykład instalacji ISAPI do PHP

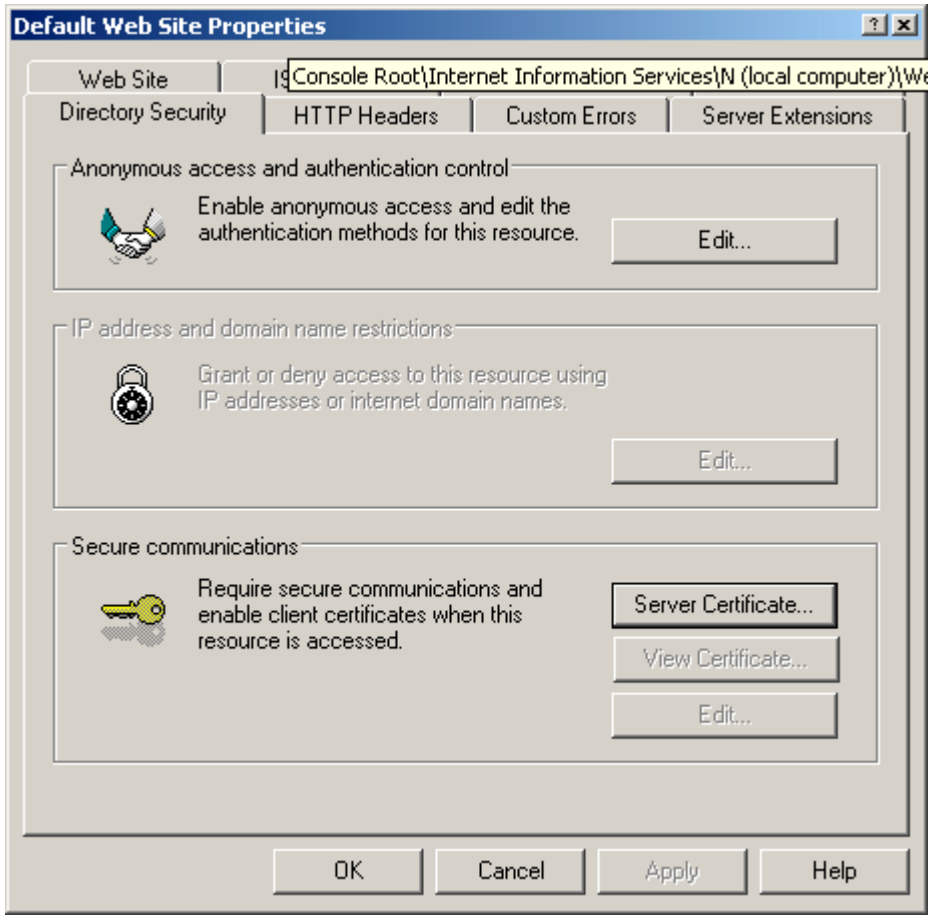
- Należy:
  - Dodać nową wtyczkę ISAPI (Wtyczki zapisane są jako biblioteki plikowe - DLL).
  - Zdefiniować nowe rozszerzenie aplikacji (.php) ze wskazaniem na tą wtyczkę
  - Wykonać restart serwera:
    - net stop iisadmin /y
    - net start w3svc

IIS - przykład instalacji ISAPI do PHP (I)



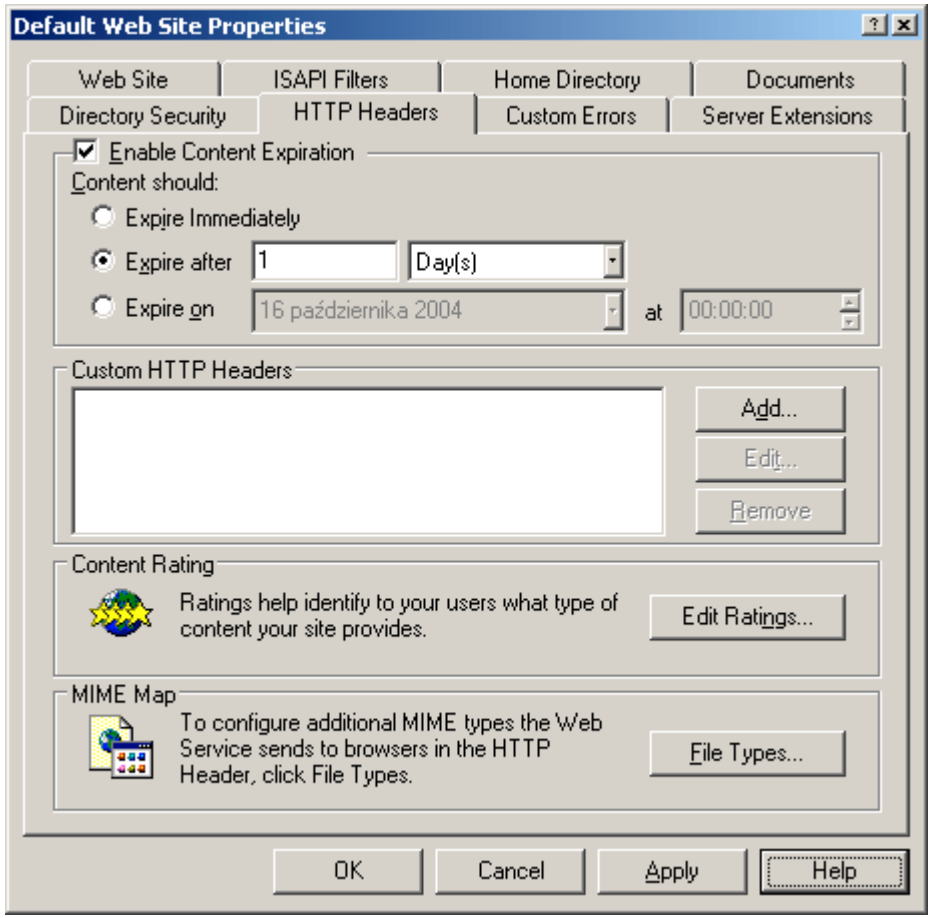
IIS - konfiguracja katalogów (I)

- Zabezpieczenia: Dostęp anonimowy, zabronione adresy IP, certyfikaty Microsoft serwera



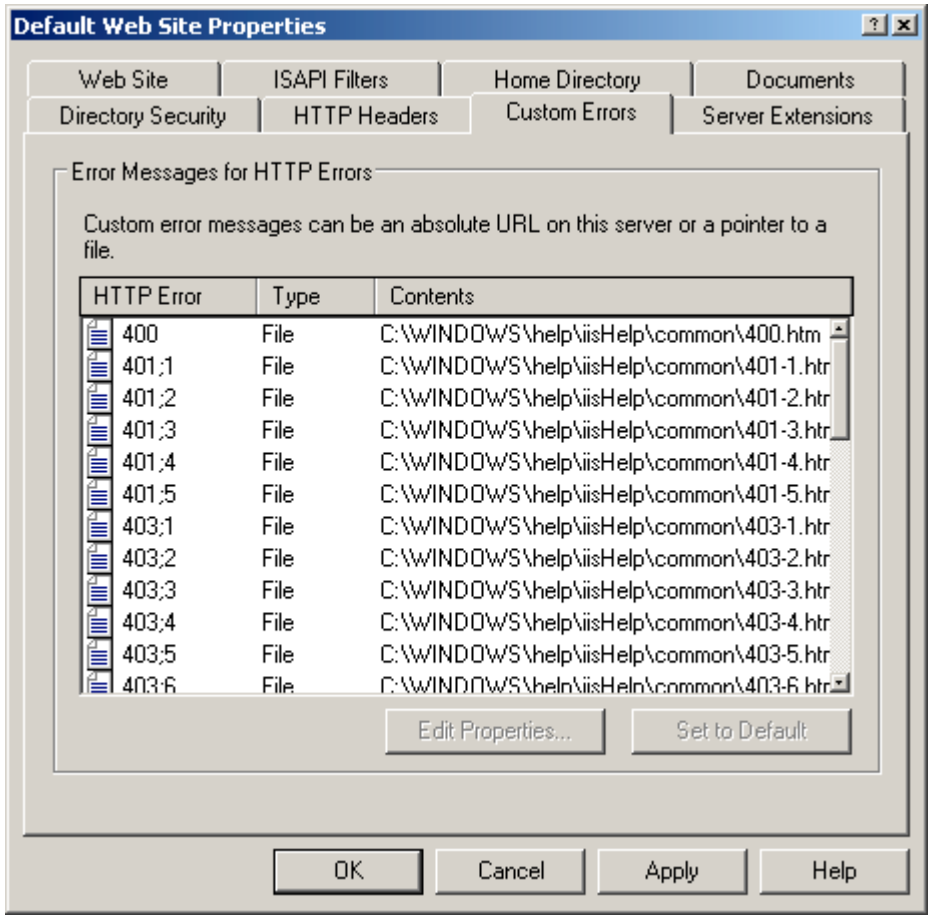
IIS - konfiguracja katalogów (II)

- Wygasanie treści u klienta, dodatkowe nagłówki HTTP, charakterystyka treści, Multipurpose Internet Mail Extensions



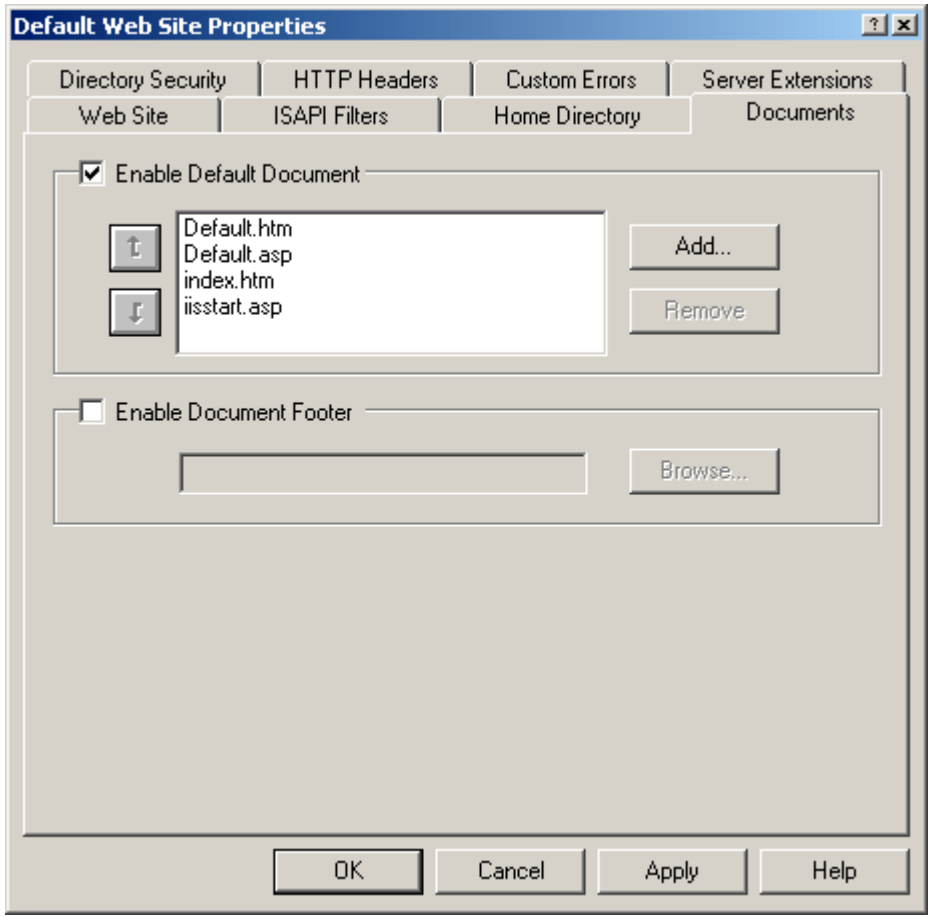
IIS - konfiguracja katalogów (III)

- Pliki prezentacji błędów HTTP



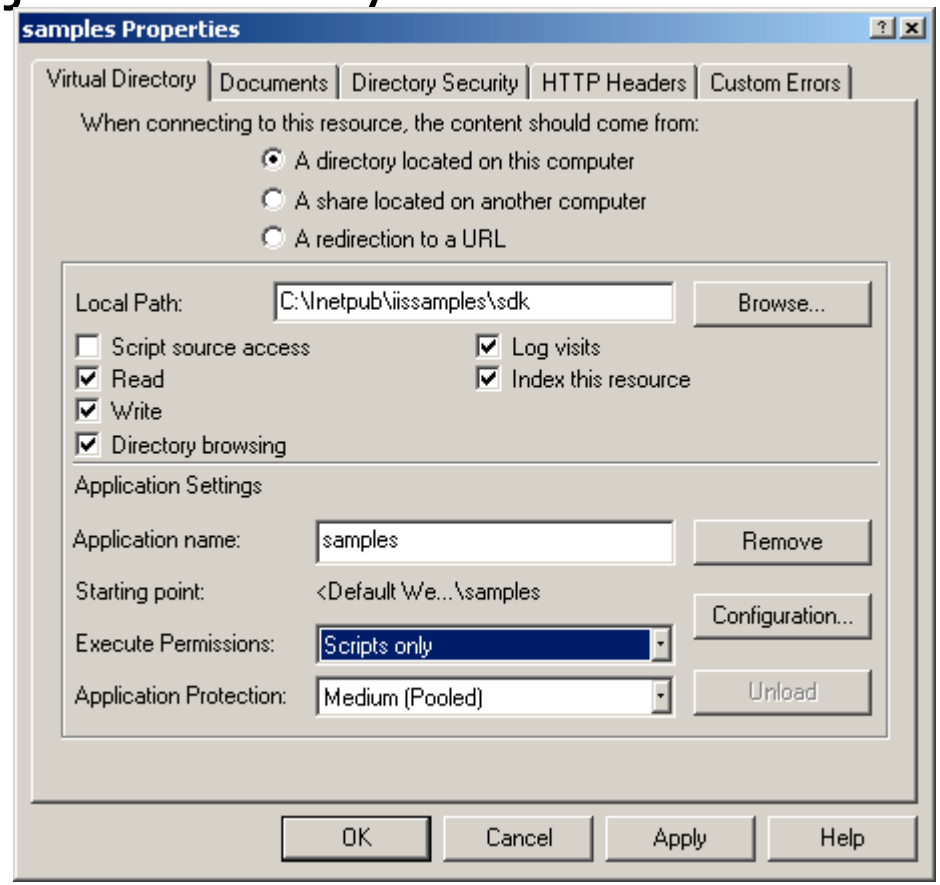
IIS - konfiguracja katalogów (IV)

- Typy plików rozpoznawane jako startowe



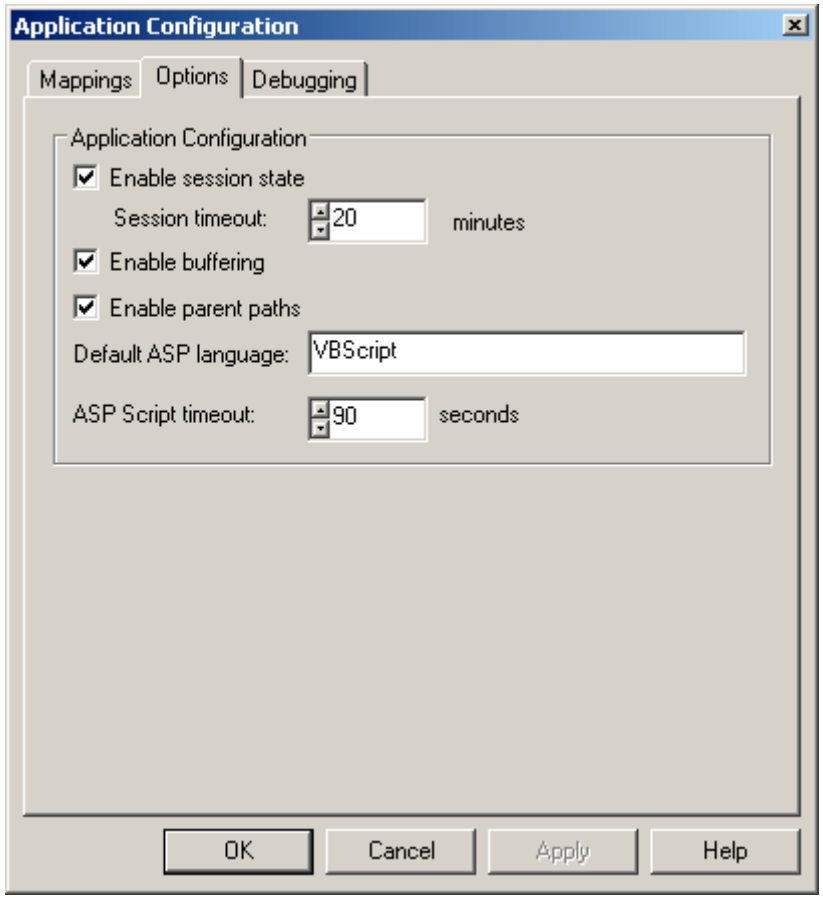
IIS - konfiguracja katalogów (V)

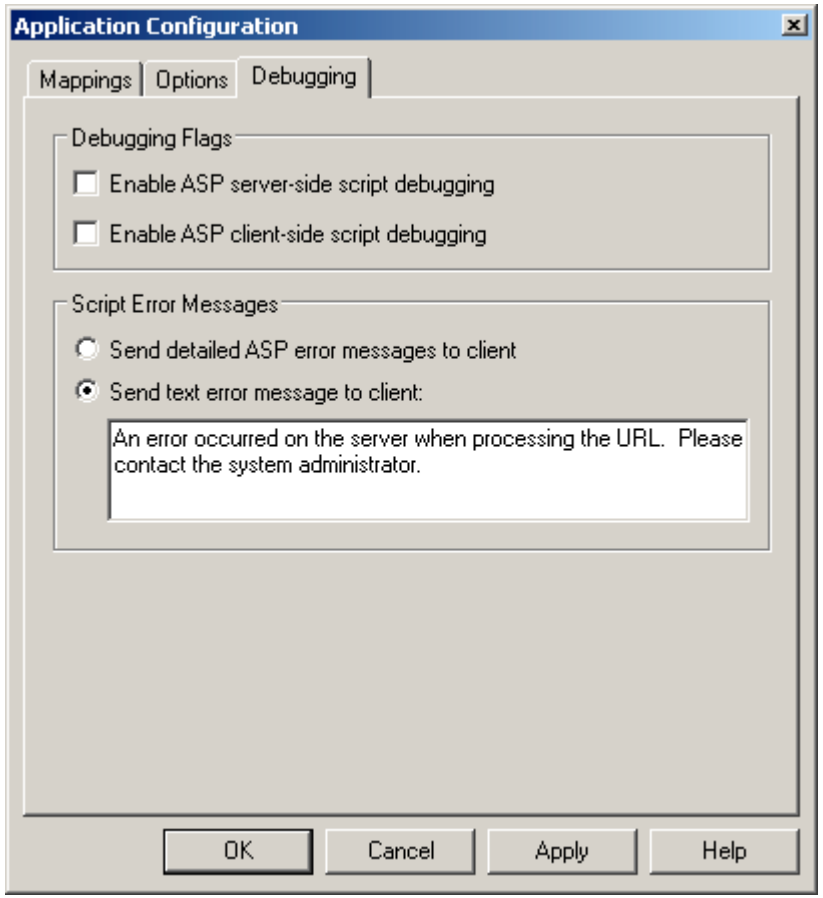
- Pochodzenie przekierowania, prawa dostępu, ograniczenia dla aplikacji internetowych



IIS - konfiguracja katalogów (VI)

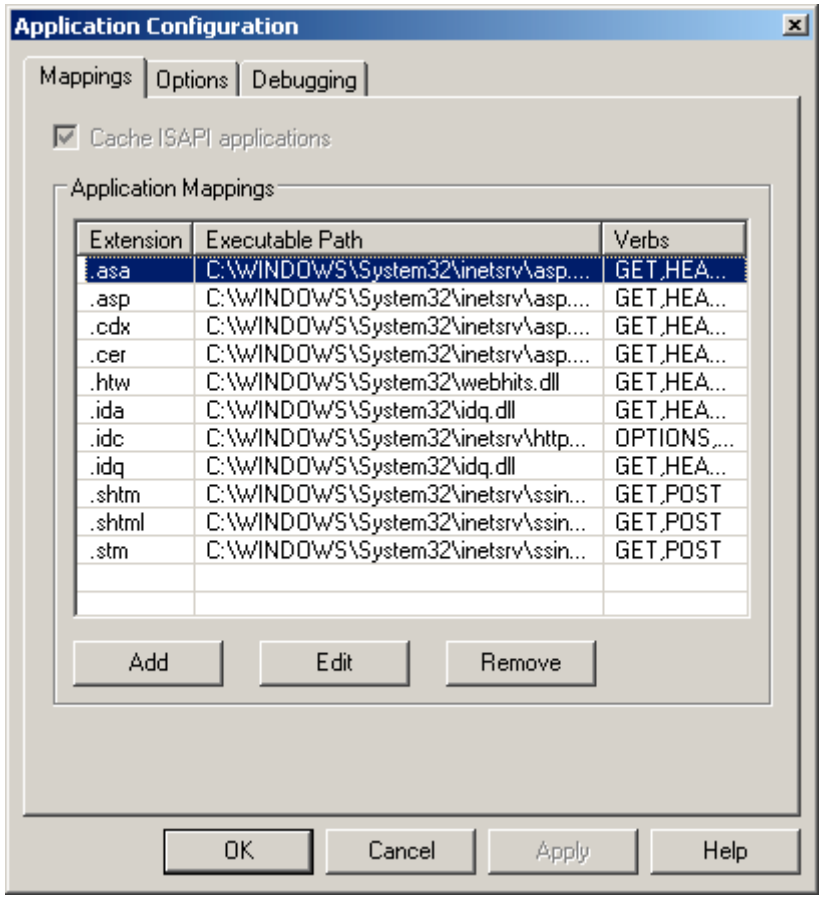
- Konfiguracja aplikacji internetowych (cd)

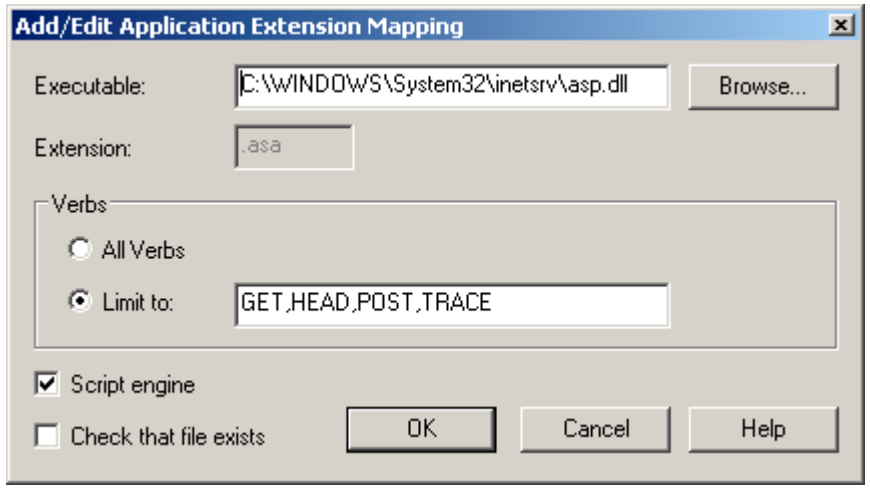




IIS - konfiguracja katalogów (VII)

- Konfiguracja aplikacji internetowych





IIS - ASP

- IIS posiada skryptowe wspomaganie generowania publikowanych dokumentów typu server-side - o nazwie ASP (*Active Server Pages*)
- Tworząc skrypty ASP można standardowo korzystać z dwóch języków tj. VBScript oraz Jscript, lecz możliwe dalsze w postaci wtyczek

ASP - standardowe obiekty

- **Application**- przechowuje informacje o stanie aplikacji.
- **Session** - przechowuje informacje dotyczące jednego użytkownika, jest "magazynem" informacji każdego użytkownika odwiedzającego stronę.
- **Request** - zawiera wszystkie informacje, które są wysyłane z przeglądarki do serwera. Zawiera dane, które zostały wysłane przy użyciu formularza (form).
- **Response** - wysyła HTML i różne informacje, łącznie z cookies i nagłówki, z powrotem do przeglądarki (klienta).
- **Server** - zwiększa funkcjonalność Active Server Pages.

ASP - wybór języka

- Na początku każdego skryptu ASP powinna znaleźć się linijka**:**

  <% @ Language=ScriptingLanguage %>

- Standardowe możliwości:
  - *<% @ Language=VBScript %>*
  - *<% @ Language=JScript %>*

ASP - tworzenie skryptu

- Przeplatanie skryptu z HTML:
  - Wszystkie elementy, które dotyczą zadeklarowanego języka powinny znaleźć się pomiędzy znacznikami <% oraz %>.
  - Do danego pliku ASP można dołączać inne pliki poprzez wpisanie następującej komendy:
    <!-#include file="plik"->
    , przy czym *plik* może być ścieżką odniesienia.

ASP - Warunki i pętle (I)

- Warunek *if* :
  - If warunek Then
  - [instukcje]
  - [ElseIf warunek n-ty Then
  - [instrukcje]]
  - [Else
  - [instrukcje]]
  - End If

ASP - Warunki i pętle (II)

- Przykład dla *if* :
  - If Hour(Now) < 12 Then
  - Response.Write "Good morning!"
  - ElseIf Hour(Now) >= 18 Then
  - Response.Write "Good evening!"
  - Else
  - Response.Write "Good afternoon!"
  - End If

ASP - Warunki i pętle (III)

- Warunek *case* :
  - Select Case wyrażenie
  - [Case wartosc
  - [instrukcje]] ...
  - [Case Else wartosc
  - [instrukcje]]
  - End Select

ASP - Warunki i pętle (IV)

- Przykład dla *case* :
  - Select Case Hour(Now)
  - Case 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11
  - Response.Write "Good morning!"
  - Case 12, 13, 14, 15, 16
  - Response.Write "Good afternoon!"
  - Case Else
  - Response.Write "Good evening!"
  - End Select

ASP - Warunki i pętle (V)

- Pętla *for..next*
  - For licznik = start To koniec [Step step]
  - [instrukcje]
  - [Exit For]
  - [instrukcje]
  - Next

ASP - Warunki i pętle (VI)

- Przykład dla *for..next :*
  - For i=0 to 10
  - Response.Write ”Hello!"
  - Next

ASP - Warunki i pętle (VII)

- Pętla *while..wend* :
  - While warunek
  - [instrukcje]
  - Wend

ASP - Warunki i pętle (VIII)

- Przykład dla *while..wend* :
  - i=0
  - While i <= 10
  - Response.Write "Witaj świecie!"
  - i=i+1
  - Wend

ASP - Warunki i pętle (IX)

- Pętla *Do ... Loop* :
  - Do [{While | Until} warunek]
  - [instrukcje]
  - [Exit Do]
  - [instrukcje]
  - Loop

ASP - Warunki i pętle (X)

- Przykład dla *Do ... Loop* :
  - Do
  - Response.Write "Witaj świecie!"
  - i=i+1
  - Loop Until i > 10

ASP - Warunki i pętle (XI)

- Pętla *For Each ...* :
  - For Each element In grupa|zbiór
  - [instrukcje]
  - Next

ASP - Warunki i pętle (XII)

- Przykład dla *For Each* ... :
  - For Each item In Request.Form
  - Response.Write Request.Form(item)
  - Next

ASP - zmienne

- Brak definicji typu zmiennych
- Przykład deklaracji: *Dim zmienna*
- Gdy użyto dyrektywy Option Explicit - dozwolone są tylko zmienne zadeklarowane: <% Option Explicit %>
- Rozróżniamy pojedyncze zmienne i grupy (kolekcje identycznych strukturalnie obiektów po których można iterować)

ASP - operacje na różnych typach zmiennych (I)

- Operacje na łańcuchach:
  - Symbol konkatenacji (operator łączenia) łańcuchów: **&**
  - Trim(łańcuch) - wycina wszystkie spacje znajdujące się na początku łańcucha do momentu napotkania innego znaku
  - Replace(łańcuch, ciąg1, ciąg2) - zamienia każdy ciag1 na ciag2 napotkany w łańcuchu
- isNumeric(x) - zwraca wartość TRUE jeśli x jest liczbą, w przeciwnym wypadku zwraca wartość FALSE
- CInt(liczba) - konwertuje liczbę na liczbę całkowitą
- CLng(liczba) - konwertuje liczbę na liczbę całkowitą długą

ASP - operacje na różnych typach zmiennych (II)

- Operacje zmiennych reprezentujących czas:
  - Now - pobranie bieżącego czasu
  - DateDiff("h", Now, #1/1/2000#) - różnica algebraiczna czasów
  - DateAdd("h", Now, #1/1/2000#) - suma algebraiczna czasów
  - isDate(x) - zwraca wartość TRUE jeśli x jest datą, w przeciwnym wypadku zwraca wartość FALSE

ASP - przekazywanie parametrów do funkcji

- Możliwe jest przekazywanie przez wartość lub przez referencję:
  - Function f1(ByRef param1, ByVal param2)
  - param1 = 10
  - param2 = 15
  - End Function
- Nie musimy stosować () przy przekazywaniu do funkcjo wartości zmiennych lub wartości natychmiastowych:
  - f1 (zmienna1, zmienna2)
  - f1 zmienna1, zmienna2
  - f2 5,”Lancuch”

ASP - komunikacja z użytkownikiem (I)

- Obiekt Response:
  - Respone.Write(tekst)
  - Response.Buffer = TRUE - buforowanie wyjścia do użytkownika
  - Response.Expires = 30 - maksymalny czas wykonania skryptu (zabezpieczenie w przypadku zapętlenia)
  - Response.Flush
  - Response.Clear
  - Response.Redirect(”nowa_lokalizacja.asp")

ASP - komunikacja z użytkownikiem (II)

- Obiekt Request :
  - Zawiera informacje przysłane przez przeglądarkę użytkownika (np zawartość wypełnionego formularza w HTML)
  - Metoda GET: *Request.QueryString("pole_1")*
  - Metoda POST: *Request.Form(" pole_1")*

ASP - operacje na plikach (I)

- Obiekt FileSystemObject - klasa operująca na plikach
- Przykład:
  Set PlikObj = Server.CreateObject("Scripting.FileSystemObject")
- Obiekt TextStream - strumień zapisu/odczytu z pliku
- Możliwość pracy na plikach w systemach kodowania: ASCII, UNICODE, systemowy.

ASP - operacje na plikach (II)

- Otwieranie pliku:

  Set PlikStream = PlikObj .OpenTextFile(Nazwa, TrybPracy, NowyPlik, Kodowanie),

- Parametry:
  - *Nazwa - ścieżka do pliku*
  - *TrybPracy - odczyt = 1, zapis = 2, dopisywanie = 8*
  - *NowyPlik - utwórz plik = 1*
  - *Kodowanie - UNICODE = -2, ASCII = -1, systemowy = 0*
- Zamykanie pliku:

  PlikStream.Close

ASP - operacje na plikach (III)

- Metody operujące na plikach:
  - Read - odczytuje określoną ilość znaków z obiektu TextStream i zwraca ją w postaci łańcucha
  - ReadLine - odczytuje całą linię z obiektu TextStream i zwraca ją w postaci łańcucha bez znaku końca linii
  - ReadAll - odczytuje cały plik TextStream i w rezultacie go zwraca
  - Write - zapisuje łańcuch do pliku TextStream
  - WriteLine - zapisuje łańcuch do pliku TextStream i dodaje znak końca linii
  - Close - zamyka obiekt TextStream

ASP - operacje na plikach (IV)

- Metody pobierające wartości atrybutów dla odczytu/zapisu plików:
  - AtEndOfLine - zwraca wartość TRUE jeśli wskaźnik pliku TextStream znajduje się przed znakiem końca linii, a FALSE w przeciwnym wypadku
  - AtEndOfStream - zwraca wartość TRUE jeśli wskaźnik pliku znajduje się na końcu pliku
  - Column - zwraca wartość określającą bieżącą kolumnę, w której znajduje się wskaźnik pliku
  - Line - zwraca wartość określającą bieżącą linię, w której znajduje się wskaźnik pliku

ASP - operacje na plikach (V)

- Przykład - wyświetlenie pliku:
  - Set PlikObj = Server.CreateObject("Scripting.FileSystemObject")
  - Set PlikStream = PlikObj.OpenTextFile("C:\1.txt", 1, 0)
  - Response.Write "<P>PLIK 1.txt:</P>" & VbCrLf
  - Do While PlikStream .AtEndOfStream <> True
  - strLine = PlikStream .ReadLine
  - strLine = Server.HTMLEncode(strLine)
  - Response.Write strLine & "
" & VbCrLf
  - Loop
  - PlikStream.Close
  - Set PlikStream = Nothing
  - Set PlikObj = Nothing

ASP - operacje na katalogach (I)

- Obiekt *Folder* - klasa operująca na plikach
- Przykład:
  Sciezka = Server.MapPath("/")
  Set PlikObj =
  Server.CreateObject("Scripting.FileSystemObject")
  Set KatalogObj = PlikObj.GetFolder(Sciezka)
- Listy
  KatalogObj .Subfolders i *KatalogObj .Files*
  umożliwiają dostęp do poszczególnych plików i podkatalogów rozważanego folderu

ASP - operacje na katalogach (II)

- Copy - kopiuje określony plik lub katalog z jednego miejsca na drugie
- Delete - usuwa określony plik lub katalog
- Move - przenosi określony plik lub katalog z jednego miejsca na drugie

ASP - operacje na katalogach (III)

- Przykład - wyświetlenie zawartości katalogu
  - Function PokazKatalog(ByRef KatalogObj)
  - Dim Katalog, Plik
  - For Each Katalog In KatalogObj.SubFolders
  - Response.Write "<B>" & Katalog.Name & "</B>
"
  - Next
  - For Each Plik In KatalogObj.Files
  - Response.Write i & Plik.Name & "
"
  - Next
  - End Function

ASP - Wysyłanie poczty (I)

- Do prawidłowego funkcjonowania należy zainstalować w IIS usługę SMTP
- Obiekt CDONTS - klasa operująca na wiadomościach email
- Przykład:
  Set MojaPoczta = Server.CreateObject(CDONTS.NewMail)
- Metoda Send (AddrOdbiorcy, AddrNadawcy, Temat, Treść, Priorytet)

ASP - Wysyłanie poczty (II)

- Przykład - wysłanie wiadomości
  - Dim MojaPoczta
  - Set MojaPoczta = Server.CreateObject(CDONTS.NewMail)
  - MojaPoczta.Send ”user1@domena.com", ”user2@domena.com", "Moja pierwsza poczta", "Pozdrowienia od webmastera."
  - Set MojaPoczta = Nothing

ASP - obiekt Session (I)

- Każde odwiedzenie strony przez użytkownika (klienta) jest reprezentowane przez obiekt Session
- Unikatowy numer sesji zapisywany w zmiennej SessionID w momencie otwarcia sesji
- W sesji można przechowywać własne zmienne:
  - Session(”Wiek") = 12
  - Session(”Imie") = „Jan”

ASP - obiekt Session (II)

- Atrybuty obiektu Session
  - SessionID - zwraca wartość identyfikatora sesji
  - TimeOut - określa maksymalny czas nieaktywności użytkownika po jakim sesja zostaje przerwana, określany w minutach
  - CodePage - numer strony kodowej w jakiej generowane są dokumenty i jaka powinna być stosowana przez klienta
- Metoda Abandon - wywołanie tej metody spowoduje usunięcie wszystkich zasobów obiektu Session

ASP - Cookies (I)

- Data ważności Cookie
- Operacje zapisu Cookie należą do obiektu Response
- Operacje odczytu Cookie należą do obiektu Request
- Pojedyncze pliki z rozszerzeniem .txt. W swojej nazwie zawierają nazwę identyfikującą użytkownika oraz nazwę identyfikującą lokalizację serwera
- W danym Cookie można przechowywać wiele wartości - struktura drzewiasta ...

ASP - Cookies (II)

- Przykład zapisania Cookie:
  Response.Cookies("Produkt")("Nazwa") = ”Lodówka"
  Response.Cookies("Produkt")(”Cena") = 1090

- Przykład odczytania Cookie:
  nazwa = Request.Cookies("Produkt")("Nazwa")

- Przykład kontroli, czy Cookie ma więcej niż jeden element:
  If Request. Cookies("Produkt").HasKeys Then ...

ASP - dostęp do baz danych

- **ADO** *(ActiveX Data Objects)* - zbiór interfejsów (poziom aplikacji), które umożliwiają dostęp do danych OLE DB przy użyciu dowolnego języka programowania
- **OLE DB** (*Object Linking and Embedding BataBase*) - zbiór interfejsów na poziomie systemu, umożliwiających dostęp do różnych danych i źródeł danych
- **ODBC** (Open Database Connectivity) - systemowy interfejs umożliwiający dostęp aplikacjom do skonfigurowanych zasobów - baz danych, plików tekstowych
- Z punktu widzenia aplikacji OLE DB i ODBC to rozwiązania alternatywne

ASP - ADO

- Parametry obiektów ADO:
  - Connection - stanowi połączenie z źródłem danych
  - Recordset - zawiera rekordy zwrócone po wykonaniu zapytania na bazie danych
  - Field - zawiera dane z pojedynczej kolumny i informacje na temat tych danych; obiekt Recordset zawiera grupę Fields, która z kolei zawiera wszystkie obiekty Field
  - Error - zawiera rozszerzoną informację o zwracanych błędach
  - Command - dodatkowe komendy szablonowane - do wielokrotnego wykorzystania

ASP - procedura dostępu do baz danych

- Otwarcie połączenia z bazą danych
- Stworzenie obiektu zapewniającego dostęp do danych każdego rekordu
- Wymiana danych z bazą
- Zamknięcie połączenia, usunięcie obiektu

ASP - łańcuch połączenia w dostępie do bazy danych

- ADO bazuje na treści tzw. łańcuchów połączeń (*connection string*).Daje to uniwersalną metoda na konfigurowanie połączeń do różnorodnych źródeł danych danych.
- W zależności o zawartości łańcucha połączenia ADO uruchamia połączenie z użyciem ODBC lub OLE DB
- Przykład łańcucha: *„DSN=Northwind;Database=Northwind;UID=sa;PWD=;"*

ASP - Recordset w dostępie do bazy danych

- Recordset jest grupą obiektów (wierszy), będących grupami innych obiektów (pól) stanowiącą odpowiednik lub odwzorowanie tabeli, albo widoku w bazie danych
- Może być sprzężony z tabelą lub być jedynie wynikiem wykonania zapytania SQL
- Metody obiektu Recordset:
  - MoveNext, MoveFirst, MoveLast
  - AddNew
  - Delete
  - Requery i atrybut Filter
  - Update

ASP - dostęp do bazy danych w pliku lokalnym (I)

- Przykład - dodawanie i usuwanie rekordu:
  - filePath = Server.MapPath("authors.mdb")
  - Set oConn = Server.CreateObject("ADODB.Connection")
  - oConn.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & filePath
  - oConn.Execute "insert into authors (author, YearBorn) values ('Paul Enfield', 1967)"
  - Set oRs= Nothing
  - oConn.Execute "Delete From authors where author='Paul Enfield' and YearBorn = 1967 "

ASP - dostęp do bazy danych w pliku lokalnym (II)

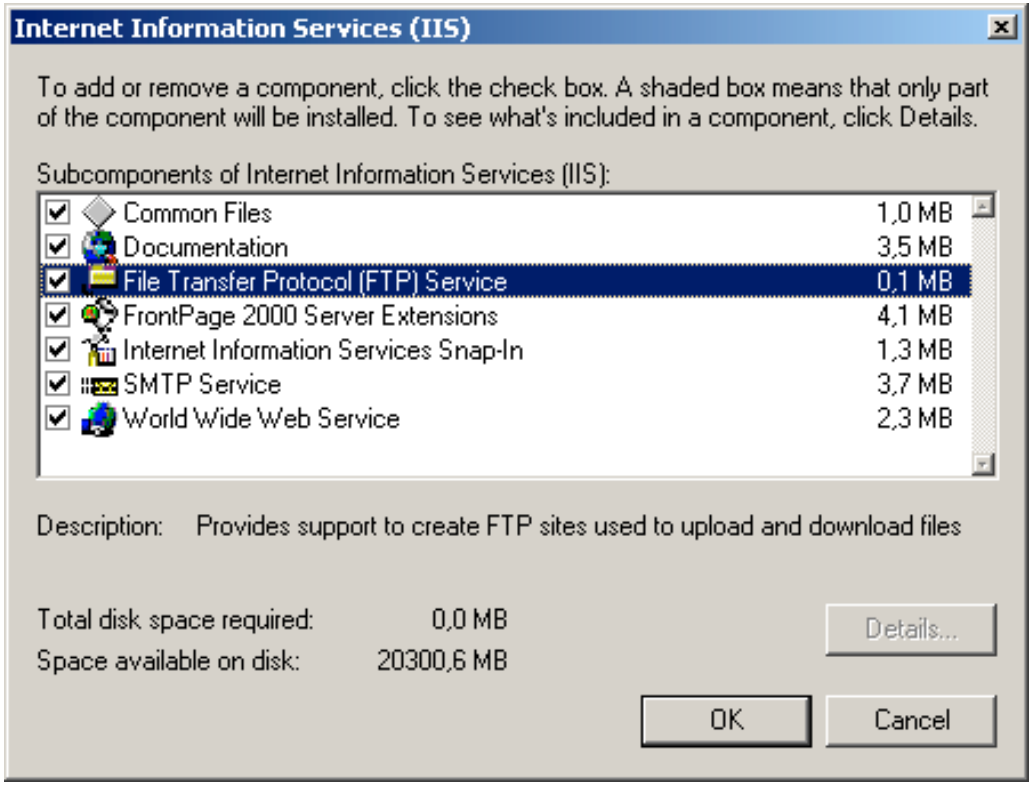
- Przykład - wyświetlanie zawartości tabeli
  - filePath = Server.MapPath("authors.mdb")
  - Set oConn = Server.CreateObject("ADODB.Connection")
  - oConn.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & filePath
  - Set oRs = oConn.Execute (" select * from authors where YearBorn =1967 ")
  - While Not oRs.EOF
  - Response.Write("<p>Inserted Author: " & oRs("Author") & "," & oRs("YearBorn"))
  - oRs.MoveNext
  - oRs.Close
  - Set oRs= Nothing

ASP - dostęp do bazy danych poprzez ODBC

- Przykład - wyświetlanie zawartości tabeli
  - Set oConn = Server.CreateObject("ADODB.Connection")
  - Set oConn.Open "DSN=Northwind;Database=Northwind;UID=sa;PWD=;"
  - Set oRs = oConn.Execute (" select * from authors where YearBorn =1967 ")
  - While Not oRs.EOF
  - Response.Write("<p>Inserted Author: " & oRs("Author") & "," & oRs("YearBorn"))
  - oRs.MoveNext
  - oRs.Close
  - Set oRs= Nothing

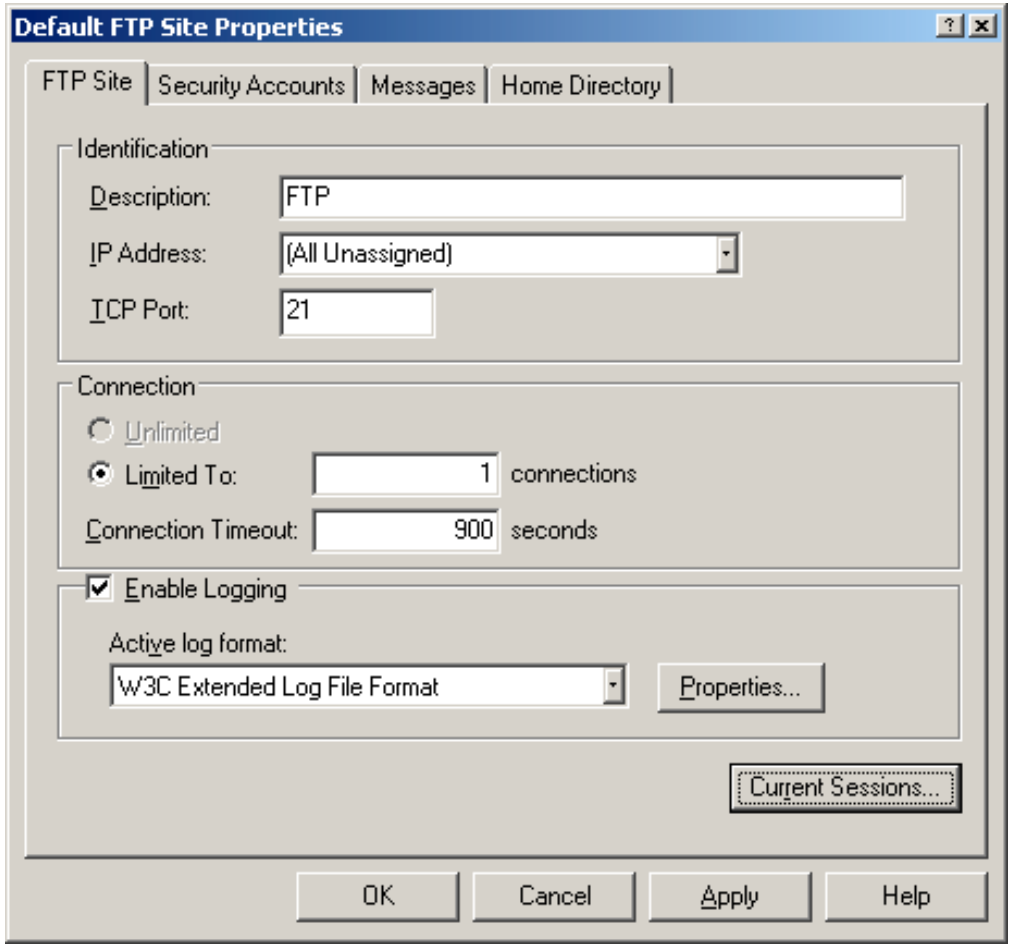
Instalacja serwera FTP

- Usługa dostępna w ramach pakietu IIS, jednak nie instalowana standardowo



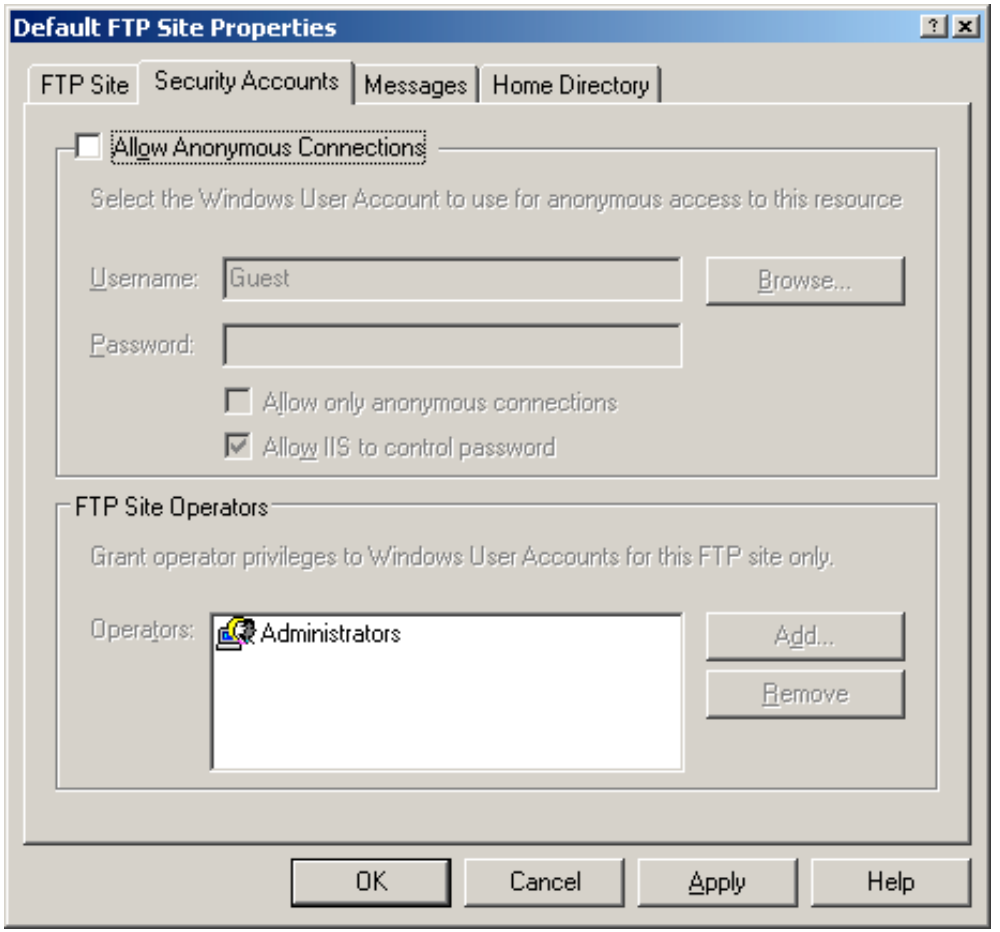
Konfiguracja usługi FTP (I)

- Port FTP, limity ilości połączeń i czasu bezczynności połączenia,
  audyt połączeń



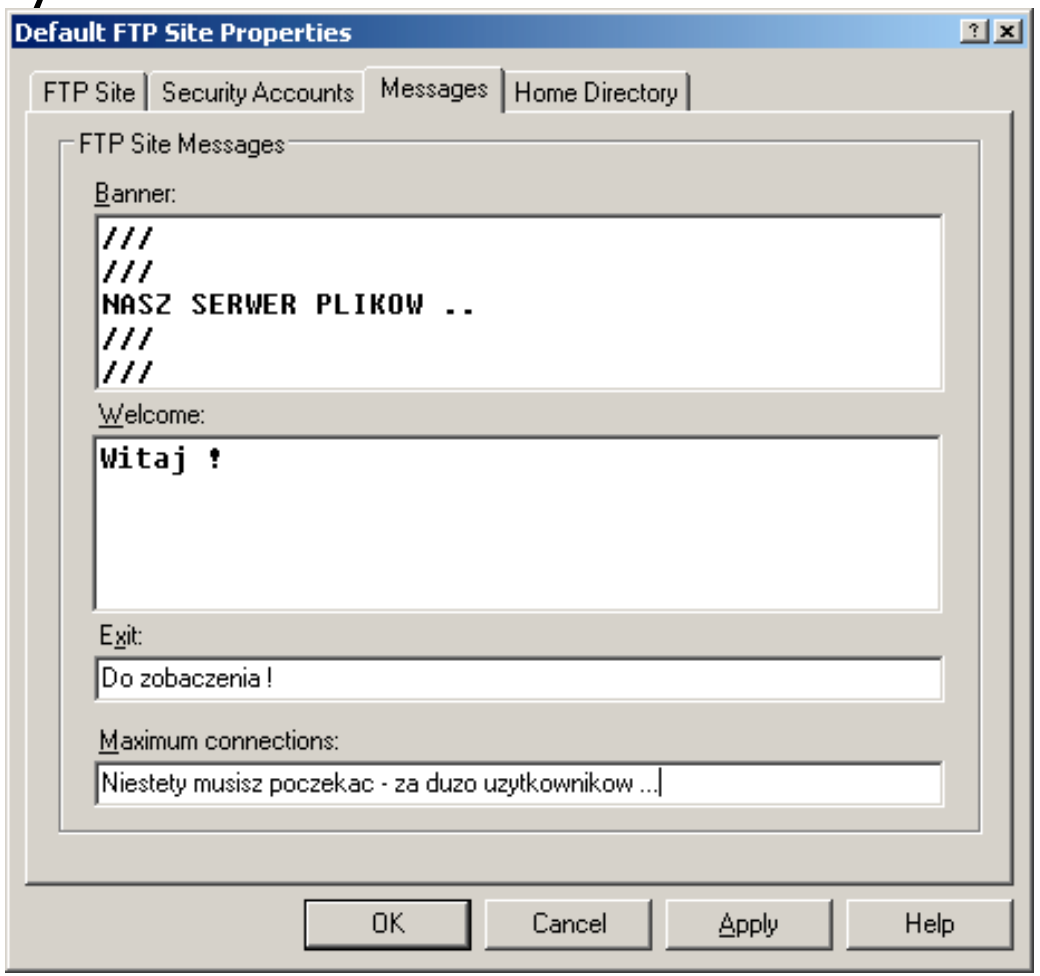
Konfiguracja usługi FTP (II)

- Autoryzacja użytkowników i operatorów - uwaga logowanie anonimowe również wymaga podania (jakiegokolwiek) użytkownika systemu.



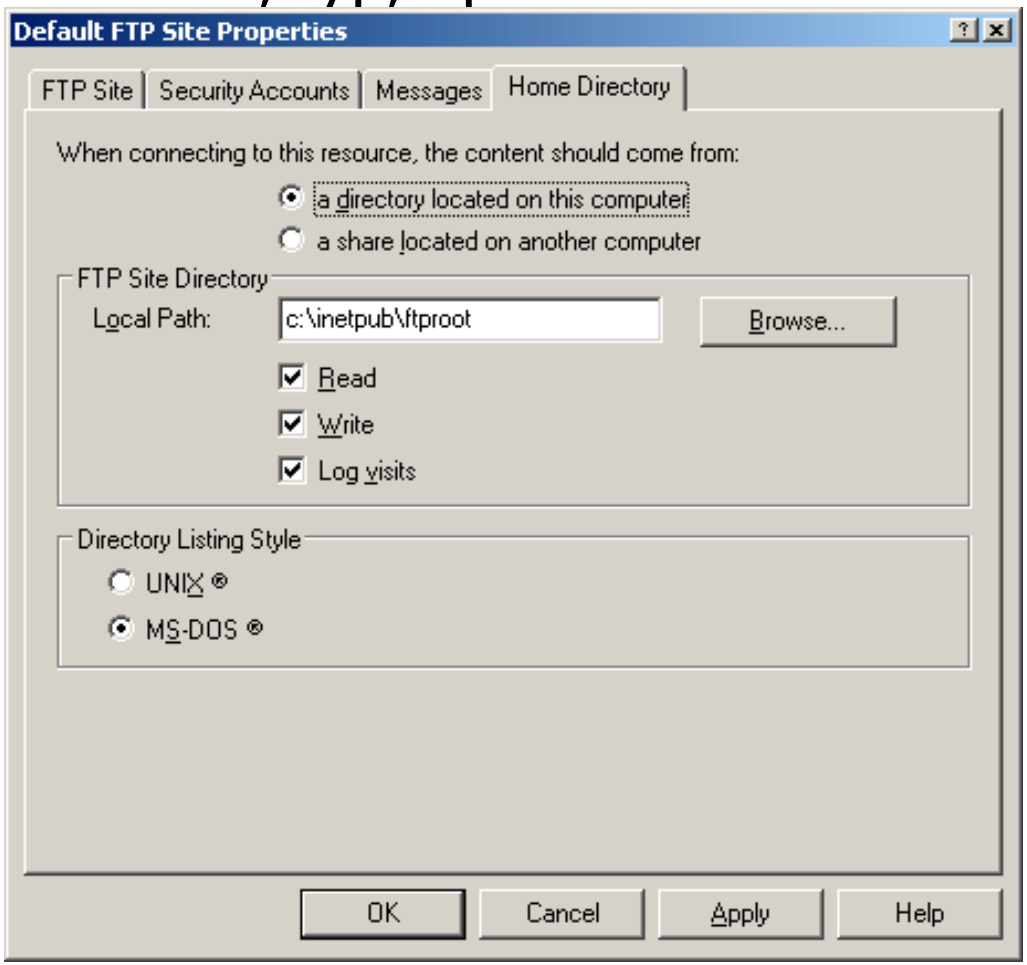
Konfiguracja usługi FTP (III)

- Komunikaty do użytkownika



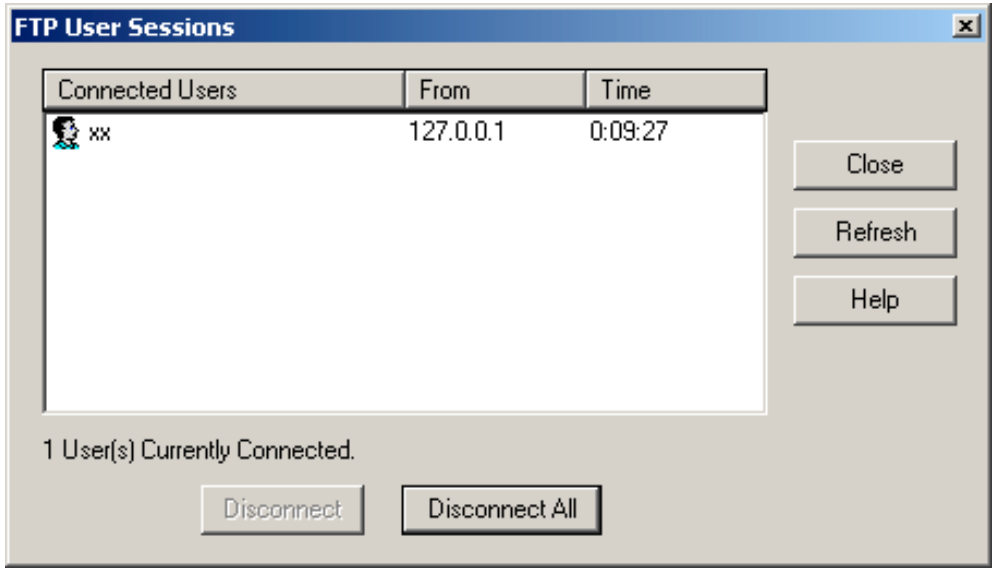
Konfiguracja usługi FTP (IV)

- Katalog domowy - ścieżka, typ, uprawnienia



FTP - monitorowanie użytkowników

- Możliwość ręcznego rozłączenia użytkownika



IIS 6.0 - (I)

- Wprowadzenie (poprzez WMI) możliwości administrowania z linii komend. Dostępne skrypty realizujące następujące czynności:
  - iisweb.vbs: tworzenie, usuwanie, uruchamianie, zatrzymywanie i wyświetlanie listy witryn sieci Web;
  - iisftp.vbs: tworzenie, usuwanie, uruchamianie, zatrzymywanie i wyświetlanie listy witryn FTP;
  - iisvdir.vbs: tworzenie i usuwanie katalogów wirtualnych lub wyświetlanie katalogów wirtualnych w danym katalogu głównym;

IIS 6.0 - (II)

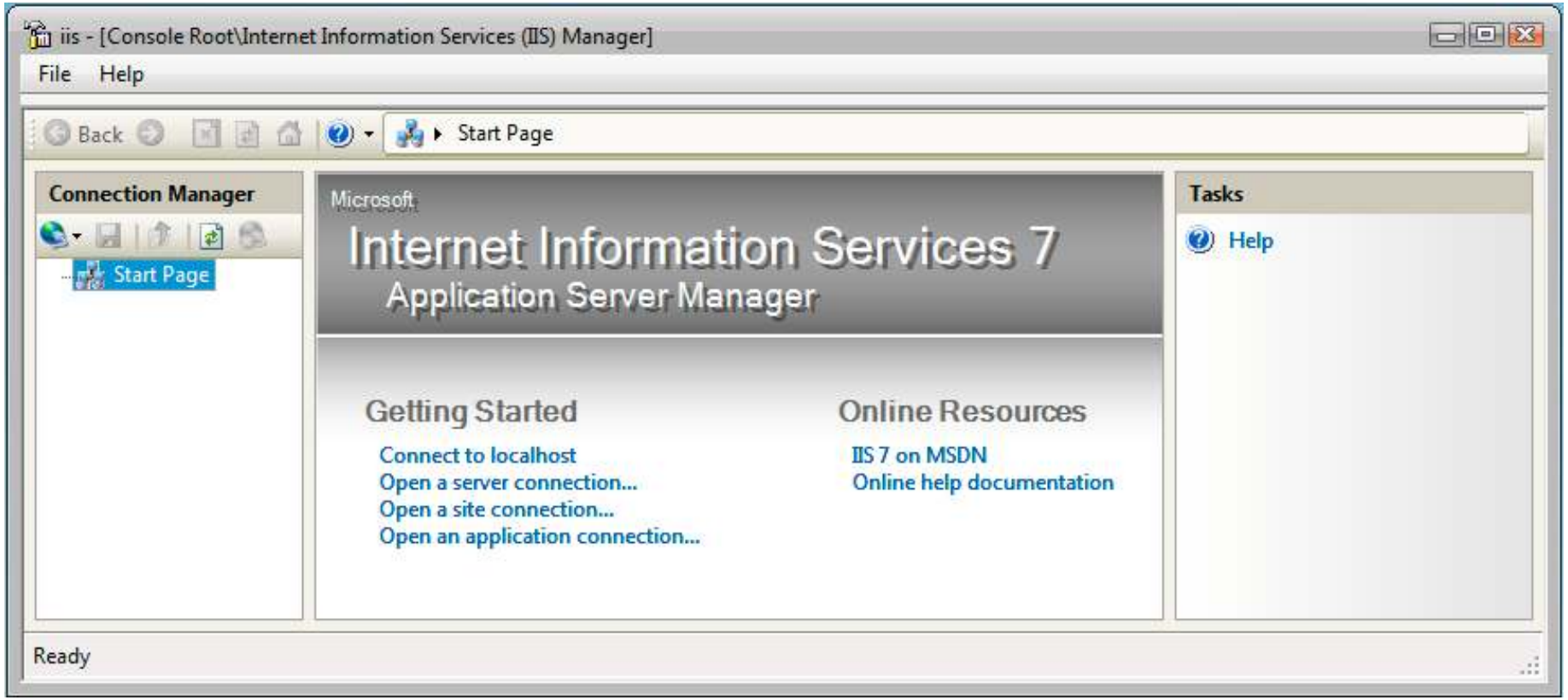
- Skrypty konfiguracyjne WMI (ciąg dalszy):
  - iisftpdr.vbs: tworzenie, usuwanie lub wyświetlanie katalogów wirtualnych w danym katalogu głównym;
  - iisconfg.vbs: eksport konfiguracji usług IIS do pliku XML i import konfiguracji z pliku XML;
  - iisback.vbs: tworzenie i odtwarzanie kopii zapasowej konfiguracji usług IIS;
  - iisapp.vbs: wyświetlanie listy identyfikatorów procesów i identyfikatorów pul aplikacji dla działających procesów roboczych (W3WP.EXE);
  - iisweb.vbs: konfigurowanie rozszerzeń usług sieci Web.

IIS 6.0 - (III)

- Nowa konsola administracyjna obsługiwana za pośrednictwem sieci Web - narzędzie Administracja zdalna (HTML)
- Edycja konfiguracji podczas działania serwera
- Lepsze zabezpieczenia:
  - Usługi IIS działają domyślnie jako konto o ograniczonych uprawnieniach
  - Wykrywanie i raportowanie konsekwencji dla IIS po zmianach zabezpieczeń maszyny (Group Policy)
  - Monitoring i usuwanie zablokowanych podprocesów.

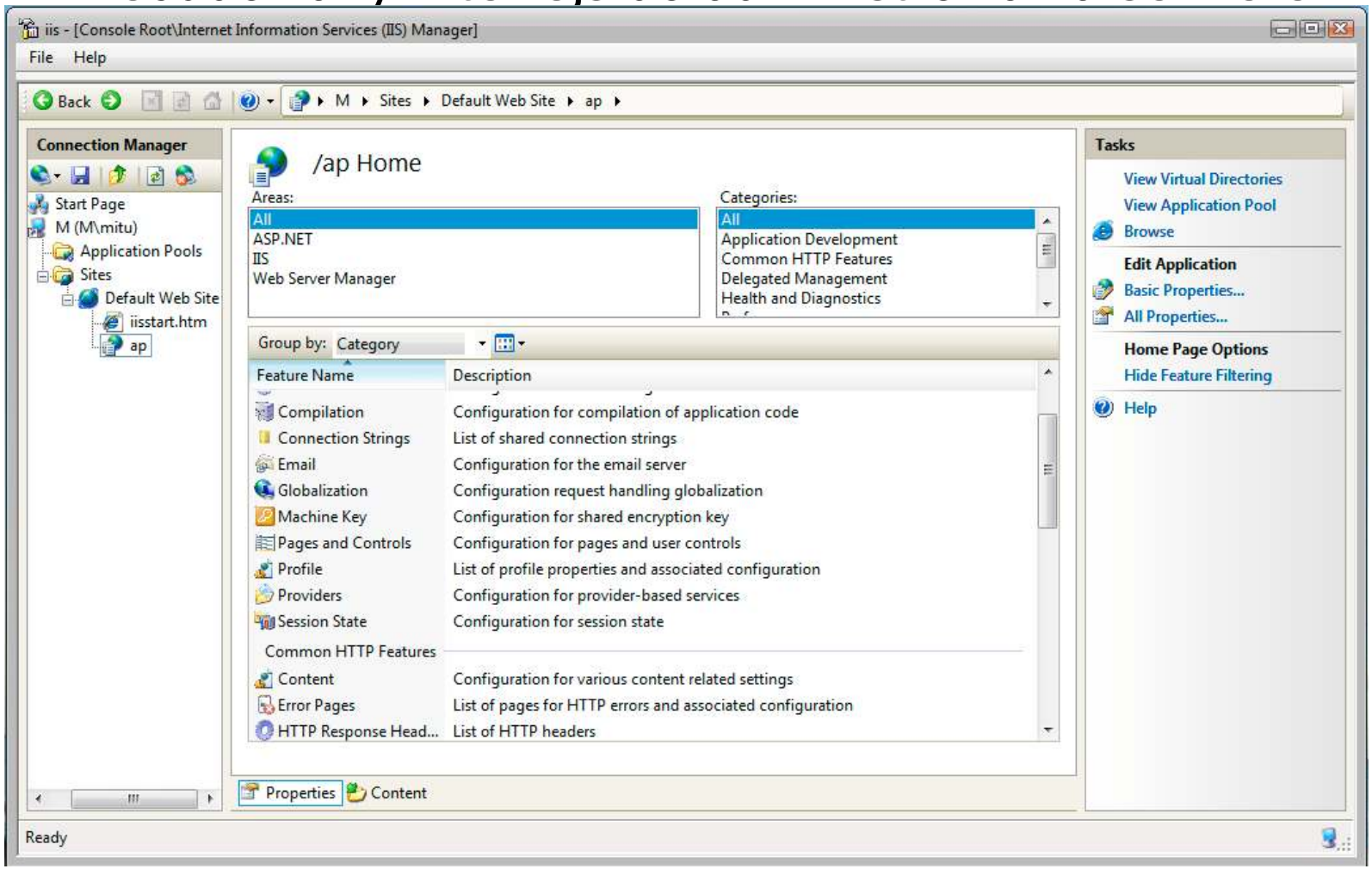
IIS 7.0 (I)

- Dostępny z wersjami Windows Vista Beta2 i Windows Serwer Longhorn



IIS 7.0 (II)

- Przebudowany interfejs do administrowania serwerem



Administrowanie Microsoft SQL Server (I)

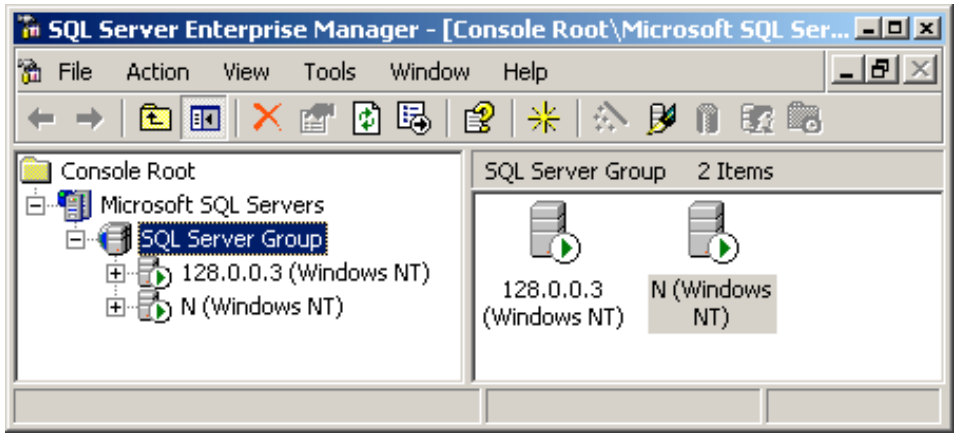
- Engine zainstalowany jako usługa (więc brak bezpośredniego interfejsu użytkownika)
- Jedyny profesjonalny serwer baz danych Microsoft
- Silna integracja z systemami operacyjnymi
- Integracja z platformą .NET i wsparcie dla serwerów aplikacji
- Dostęp poprzez TCP/IP, named pipes, IPX/SPX, na protokołach warstw wyższych Windows

Administrowanie Microsoft SQL Server (II)

- Popularne wersje profesjonalne: 6.0, 7.0, 2000, 2003
- Wersje darmowe: MSDE (Microsoft Data Engine), 2005 Express
- Szereg narzędzi towarzyszących do konfiguracji (*SQL Network*), administrowania (*Enterprise Manager*), zarządzania danymi (*SQL Query Analyser*), śledzenia (*SQL Server Profiler*), optymalizacji (*SQL Performance Monitor*) itp.
- W wersjach darmowych narzędzia wymagają dodatkowego instalowania.

SQL Server
- Enterprise Manager

- Zarządzanie wieloma serwerami i grupami serwerów jednocześnie
- Automatyczne logowanie na serwerze -na bazie ustawień definiowanych indywidualnie dla każdego serwera

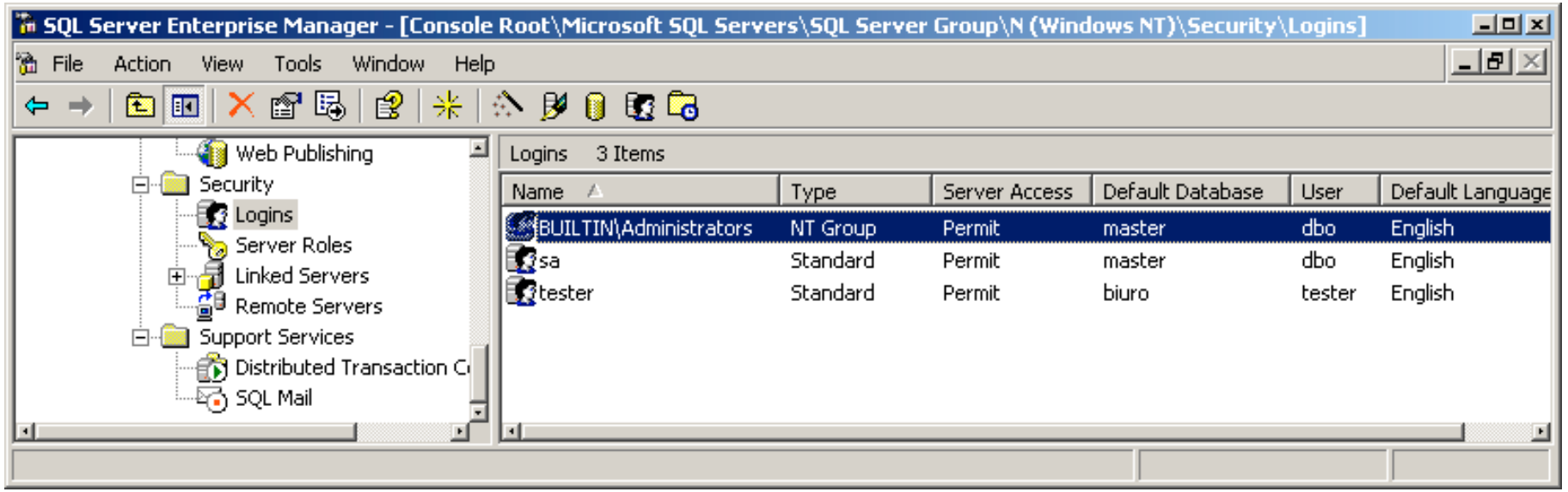


SQL Server
- zarządzanie uprawnieniami

- Użytkownicy - zależnie od opcji mogą dziedziczyć tożsamość z domeny (NT, Active Directory) lub być autoryzowani na podstawie bazy SQL Server.
- Istnieją trzy warstwy uprawnień użytkownika:
  - Login name
  - Baza danych
  - Operacja na obiekcie
- Użytkownik predefiniowany: sa (System Administrator)

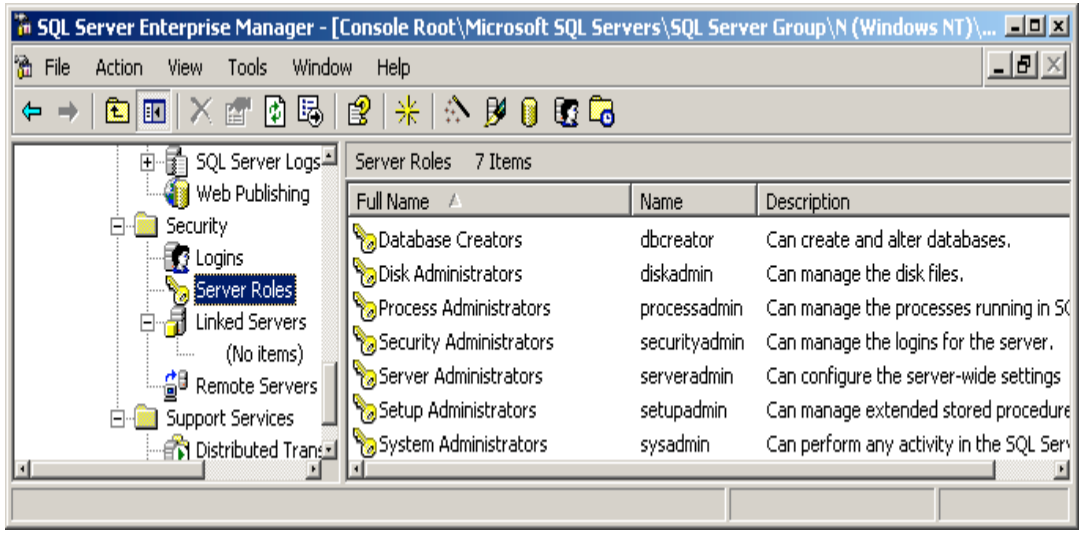
SQL Server
- login names

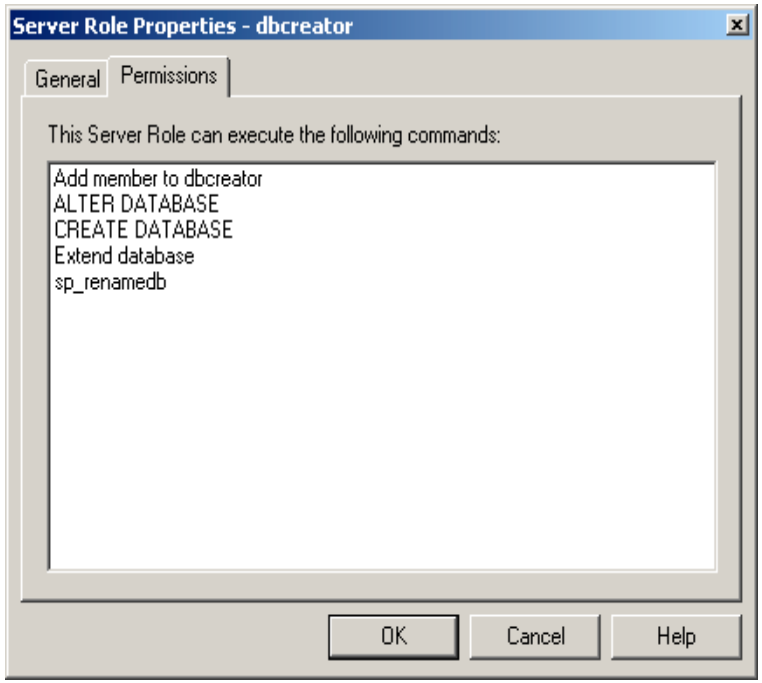
- Definiowane wraz z hasłami dostępu, na podstawie roli w serwerze (*Server roles*).
- Podział na nazwy wbudowane (domenowe) i definiowane lokalnie



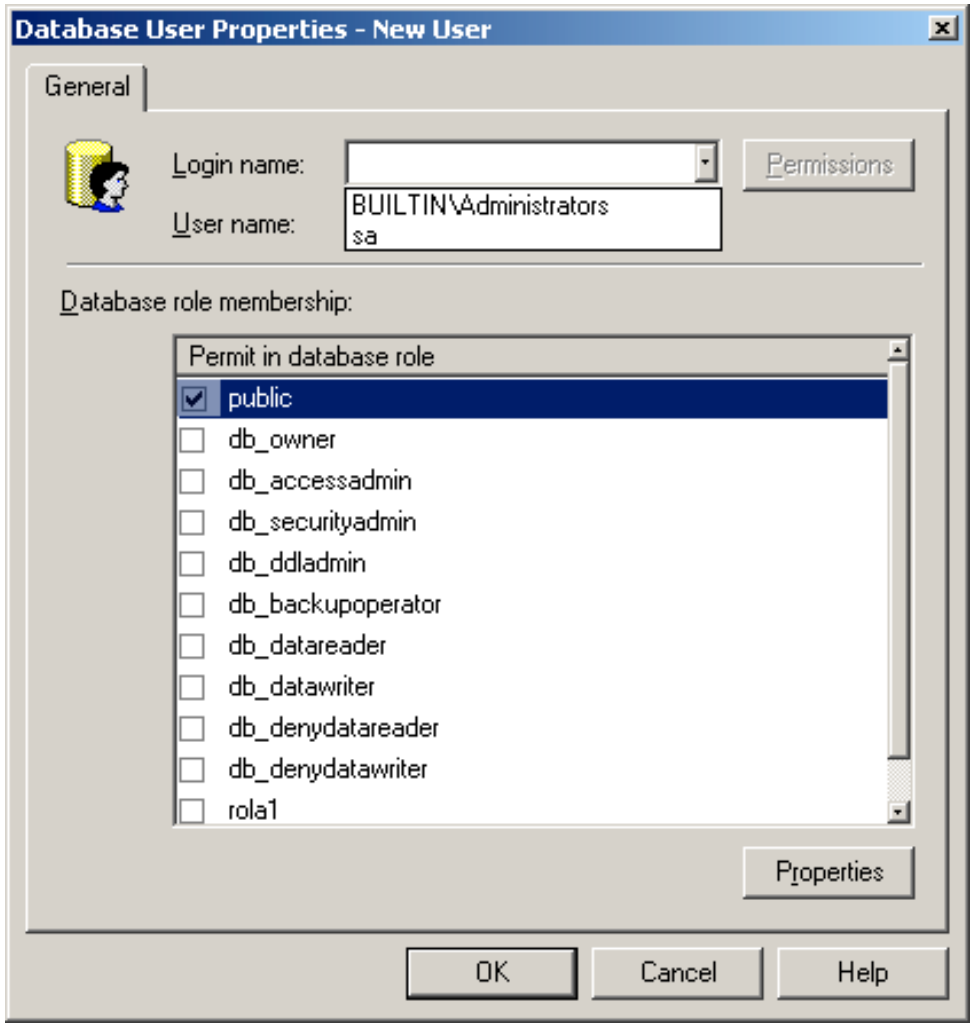
SQL Server
- server roles

- Role definiowane są przy użyciu łańcuchowych nazw czynności,jakie użytkownicy podejmują.

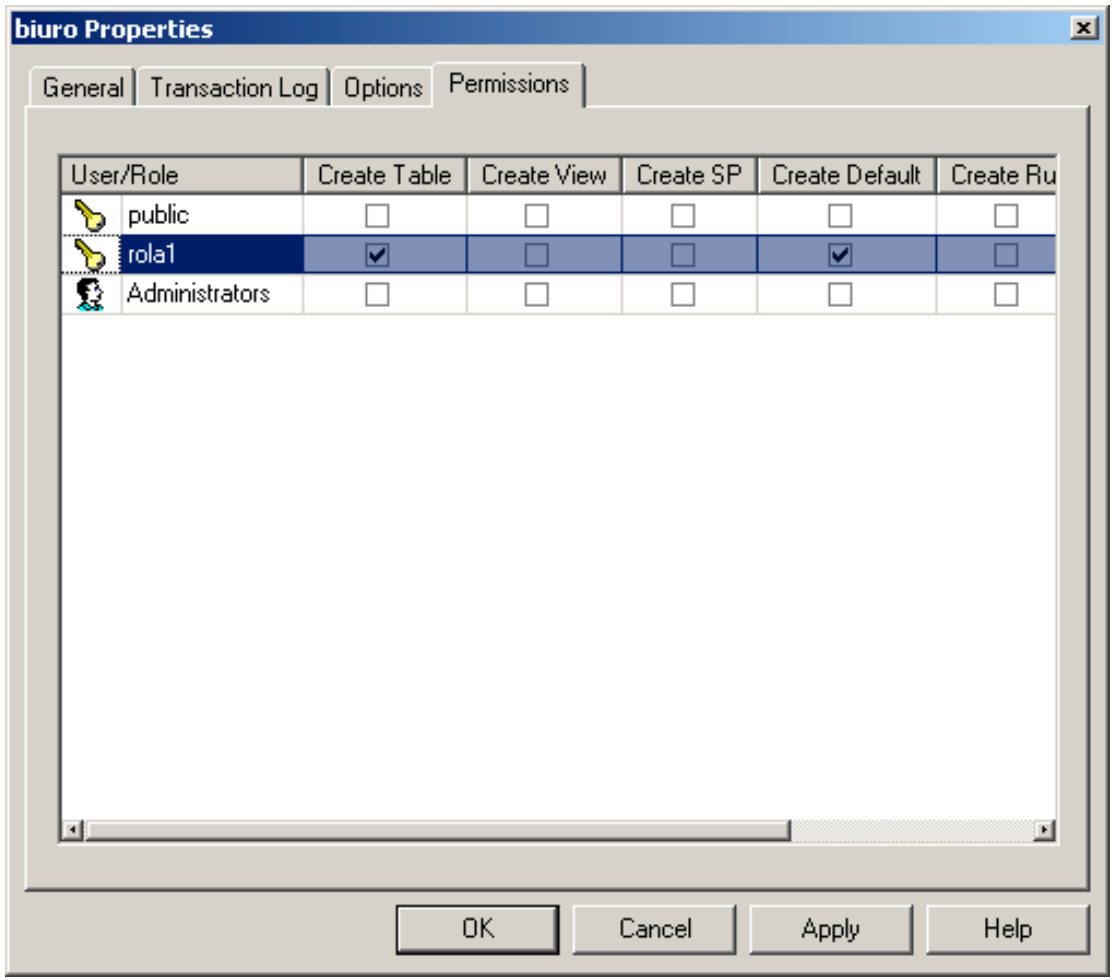




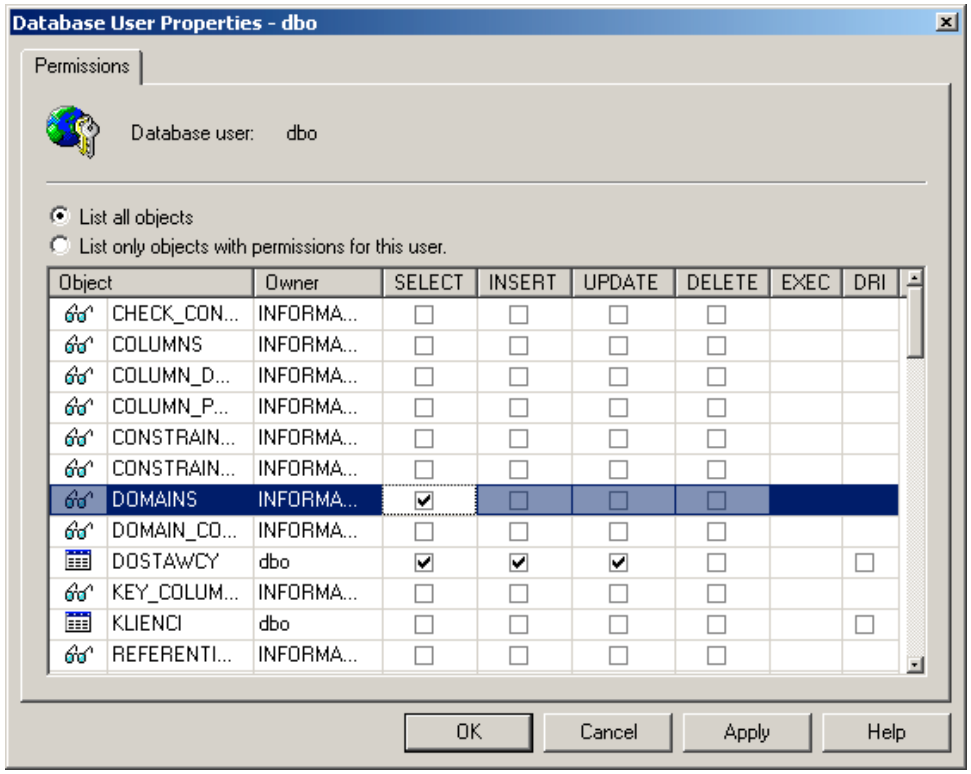
SQL Server - przydział *login name* do bazy danych

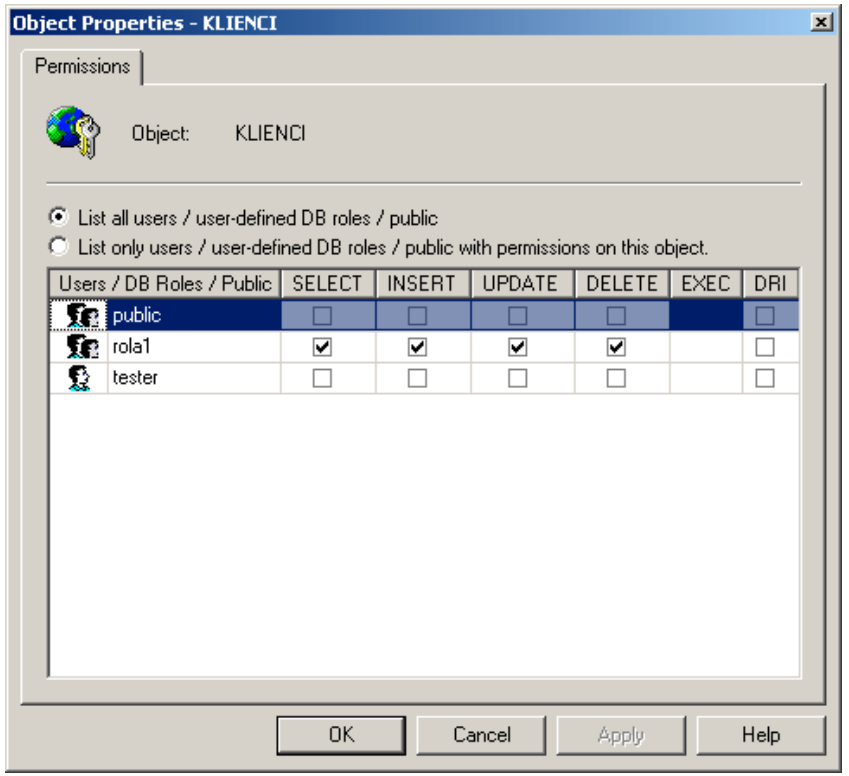


SQL Server - uprawnienia *user name* dla bazy danych

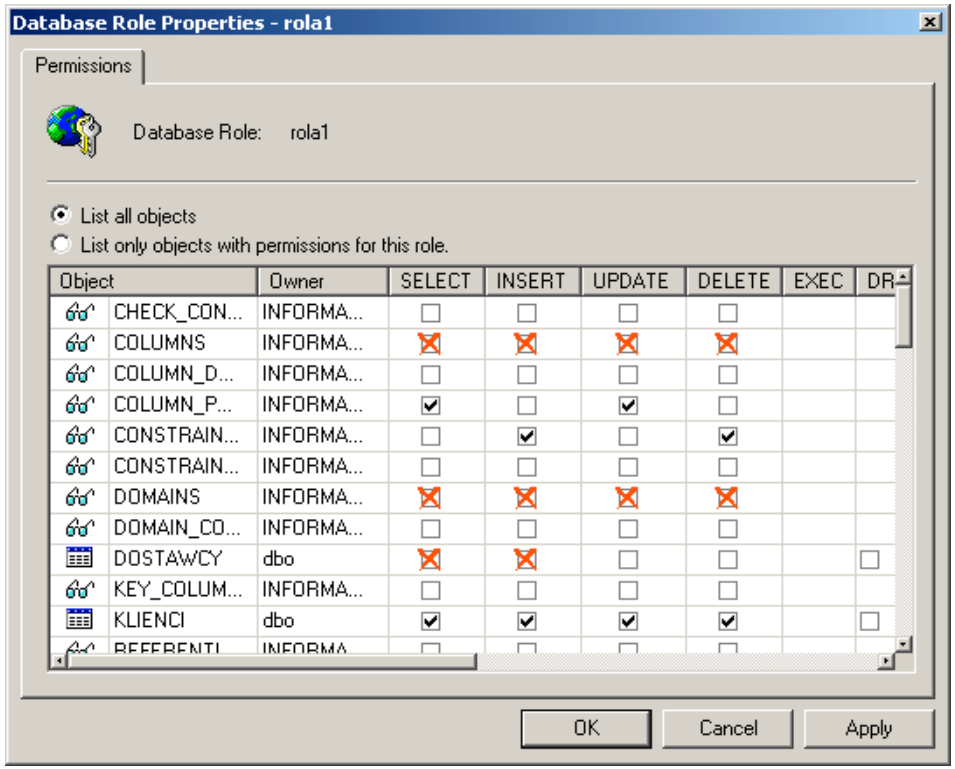


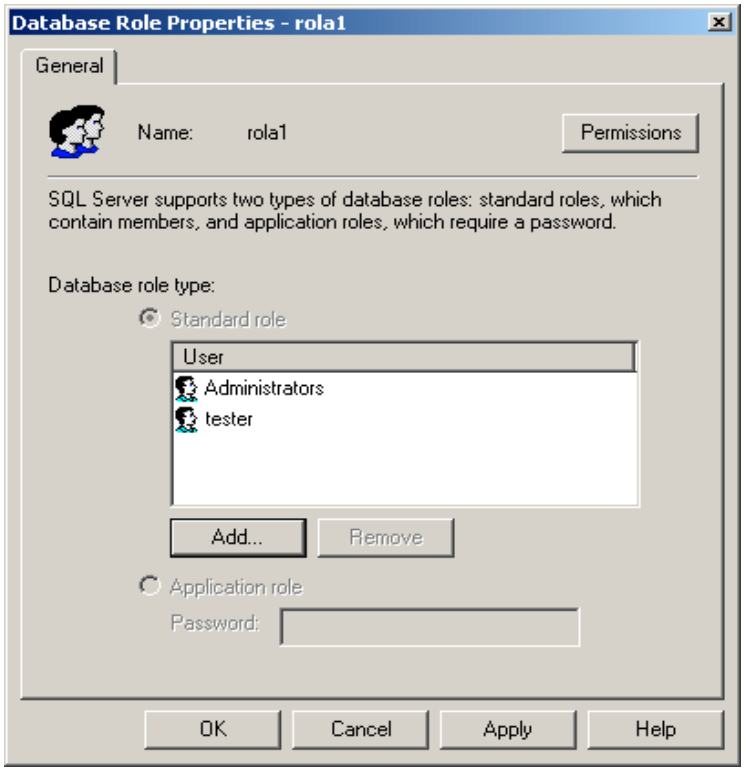
SQL Server - uprawnienia użytkowników bazy danych





SQL Server
- role użytkowników

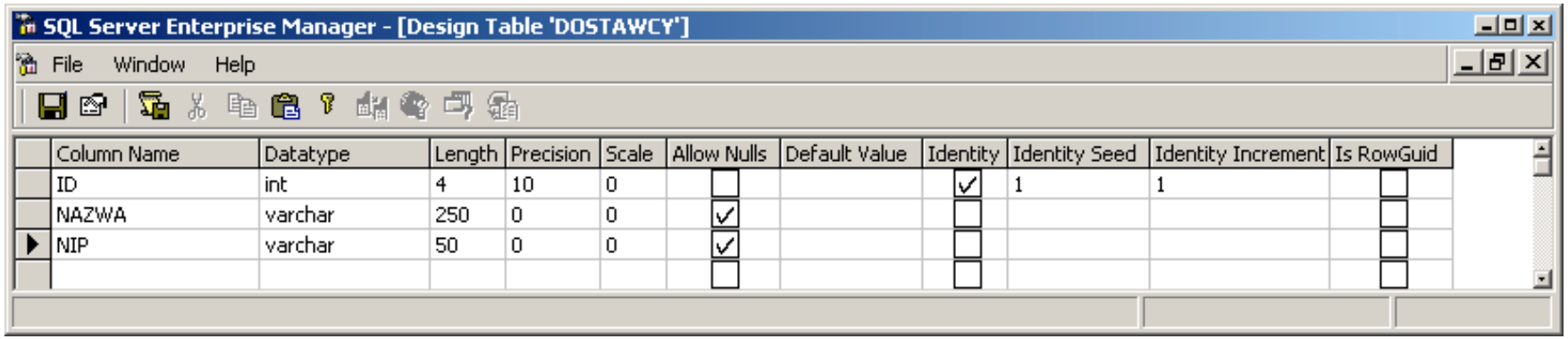


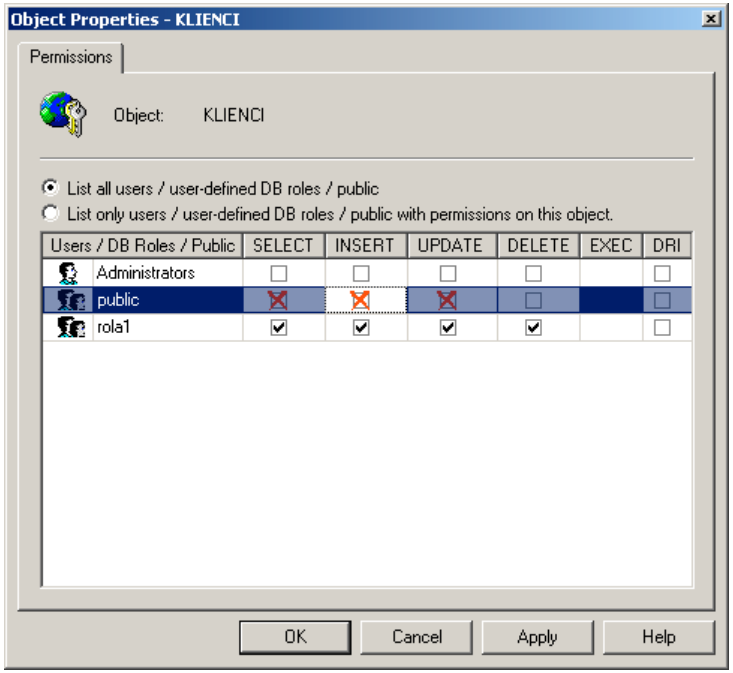


SQL Server
- stałe elementy bazy danych

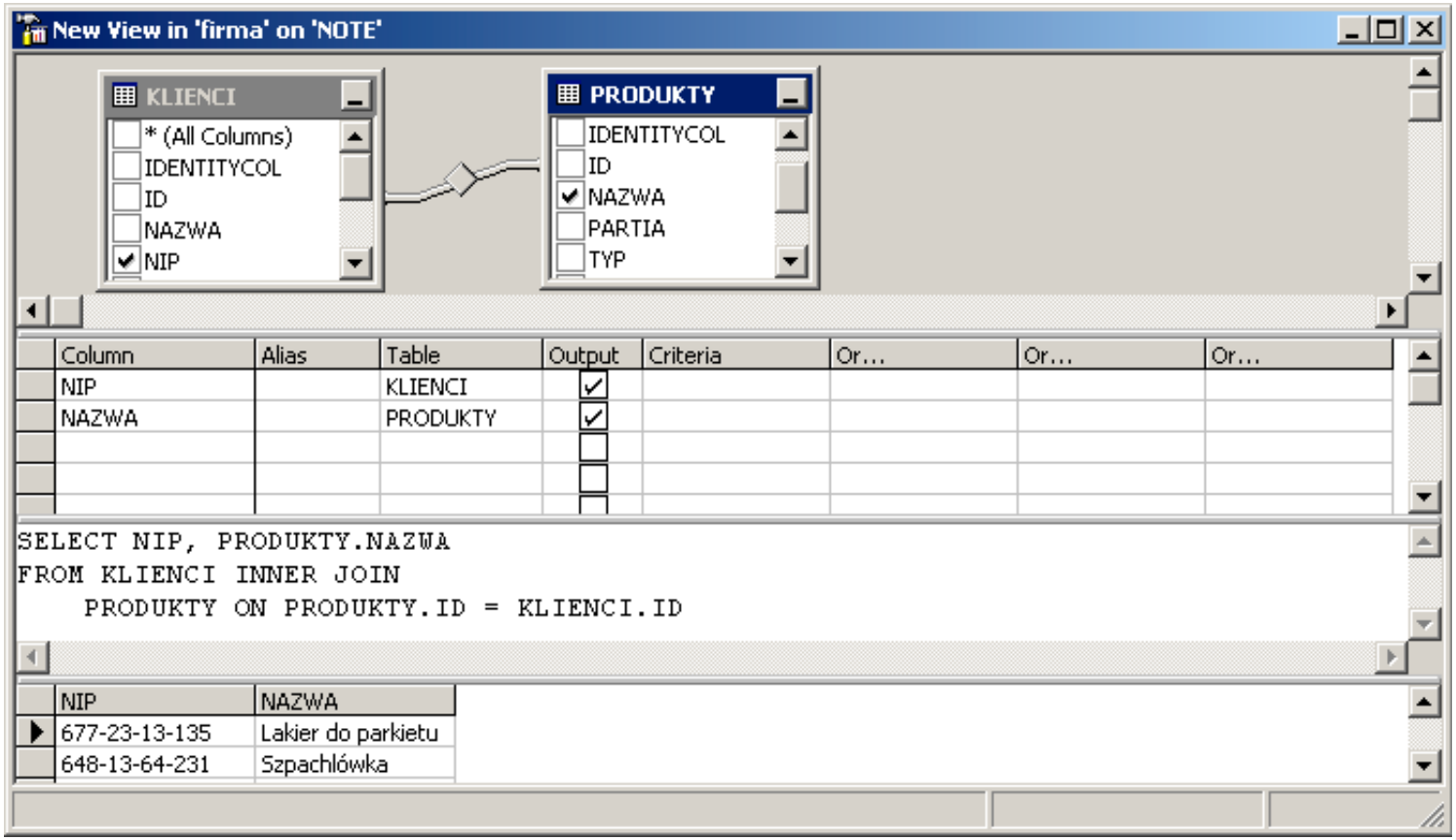
- W bazie danych mogą występować:
  - Diagramy (Diagrams)
  - Tabele (Tables)
  - Widoki (Views)
  - Procedury utrzymywane/osadzone (Stored Procedures)
  - Użytkownicy przydzieleni (Users)
  - Role dla użytkowników (Roles)
  - Reguły zawartości (Rules)
  - Wartości domyślne (Defaults)
  - Niestandardowe typy danych (User Def. Data Types)

SQL Server
- definiowanie tabeli

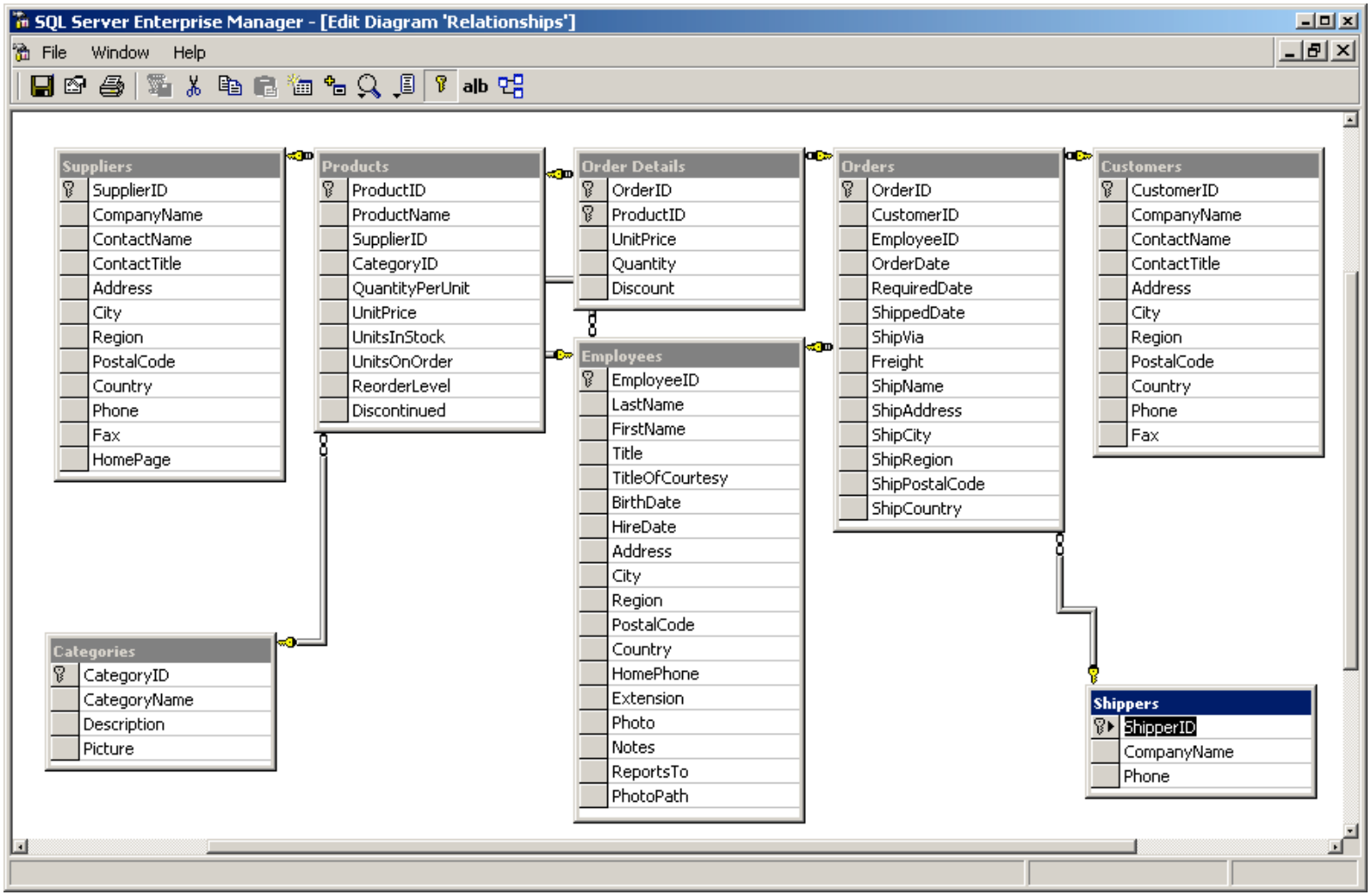




SQL Server
- definiowanie widoku

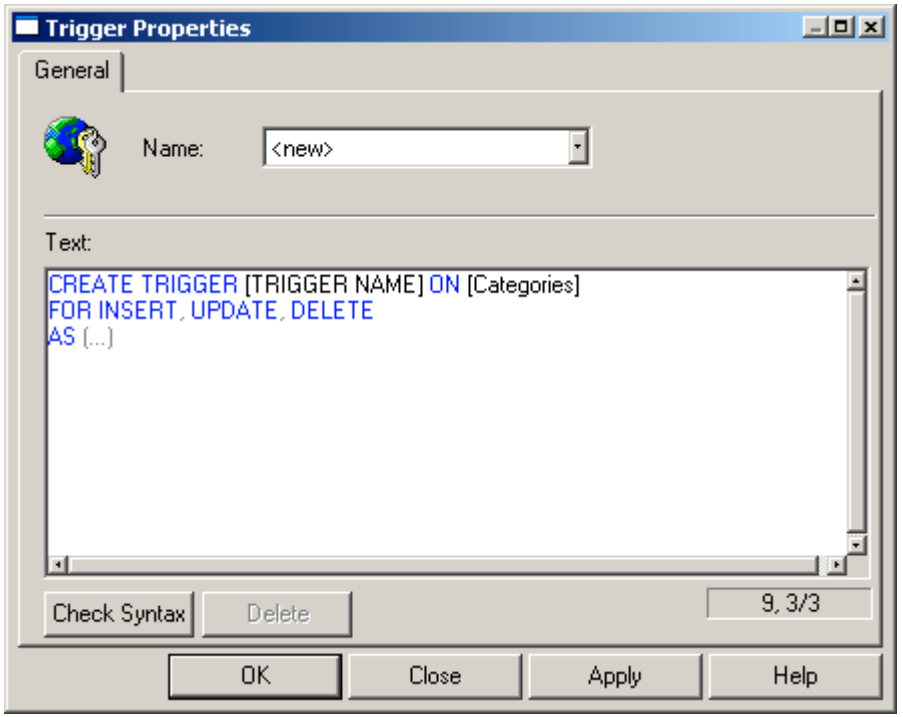


SQL Server
- diagramy tabel



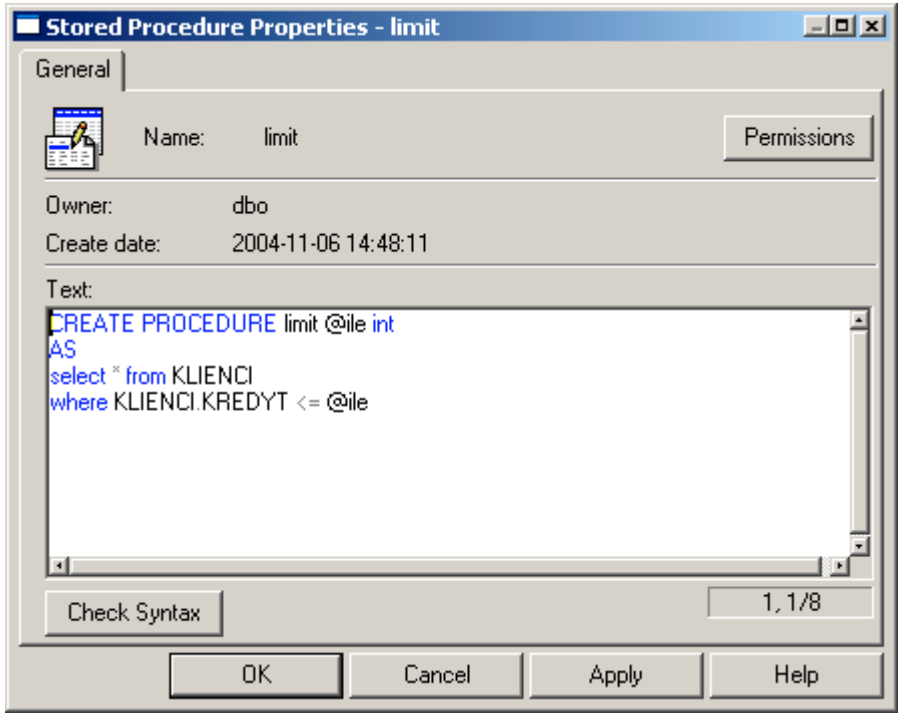
SQL Server - triggery

- Krótkie procedury „wyzwalane” w momencie gdy wykonywana jest operacja na bazie danych
- Możliwość monitorowania operacji SQL
- Przykład triggera:



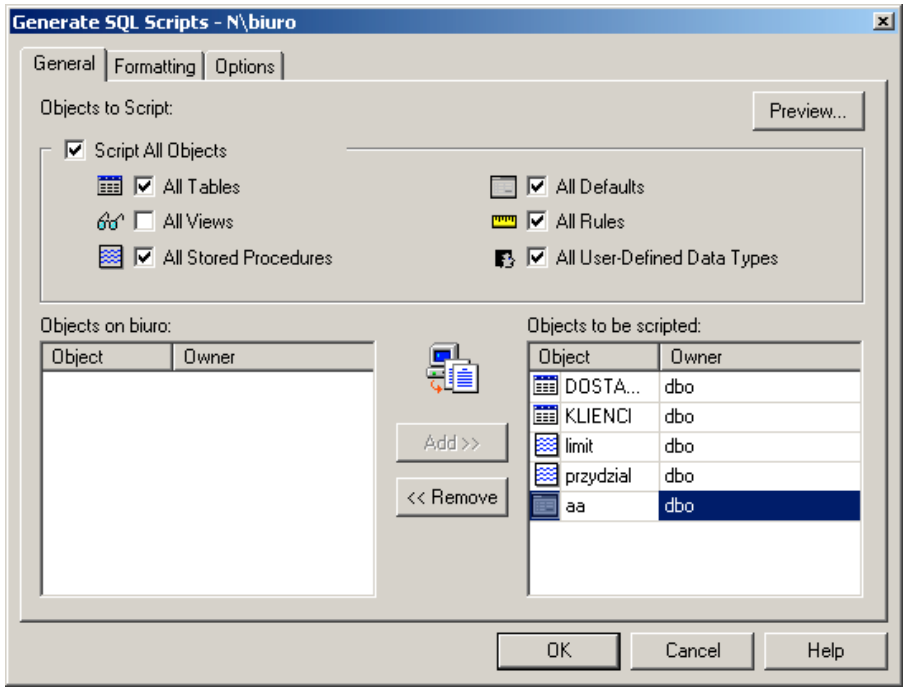
SQL Server
- procedury osadzone

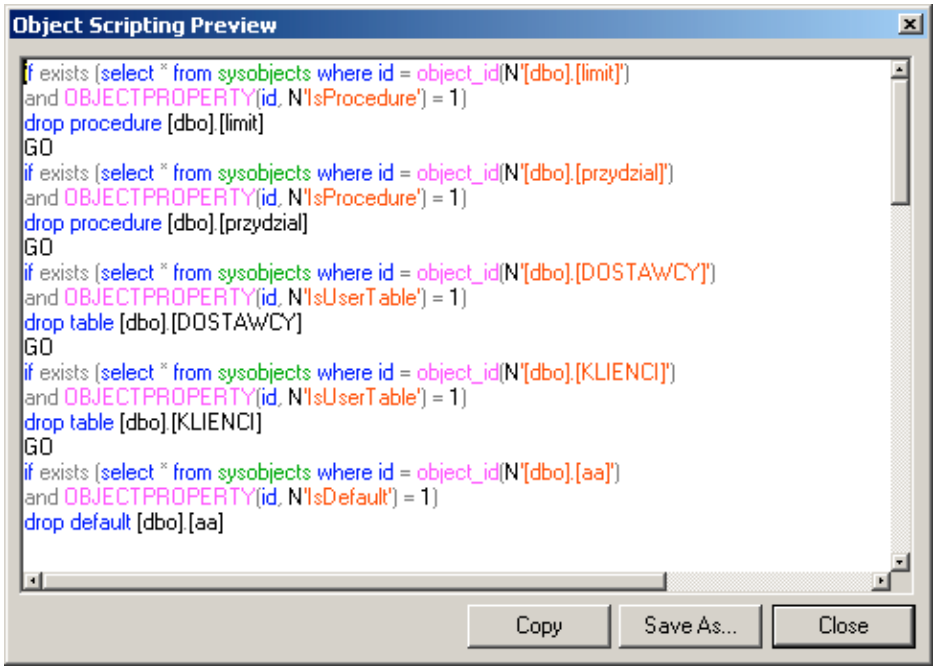
- Zdalne wywołanie procedur wraz z przekazaniem do nich parametrów i odebraniem wartości wynikowych



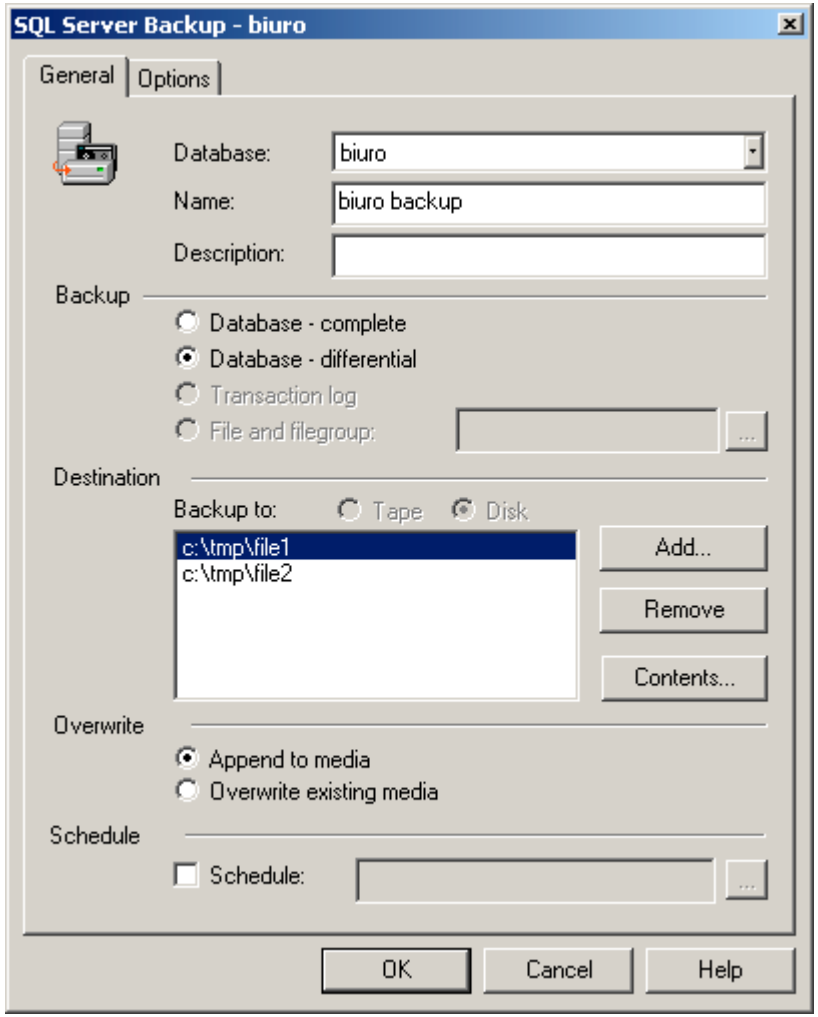
SQL Server
- skrypty SQL

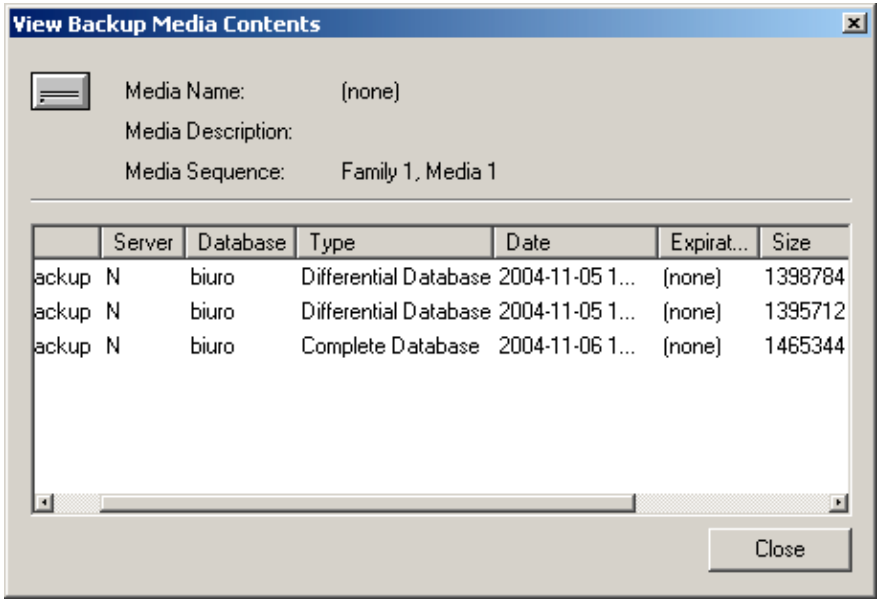
- Możliwość zachowania/przenoszenia struktury bazy danych



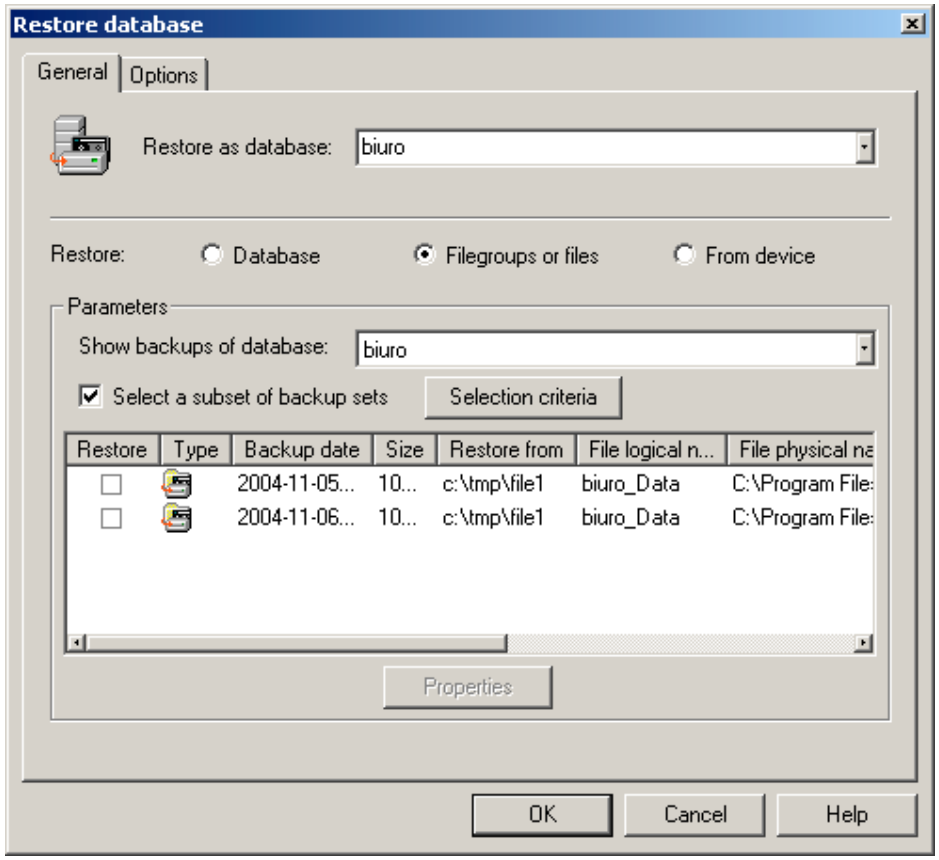


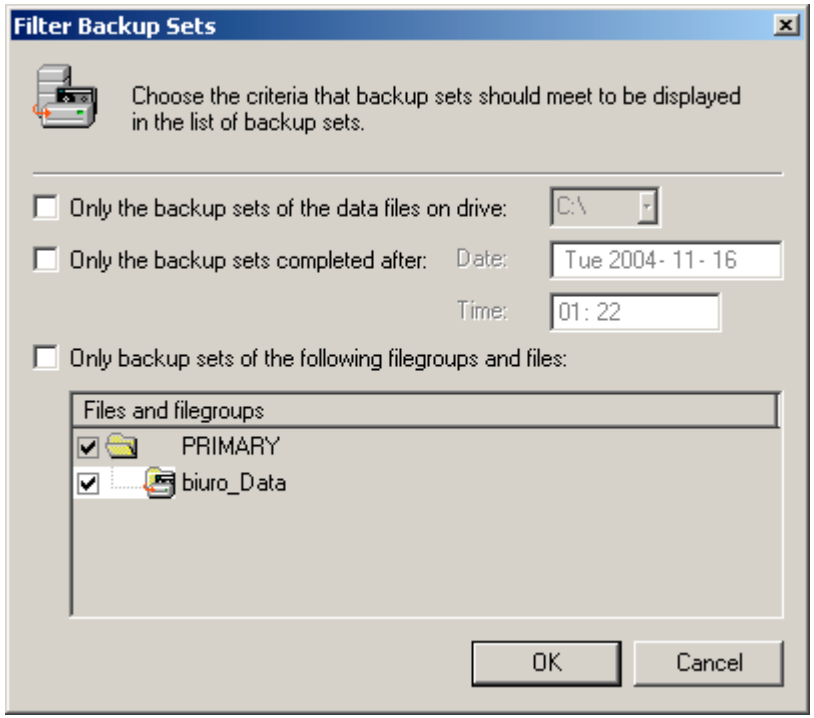
SQL Server
- backup bazy danych



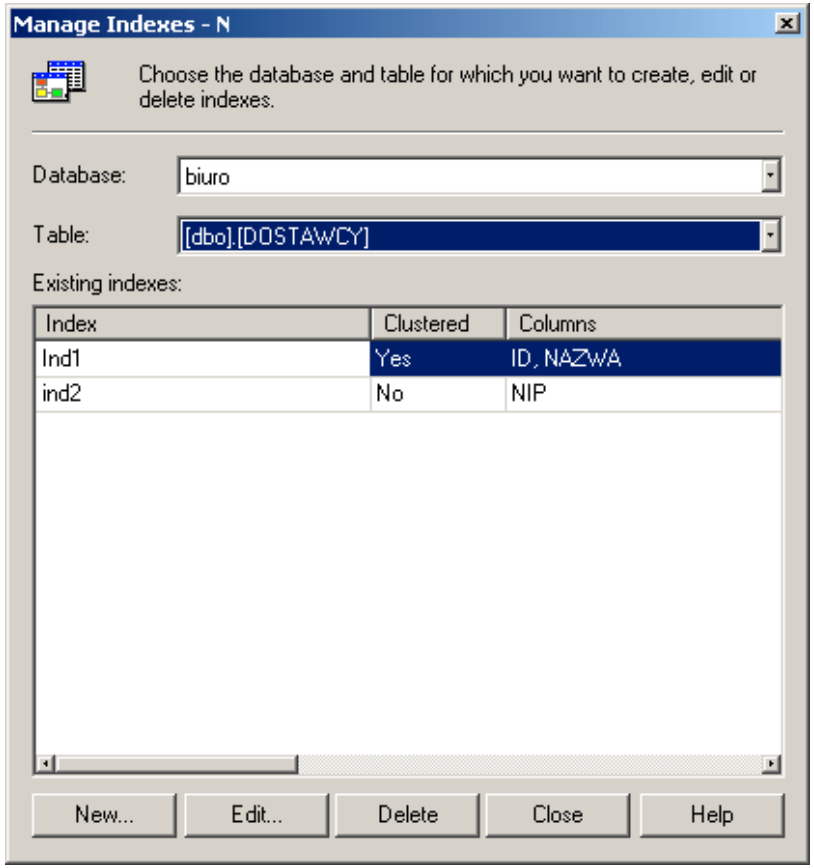


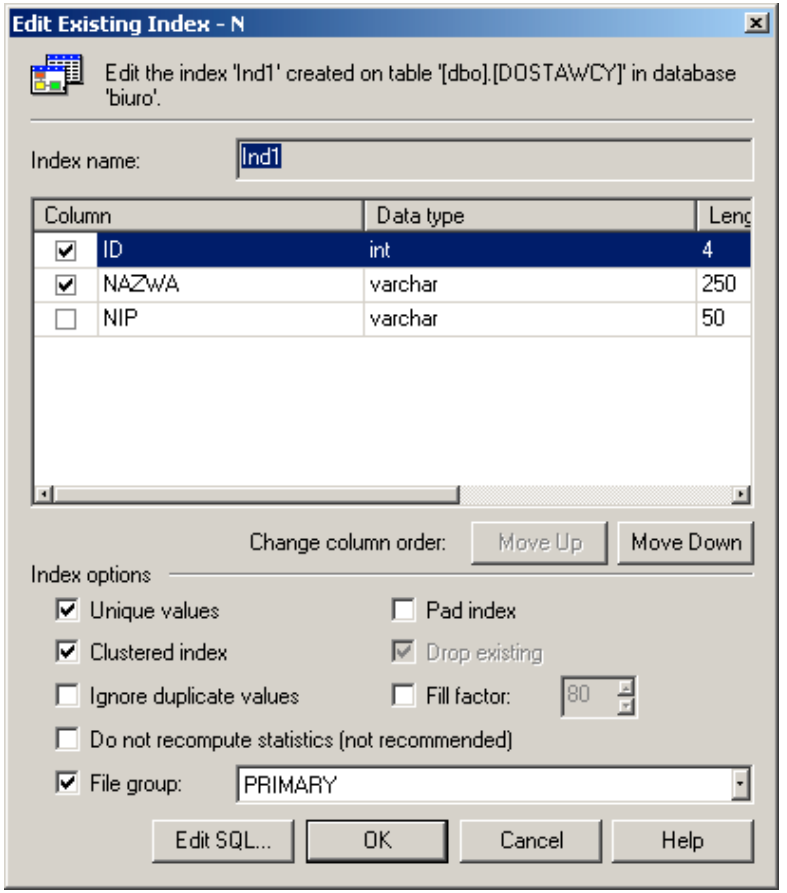
SQL Server
- odzyskiwanie danych





SQL Server
- indeksy w bazie danych



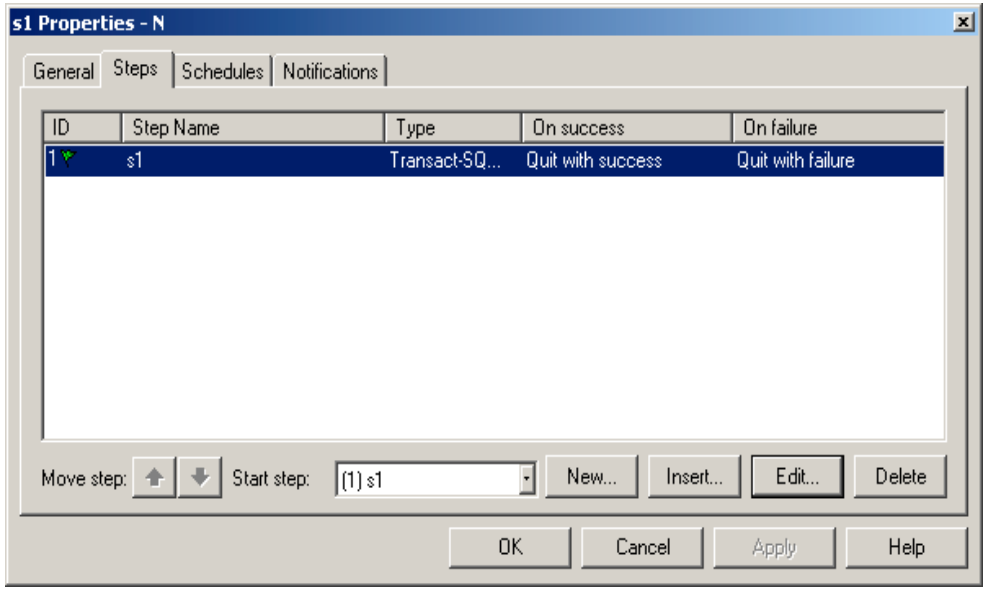


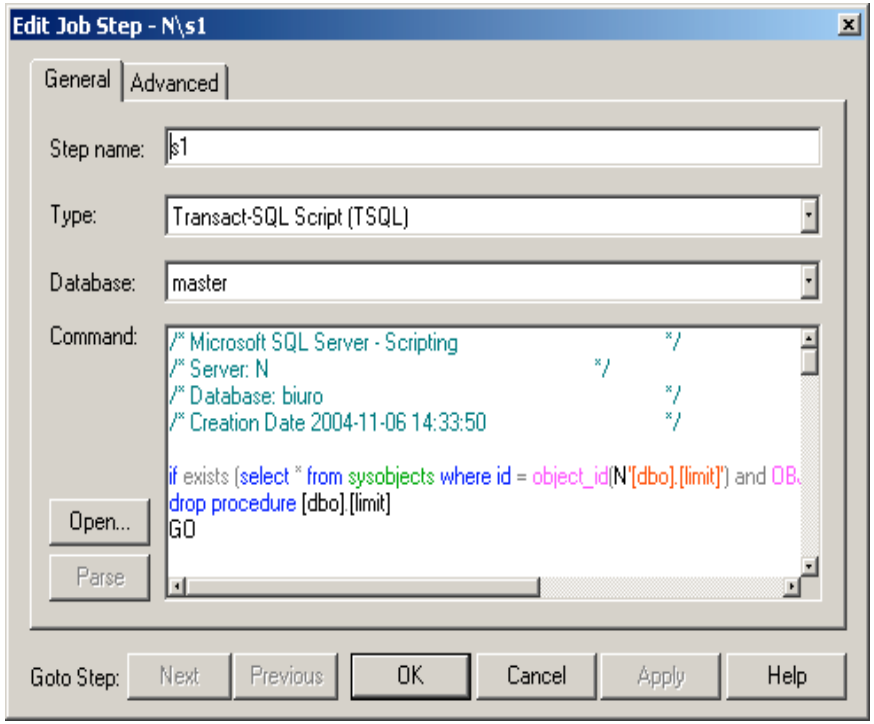
SQL Server - zadania zaplanowane (I)

- SQL Server Agent - usługa systemowa zapewniająca utrzymanie harmonogramu zadań serwera SQL wykonywanych automatycznie
- Operatorzy zadań
- Zadania wykonywane w ustalonych chwilach czasowych
- Wiele kroków w zadaniu. Obsługa statusu wykonania kroku zadania
- Powiadomienia wysyłane do operatorów zadań

SQL Server - zadania zaplanowane (II)

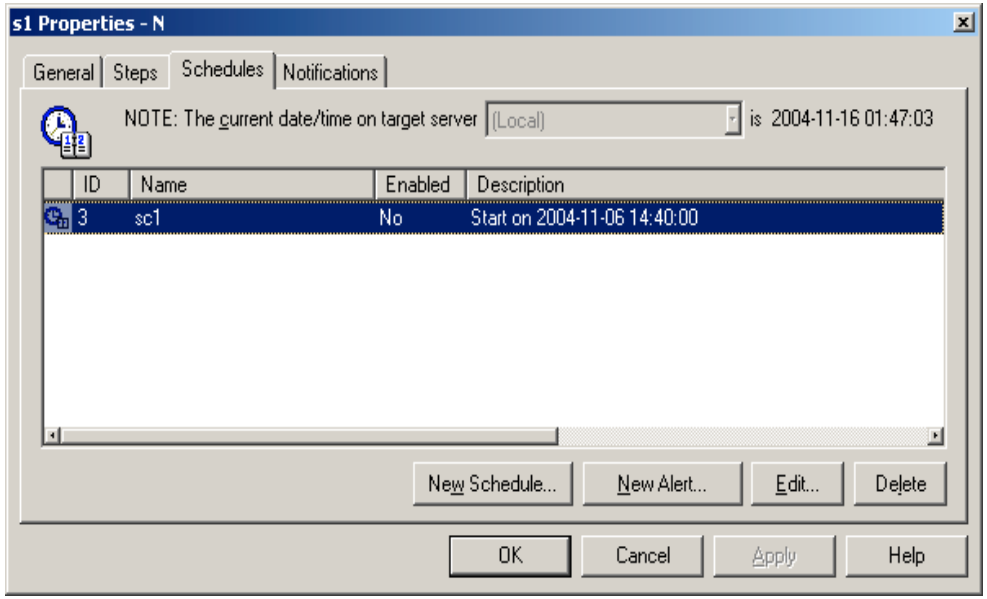
- Krok zadania:

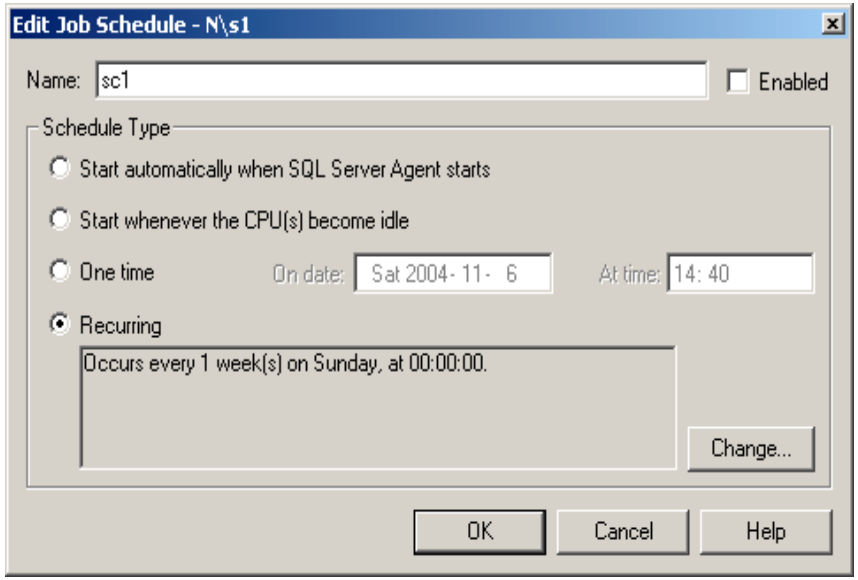




SQL Server - zadania zaplanowane (III)

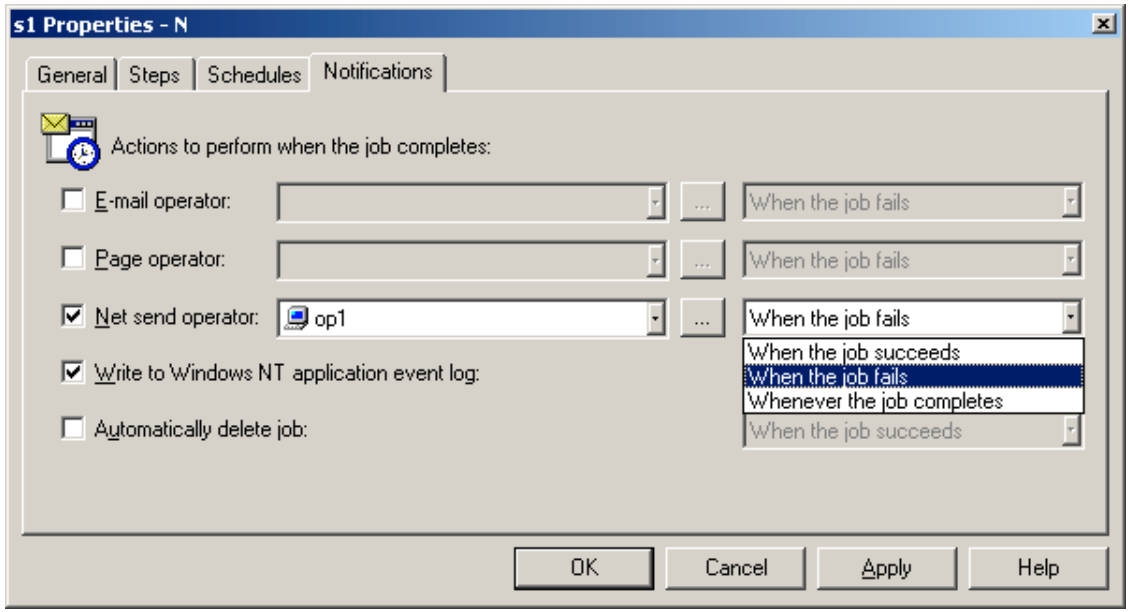
- Czas wykonania zadania:





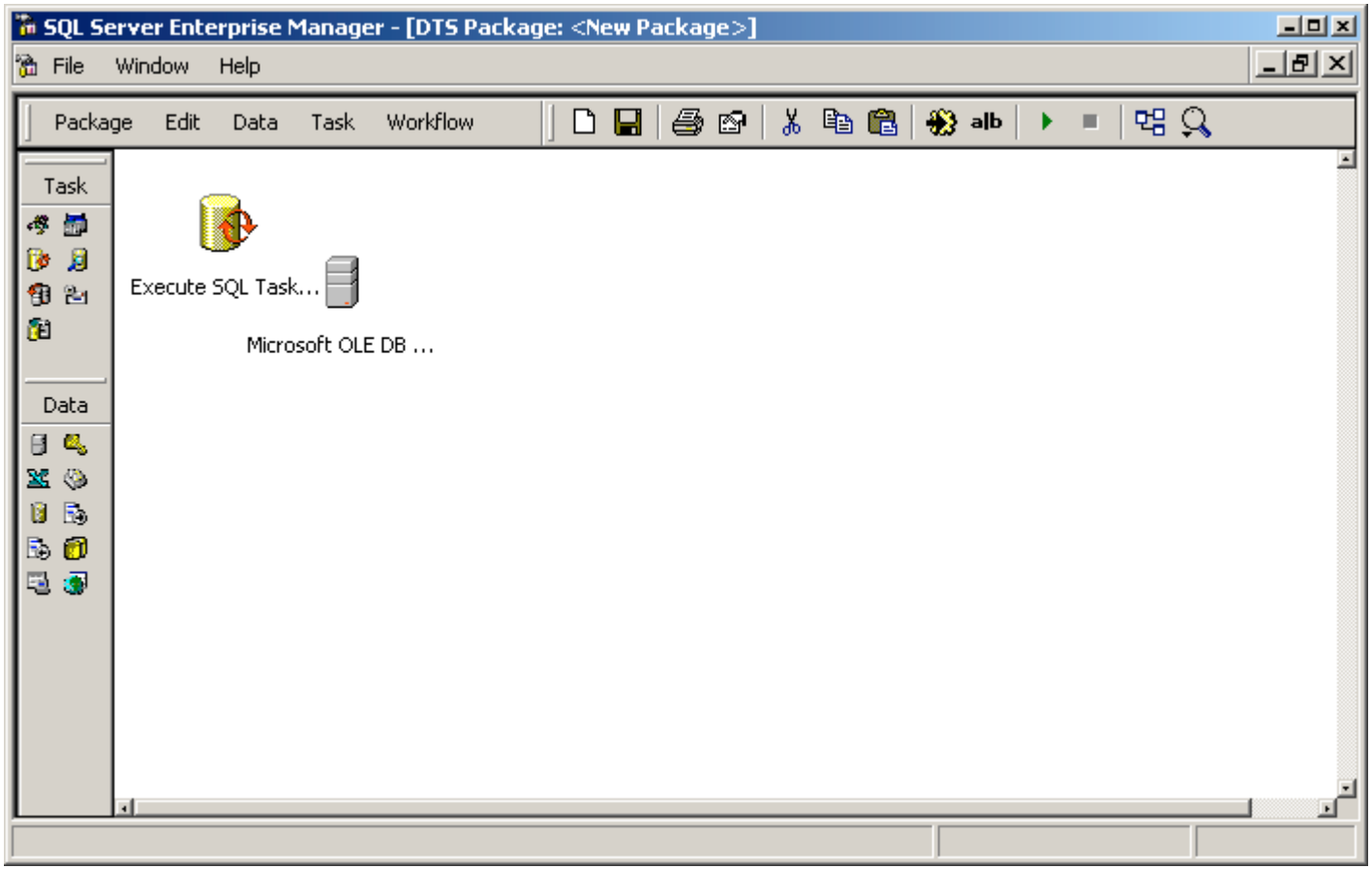
SQL Server - zadania zaplanowane (IV)

- Powiadomienia:



SQL Server - DTS (I)

- DTS - *Data Transformation Services*

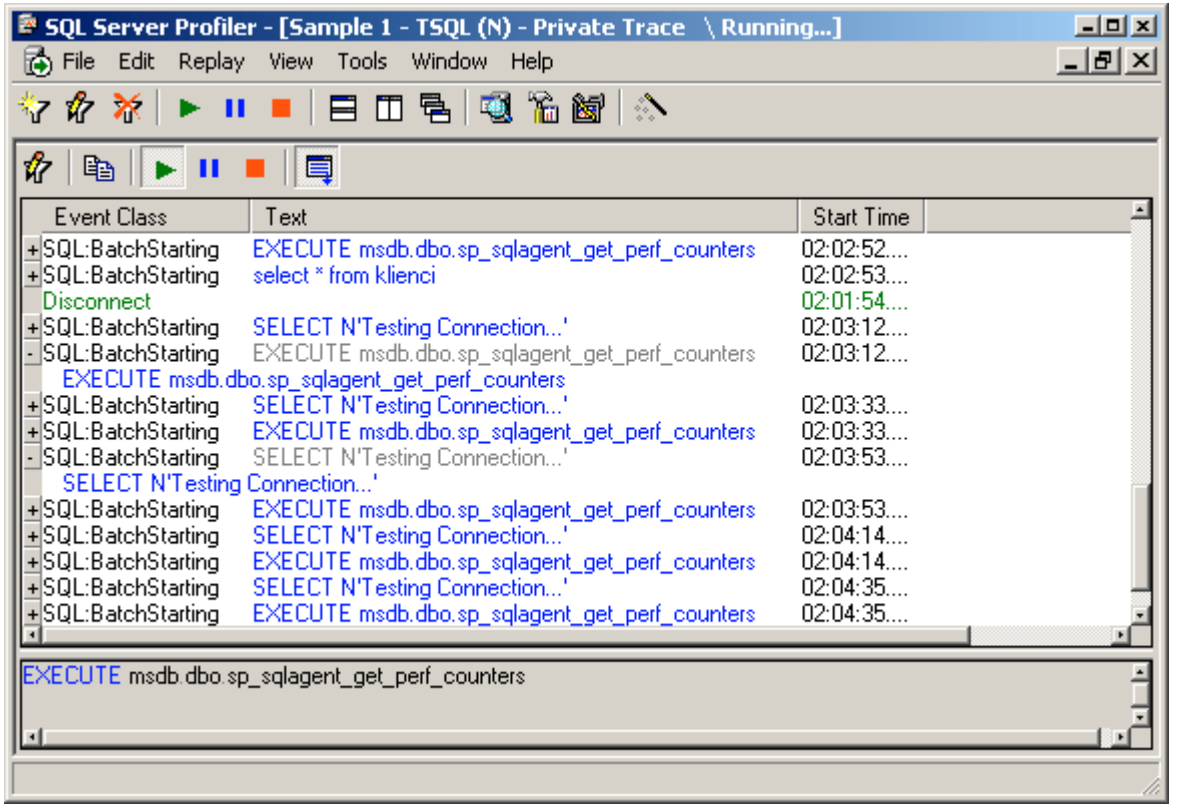


SQL Server - DTS (II)

- Active Scripting
- Aplikacje win32 przetwarzające dane
- Procesor skryptów SQL
- Transfer obiektów SQL Server
- Procesy Data Driven
- Agent poczty - wysyłanie na podstawie bazy danych
- Importy OLE danych tekstowych

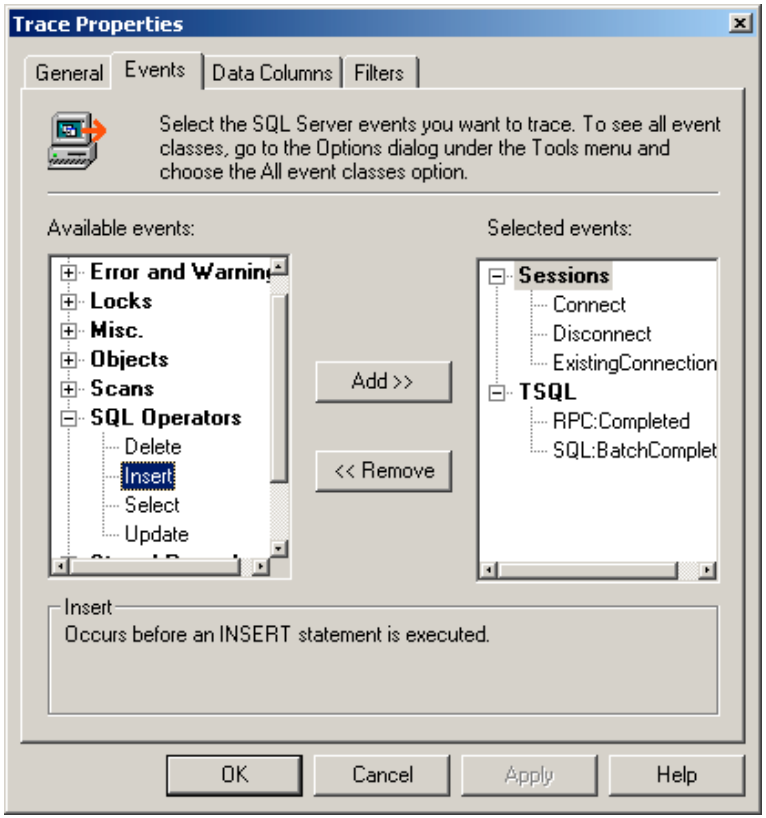
SQL Server Profiler (I)

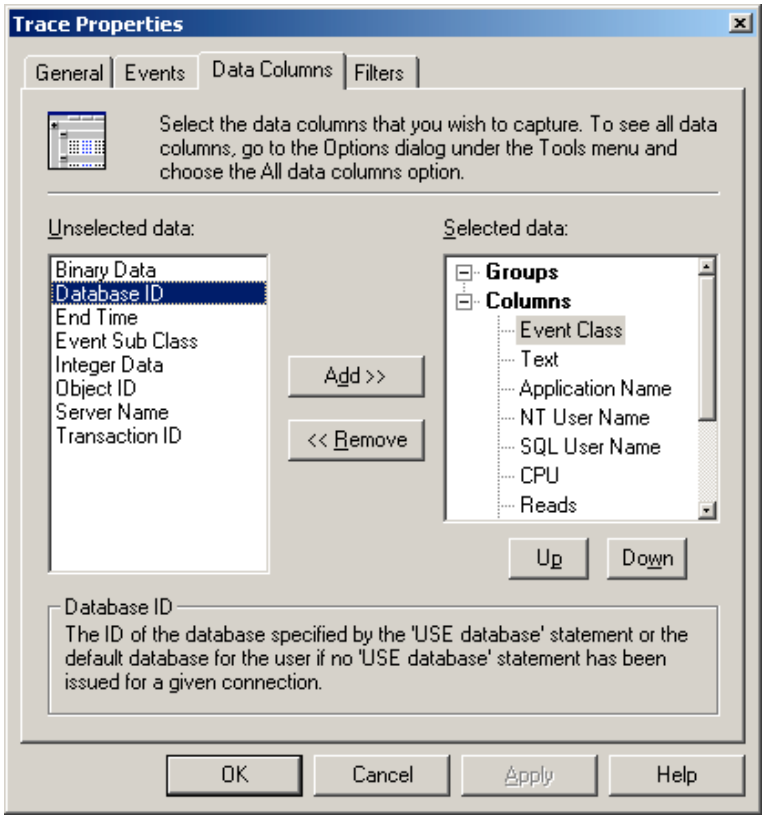
- Śledzenie ruchu w usługach SQL Server



SQL Server Profiler (II)

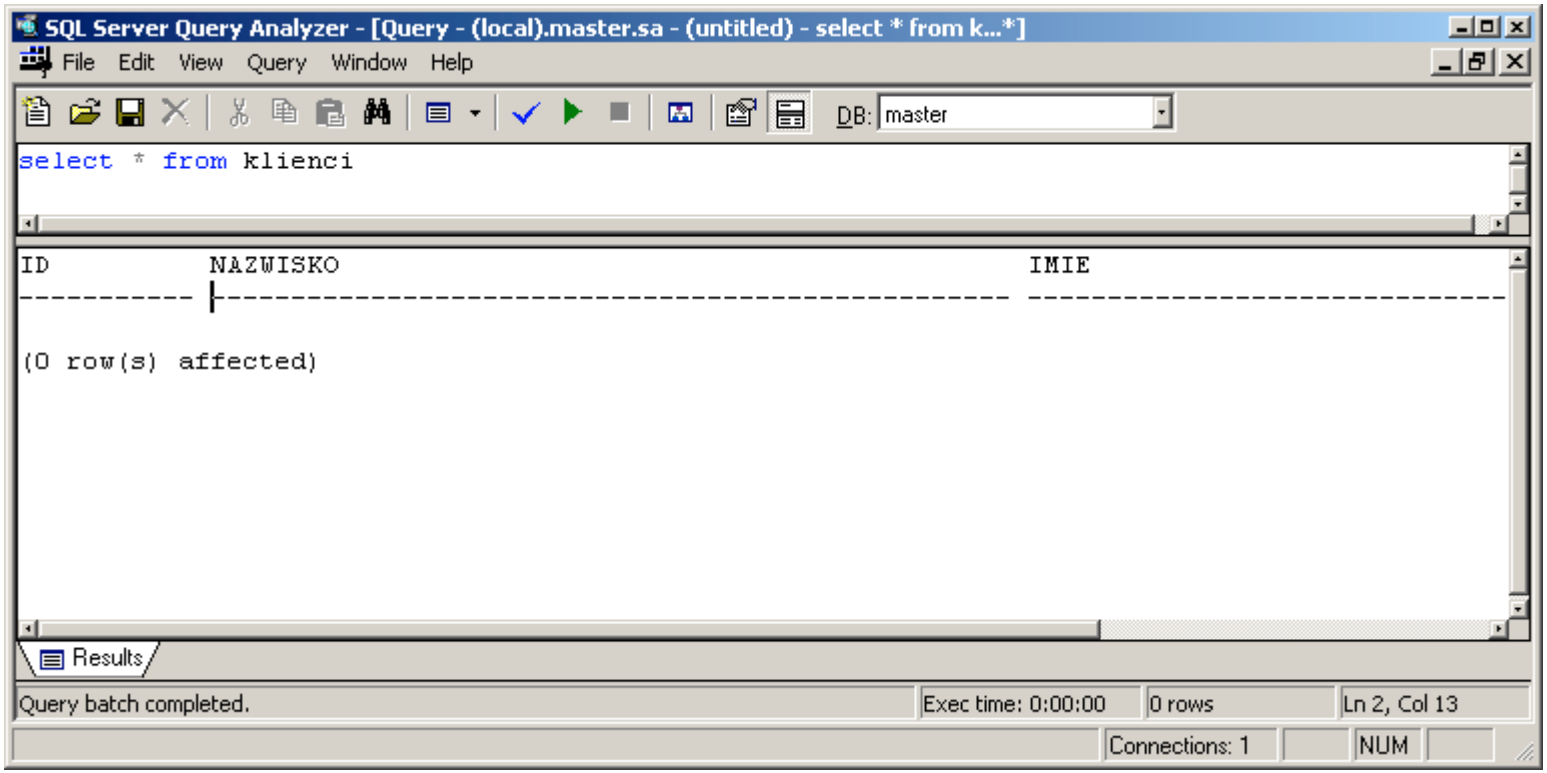
- Filtry śledzenia:





SQL Server Query Analyser

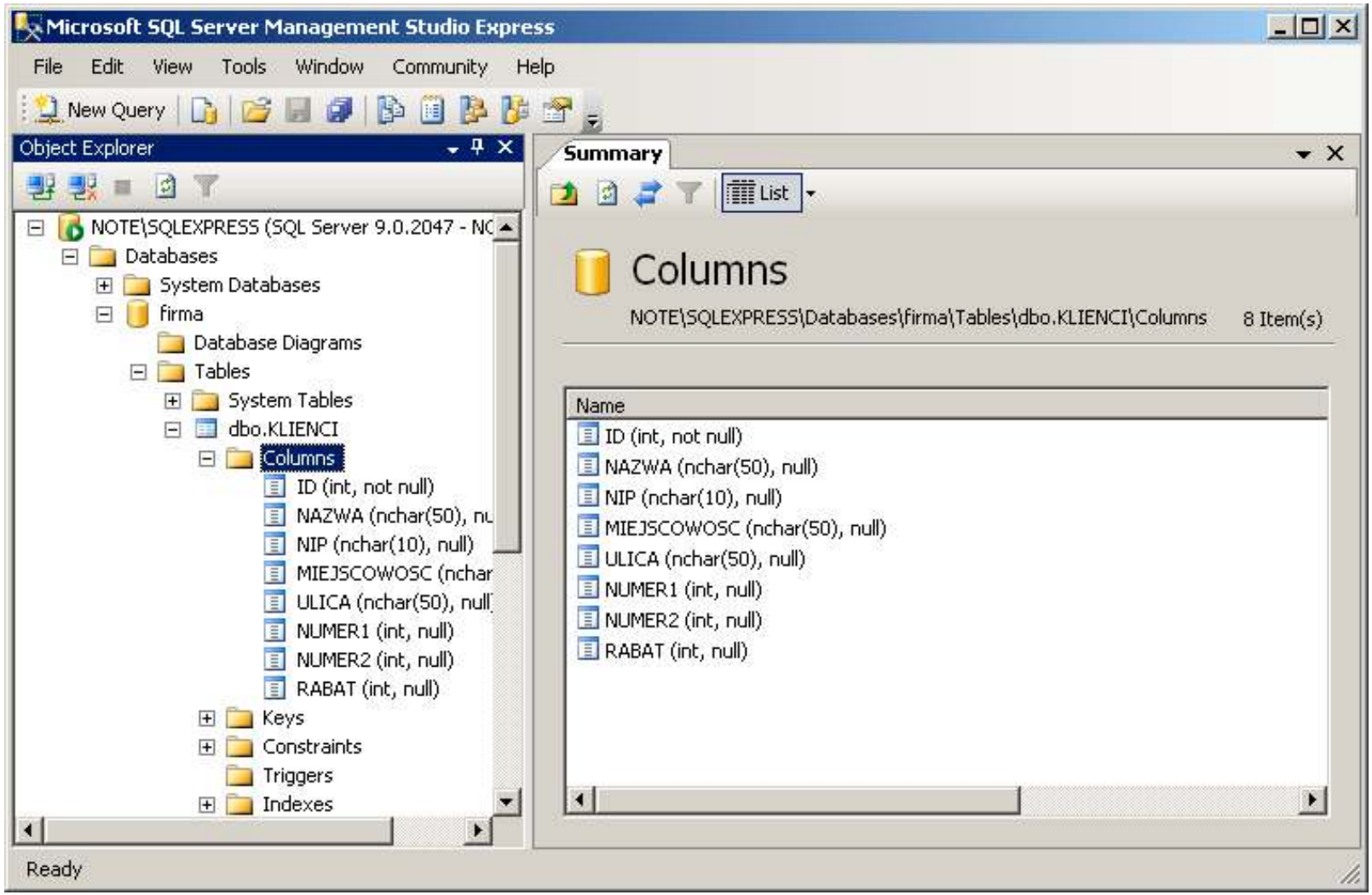
- Monitoring wyników realizacji zapytań zwracanych przez serwer. Możliwość realizacji zapytań modyfikujących bazę danych, także w trybie skryptowym.



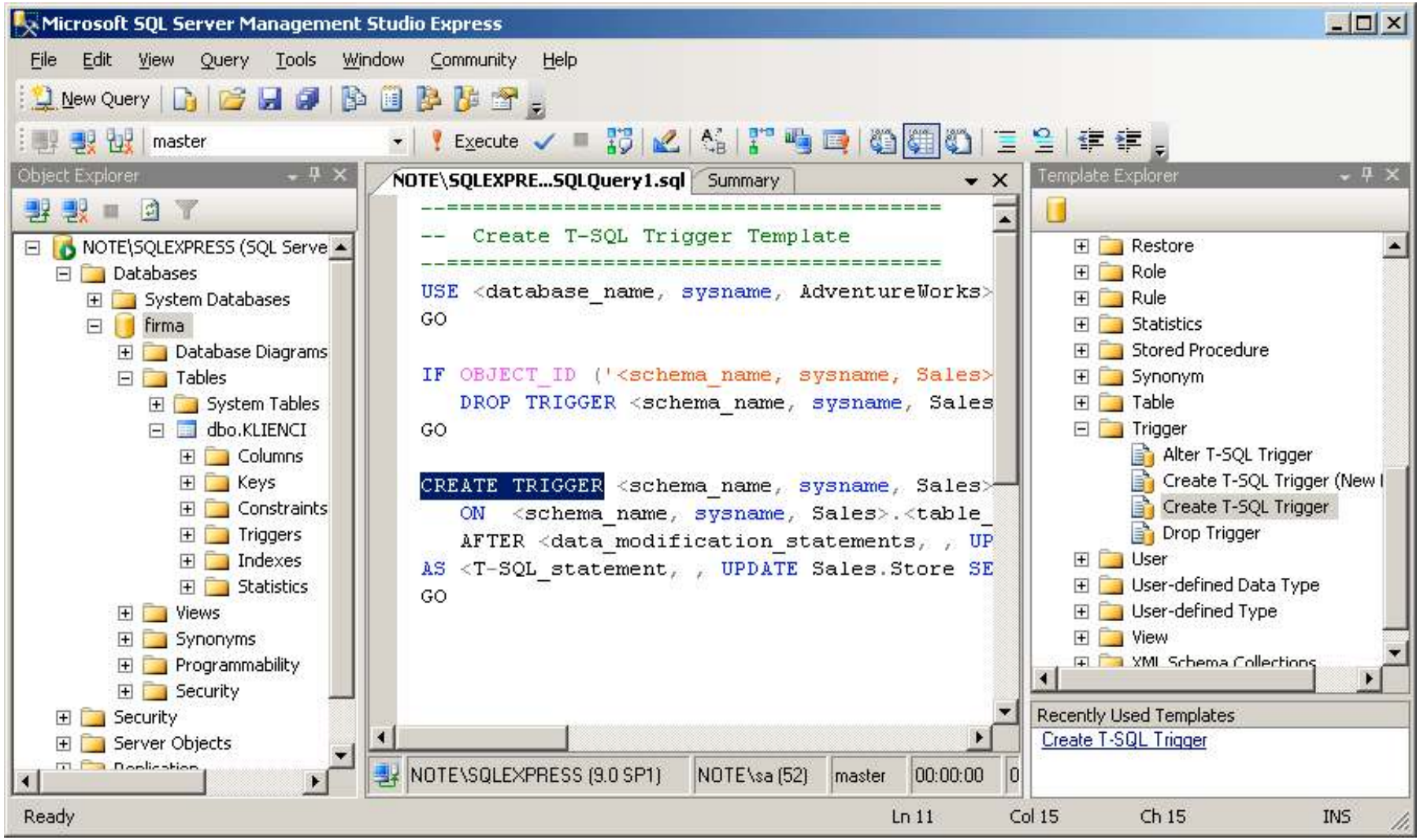
SQL Server Management Studio (I)

- Wymagania: .Net Framework, w późniejszych wersjach .Net Framework 2.0
- Konsola umożliwiająca graficzne zarządzanie bazami danych SQL Server.
- Wymaga indywidualnej instalacji w przypadku wersji SQL Server Express
  - nie wyposażonych w żadne graficzne narzędzie do zarządzania bazami danych

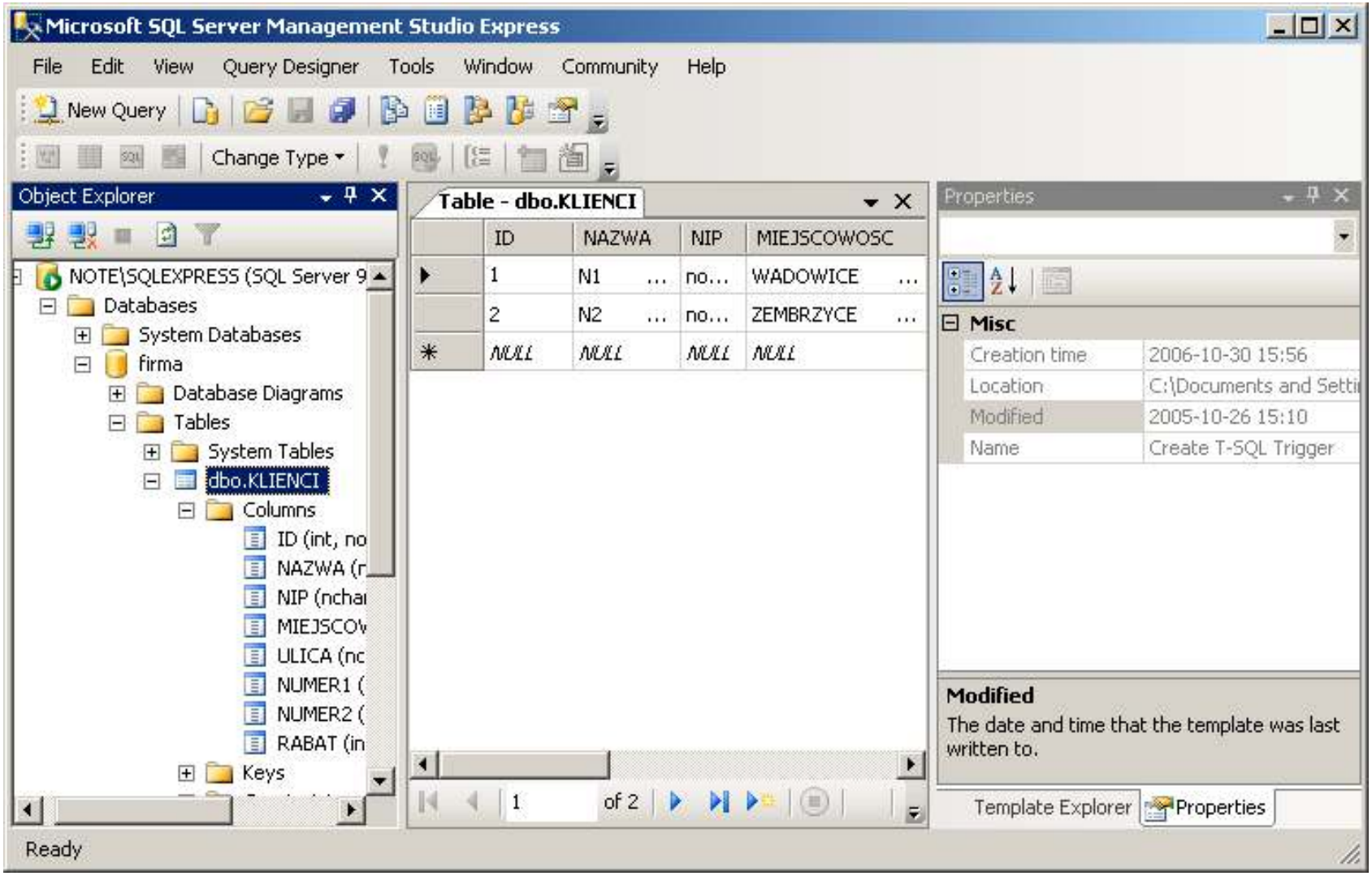
SQL Server Management Studio (II)



SQL Server Management Studio - generatory SQL



SQL Server Management Studio - interfejs edycji danych



Konsole tekstowe MS SQL Server (I)

- Starszy wariant:
  osql.exe
- Bieżący wariant:
  sqlcmd.exe

  Wspólne podstawowe opcje:
  - -S server - ustalenie docelowej instancji SQL Server
  - -U user - podanie użytkownika i hasła w przypadku zarządzania użytkownikami delegowanego do SQL Server

Konsole tekstowe MS SQL Server (II)

- Przykłady logowania:

  sqlcmd -S .\SQLEXPRESS
  sqlcmd -S (local) -U sa

- Komendy wpisywane do konsoli akumulują się w kolejnych liniach, aż do wydania polecenia ‘go’. Komend nie kończymy średnikiem.
- Przykład użycia SQL:

```
use [FIRMA]
INSERT INTO KLIENCI ([NAZWA], [NIP]) VALUES ('ARSENAL',
        '644-190-67-22')
SELECT * FROM KLIENCI
go
```

Użytkownicy w systemach Windows

- Konta użytkowników Windows organizowane są w jednym z dwóch systemów:
  - Domenowym (centralnie, na tzw. Kontrolerze domeny),
  - Grupy roboczej (indywidualnie na każdym komputerze).
- W obydwu przypadkach użytkownicy mogą być grupowani. Grupy (podobnie jak sami użytkownicy) także posiadają zdefiniowane uprawnienia.
- W Windows występują konta szablonowe (wbudowane), będące zestawami uprawnień.

Charakterystyka grupy roboczej

- *Grupa robocza* jest to mała grupa komputerów pracujących w sieci, która nie zapewnia centralnej administracji zasobami. Każdy komputer w grupie roboczej posiada własną przechowywaną lokalnie bazę kont *SAM* (ang. *Security Account Manager*) w związku z tym użytkownik musi mieć konto na każdym komputerze, do którego chce się zalogować. Podczas pracy w grupie roboczej nie obowiązuje zasada pojedynczego konta i pojedynczego logowania. Grupa robocza jest preferowana wyłącznie w małych sieciach biurowych oraz domowych, ponieważ w miarę zwiększania ilości komputerów pracujących w sieci, staje się trudna do zarządzania.

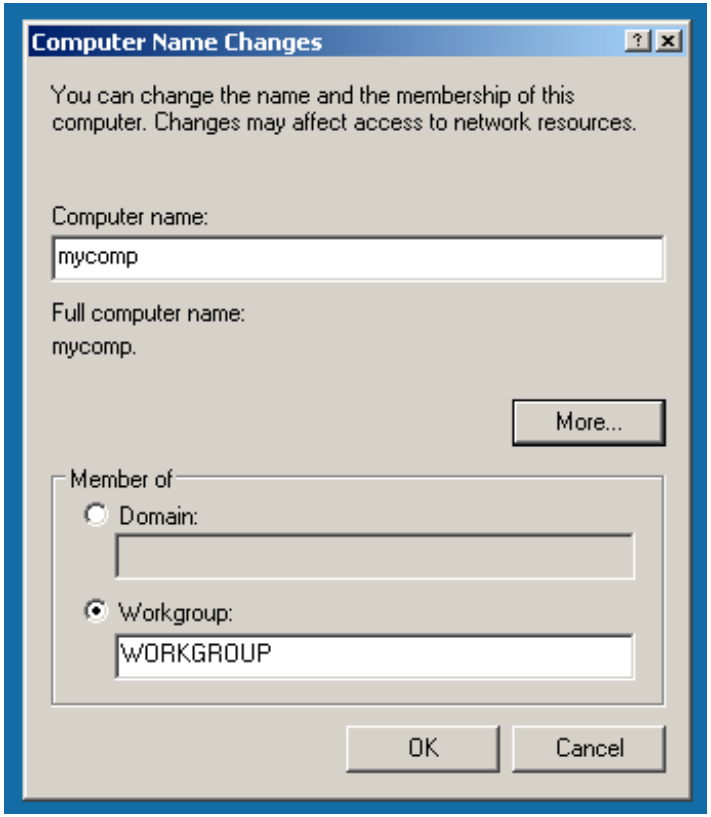
Charakterystyka domeny

- *Domena*, podobnie jak grupa robocza, jest logiczną grupą komputerów, lecz w przeciwieństwie do grupy roboczej w domenie występuje pojedyncza baza kont przechowująca wszystkie domenowe konta użytkowników, grup oraz komputerów. Wspomniana baza jest przechowywana na serwerach pełniących funkcję *Kontrolerów domeny* (ang. *DC — Domain Controller*) usługi *Active Directory*. Domena jest bardzo skalowalna, począwszy od kilku do tysięcy komputerów. Podczas pracy w domenie użytkownik potrzebuje wyłącznie jednego konta, aby logować się do dowolnych zasobów. To udogodnienie nazywamy *pojedynczym logowaniem* (ang. *Single logon process*)

Podłączanie komputera do domeny (I)

- W domenie każdy komputer posiada własne unikatowe konto komputera.
- W momencie przyłączania komputera do domeny użytkownik wykonujący operację przyłączenia musi posiadać konto użytkownika w usłudze *Active Directory* domeny, do której przyłączany jest komputer.
- W domenie musi istnieć konto komputera, którego nazwa jest identyczna z nazwą dodawanej maszyny.

Podłączanie komputera do domeny (II)



Konta użytkowników

- *Konto użytkownika* - unikatowy obiekt zawierający informacje o prawach, jakie posiada użytkownik i pozwalający na na logowanie do systemu oraz uzyskiwanie dostępu do zasobów sieciowych
- Możliwość personalizacji środowiska bez wpływu na pracę i dane innych użytkowników komputera
- Mechanizmy nadawania uprawnień to efektywne i bezpieczne korzystanie z zasobów

Podział kont użytkowników

- *Lokalne konta użytkowników*,
- *Domenowe konta użytkowników*,
- *Konta wbudowane*.

Lokalne konto użytkownika

- Najczęściej wykorzystywane podczas pracy w małej grupie roboczej lub na komputerach autonomicznych
- Umożliwia użytkownikowi logowanie do komputera oraz dostęp do zasobów
- Umożliwia dostęp do zasobów wyłącznie na komputerze, w którego bazie zostało stworzone
- Aby skorzystać przez sieć z zasobów na innym komputerze, użytkownik musi posiadać konto lokalne w bazie tamtego komputera

Domenowe konto użytkownika

- Wykorzystywane wyłącznie wtedy, gdy komputer pracuje w domenie
- Konta domenowe są zakładane przez administratora domeny w usłudze *Active Directory*
- Konto domenowe umożliwia logowanie do domeny - i uzyskanie dostępu do określonych zasobów w całej domenie
- Podczas logowania do komputera po wpisaniu przez użytkownika nazwy i hasła dane są przesyłane przez sieć do kontrolera domeny, który autoryzuje użytkownika, sprawdzając bazę kont

Wbudowane konta użytkowników

- Tworzone automatycznie podczas instalacji systemu lub kontrolera domeny
- Zapewniają możliwość wykonywania zadań administracyjnych w systemie
- *Administrator* i *Gość* - dwa trwałe konta w systemie

Typy lokalnych kont użytkowników

- Zróżnicowane pod kątem przynależności do grupy *Użytkownicy* lub *Administratorzy*
- *Podział:*
  - *Konto z ograniczeniami*
  - *Standardowe konto użytkownika*
  - *Administrator komputera*

Lokalne konto z ograniczeniami

- Należy do grupy „Użytkownicy”
- Uprawnienia:
  - Zmiana obrazu skojarzonego z kontem użytkownika.
  - Zmiana własnego hasła.
  - Usuwanie własnego hasła.
  - Dostęp do własnych plików

Lokalne standardowe konto użytkownika

- Należy do grupy „Użytkownicy”, czasem modyfikowanej i zwanej „użytkownicy zaawansowani”
- Uprawnienia:
  - Prawa, jakie posiada konto z ograniczeniami.
  - Możliwość zmiany podstawowych ustawień komputera takich jak np. ustawienia wyświetlania.

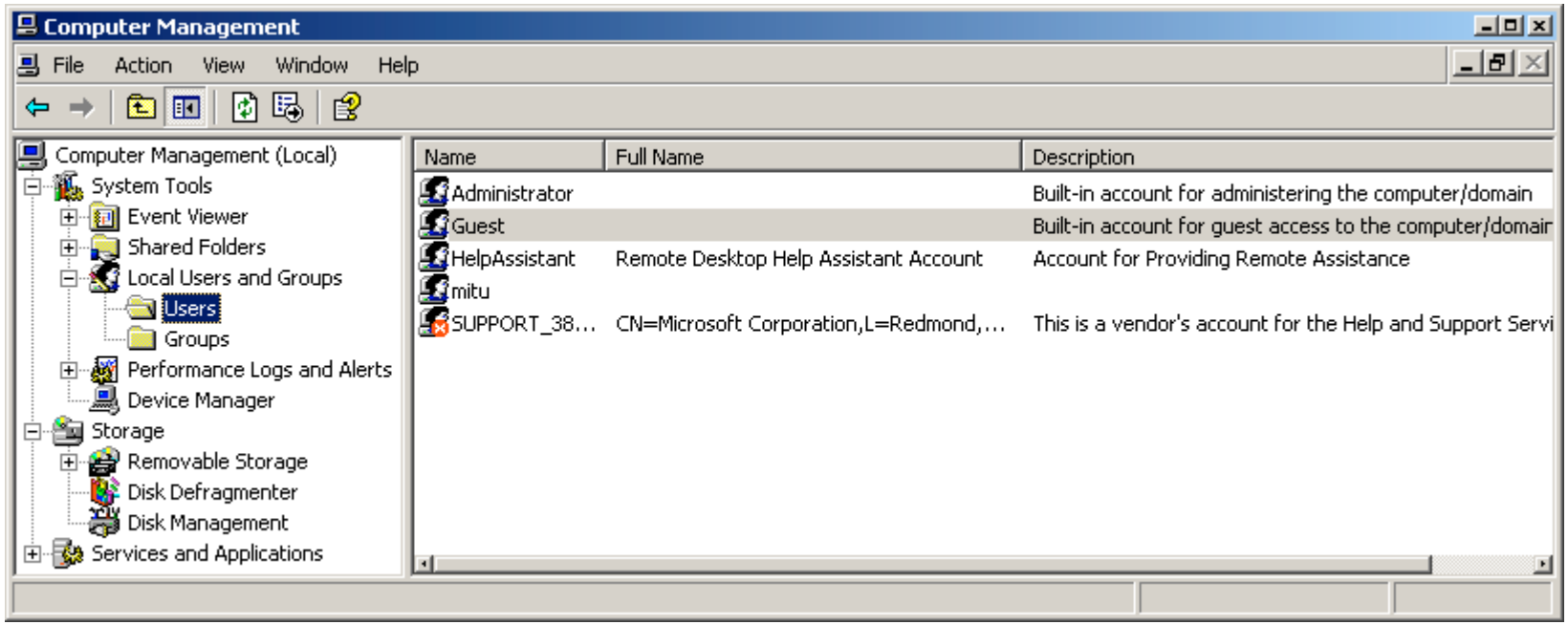
Lokalne konto administratora

- Należy do grupy „Administratorzy”
- Uprawnienia:
  - Prawa konta standardowego
  - Tworzenie, zmiana i usuwanie kont użytkowników
  - Zmiany w konfiguracji komputera
  - Dostęp do wszystkich plików na komputerze
  - Instalacja sprzętu i oprogramowania

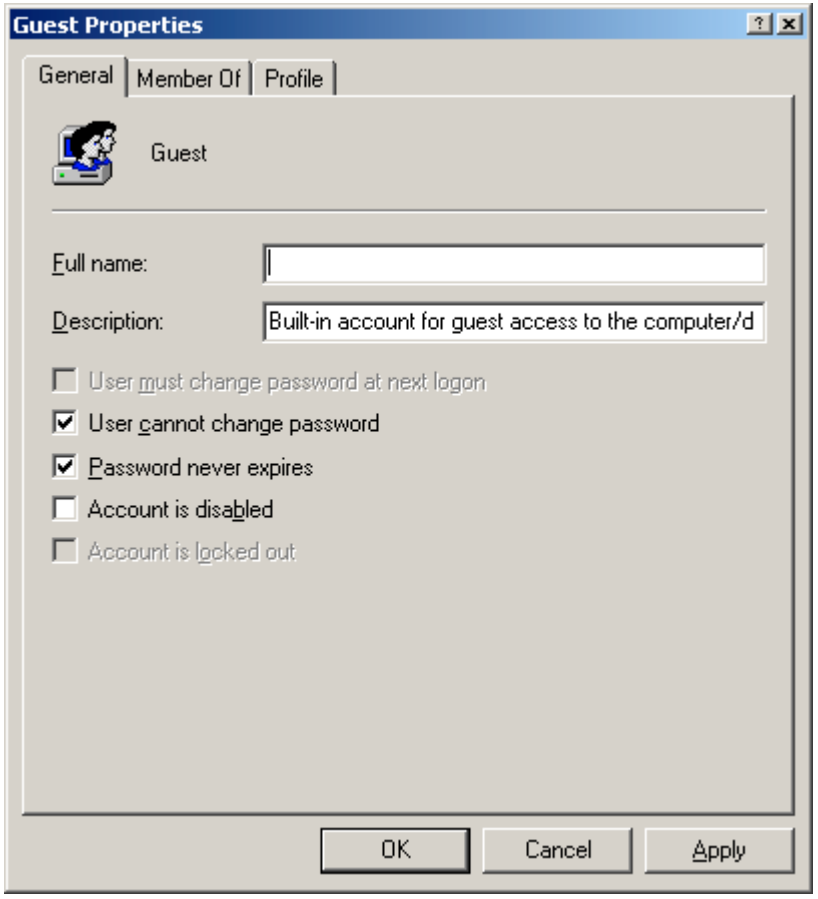
Zarządzanie kontami (I)

- Do pełnego zarządzania kontami użytkowników służy odpowiedni applet MMC
- Obsługa skrócona dostępna jest poprzez link towarzyszący procesowi logowania/ zmiany użytkownika
- Nazwa i lokalizacja appletu: WindowsDir/system32/compmgmt.msc

Zarządzanie kontami (II)



Modyfikacja konta



Profile użytkownika (I)

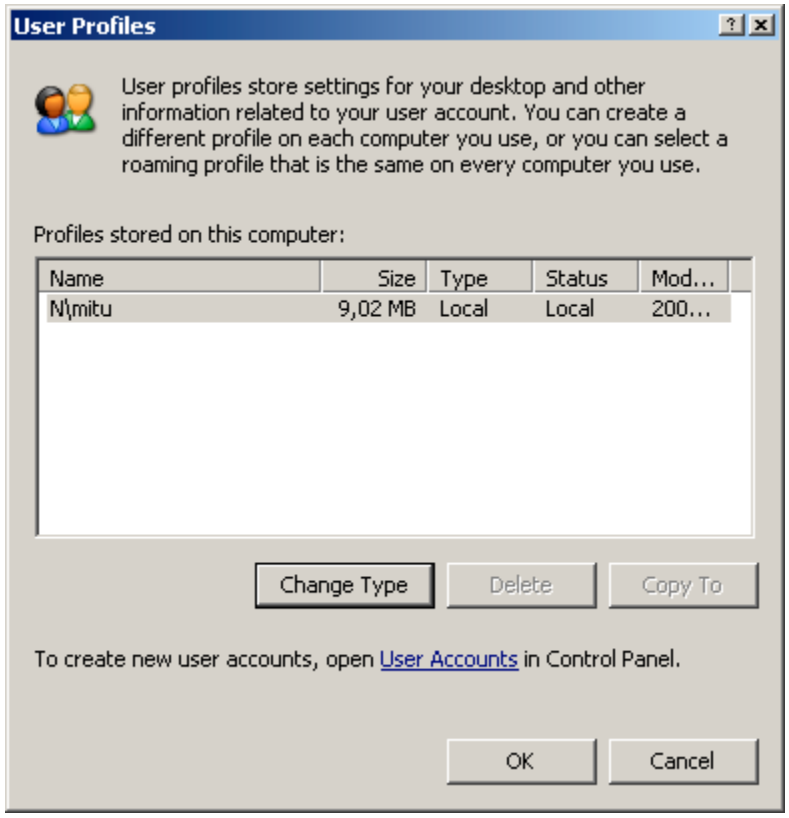
- **Profil użytkownika** to zbiór ustawień środowiska pracy, do którego zaliczamy ustawienia pulpitu, menu *Start*, ustawienia aplikacji, folder *Moje dokumenty*, pliki *cookies*
- **Profile lokalne** - przechowywane lokalnie na stacji - na partycji rozruchowej w folderze *Documents and Settings*, w którym każdy użytkownik choć raz zalogowany do komputera, posiada swój folder profilu o nazwie zgodnej z nazwą użytkownika

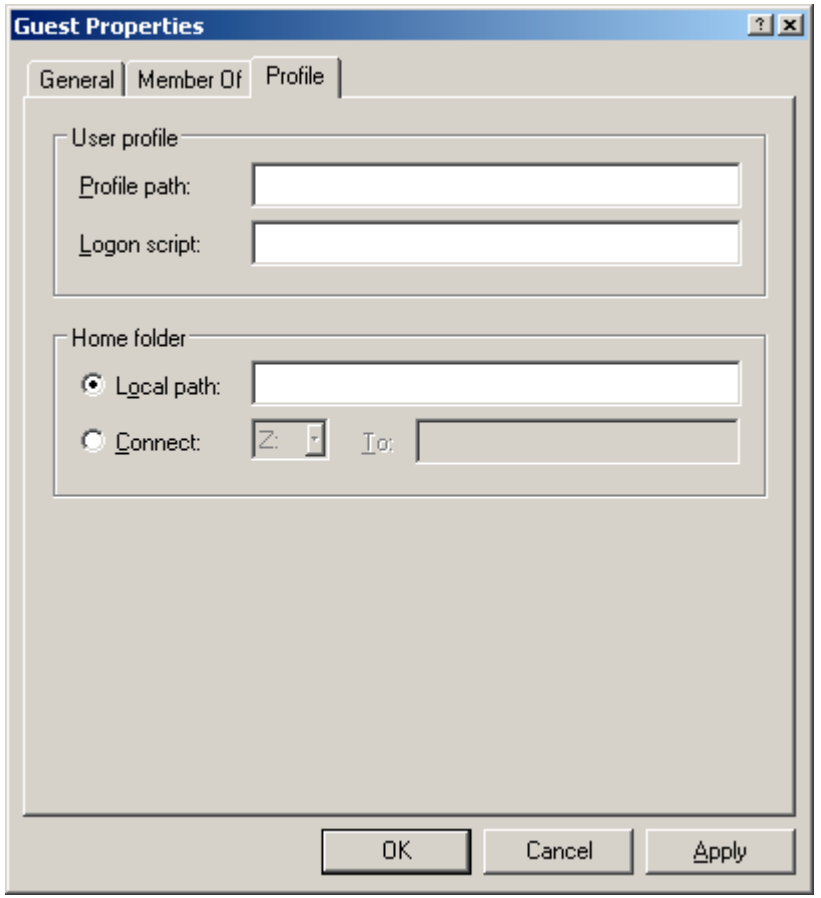
Profile użytkownika (II)

- **Profile wędrujące (mobilne)** (ang. *Roaming User Profile*) - wykorzystywane głównie podczas pracy w domenie
- Profile mobilne są przechowywane w sieciowym udziale (udostępnionym folderze), do którego użytkownik ma dostęp z każdego komputera pracującego w sieci
- Profil ten jest pobierany przez sieć do komputera lokalnego podczas logowania zdalnego

Profile użytkownika (III)

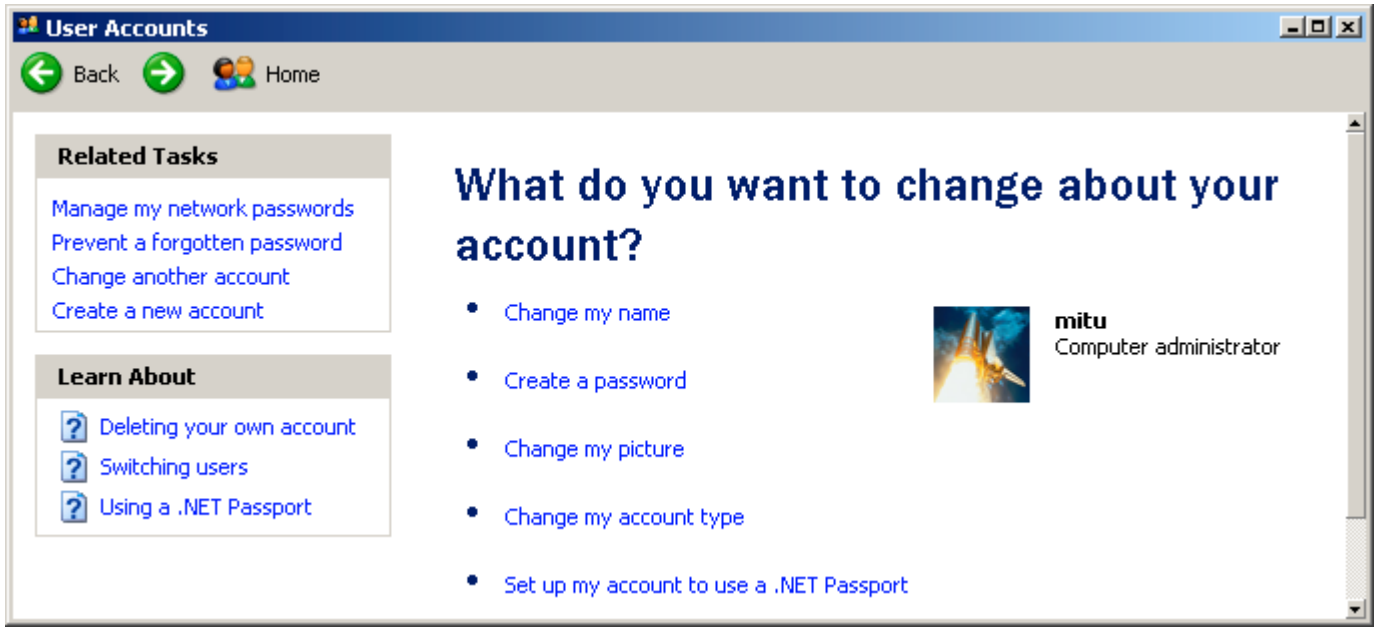
- Zarządzanie profilami





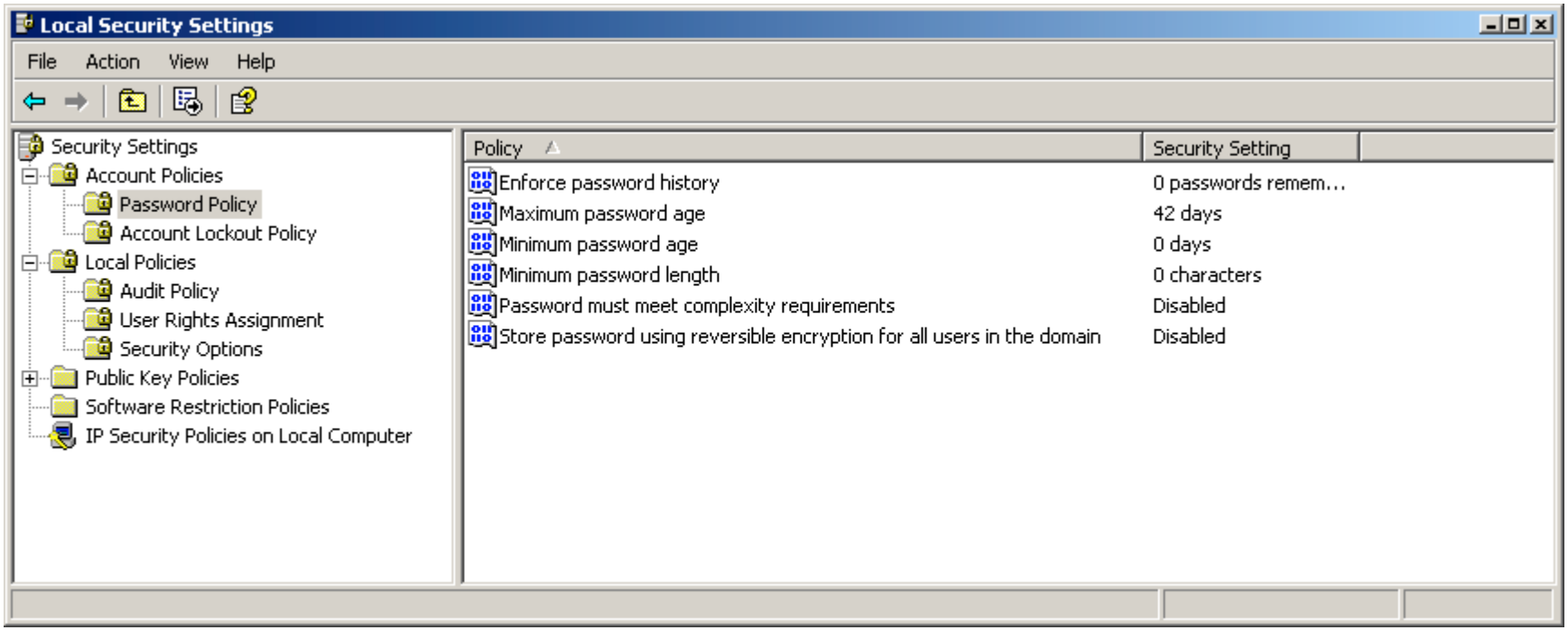
Hasła - problem zapominania haseł

- Metody odzyskania: backup na dysku wymiennym lub logowanie hasłem przechowywanym na zdalnej maszynie



Hasła - polityka haseł

- Ustawianie wymogów dla haseł użytkowników:



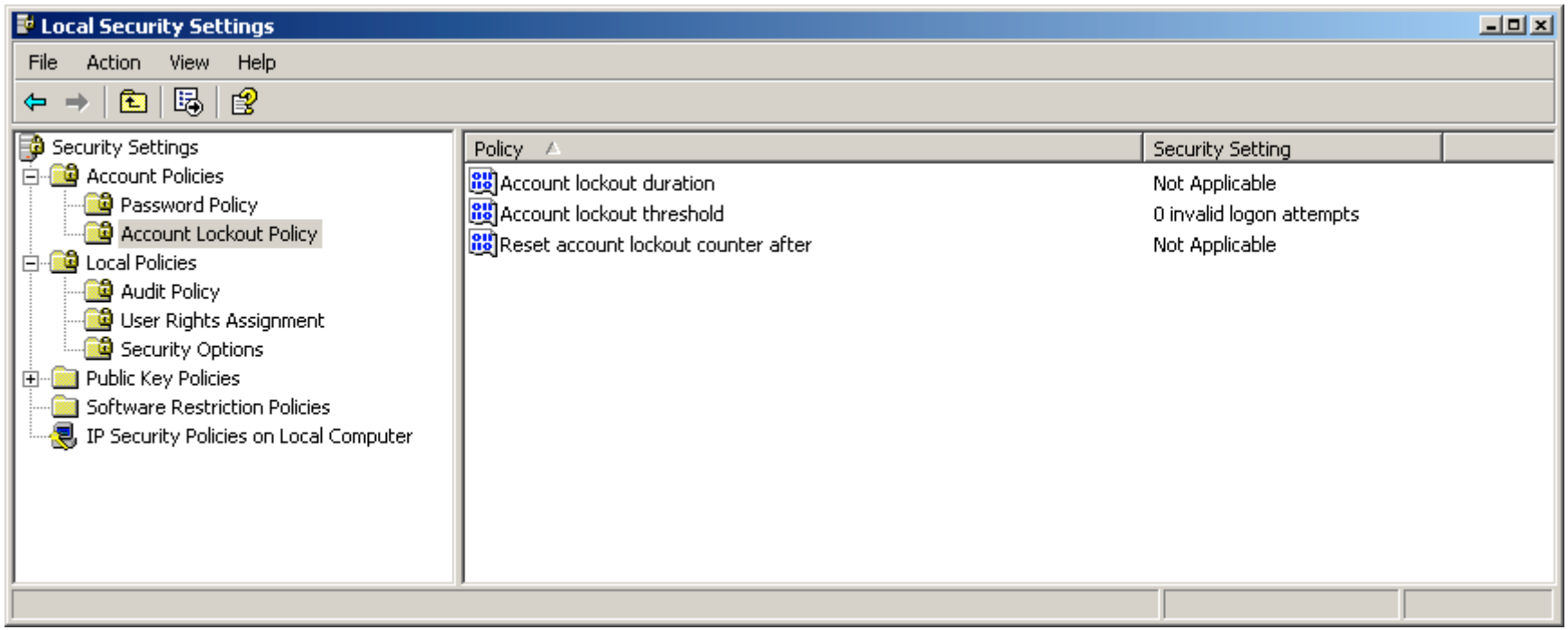
Hasła - wymogi dla haseł w Windows

- Hasła systemu Windows XP, 2000, Server mogą liczyć maksymalnie 127 znaków,
- Hasła systemu Windows 95 lub Windows 98 mogą liczyć maksymalnie 14 znaków
- Hasła mogą zawierać znaki z trzech kategorii:
  - Litery: od A do Z, oraz od a do z
  - Cyfry: od 0 do 9
  - Znaki specjalne (` ~ ! @ # $ % ^ & * ( ) _ + - = { } | [ ] \ : " ; ' < > ? , . /).

Hasła - wymogi dla haseł „silnych” w Windows

- Nie mogą zawierać fragmentu lub całej nazwy konta użytkownika ani jego nazwiska,
- Muszą mieć długość minimum siedmiu znaków,
- Muszą (nie zawsze) zawierać znaki z wszystkich trzech kategorii: Litery, Cyfry, Znaki specjalne
- Muszą znacznie różnić się od poprzednich haseł,
- Muszą mieć co najmniej jeden znak specjalny na pozycji od drugiej do szóstej.

Hasła - blokowanie dostępu



Grupy

- Grupy służą do łączenia użytkowników w celu łatwiejszego zarządzania uprawnieniami
- Użytkownicy przejmują prawa, które zostały nadane grupie
- Użytkownicy mogą być członkami wielu grup jednocześnie
- Grupy dzielimy na lokalne i domenowe
- Dodatkowy rodzaj grup - *grupy wbudowane*, którymi nie można zarządzać tak jak pozostałymi

Charakterystyka grup wbudowanych (I)

- **Administratorzy**
  Członkowie tej grupy mają całkowitą kontrolę nad komputerem. Prawa, które nie zostały nadane grupie Administratorzy automatycznie, każdy członek tej grupy może sobie nadać. Administratorzy mogą tworzyć, usuwać oraz modyfikować konta wszystkich użytkowników i grup. Mogą również nadawać uprawnienia do wszystkich zasobów komputera.
- Inne prawa:
  - Instalacja systemu operacyjnego oraz jego komponentów uwzględniając sterowniki, urządzenia sprzętowe, usługi systemowe itd.
  - Instalacja pakietów *Service Pack* oraz łat *hot fix*.
  - Naprawa systemu operacyjnego.
  - Przejmowanie własności obiektów.

Charakterystyka grup wbudowanych (II)

- **Operatorzy kopii zapasowych**
  Użytkownicy należący do tej grupy mogą tworzyć i odtwarzać kopie zapasowe niezależnie od posiadanych uprawnień do plików i folderów. Grupa ta powinna zawierać wyłącznie konta osób odpowiedzialnych za wykonywanie kopii zapasowych.
- **Goście**
  Domyślnie członkowie grupy Goście mają zablokowany dostęp do dzienników zdarzeń aplikacji i systemu. Po za tym członkowie tej grupy mają takie same prawa jak grupa Użytkownicy. Domyślnie członkiem grupy Goście jest tylko konto Gość, które jest wyłączone.

Charakterystyka grup wbudowanych (III)

- **Grupa usług pomocy**
  Członkowie tej grupy mogą pomagać w diagnozowaniu problemów z systemem. Grupa ta może być wykorzystana przez *Centrum pomocy i obsługi technicznej* podczas dostępu do komputera z sieci lub lokalnie.
- **Operatorzy konfiguracji sieci**
  Członkowie tej grupy mają ograniczone uprawnienia administracyjne, które pozwalają na konfigurowanie ustawień sieci takich jak adres IP.

Charakterystyka grup wbudowanych (IV)

- **Użytkownicy zaawansowani**
  Grupa Użytkownicy zaawansowani posiada mniejsze prawa do systemu niż grupa **Administratorzy**, ale większe niż grupa **Użytkownicy**. Domyślnie członkowie tej grupy posiadają uprawnienia odczytu i zapisu do większości katalogów systemu operacyjnego. **Użytkownicy zaawansowani** mogą wykonywać wiele operacji konfigurujących system takich jak zmiana czasu systemowego, ustawień wyświetlania, tworzenie kont użytkowników i udostępnianie zasobów. Osoby należące do grupy **Użytkownicy zaawansowani** mogą instalować większość oprogramowania, jednakże może dojść do sytuacji, kiedy uprawnienia tej grupy będą niewystarczające.
- **Replikator**
  Członkowie tej grupy mają możliwość kopiowania plików systemowych.

Charakterystyka grup wbudowanych (V)

- **Użytkownicy pulpitu zdalnego**
  Użytkownicy kont należących do tej grupy mogą logować się zdalnie.
- **Użytkownicy**
  Użytkownicy mają ograniczony dostęp do systemu. Domyślnie członkowie grupy **Użytkownicy** mają uprawnienie odczytu i zapisu tylko do swojego profilu. Grupa ta, podczas gdy komputer pracuje w grupie roboczej lub autonomicznie jest nazywana **użytkownicy z ograniczeniami**. Została zaimplementowana po to by swoim członkom nadać uprawnienia niezbędne do pracy na komputerze, a jednocześnie zapobiegać modyfikacji konfiguracji i ustawień komputera

Uprawnienia i zabezpieczenia w systemie Windows (I)

- Podstawa funkcjonowania - listy ACL (Access Control List)
  - Zbiory rekordów opisywanych łańcuchami (werbalnie),
  - Zbiory powiązane są z obiektami.
- Podział:
  - Access Control Lists (ACL) - dotyczą zasobów lokalnych (plików i udziałów)
  - Discretionary Access Control Lists (DACL) - dotyczą zasobów Active Directory,
  - System access control lists (SACL) - dotyczą reguł inspekcji.
- W ramach każdego typu - Access Control Entry (ACE).

Uprawnienia i zabezpieczenia w systemie Windows (II)

- Dodatkowy podział w ramach każdego ze zbiorów:
  - Expicit Rules - reguły dotyczące bezpośrednio danego obiektu (w tym Explicit Deny, Explicit Allow)
  - Inherited Rules - reguły przekazywane do potomnków obiektu - np z nadkatalogu w stosunku do plików (w tym Parent Allow, Parent Deny)
- Każde ACE zawiera:
  - Identyfikator SID (Security IDentifier), który jednoznacznie identyfikuje podmiot zabezpieczeń (użytkownik, grupa, komputer itp),
  - Maskę dostępu (access mask) wskazującą na uprawnienia podmiotu zabezpieczeń w stosunku do obiektu.

Uprawnienia i zabezpieczenia w systemie Windows (III)

- Kodowanie ACL - dziedziczenie uprawnień:
  - **IO**: Tylko dziedziczenie — Ta flaga oznacza, że dany wpis ACE nie dotyczy bieżącego obiektu.
  - **CI**: Dziedziczenie na poziomie kontenera — Ta flaga oznacza, że podrzędne kontenery będą dziedziczyć ten wpis ACE.
  - **OI**: Dziedziczenie na poziomie obiektu — Ta flaga oznacza, że podrzędne pliki będą dziedziczyć ten wpis ACE.
  - **NP**: Bez propagowania – Ta flaga oznacza, że podrzędne obiekty nie będą propagować dziedziczonego wpisu ACE.
- Kodowanie ACL - uprawnienia globalne:
  - **F**: Pełna kontrola, **C**: Zmiana, **W**: Zapis itp.
- Kodowanie ACL - upranienia szczegółowe:
  - **GENERIC_READ, GENERIC_EXECUTE** itp

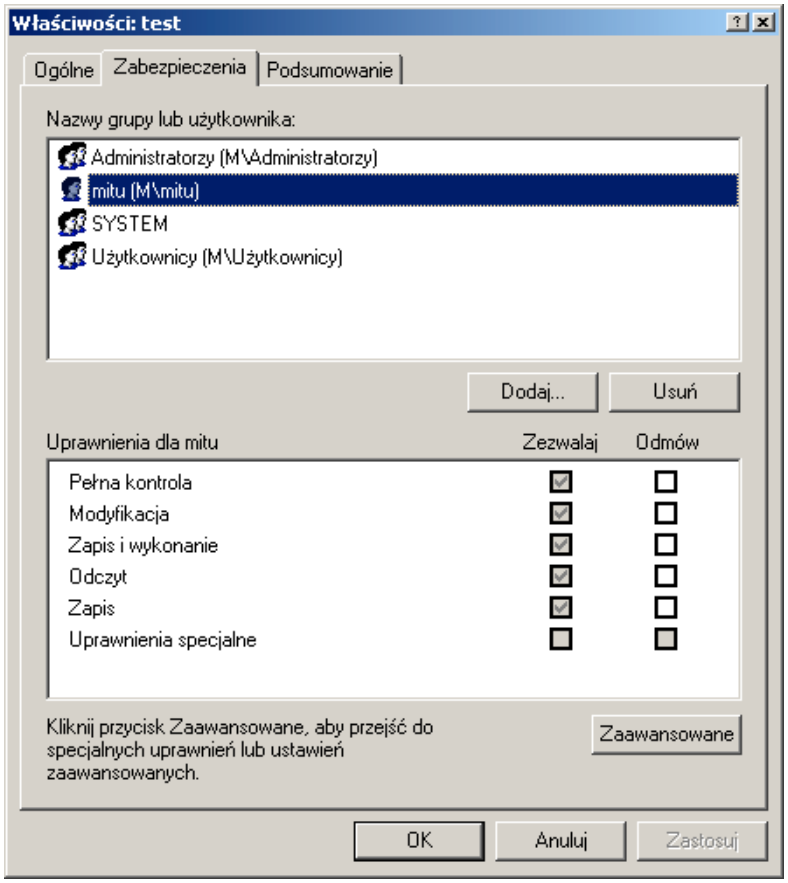
Uprawnienia i zabezpieczenia w systemie Windows (IV)

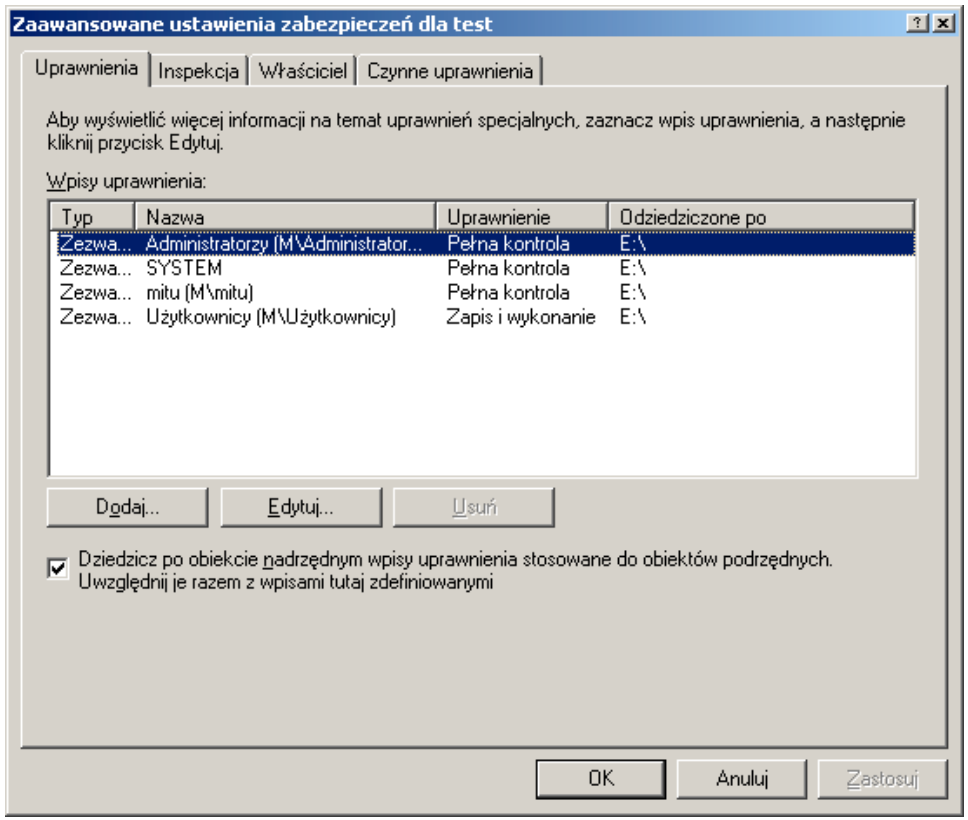
- Przykłady ACL (dla pliku):
  - BUILTIN\Users:(CI)(IO)(special access): GENERIC_READ
    - odczyt dla grupy wbudowanej Users z dziedziczeniem do podkatalogów podrzędnych (CI) i do obiektów bezpośrednio w podkatalogach (IO)
  - NOTE\mitu:F
    - pełna konrola dla użytkownika mitu
  - BUILTIN\Administrators:F
    - pełna kontrola dla grupy wbydowanej users

Prawa dostępu do plików i folderów

- Lista uprawnień (przykład dla systemów NT/2000/XP):
  - Pełna kontrola
  - Modyfikacja
  - Odczyt i wykonanie
  - Wyświetlanie zawartości folderu (tylko foldery)
  - Odczyt
  - Zapis
- Uprawnienia są dziedziczone zgodnie z drzewem katalogowym

ACL - Ustawianie praw dostępu do plików





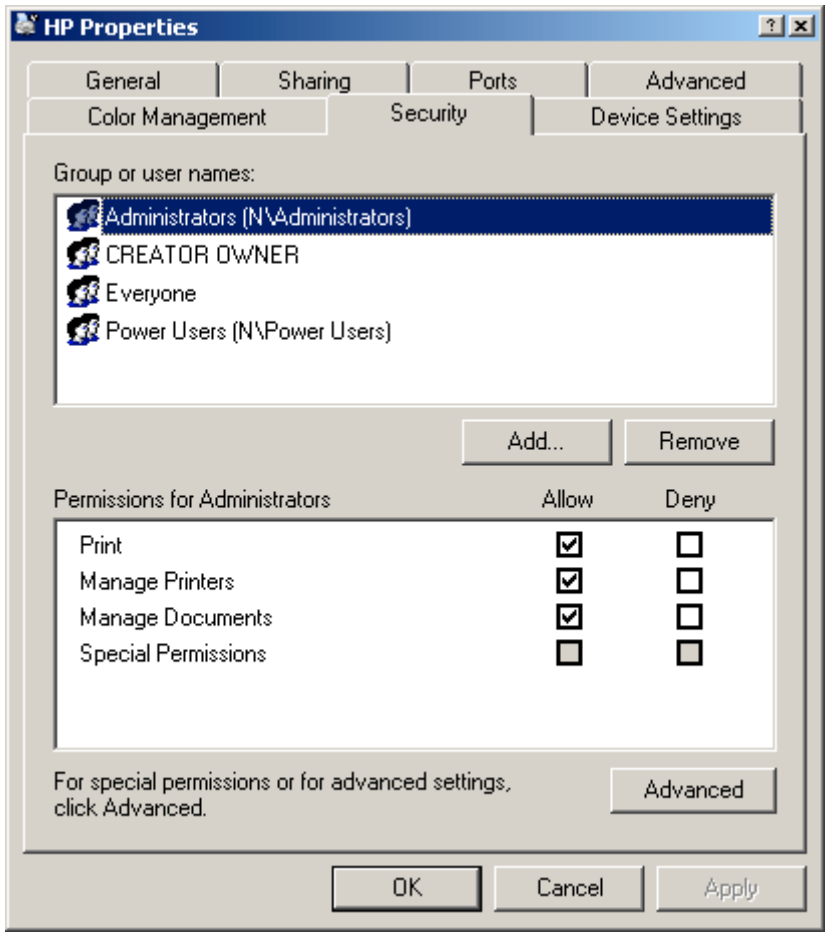
Szczegóły praw dostępu do plików i folderów (I)

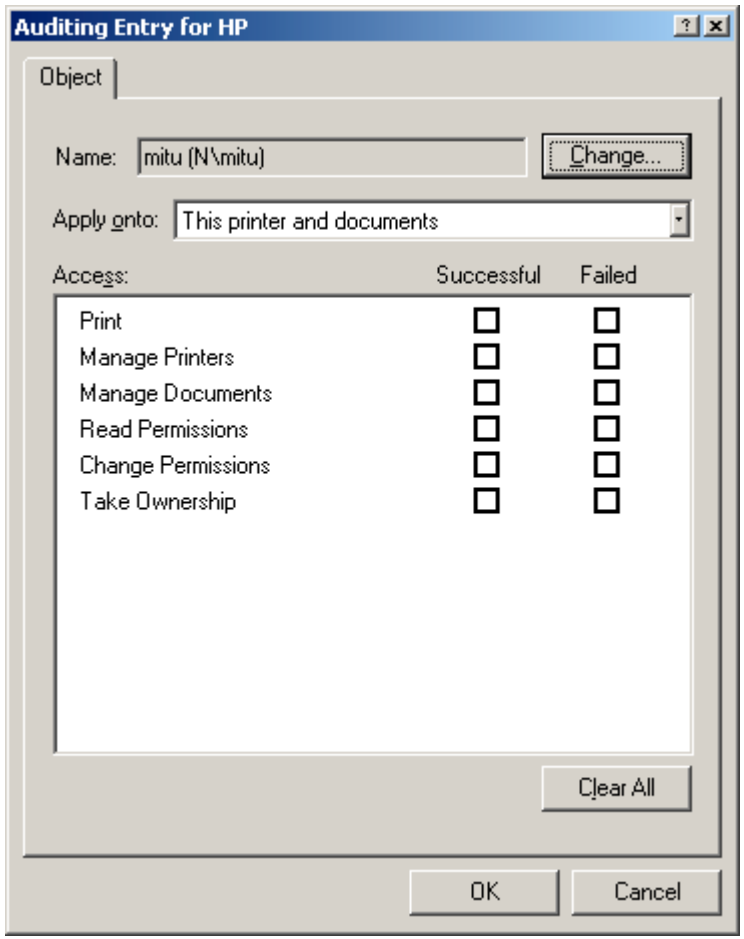
Uprawnienia specjalne	Pełna kontrola	Modyfikacja	Odczyt i wykonanie	Wyświetlanie zawartości folderu (tylko foldery)	Odczyt	Zapis
Przechodzenie przez folder/Wykonywanie pliku	x	x	x	x		
Wyświetlanie zawartości folderu/Odczyt danych	x	x	x	x	x	
Odczyt atrybutów	x	x	x	x	x	
Odczyt atrybutów rozszerzonych	x	x	x	x	x	
Tworzenie plików/Zapis danych	x	x				x
Tworzenie folderów/Dołączanie danych	x	x				x

Szczegóły praw dostępu do plików i folderów (II)

Uprawnienia specjalne	Pełna kontrola	Modyfikacja	Odczyt i wykonanie	Wyświetlanie zawartości folderu (tylko foldery)	Odczyt	Zapis
Zapis atrybutów	x	x				x
Zapis atrybutów rozszerzonych	x	x				x
Usuwanie podfolderów	x					
Usuwanie	x	x				
Odczyt uprawnień	x	x	x	x	x	x
Zmiana uprawnień	x					
Przejęcie na własność	x					

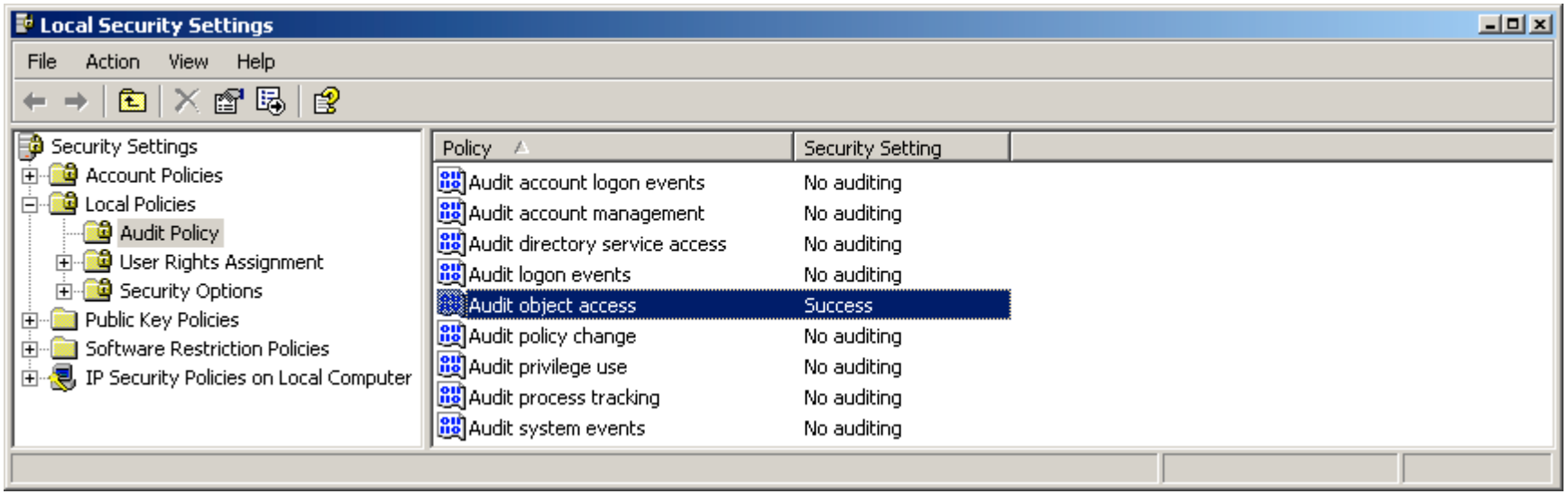
ACL - Uprawnienia użytkowników do drukowania





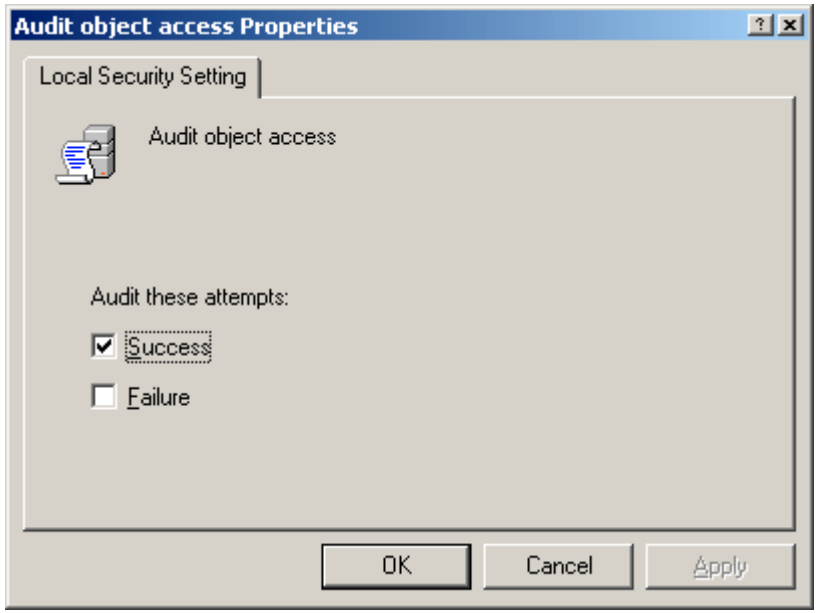
Audyty dostępu do plików (I)

- Możliwość monitorowania różnorodnych zdarzeń systemowych



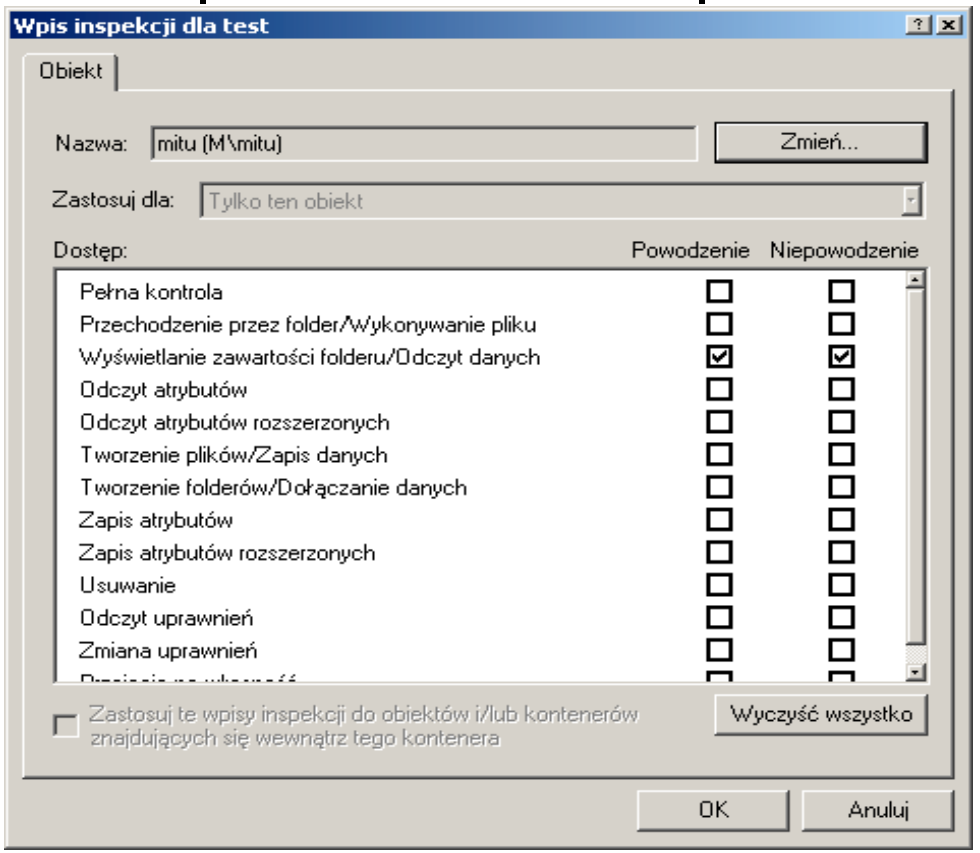
Audyty dostępu do plików (II)

- Audyt danego zdarzenia rozróżnia operacja zakończone powodzeniem i zakończone porażką (np odmowa dostępu)



Audyty dostępu do plików (III)

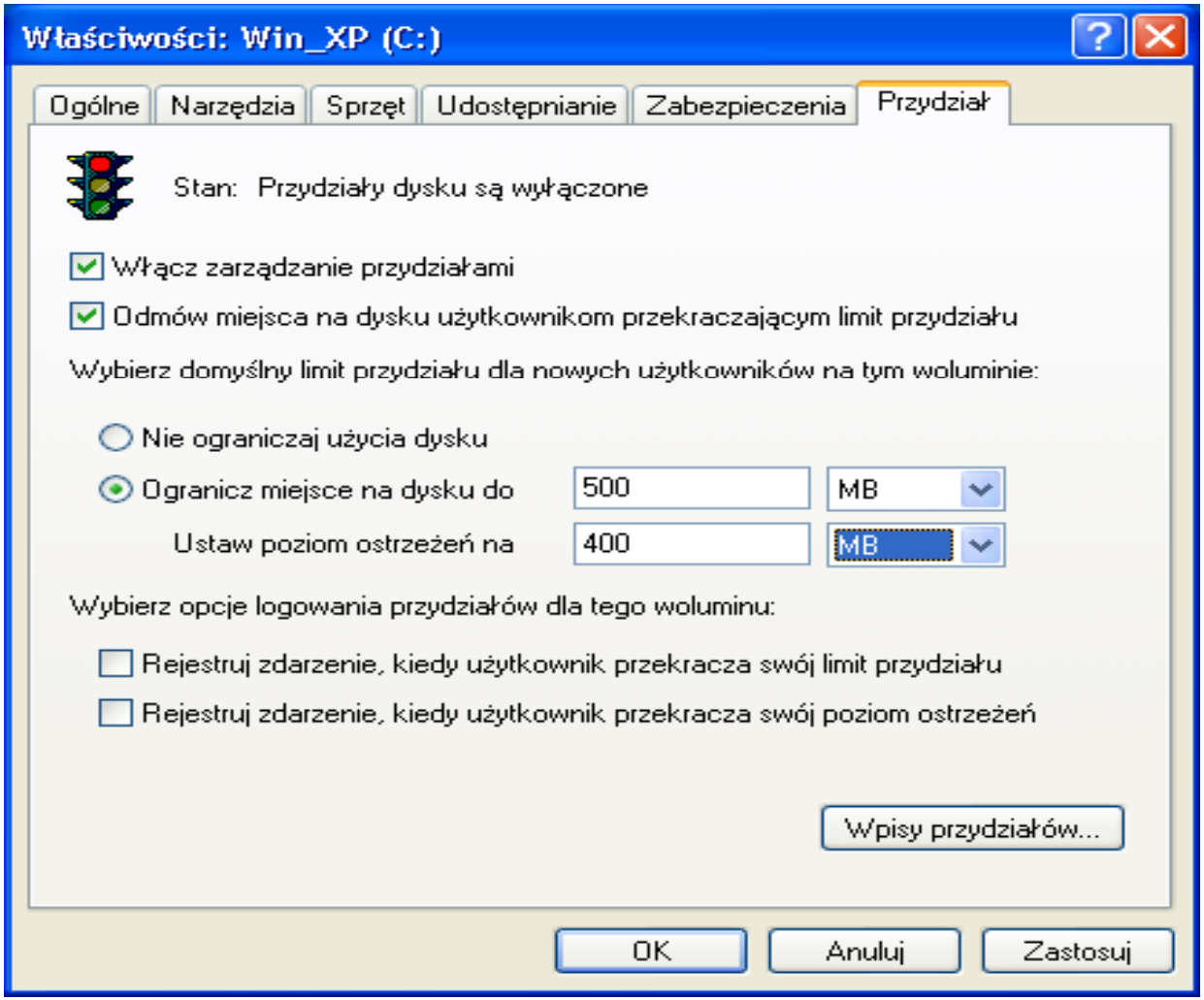
- Przykład dla monitorowania konkretnego katalogu
- Właściwości | Zabezpieczenia | Zaawansowane | Inspekcja



Uprawnienia dotyczące ilości przestrzeni dyskowej (I)

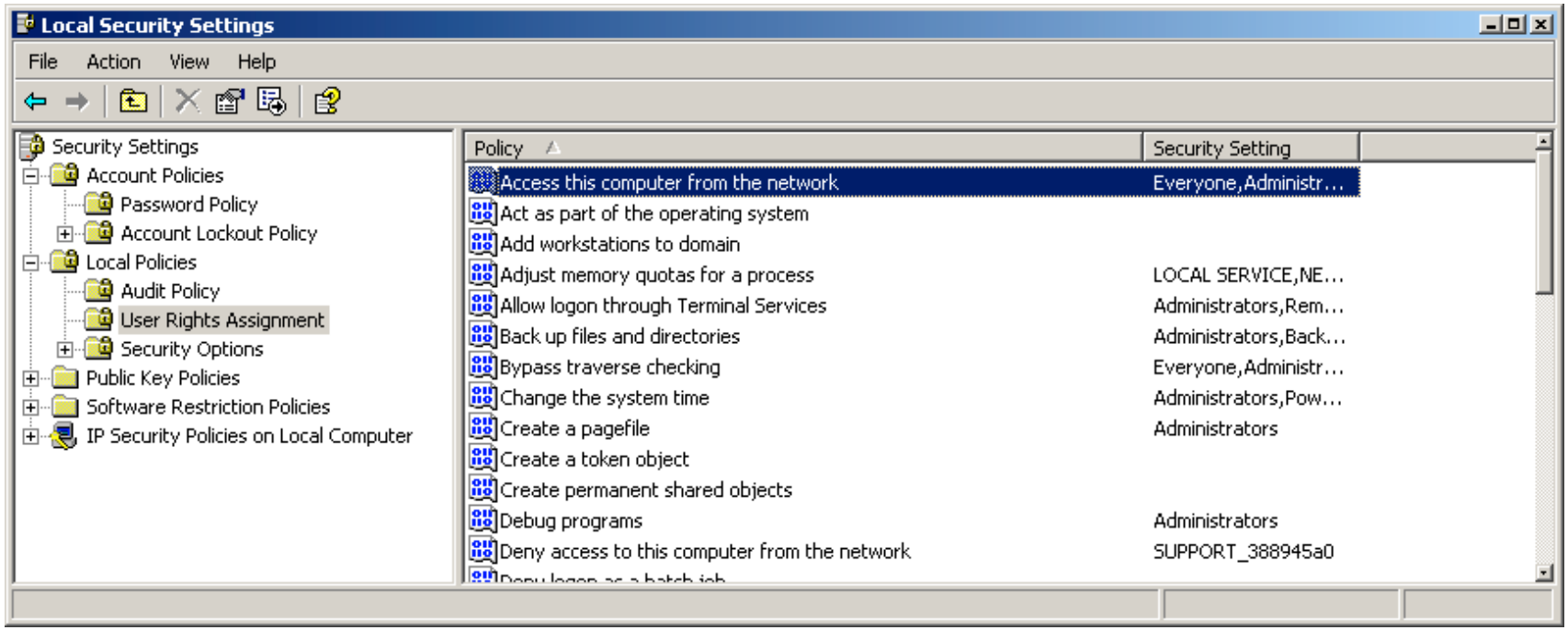
- Ograniczenia są przypisane do woluminu i nie można przypisać ich do folderu
- Ograniczenia są przypisane właścicielom plików i nie można przypisać ich grupom użytkowników
- Członkowie grupy **Administratorzy** nie podlegają ograniczeniom.
- Limity ograniczeń są zapisywane na dysku (NTFS)

Uprawnienia dotyczące ilości przestrzeni dyskowej (II)

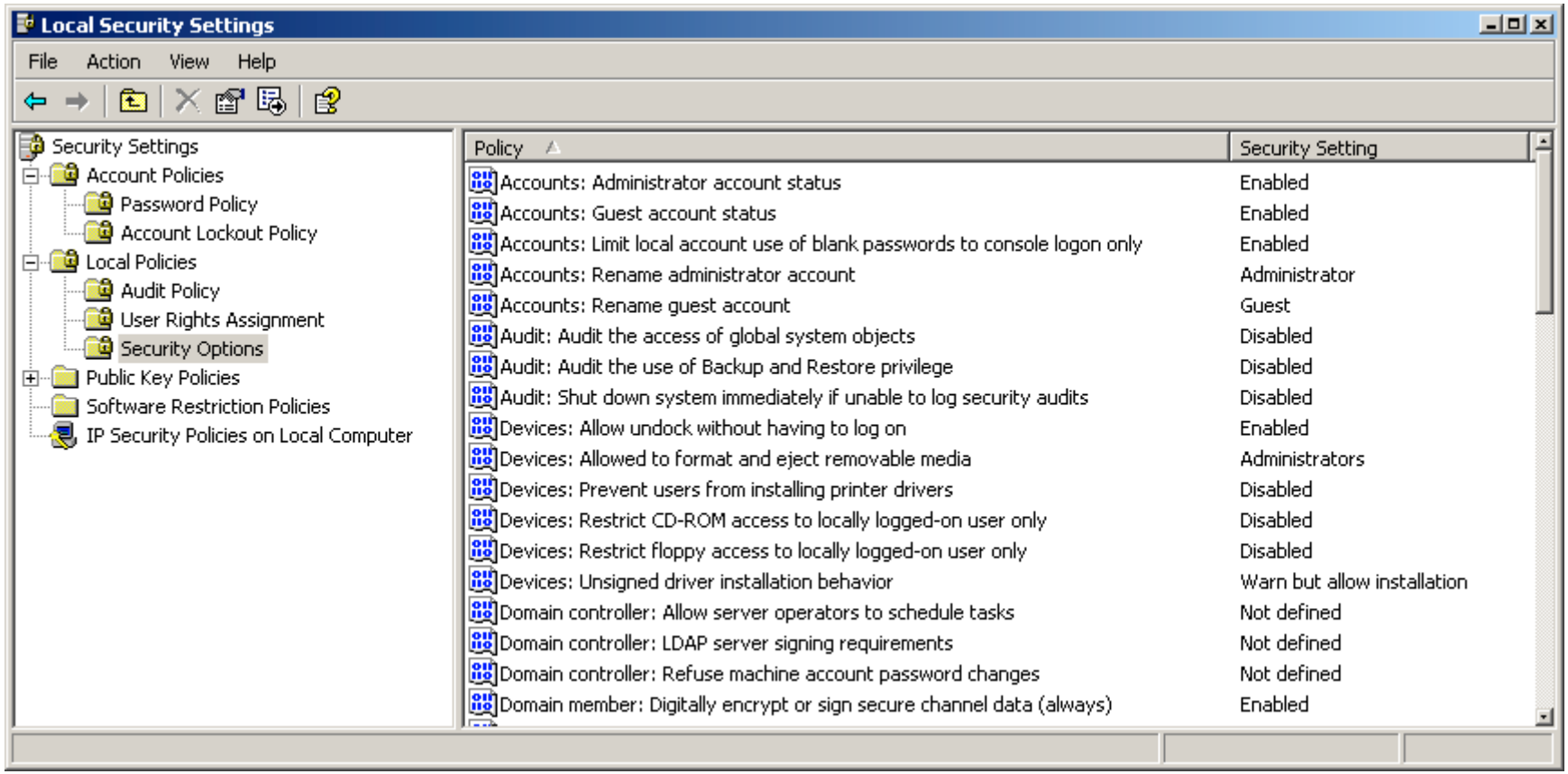


Modyfikacja uprawnień użytkowników

- Możliwe jest ustalanie grup lub użytkowników jako uprawnionych do wykonywania określonych operacji (nie związanych z konkretnymi obiektami)
  - z puli zdefiniowanych operacji systemowych

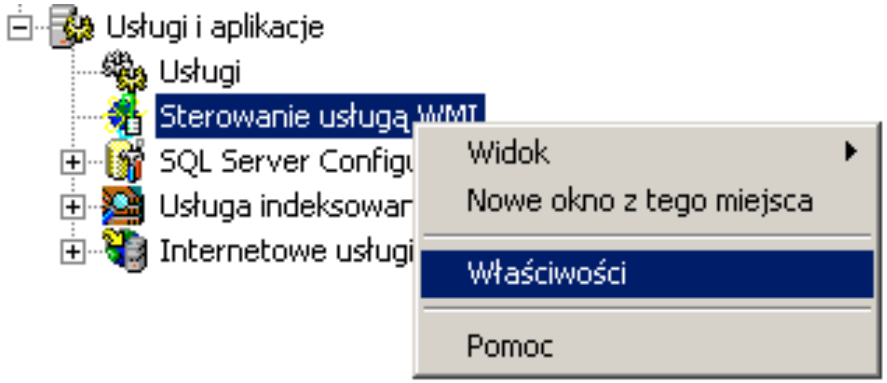


Modyfikacja zabezpieczeń



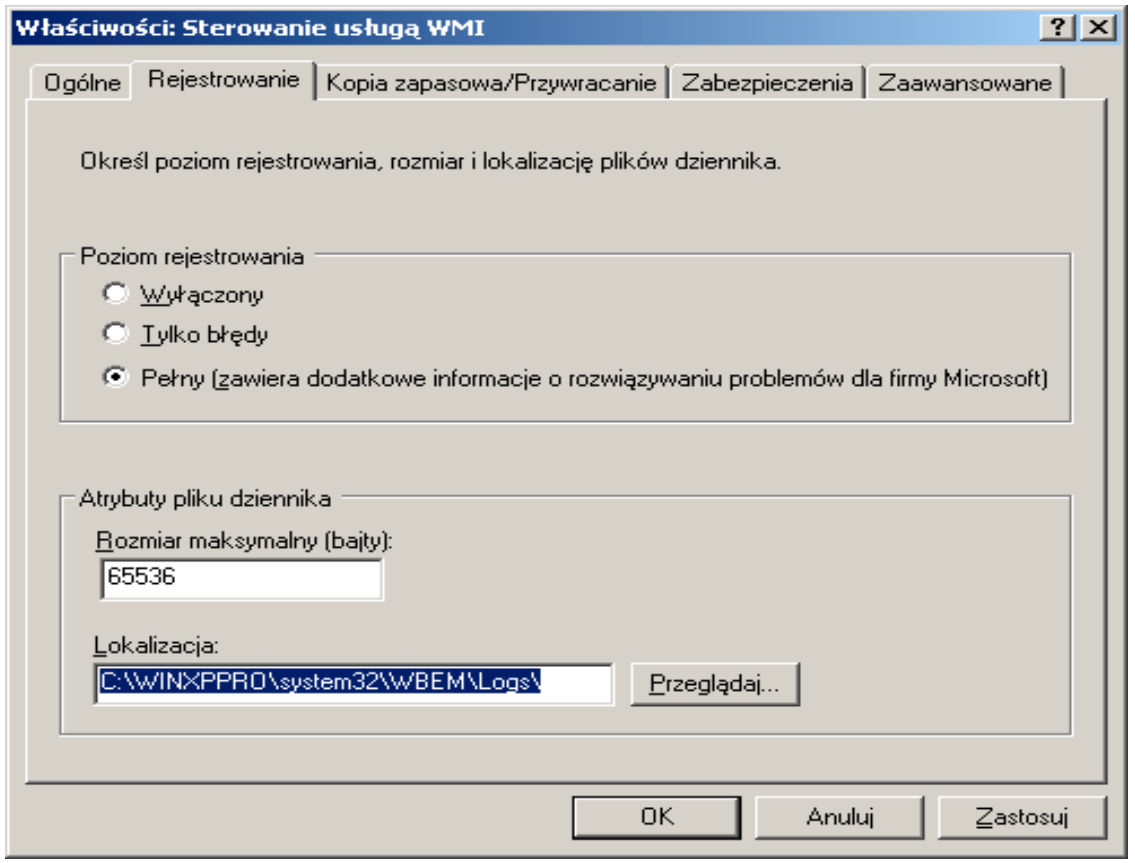
Zabezpieczenia WMI (I)

- Szczególnie istotne przy próbach dostępu zdalnego
- Manager: Dostępny poprzez compmgmt.msc->Usługi i aplikacje -> Sterowanie usługą WMI



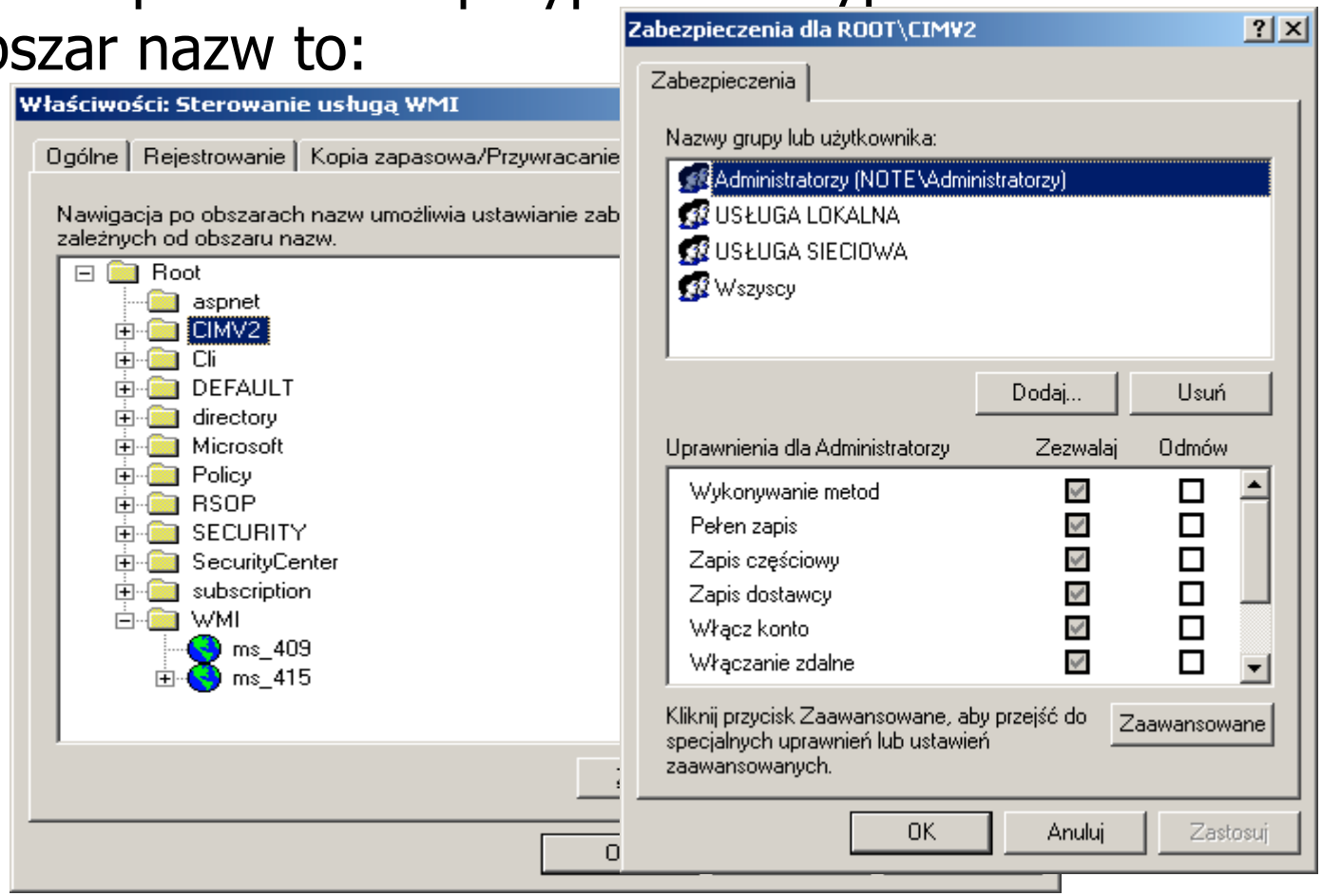
Zabezpieczenia WMI (II)

- Ustawienia logów WMI - domyślna ścieżka:
  C:\WINDOWS_DIR\system32\WBEM\Logs\



Zabezpieczenia WMI (III)

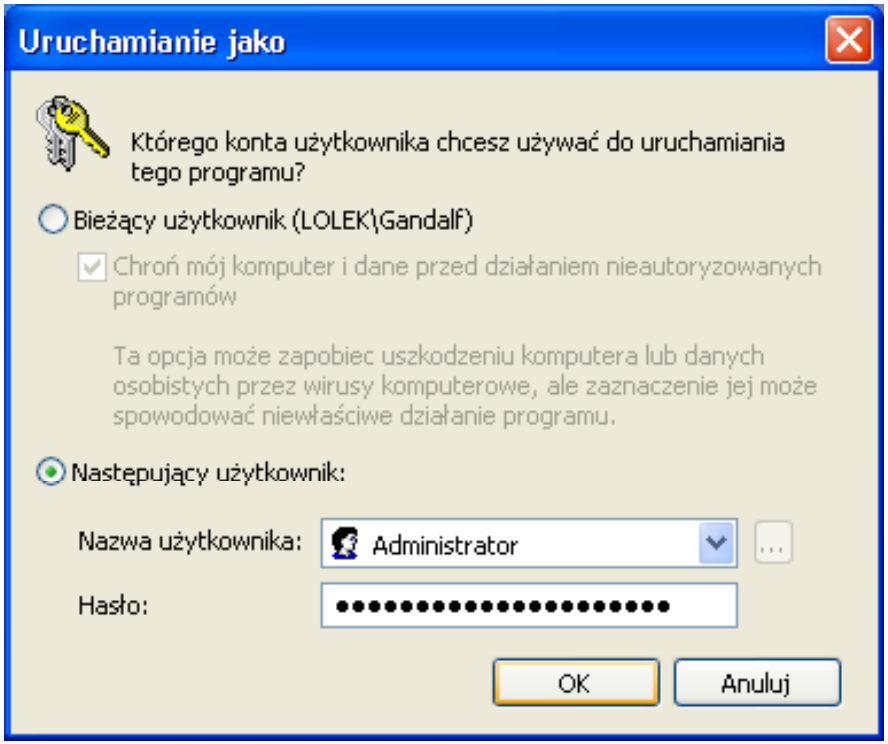
- Ustawienia zabezpieczeń - w przypadku skryptów domyślny obszar nazw to: \root\cimv2



Uruchomienie programu z prawami innego użytkownika (I)

- Opcja umożliwia korzystanie z uprawnień np. administratora w celu wykonania przez dany program operacji wymagających konkretnych praw dostępu,
- Dostępna w menu podręcznym (*Uruchom jako../Run as...*),
- Logowanie na konto innego użytkownika jest każdorazowe - uprawnienia przyznawane są tylko na rzecz wykonania konkretnego programu.

Uruchomienie programu z prawami innego użytkownika (II)

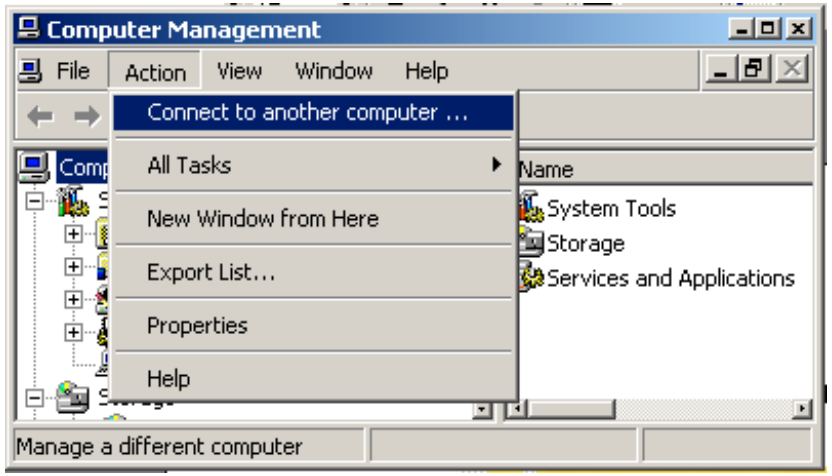


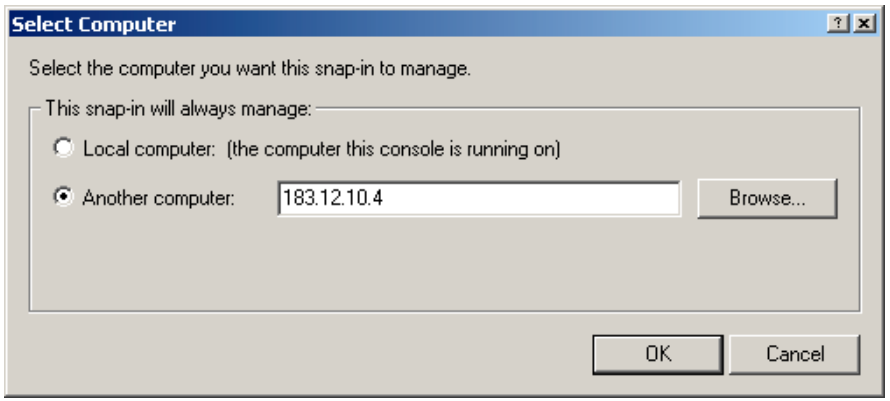
Polecenie runas

- Runas - opcje w wersji polecenie shell'a:
  - /user:[komputer/]użytkownik - ustalenie użytkownika lub użytkownika zdalnego
  - /noprofile - brak ładowania profilu użytkownika (zwiększenie prędkości ładowania aplikacji)
  - /profile - wytypowanie innego profilu użytkownika
  - /netonly - stosowanie praw użytownika tylko przy połączeniach sieciowych (przydatne np. przy zdalnym administrowaniu)
  - /env - uruchomienie z zachowaniem środowiska (np. katalogu bieżącego). Bardzo istotna opcja

Polecenie runas - przykład

- Z linii komend:
  - ***runas /netonly /user:183.12.10.4\Administrator "mmc compmgmt.msc"***
  - Enter the password for *183.12.10.4\Administrator : *******
- Konsola przy połączeniach sieciowych zastosuje uprawnienia zdalnego użytkownika

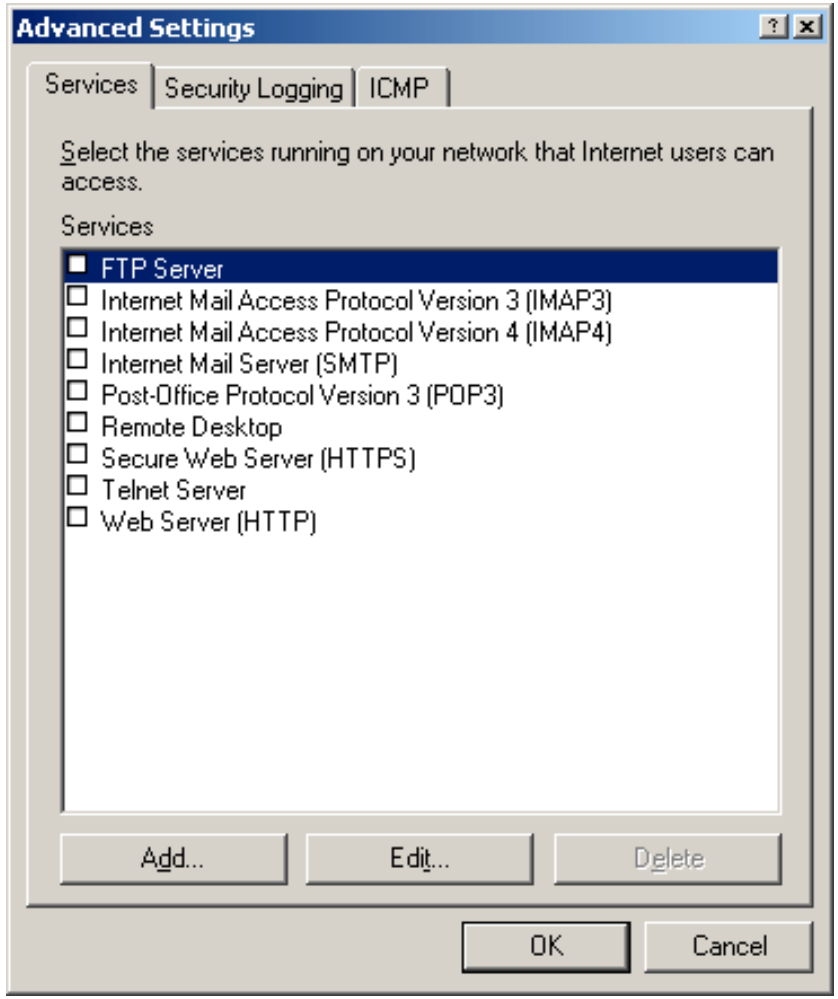




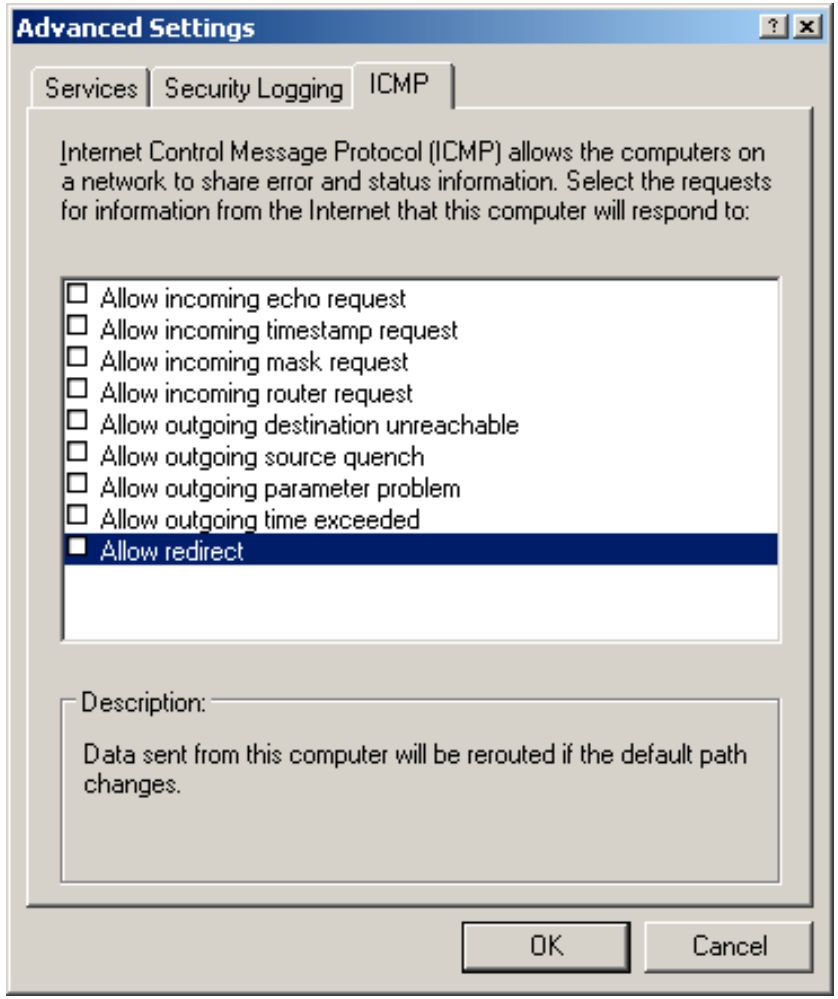
Zabezpieczenia i audyty sieci (I)

- Blokowanie dostępu do niektórych usług warstwy 4
- Blokowanie funkcjonalności ICMP
- Audyt pakietów i połączeń - plik *WindowsDir/pfirewall.log*

Zabezpieczenia i audyty sieci (II)

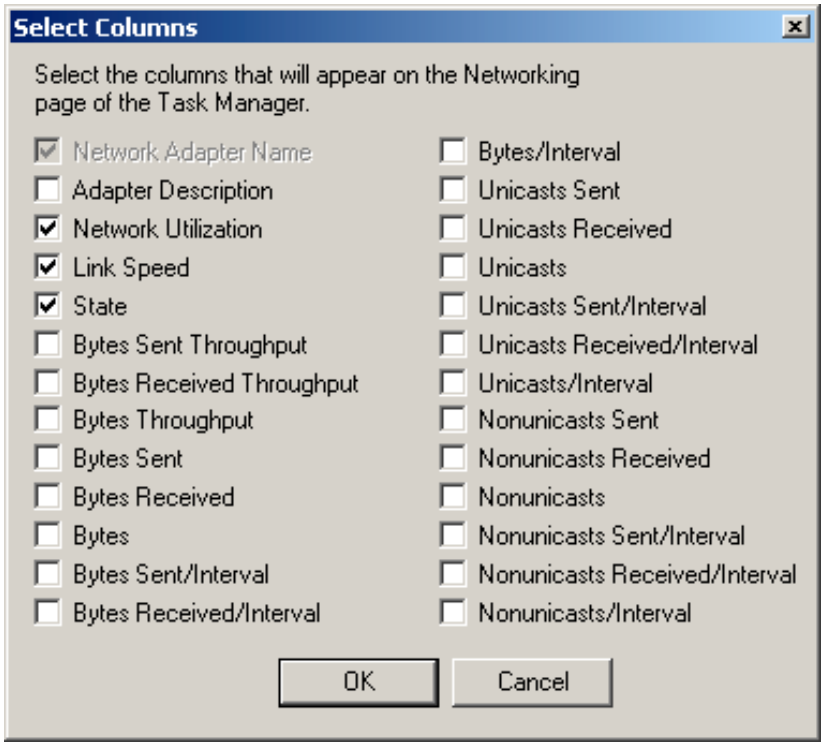


Zabezpieczenia i audyty sieci (III)



Zabezpieczenia i audyty sieci (IV)

- Zaawansowane monitorowanie ruchu w sieci po modyfikacji monitora sieci w Menedżerze zadań Windows



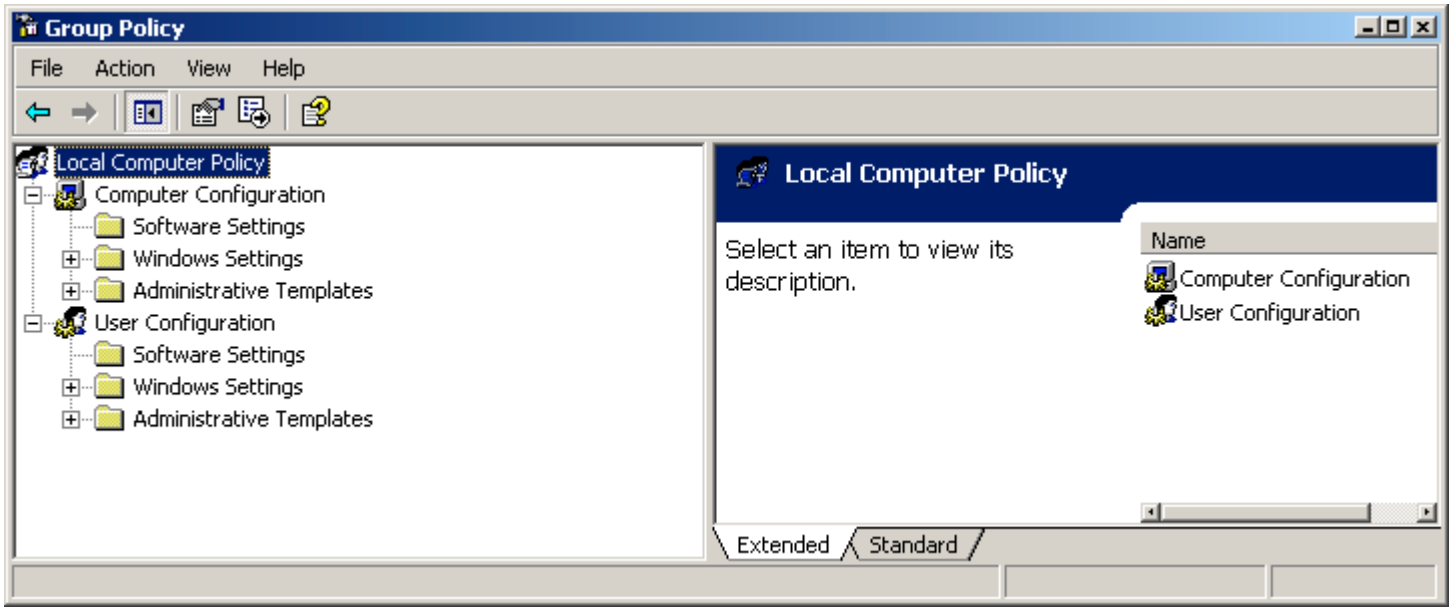
Drukowanie w nietypowych sytuacjach

- Drukowanie z DOS:
  - **net use lpt1: \\dir1\plik**
- Drukowanie na serwerach UNIX:
  - na zdalnej maszynie wymagane jest funkcjonowanie daemona LPD (drukarki wierszowej)
  - **lpr -S serwer -P printer plik**

Zbiorcza konfiguracja systemu i polityka uprawnień

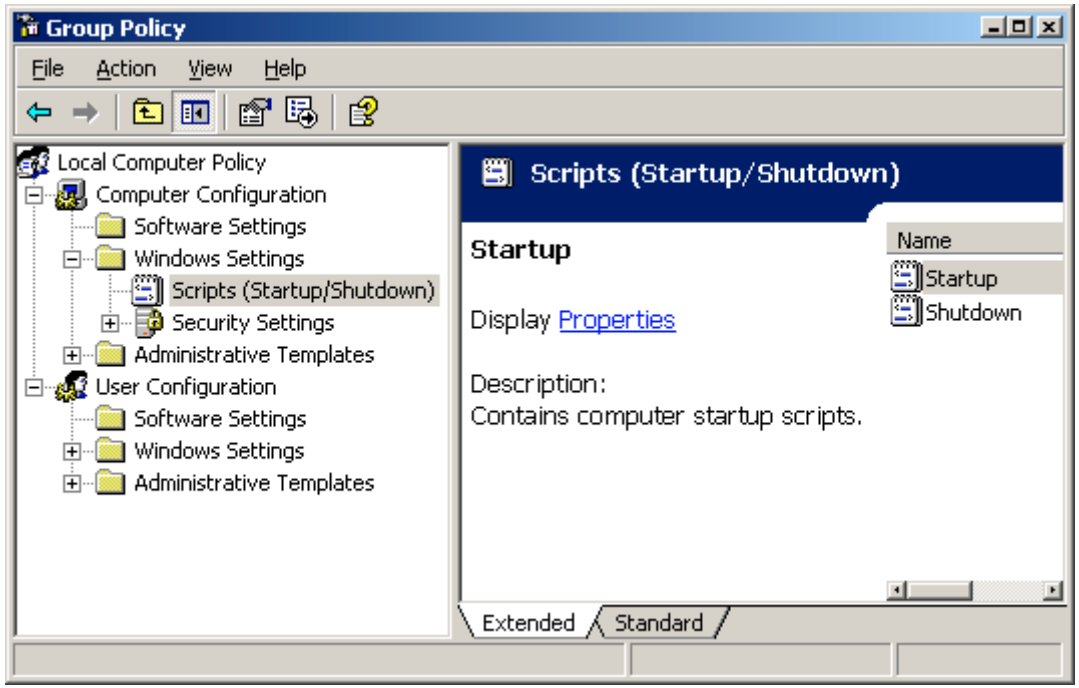
- Konsola gpedit (Group policy)
- Globalny podział ustawień: system i użytkownik - o niemal homogenicznej strukturze
- Ustawienia należy traktować jako uzupełnienie - początkowo wszystkie mają status „nieskonfigurowane” co powoduje stosowanie ustawień standardowych
- Ustawienia systemowe nadrzędne

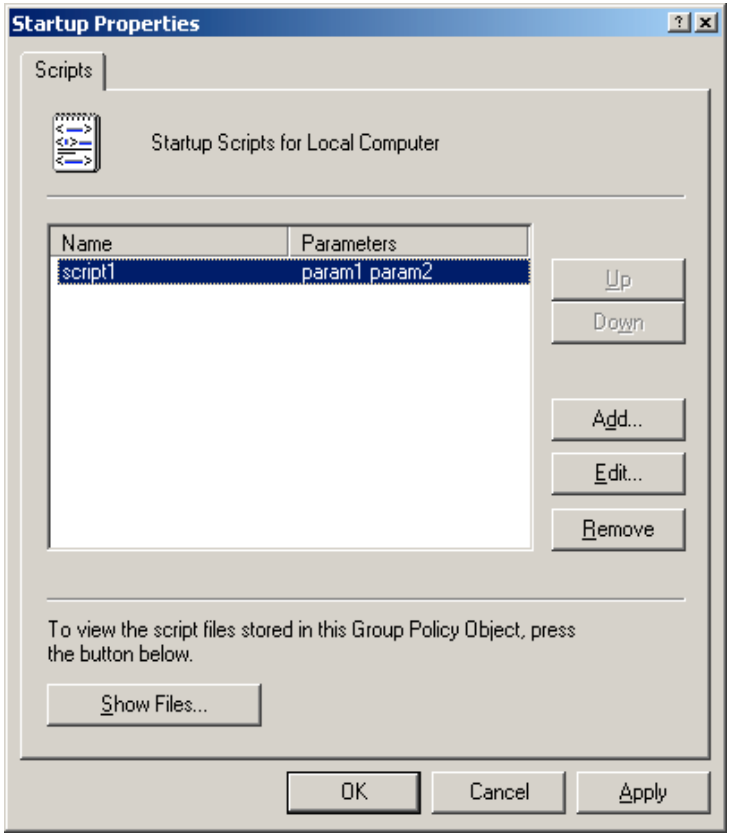
Konsola gpedit



Skrypty startu i zatrzymania systemu

- Możliwe jest zdefiniowanie wielu skryptów wraz z kolejnością uruchomienia.

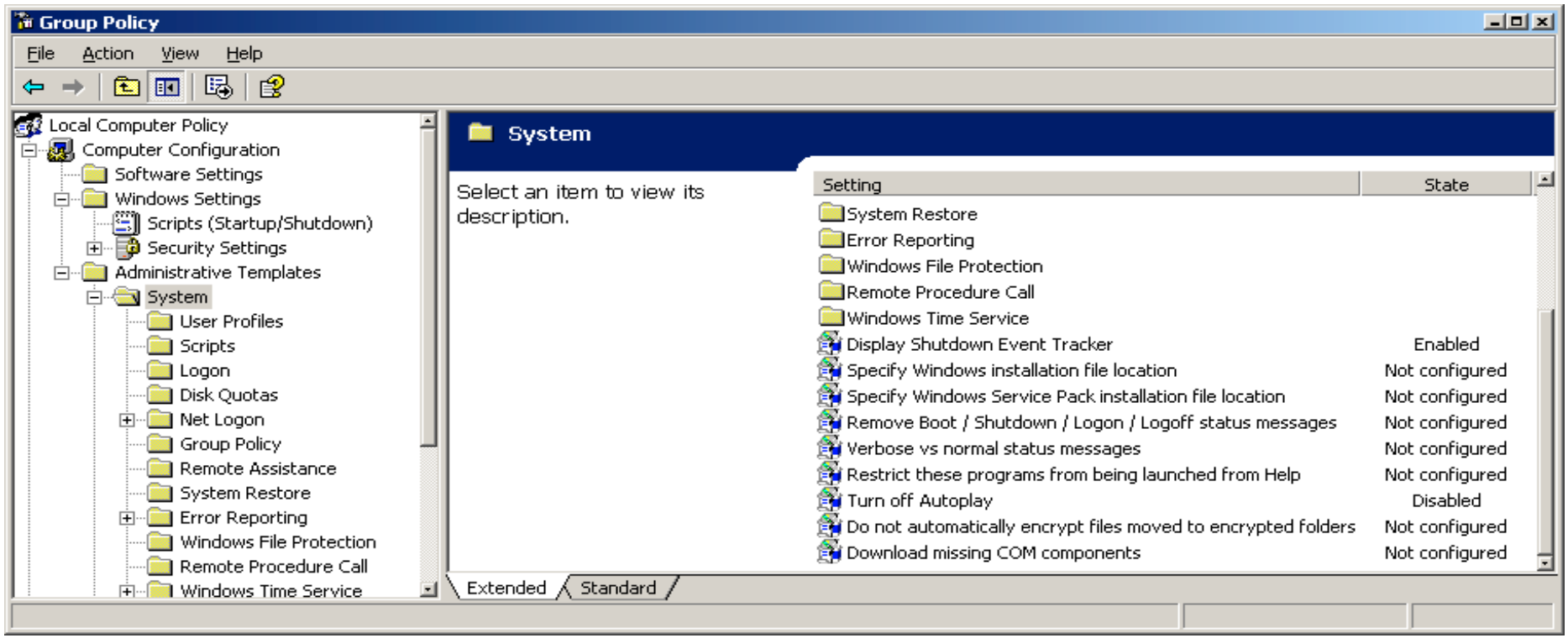




Szablony administrowania

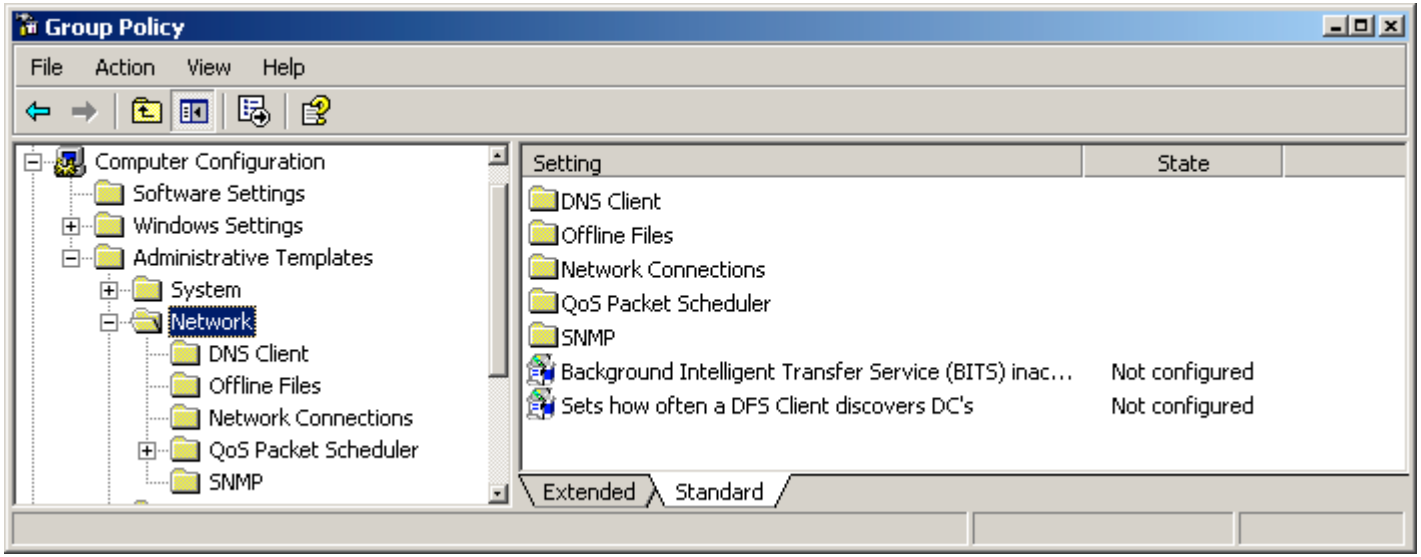
- Istnieje format plików konfiguracji (*.adm), służący do tekstowej prezentacji zbiorów ustawień w konsoli . Faktyczne ustawienia to wartości określonych podkluczy rejestru systemowego.
- Możliwość zarządzania plikami .adm z poziomu konsoli MMC
- Ustawienia definiowane w takich plikach dotyczą głównie: profili użytkowników, trybu automatycznego uruchomienia skryptów, procedur logowania, automatycznego raportowania o błędach aplikacji i odzyskiwania systemu po awarii, synchronizacji czasu systemowego itp

Szablony administrowania - ustawienia systemu



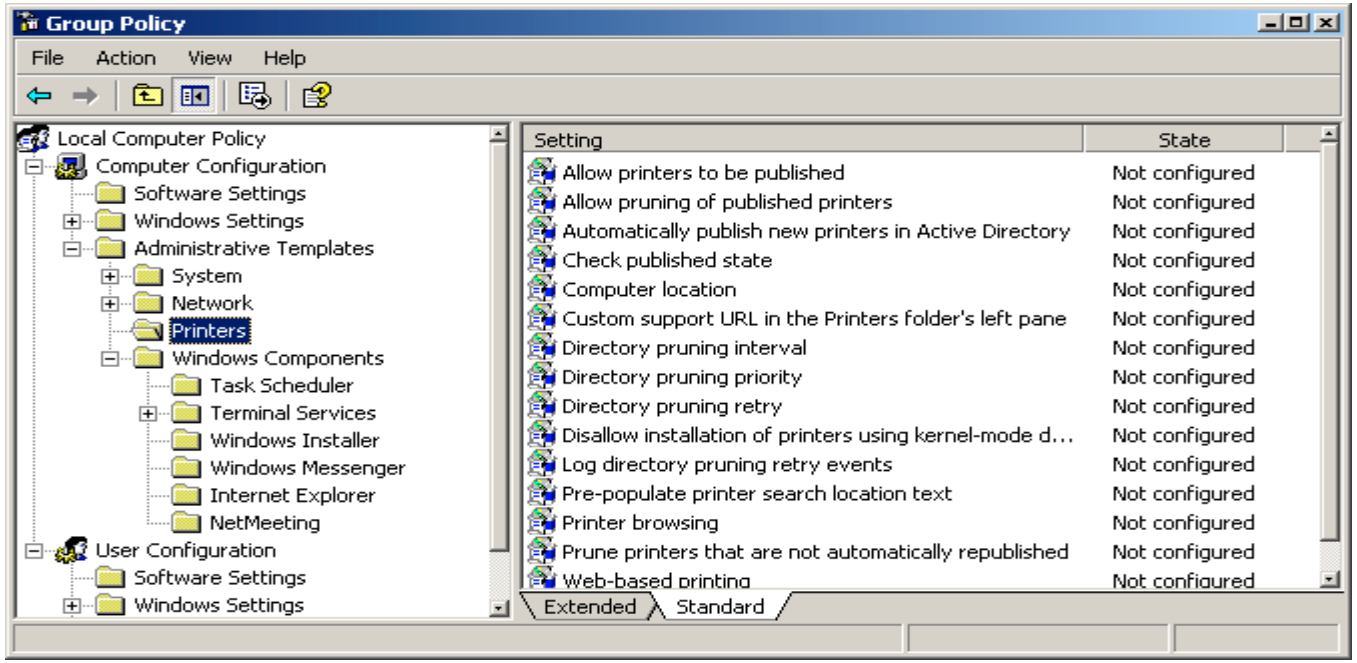
Szablony administrowania - ustawienia sieci

- Zaawansowane ustawienia sieci: klient DNS, zezwolenia dla protokołów, ustawienia pułapek diagnostycznych SNMP, kolejkowanie pakietów, synchronizacja katalogów offline



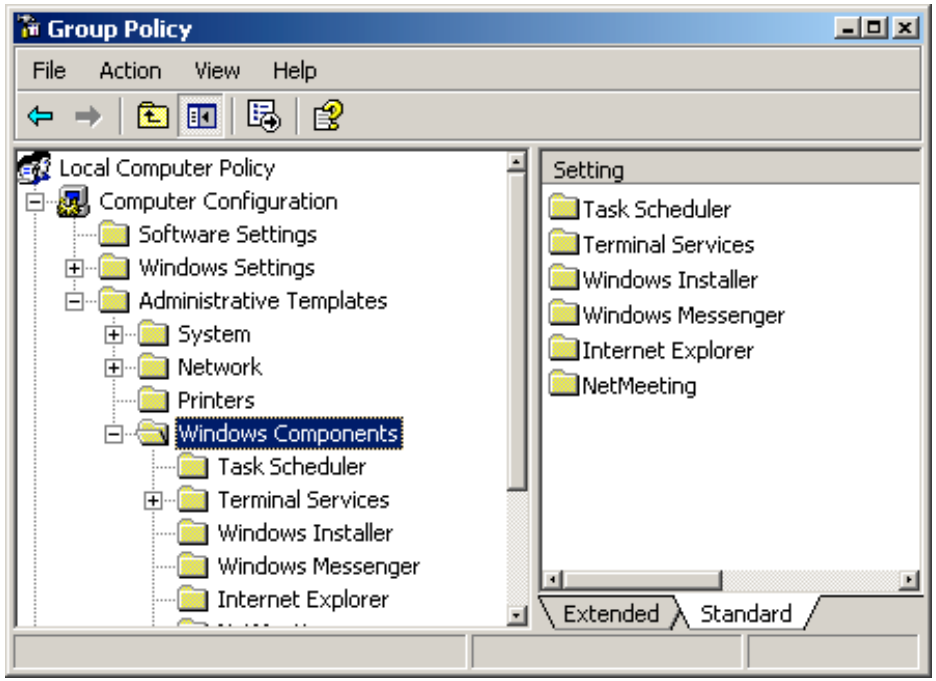
Szablony administrowania - ustawienia wydruku

- Ustawienia uprawnień do upublicznienia serwera wydruku, stref automatycznego wyszukiwania urządzeń drukujących, ustawienia parametrów dynamicznego uaktualniania informacji o serwerach wydruku, ustawienia poszukiwania drukarek w Active Directory.



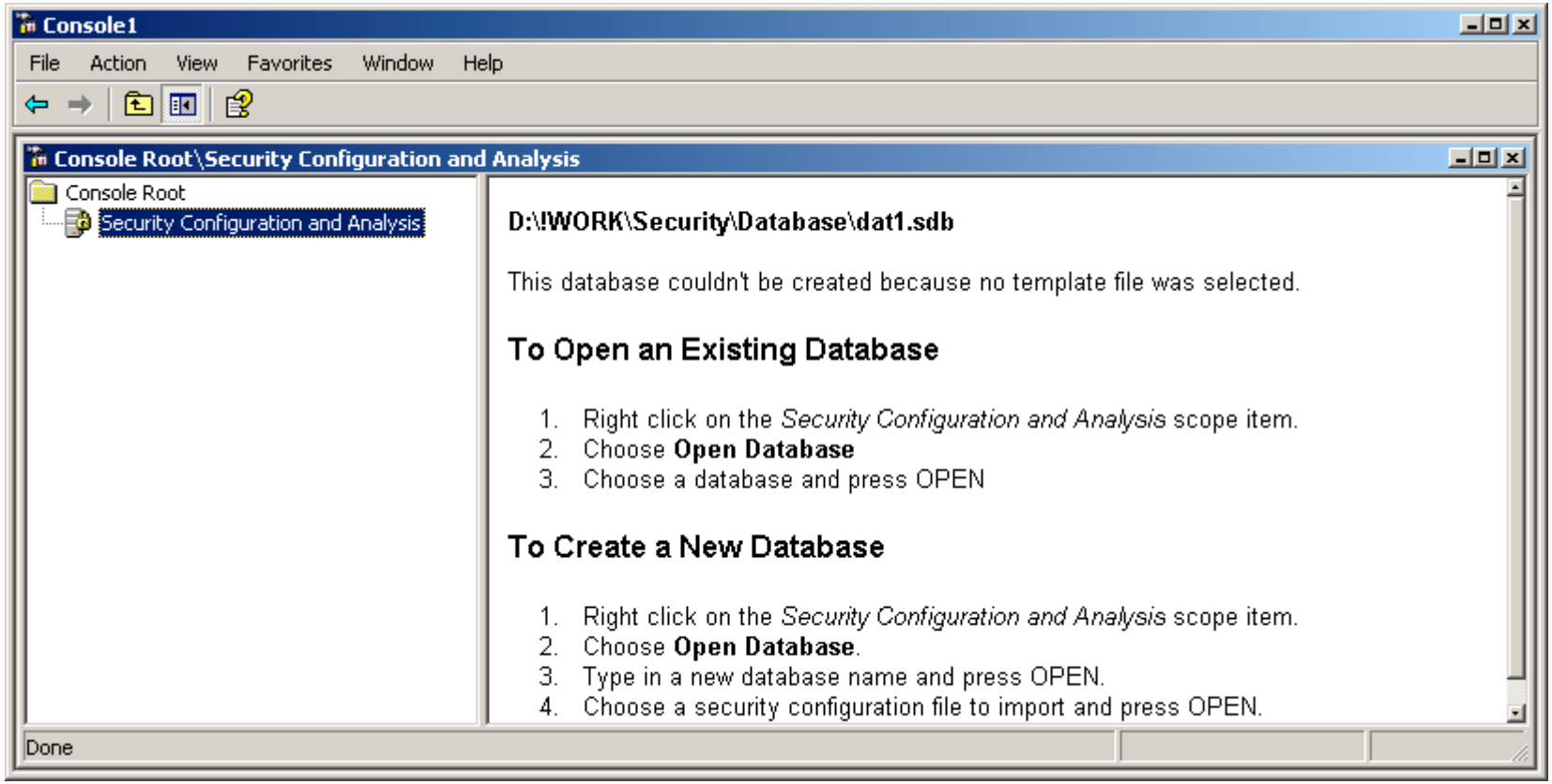
Szablony administrowania - komponenty Windows

- Konfiguracja niektórych usługi systemu występujących pod postacią oddzielnych aplikacji. Standardowo są to: Scheduler, Usługi Terminalowe, Installer Windows, Messenger, Intrernet Explorer.



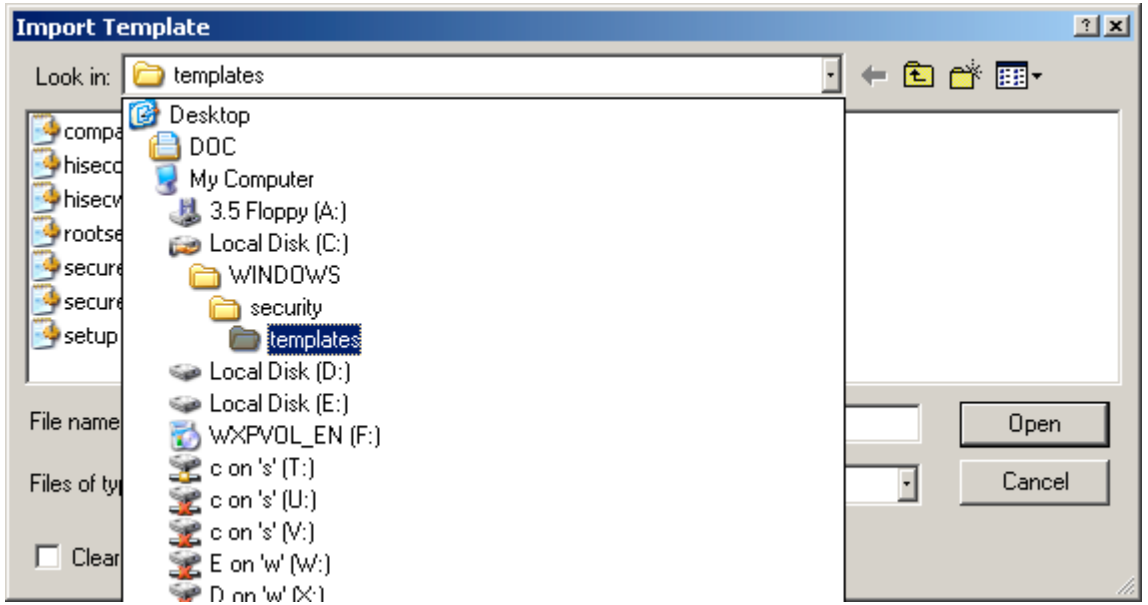
Konfiguracja i analiza zabezpieczeń

- *Security Configuration and Analysis:*



Analiza zabezpieczeń (I)

- Porównanie bieżących zabezpieczeń z szablonem:
  - Tworzenie bazy danych zabezpieczeń na podstawie szablonu (plik *.sdb) (*Open Database* z menu kontekstowego)
  - Analiza porównawcza bazy danych i stanu faktycznego (*Analyse Computer Now...* z menu kontekstowego)



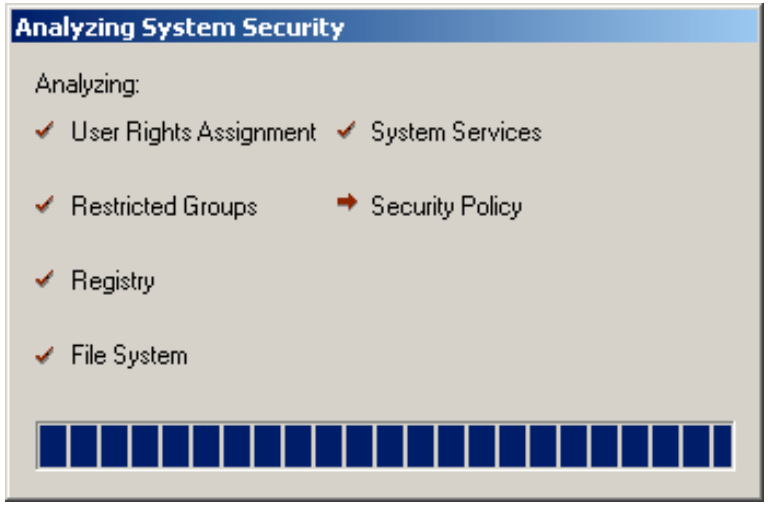
Analiza zabezpieczeń (II)

- Wyniki są składowane katalogu zabezpieczeń, plik Security/Logs/*.log

Fragment pliku:

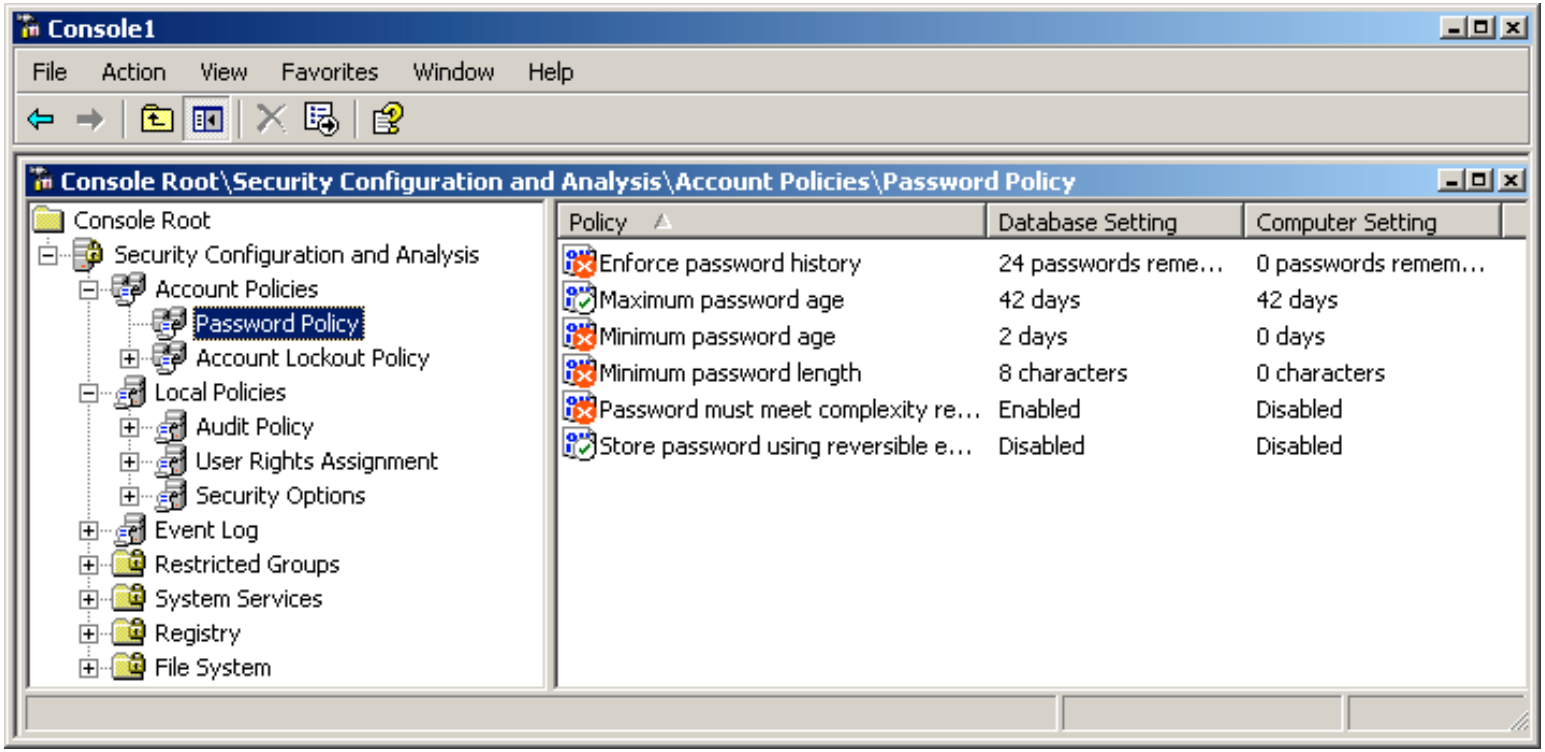
```
Not Configured - RetentionDays.
Mismatch       - MaximumLogSize.
Mismatch       - AuditLogRetentionPeriod.
Not Configured - RetentionDays.
Not Configured - MaximumLogSize.
Not Configured - AuditLogRetentionPeriod.
Not Configured - RetentionDays.
                 Analyze log settings.
Mismatch       - AuditSystemEvents.

Mismatch       - AuditLogonEvents.
```



Analiza zabezpieczeń (III)

- Okno szczegółów zawiera trzy kolumny:
  - Zasady – nazwa analizowanej zasady
  - Ustawienie bazy danych — ustawienie pobrane z bazy .sdb
  - Ustawienie komputera — parametry pobrane z komputera lokalnego



Automatyczna konfiguracja zabezpieczeń

- Zapisanie zabezpieczeń pochodzących z szablonu:
  - Tworzenie bazy danych zabezpieczeń na podstawie szablonu (plik *.sdb) (*Open Database* z menu kontekstowego)
  - Zapis zabezpieczeń w ustawieniach komputera (*Configure Computer Now...* z menu kontekstowego)
- Czynności konfiguracyjne można także wykonać z linii komend:
  - *secedit.exe* [*/analyze* | */configure* | */export*][*/areas*][*/db baza*]

Szablony zabezpieczeń

- W systemach Windows przewidziano możliwość kodowania zestawów zabezpieczeń do postaci szablonów.
- Na podstawie szablonu można wygenerować bazę danych - wykorzystywaną następnie do analizy lub konfiguracji komputera
- Możliwości:
  - Instalacja wybranego standardowego zestawu zabezpieczeń
  - Porównywanie stanu bieżącego z poprzednim w celu wykrycia ingerencji w zabezpieczenia
  - Szybka konfiguracja bliźniaczych systemów

Standardowe szablony zabezpieczeń

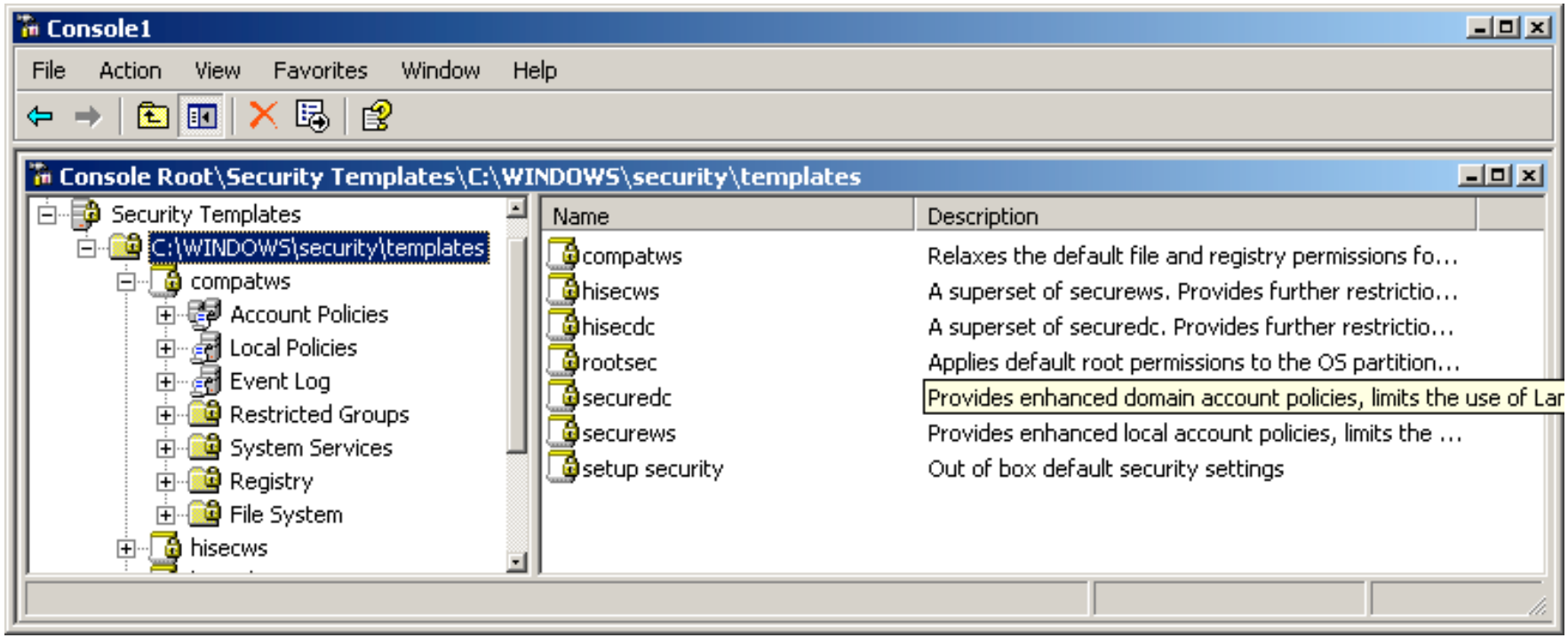
- **Poziom podstawowy** - standardowy poziom zabezpieczeń, jaki jest stosowany podczas instalacji systemu Windows XP.
  Nazwa szablonu setup security.
- **Poziom zgodny** - wyższy poziom zabezpieczeń zapewniający działanie wszystkich aplikacji biurowych.
  Nazwa szablonu compatws.
- **Poziom bezpieczny** - konfiguracja wysokiego poziomu zabezpieczeń. Ryzyko niepoprawnej pracy aplikacji.
  Nazwa szablonu: securews.
- **Poziom wysoki** - maksymalny poziom zabezpieczeń systemu, osiągnięty kosztem poprawnej pracy aplikacji.
  Nazwa szablonu: hisecws.

Zakres obejmowany przez szablony zabezpieczeń

- Zasady kont
  - Zasady haseł
  - Zasady blokady konta
  - Zasady protokołu Kerberos (tylko Windows Server)
- Zasady lokalne
  - Zasady inspekcji
  - Przypisywanie praw użytkownika
  - Opcje zabezpieczeń
- Dziennik zdarzeń: ustawienia dziennika zdarzeń dotyczące aplikacji, systemu i zabezpieczeń
- Grupy z ograniczeniami: członkostwo w grupach istotnych dla zabezpieczeń (tylko Windows Server)
- Usługi systemowe: uruchamianie i uprawnienia dotyczące usług systemowych
- Rejestr: uprawnienia dotyczące kluczy rejestru
- System plików: uprawnienia dotyczące folderów i plików

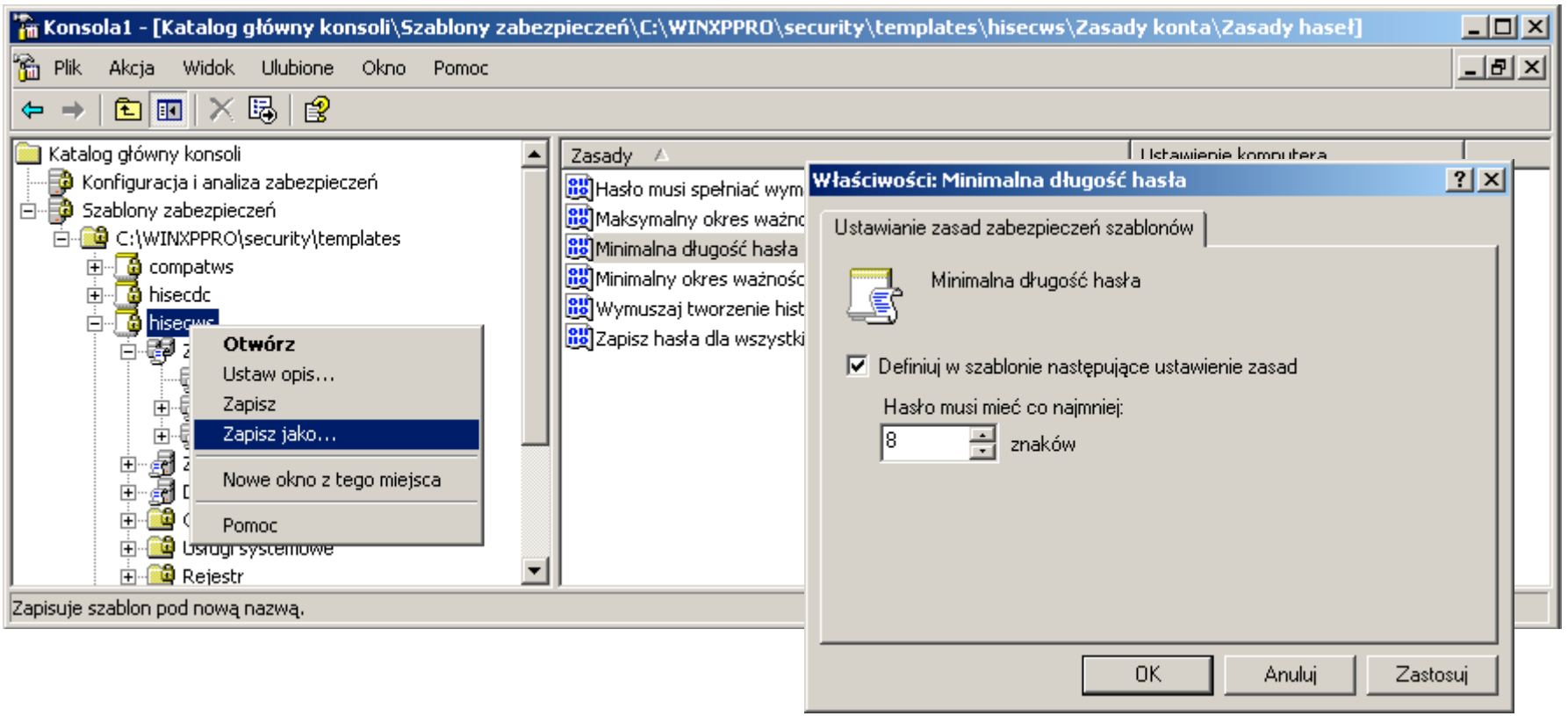
Standardowe szablony zabezpieczeń - przykład (I)

- Końcówki nazw szablonów:
  - ws - przeznaczone dla stacji roboczej
  - dc - przeznaczone dla serwera domeny



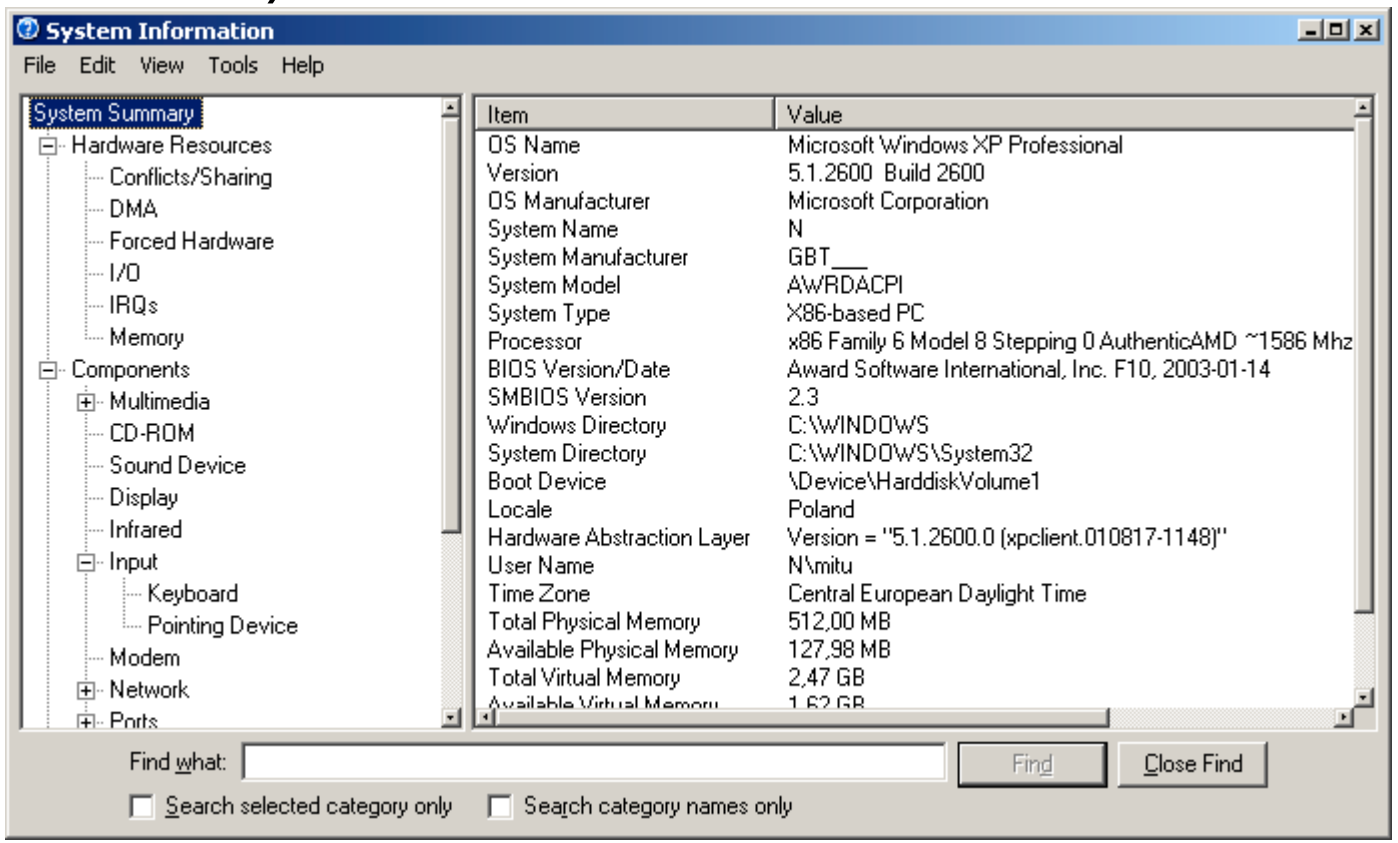
Standardowe szablony zabezpieczeń - przykład (II)

- Możliwość modyfikowania szablonów i powielania ich na poziomie pliku.



Konsola MSINFO

- Aplikacja dostarczająca zestawień informacji o systemie.
  WindowsDir/msinfo32



Narzędzia użytkowe - SubInACL

- Modyfikator uprawnień ACL (Access Control List) dostępny z linii komend, składnik większości pakietów Windows Resource Kit.
- Przykłady:
  - Nadanie praw odczytu do udziału sh1 dla użytkownika skrzynia:
    subinacl.exe" /share sh1 /grant=skrzynia=R
  - Nadanie praw odczytu do pliku „C:\tmp\plik.txt” dla użytkownika skrzynia:
    subinacl.exe" /file C:\tmp\plik.txt /grant=skrzynia=R

Narzędzia użytkowe - SetACL (I)

- Alternatywne do SubInACL narzędzie Open Source służące do konfigurowania zabezpieczeń z poziomu linii komend lub skryptu.
- Narzędzie polecane przez Microsoft, mimo iż całkowicie zewnętrzne.
- Dwa warianty:
  - Narzędzie linii komend
  - Kontrolka OCX - stosowana w skryptach

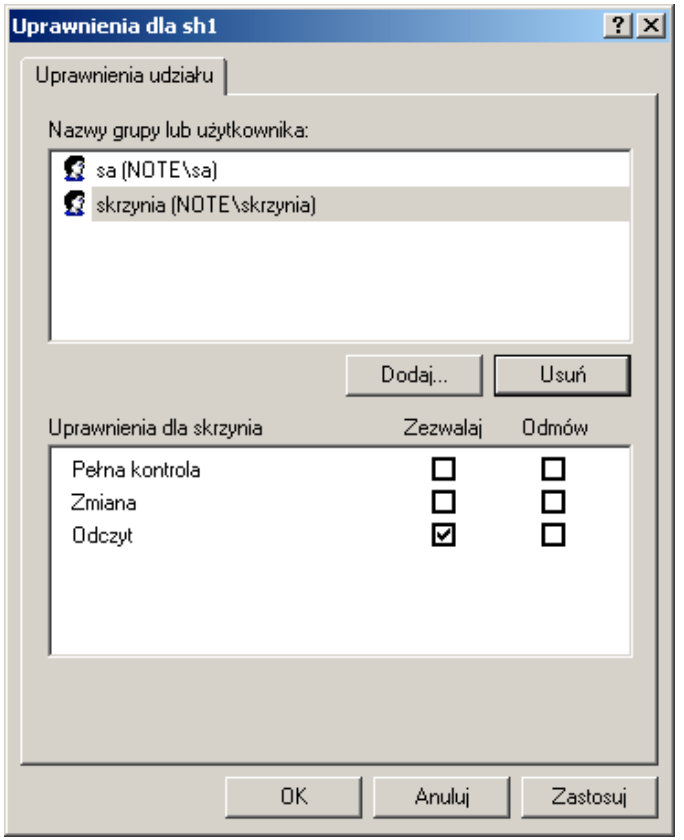
Narzędzia użytkowe - SetACL (II)

- Użycie z linii komend - logika podawania parametrów: typ obiektu (-ot), nazwa obiektu (-on), akcja (-actn), szczegóły akcji
- Przykłady nazw parametrów:
  - Typy obiektów (-ot) : file (plik), reg (rejestr), svr (usługa), prn (drukarka), shr (udział)
  - Akcje: ace (uprawnienia ACE), list (wypisz), setowner (ustal właściciela), trustee (ustal uprawnienia inne niż do obiektów) clear (usuń wpisy ALC)
- Składnia opisów szczegółów akcji:
  - „typ1:id1;typ2:id2 ”
    gdzie typ[..] to przykładowo: n d użytkownika, p dla uprawnień, m dla audytu zmian itp..

Narzędzia użytkowe - SetACL (III)

- Przykład1:
  Przydział uprawnień dla udziału sieciowego sh1 -
  użytkownik „sa”: pełna kontrola,
  użytkownik „skrzynia”: tylko odczyt

  setacl.exe
  -ot shr
  -on sh1
  -actn ace
  -ace "n:skrzynia;p:read"
  -ace "n:sa;p:full"



Narzędzia użytkowe - SetACL (IV)

- Przykład1:
  Przydział takich samych uprawnień
  dla pliku plik.txt

setacl.exe
-ot file
-on "c:\tmp\plik.txt"
-actn ace
-ace "n:skrzynia;p:read”
-ace "n:sa;p:full"

```
C:\tmp>cacls plik.txt
C:\tmp\plik.txt NOTE\skrzynia:(dostęp specjalny:)
                              READ_CONTROL
                              SYNCHRONIZE
                              FILE_GENERIC_READ
                              FILE_READ_DATA
                              FILE_READ_EA
                              FILE_READ_ATTRIBUTES

                BUILTIN\Administratorzy:F
                ZARZĄDZANIE NT\SYSTEM:F
                NOTE\sa:F
                BUILTIN\Użytkownicy:R
                Wszyscy:C
```

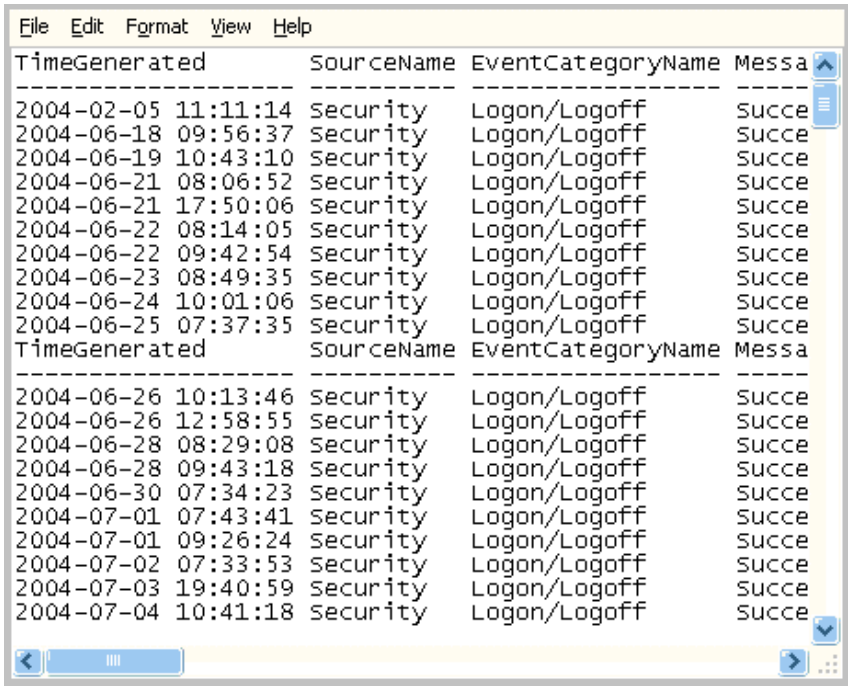
Narzędzia użytkowe – Microsoft Log Parser (I)

- Analizator logów
- Funkcjonalność:
  - Tworzenie raportów z logów
  - Poszukiwanie danych
- Źródła danych
  - Logi (.log)
  - Pliki XML
  - Dziennik zdarzeń
  - Rejestr systemowy
  - System plików (przeszukiwanie nazw itp)
  - Źródła danych

Narzędzia użytkowe – Microsoft Log Parser (II)

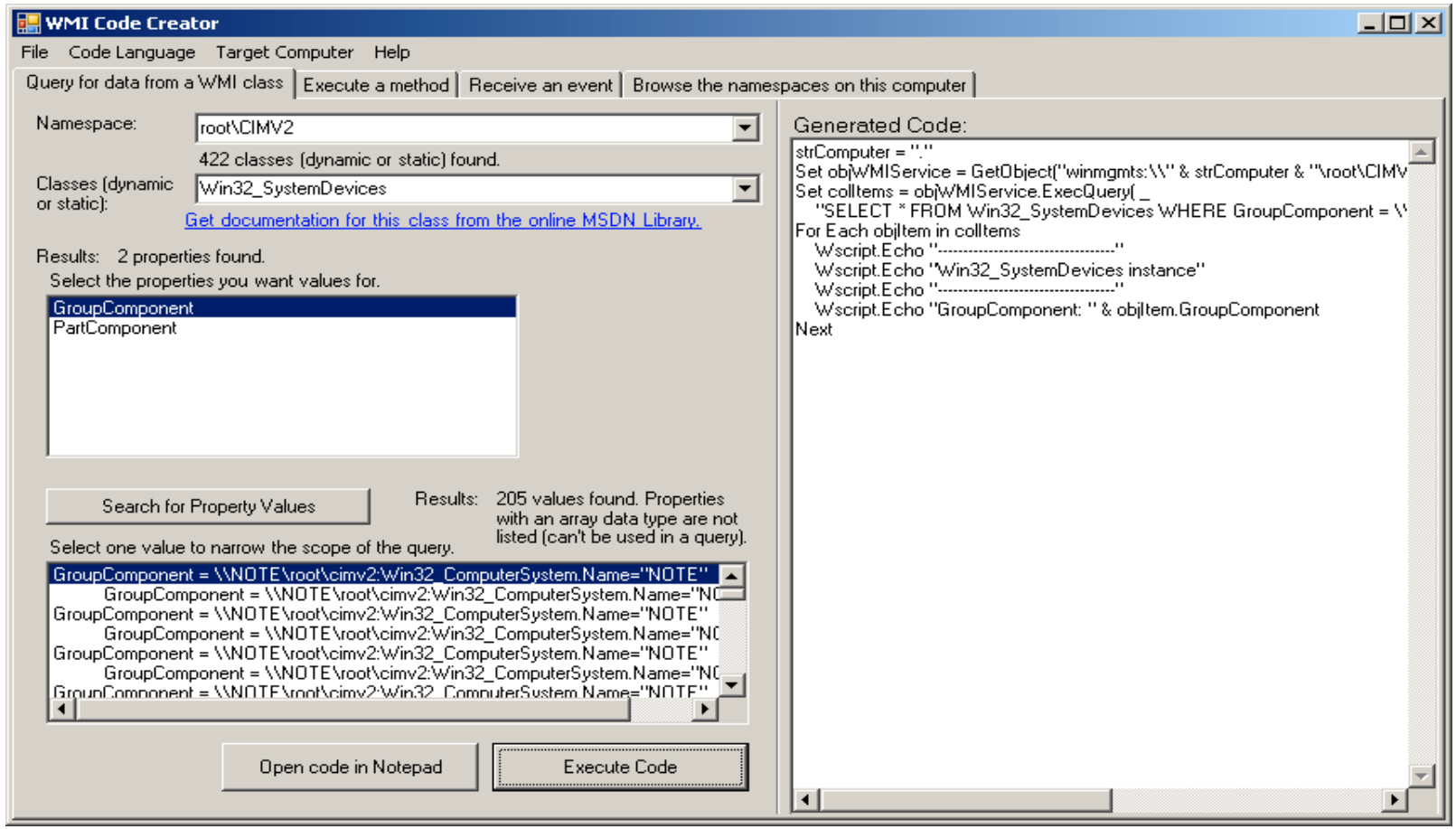
- Przykład użycia:

LogParser "SELECT TimeGenerated, SourceName, EventCategoryName, Message INTO plik.txt FROM Security WHERE EventID = 528 AND SID LIKE '%%'" -resolveSIDs:ON



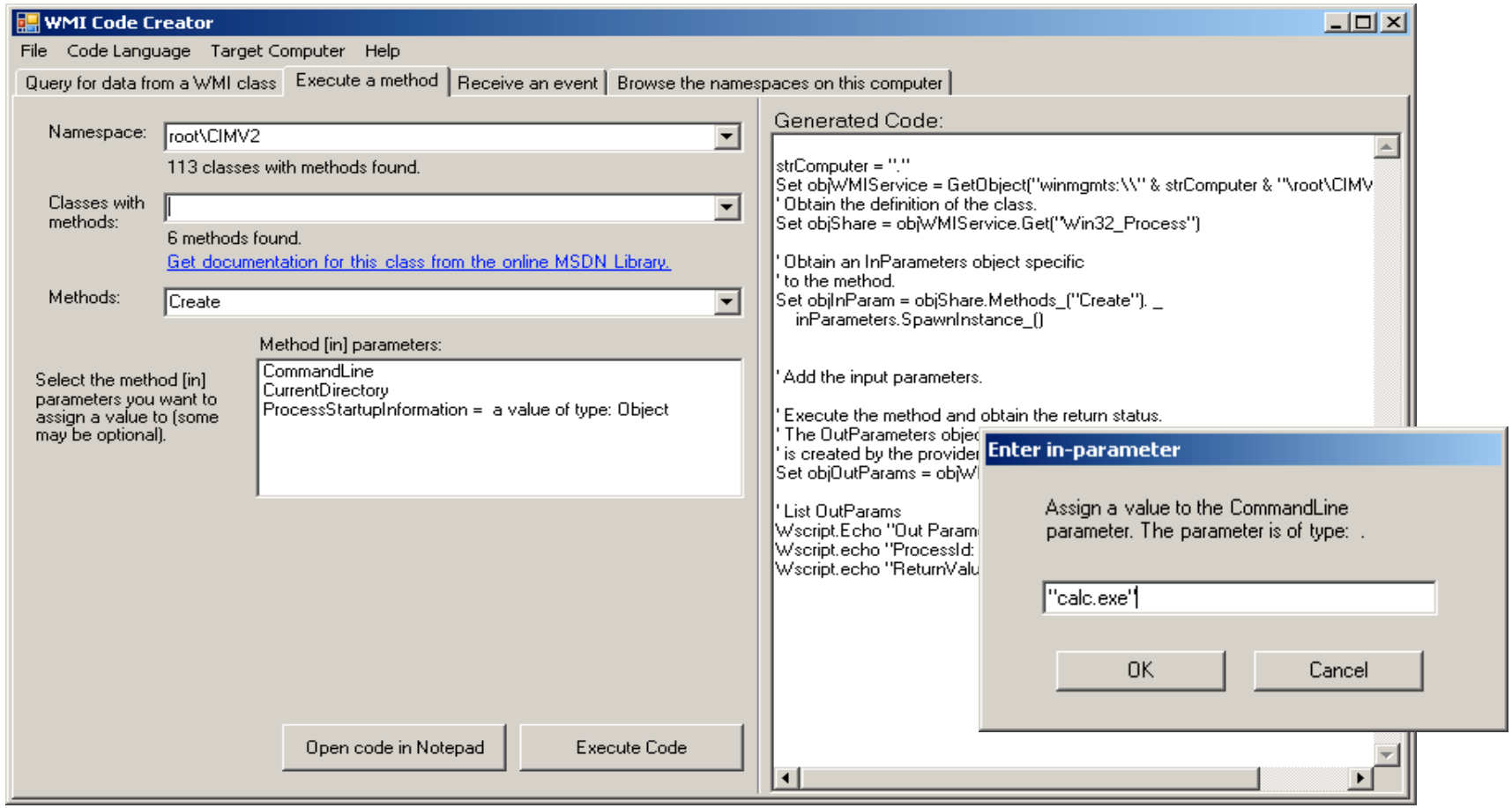
Narzędzia użytkowe – WMI Code Creator (I)

- Generowanie skryptów składających zapytania do WMI



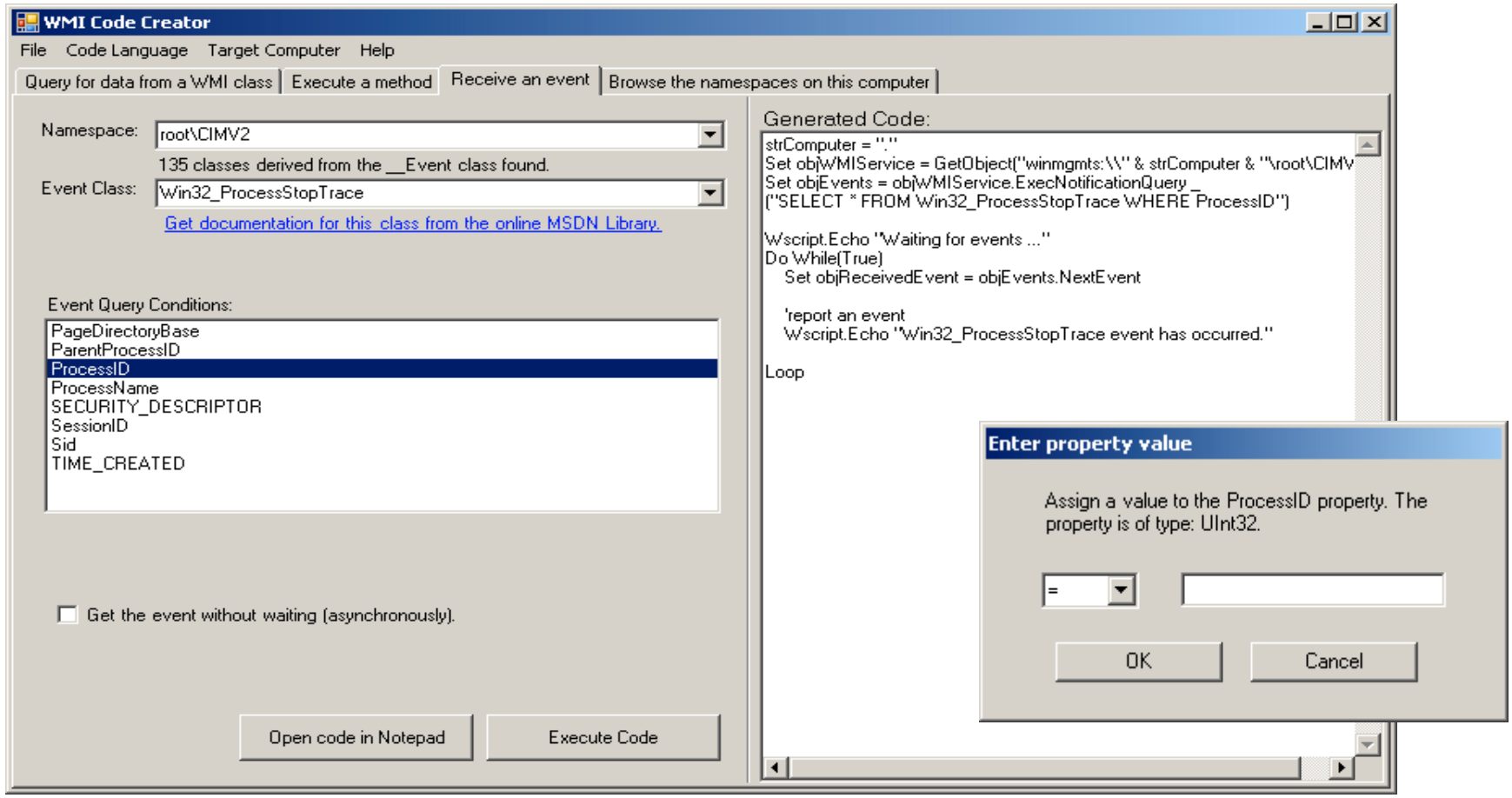
Narzędzia użytkowe – WMI Code Creator (II)

- Generowanie skryptów wywołujących metody klas WMI



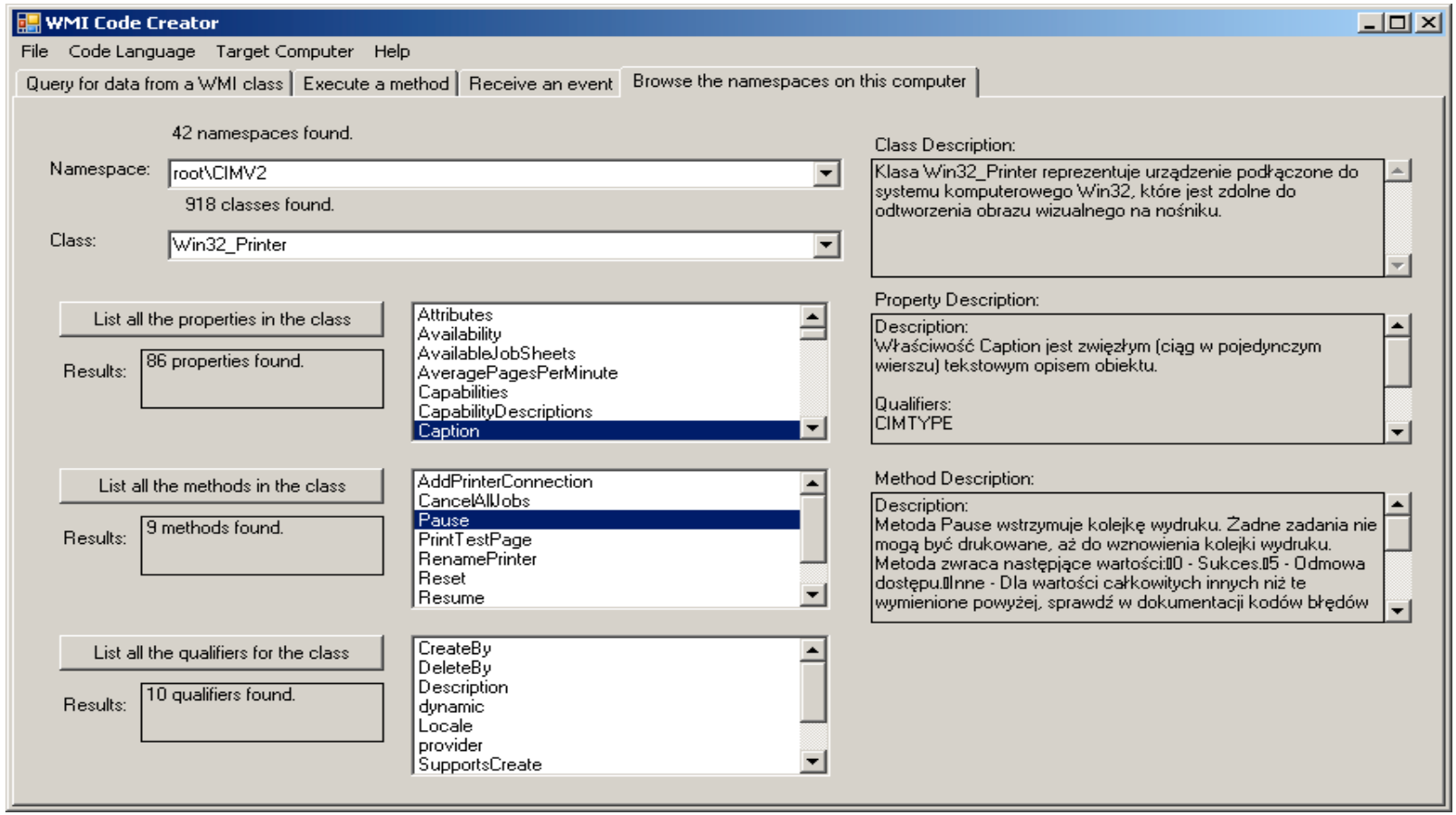
Narzędzia użytkowe – WMI Code Creator (III)

- Generowanie skryptów śledzących zdarzenia WMI



Narzędzia użytkowe – WMI Code Creator (IV)

- Przeglądarka przestrzeni nazw WMI



Komponenty (I)

- COM: Component Object Model Technologies - technologia umożliwia wzajemną komunikację i współdzielenie kodu przez różne niezależne od siebie aplikacje w systemie
- W modelu można wyróżnić trzy grupy: COM+, Distributed COM (DCOM), ActiveX Controls
- Prosty przykład zastosowania: formatka IE osadzona w aplikacji, import danych zapisanych w formacie specyficznym dla innej aplikacji

Komponenty (II)

- COM+ - począwszy od Win2000, zintegrowanie z technologią COM usług dostarczanych przez Microsoft Transaction Server (MTS)
- Rozbudowana funkcjonalność:
  - przechowywanie i udostępnianie zasobów współdzielonych
  - publikowanie zdarzeń systemowych dla aplikacji
  - transakcje rozproszone
  - komunikacyjne pomosty sieciowe wraz z zarządzaniem połączeniami

Zarządzanie komponentami

- Komponenty systemu występują w głównie w formie bibliotek funkcji (.dll) lub plików formatek ActiveX (.ocx)
- Komponenty są rejestrowane w systemie. Po rejestracji komponent jest dostępny dla aplikacji systemowej.
- Rejestracji komponentu dokonuje się także z linii komend, przy użyciu narzędzia *system32/regsvr32*
- Rejestrowany komponent jest automatycznie kopiowany do podkatalogu systemowego (przeważnie */system32*)

Rejestrowanie komponentu

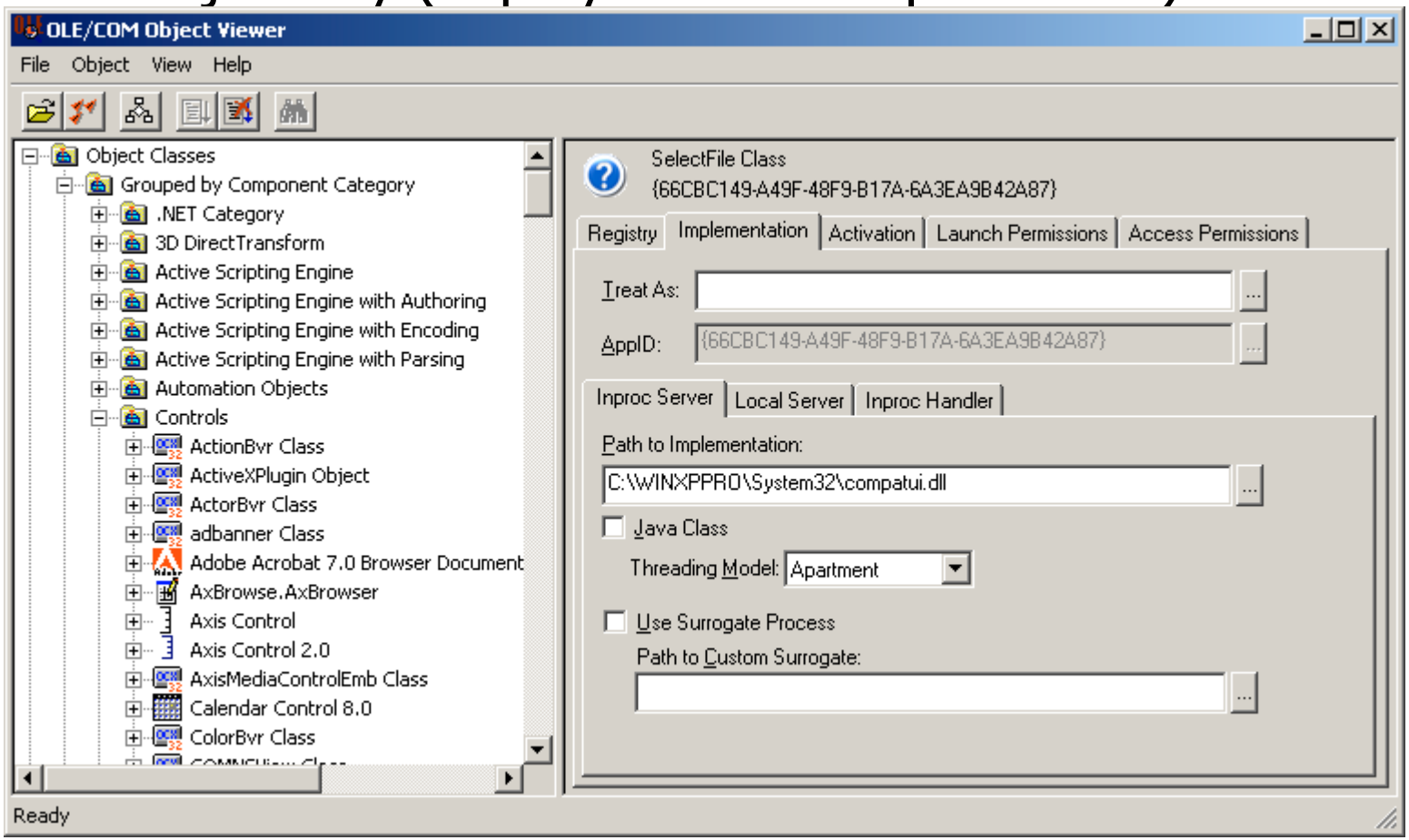
- Przykład wywołania:
  - *regsvr32 formatka.ocx*

Kontrola komponentów Windows (I)

- Narzędzie: Oleview.exe, dostępne w pakietach Resource Kit lub indywidualnie (do pobrania z MSDN)
  - Przeglądanie kolekcji klas - na poziomie kolekcji COM, COM+ lub na poziomie plików (biblioteki)
  - Przeglądanie funkcjonalności klasy (zasoby, metody, parametry wywołania)
  - Sterowanie uprawnieniami (z rozbiciem na pojedynczych użytkowników)

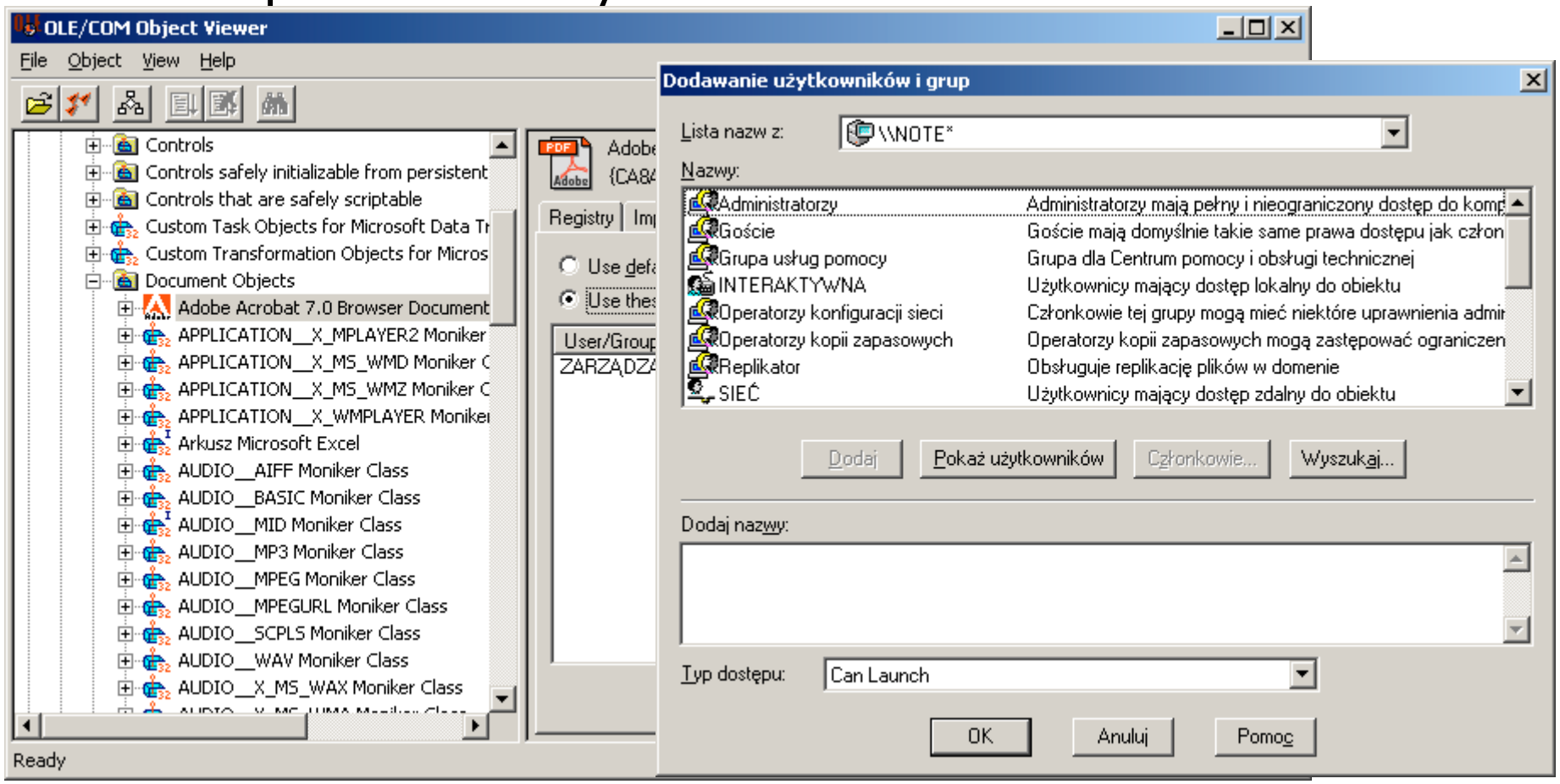
Kontrola komponentów Windows (II)

- Implementacja klasy (w przykładzie: na plikach dll)



Kontrola komponentów Windows (III)

- Zabezpieczenia klasy



Start systemu Windows

- Wybór systemu - dokonany na podstawie ścieżek ARC (*Advanced RISC Computing*) zapisanych w *boot.ini:*
  - SCSI (x) — numer kontrolera (liczony od 0) SCSI, do którego podłączony jest dysk systemowy
  - Multi (x) — numer kontrolera (liczony od 0) IDE, do którego podłączony jest dysk systemowy
  - Disk (y) — numer dysku systemowego SCSI (liczony od 0)
  - Rdisk (y) — numer dysku systemowego IDE (liczony od 0)
  - Partitions (z) — numer partycji systemowej (liczony od 1)

Start systemu - zestawy kontrolne

- Kilka zestawów przechowujących konfiguracje do uruchomienia systemu (przykład dla win 2000/XP):
  - **Current -** przechowuje dane o bieżącej konfiguracji komputera. Dodanie nowego urządzenia lub zainstalowanie nowego programu modyfikuje ten właśnie zestaw.
  - **Default -** przechowuje dane do wykorzystania podczas następnego normalnego uruchomienia systemu.
  - **Failed -** zawiera ostatnią znaną konfigurację po której system był uruchomiony z wykorzystaniem opcji Ostatnia dobra konfiguracja.
  - **LastKnownGood -** zawiera kopię zestawu kontrolnego, który był wykorzystany podczas ostatniego udanego uruchomienia systemu.
- **HKEY_LOCAL_MACHINE\SYSTEM\Select** - informacja, który zestaw jest obecnie używany

Specjalne tryby uruchamiania systemu (Win2000/XP) - I

- **Tryb awaryjny** — system zostanie uruchomiony z minimalną liczbą sterowników (np. nie zostaną zainstalowane sterowniki napędu CD-ROM). Wszystkie uruchomione w tym trybie urządzenia korzystać będą z podstawowej wersji sterowników. Jeżeli pomimo wybrania tej opcji system nadal się nie uruchamia, świadczy to o bardzo poważnych błędach urządzeń niezbędnych do jego działania (dysku twardego, procesora, pamięci lub płyty głównej).
- **Tryb awaryjny z obsługą sieci** — od poprzedniego trybu różni się tym, że zostanie uruchomiona obsługa sieci (sterowniki karty sieciowej i najważniejsze usługi sieciowe).
- **Tryb awaryjny z wierszem poleceń** — system zostaje uruchomiony w trybie awaryjnym, ale nie zostaje uruchomiony *graficzny interfejs użytkownika*. Ten sposób uruchamiania komputera może pomóc przy rozwiązywaniu poważnych problemów związanych z obsługą karty grafiki.

Specjalne tryby uruchamiania systemu (Win2000/XP) - II

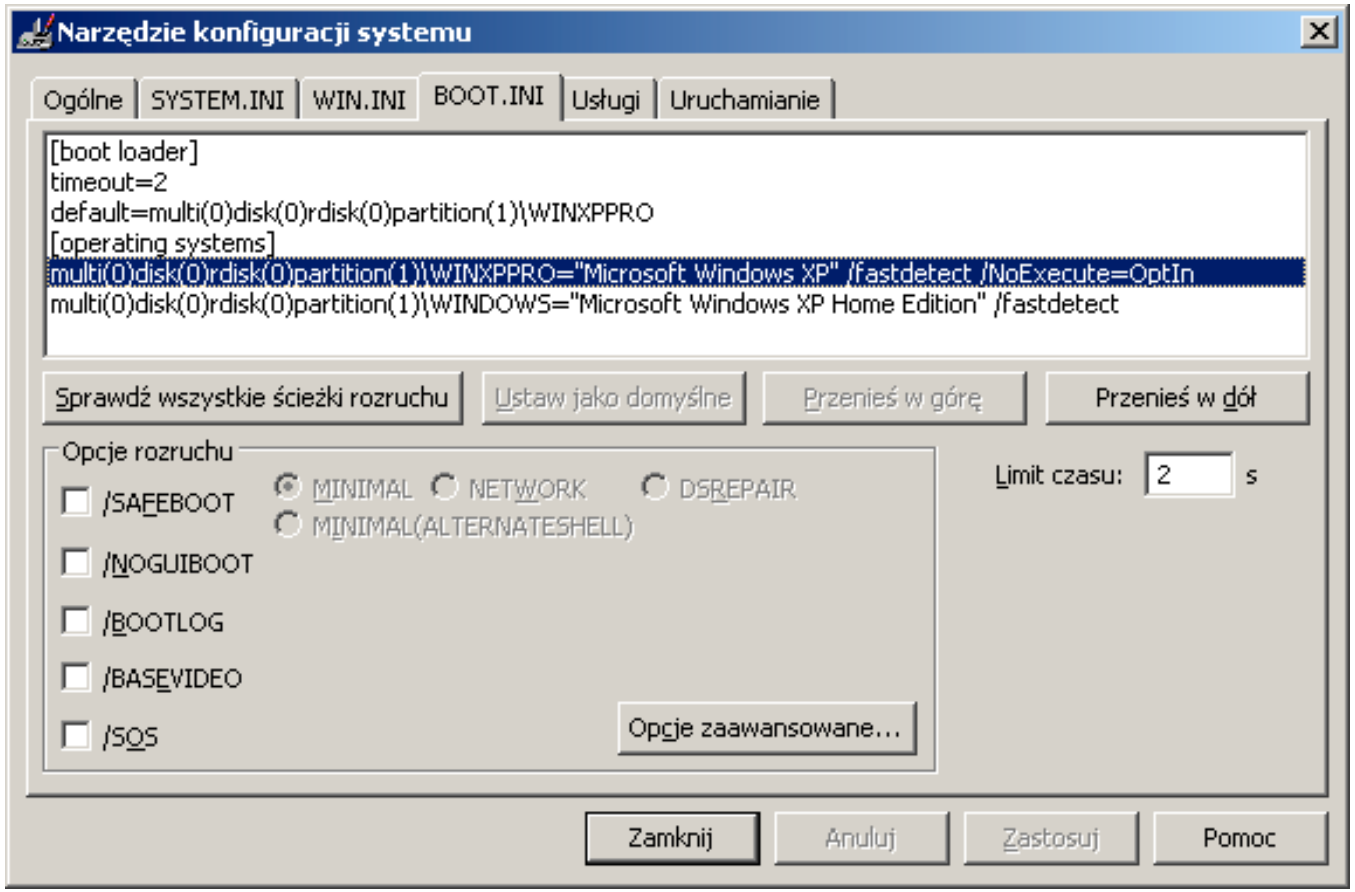
- **Włącz rejestrowanie uruchamiania** — podczas uruchamiania systemu w pliku rejestrowane są wszystkie sterowniki i usługi uruchamiane przez system. Plik wynikowy *ntbtlog.txt* umieszczony zostanie w głównym katalogu systemu *Windows*.
- **Włącz tryb VGA** — system jest uruchamiany przy użyciu podstawowego sterownika *VGA*. Ten tryb jest przydatny po zainstalowaniu nowego sterownika karty grafiki, który uniemożliwił prawidłowe uruchomienie systemu.
- **Ostatnia dobra konfiguracja** — system zostaje uruchomiony w tym stanie, w którym po raz ostatni użytkownikowi udało się do niego zalogować. W rezultacie wszystkie zmiany dokonane od ostatniego pomyślnego uruchomienia zostaną utracone.

Specjalne tryby uruchamiania systemu (Win2000/XP) - III

- **Tryb przywracania usług katalogowych** — opcja wykorzystywany wyłącznie przy uruchamianiu komputerów pełniących funkcję kontrolera domeny umożliwiająca odtworzenie lub przetestowanie działania usług katalogowych (ang. *Active Directory*).
- **Tryb debugowania** — specjalny tryb diagnostyczny umożliwiający przesyłane za pośrednictwem kabla szeregowego informacji o pracy systemu do innego komputera.

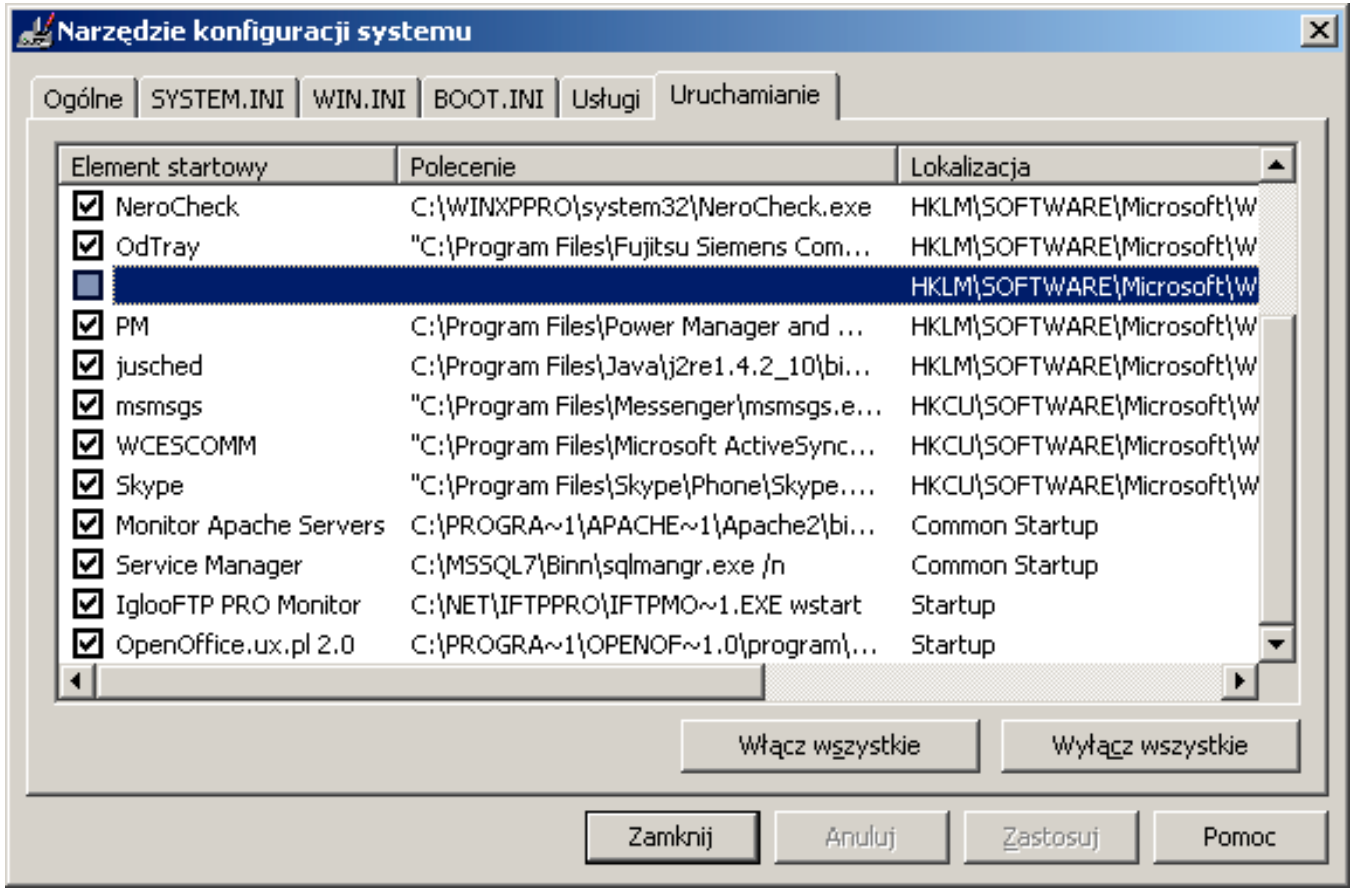
MSConfig (I)

- Konfiguracja startu systemu



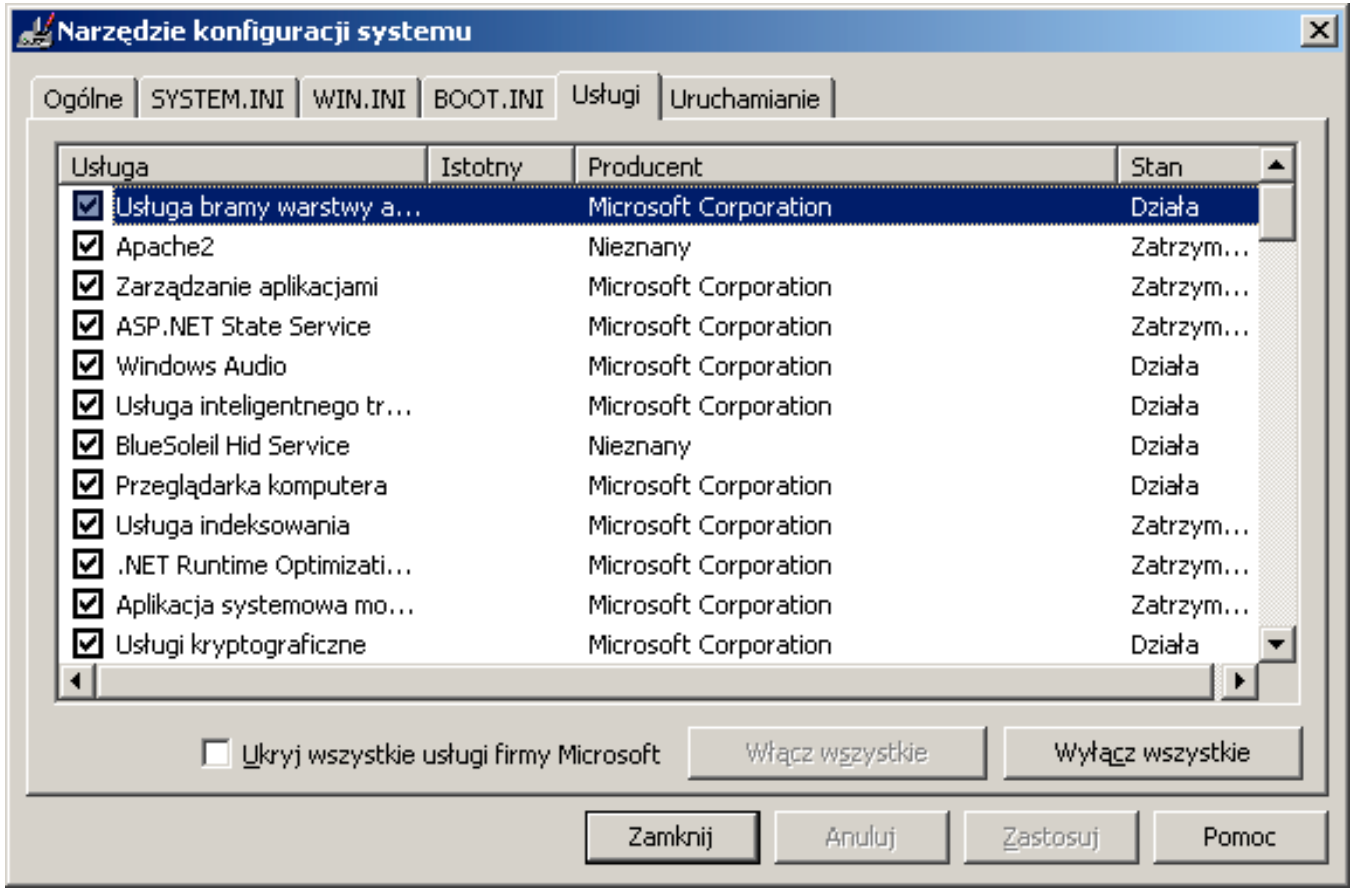
MSConfig (II)

- Autostart oprogramowania



MSConfig (III)

- Autostart usług

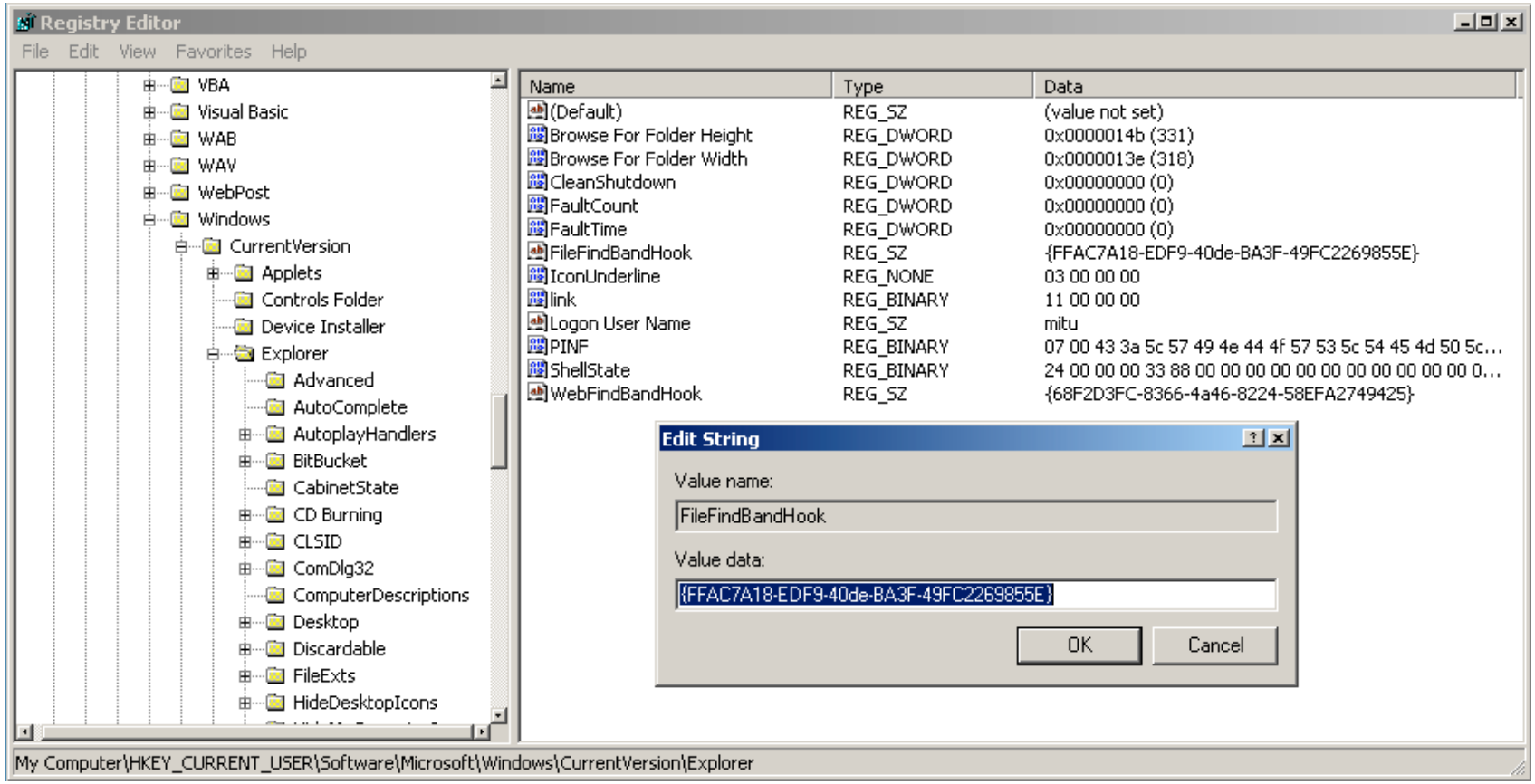


Rejestr systemu (I)

- Zintegrowana z systemem baza danych zawierająca klucze o zadanych wartościach
- Klucze uporządkowane hierarchicznie - drzewo rejestru systemowego
- Dostęp do niektórych poddrzew rejestru z poziomu API języków programowania
- Podstawowy edytor administracyjny - *regedit*
- Brak kontroli poprawności wpisów w rejestrze
- Kontrola uprawnień w zakresie odczytu i modyfikacji rejestru systemowego
- Klucze przechowywane w postaci trójki: NAZWA, TYP, DANE

Rejestr systemu (II)

- Regedit



Rejestr systemu (III)

- 6 kluczy głównych:
  - HKEY_CLASSES_ROOT
  - HKEY_CURRENT_USER
  - HKEY_LOCAL_MACHINE
  - HKEY_USERS
  - HKEY_CURRENT_CONFIG
  - HKEY_DYN_DATA (opcjonalnie)

Rejestr systemu - HKEY_CLASSES_ROOT

- Zawiera definicje wszystkich rozpoznawanych przez system typów plików. Także informacje na temat łączenia obiektów OLE dla wszystkich aplikacji obsługujących OLE.
- Częścią tej gałęzi jest klucz CLSID (skrót od Class ID - identyfikator klasy), który zawiera informacje używane do komunikacji pomiędzy aplikacjami. Cała gałąź jest lustrzanym odbiciem klucza HKEY_LOCAL_MACHINE\SOFTWARE\Classes
- Informacja w większości reprezentowana poprzez wartości kodowe - trudna do edycji.
- Kategoria **HKEY_CLASSES_ROOT** została zachowana jedynie dla zgodności z Rejestrem Windows 3.x - w rzeczywistości jest jedynie wskaźnikiem do klucza składującego wyżej opisane dane **HKEY_LOCAL_MACHINE\Software\Classes**

Rejestr systemu - HKEY_CURRENT_USER

- Charakterystyka bieżącego użytkownika
- Wskazuje na część gałęzi **HKEY_USERS**, określając aktualnego użytkownika.
- Kilka kategorii. Ważniejsze to: AppEvents, Control Panel, Keyboard Layout, Network, RemoteAccess i *Software*.
- Podgałąź *Software* - dane zainstalowanych aplikacji. Zwyczajowo katalogowane po nazwie producenta.

Rejestr systemu - HKEY_LOCAL_MACHINE

- Informacje na temat urządzeń fizycznych i oprogramowania zainstalowanego w systemie (uzupełnienie zbiorcze)
- Ustawienia zabezpieczeń systemu
- W opcjonalnej podgałęzi *Config* znajdują się alternatywne zestawy konfiguracyjne komputera, **HKEY_CURRENT_CONFIG** - bieżący
- Podgałąź *Software* - dane zainstalowanych aplikacji - niezależne od użytkownika.
- W podkluczu *Software/Microsoft/Windows* szereg dalszych ustawień konfiguracyjnych

Rejestr systemu - HKEY_USERS

- Definicje ustawień użytkowników, definicja użytkownika domyślnego
- W momencie logowania użytkownika do systemu operacyjnego, zawartość odpowiedniego klucza przenoszona jest do kategorii **HKEY_CURRENT_USER**. W przypadku, gdy nie korzystamy z wielu profili użytkowników lub użytkownik nie posiada jeszcze własnego, używany jest podklucz **.Default**. Przenoszenie w przeciwnym kierunku następuje w momencie wylogowania się użytkownika lub zamknięcia systemu.

Rejestr systemu - HKEY_CURRENT_CONFIG

- Miejsce przechowywania danych dla bieżącej konfiguracji sprzętowej komputera.
- Są tu przechowywane jedynie informacje podstawowe, niezbędne do uruchomienia systemu.
- Istnienie kategorii **HKEY_CURRENT_CONFIG** jest jedynie tymczasowe. Zbiór danych dla wszystkich konfiguracji sprzętowych znajduje się w gałęzi **HKEY_LOCAL_MACHINE\Config**. Tam też należy dokonywać ewentualnych zmian w konfiguracjach innych niż bieżąca.

Rejestr systemu - HKEY_DYN_DATA

- Drzewo tzw. *Klucza dynamicznego*. Przechowywane wyłącznie w pamięci, na bieżąco aktualizowane. Często ukryte.
- Dane zebrane w dwie grupy:
  - *Config Manager|Enum* - bieżący stan konfiguracji urządzeń fizycznych
  - ***PerfStats***. - dane o wydajności (użycie pamięci, procesora, sieci itp.) - można je przeglądać np. za pomocą monitora systemu

Rejestr systemu - dalsze informacje

- Flagi - wszystkie wartości flag typu *tak/nie* zapisywane są w polach DW (*doubleword*). Wartośc 1 oznacza „włączone”, 0 - „wyłączone”
- Istnieją narzędzia do monitorowania rejestru i analizy zmian - przykład: *Advanced Registry Tracer*
- Pliki skryptu modyfikującego rejestr - **.reg.* Wymiana informacji pomiędzy rejestrem a plikami dyskowymi.
- Możliwe wartości danych w kluczu: łańcuch, wartość binarna, liczba 32 bitowa (double word) , multi string (wiele linii)

Rejestr systemu - pliki z kluczami rejestru

- Zapis i odczyt - przy pomocy *regedit* z linii komend lub z użyciem funkcji *import/export*:
  - Przykładowy zapis do pliku: *regedit /e rejestr.reg HKEY_CURRENT_USER|Software|App1*
  - Przykładowe pobranie z pliku: *regedit -s rejestr.reg*
- Format pliku:
  - *[HKEY_LOCAL_MACHINE|Software|App1]* - lokalizator klucza w linii pliku powoduje przejście do tego klucza lub jego stworzenie gdy stwierdzono brak
  - Po lokalizatorze w kolejnych liniach występują definicje danych w kluczu. Format: *"nazwa"=typ:wartosc*

Rejestr systemu - ciekawe miejsca

- HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion szereg ustawień np. pełne ścieżki do aplikacji, ustawienia specjalne pulpitu, fonty, wygaszacze, ustawienia sieci, dane do deinstalacji aplikacji, autostart aplikacji
- Wyłączanie dostępu do funkcjonalności systemu:
  - HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\ Policies\Explorer - ograniczenia interfejsu
  - HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\ Policies\Network - ograniczenia stref sieciowych
  - HKEY_CURRENT_USER\Soft.\Microsoft\Windows\ CurrentVersion\Policies\WinOldApp - ograniczenia aplikacji (np. możliwość uruchomienia aplikacji DOS,
- HKLM\Software\Microsoft\Windows\CurrentVersion\ Policies\Network - ograniczenie udostępniania własnych zasobów, manipulowania hasłami

Biblioteki funkcji systemowych

- Większość funkcjonalności interfejsu okienkowego Windows jest realizowana poprzez systemowe biblioteki funkcji, głównie *shell.dll, shell32.dll, krnl386.exe, krnl386.exe.*
- Możliwe jest korzystanie z tych funkcji z poziomu skryptu czy linii komend.
- W celu uruchomienia funkcji systemowej należy posłużyć się programem *rundll32.exe podając biblioteką i funkcję jako parametr,np.: rundll32.exe shell32,Control_RunDLL* - uruchomienie funkcji „Panel sterowania” biblioteki Shell32

Backup systemu (I)

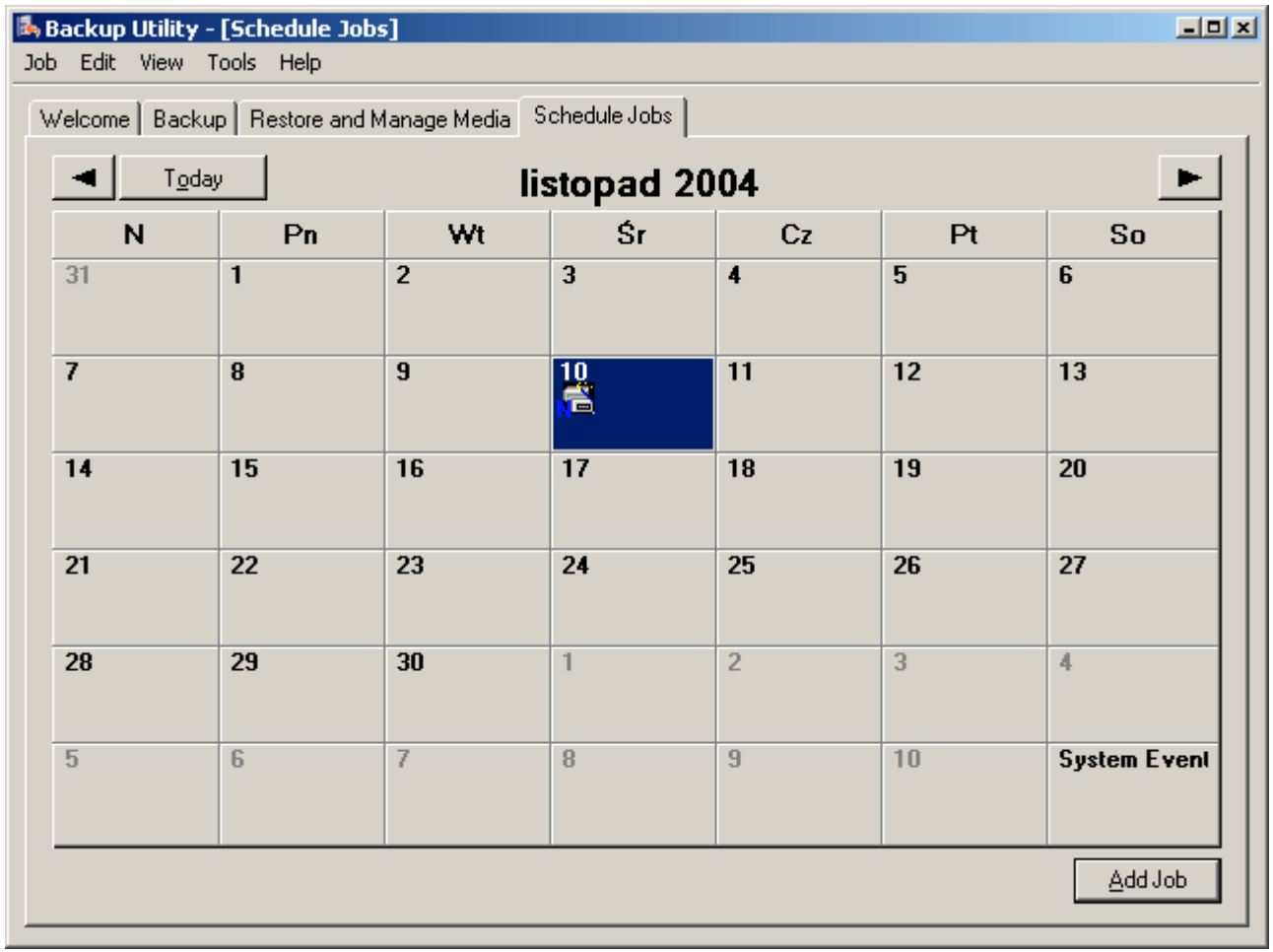
- Począwszy od win9x do systemu dołączone jest narzędzie wspomagające automatyczny backup ustawień systemowych, ustawień użytkowników, krytycznych plików systemowych, poczty itp.
- Standardowa nazwa: *system32|ntbackup.exe*
- Pliki z danymi: **.bkf*
- Funkcjonalność ASR - *Automated System Recovery*
- Zarządzanie wieloma kopiami zapasowymi
- Scheduler

Backup systemu (II)

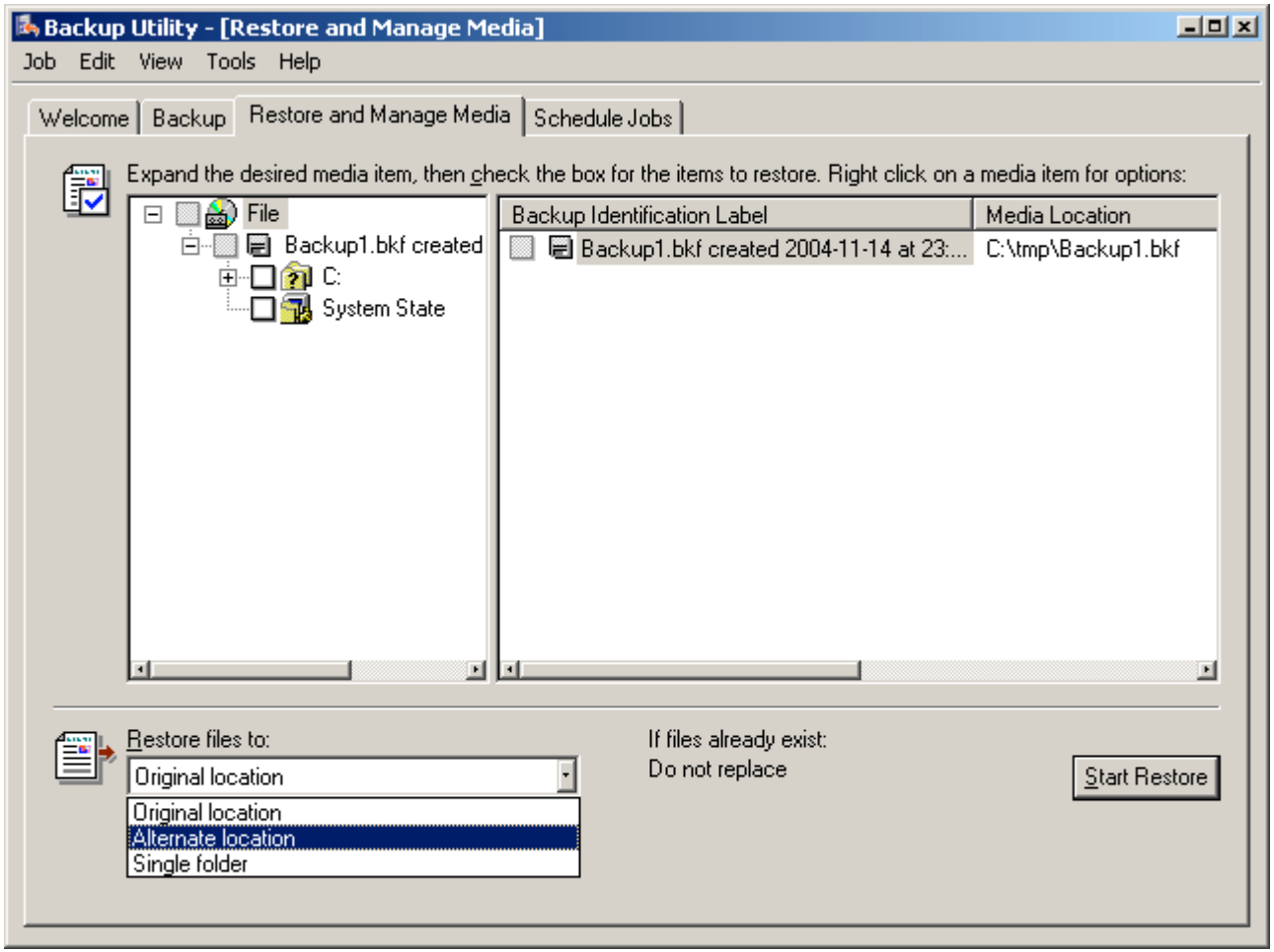
- Tryby backup podzielono na trzy kategorie: całość, backup na poziomie plików, backup ustawień (stanu komputera).
- W późniejszych wersjach systemów (2003) wprowadzono backup różnicowy i przyrostowy.
- Logi systemu kopii zapasowych (od wersji 2000):

  "C:|Documents and Settings|username|Local Settings|Application Data|Microsoft|Windows NT|NTBackup|data|"

Backup systemu (III)

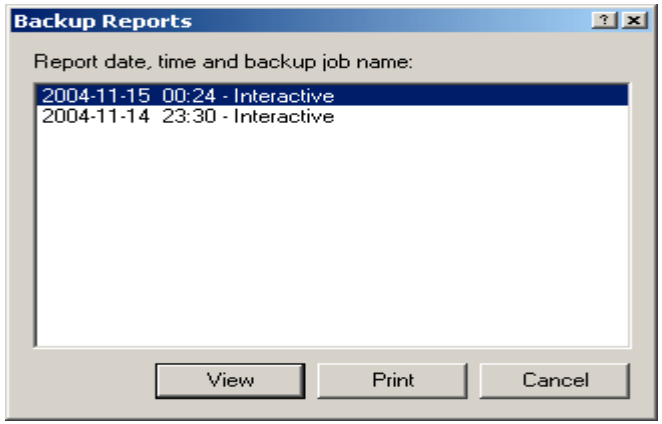


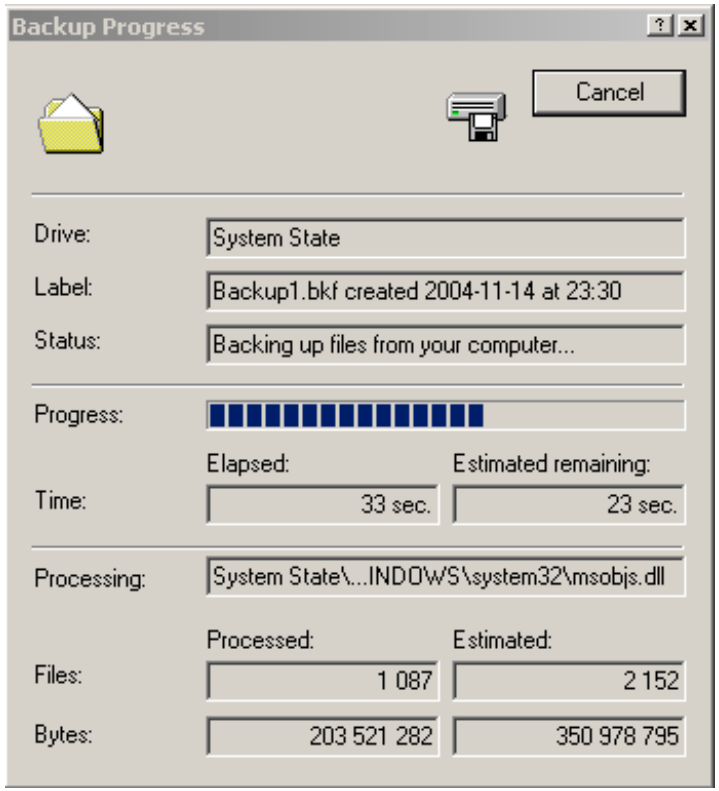
Backup systemu (IV)



Backup systemu (V)

- Backup of "System State"
- Backup set #2 on media #1
- Backup description: "Set created 2004-11-15 at 00:24"
- Media name: "Backup1.bkf created 2004-11-14 at 23:30"
- Backup Type: Copy
- Backup started on 2004-11-15 at 00:25.
- Backup completed on 2004-11-15 at 00:26.
- Directories: 141
- Files: 2152
- Bytes: 351 026 711
- Time: 1 minute and 0 seconds

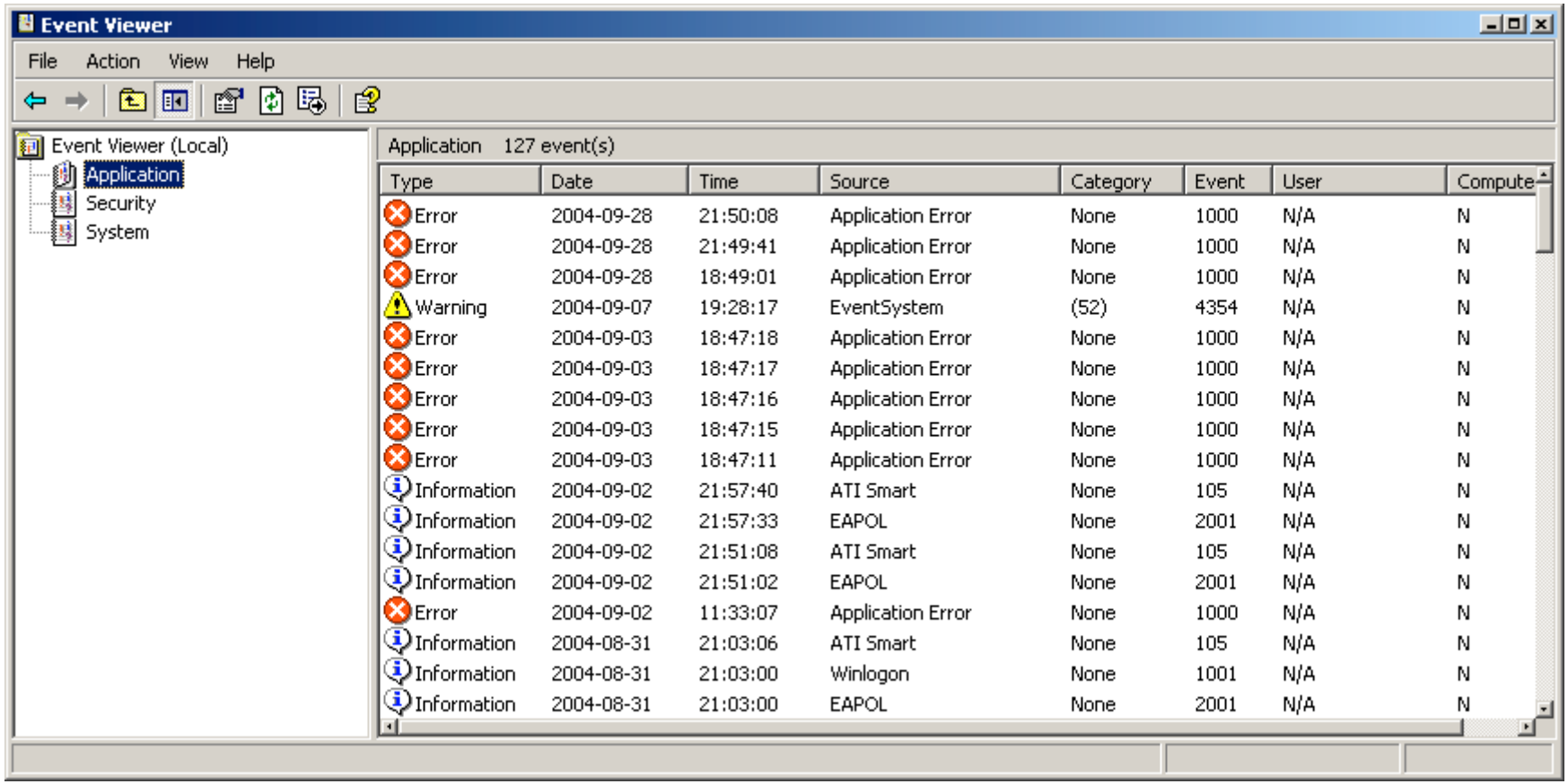




Konsola Dziennika Zdarzeń (I)

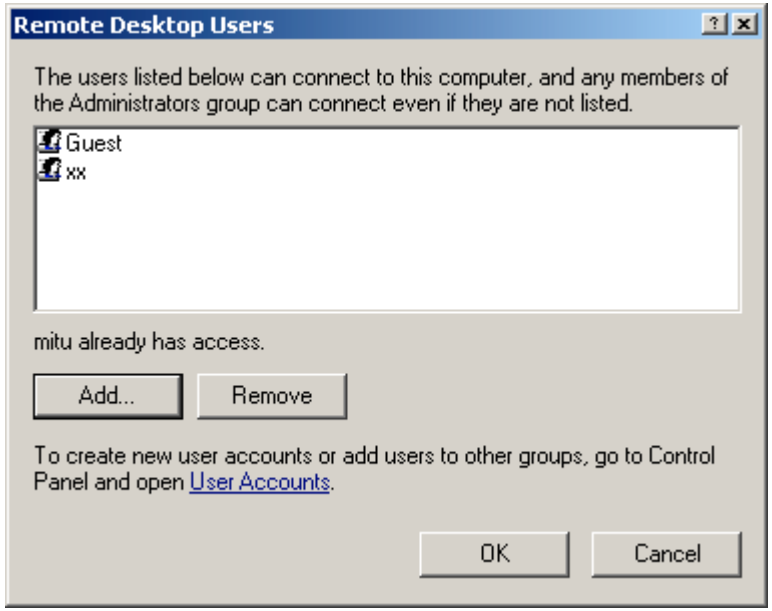
- **Dziennik aplikacji** - informacje o zdarzeniach generowanych przez programy,
- **Dziennik systemu** - informacje o zdarzeniach wywołanych przez system operacyjny,
- **Dziennik zabezpieczeń** - informacje o zdarzeniach związanych z bezpieczeństwem danych (domyślnie brak monitorowania, tutaj są umieszczane komunikaty np. audytu plików)

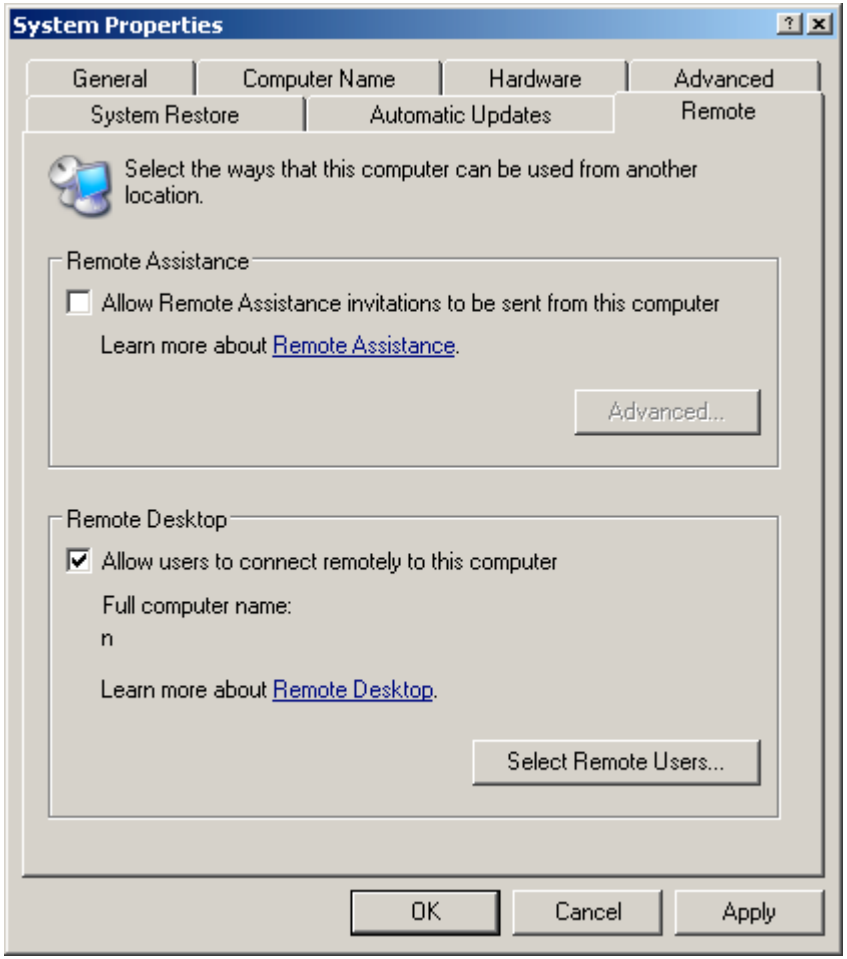
Konsola Dziennika Zdarzeń (II)



Usługa
Remote Desktop Connection

- Konfiguracja serwera



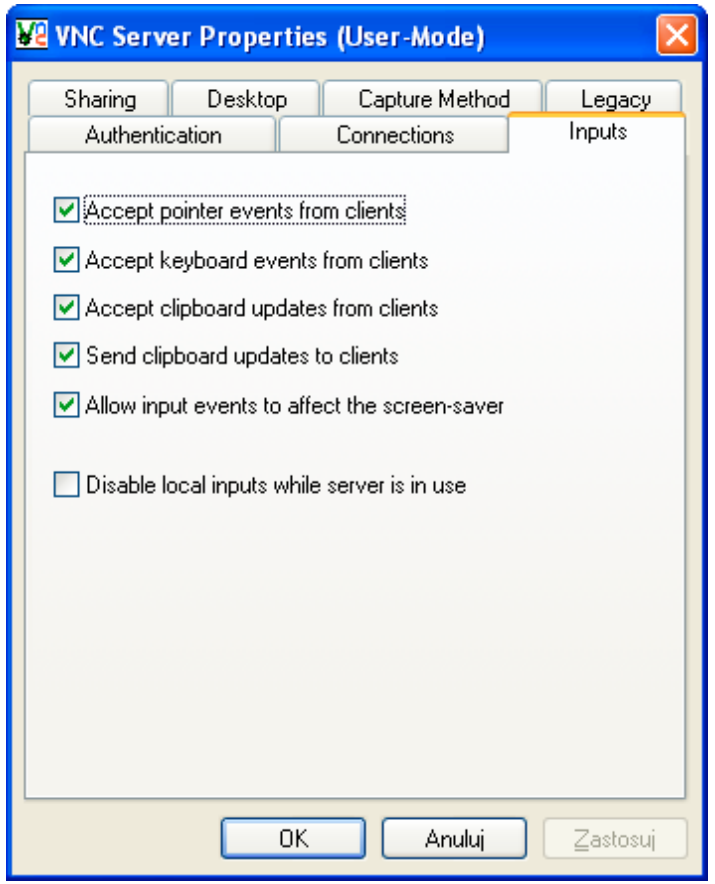


Narzędzia wspomagające zdalny dostęp w Windows (I)

- Klient Remote Desktop
  - Aplikacja klienta: WindowsDir/system32/mstsc.exe
  - Pliki konfiguracji połączenia zdalnego (*. RDP), przechowujące lokalizacje, hasła, ustawienia wyświetlania, strumieni multimedialnych maszyny zdalnej, urządzeń terminalnych maszyny lokalnej

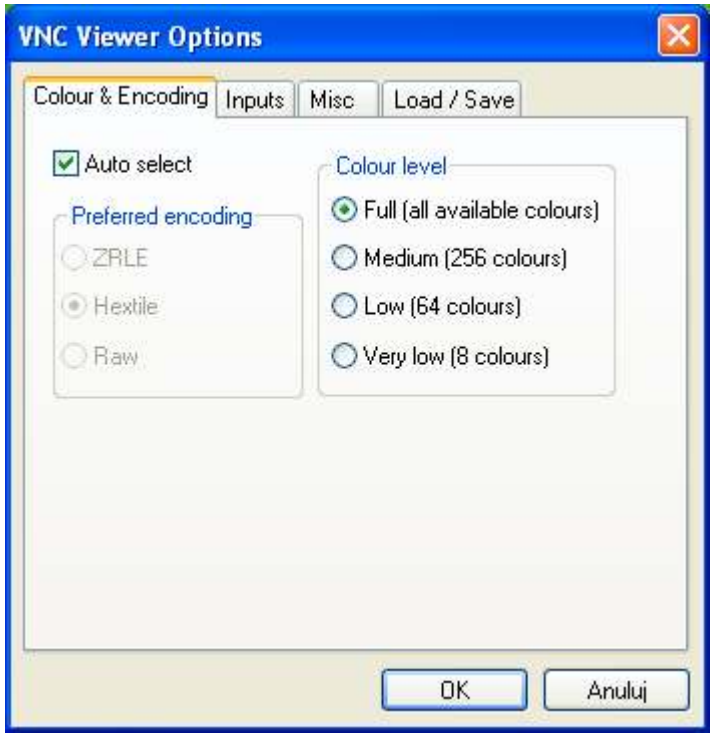
Narzędzia wspomagające zdalny dostęp w Windows (II)

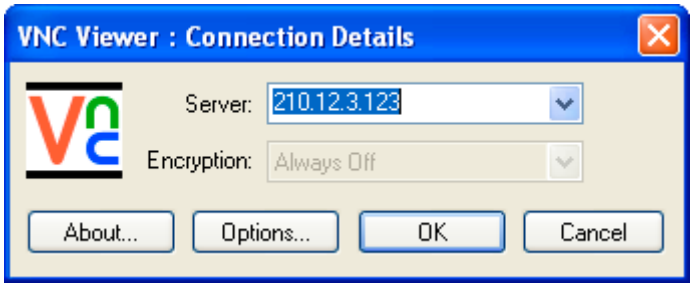
- VNC
  - Narzędzie: RealVNC
  - Architektura client/server
  - Filtry IP, logowanie własne lub NT, integracja funkcjami logowania Windows (automatyczne logowanie/wylogowanie użytkownika sesji)

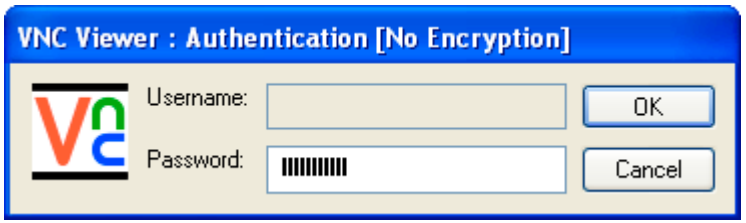


Narzędzia wspomagające zdalny dostęp w Windows (III)

- Klient VNC

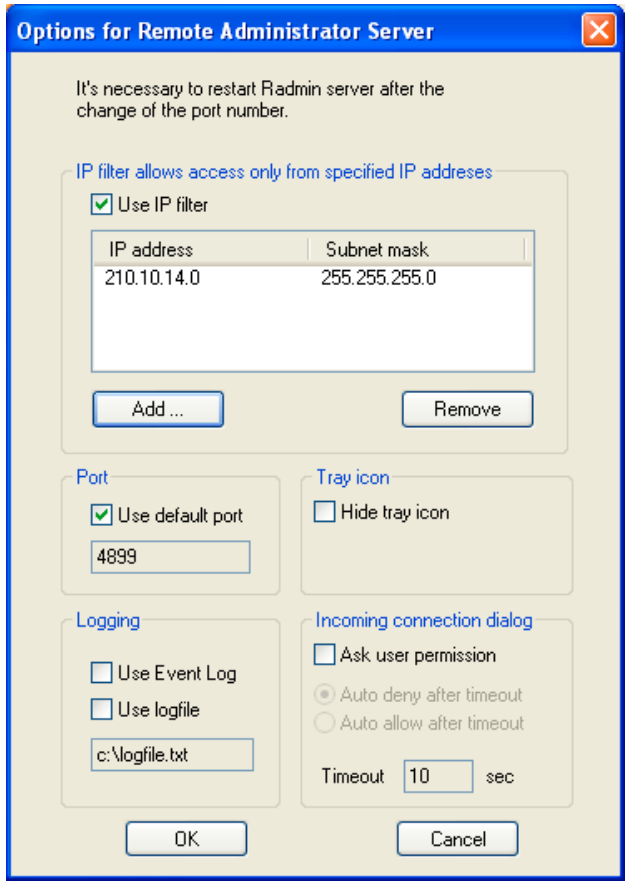






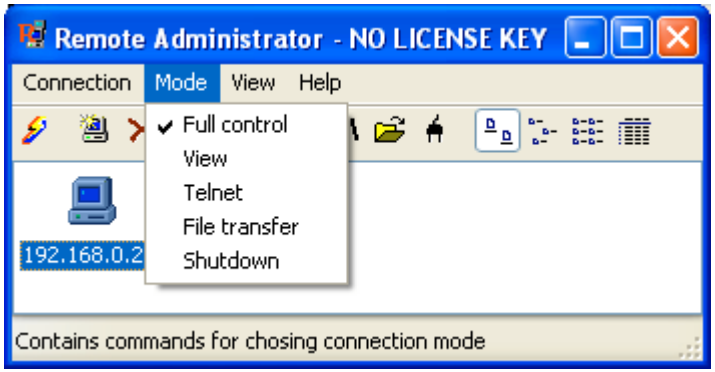
Narzędzia wspomagające zdalny dostęp w Windows (IV)

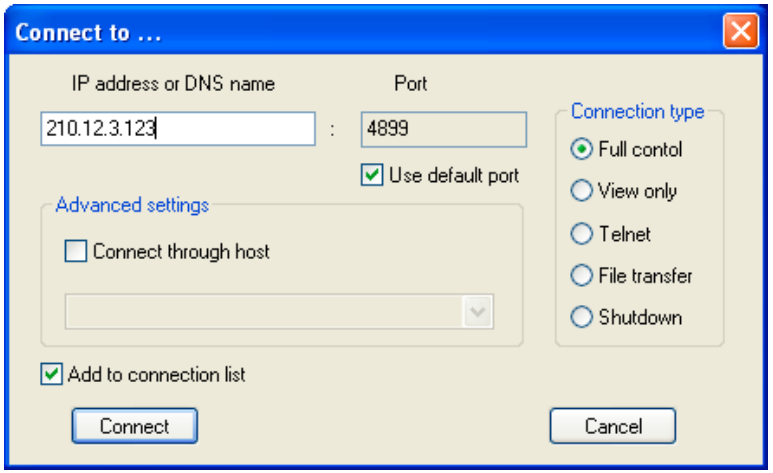
- Remote Administrator (Radmin)
  - Wiele usług:
    - Okno graficzne
    - Telnet
    - Transfer plików
    - Zdalny reset
  - Filtry IP, logowanie własne lub NT, audyty
  - Słaba wydajność w porównaniu z innymi narzędziami

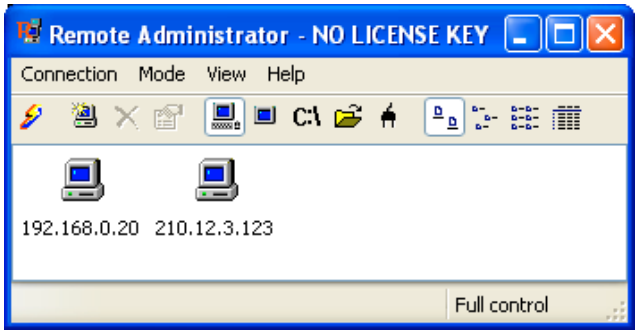


Narzędzia wspomagające zdalny dostęp w Windows (V)

- Klient Radmin







Konfigurowanie systemów typu Windows Server

- Role serwera - grupy funkcjonalności serwera połączone znaczeniowo
- Powiązanie z serwerami usług
- Wspólny interfejs konfigurujący usługi
- Dynamiczne instalacja serwerów usług z chwilą pierwszej konfiguracji

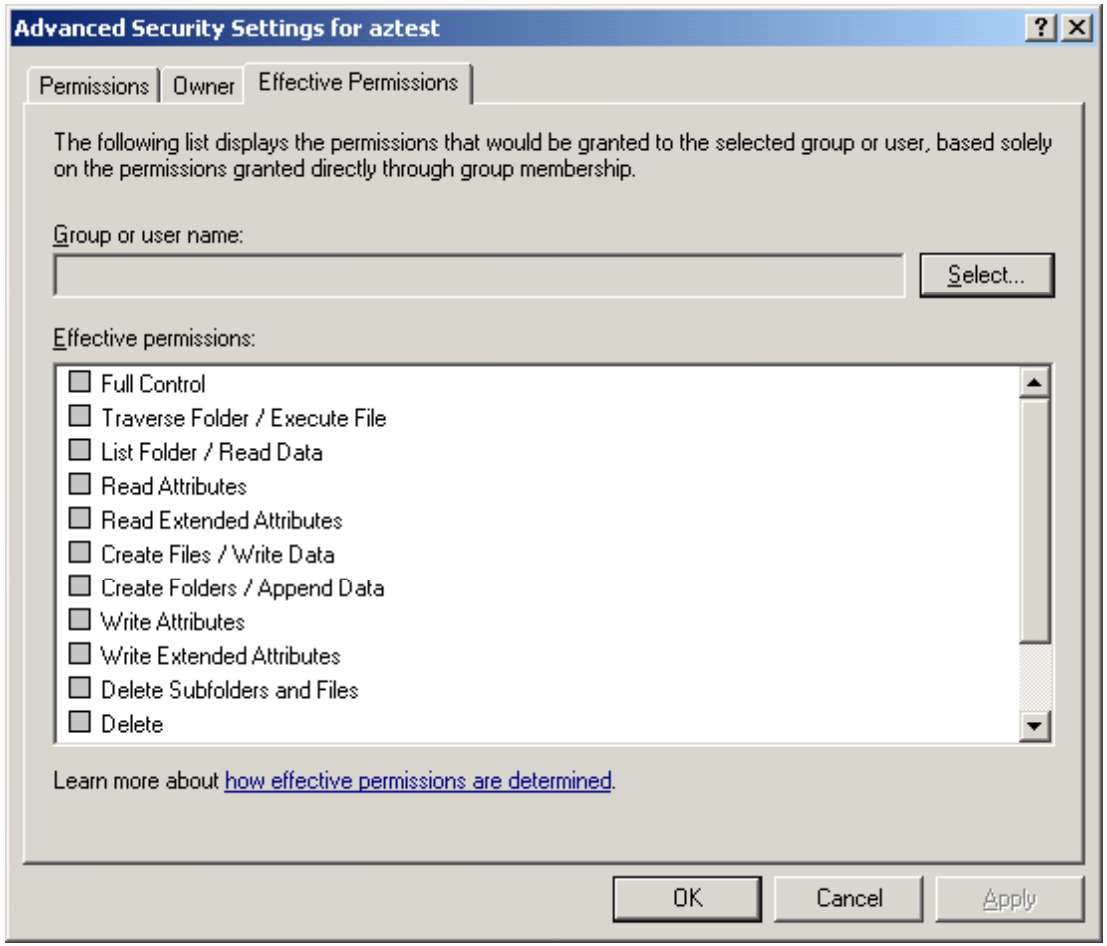
Uprawnienia Windows Serwer

- Można je przypisać:
  - Grupom, użytkownikom i specjalnym tożsamościom w danej domenie.
  - Grupom i użytkownikom w danej domenie i we wszystkich zaufanych domenach.
  - Lokalnym grupom i użytkownikom komputera, na którym rezyduje obiekt.
- Pula uprawnień wspólnych dla wszystkich obiektów (niezależnie od typu):
  - Uprawnienia do odczytu
  - Uprawnienia do modyfikacji
  - Uprawnienia do zmiany właściciela
  - Uprawnienia do usunięcia

  Pozostałe uprawnienia są zależne od typu obiektu

Efektywny zestaw uprawnień

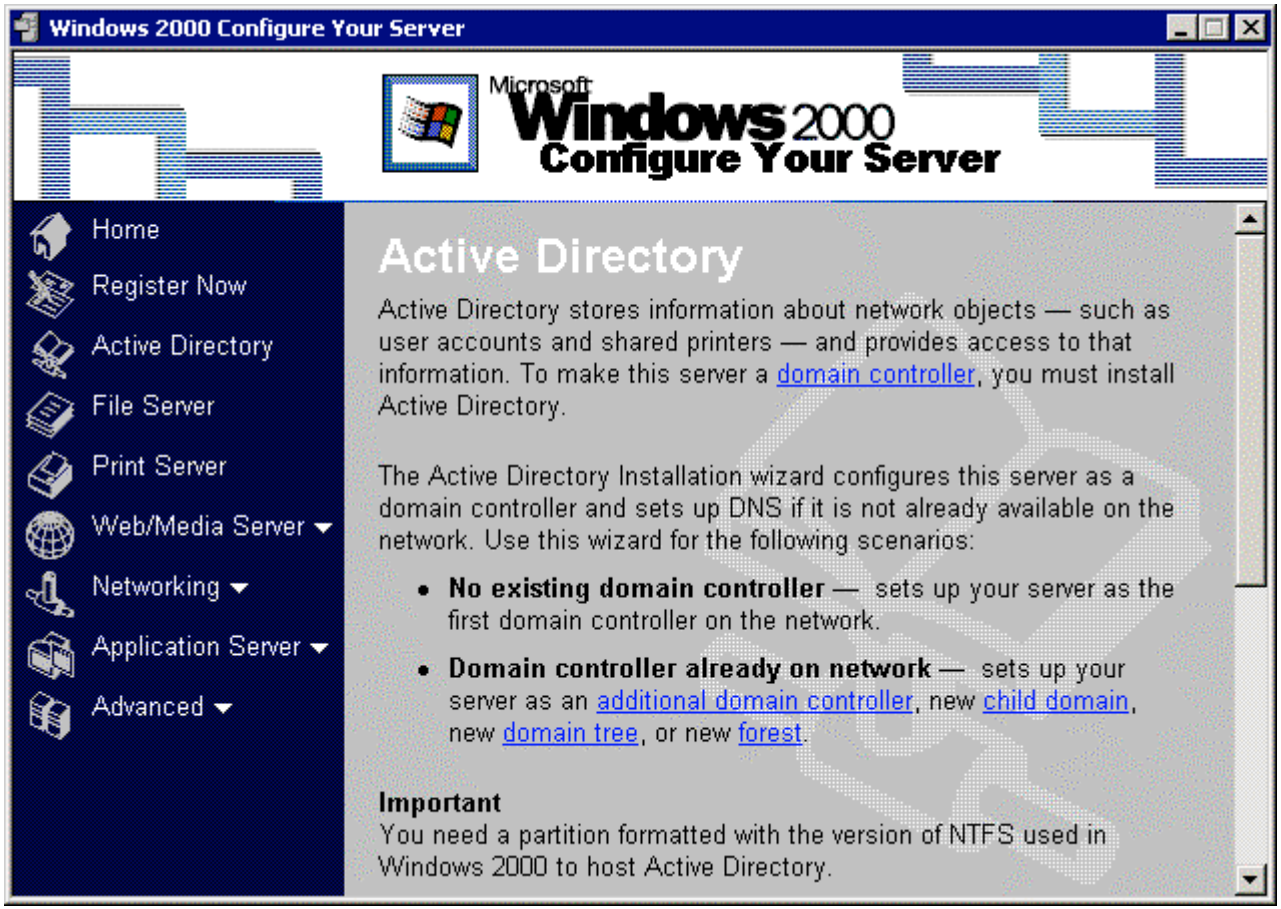
- Aplet, obliczający wynikowe uprawnienia do obiektu:



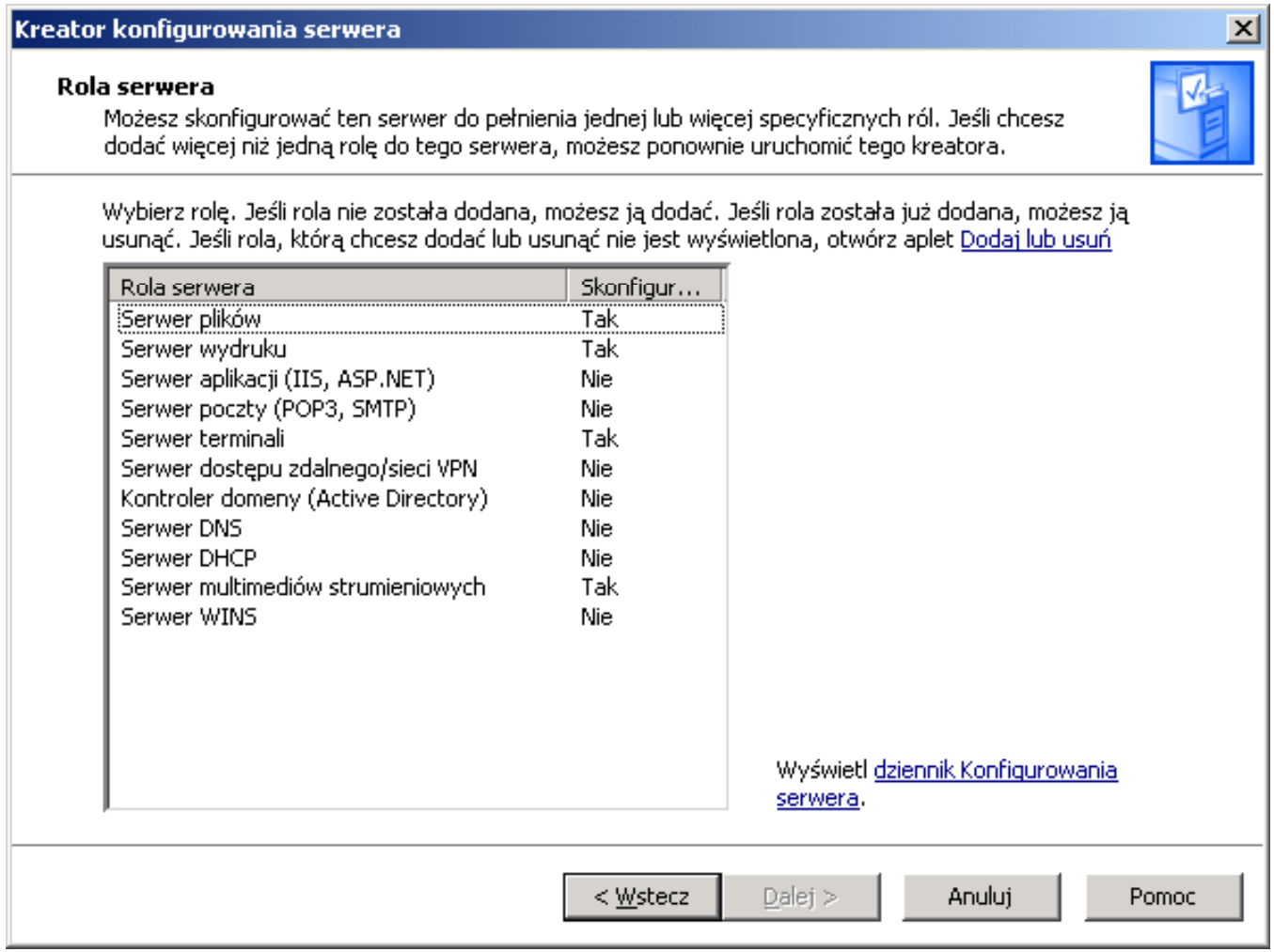
Role serwera (I)

Role serwera (II)

- Wersja Server 2000



Role serwera (III)



Rola - Serwer plików

- Udostępniane udziałów plikowych,
- Limity powierzchni dyskowej nakładane na użytkownika
- Zarządzanie usługą indeksowania, określanie katalogów i plików nieindeksowanych,
- Zarządzanie systemem Volume Shadow Copy – system umożliwia przykładowo automatyczny backup przed zmianami w plikach i dostęp do poprzednich wersji plików zapisanych na serwerze.

Rola - Serwer wydruku

- Zarządzanie drukarkami oraz zadaniami drukowania,
- Instalacja portów drukarek
- Administrowanie drukarkami na innych (w tym sprzętowych) serwerach wydruku
- Konfiguracja serwisu WEB instalacji drukarek. Klient w LAN może zainstalować drukarkę sieciową przez WWW

Rola - Serwer aplikacji

- Konfigurator IIS 6.0 - w tym nowości względem 5.x:
  - Usługi hostingowe - wiele wirtualnych serwerów WWW w ramach jednej fizycznej lokalizacji
  - Definiowanie puli zasobów reużywalnych
  - Mechanizmy failover - akcje na wypadek awarii usługi
  - Monitorowanie obiektów

Rola - Serwer poczty

- Serwer poczty instalowany i zarządzany niezależnie od IIS
- Konfiguracja SMTP pozwalającego na wymianę listów e-mail i POP3 (Post Office Protocol 3)
- Serwer dedykowany dla klientów z rodziny Microsoft - implementuje skrzynki pocztowe użytkowników w domenie, konfigurowane następnie np. z narzędzia Microsoft Outlook.

Rola - Serwer usług terminalowych

- Zdalne zarządzanie za pośrednictwem terminali - więcej niż dwa połączenia równocześnie (dwa dopuszczone przez usługę Remote Desktop for Administration)
- Udostępnianie aplikacji - unifikacja wersji kodu aplikacji wykonywanego na stacjach klienckich
- Serwer licencji - limit czasu (120 dni) na licencję dla usług terminalowych

Rola - Active Directory

- Zarządzanie:
  - centralnym katalogiem informacji o użytkownikach
  - domenami
  - informacjami o autoryzowanych komputerach
  - informacjami o usługach w sieci
- Konfiguracja do pracy w dowolnej podgałęzi struktury katalogów,
- Ustalanie trybu dystrybucji zabezpieczeń sieci w obrębie domeny

Rola - Serwer DNS

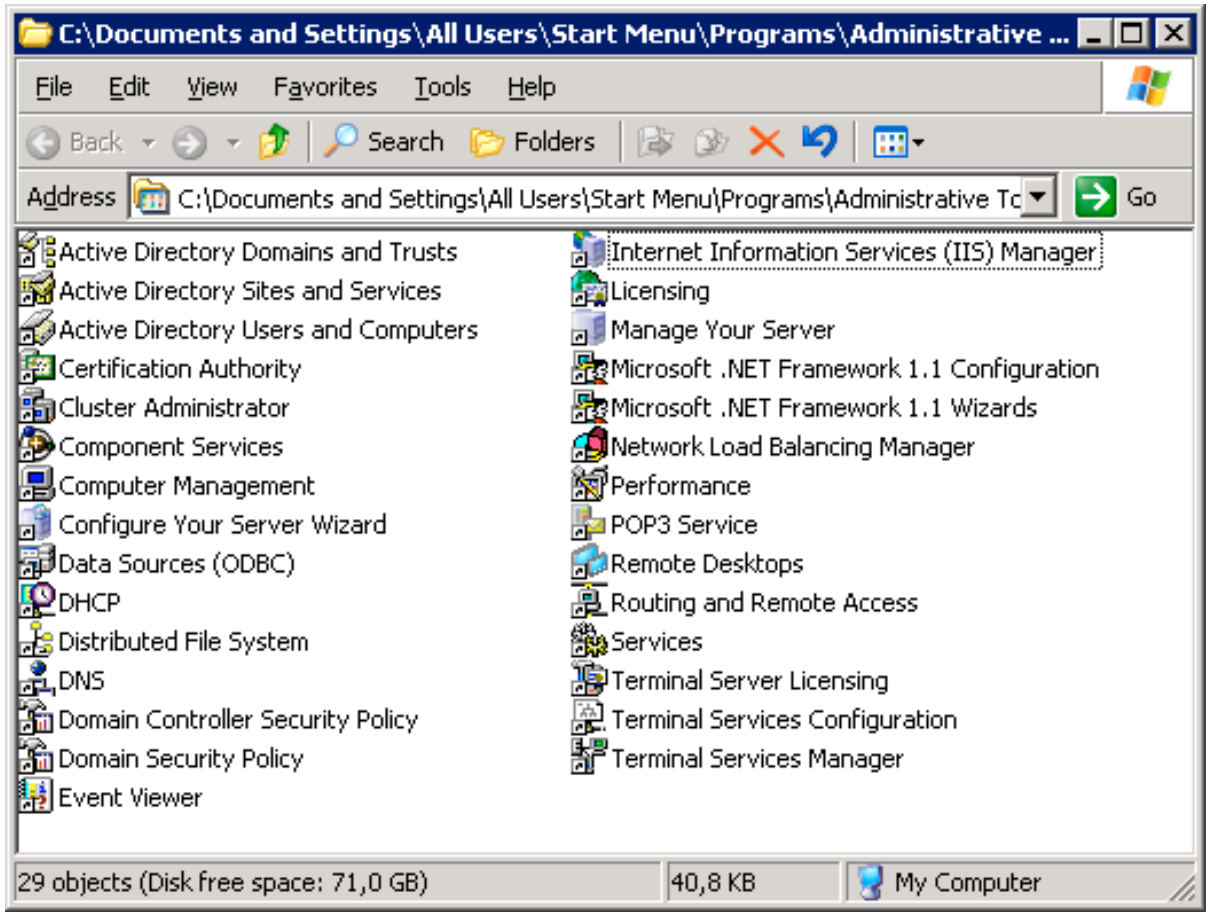
- Strefy przeszukiwania wrzód i przeszukiwania wstecznego
- Modyfikacja rekordów DNS
- Zasady migracji rekordów DNS
- Alternatywa (Windows 2003 Server): Centralny system niezależny rozpoznawania nazw w sieci lokalnej lub usługa zintegrowana z Active Directory.

Rola - Serwer DHCP

- Zarządzanie pulą adresów IP
- Ustalanie parametrów połączenia TCP/IP, automatycznie przekazywanych do nowych stacji klienckich w sieci (np. adresu bramki, serwerów DNS)
- Mechanizmy automatycznego odświeżania – dostosowanie numeracji IP do typów urządzeń

Pula narzędzi administracyjnych

- Stan po zainstalowaniu większości istotnych ról:



Active Directory - kontrolery domeny (I)

- Kontroler domeny - zarządza dostępem do współdzielonych zasobów w ramach domeny
- uwierzytelnia logowania do domeny i utrzymuje zasady zabezpieczeń oraz główna bazę danych dla domeny
- Występuje w postaci tzw. *Roli* fizycznego komputera (serwera), w systemie którego jest zainstalowany
- Poszczególne kontrolery domen posiadają hierarchię drzewiastą, z reguły zgodną z hierarchią nazw domen sieci rozległej (jeśli w bieżącej sieci taka istnieje)

Active Directory - kontrolery domeny (II)

- Komputery podłączone do domeny posiadają odpowiednie konta w kontrolerze domeny
- Maszyna z kontrolerem domeny może być jednocześnie bramą podsieci i udostępniać sieć zewnętrzną również dla maszyn nie będących członkami domeny
- Możliwy jest dostęp do zasobów domeny spoza komputerów w domenie - jednak po autoryzacji użytkownika zarejestrowanego w domenie, przykładowo:
  - net use \\serwer\zasób /user:domena\konto

Active Directory - LDAP (I)

- *Lightweight Directory Access Protocol* (LDAP) - standard zakładający składowanie zasobów w hierarchicznej bazie danych
- Protokół użytkowany przez Active Directory
- Centralne przechowywanie danych i informacje o lokalizacji danych
- Centralne zarządzanie strukturą bazy danych
- Eliminacja kolizji nazewnictwa zasobów
- Podobieństwo do struktury DNS (Domain Name System) - LDAP jest tworzone z użyciem nazewnictwa DNS wkomponowanego we własną bazę danych

Active Directory - LDAP (II)

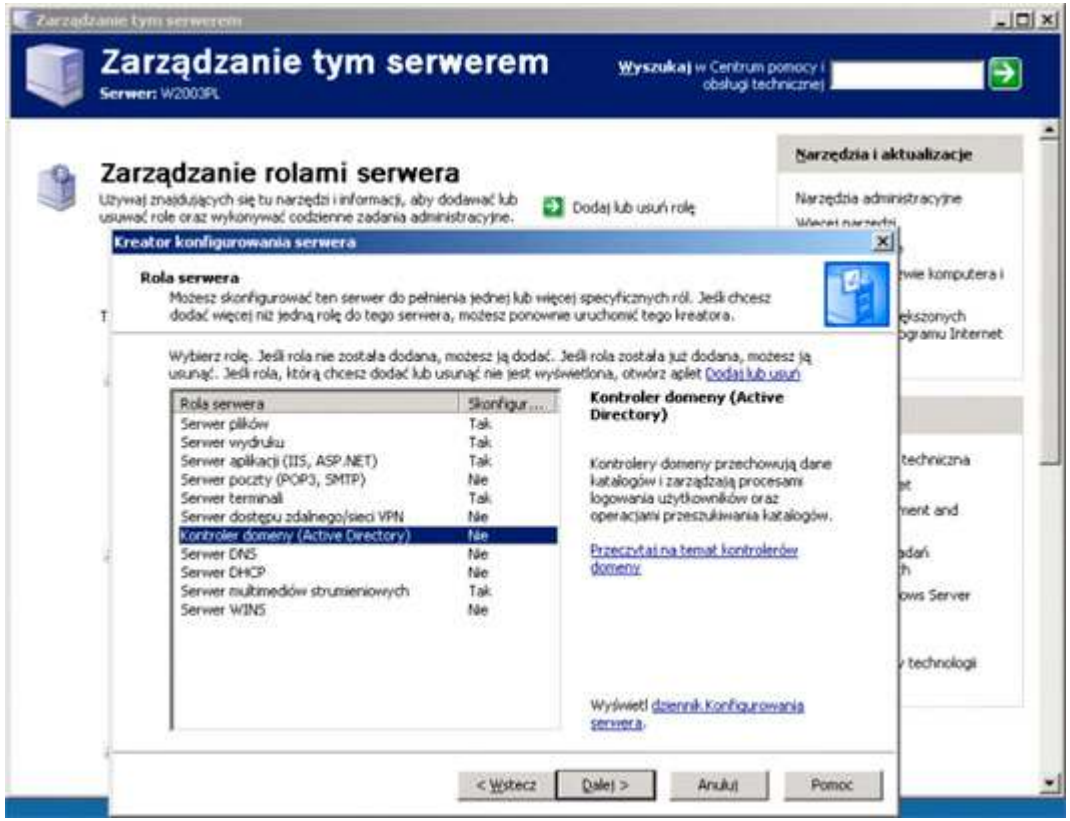
- Przykłady popularnych kontenerów LDAP:
  - **DC** (*Directory Context*) - kontener ogólnego stosowania, używany np. do oznaczenia domeny.
  - **OU** (*Organizational Unit*) - jednostka organizacyjna zawierająca inne obiekty.
  - **CN** (*Common Name*) - wykorzystywany do reprezentowania użytkowników, komputerów, drukarek.

Active Directory a Domena NT

- Domena NT - jednopoziomowa, kolekcje homogenicznych obiektów
- Active Directory - wprowadzenie uniwersalnych obiektów, reprezentujących bliżej niesprecyzowaną jednostkę w sieci - użytkownika, komputer, serwer, bazę danych, drukarkę
- Wprowadzenie API udostępniającego Active Directory z poziomu języków programowania np. ADSI (Active Directory Service Interface).

Instalacja kontrolera domeny (I)

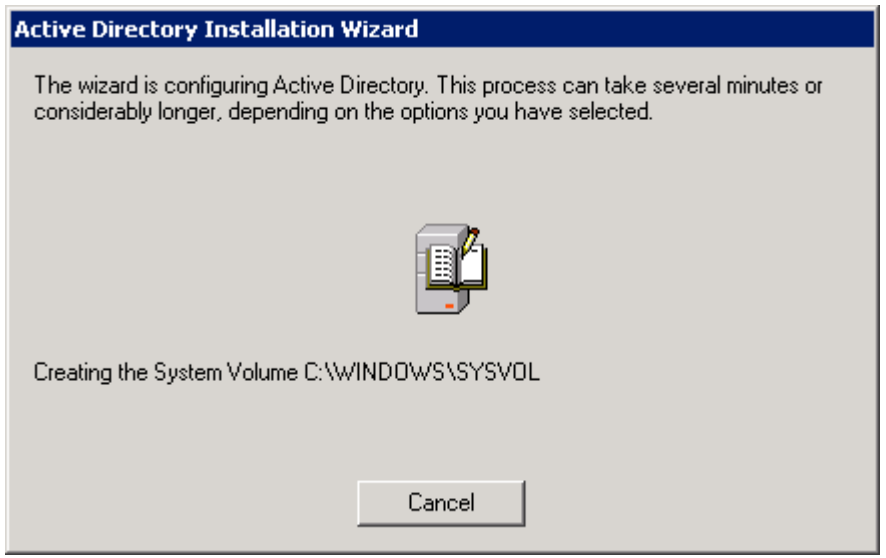
- Instalacja uruchamiana ze standardowego panelu „Role serwera”



Instalacja kontrolera domeny (II)

Instalacja kontrolera domeny (III)

- Domena specjalna - *local*
- *Katalog systemowy domeny to SYSVOL.* Jest replikowany z innych kontrolerów w obrębie domeny - koniecznie na partycji NTFS
- Lokalizacja baz danych *Active Directory* i dziennika

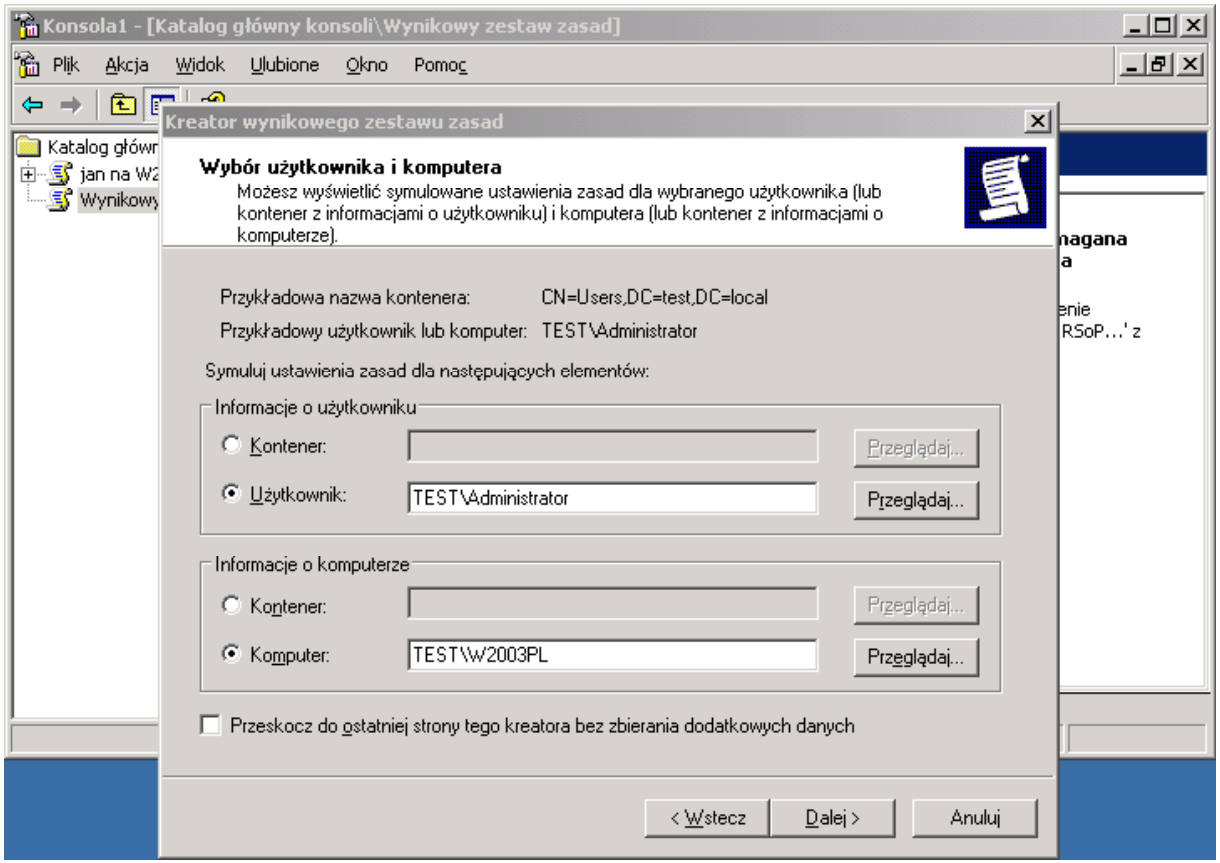


Kontroler domeny - zabezpieczenia (I)

- Applet MMC - Wynikowy zestaw zasad
- Symulacja lub planowanie zasad na podstawie polityki grup serwera, praw własności zasobów, ustawień prywatnych docelowo w odniesieniu do całej domeny
- Na poziomie użytkownika, grupy, maszyny

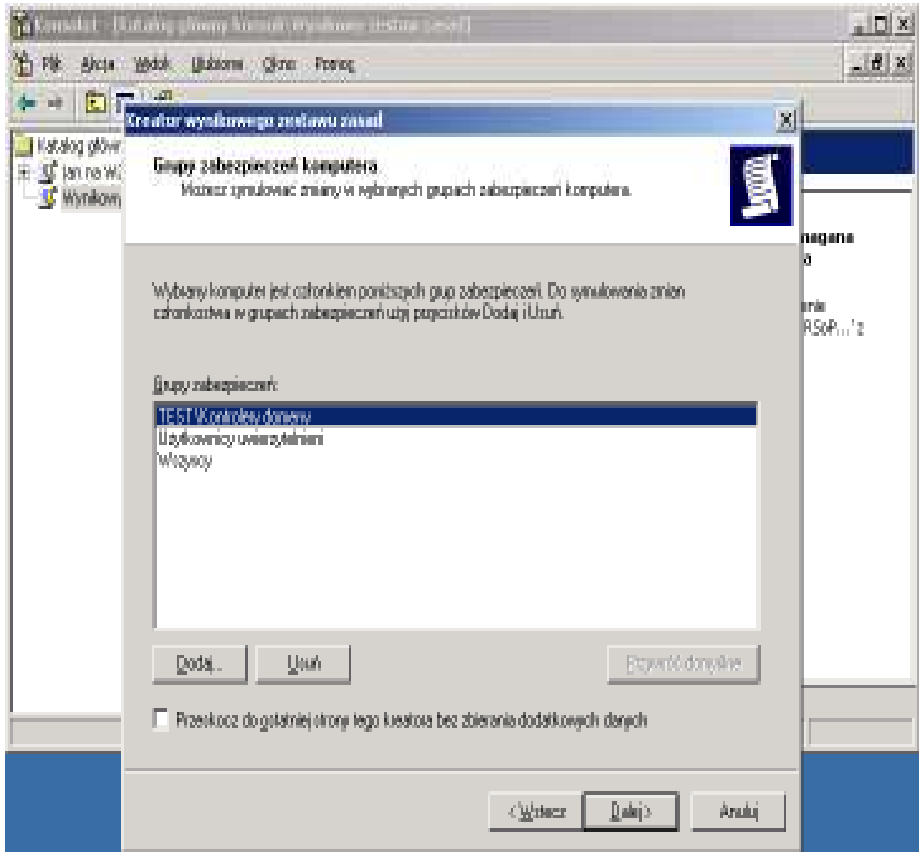
Kontroler domeny - zabezpieczenia (II)

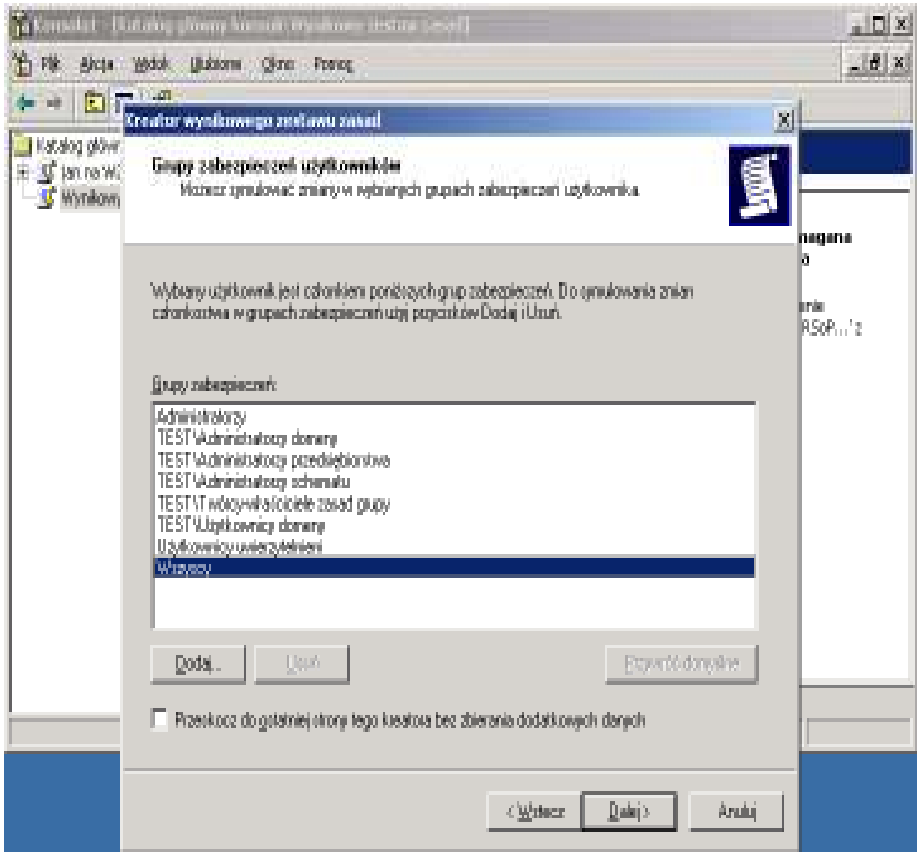
- Tworzenie zestawu zasad



Kontroler domeny - zabezpieczenia (III)

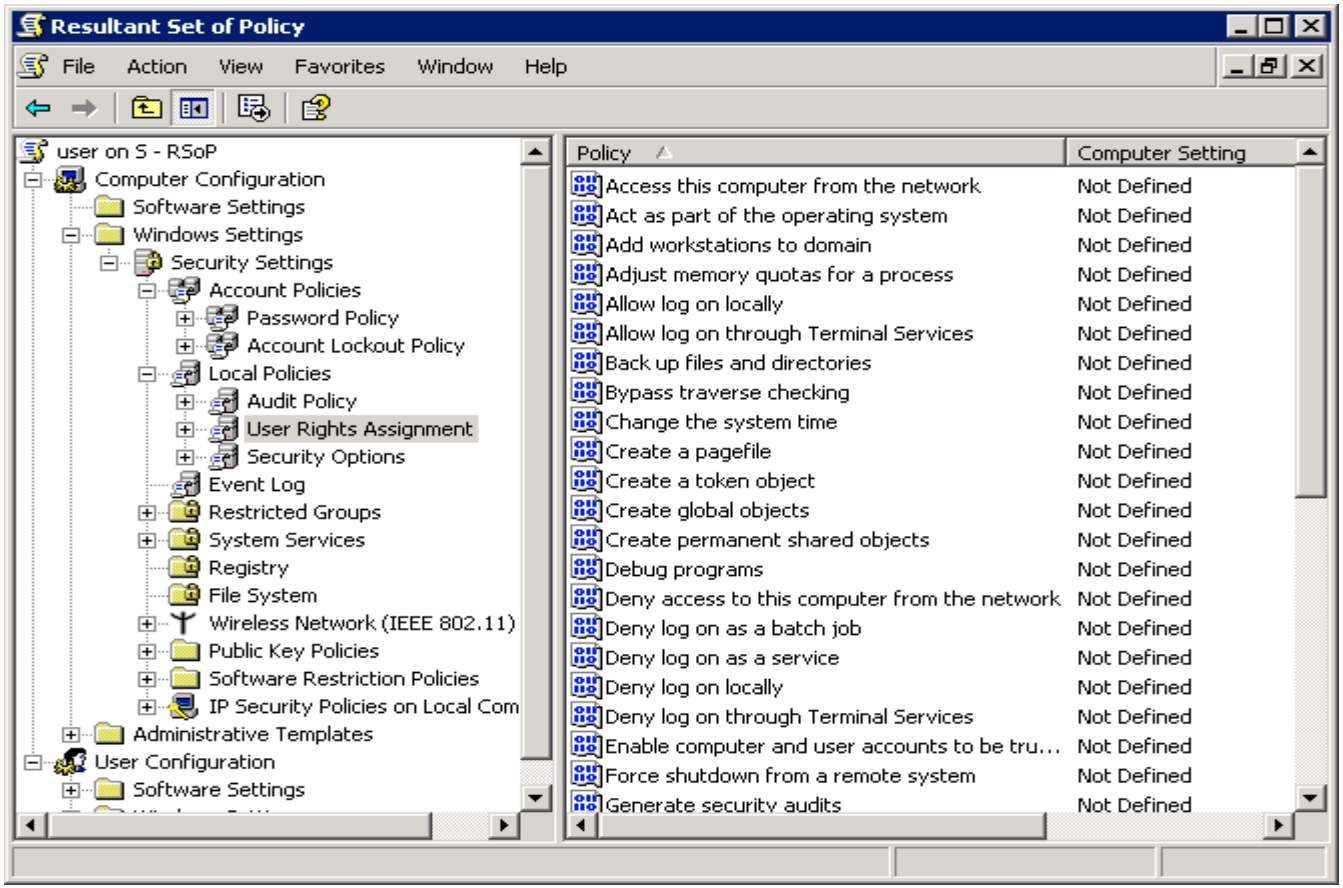
- Wybór grup użytkowników, komputerów w domenie





Kontroler domeny - zabezpieczenia (IV)

- Zestawienie zbiorcze

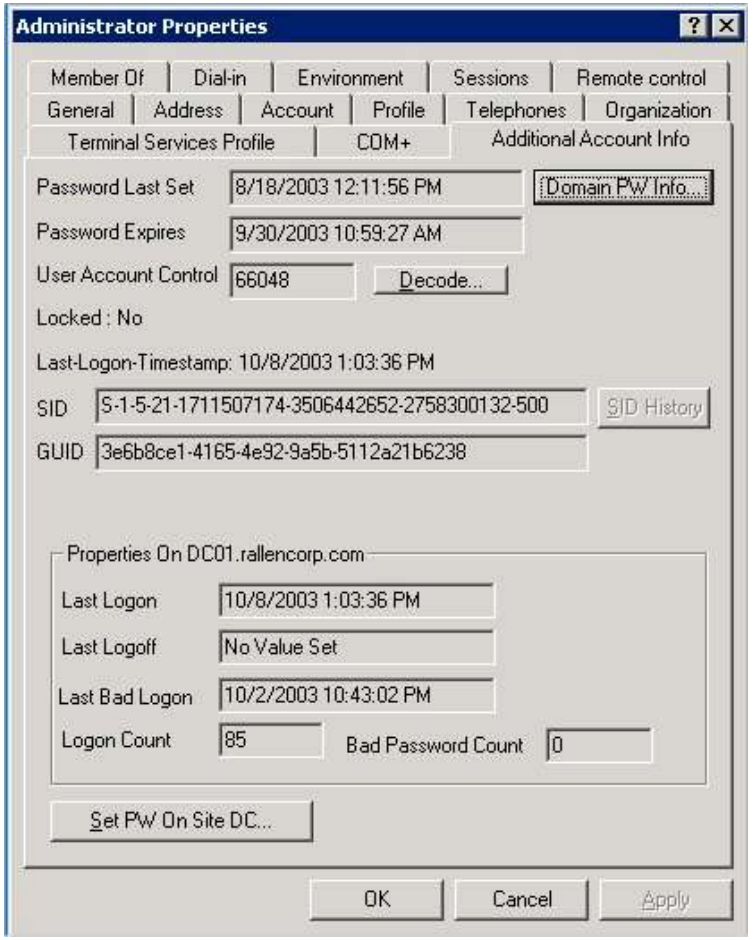


Konsole zarządzania Active Directory

- Active Directory Users and Computers (dsa.msc) zarządzanie użytkownikami, także zapytaniami filtrującymi informacje o użytkownikach i maszynach.
- Active Directory Domains and Trusts (domain.msc) - zarządzanie domeną i uprawnieniami globalnymi w domenie.
- Active Directory Sites and Services (dssite.msc) - zarządzanie usługami w domenie
- Uruchomienie: *.msc /server=*nazwa_serwera.domena*

Narzędzia Active Directory i kontroli domeny (I)

- *AcctInfo Property Page* - dostępne w konsoli Active Directory Users and Computers



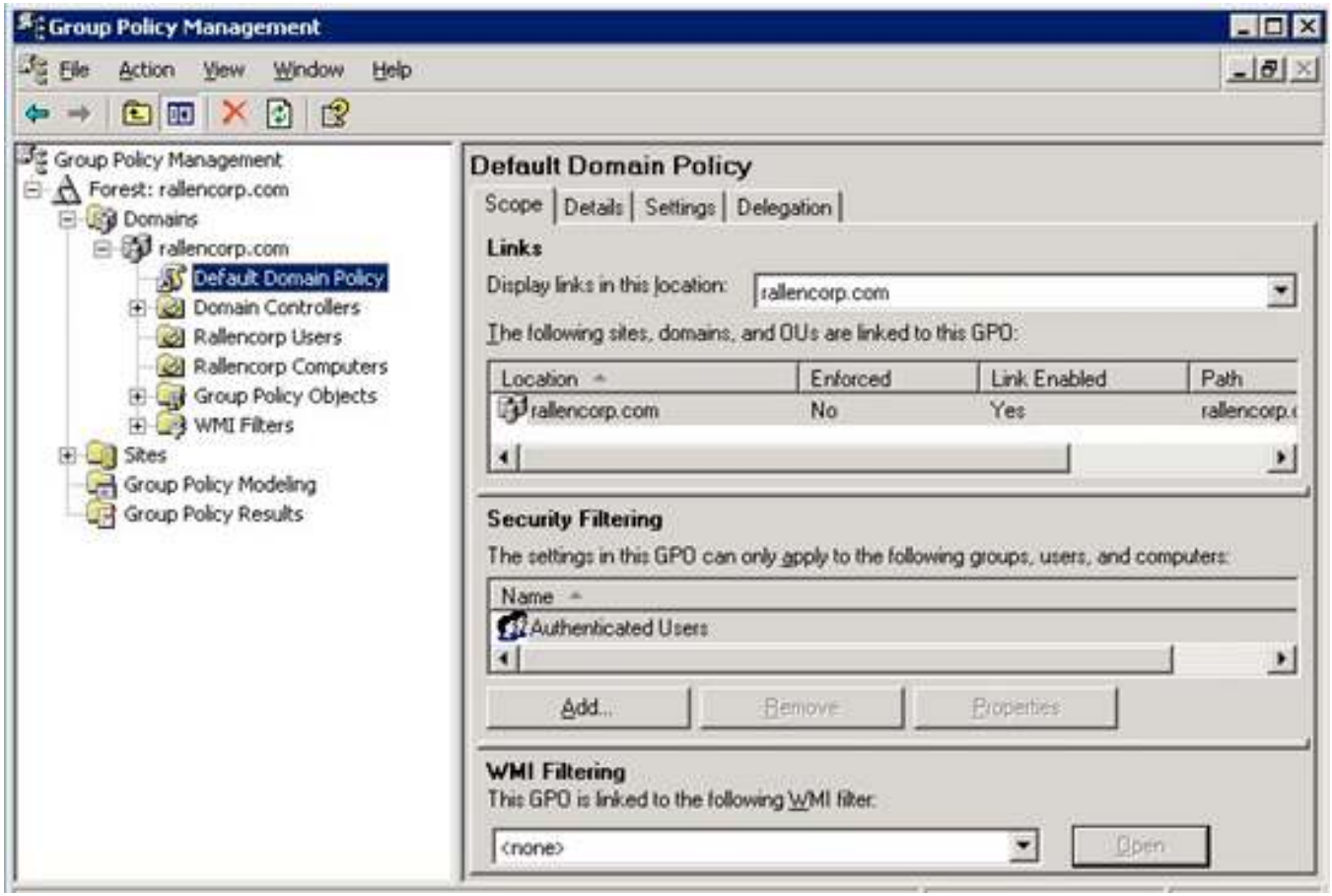
Narzędzia Active Directory i kontroli domeny (II)

- *LockoutStatus.exe* - narzędzie do wyszukiwania kont zablokowanych

- *dsadd.exe, dsget.exe, dsmove.exe, dsquery.exe, dsrm.exe* - narzędzia do modyfikowania składników domeny z linii komend

Narzędzia Active Directory i kontroli domeny (III)

- Konsola Group Policy Management (gpmc.msc)



DNS - Integracja z Active Directory (I)

- **Częściowa integracja -** informacje domenowe przechowywane są w standardowej formie plikowej. Dane DNS zapisywane są w plikach tekstowych o rozszerzeniu .dns. Domyślną lokalizacja dla tych plików jest katalog %SystemRoot%System32/Dns. Aktualizacje danych DNS są obsługiwane przez pojedynczy autorytatywny serwer DNS, wskazany jako serwer podstawowy dla konkretnej domeny lub obszaru w obrębie domeny nazywanego strefą. Klienci korzystający z dynamicznego uaktualniania DNS poprzez DHCP muszą zostać skonfigurowani tak, aby korzystali z podstawowego serwera DNS, gdyż w przeciwnym wypadku dynamiczne aktualizacje nie będą mogły być wykonywane. Podobnie dynamiczne aktualizacje poprzez DHCP nie będą wykonywane, jeśli podstawowy serwer DNS nie będzie dostępny.

DNS - Integracja z Active Directory (II)

- **Pełna integracja -** informacje DNS są zapisywane bezpośrednio w katalogu DNS i są dostępne w kontenerze obiektu *dnsZone*. Ze względu na fakt, że informacje te są częścią katalogu Active Directory, każdy kontroler domeny może uzyskać dostęp do tych danych. Można też wykorzystać model o wielu wzorcach do dynamicznego uaktualniania wpisów DNS poprzez DHCP. Umożliwia to każdemu kontrolerowi domeny na którym uruchomiony jest serwer DNS, obsługę dynamicznych aktualizacji. Dalej, każdy klient DHCP używający aktualizacji dynamicznej może skorzystać z dowolnego serwera DNS w danej strefie. Dodatkową korzyścią pełnej integracji jest możliwość wykorzystania mechanizmów zabezpieczeń katalogu do ochrony danych i sterowania dostępem do informacji DNS.

DNS - Zalety integracji z Active Directory

- Replikacja danych Active Directory jest szybsza i wydajniejsza niż standardowa replikacja systemu DNS - przesyłane są tylko istotne zmiany. Rozdzielenie DNS i Active Directory zwiększa obciążenie sieci i wydłuża czas potrzebny do rozpropagowania zmian DNS w całej sieci.
- Usprawnione planowanie replikacji bazy danych w sieci.
- Po dodaniu nowego kontrolera do domeny usługi Active Directory strefy są automatycznie replikowane i synchronizowane z takim kontrolerem.

Podział serwerów DNS (I)

- Typy serwerów DNS:
  - **Serwer podstawowy zintegrowany z Active Directory** - w pełni zintegrowany z usługą Active Directory. Wszystkie dane DNS przechowywane są w katalogu Active Directory.
  - **Serwer podstawowy** - Główny serwer DNS dla domeny, wykorzystujący częściową integrację z Active Direcory. Serwer ten przechowuje główną kopię rekordów DNS i pliki konfiguracyjne domeny - w postaci tekstowej w plikach z rozszerzeniem .dns.

Podział serwerów DNS (II)

- Typy serwerów DNS (cd):
  - **Serwer pomocniczy** - zapewniający dodatkową usługę DNS dla domeny. Przechowuje kopię rekordów DNS otrzymaną z preferowanego serwera DNS, wykonując aktualizację za pomocą transferu stref. Serwery pomocnicze uzyskują informacje z serwera preferowanego i przechowują ją do momentu jej odświeżenia lub wygaśnięcia.
  - **Serwer przesyłania dalej** - buforujący informacje DNS, zawsze przekazujący zapytania do innych serwerów. W odróżnieniu od serwerów pomocniczych, serwery takie nie przechowują pełnych kopii danych dla strefy.

Instalacja DNS (I)

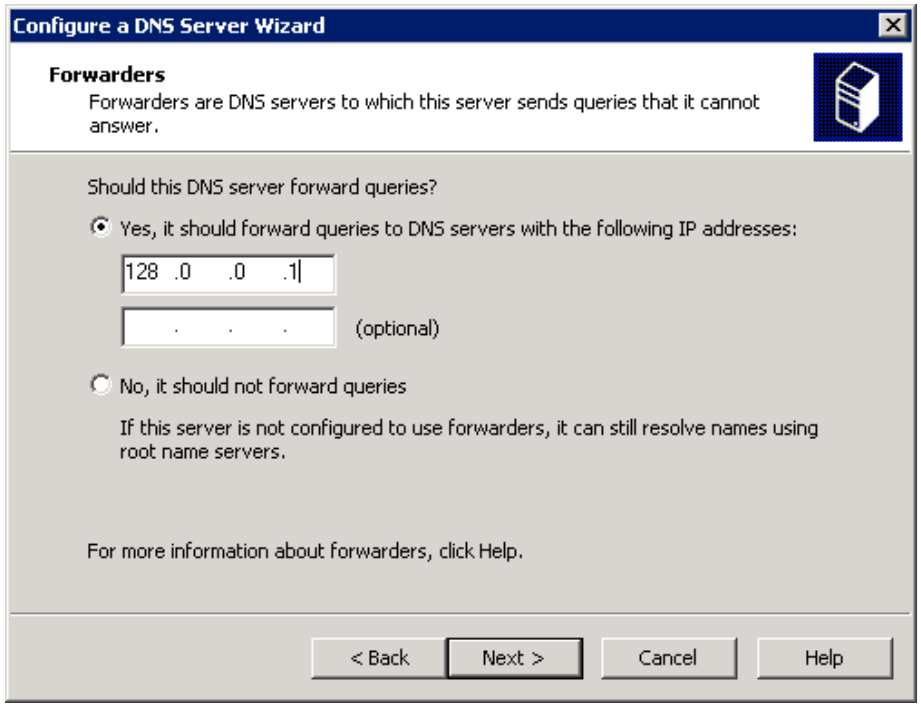
- Możliwość synchronizowania DNS z usługami Active Directory.
- Serwer nazw posiada komplet wpisów identyfikujących maszyny w obrębie domeny.
- Zapytania dotyczące innych maszyn kieruje do domeny nadrzędnej lub innego wytypowanego do tego celu serwera nazw domen.

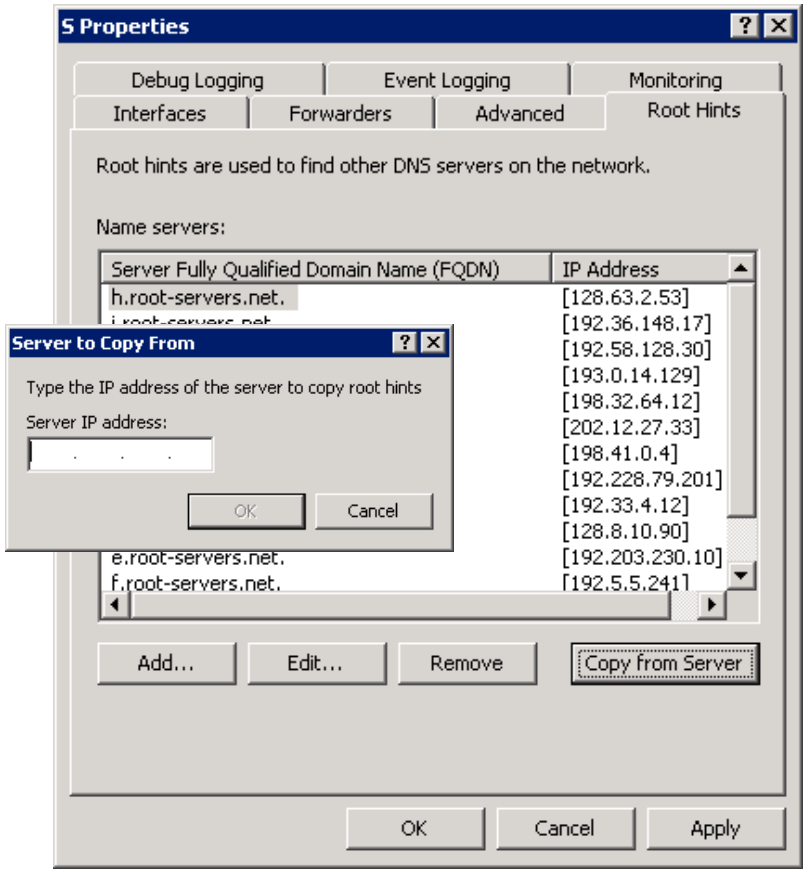
Instalacja DNS (II)

- Z narzędzi administracyjnych należy wybrać *Kreator konfigurowania serwera*. Wybrać rolę *serwer DNS*.
- Serwery podstawowe powinny zawierać strefy wyszukiwania do przodu oraz strefy wyszukiwania wstecznego dla danej domeny
- Przypomnienie: Wyszukiwanie do przodu umożliwia zmianę nazw domen na adresy IP. Wyszukiwanie wsteczne pozwala potwierdzić zapytania DNS poprzez znalezienie nazw odpowiadających podanym adresom IP.

Instalacja DNS (III)

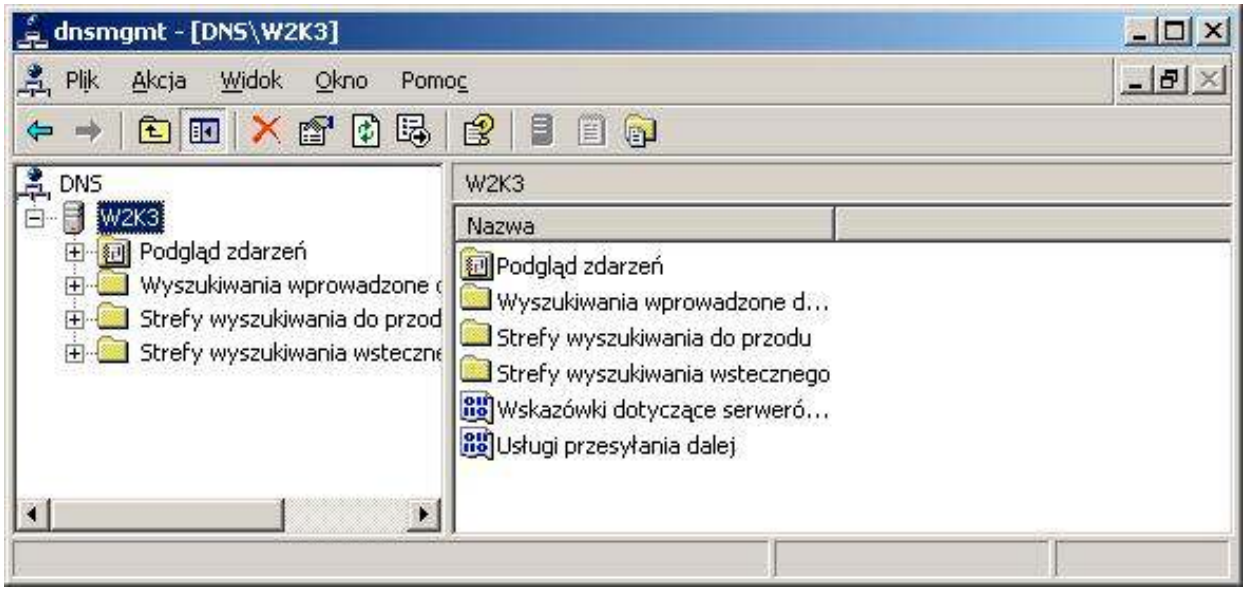
- Nowa strefa - przekierowanie dalej
- Opcje wyszukiwania





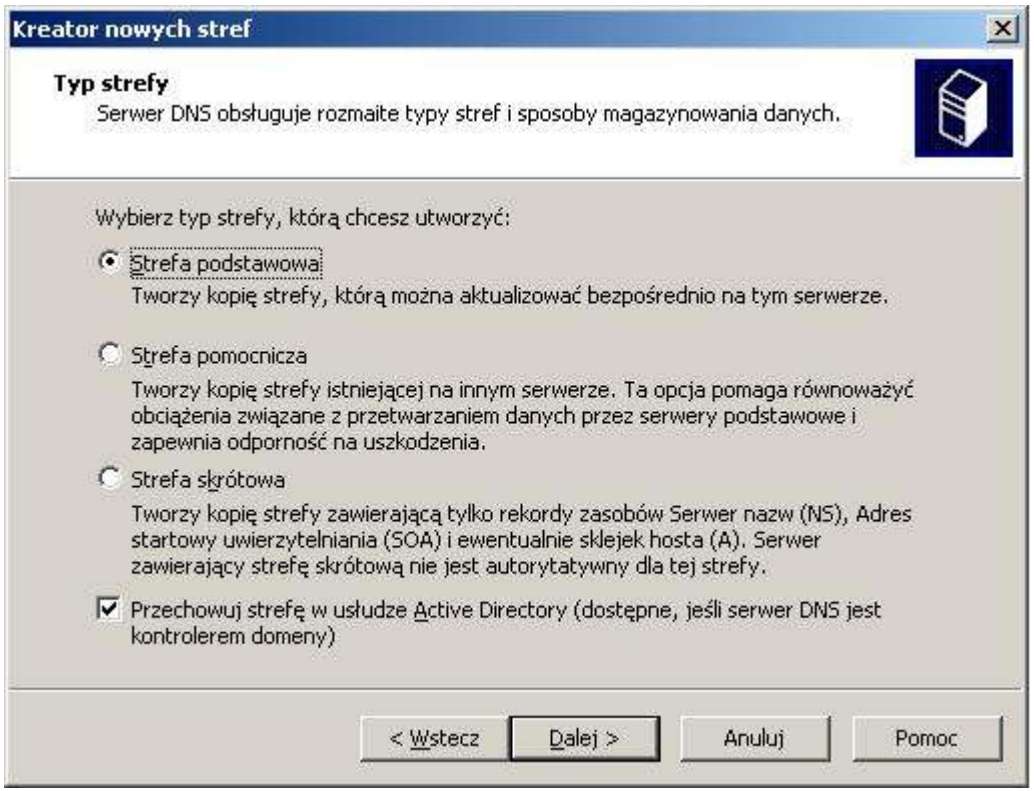
Konsola DNS

- W systemach typu server - zawarta jako oddzielny aplet w Narzędziach Administracyjnych lub jako gałąź Usługi i aplikacje konsoli Zarządzanie komputerem



DNS
- tworzenie nowej strefy (I)

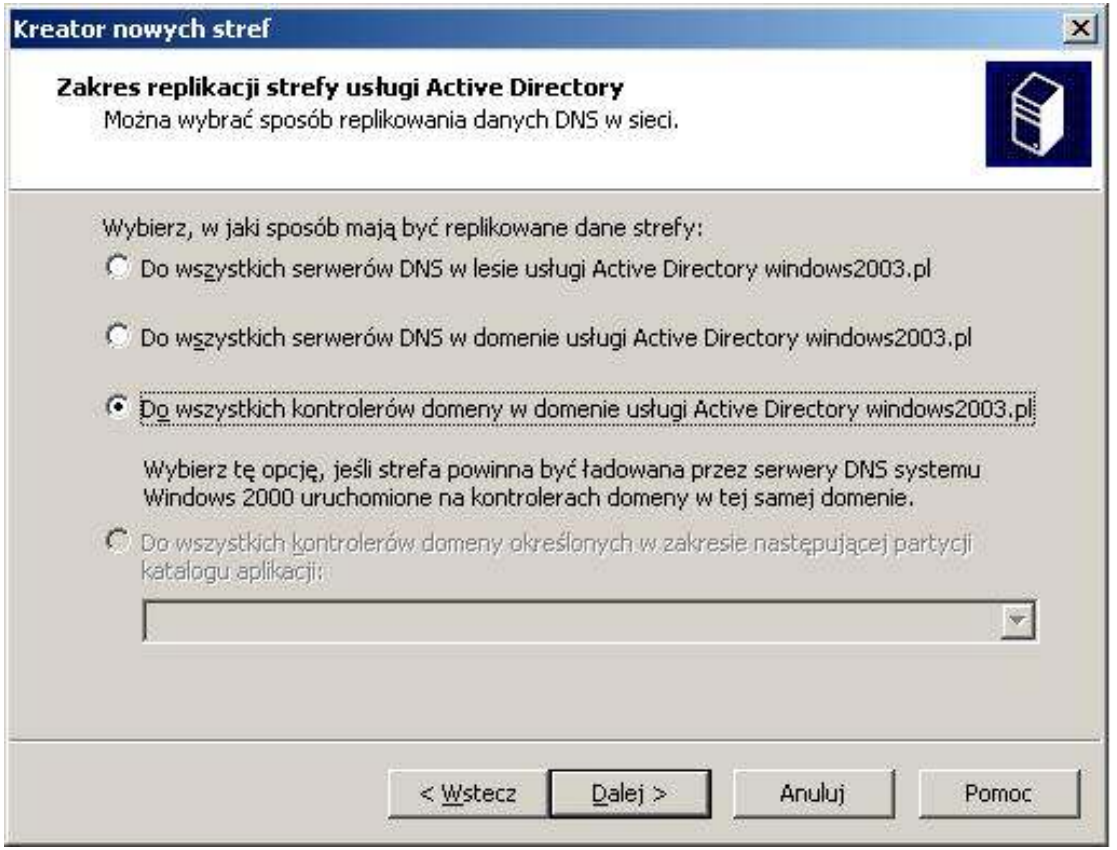
- Strefy są tworzone oddzielnie dla wyszukiwania do przodu i wstecznego.



DNS
- tworzenie nowej strefy (II)

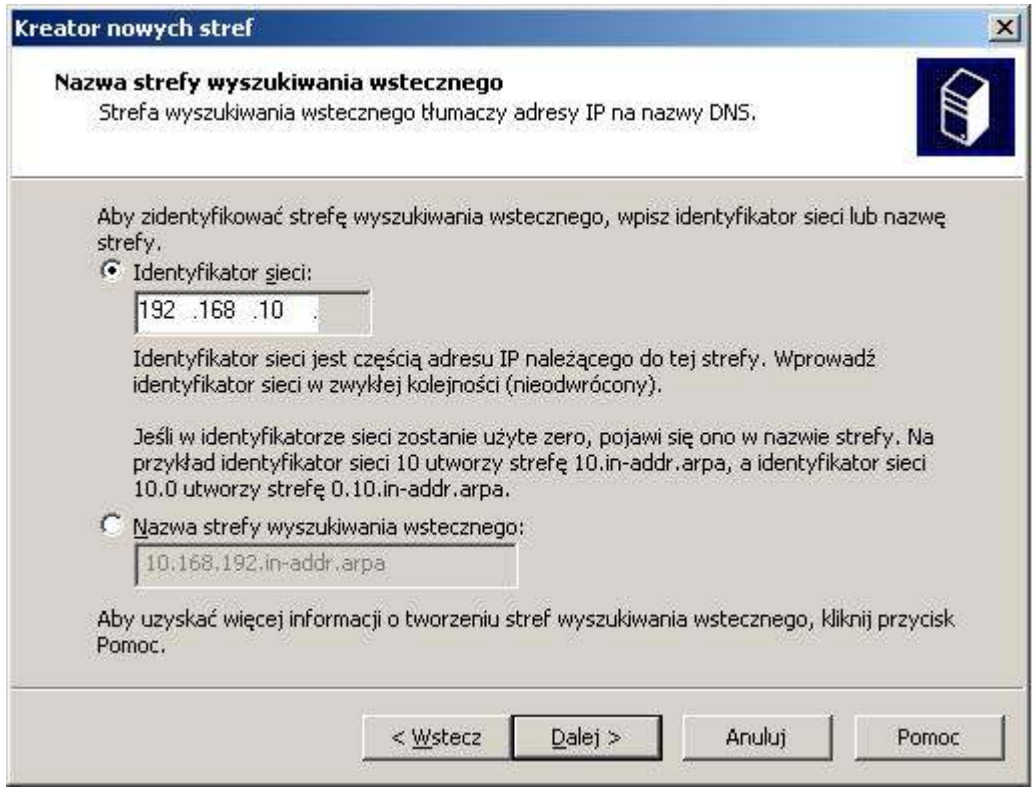
- Replikacja danych w obrębie Active Directory występuje w trzech trybach:
  - *Do wszystkich serwerów DNS w lesie domen Active Directory*. Opcja ta powoduje najszersze replikowanie danych DNS. Należy pamiętać, że las domen tworzą wszystkie domeny Active Directory mające wspólne dane katalogowe z aktualną domeną.
  - *Do wszystkich serwerów DNS w domenie usługi Active Directory*. Opcja ta powoduje replikowanie informacji DNS tylko w obrębie bieżącej domeny Active Directory i jej poddomen.
  - *Do wszystkich kontrolerów domeny w domenie usługi Active Directory*. Strategia ta powoduje replikowanie danych DNS do wszystkich kontrolerów domeny w domenie bieżącej i domenach podrzędnych.

DNS
- tworzenie nowej strefy (III)



DNS
- tworzenie nowej strefy (IV)

- Strefa adresacji i sufiksy DNS



DNS
- rodzaje rekordów DNS (I)

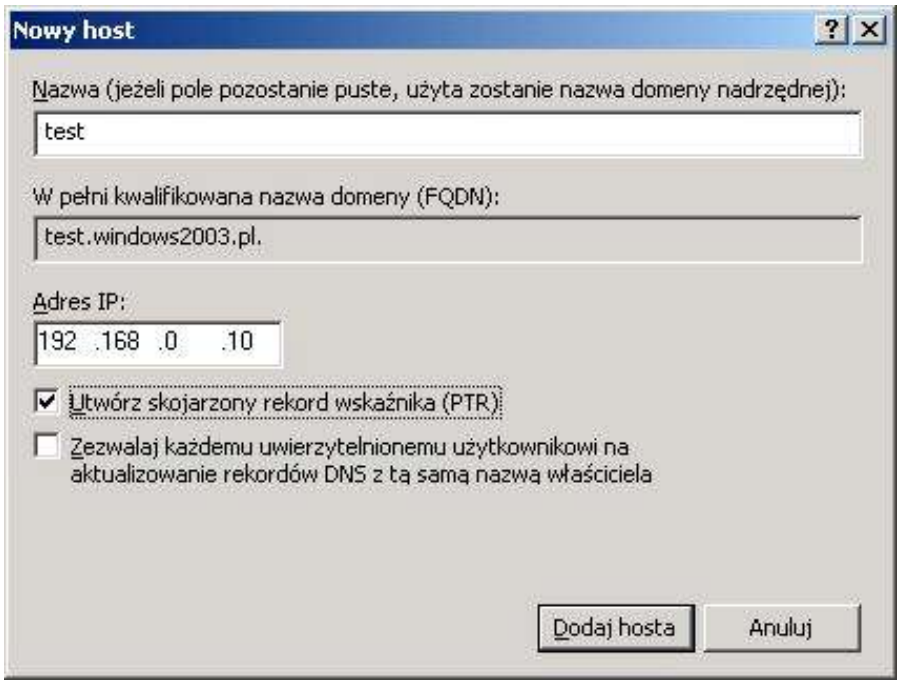
- A (adres). Rekord przypisujący nazwę hosta do adresu IP. W przypadku komputera posiadającego kilka kart sieciowych lub adresów IP należy tworzyć rekord adresu dla każdego z nich.
- CNAME (nazwa kanoniczna). Definiuje alias dla nazwy hosta. Przy użyciu tego rekordu komputer może być identyfikowany pod inna nazwą, np. host *alfa.microsoft.com* może posiadać alias *www.microsoft.com*.
- MX (usługa wymiany poczty). Umożliwia zdefiniowanie serwera wymiany poczty dla domeny. Dzięki temu rekordowi poczta elektroniczna będzie dostarczana do właściwego serwera pocztowego w domenie.

DNS
- rodzaje rekordów DNS (II)

- NS (serwer nazw). Wskazuje na serwer nazw dla domeny, umożliwiając wyszukiwanie nazw DNS w różnych strefach.
- PTR (wskaźnik). Tworzy powiązanie pomiędzy adresem IP i nazwą hosta w strefach wyszukiwania wstecznego.
- SOA (adres startowy uwierzytelniania). Deklaruje host, który jest najbardziej autorytatywny dla danej strefy i tym samym jest najlepszym źródłem informacji DNS dla tej strefy. Każda strefa musi posiadać rekord SOA, który jest generowany automatycznie w momencie utworzenia strefy.

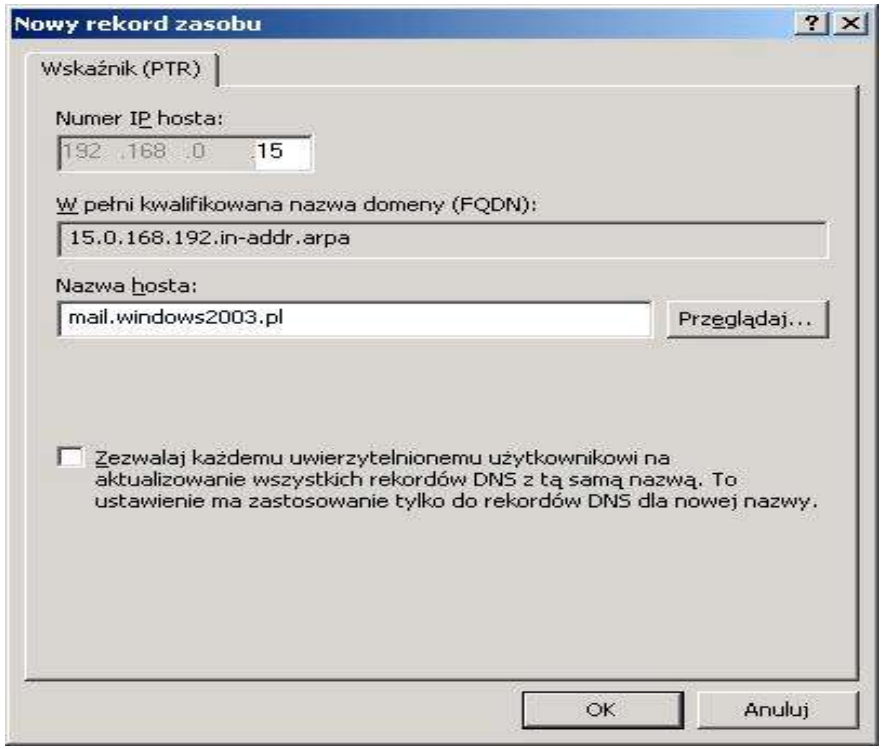
DNS – dodawanie rekordów adresowych

- Tworzenie w konsoli MMC - Strefy wyszukiwania do przodu dla wybranego serwera -> Nowy host z menu podręcznego



DNS – dodawanie rekordów przeszukiwania wstecznego

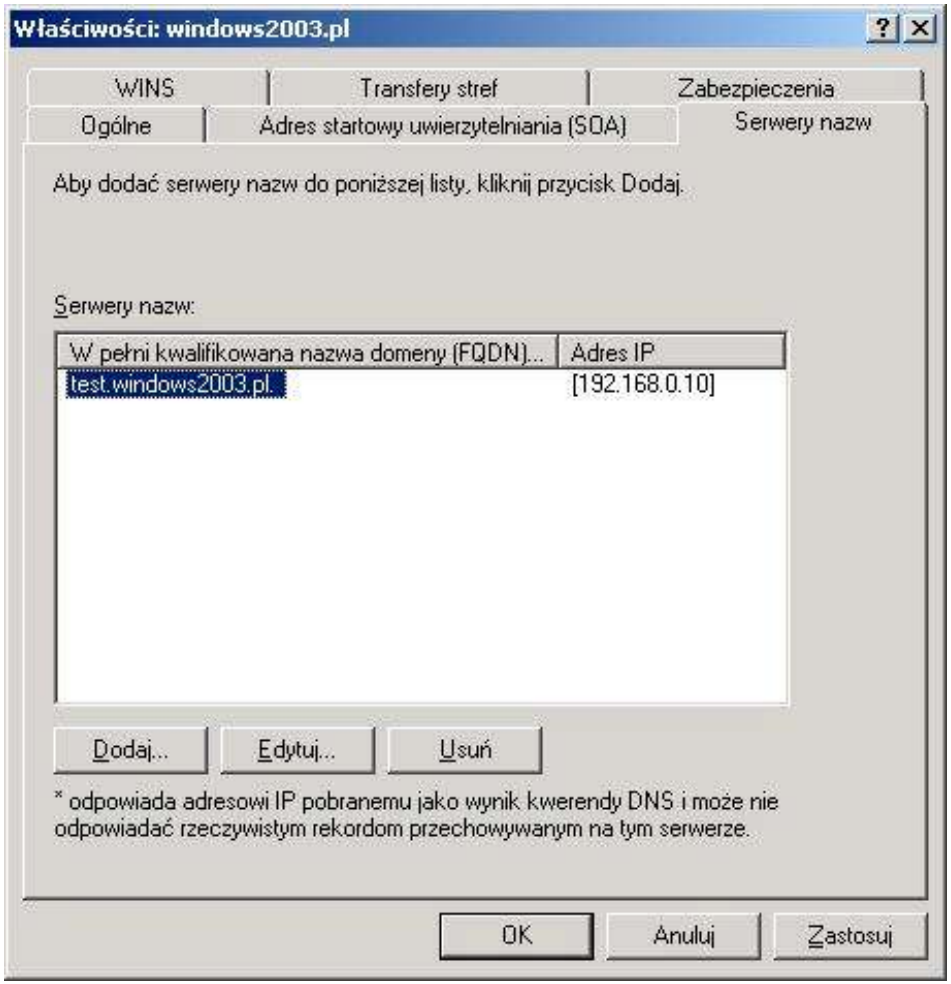
- Rekord PTR trzeba dodać obowiązkowo dla każdego hosta.
- Tworzenie w konsoli MMC - Strefy wyszukiwania wstecznego dla wybranego serwera -> Nowy wskaźnik z menu podręcznego



DNS – dodawanie serwerów nazw (I)

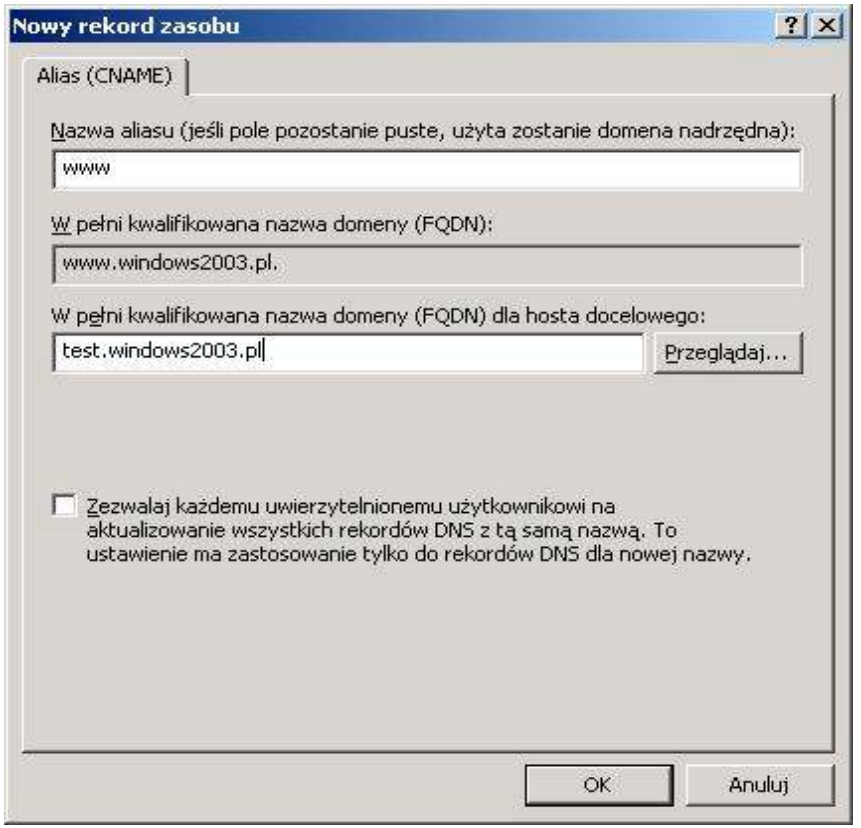
- Każdemu podstawowemu i pomocniczemu serwerowi DNS musi odpowiadać rekord NS zawierający odpowiednia deklarację
- Ustawienie: Strefy wyszukiwania do przodu -> rekordy DNS dla domeny -> Serwer nazw (NS)

DNS – dodawanie serwerów nazw (II)



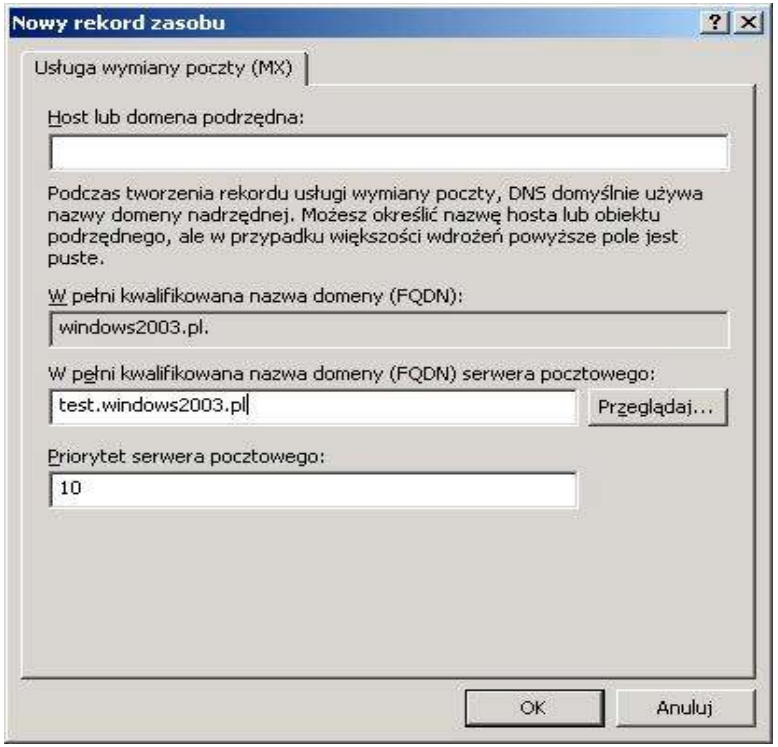
DNS
- Tworzenie aliasów nazw

- Tworzenie w konsoli MMC - strefy wyszukiwania do przodu dla wybranego serwera -> *Nowy alias (CNAME)* z menu podręcznego



DNS – dodawanie serwerów wymiany poczty

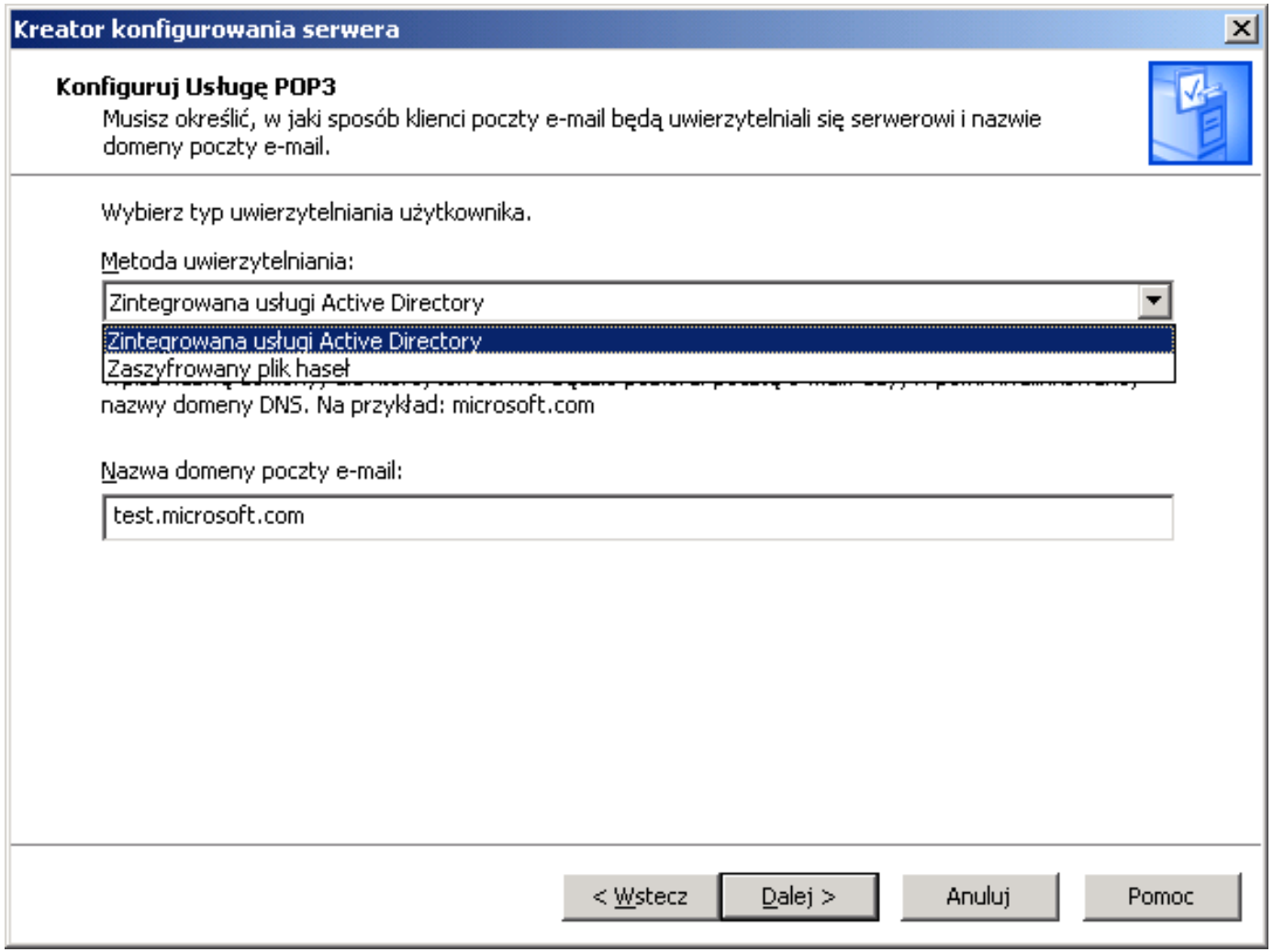
- Serwery poczty funkcjonują w obrębie domeny
- Liczbowe priorytety dla dostaw poczty (0 65535, niższa wartość ma pierwszeństwo)



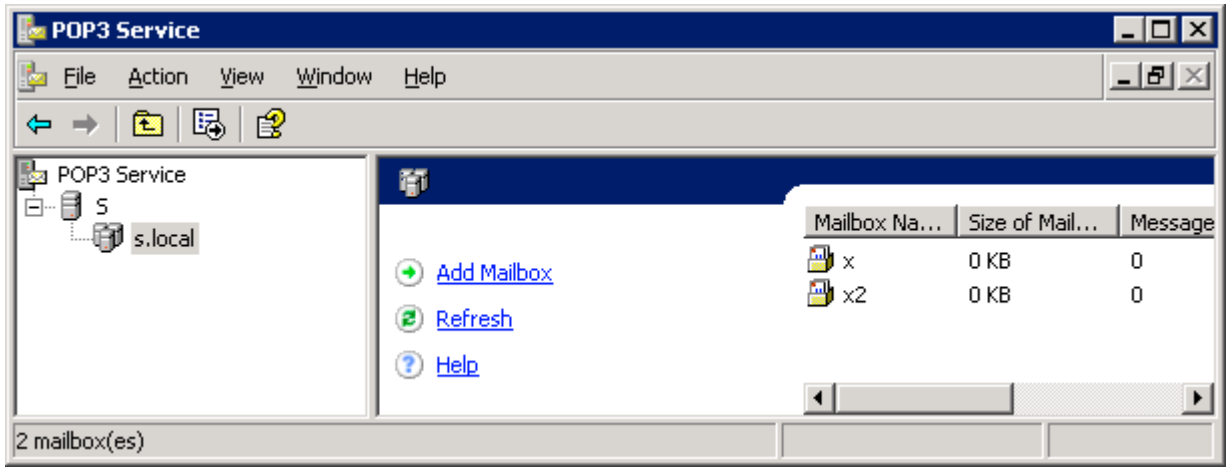
Serwer poczty

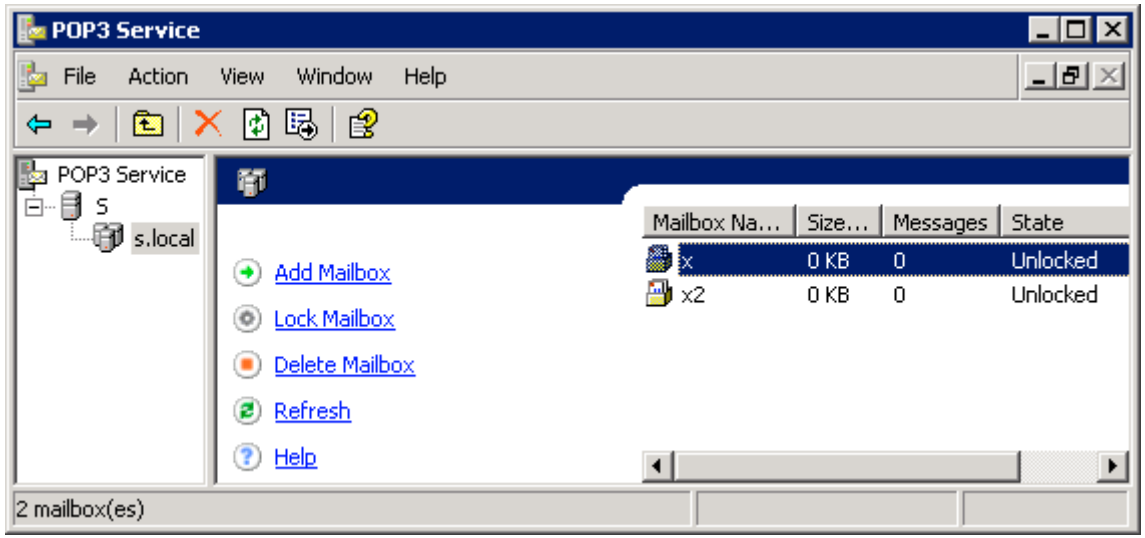
- Przechowywanie poczty w skrzynkach i udostępnianie na żądanie (POP3)
- Konieczność konfigurowania rekordu MX DNS dla serwera SMTP
- Dwa opcjonalne składy kont pocztowych - uwierzytelnianie za pośrednictwem Active Directory lub indywidualnie - w „szyfrowanym pliku z hasłami”. Druga możliwość pozwala na definiowanie użytkowników spoza domeny.
- Katalog skrzynek pocztowych \Inetpub\mailroot\Mailbox\nazwa_domeny\nazwa_użytkownika.mbx
- Bardziej zaawansowane rozwiązanie - produkt Microsoft Exchange Server

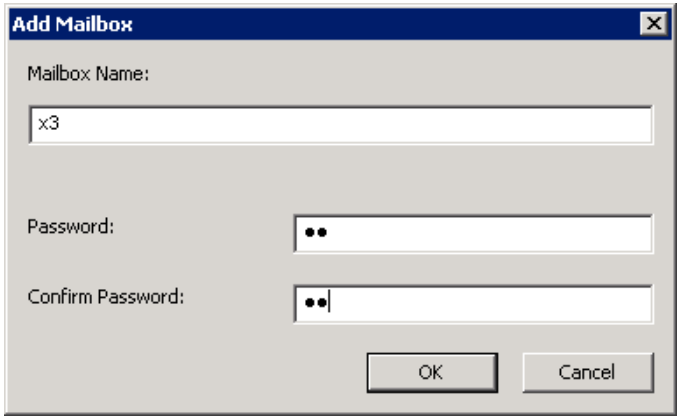
Instalowanie serwera poczty



Konfigurowanie serwera poczty (I)

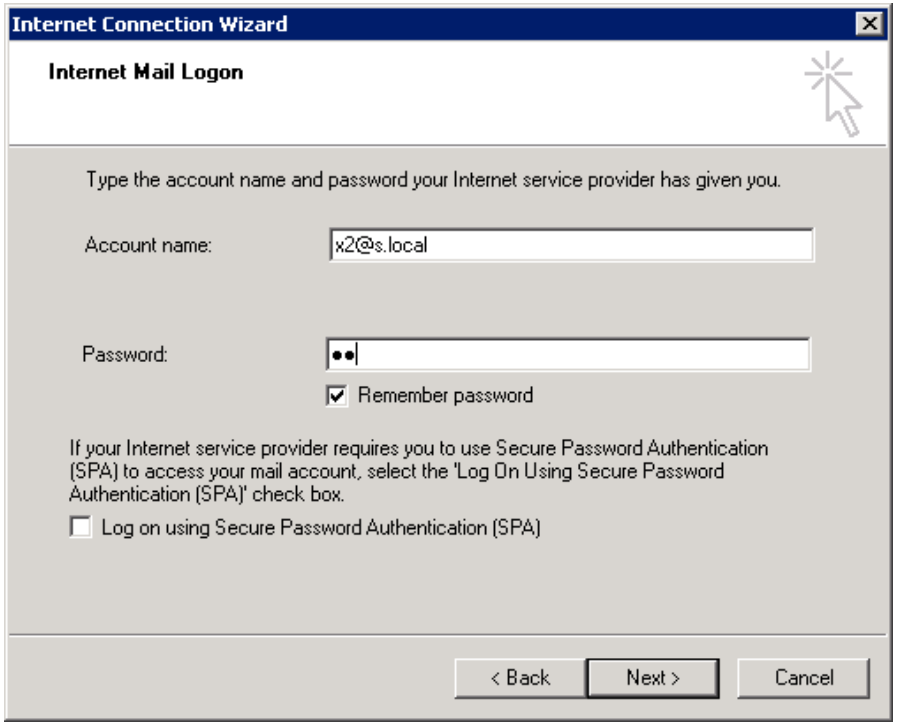


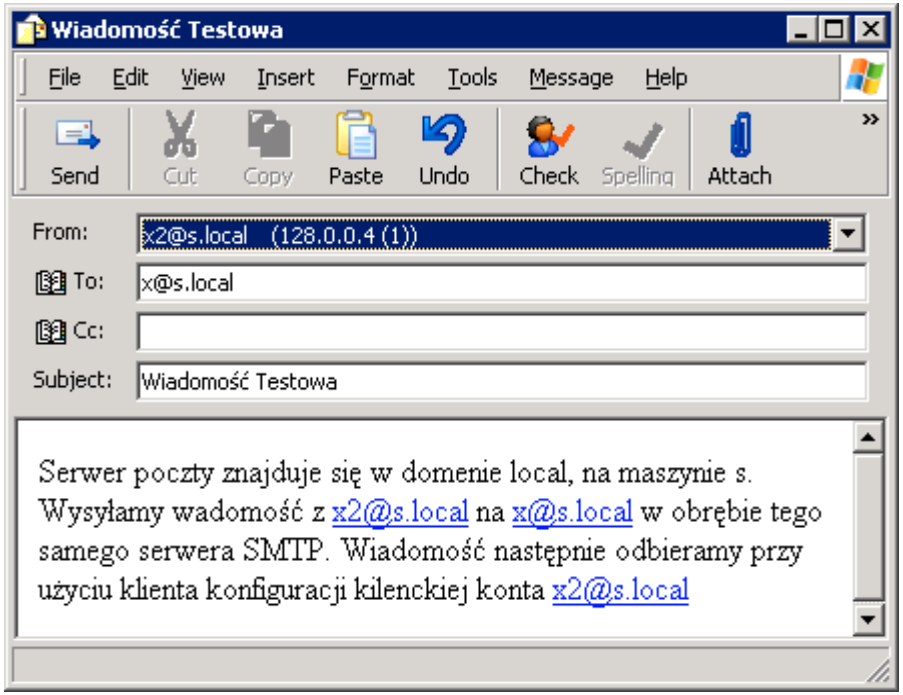




Konfigurowanie serwera poczty (II)

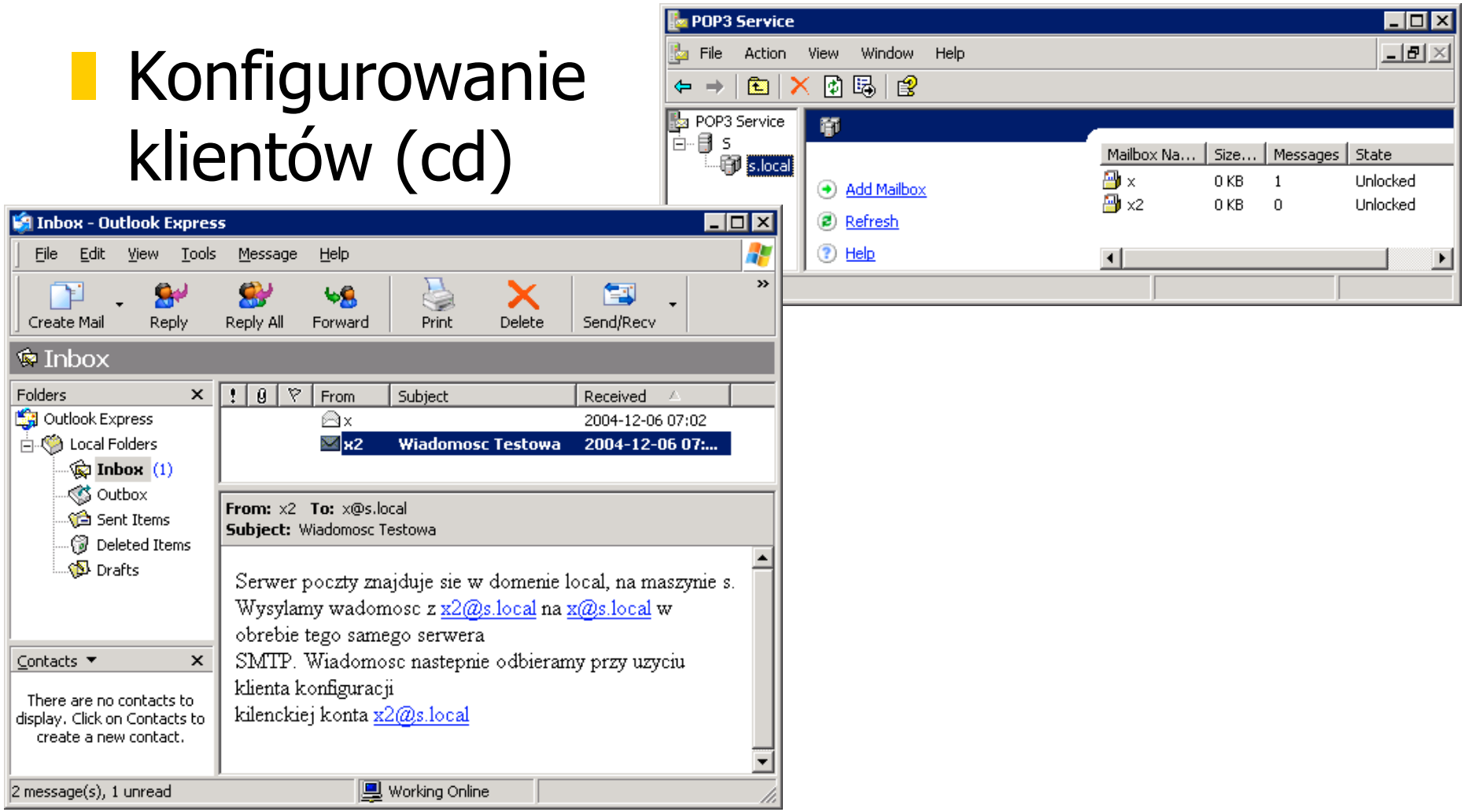
- Konfigurowanie klientów





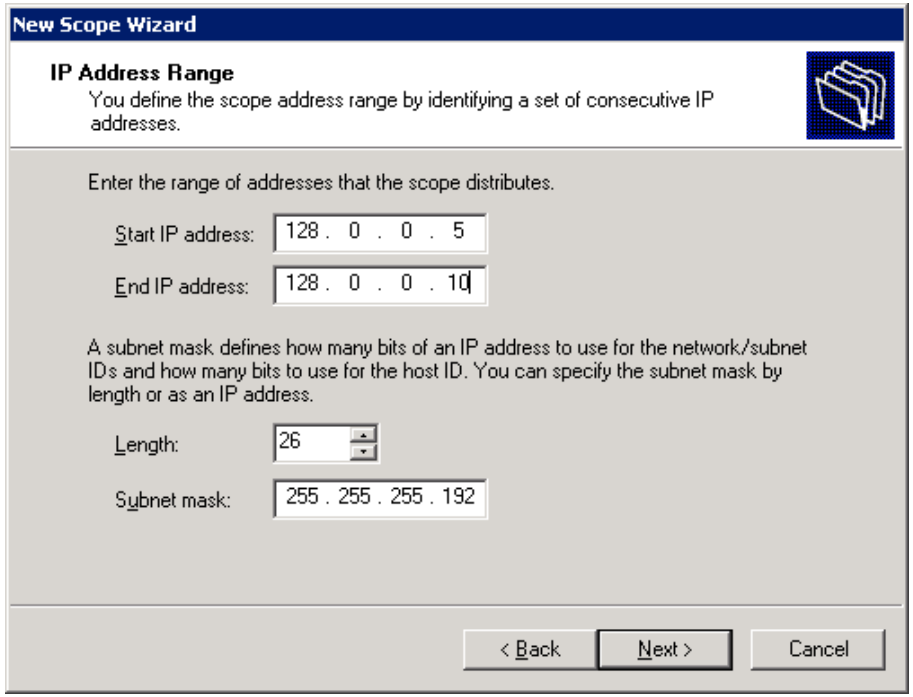
Konfigurowanie serwera poczty (III)

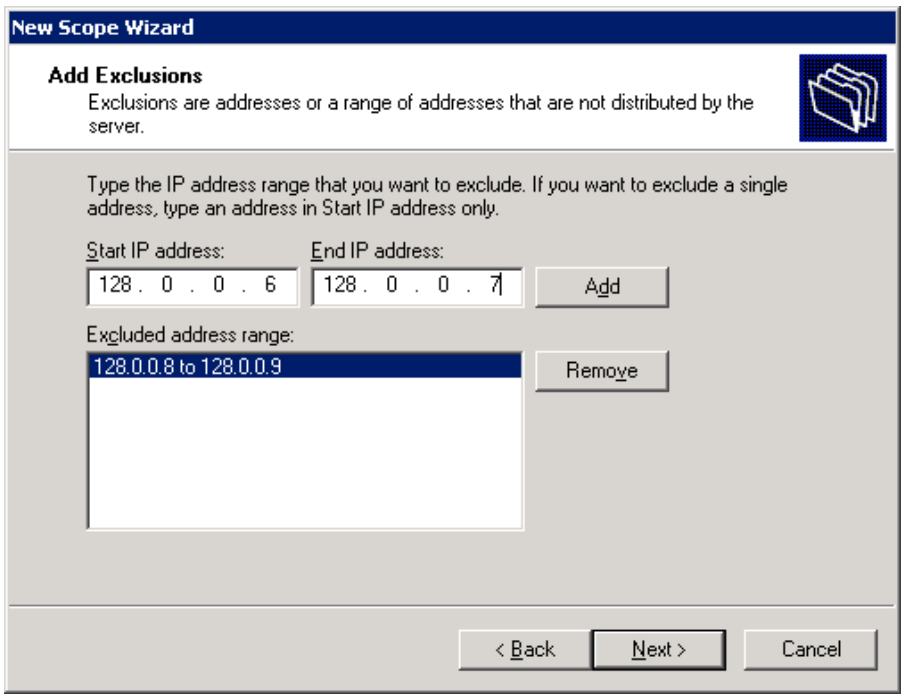
- Konfigurowanie klientów (cd)



Konfigurowanie DHCP (I)

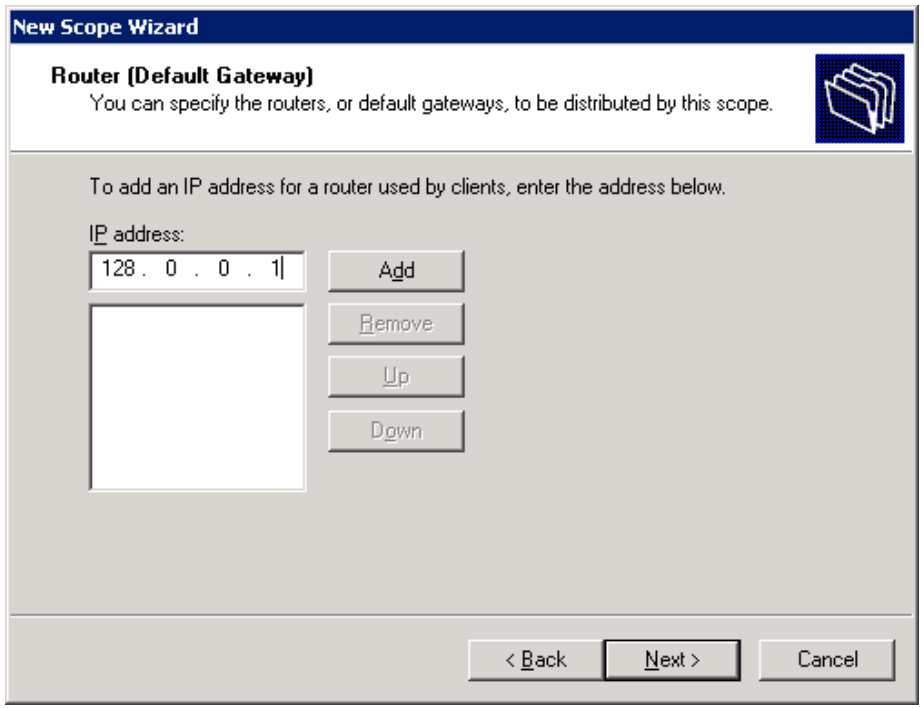
- Zakres adresów, adresy wyłączone

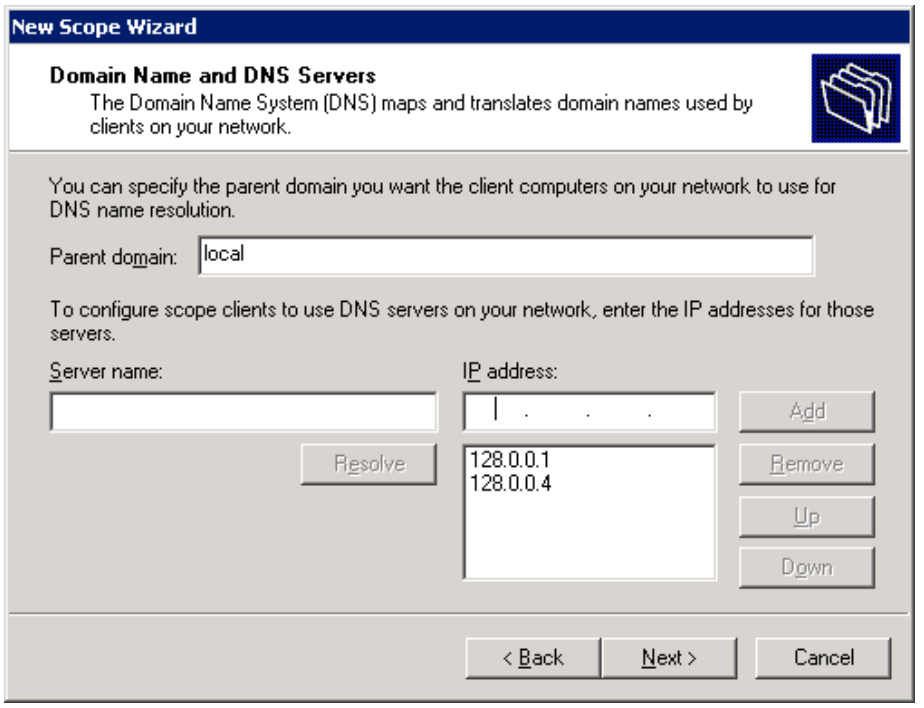




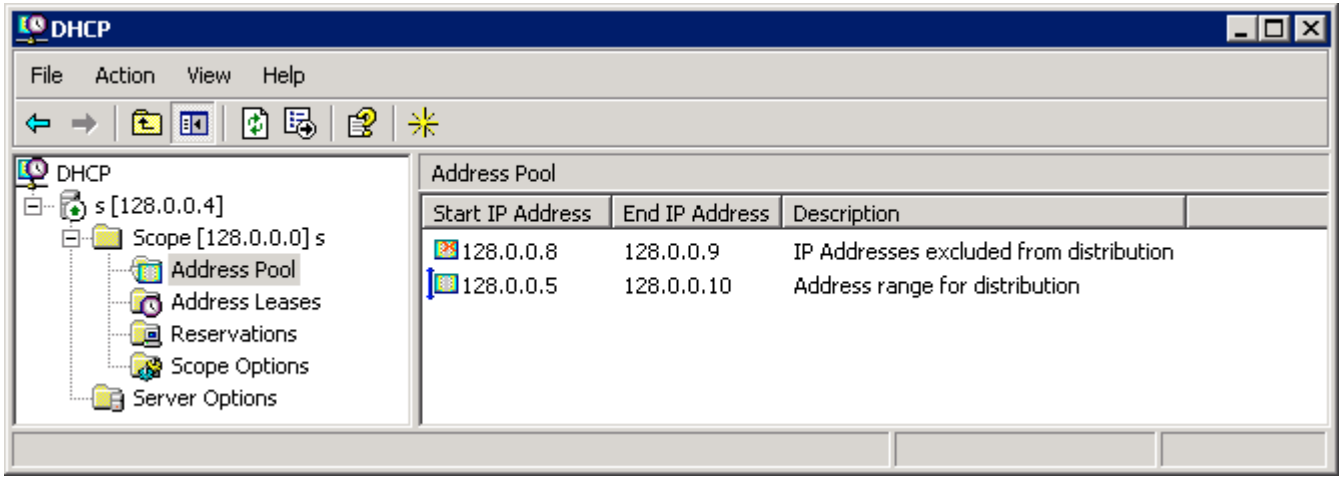
Konfigurowanie DHCP (II)

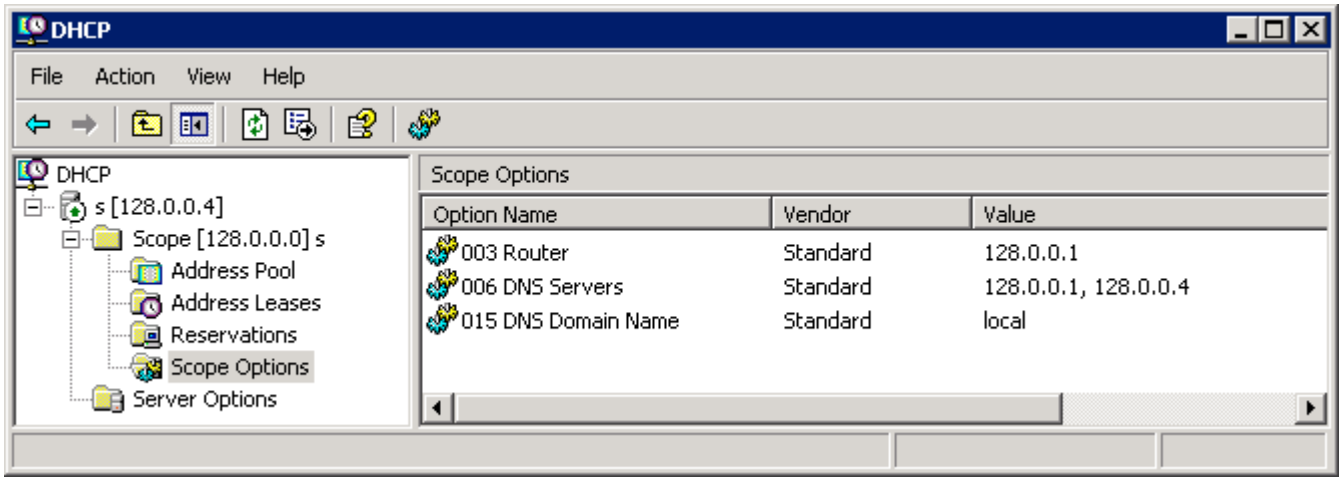
- Adresy bramek, DNS





Konfigurowanie DHCP (III)



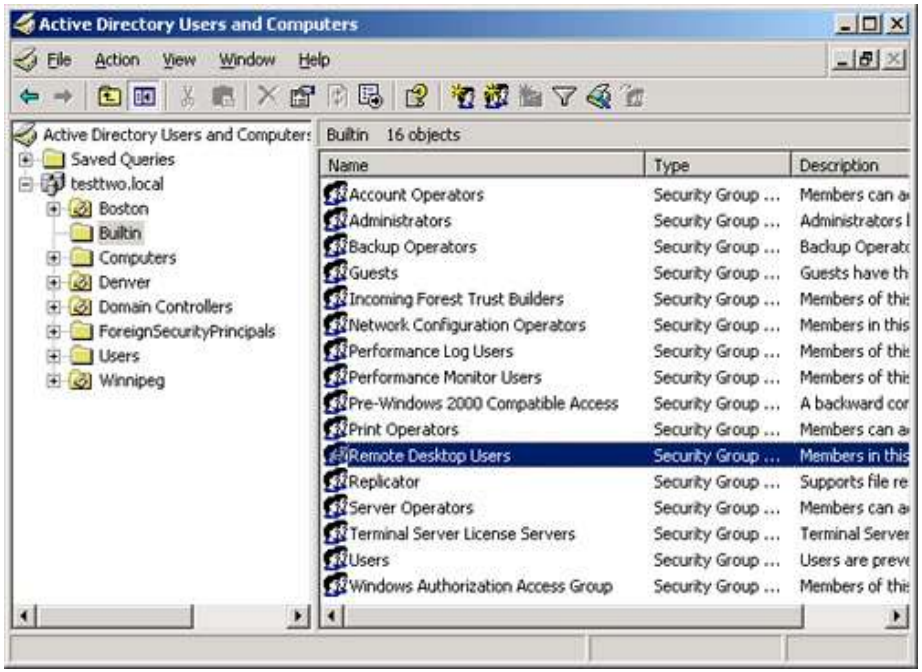


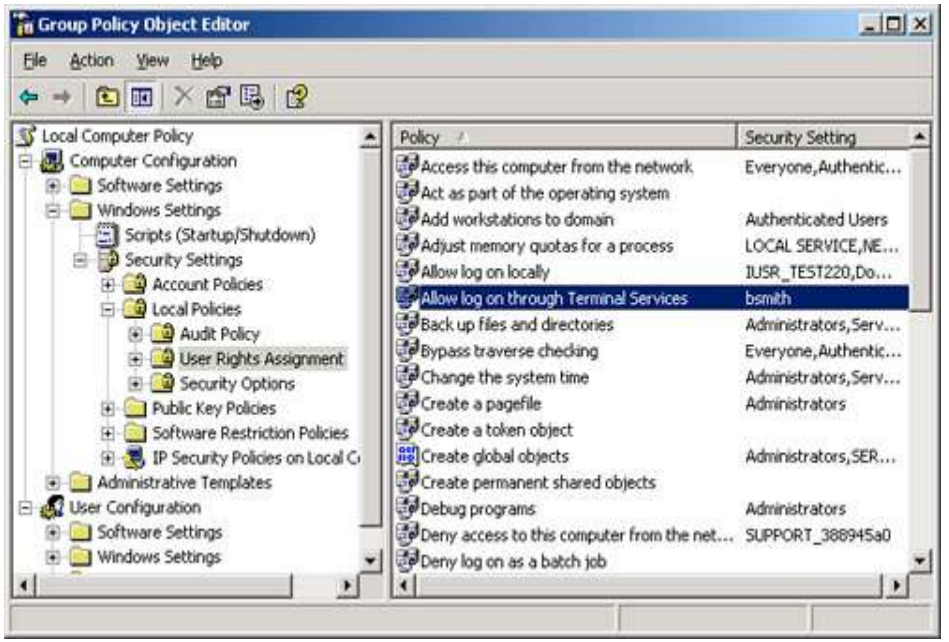
Konfigurowanie usług terminalowych (I)

- Użytkownicy - dwa tryby nadawania praw do korzystania z usług terminalowych;
  - Dodanie do Remote Desktop User Group,
  - Przypisanie użytkownikowi (lub grupie) praw do logowania do serwera terminali (w Security Policy Editor).

Konfigurowanie usług terminalowych (II)

- Computer Configuration\Windows Settings\Security Settings\Local Policies\User Rights Assignment\Allow log on through Terminal Services



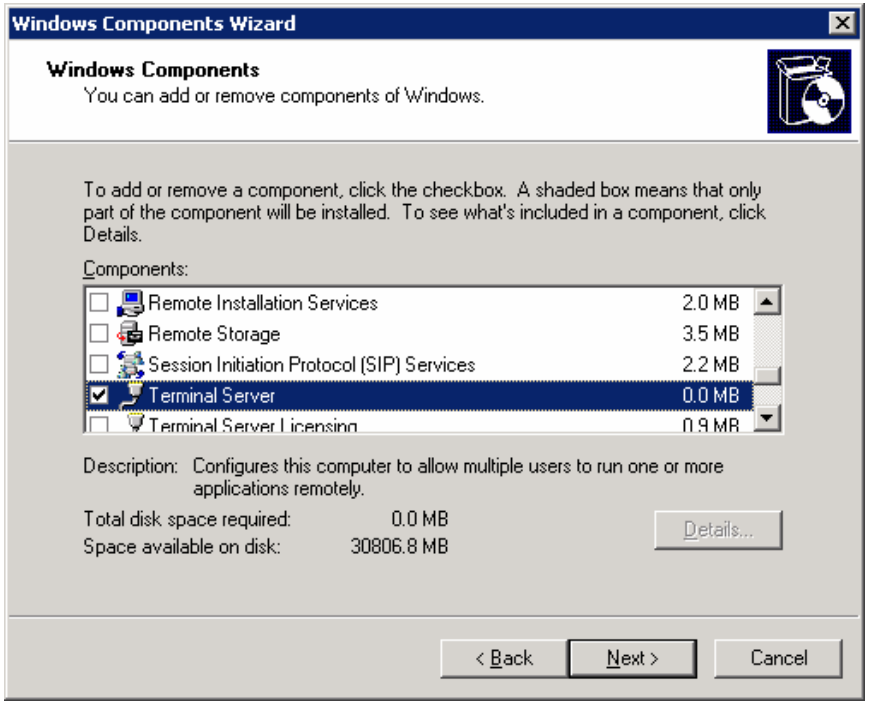


Konfigurowanie usług terminalowych (III)

- Instalowanie aplikacji:
  - standardowe (automatyczne) : przy użyciu apletu Dodawanie lub usuwanie programów
  - zaawansowane, z wiersza komend (np. wiele procesów instalacyjnych):
    - **change user /install**
    - instalacja
    - **change user /execute**
- Uwaga: instalację oprogramowania należy przeprowadzić po dodaniu roli serwera terminali

Konfigurowanie usług terminalowych (IV)

- W Windows 2003 Server rozdzielono funkcje administracji zdalnej i Usług terminalowych w postaci oddzielnych składników, które można konfigurować niezależnie (Manager Usług Terminalowych dla sesji zwykłych użytkowników i Pulpit zdalny dla usługi „Zdalny Pulpit dla administracji”.
- Instalacja - dodanie składnika “Serwer terminali”

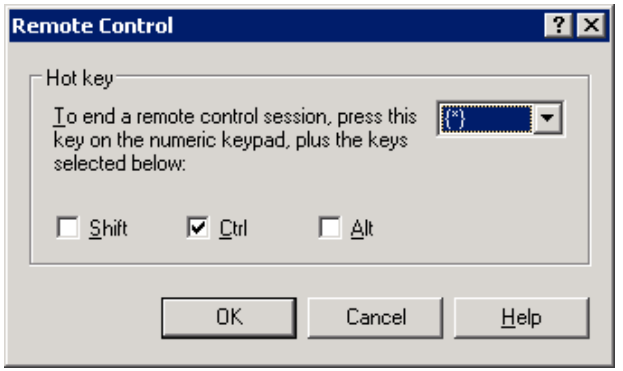


Usługi terminalowe - połączenia administracyjne

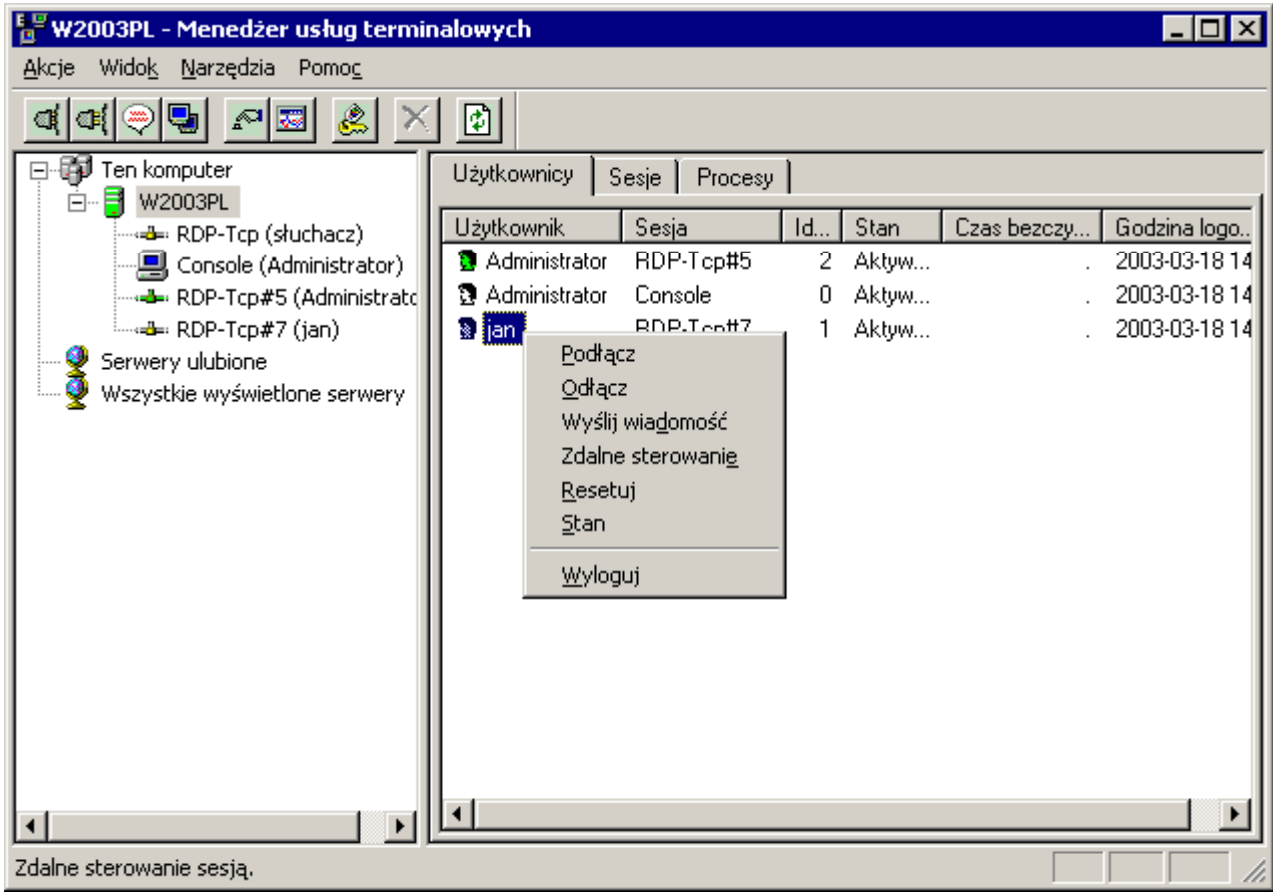
- Wiele równoczesnych sesji terminali graficznych
- Sesja wyróżniona - tzw. zerowa:
  - zdalnie uruchomiana sesja z parametrem /*console*,
  - uruchomiona przy pomocy appletu na stronie Web (Remote Desktop Web Connection) z włączonym parametrem ConnectToServerConsole
- Liczbowe identyfikatory sesji. ID=0 to sesja administratora

Manager usług terminalowych (I)

- Konsola MMC (uruchomiona lokalnie na maszynie serwera):
  - Łączenie z dowolnymi sesjami - przejęcie sesji od użytkownika
  - Reset sesji
  - Praca z trybie zdalnego sterowania (hot-key zakończenia, np CTRL-*)
  - Likwidacja sesji

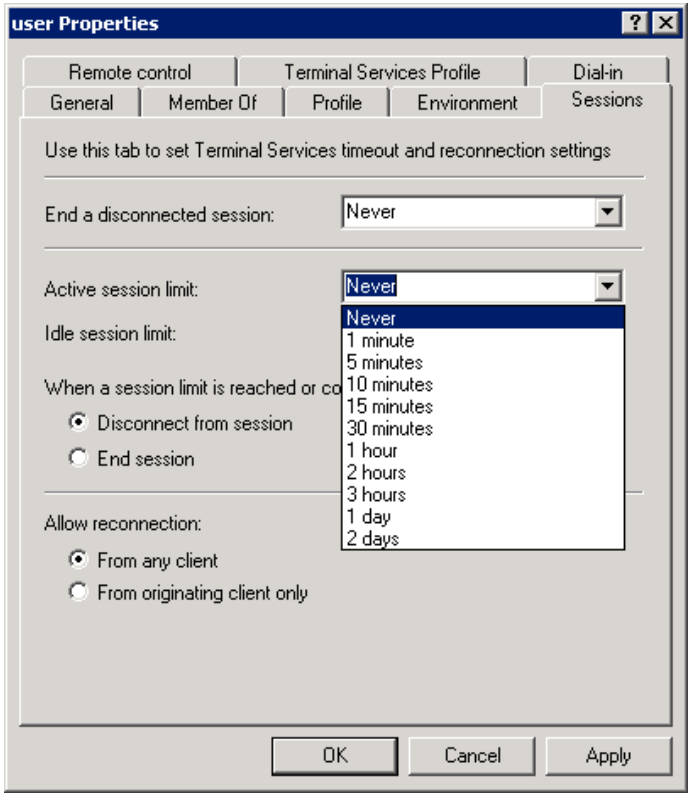


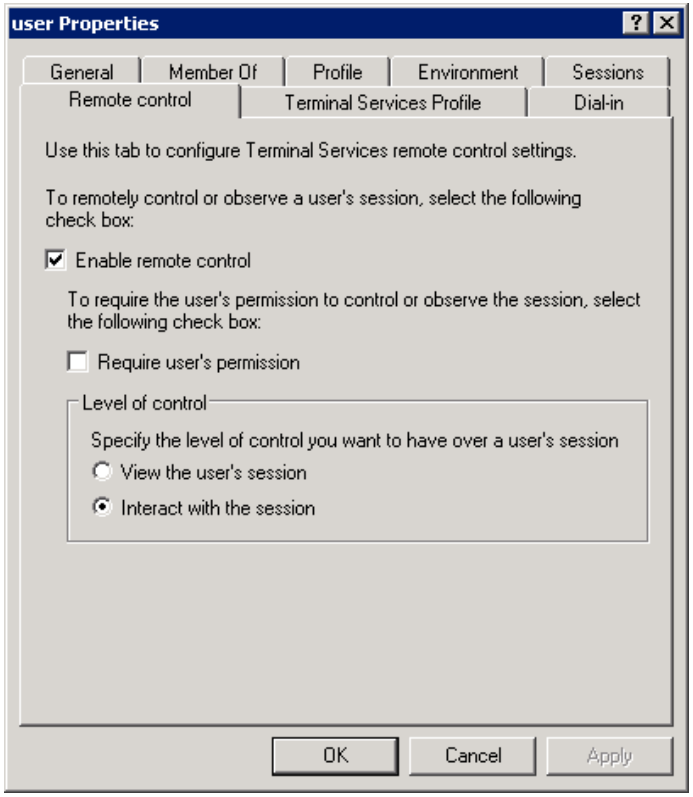
Manager usług terminalowych (II)



Usługi terminalowe - użytkownicy (I)

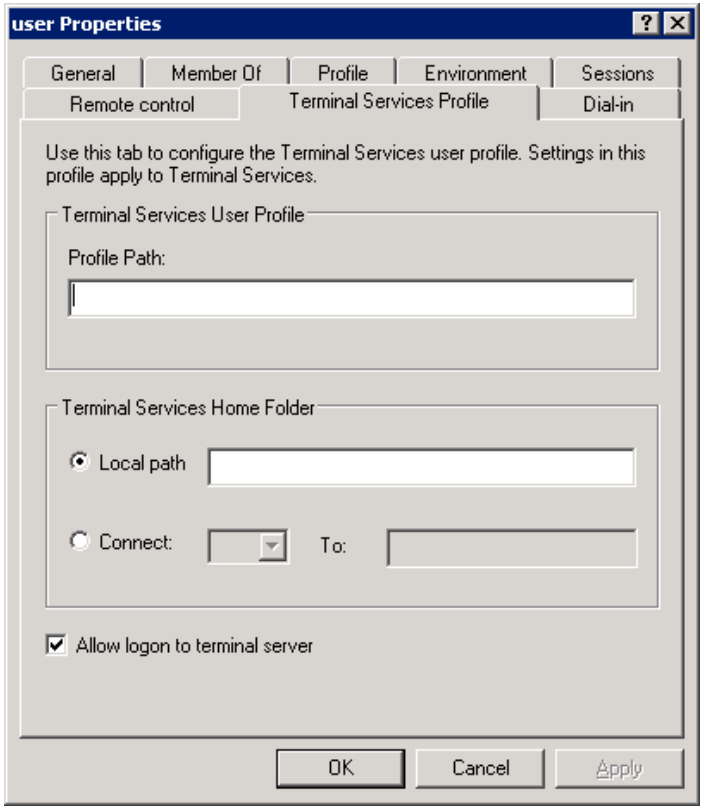
- Kontrola właściwości sesji i zdalnego nadzoru nad sesją.

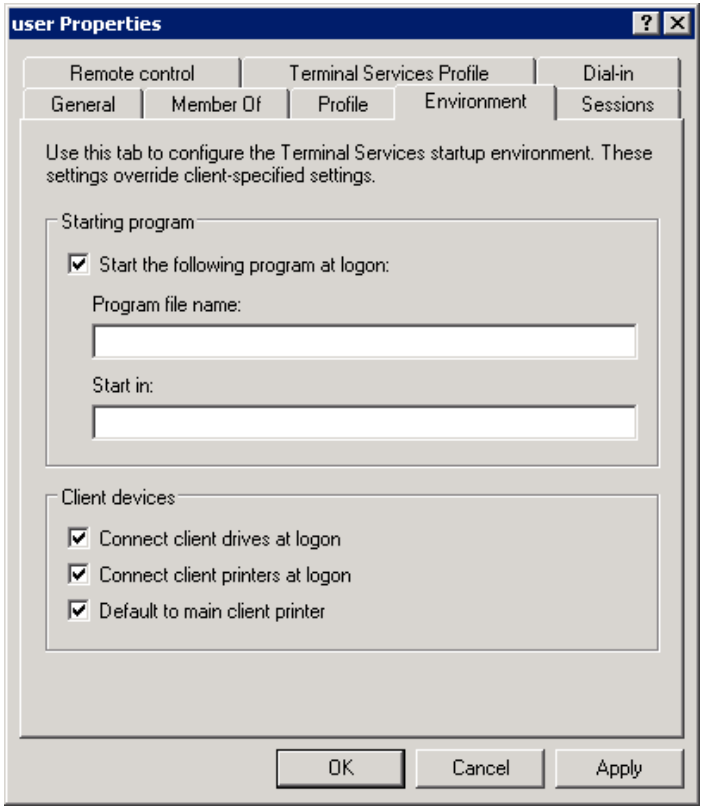




Usługi terminalowe - użytkownicy (II)

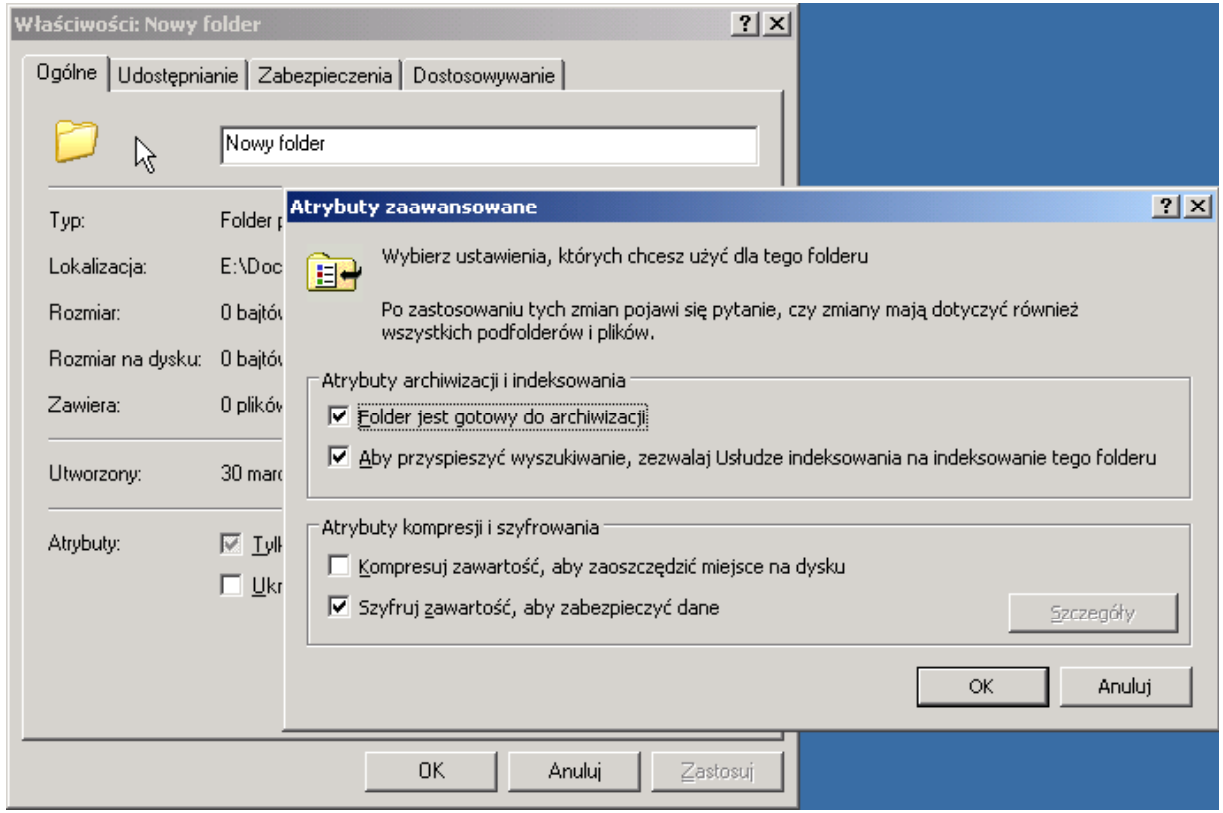
- Definiowanie profilu ustawień dla zdalnego użytkownika, skryptów startowych, aplikacji startowych, automatycznie montowanych zasobów.





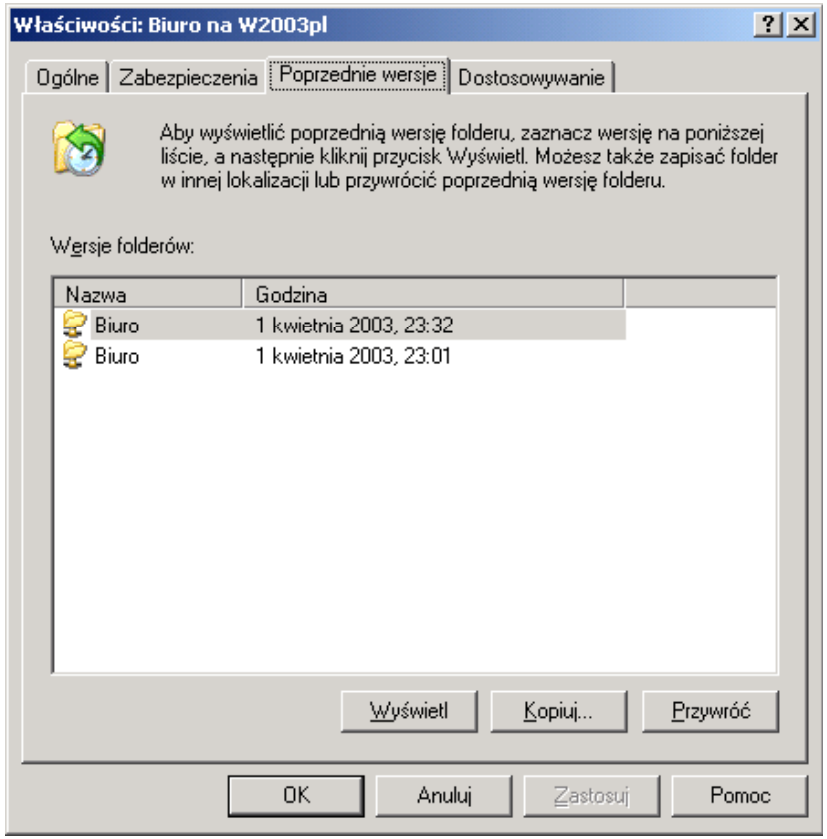
Zarządzanie plikami (I)

- Szyfrowanie i kompresja automatyczna



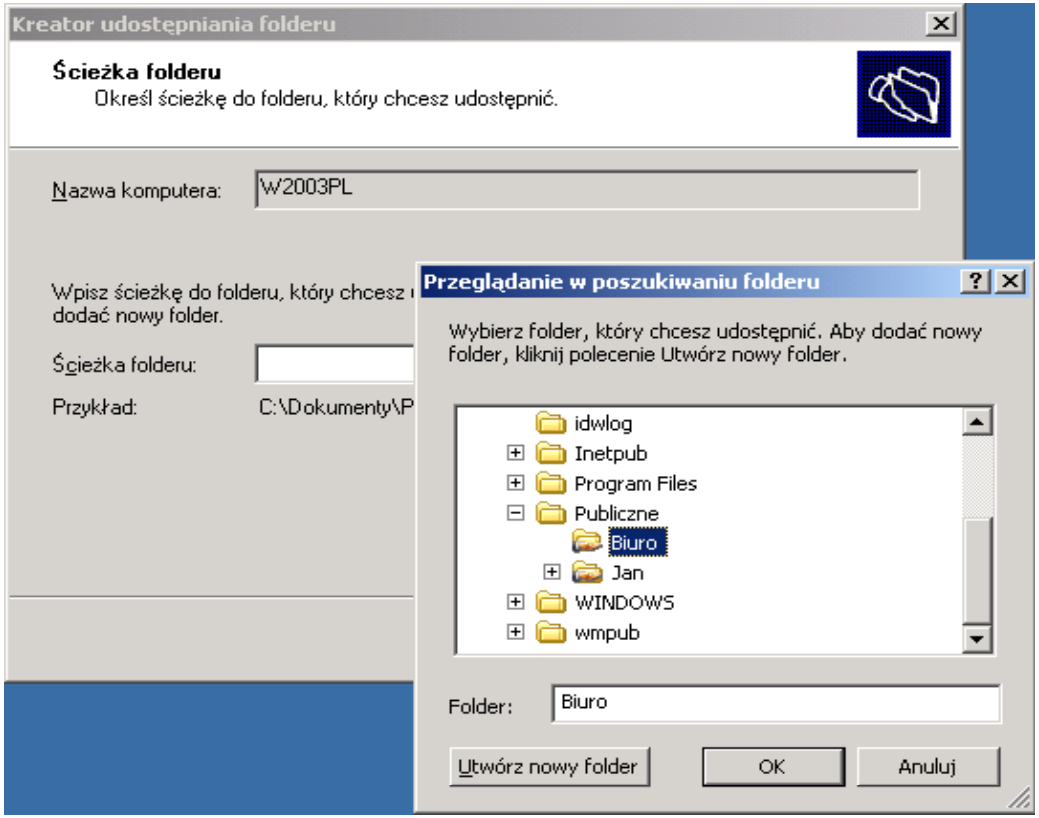
Zarządzanie plikami (II)

- Mechanizm kopii w tle



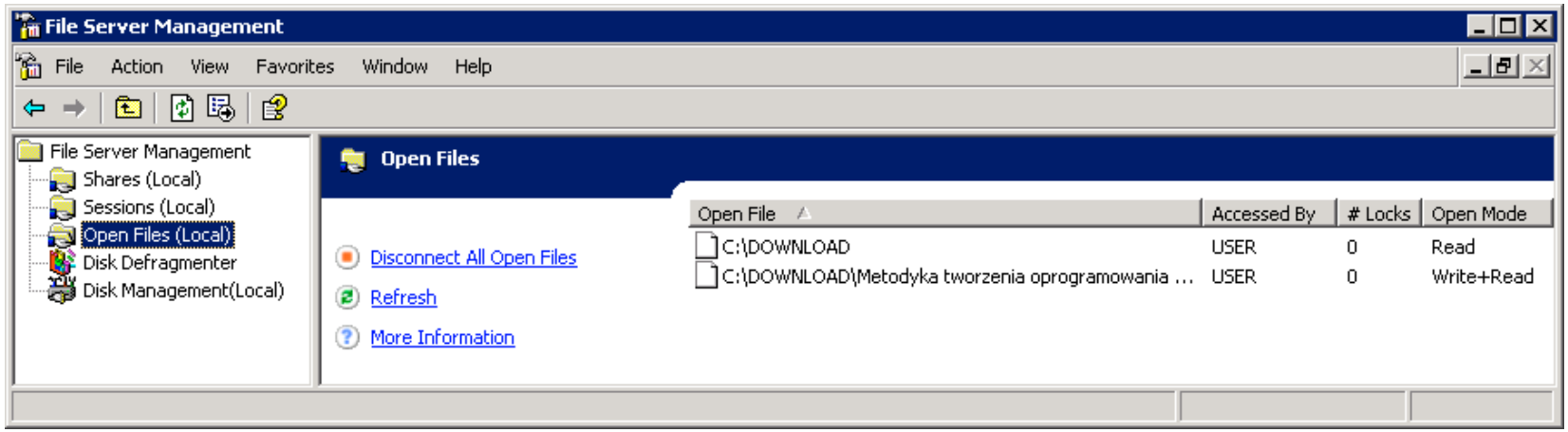
Konfigurowanie roli serwera plików (I)

- Optymalizacja - duplikowanie plików do wersji offline na kliencie



Konfigurowanie roli serwera plików (II)

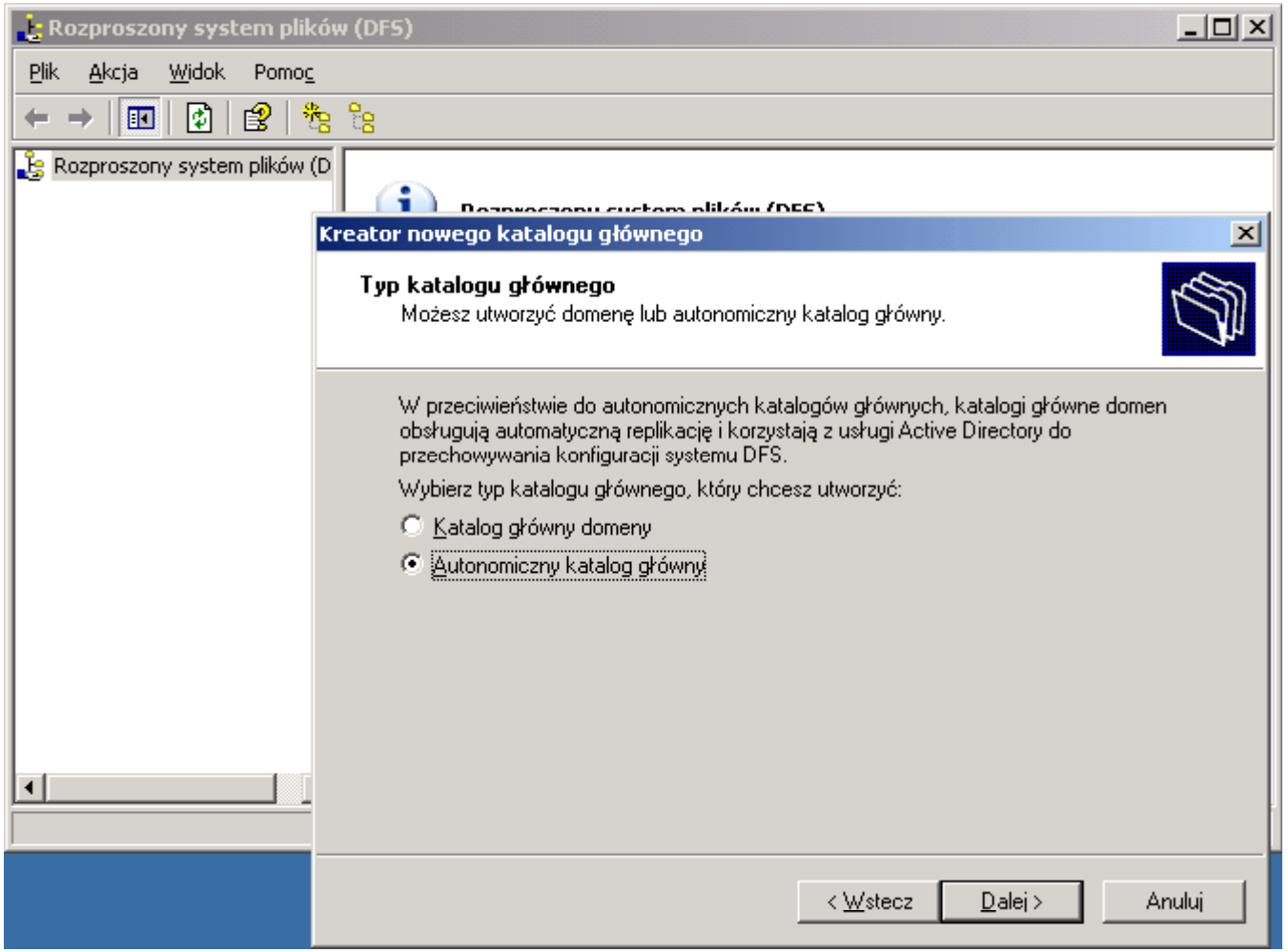
- Konsola filesvr.msc
- Zarządzanie operacjami dyskowymi, udostępnianie zasobów, kontrola sesji i bieżącego dostępu do plików.



Serwer plików - DFS (I)

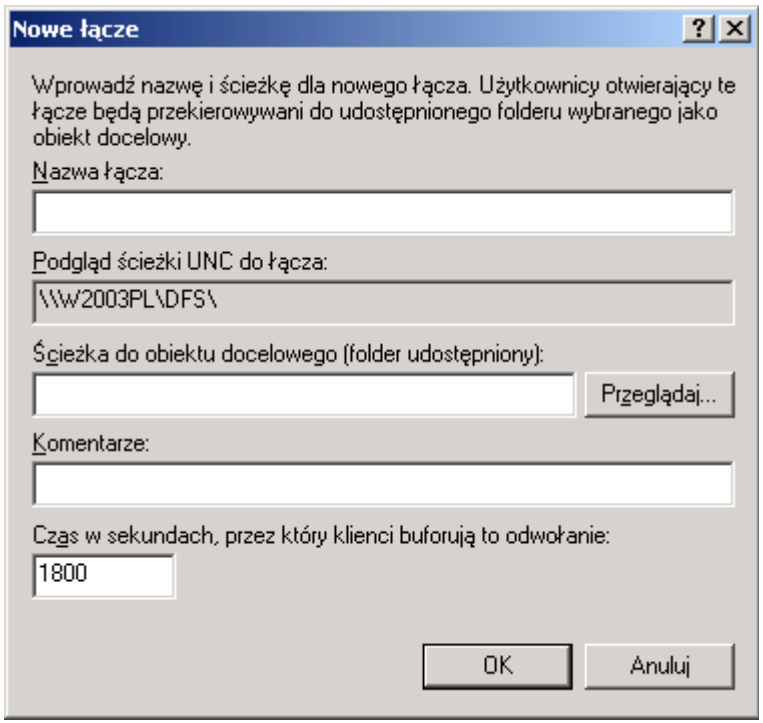
- Rozproszony system plików (DFS - Distributed File System)
- Korzeń drzewa jest elementem domeny (w są nim zasoby na maszynach) lub katalogiem autonomicznym
- W korzeniu znajdują się definicje katalogów wirtualnych prowadzących do udziałów sieciowych (katalogów udostępnionych)
- Konsola MMC - " Rozproszony system plików”

Serwer plików - DFS (II)



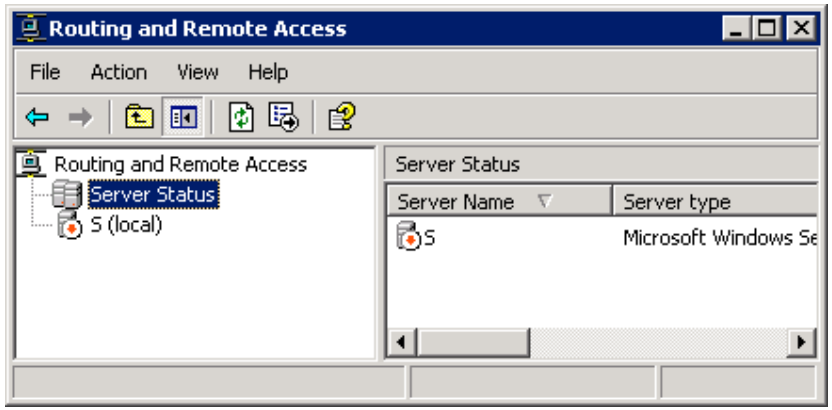
Serwer plików - DFS (III)

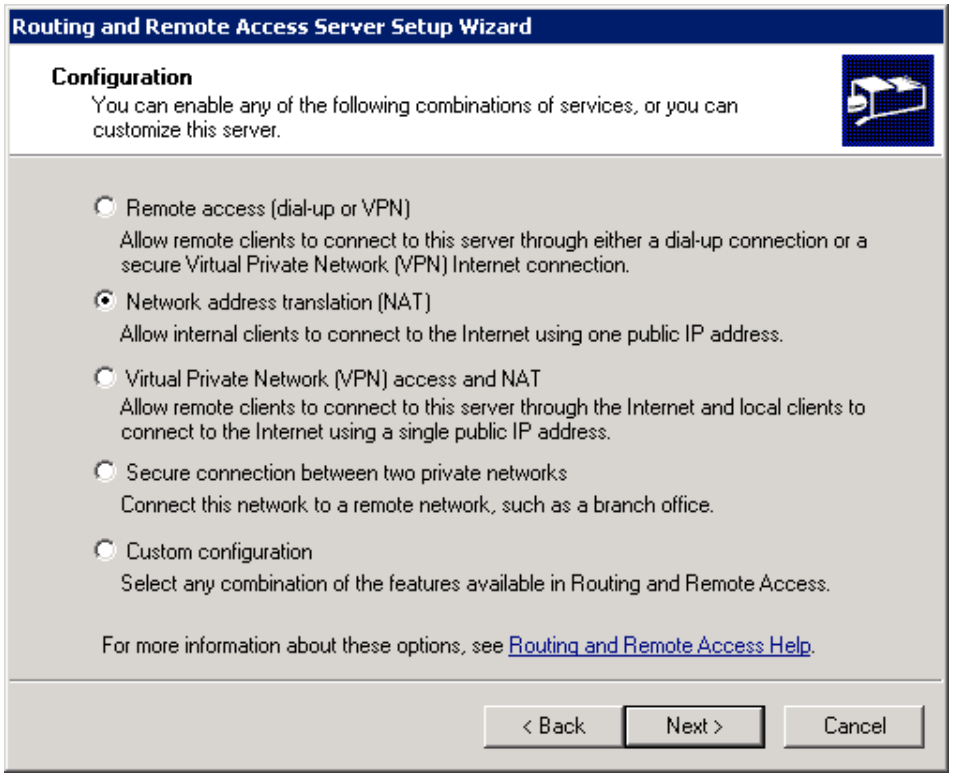
- Dodawanie udziałów sieciowych. Informacja o udziałach jest odświeżana do klienta w ustalonym interwale czasowym.



Routing w użyciem Windows Server

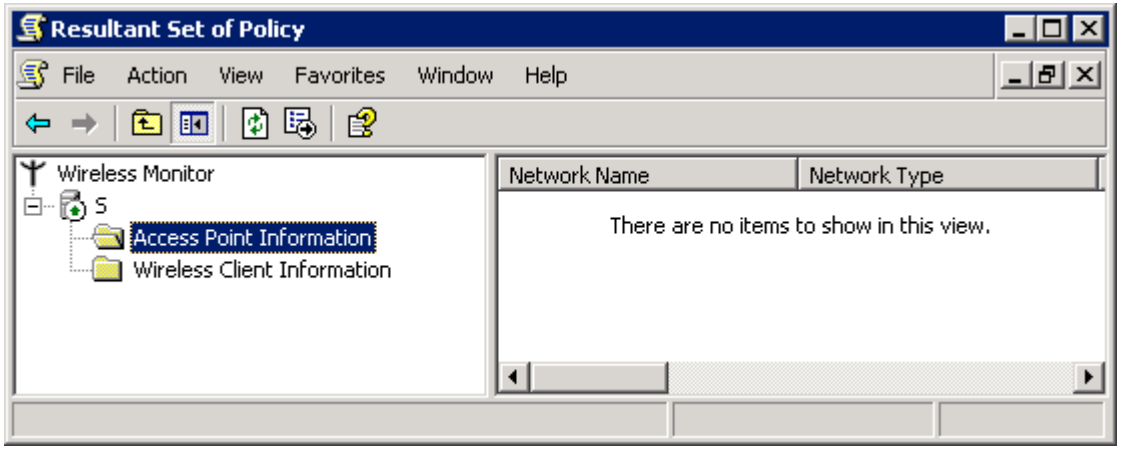
- Usługi bramki sieciowej:
  - filtrowanie po IP
  - NAT
  - VPN





Zarządzanie sieciami bezprzewodowymi

- Dostęp do konfiguracji urządzeń Access Point
- Monitoring i sterowanie kartami klientów, mających połączenia z urządzeniami Access Point



Platforma .NET

- Oprogramowanie łączące informacje, aktorów, systemy i urządzenia.
- Narzędzia klienckie, serwery oraz środowiska programistyczne.
- Podstawowe założenia:
  - Obsługa systemów operacyjnych
  - Wiele języków programowania operujących na tych samych kolekcjach bibliotek

.NET - komponenty

- Środowisko .NET Framework, używane do tworzenia i uruchamiania dowolnego typu programów,
- Narzędzia programistyczne: Visual Studio .NET 2003, Web Matrix (wersja darmowa),
- Microsoft Windows Server 2003, Microsoft SQL Server 2000, platformy integracji działania i zarządzania usługami XML Web Services oraz aplikacjami internetowymi,
- Oprogramowanie klienckie - Windows XP, Microsoft Office XP i inne.

.NET Framework

- Podstawowy składnik niezbędny do tworzenia i uruchamiania aplikacji i usług działających na platformie .NET
- Platforma utrzymująca: Aplikacje typu desktop, usługi systemowe, aplikacje internetowe, aplikacje dla urządzeń przenośnych, usługi XML Web Services
- Wsparcie dla ponad 20 języków programowania: C++, C#, Visual Basic, J#, Fortran, Pascal, Perl, Python, COBOL, SmallTalk i inne
- Struktura warstw: Common Language Runtime -> klasy środowiska -> klasy przetwarzania danych -> ASP .NET lub Windows Forms.

.NET - Common Language Runtime (I)

- *Common Language Runtime* (CLR) podstawowy komponent .NET Framework. Pod jego kontrolą są wykonywane wszystkie korzystające z tej platformy aplikacje.
- Usługi CLR (I)
  - Zarządzanie kodem (uruchamianie i nadzór nad jego wykonywaniem),
  - Izolowanie obszarów pamięci przydzielonych poszczególnym aplikacjom,
  - Weryfikacja zgodności typów,
  - Konwersja języka pośredniego IL do kodu natywnego (maszynowego),

.NET - Common Language Runtime (II)

- Usługi CLR (II)
  - Dostęp do metadanych (rozszerzonej informacji o typach),
  - Zarządzanie pamięcią (w przypadku obiektów zarządzanych),
  - Stosowanie zabezpieczeń dostępu kodu do zasobów,
  - Obsługa wyjątków i przekazywanie ich pomiędzy różnymi językami programowania,
  - Obsługa współpracy pomiędzy kodem zarządzanym, obiektami COM i starszymi bibliotekami DLL,
  - Automatyzacja tworzenia obiektów,
  - Usługi związane z tworzeniem oprogramowania (debugowanie, profilowanie itp.)

.NET - Wspólny system typów

- Wspólny system typów (*ang. Common Type System, CTS*)
- Wszystkie języki platformy .NET dzielą jeden wspólny system typów danych
- Standaryzacja metod operujących na określonych typach danych
- Możliwość wzajemnego wykorzystania danych produkowanych przez aplikacje w różnych językach

.NET - język pośredni

- Język pośredni IL (*Microsoft Intermediate Language /MSIL/*)
- Niezależny od procesora zestaw rozkazów, do którego kompilowane są wszystkie programy korzystające ze środowiska .NET
- Standaryzacja instrukcji niezbędnych do ładowania, przechowywania i inicjalizowania obiektów oraz do wywoływania ich metod
- Pełna integracja pomiędzy różnymi językami programowania

.NET - uruchamianie aplikacji

- Kod aplikacji przechowywany w .dll lub .exe, lecz nie w postaci kodów maszynowych,
- Just-In-Time Compilation - kompilacja fragmentu kodu na żądanie - tuż przed uruchomieniem,
- Kod zarządzany (*Managed code*) - wymagania dla kodu aplikacji dotyczące udostępniania informacji o aplikacji (metadane)

.NET - automatyczne usuwanie zbędnych danych

- Proces odśmiecający (*ang. garbage collector*)
- Mechanizm pozwalający na wykrycie obiektów, które nie będą już używane przez aplikację
- Finalizatory w aplikacjach - przygotowane przez użytkownika fragmenty kodu mające na celu poprawne zwolnienie wszystkich związanych z obiektem zasobów (np. połączeń z bazą danych),
- Dynamiczna defragmentacja pamięci

.NET - podzespoły

- Autonomiczne pakiety pełniące funkcję komponentów udostępniających swoją funkcjonalność i posiadające własne pakiety opisowe, w nich:
  - Tożsamość podzespołu (czyli jego nazwę), numer wersji i kulturę (informacje o regionach i językach obsługiwanych przez podzespół),
  - Podpis cyfrowy podzespołu (jeżeli podzespół ma być używany przez różne aplikacje),
  - Składające się na podzespół pliki,
  - Typy i zasoby składające się na podzespół, wraz z informacją, które z nich są eksportowane,
  - Informacje o powiązaniach z innymi podzespołami,
  - Zestaw uprawnień niezbędnych do poprawnego działania.

.NET - domeny aplikacji

- Domena aplikacji *(ang. application domain)*
- Wirtualny proces, izolujący od siebie poszczególne aplikacje .NET.
- W jednym rzeczywistym procesie systemu operacyjnego może istnieć wiele domen aplikacji
- Obiekty działające w tej samej domenie (i za razem aplikacji) mają do siebie wzajemny dostęp

.NET - biblioteka klas

- Składa się z klas, interfejsów i stałych, które są zawarte w pakiecie Microsoft .NET Framework SDK,
- Elementy biblioteki klas pogrupowane za pomocą hierarchicznej struktury przestrzeni nazw (ang. namespace),
- Zawężenie zakresu, w którym obowiązują nazwy typów - w aplikacji mogą istnieć dwie klasy o tej samej nazwie pod warunkiem, że są zdefiniowane w różnych przestrzeniach nazw
- Dwie przestrzenie główne: System i Microsoft.

.NET
- przestrzenie nazw (I)

Microsoft.CSharp	Klasy obsługujące kompilację i generowanie kodu przy użyciu języka programowania C#.
Microsoft.JScript	Klasy obsługujące kompilację i generowanie kodu przy użyciu języka programowania JScript.
Microsoft.VisualBasic	Klasy obsługujące kompilację i generowanie kodu przy użyciu języka programowania Visual Basic .NET.
Microsoft.Win32	Zawiera dwa rodzaje klas: obsługujące zdarzenia wyzwalane przez system operacyjny oraz pozwalające na manipulowanie rejestrem systemowym.
System	Podstawowe klasy oraz klasy bazowe, które definiują najczęściej używane typy danych, zdarzenia, procedury obsługi zdarzeń, interfejsy, atrybuty i wyjątki. Inne klasy w tej przestrzeni nazw udostępniają usługi związane z konwersja typów danych, manipulowaniem parametrami metod, obliczeniami matematycznymi, zdalnym i lokalnym wywoływanie programów, zarządzaniem środowiskiem aplikacji, oraz nadzorowaniem zarządzanego i niezarządzanego kodu.

.NET
- przestrzenie nazw (II)

System.Collections	Interfejsy i klasy, które definiują przeróżne kolekcje obiektów - listy, kolejki, tablice bitów, tablice skrótów (ang. hashtables) i słowniki (ang. dictionaries).
System.Configuration	Klasy i interfejsy które pozwalają na programowy dostęp do ustawień konfiguracyjnych .NET Framework i obsługę błędów w plikach konfiguracyjnych (pliki .config).
System.Data	Głównie klasy składające się na architekturę dostępu do danych ADO .NET. Architektura ta umożliwia tworzenie komponentów, które pozwalają na wydajne zarządzanie danymi pochodzącymi z różnych źródeł. ADO .NET.
System.Data.Common	Klasy współdzielone przez dostawców danych .NET Framework. Dostawca danych – to zbiór klas pozwalających na korzystanie z danych pochodzących z określonego źródła, np. bazy danych.
System.Data.ODBC	Klasy składające się na .NET Framework Data Provider for ODBC, czyli dostawcę danych dla interfejsu ODBC.
System.Data.OleDB	Klasy składające się na .NET Framework Data Provider for OleDB, czyli dostawcę danych dla interfejsu OleDB.

.NET
- przestrzenie nazw (III)

System.Data.SqlClient	Klasy składające się na .NET Framework Data Provider for SQL Server, czyli dostawcę danych dla serwera MS SQL Server.
System.Data.SqlTypes	Klasy reprezentujące rodzime typy danych serwera Microsoft SQL Server.
System.DirectoryServices	Dostęp do Active Directory z poziomu kodu zarządzanego.
System.Drawing	Dostęp do podstawowej funkcjonalności graficznej biblioteki systemowej GDI+.
System.Drawing. Drawing2D	Zaawansowana funkcjonalność związana z grafiką 2D i grafiką wektorową.
System.Drawing.Text	Zaawansowana funkcjonalność związana z typografią i wykorzystaniem kolekcji czcionek.
System.IO	Typy umożliwiające synchroniczny i asynchroniczny odczyt i zapis strumieni danych i plików.

.NET
- przestrzenie nazw (IV)

System.Net	Proste interfejsy programistyczne dla protokołów sieciowych.
System.Net.Sockets	Zarządzana implementacja interfejsu Windows Sockets.
System.Runtime. Serialization	Klasy które mogą być używane do serializowania i deserializowania obiektów.
System.Security	Dostęp do podsystemu zabezpieczeń .NET Framework.
System.Security. Cryptography	Usługi kryptograficzne - bezpieczne kodowanie i dekodowanie danych, tworzenie skrótów jednokierunkowych, generowanie liczb losowych i uwierzytelnianie komunikatów.
System.Text. RegularExpressions	Klasy umożliwiające manipulacje na wyrażeniach regularnych.
System.Threading	Klasy i interfejsy umożliwiające programowanie wielowątkowe.

.NET
- przestrzenie nazw (V)

System.Timers	Komponent Timer umożliwiający wywoływanie zdarzenia w określonych odstępach czasu.
System.Web	Klasy i interfejsy umożliwiające komunikację przeglądarki internetowej z serwerem, a ponadto: manipulacje plikami cookie, transfer plików, obsługę wyjątków i zarządzanie buforem wyjściowym.
System.Web.Caching	Klasy pozwalające na sterowanie buforowaniem często używanych danych po stronie serwera.
System.Web.Configuration	Klasy pozwalające na odczyt i modyfikację konfiguracji ASP .NET
System.Web.Mail	Klasy umożliwiające tworzenie i wysyłanie wiadomości email.
System.Web.Security	Klasy używane do implementowania zabezpieczeń ASP .NET w aplikacjach sieci Web.
System.Web.UI	Klasy i interfejsy pozwalające na tworzenie kontrolek i stron składających się na interfejs użytkownika aplikacji ASP .NET.

.NET
- przestrzenie nazw (VI)

System.Web.UI. HtmlControls	Klasy umożliwiające tworzenie kontrolek HTML działających po stronie serwera. Kontrolki HTML są mapowane bezpośrednio na tagi HTML interpretowane przez przeglądarki internetowe i pozwalają na programowe manipulowanie elementami HTML na stronie internetowej.
System.Web.UI. WebControls	Klasy umożliwiające umieszczanie na stronie internetowej kontrolek działających po stronie serwera. Zaliczają się do nich elementy formularzy, takie jak przyciski i pola tekstowe, oraz kontrolki takie jak kalendarz czy DataGrid.
System.Windows.Forms	Klasy umożliwiające tworzenie funkcjonalnych aplikacji działających pod kontrolą systemu Microsoft Windows
System.Xml	Klasy umożliwiające przetwarzanie dokumentów XML.

- Aplikacje internetowe - możliwość całkowitego oddzielenia kodu skryptowego (aplikacji) wykonywanego po stronie serwera od kodu HTML prezentowanego klientowi.

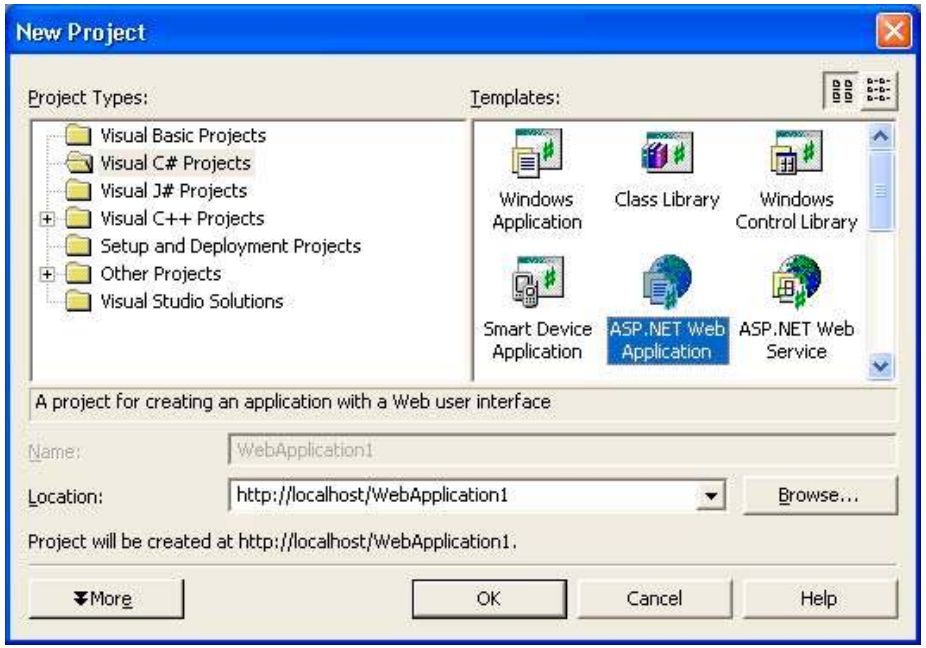
.NET
- najprostsza próbka kodu

- C sharp (C#) - mogący pełnić zarówno rolę języka skryptowego jak i aplikacji, łączący w sobie cechy Javy i CPP:

```
// Hello.cs
public class Hello {
        public static void Main() {
                System.Console.WriteLine("Hello, World!");
        }
}
```

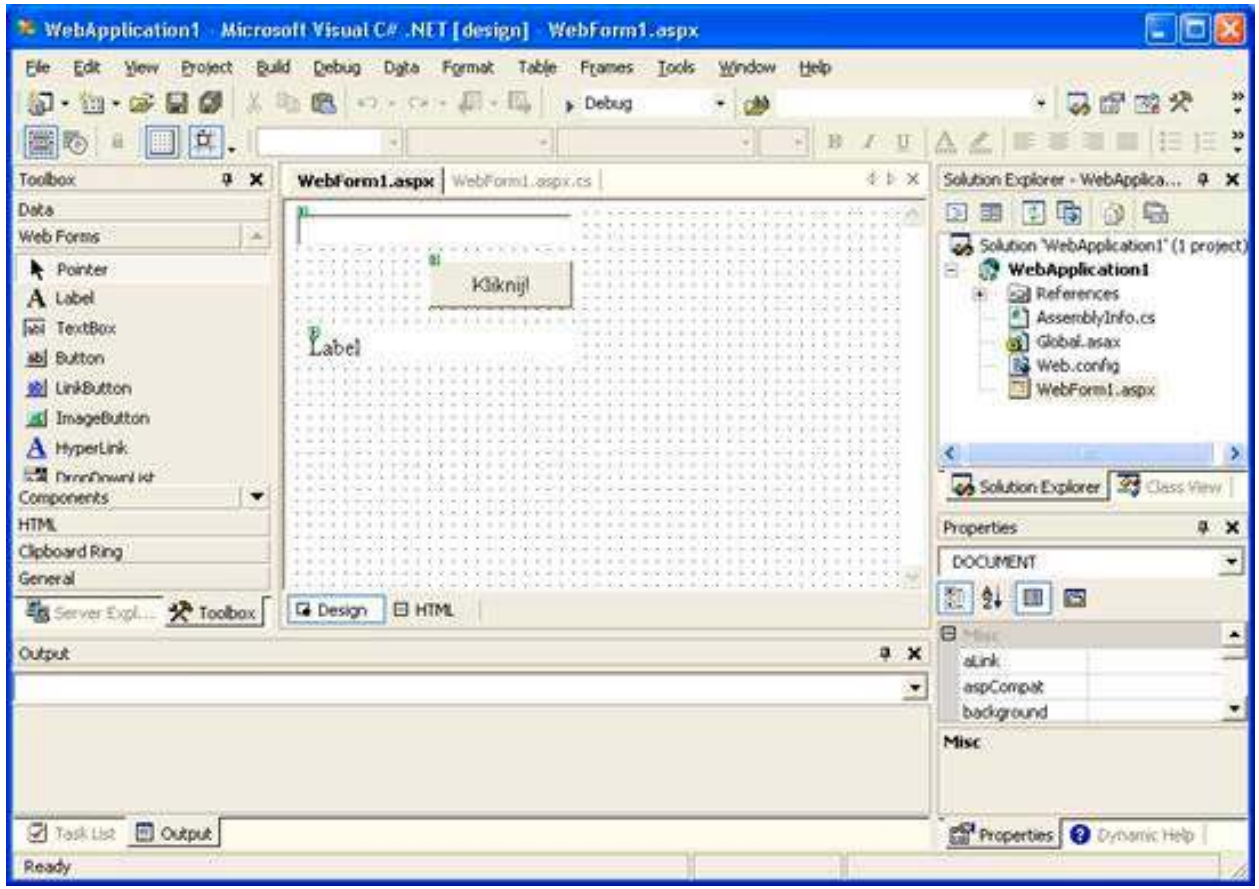
.NET - aplikacje internetowe (I)

- ASP .NET
- Nowe podejście(technika *Code Behind*): rozdzielenie HTML (który teraz jest generowany i wizualnie edytowany) od kodu aktywnego (C#)
- Pełna integracja z Visual Studio .NET



.NET - aplikacje internetowe (II)

- Zintegrowany edytor formularzy



.NET - aplikacje internetowe (III)

- Kodowanie w języku obiektowym i w modelu zdarzeniowym
- Dla C#:

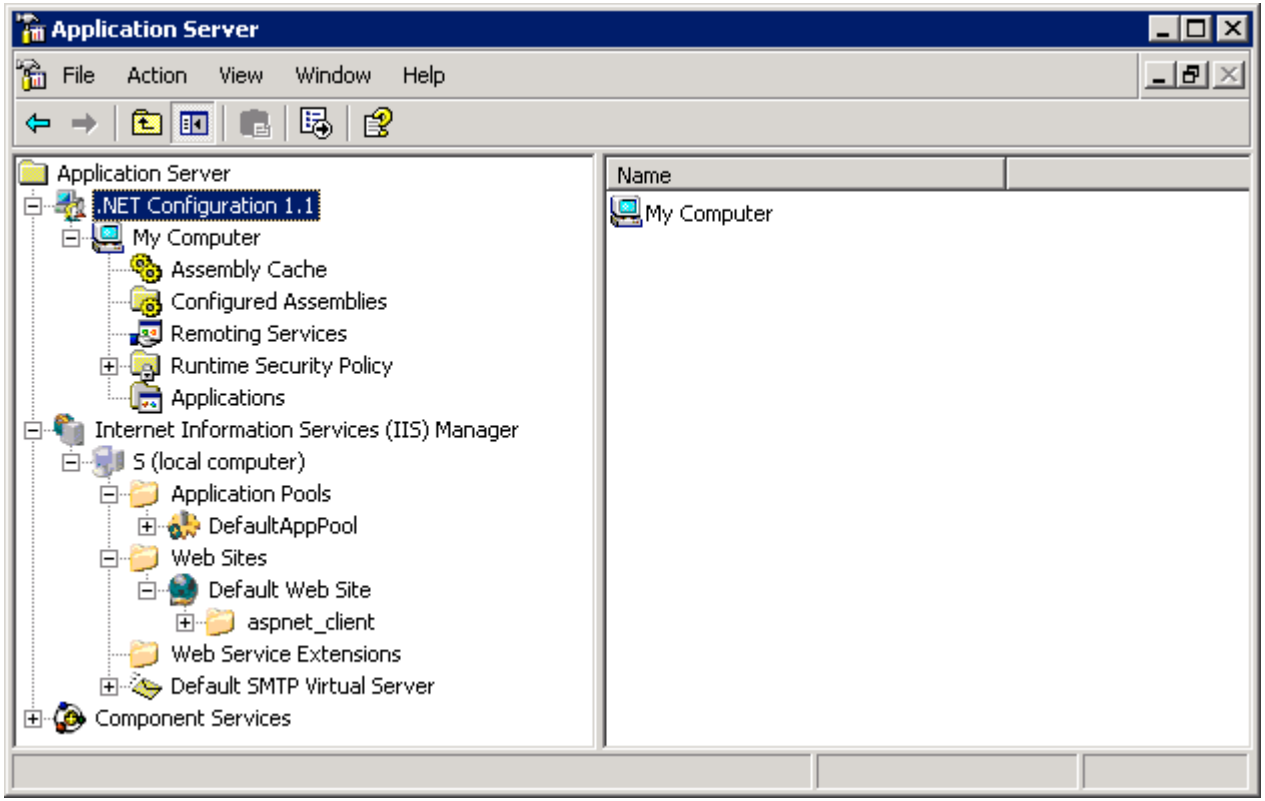
```
private void Przycisk1_Click(object sender, System.EventArgs e)
{
    if (Tekst1.Text.Length > 0)
        Etykieta1.Text = ”Podałeś: " + Tekst1.Text;
    else
        Etykieta1.Text = ”Nic nie wpisano";
}
```

.NET - aplikacje internetowe (IV)

- Uporządkowanie zasobów aplikacji. Wprowadzenie standardowych plików, definiujących jej zachowanie:
  - *Global.asax* - zawiera kod obsługi zdarzeń wspólnych dla wszystkich stron tworzących aplikację internetową, np. uruchomienie i zamknięcie aplikacji, otwarcie i zamknięcie sesji itp.
  - *Web.config* - zawiera konfiguracyję aplikacji i jest zapisany w formacie XML. Umożliwia konfigurację praw dostępu do aplikacji, uwierzytelniania użytkowników.
  - Zbiór plików *.aspx*, tworzących poszczególne strony internetowe.
  - Pliki z kodem (np. *.cs dla języka c#)

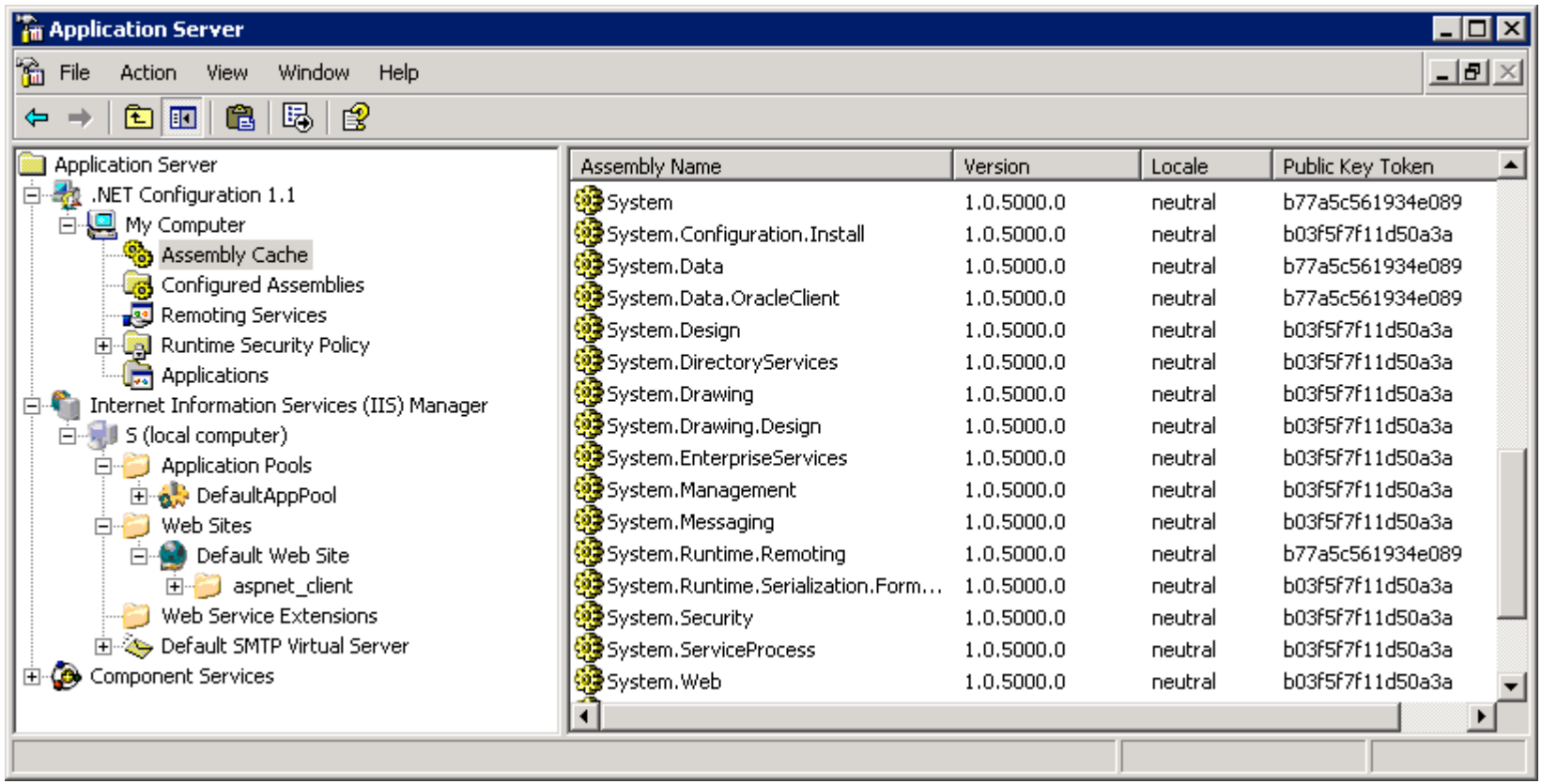
Application Server (I)

- Serwer aplikacji ASP, ASP.NET, serwer komponentów.



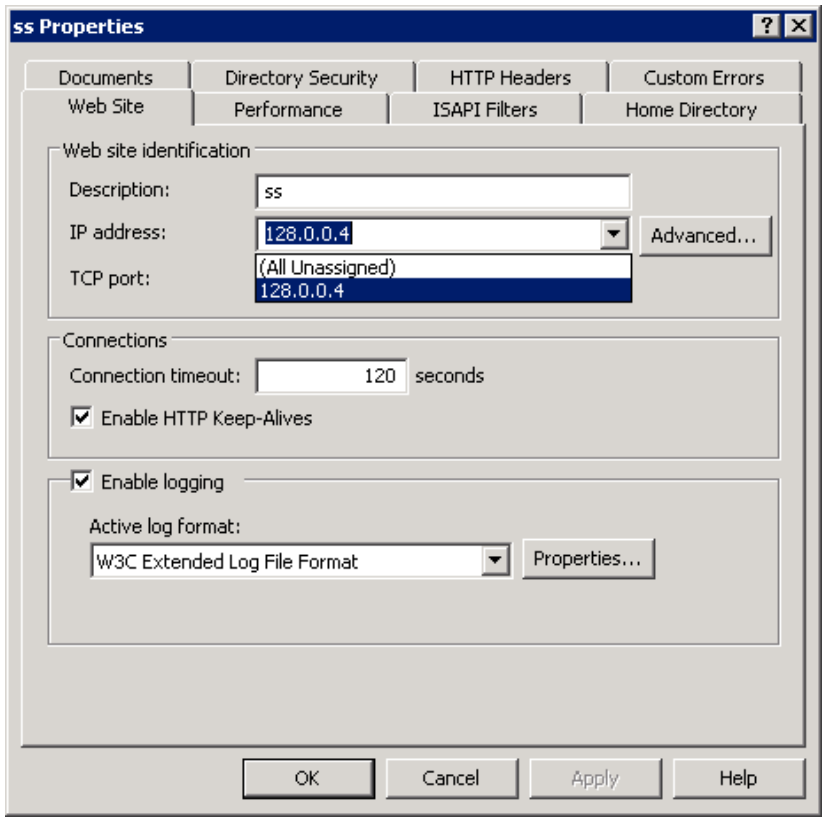
Application Server (II)

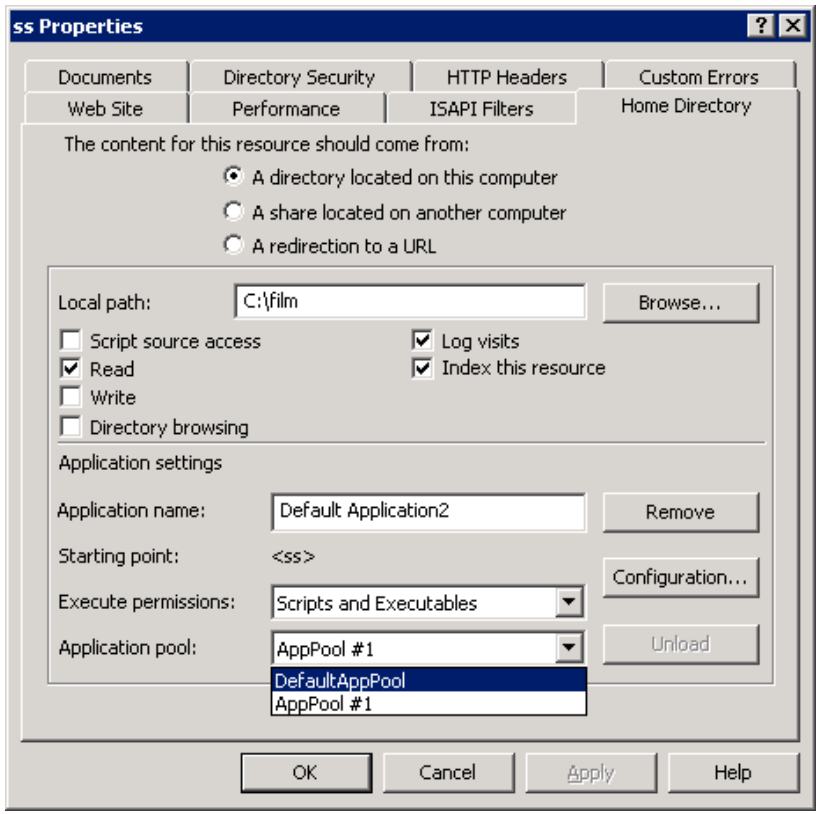
- .NET - sterowanie Assembly Cache i schowkiem skonfigurowanych aplikacji.



Application Server (III)

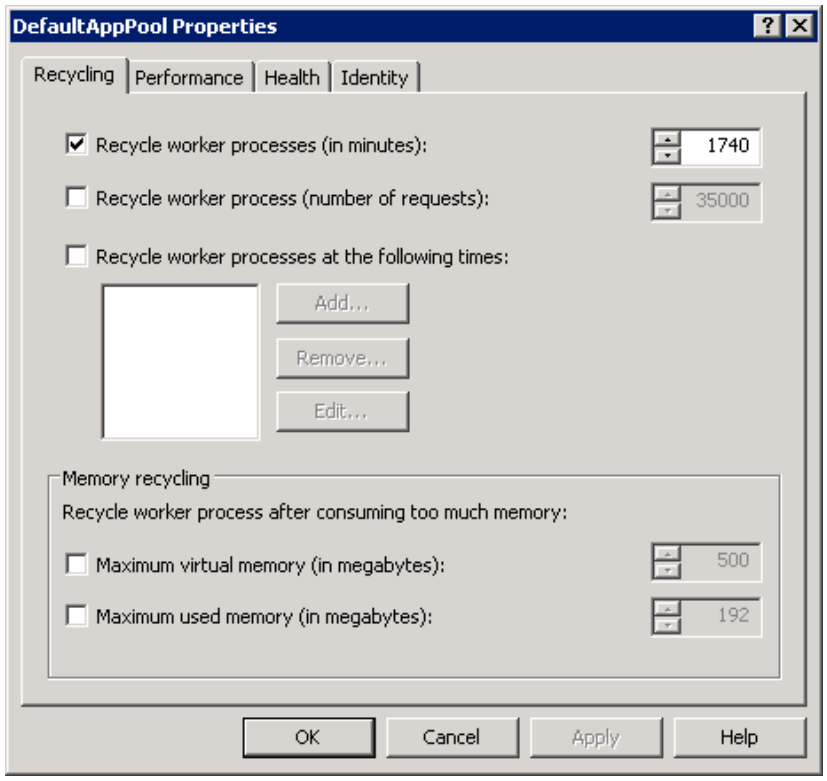
- Konfigurowanie wielu aplikacji kojarzonych z różnymi adresami IP i portami:

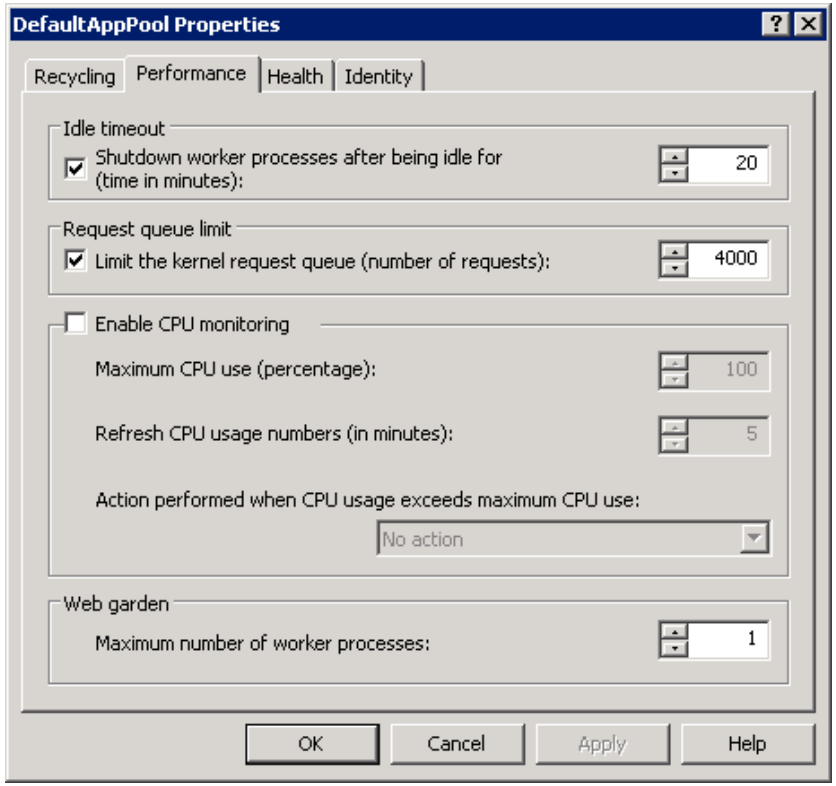




Application Server (IV)

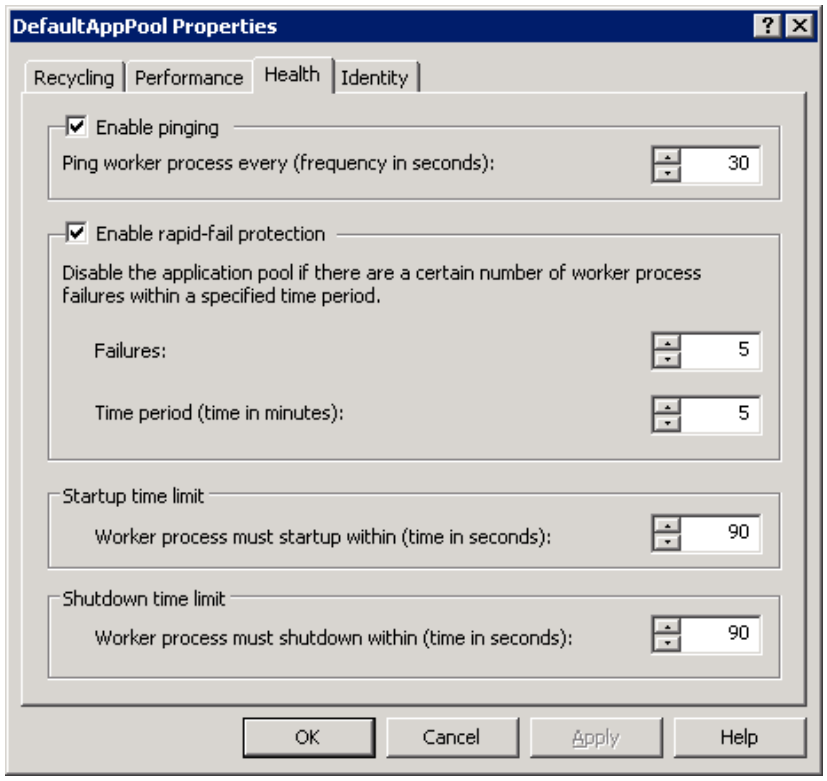
- *Application pools* - zaawansowane zarządzanie procesami wykonawczymi aplikacji

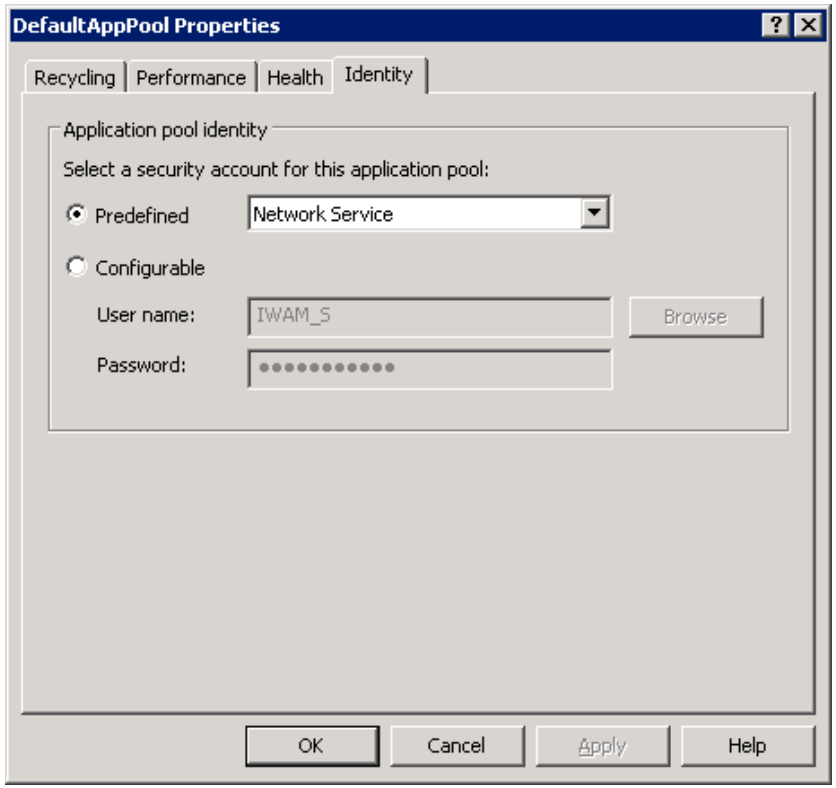




Application Server (V)

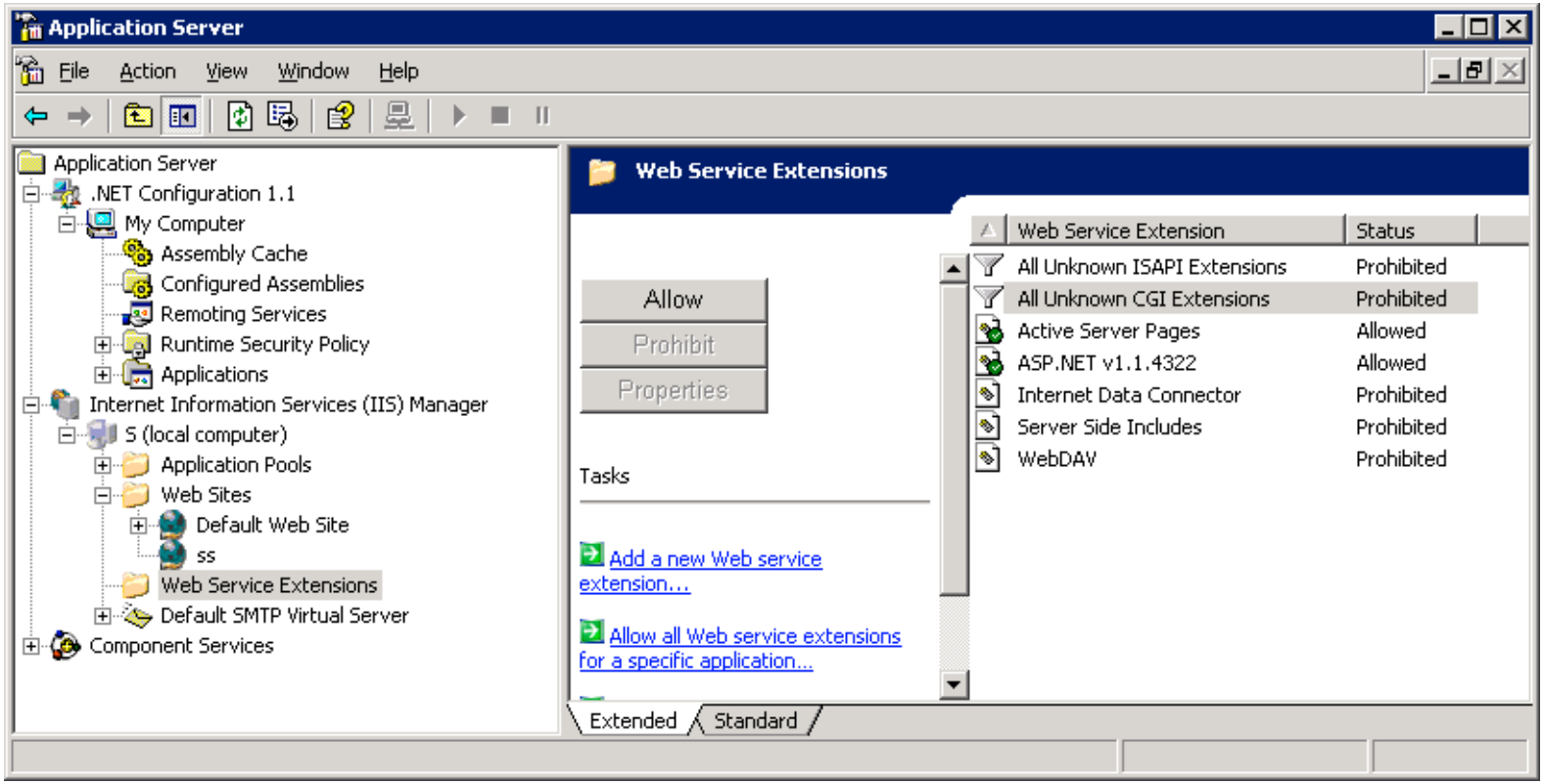
- *Application pools* - zaawansowane zarządzanie procesami wykonawczymi aplikacji (cd)





Application Server (VI)

- Dodatkowa warstwa zezwoleń dla modułów aktywnych ASP (rozszerzeń serwera aplikacji):



Konfigurowanie prostej aplikacji typu WebForms (I)

- Kroki konfigurowania aplikacji WebForms
  - Zadanie dla twórcy aplikacji: Stworzyć formularze (*.aspx) oraz sprzężone z nimi pliki z kodem obsługującym po stronie serwera (*.cs) - generatory. Narzędzia (generatory) to przykładowo: Visual Studio .NET, Visual C# Express, SharpDevelop
  - Zbudować projekt (z plików *.cs) do postaci bibliotek (*.dll).
  - Stworzyć fizyczny katalog aplikacji
  - Umieścić pliki *.dll w podkatalogu /bin katalogu aplikacji

Konfigurowanie prostej aplikacji typu WebForms (II)

- Kroki konfigurowania aplikacji WebForms (cd):
  - Stworzyć katalog wirtualny aplikacji - o nazwie **zbieżnej** z nazwą głównego namespace aplikacji (zdefiniowanego w głównym pliku *.cs)
  - Umieścić w katalogu fizycznym pliki *.aspx aplikacji.
  - W niektórych przypadkach - umieścić w plikach *.aspx dyrektywy importujące główny namespace.

Własne komponenty w .NET (I)

- Przykład kodu testowego komponentu:

```
// File: TempConverter.cs
using System;
using System.Reflection;
[assembly : AssemblyVersion("1.0.0.0")]
namespace EssentialAspDotNet.Architecture
{
        public class TempConverter
        {
                static public double FahrenheitToCentigrade(double val)
                {
                        return ((val-32)/9)*5;
                }
                static public double CentigradeToFahrenheit(double val)
                {
                        return (val*9)/5+32;
                }
        }
}
```

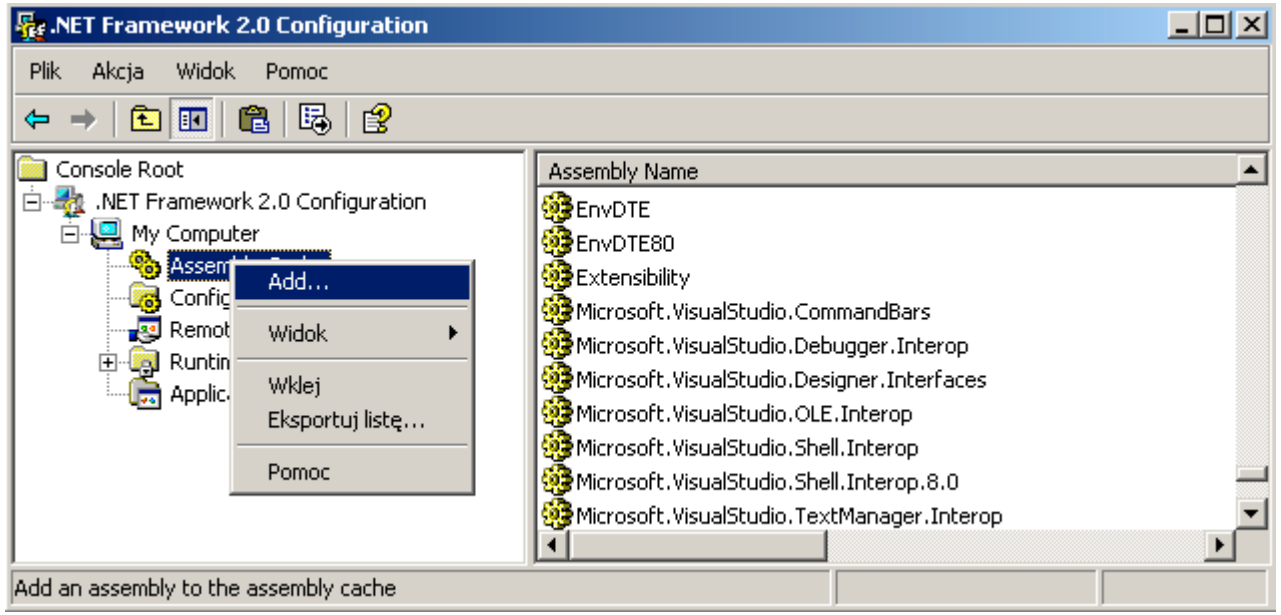
Własne komponenty w .NET (II)

- Kompilacja komponentu (wymagane .NET SDK):
  - Kompilator C# : csc.exe
  - Format docelowy: biblioteka dll, Assembly binary
  - Przykładowa Komenda kompilacji (pliku TempConverter.cs) :

    csc /t:library /out:TempConverter.dll TempConverter.cs

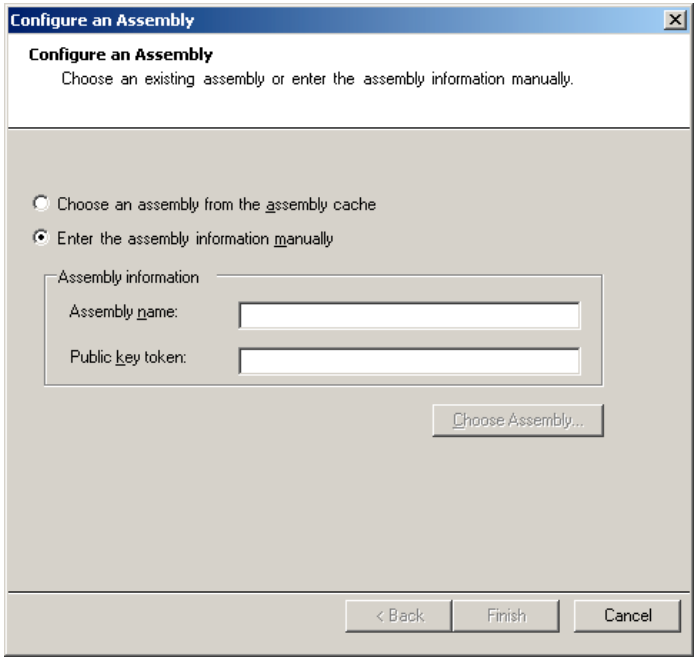
Własne komponenty w .NET (III)

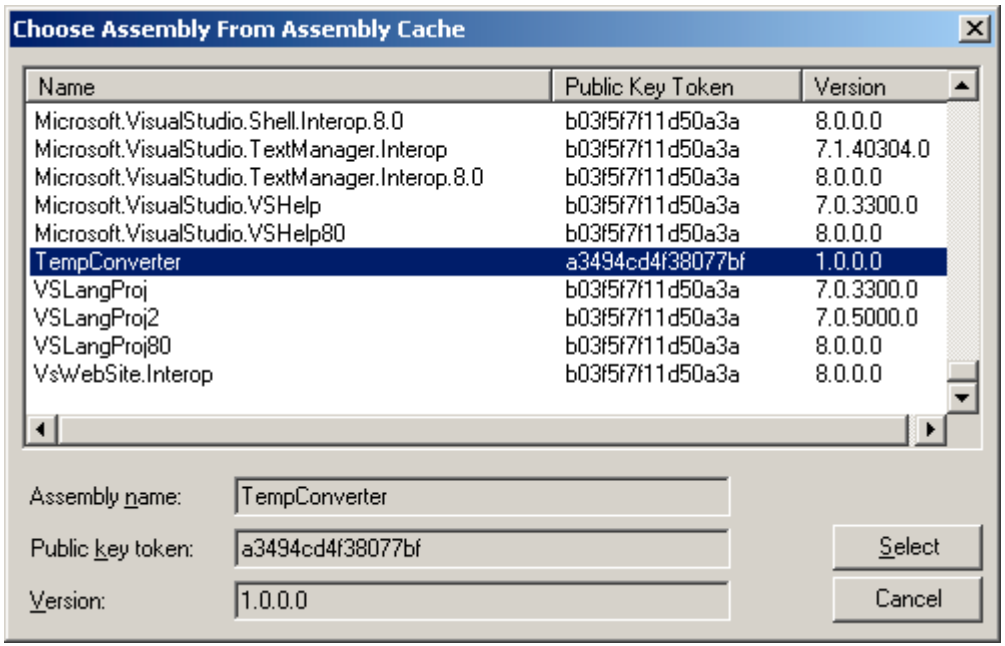
- Instalacja Assembly:



Własne komponenty w .NET (IV)

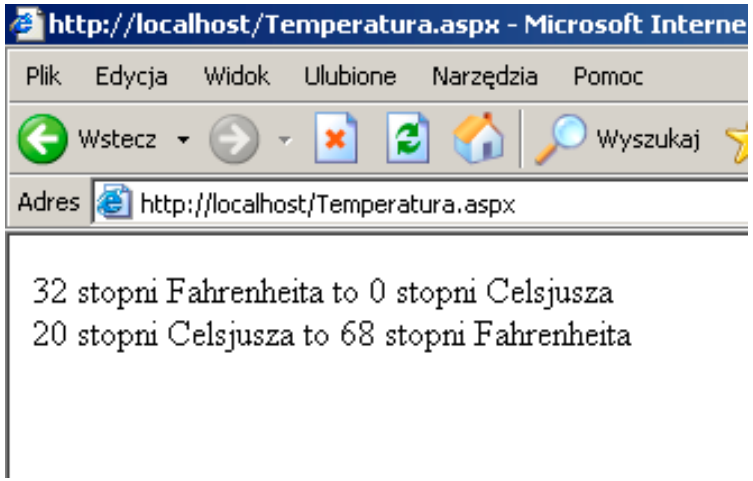
- Konfiguracja Assembly:
  - Dodanie assembly do listy konfiguracji wymaga podania jej nazwy (Name) oraz klucza haszującego (Public Key Token)





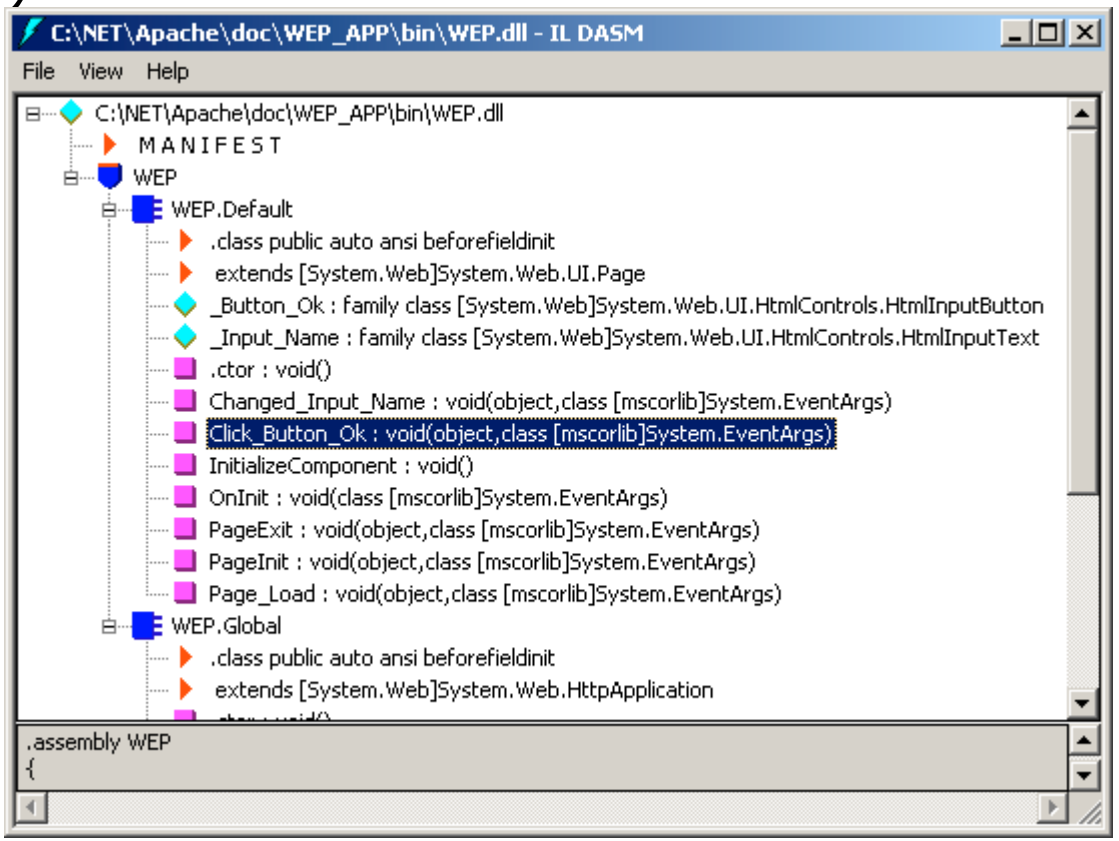
Własne komponenty w .NET (V)

```
<!-- File: Temperatura.aspx -->
<%@ Page Language='C#' %>
<%@ Assembly Name="TempConverter, Version=1.0.0.0,
Culture=Neutral,PublicKeyToken=a3494cd4f38077bf" %>
<html><body>
32 stopni Fahrenheita to
<%= EssentialAspDotNet.Architecture.TempConverter.FahrenheitToCentigrade(32)%>
stopni Celsjusza<br>
20 stopni Celsjusza to
<%= EssentialAspDotNet.Architecture.TempConverter.CentigradeToFahrenheit(20)%>
stopni Fahrenheita<br>
</body></html>
```



Dekompilacja Assembly

- Umożliwia przeglądanie treści
- Pomocna w lokalizacji brakujących komponentów (w analizie błędów braku komponentu)
- Narzędzie:ildasm.exe



Klastry serwerów

- **Klaster serwerów** - grupa komputerów będących serwerami, które są połączone w sieć zarówno fizycznie, jak i za pomocą oprogramowania. Klaster świadczy usługi na rzecz klientów podobnie jak indywidualny serwer, jednak usługi te realizowane są na wielu serwerach wchodzących w jego skład, przeważnie homogenicznych, zgodnie z polityką realizacji tych usług prowadzoną w ramach klastra.

Zalety klastrów serwerów

- Odporność na awarie - przejmowanie funkcjonalności serwerów, które uległy awariom
- Stabilność przeciążeniowa serwerów - poprzez zrównoważenie obciążenia na składowych serwerach klaster jest w stanie zapewnić dużą wydajność również wtedy, gdy zapotrzebowanie na różne rodzaje usług zmienia znacząco się w czasie.

Klastry serwerów Windows (I)

- Konfiguracja węzłów na poziomie adresacji IP - rozdzielenie ruchu IP na poszczególne serwery w sieci pracujące w klastrze
- Obsługa awarii - uszkodzenie serwera powoduje uruchomienie procesu odtwarzania, automatycznie restartującego aplikacje serwera na maszynie zastępczej w klastrze
- Łatwość zarządzania - Aby zarządzać istniejącym klastrem – wystarczy dołączyć się do jednego z hostów.

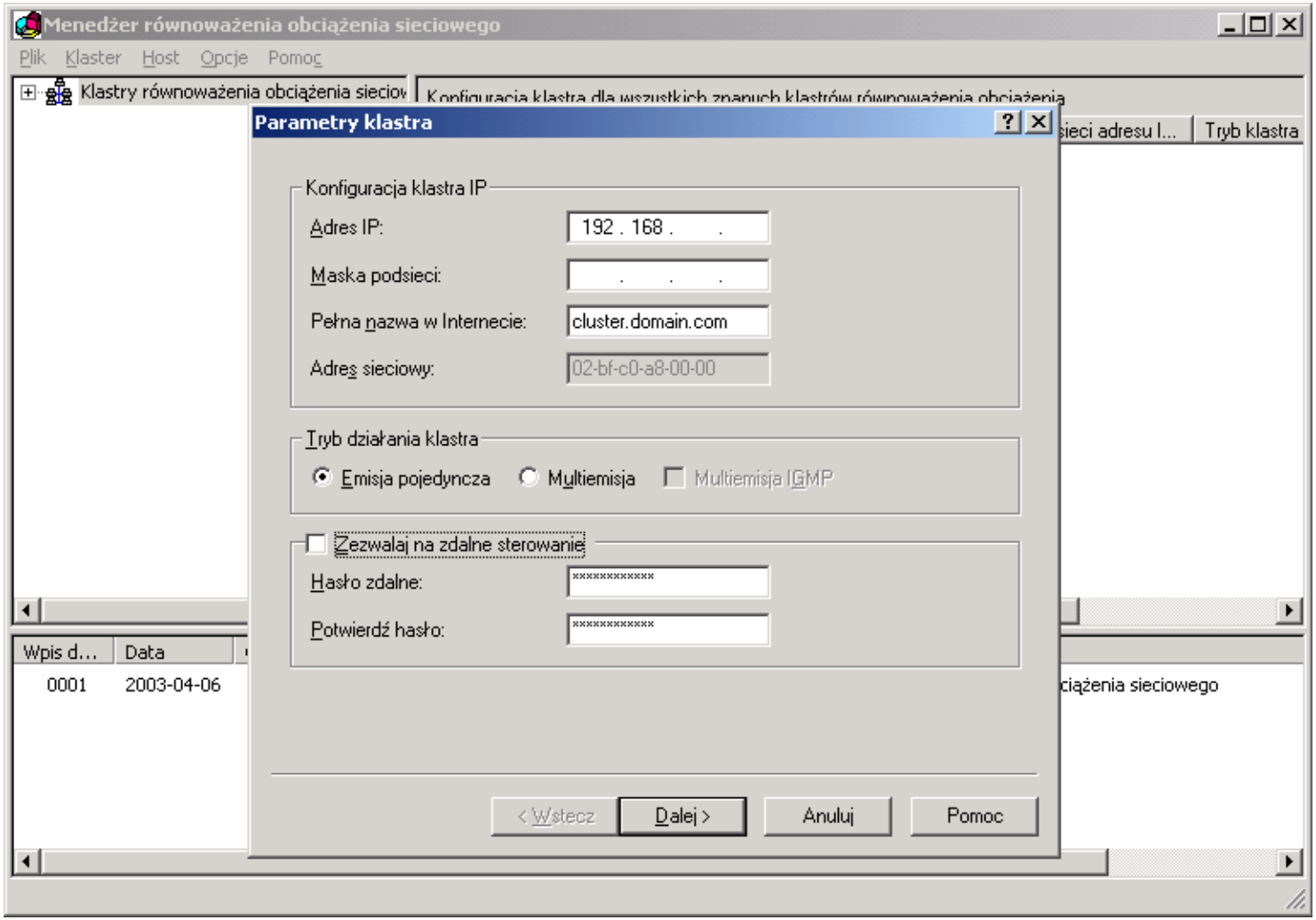
Klastry serwerów Windows (II)

- Zastosowania:
  - serwery web - poddomeny (z za tym np. aplikacje internetowe) rozproszone na różnych , możliwość realizowania rozproszonych transakcji,
  - zasoby danych - najnowsze wersje SQL Server przystosowane do rozproszenia lub zwielokrotnienia (backup) baz danych
  - ochrona i zarządzanie zasobami

Klastry serwerów Windows (III)

- Mechanizm wyrównywania obciążeń (NLB – Network Load Balancing).
- Dostępny w każdej edycji Windows 2003 serwer
- Pozwala równomiernie rozłożyć przychodzący ruch IP, również na poziomie portu.
- W Windows 2003 Server: Menedżer równoważenia obciążenia sieciowego - pozwala tworzyć klastry, propaguje parametry na wszystkie maszyny zdefiniowane jako klaster
- NLB na serwerach z dowolną liczbą kart sieciowych powoduje, jeden fizyczny serwer może być częścią różnych klastrów NLB

Klastry serwerów Windows (IV)



Ciekawsze miejsca do poszukiwań informacji dla administratorów Windows

- http://www.microsoft.com/poland/windowsserver2003
- http://msdn.microsoft.com
- http://windows.about.com
- http://www.windows2003.pl
- http://www.codeguru.pl
- http://www.windowsdevcenter.com
- http://www.microsoft.com/poland/technet
- http://groups.msn.com/windowsscript
- http://www.winguides.com/scripting/
- http://www.windowsitpro.com/