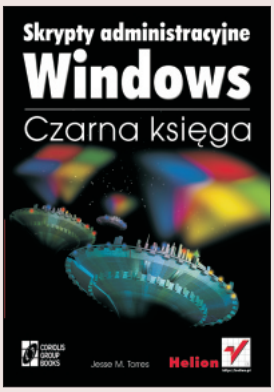
Skrypty administracyjne Windows. Czarna księga



Autor: Jesse M. Torres
Tłumaczenie: Marcin Jędrysiak
ISBN: 83-7197-516-3
Tytuł oryginału: Windows Admin Scripting Little Black Book
Format: B5, stron: około 400

Książka „Skrypty administracyjne Windows" została napisana po to, aby szybko nauczyć Cię przekształcenia w proste skrypty rutynowych, powtarzalnych i skomplikowanych zadań administracyjnych, które pożerają mnóstwo czasu.

W książce znajdziesz informacje dotyczące różnych metod tworzenia skryptów oraz technik automatyzacji wszystkich typów zadań administracyjnych. Podstawą książki jest omówienie i zilustrowanie trzech głównych metod tworzenia skryptów – skryptów powłoki, KiXtarta i Windows Script Hosta. Dowiesz się również o zasadach działania Active Directory Service Interfaces (ADSI) i Windows Management Instrumentation, a także jak użyć przedstawionych przykładów do zarządzania organizacją. Poznasz również alternatywne metody, takie jak użycie ScriptIt i AutoIt w sytuacjach, kiedy konwencjonalne skrypty nie działają. Oprócz szczegółowych przykładów skryptów i informacji, ta książka opisuje dokładnie tworzenie skryptów dla systemów Windows NT i 2000.

Wydawnictwo Helion
ul. Chopina 6
44-100 Gliwice
tel. (32)230-98-63
e-mail: helion@helion.pl

Spis treści

Rozdział 4.

Automatyzacja Windows i aplikacji

Streszczenie

Z tego rozdziału dowiesz się najpierw, jak poprzez wiersz poleceń tworzyć skrypty dla aplikacji, apletów Panelu sterowania, systemu Windows oraz kreatorów. Potem przejdziemy do automatyzacji i tworzenia skryptów powłoki Windows oraz najbardziej typowych aplikacji (na przykład Word, Excel czy Internet Explorer). Nauczysz się także sposobu wysyłania klawiszy w celu automatyzacji aplikacji, które nie obsługują konwencjonalnych metod tworzenia skryptów. W kolejnych rozdziałach dowiesz się, jak wykonać konkretne zadania, takie jak dodawanie plików użytkowanych, sterowanie usługami lub archiwizacja.

Automatyzacja

Funkcja automatyzacji została początkowo zaprojektowana jako metoda wzajemnego dostępu i sterowania między aplikacjami. Automatyzacja aplikacji wywodzi się z mechanizmu dynamicznej wymiany danych (Dynamic Data Exchange — DDE), który przerodził się następnie w mechanizm łączenia i osadzania obiektów (Object Linking and Embedding — OLE) i automatyzację OLE, a potem po prostu został nazwany automatyzacją (Automation). Automatyzacja współpracuje z aplikacjami poprzez obiekty Component Object Model (COM). Obiekty COM to obiekty sterujące ActiveX, które zawierają oddzielne sekcje wielokrotnie wykorzystywanego kodu. Dzięki automatyzacji możesz tworzyć dokumenty, zapisywać pliki, odgrywać dźwięki, a nawet sterować systemem operacyjnym, jeśli tylko ma on model obiektu.

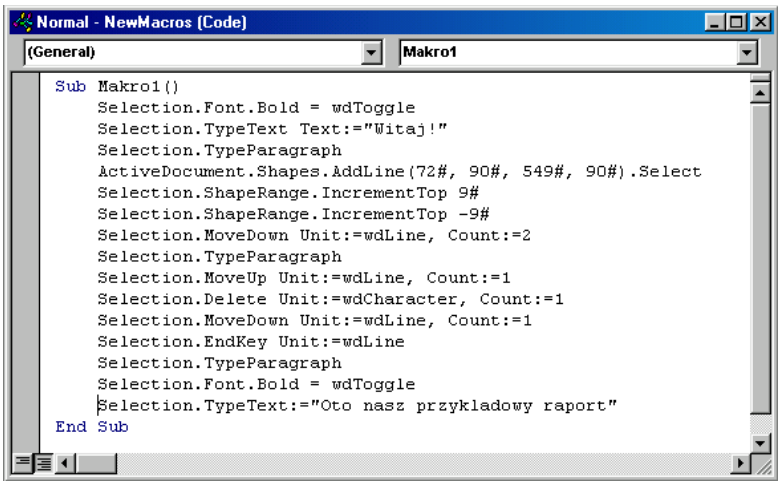
Visual Basic for Applications

Aplikacje Microsoft Office obsługują język skryptowy o nazwie Visual Basic for Applications (VBA). VBA, oparty na Visual Basicu, jest standardowym językiem programowania przeznaczonym do zdalnego sterowania funkcjami aplikacji Microsoft Office. Twórcy aplikacji mogą używać VBA do wywoływania ze swoich projektów funkcji innych aplikacji.

Aplikacje obsługujące VBA są znane jako „aplikacje dostosowywane" (customizable applications).

Typową metodą tworzenia prostego kodu VBA jest nagranie i edycja makra we wbudowanym edytorze Visual Basica. Aby nagrać nowe makro, uruchom aplikację Office i wybierz *Narzędzia|Makro|Zarejestruj nowe makro*. Po rozpoczęciu nagrywania wykonaj czynności, które chcesz zakodować, a następnie zatrzymaj nagrywanie. Uruchom teraz Edytor Visual Basic poprzez wybranie *Narzędzia|Makro|Edytor Visual Basic*. Po otwarciu edytora wybierz *Narzędzia|Makro*, wybierz swoje makro i kliknij *Edytuj*. Na rysunku 4.1 można zobaczyć przykładowy kod VBA wszystkich zarejestrowanych funkcji.

Rysunek 4.1. *Edycja nagranego makra Office*



Poprzez Windows Script Hosta możesz użyć VBScriptu do wywołania wielu funkcji VBA (w celu automatyzacji aplikacji pakietu Office). Automatyzacja aplikacji składa się z trzech etapów: uzyskiwania dostępu do obiektu aplikacji poprzez mechanizm Automation, sterowania aplikacją oraz zamknięcia obiektu aplikacji.

Uzyskiwanie dostępu do obiektu aplikacji

Obiekt aplikacji to obiekt najwyższego poziomu, który pozwala na wysyłanie danych do obiektu aplikacji i manipulacji programem poprzez niego. Jak już dowiedziałeś się w poprzednim rozdziale, aby uzyskać dostęp do obiektu, musisz najpierw użyć metody *CreateObject* i ustawić ją na zmienną:

```
Set zmienna = CreateObject("object.Application"}
```

Po utworzeniu egzemplarza możesz używać tej zmiennej w skrypcie, aby uzyskać dostęp do wszystkich metod zawartych w obiekcie. Oto lista najbardziej typowych identyfikatorów automatyzacji:

- `Access.Application` — używany do automatyzacji Microsoft Access,
- `Excel.Application` — używany do automatyzacji Microsoft Excel,
- `InternetExplorer.Application` — używany do automatyzacji Microsoft Internet Explorer,

- `Outlook.Application` — używany do automatyzacji Microsoft Outlook,
- `PowerPoint.Application` — używany do automatyzacji Microsoft PowerPoint,
- `Shell.Application` — używany do automatyzacji Microsoft Windows,
- `Word.Application` — używany do automatyzacji Microsoft Word.

Microsoft Office zawiera pliki pomocy, które opisują sposoby automatyzacji różnych aplikacji tego pakietu. Aby przejrzeć te pliki, uruchom program instalacyjny Office i zainstaluj pliki pomocy dla Visual Basica. Uruchom funkcję pomocy danej aplikacji i wyszukaj hasło „pomoc dla VBA".

Zmiana widzialności aplikacji

Po utworzeniu egzemplarza obiektu aplikacji większość obiektów znajduje się w trybie ukrytym. Pozwala to na manipulacje obiektem oraz wykonywanie różnych zadań, zanim stanie się on widoczny. Aby sprawić, by obiekt był widoczny, ustaw jego stan widzialności na True:

```
Zmienna.Visible = True
```

Analogicznie, obiekt możesz ukryć poprzez ustawienie stanu widzialności na `False`.

Zamykanie obiektu aplikacji

Po zakończeniu pracy z obiektem aplikacji powinno się go zamknąć, by umożliwić zwolnienie zasobów systemowych. Aby zamknąć obiekt aplikacji, wykonaj poniższe polecenie:

```
zmienna.Quit
```

Jeśli obiekt aplikacji nie zostanie poprawnie zamknięty, aplikacja ta pozostanie w pamięci niezależnie od jej widzialności czy użytkowania. Obiekty powinny zostać otwarte, tylko jeśli planujesz ich późniejsze użycie, na przykład jeśli chcesz użyć Microsoft Outlook do wysyłania alarmów do administratora.

ScriptIt a AutoIt

Niestety, nie wszystkie programy, dla których chcesz tworzyć skrypty, mają obiekt automatyzacji. W rozdziale 2. dowiedziałeś się, jak wykonać skrypty instalacji przy użyciu Microsoft ScriptIt. Microsoft ScriptIt to narzędzie, które odczytuje plik skryptowy zawierający proste polecenia tekstowe, a następnie wysyła klawisze do bieżąco aktywnego okna. AutoIt to aplikacja, której możesz użyć do utworzenia skryptów o większych możliwościach zamiast Microsoft ScriptIt.

Ograniczenia Microsoft ScriptIt

Microsoft ScriptIt to narzędzie o wielkości 808 kB, które potrafi jedynie wysyłać klawisze do aktywnych okien. Nie jest to język skryptowy, nie jest również możliwe używanie w nim poleceń typu `IF` lub `GOTO`. ScriptIt nie pozwala na przenoszenie okien, wysyłanie kliknięć myszki, edycję plików *INI* i rejestru, wyświetlanie komunikatów, wprowadzenie danych przez użytkownika, ani na wiele innych działań. Co gorsza, Microsoft nie oferuje wsparcia technicznego ani aktualizacji programu.

AutoIt na ratunek!

AutoIt to darmowe narzędzie przeznaczone do automatyzacji, o wielkości 59 kB, utworzone przez firmę HiddenSoft (*www.hiddensoft.com/autoit*), oferujące znacznie więcej funkcji niż Microsoft ScriptIt. Działając identycznie jak ScriptIt, AutoIt umożliwia również:

- dostęp do Schowka;
- dzięki wbudowanym zmiennym, pozwala na ustalenie wersji systemu operacyjnego, daty i informacji o skrypcie;
- tworzenie skryptów dla następujących działań:
  - uniemożliwienia użycia myszki i klawiatury,
  - wyświetlenia okien komunikatów i wprowadzania danych przez użytkownika,
  - manipulacji zmiennymi DOS,
  - manipulacji plikami tekstowymi i *INI*,
  - manipulacji rejestrem,
  - przesuwania i manipulacji oknami,
  - przesuwania kursora myszki i emulacji kliknięć,
  - wysyłania znaków ASCII,
  - wysyłania poleceń klawiszowych, takich jak PrintScreen, Break czy Windows,
  - zamykania Windows i wymuszania zamknięcia okna;
- wprowadzenie opcji „cichej" pracy,
- zastosowanie podprocedur, pętli i wyrażeń warunkowych.

Więcej informacji na temat użycia tego programu znajdziesz w dokumentacji AutoIt, dołączonej do programu instalacyjnego tego narzędzia.

Konwersja plików skryptowych na pliki EXE

Do pakietu instalacyjnego AutoIt dołączone jest narzędzie o nazwie *AUT2.EXE*, które dokonuje konwersji plików skryptowych AutoIt na pliki wykonywalne EXE. Dzięki konwersji skryptów możesz uniemożliwić odczyt kodu i modyfikację skryptów przez użytkowników. Narzędzie przeznaczone do konwersji jest sterowane przez menu i pozwala na wybór ikonki pliku wykonywalnego, która musi mieć wymiary 32 na 32 piksele w 16 kolorach.

Tworzenie skryptów obiektu sterującego ActiveX w AutoIt

Możesz użyć skryptowalnej wersji obiektu sterującego AutoIt ActiveX w Windows Script Hoście. Aby uzyskać dostęp do obiektu *AutoIt*, musisz najpierw użyć funkcji `CreateObject` i ustawić ją na zmienną:

```
Set zmienna = CreateObject("AutoItX.Control")
```

Więcej informacji na temat użycia tego obiektu znajdziesz w dokumentacji obiektu sterującego AutoIt ActiveX, dołączonej do programu instalacyjnego tego narzędzia.

Gotowe rozwiązania

Automatyzacja aplikacji poprzez wiersz poleceń

Większość aplikacji Windows obsługuje jakiś poziom skryptów powłoki. Początkowo miało to na celu zapewnienie wstecznej zgodności z plikami wsadowymi DOS, ale po powstaniu obiektów automatyzacji funkcja ta powoli „umiera". Sterowania aplikacjami z wiersza poleceń jest niezmiernie przydatne, jeśli chcesz wykonać proste zadania z poziomu pliku wsadowego DOS lub skrótu Windows.

Tworzenie skryptów Windows 9x Scandisk

Microsoft Windows 9x zawiera oparte na Norton Disk Doctor narzędzie Scandisk, które wyszukuje i naprawia błędy dyskowe, spowodowane zwykle zawieszeniem się systemu Windows lub jego niewłaściwe zamknięcie. Interfejs graficzny Scandiska to *SCANDSKW.EXE*, który wywołuje *DISKMAINT.DLL* w celu przeskanowania dysku. *SCANDSKW.EXE* obsługuje następujące opcje wiersza poleceń:

- `/ALLFIXEDDISKS` — skanuje wszystkie dyski lokalne,
- `/NONINTERACTIVE` — rozpoczyna skanowanie automatycznie,
- `/OLDNFS` — usuwa długie nazwy plików,
- `/PREVIEW` — uruchamia Scandiska w trybie podglądu,
- `/SILENT` — nie wyświetla ekranu podsumowującego.

Tworzenie skryptów skanowania systemu Windows 9x

Aby zautomatyzować skanowanie wszystkich dysków systemowych za pomocą Scandisku, wybierz *Start|Uruchom* i wprowadź poniższe polecenie:

```
SCANDSKW /ALLFIXEDDISKS /NONINTERACTIVE /SILENT
```

Aby zautomatyzować funkcję skanowania dla jednego dysku, wprowadź `scandskw` *dysk* `/NONINTERACTIVE`. *dysk* jest tutaj dyskiem, przeznaczonym do przeskanowania.

Użycie SCANDSKW.EXE do konwersji długich nazw na krótkie

Aby dokonać konwersji długich nazw plików do krótkich, wybierz *Start|Uruchom* i wprowadź poniższe polecenie:

```
SCANDSKW /OLDNFS
```

Ta konwersja jest nieodwracalna i nie powinna być wykonywana na dyskach systemowych.

Tworzenie skryptów Windows 9x Defrag

Kiedy plik lub folder jest aktualnie tworzony albo modyfikowany, części tego pliku lub folderu zostają rozrzucone po całym dysku twardym. To zjawisko znane jest jako fragmentacja dysku. Chociaż jest to proces naturalny, może spowodować spowolnienie czasu dostępu do danych znajdujących się na dysku. Reorganizacja tych plików lub folderów znacznie poprawia wydajność; czynność ta nazywana jest defragmentacją. Microsoft Windows 9x zawiera skryptowalne narzędzie przeznaczone do defragmentacji, które oparte jest o Norton Speed Disk. Dostępne opcje wiersza poleceń to:

- `/ALL` — defragmentuje wszystkie dyski lokalne,
- `/CONCISE` — wyświetla widok *Ukryj szczegóły* (Hide Details),
- `/DETAILED` — wyświetla widok *Pokaż szczegóły* (Show Details),
- *`dysk:`* — dysk, który ma być zdefragmentowany,
- `/F` — wykonuje defragmentację plików i wolnej przestrzeni,
- `/NOPROMPT` — tryb pracy bez kontroli (nie są wyświetlane zapytania),
- `/P` — optymalizuje pliki systemowe i ukryte,
- `/Q` — defragmentuje tylko wolną przestrzeń,
- `/U` — defragmentuje tylko pliki.

Tworzenie skryptów defragmentacji systemu Windows 9x

Poniższe polecenie automatycznie defragmentuje wszystkie dyski systemowe:

```
DEFRAG /ALL /P /CONCISE /NOPROMPT
```

Tworzenie skryptów Norton Antivirus 2000

Chociaż Norton Antivirus 2000 jest graficznym skanerem antywirusowym działającym w systemie Windows, obsługuje skrypty uruchamiane z wiersza poleceń. Podstawowa składnia skryptu linii systemowej wygląda w następujący sposób:

```
NAVW32.EXE ścieżka opcje
```

`ścieżka` oznacza tutaj dowolny dysk, folder, plik lub ich kombinację, która ma być przeskanowana, a *`opcje`* to dowolne, poprawne przełączniki wiersza poleceń, przekazywane do *NAVW32.EXE*. Oto lista dostępnych przełączników:

- `/A` — skanuje wszystkie napędy, z wyjątkiem napędów A i B. Dyski sieciowe będą skanowane, jeśli wybrano opcję *Allow Network Scanning*.
- `/L` — skanuje wszystkie dyski lokalne, z wyjątkiem napędów A i B.
- `/S` — skanuje wszystkie podane w ścieżce podkatalogi.
- `/M`*`opcja`* — włącza lub wyłącza skanowanie pamięci. *`opcja`* oznacza tutaj + (włączenie funkcji) lub – (wyłączenie funkcji).
- `/MEM` — skanuje tylko pamięć.

- /B*opcja* — włącza lub wyłącza skanowanie sektora rozruchowego. *opcja* oznacza tutaj + (włączenie funkcji) lub - (wyłączenie funkcji).
- /BOOT — skanuje tylko sektory rozruchowe.
- /NORESULTS — nie wyświetla wyników skanowania.
- /DEFAULT — resetuje ustawienia na wartości domyślne.
- /HEUR:*opcja* — ustawia wrażliwość skanowania heurystycznego. *opcja* może tutaj mieć wartość od 0 do 4, gdzie 4 to wartość najwyższa, a 0 oznacza wyłączenie funkcji.

Tworzenie skryptów FTP

FTP, czyli protokół transferu plików (File Transfer Protocol), jest typową metodą przenoszenia plików między dwiema lokalizacjami. Możesz użyć klienta FTP innej firmy (na przykład CuteFTP), jednak Microsoft FTP jest właściwie niezbędnym narzędziem przeznaczonym do transferu plików. Obsługuje on przełączniki wiersza poleceń, polecenia oraz pliki skryptowe. Przełączniki wiersza poleceń FTP decydują o sposobie uruchomienia klienta FTP. Najczęściej używane przełączniki wiersza poleceń to:

- -i — tryb interaktywny, w przypadku transferu wielu plików wyłącza zapytania interaktywne,
- -n — zapobiega automatycznemu logowaniu,
- -s:*skrypt* — podaje *skrypt* FTP, który ma być wykonany,
- -v — tryb rozszerzony, włącza funkcję statystyk transferu plików oraz odpowiedzi.

Aby uruchomić klienta FTP w trybie rozszerzonym i interaktywnym, przejdź do wiersza poleceń i wprowadź poniższe polecenie:

```
ftp -v -i
```

Po uruchomieniu klienta FTP możesz wprowadzać różne polecenia, które pozwalają na wyświetlenie listy plików, usunięcie, załadowanie i pobranie plików. Najczęściej używane polecenia FTP to:

- ascii — ustawiany domyślnie, ustawia typ transferu plików na ASCII (shar, uu),
- binary — ustawia typ transferu plików na binarny (z, arc, tar, zip),
- bye — kończy aktywną sesję FTP i zamyka program FTP,
- cd *katalog* — zmienia *katalog* w systemie zdalnym,
- close — kończy aktywną sesję FTP,
- delete *plik* — usuwa zdalny *plik*,
- get *plik* — pobiera ze zdalnego systemu pojedynczy *plik*,
- lcd *katalog* — zmienia *katalog* w systemie lokalnym,
- mdelete *pliki* — usuwa zdalne *pliki*,

- mget *pliki* — pobiera ze zdalnego systemu wiele *plików*,
- mput *pliki* — ładuje lokalne *pliki* na zdalny system,
- open *serwer* — ustanawia połączenie z *serwerem* o podanej nazwie,
- password *hasło* — podaje *hasło* dla wybranej nazwy konta,
- prompt — przełącza zapytania interaktywne,
- put *plik* — ładuje lokalny *plik* na zdalny system,
- user *nazwa* — podaje *nazwę* konta do połączenia ze zdalnym systemem.

Aby zobaczyć wszystkie dostępne przełączniki FTP, wpisz w wierszu poleceń `FTP -?`.

Tworzenie skryptów ładowania plików przez FTP

Codzienne ładowanie plików na serwer FTP jest typowym zadaniem administracyjnym. Aby utworzyć skrypt ładowania plików przez FTP, wybierz *Start|Uruchom* i wprowadź `FTP -I -S:`*`plik_skryptowy`*.

Przełącznik `-I` powoduje wyłączenie zapytań w czasie ładowania wielu plików, przełącznik `-S` wskazuje plik skryptowy do użycia, a *plik_skryptowy* to pełna ścieżka i nazwa pliku skryptowego, który zawiera następujące polecenia:

```
OPEN serwer_FTP
nazwa_użytkownika
hasło
CD katalog_FTP
LCD katalog_plików
MPUT pliki
BYE
```

serwer_FTP oznacza tutaj serwer, z którym należy się połączyć, *nazwa_użytkownika* i *hasło* to informacje potrzebne do zalogowania się, *katalog_FTP* to katalog na serwerze FTP, do którego należy załadować pliki, *katalog_plików* to katalog lokalny, w którym znajdują się pliki, a *pliki* to pliki do załadowania (na przykład *.*, .txt, dzienne.*).

Aby załadować jeden plik, zamień polecenie `MPUT` na `PUT`.

Tworzenie skryptów pobierania pliku przez FTP

Pobieranie plików z serwera FTP jest typowym zadaniem administratora. Aby utworzyć skrypt pobierania plików przez FTP, wybierz *Start|Uruchom* i wprowadź `FTP -I -S:` *`plik_skryptowy`*.

Przełącznik `-I` powoduje wyłączenie zapytań w czasie pobierania wielu plików, przełącznik `-S` wskazuje plik skryptowy do użycia, a *plik_skryptowy* to pełna ścieżka i nazwa pliku skryptowego, który zawiera następujące polecenia:

```
OPEN serwer_FTP
nazwa_użytkownika
hasło
CD katalog_FTP
LCD katalog_plików
MGET *.*
BYE
```

serwer_FTP oznacza tutaj serwer, z którym należy się połączyć, *nazwa_użytkownika* i *hasło* to informacje potrzebne do zalogowania się, *katalog_FTP* to katalog na serwerze FTP, w którym znajdują się pliki do pobrania, a *katalog_plików* to katalog lokalny, do którego należy pobrać pliki.

Tworzenie skryptów pobierania przez FTP plików aktualizacyjnych Norton Antivirus

Wielu administratorów przygotowało dysk sieciowy, na którym znajduje się najnowsza wersja plików aktualizacyjnych programu antywirusowego, a następnie, w celu pobrania tychże plików, skierowało tam programy antywirusowe użytkowników. Dzięki temu możliwe stało się wcześniejsze przetestowanie przez administratora najnowszych aktualizacji. Aby pobrać pliki aktualizacyjne Norton Antivirus na wspólny dysk sieciowy przy użyciu FTP i skryptów powłoki, wykonaj poniższe czynności:

1. Utwórz nowy katalog dla wszystkich plików, które są wykorzystywane w tym przykładzie.
2. Wybierz *Start*|*Uruchom* i wprowadź *plik_skryptowy.bat*.

plik_skryptowy oznacza tutaj pełną ścieżkę i nazwę pliku skryptowego, który zawiera poniższe polecenia:

```
@Echo Off
Net Use Z: \\serwer\dysk_sieciowy
ftp -n -s:skrypt_FTP >> dziennik
Net Use Z: /Delete
```

serwer oznacza tutaj system zawierający publiczny dysk sieciowy, na którym mają być zapisywane pliki aktualizacyjne programu antywirusowego, *dziennik* to pełna ścieżka i nazwa pliku tekstowego, w którym zapisywane będą informacje o transferze FTP, a *skrypt_FTP* to pełna ścieżka i nazwa pliku skryptowego, który zawiera poniższe polecenia:

```
open ftp.symantec.com
user anonymous
twoj_email@twoja_domena.com
lcd Z:\
cd \public\english_us_Canada\antivirus_definitions\norton_antivirus\static
bin
get sarci32.exe
bye
```

Wyróżniony kod musi znaleźć się w jednej linii.

Tworzenie skryptów pobierania plików aktualizacyjnych McAfee Antivirus przez FTP

Aby pobrać pliki aktualizacyjne McAfee Antivirus na wspólny dysk sieciowy przy użyciu FTP i skryptów powłoki, wykonaj poniższe czynności:

1. Utwórz nowy katalog dla wszystkich plików, które są wykorzystywane w tym przykładzie.
2. Wybierz *Start*|*Uruchom* i wprowadź *plik_skryptowy.bat*.

plik_skryptowy oznacza tutaj pełną ścieżkę i nazwę pliku skryptowego, który zawiera poniższe polecenia:

```
@Echo Off
Net Use Z: \\serwer\dysk_sieciowy
ftp -n -s:skrypt_FTP >> dziennik
Net Use Z: /Delete
```

serwer oznacza tutaj system zawierający publiczny dysk sieciowy, na którym mają być zapisywane pliki aktualizacyjne programu antywirusowego, *dziennik* to pełna ścieżka i nazwa pliku tekstowego, w którym zapisywane będą informacje o transferze FTP, a *skrypt_FTP* to pełna ścieżka i nazwa pliku skryptowego, który zawiera poniższe polecenia:

```
open ftp.nai.com
user anonymous
twoj_email@twoja_domena.com
lcd Z:\
cd \pub\antivirus\datiles\4.x
prompt
bin
mget *
bye
```

Powyższy skrypt pobiera aktualizacje dla McAfee VirusScan 4.x. Możesz zmienić wyróżnioną linię, jeśli chcesz uzyskać aktualizacje dla innej wersji tego programu.

Tworzenie skryptów apletów Panelu sterowania

CONTROL.EXE, znajdujący się w katalogu Windows, jest praktycznie Panelem sterowania Windows. Aby otworzyć *Panel sterowania*, wybierz *Start*|*Uruchom* i wprowadź `control`. Używając tego pliku wykonywalnego możesz uruchomić dowolny aplet Panelu sterowania.

Aplety Panelu sterowania są przechowywane jako pliki CPL (Control Panel). Aby wywołać aplet, wybierz *Start*|*Uruchom* i wprowadź `control aplet`. Jeden plik CPL może w rzeczywistości przechowywać wiele apletów. Aby wywołać kilka apletów z jednego pliku CPL, wybierz *Start*|*Uruchom* i wprowadź `control aplet`, `@#`. # oznacza tutaj numer apletu, który ma być wywołany. Jeśli nie podasz numeru apletu, *CONTROL.EXE* automatycznie otworzy pierwszy z nich (o numerze 0).

W przypadku apletów zawierających kilka zakładek możesz otworzyć wybraną zakładkę poprzez wybranie *Start*|*Uruchom* i wprowadzenie `control aplet, ,#`, gdzie # jest numerem zakładki przeznaczonej do otwarcia. Jeśli nie podasz numeru zakładki, *CONTROL.EXE* automatycznie otworzy pierwszą z nich (o numerze 0).

Dlaczego warto zainteresować się uruchamianiem apletów Panelu sterowania? Dzięki uruchomieniu tego pakietu możesz wykonywać dowolne zadanie za pomocą narzędzia do wysyłania klawiszy.

Aby odnaleźć w systemie wszystkie aplety i ich funkcje, wyszukaj pliki CPL i eksperymentuj, otwierając różne aplety i ich zakładki.

Modyfikacja właściwości myszki

Oto krótki przykład pokazujący użycie skryptów apletów Panelu sterowania w połączeniu z wysyłaniem klawiszy. Aby ustawić myszkę na opcję użycia właściwości przycisków dla osoby leworęcznej, wykonaj poniższe czynności:

1. Utwórz nowy katalog dla wszystkich plików, które są wykorzystywane w tym przykładzie.
2. Pobierz Microsoft ScriptIt z witryny *www.microsoft.com* do nowego katalogu i rozpakuj go.
3. Wybierz *Start*|*Uruchom* i wprowadź *ścieżka nowego katalogu*\scriptit *plik_skryptowy*.

ścieżka nowego katalogu oznacza tutaj pełną ścieżkę nowego katalogu utworzonego w kroku 1, a *plik_skryptowy* to plik tekstowy, który zawiera poniższe polecenia:

```
[SCRIPT]
RUN=CONTROL MOUSE.CPL
Mouse=~WINWAITACTIVE#!L{ENTER}
```

Tworzenie skryptów kreatorów i okien dialogowych

RUNDLL32.EXE to 32-bitowe narzędzie wiersza poleceń, które pozwala na wywoływanie funkcji z plików DLL. Pliki te zostały zaprojektowane tak, aby pozwalały na wywołania tego typu. Takie wywołania można dołączyć do skryptów, a następnie, w celu wykonania konkretnych zadań, użyć narzędzi do wysyłania klawiszy. Tabela 4.1 pokazuje najbardziej typowe wywołania RUNDLL32.

Automatyzacja aplikacji poprzez obiekt aplikacji

Większość nowych aplikacji zawiera skryptowalny model obiektu automatyzacji, który pozwala na wykonywanie skryptów przez użytkownika i inne aplikacje.

Użycie Microsoft Internet Explorer jako narzędzia do wyświetlania

Poza oknami dialogowymi i oknem DOS, Windows Script Host nie udostępnia żadnej metody wyświetlania komunikatów dla użytkowników. Możesz jednak użyć Microsoft Internet Explorer do wyświetlenia informacji dla użytkownika lub utworzenia dokumentów HTML. Aby wyświetlić zawartość *C:\TEMP* w Microsoft Internet Explorer, wykonaj następujące czynności:

Tabela 4.1. *Kreatory i okna dialogowe*

Zadanie	Wywołanie RUNDLL32
Dodaj nową drukarkę	`RUNDLL32.EXE SHELL32.DLL, SHHelpShortcuts_RunDLL AddPrinter`
Okna kaskadowo	`RUNDLL32.EXE USER.DLL,cascadechildwindows`
Skopiuj dyskietkę	`RUNDLL32.EXE DISKCOPY.DLL,DiskCopyRunDll`
Utwórz nową aktówkę	`RUNDLL32.EXE SYNCUI.DLL,Briefcase_Create`
Utwórz nowe połączenie dialup	`RUNDLL32.EXE RNAUI.DLL,RnaWizard @1`
Utwórz nowy plik współużytkowany	`RUNDLL32.EXE NTLANUI.DLL,ShareCreate`
Wyłącz klawiaturę	`RUNDLL32.EXE KEYBOARD,disable`
Wyłącz myszkę	`RUNDLL32.EXE MOUSE,diable`
Odłącz dysk sieciowy	`RUNDLL32.EXE USER.DLL,wnetdisconnectdialog`
Sformatuj dysk	`RUNDLL32.EXE SHELL32.DLL,SHFormatDrive`
Zainstaluj nowy modem	`RUNDLL32.EXE SHELL32.DLL,Control_RunDLL modem.cpl, ,add`
Wyloguj z Windows	`RUNDLL32.EXE SHELL32.DLL,SHExitWindowsEx 0`
Zarządzaj plikiem współużytkowanym	`RUNDLL32.EXE NTLANUI.DLL,ShareManage`
Mapuj dysk sieciowy	`RUNDLL32.EXE USER.DLL,wnetconnectdialog`
Otwórz folder czcionek	`RUNDLL32.EXE SHELL32.DLL,SHHelpShortcuts_RunDLL FontsFolder`
Otwórz folder drukarek	`RUNDLL32.EXE SHELL32.DLL,SHHelpShortcuts_RunDLL PrintersFolder`
Otwórz z…	`RUNDLL32.EXE SHELL32.DLL,OpenAs_RunDLL` *rozszerzenie*
Wydrukuj stronę testową	`RUNDLL32.EXE SHELL32.DLL,SHHelpShortcuts_RunDLL PrintTestPage`
Zrestartuj	`RUNDLL32.EXE SHELL32.DLL,SHExitWindowsEx 2`
Odśwież	`RUNDLL32.EXE USER.DLL,repaintscreen`
Zamknij Windows	`RUNDLL32.EXE USER.DLL,ExitWindows`
Zamknij Windows	`RUNDLL32.EXE SHELL32.DLL,SHExitWindows`
Zamknij Windows (wymuszenie)	`RUNDLL32.EXE KRNL386.EXE,exitkernel`
Zamień przyciski myszki	`RUNDLL32.EXE USER.DLL,swapmousebutton`
Ułóż okna w sąsiednim położeniu	`RUNDLL32.EXE USER.DLL,tilechildwindows`

1. Utwórz nowy katalog dla wszystkich plików, które są wykorzystywane w tym przykładzie.
2. Pobierz z witryny *www.microsoft.com* i zainstaluj w nowym katalogu najnowszą wersję Windows Script Hosta.
3. Wybierz *Start*|*Uruchom* i wprowadź `cscript` *`plik_skryptowy.vbs`*.

`plik_skryptowy` oznacza tutaj pełną ścieżkę i nazwę pliku skryptowego, który zawiera poniższe polecenia:

```
Set FSO = CreateObject("Scripting.FileSystemObject")
Set MSIE = CreateObject("InternetExplorer.Application")
sDIR = "C:\TEMP"
sTITLE = "Tworzenie listy plików katalogu…"
```

```
Set objDIR = GetFolder(sDIR)
SetupMSIE
MSIE.Document.Write "<HTML><TITLE>" & sTitle & _
"/TITLE><BODY bgcolor=#C0C0C0><FONT FACE=ARIAL>"
MSIE.Document.Write "<B>Wyświetlanie zawartości " & _
sDIR & ":</B><BR><BR>table border=0 width=100% " & _
"cellspacing=0 cellpadding=0>"
GoSubFolders objDIR
MSIE.Document.Write "</table><BR><B>Koniec listy</B> & _
"</FONT></BODY>"

Sub SetupMSIE
    MSIE.Navigate "About:Blank"
    MSIE.ToolBar = False
    MSIE.StatusBar = False
    MSIE.Resizable = False

Do
Loop While MSIE.Busy

SWidth = MSIE.Document.ParentWindow.Screen.AvailWidth
SHeight = MSIE.Document.ParentWindow.Screen.AvailHeight
MSIE.Width = SWidth/2
MSIE.Height = SHeight/2
MSIE.Left = (SWidth - MSIE.Width)/2
MSIE.Top = (SHeight - MSIE.Height)/2

MSIE.Visible = True
End Sub

Sub ListFiles (objDIR)
    For Each efile in objDIR.Files
            MSIE.Document.Write "<tr><td>" & efile & "</td>" & _
            "<td> </td><td align=right>" & efile.size & _
            "</td></tr>"
    Next
End Sub

Sub GoSubFolders (objDIR)
    If objDIR <> "\System Volume Information" Then
            ListFiles objDIR
            For Each eFolder in objDIR.SubFolders
                MSIE.Document.Write "<tr><td>" & _
                efolder & "</td><td>&lt;DIR&gt;<td><td> " & _
                "align=right>" & efolder.size & </td></tr>"
                GoSubFolders eFolder
            Next
    End If
End Sub
```

Aby skrypt ten zadziałał, musisz dołączyć do niego przedstawioną w rozdziale 3. procedurę *GetFolder*. W tym przykładzie okno nie będzie odświeżane, dopóki nie zostanie zakończony proces wyświetlania zawartości katalogu.

Tworzenie szczegółowych raportów w Microsoft Wordzie

Skryptów Microsoft Word możesz użyć w celu utworzenia dzienników i raportów Windows Script Hosta. Aby usunąć z systemu wszystkie tymczasowe pliki i zapisać te działania w dokumencie Microsoft Word, wykonaj poniższe czynności:

1. Utwórz nowy katalog dla wszystkich plików, które są wykorzystywane w tym przykładzie.
2. Pobierz z witryny *www.microsoft.com* i zainstaluj w nowym katalogu najnowszą wersję Windows Script Hosta.
3. Wybierz *Start*|*Uruchom* i wprowadź `cscript plik_skryptowy.vbs`.

plik_skryptowy oznacza tutaj pełną ścieżkę i nazwę pliku skryptowego, który zawiera poniższe polecenia:

```
Set FSO = CreateObject("Scripting.FileSystemObject")
Set WordApp = CreateObject("Word.Application")
sDIR = "C:\ "
sEXT = "TMP"
sTITLE = "Usuwanie plików"

WordApp.Documents.Add
WordApp.Visible = True
WordApp.Caption = sTITLE
WordApp.Selection.Font.Bold = True
WordApp.Selection.TypeText "Dziennik usuwania:" & " plików: " & _sEXT & " Pliki: "
WordApp.Selection.InsertDateTime
WordApp.Selection.Font.Bold = False
WordApp.Selection.TypeText vblf & vblf

Set objDIR = GetFolder(sDIR)
GoSubFolders objDIR
WordApp.Selection.Font.Bold = True
WordApp.Selection.TypeText vblf & "**KONIEC DZIENNIKA**"

Sub MainSub (objDIR)
    For Each eFile in objDIR.Files
           fEXT = FSO.GetExtensionName(efile.Path)
           If LCase(fEXT) = LCase(sEXT) Then
              DelFile efile
           End If
    Next
End Sub

Sub DelFile(sFILE)
    On Error Resume Next
    FSO.DeleteFile sFILE, True
    If ErrNumber <> 0 Then
           WordApp.Selection.TypeText "Błąd w czasie usuwania: " & _
           sFILE & vblf
    Else
           WordApp.Selection.TypeText "Usunięto: " & sFILE & vblf
      End If
End Sub
```

Aby skrypt ten zadziałał, musisz dołączyć do niego przedstawioną w rozdziale 3. procedurę *GetFolder*.

Tworzenie szczegółowych arkuszy w Microsoft Excel

Aby utworzyć arkusze poprzez Windows Script Hosta możesz użyć skryptów Microsoft Excel, aby usunąć z systemu wszystkie tymczasowe pliki i zapisać te działania w arkuszu Microsoft Excel, wykonaj poniższe czynności:

1. Utwórz nowy katalog dla wszystkich plików, które są wykorzystywane w tym przykładzie.
2. Pobierz z witryny *www.microsoft.com* i zainstaluj w nowym katalogu najnowszą wersję Windows Script Hosta.
3. Wybierz *Start|Uruchom* i wprowadź `cscript plik_skryptowy.vbs`.

plik_skryptowy oznacza tutaj pełną ścieżkę i nazwę pliku skryptowego, który zawiera poniższe polecenia:

```
Set FSO = CreateObject("Scripting.FileSystemObject")
Set ExcelApp = CreateObject("Excel.Application")
Row = 1
Column = 1
ExcelApp.Workbooks.Add
ExcelApp.Visible = True

sDIR = "C:\ "
sEXT = "TMP"
sTITLE = "Usuwanie plików"

ExcelApp.Caption = sTITLE
ExcelApp.Range("A1").Select
ExcelApp.Selection.Font.Bold = True
ExcelApp.Cells(Row,Column).Value "Dziennik usuwania:" & sEXT & _" Pliki"
Row = Row + 1
Set objDIR = GetFolder(sDIR)
GoSubFolders objDIR
ExcelApp.Selection.Font.Bold = True
Row = Row + 1
ExcelApp.Cells(Row,Column).Value = "**KONIEC DZIENNIKA**"

Sub MainSub (objDIR)
  For Each eFile in objDIR.Files
    fEXT = FSO.GetExtensionName(efile.Path)
    If LCase(fEXT) = LCase(sEXT) Then
      DelFile efile
    End If
  Next
End Sub

Sub GoSubFolders (objDIR)
  If objDIR <> "\System Volume Information" Then
    MainSub objDIR
    For Each eFolder in objDIR.SubFolders
      GoSubFolders eFolder
```

```
    Next
  End If
End Sub

Sub DelFile(sFILE)
  On Error Resume Next
  FSO.DeleteFile sFILE, True
  If ErrNumber <> 0 Then
  ExcelApp.Cells(Row,Column).Value = "Błąd w czasie usuwania: " & _ sFILE
  Else
  ExcelApp.Cells(Row,Column).Value = "Usunięto: " & sFILE
  End If
  Row = Row + 1
End Sub
```

Aby skrypt ten zadziałał, musisz dołączyć do niego przedstawioną w rozdziale 3. procedurę *GetFolder*.

Tworzenie skryptów powłoki Windows

Windows ma własny obiekt automatyzacji, nazwany *shell.automation*. Chociaż można przypuszczać, iż umożliwia on pełną automatyzację każdej funkcji Windows, tak naprawdę pozwala on tylko na sterowanie ograniczonym zestawem obiektów dostępnych dla skryptów. Aby uzyskać dostęp do powłoki Windows, musisz utworzyć egzemplarz obiektu powłoki w następujący sposób:

```
Set zmienna = CreateObject("Shell.Application")
```

Sterowanie systemem Windows

W Microsoft Windows element jest otwierany w oknie systemowym. Standardowymi elementami sterującymi okna są funkcję minimalizacji i maksymalizacji. Możliwe jest tworzenie skryptów dla tych i innych poleceń Windows poprzez obiekt powłoki Windows. Poniżej znajduje się lista obiektów okna i ich funkcje:

- `CascadeWindows` — kaskada otwartych okien,
- `MinimizeAll` — minimalizacja otwartych okien,
- `TileHorizontally` — układa okna poziomo,
- `TileVertically` — układa okna pionowo,
- `UndoMinimizeAll` — przywraca zminimalizowane okna.

Aby wywołać te metody, wykonaj następujące polecenie:

```
Set Shell = CreateObject("Shell.Application")Shell.Method
```

Wyszukiwanie folderów

Korzystając z metody *BrowseForFolder*, możesz wykorzystać typowe okno dialogowe *Browse For Folder Windows*, stosowane w większości aplikacji Windows. Aby wywołać to okno dialogowe, wykonaj następujące czynności:

1. Utwórz nowy katalog dla wszystkich plików, które są wykorzystywane w tym przykładzie.
2. Pobierz z witryny *www.microsoft.com* i zainstaluj w nowym katalogu najnowszą wersję Windows Script Hosta.
3. Wybierz *Start*|*Uruchom* i wprowadź `cscript plik_skryptowy.vbs`.

plik_skryptowy oznacza tutaj pełną ścieżkę i nazwę pliku skryptowego, który zawiera poniższe polecenia:

```
Set Shell = CreateObject("Shell.Application")
Set Folder = Shell.BrowseForFolder (handle, "Nazwa", opcje, katalog_macierzysty)
Wscript.Echo "FOLDER: " & Folder.Title & vblf & _
"MACIERZYSTY: " & Folder.ParentFolder
```

katalog_macierzysty może być tutaj ścieżką katalogu lub folderem specjalnym.

Tabela 4.2 przedstawia foldery specjalne.

Uruchamianie apletu Panelu sterowania

Panel sterowania zawiera różne aplety, które możesz wykorzystać do wykonania wielu zadań. Aplety mają rozszerzenie *.cpl* i znajdują się w katalogu systemowym. Aby wywołać aplet Panelu sterowania poprzez obiekt automatyzacji powłoki, wykonaj następujące polecenia:

```
Set Shell = CreateObject("Shell.Application")
Shell.ControlPanelItem "aplet.cpl"
```

Odłączanie PC

Aby odłączyć komputer przenośny za pomocą obiektu automatyzacji powłoki, wykonaj następujące polecenia:

```
Set Shell = CreateObject("Shell.Application")
Shell.EjectPC
```

Eksploracja folderu

Aby dokonać eksploracji obiektu za pomocą obiektu automatyzacji powłoki, wykonaj następujące polecenia:

```
Set Shell = CreateObject("Shell.Application")
Shell.Explore katalog_macierzysty
```

katalog_macierzysty może być tutaj ścieżką katalogu lub folderem specjalnym.

Otwarcie folderu

Aby otworzyć folder za pomocą obiektu automatyzacji powłoki, wykonaj następujące polecenia:

Tabela 4.2. *Foldery specjalne*

Stała	Folder lub ścieżka katalogu
&H0	Pulpit wszystkich użytkowników
&H2	Folder programów wszystkich użytkowników
&H3	Panel sterowania
&H4	Folder Drukarki
&H5	Folder Osobiste
&H6	Folder Ulubione
&H7	Folder Autostart
&H8	Folder Ostatnie
&H9	Folder Wyślij do
&Ha	Kosz
&Hb	Menu Start
&H10	Katalog Pulpit
&H11	Dyski (Mój komputer)
&H12	Otoczenie sieciowe
&H13	Folder Czcionki
&H14	Folder Szablony
&H15	Wspólne menu Start
&H16	Wspólny folder Programy
&H17	Wspólny folder Programy
&H18	Wspólny folder Autostart
&H19	Wspólny katalog Pulpit
&H1a	Folder Dane aplikacji
&H1b	Folder Drukarki
&H1c	Lokalny folder Dane aplikacji
&H1d	Alternatywny folder Autostart
&H1e	Wspólny alternatywny folder Autostart
&H1f	Wspólny folder Ulubione
&H20	Wspólny folder Internet Cache
&H21	Wspólny folder Cookies
&H22	Folder Historia
&H23	Wspólny folder Dane aplikacji
&H24	Folder Windows
&H25	Folder System
&H26	Folder Plików programów
&H27	Folder Moje zdjęcia
&H28	Folder Profil

```
Set Shell = CreateObject("Shell.Application")
Shell.Open katalog_macierzysty
```

katalog_macierzysty może być tutaj ścieżką katalogu lub folderem specjalnym.

Wywoływanie systemowych okien dialogowych

Systemowe okna dialogowe to okna, które wymagają działania użytkownika; są to na przykład okna *Znajdź pliki* lub *Uruchom*. Wywołanie takich okien dialogowych możliwe jest poprzez skrypt, a także wysłanie do nich klawiszy w celu wykonania typowych zadań użytkownika. Aby wywołać systemowe okno dialogowe za pomocą obiektu automatyzacji powłoki, wykonaj następujące polecenia:

```
Set Shell = CreateObject("Shell.Application")
Shell.SysDialog
```

`SysDialog` obejmuje tutaj następujące metody:

- `FileRun` — wywołuje okno dialogowe *Start|Uruchom*,
- `FindComputer` — wywołuje okno dialogowe *Start|Znajdź|Komputer*,
- `FindFiles` — wywołuje okno dialogowe *Start|Znajdź|Pliki lub foldery*,
- `SetTime` — wywołuje okno dialogowe *Data/Czas*,
- `ShutdownWindows` — wywołuje okno dialogowe *Start|Zamknij*,
- `TrayProperties` — wywołuje okno dialogowe *Właściwości paska zadań*.

Odświeżanie menu Start

Aby odświeżyć zawartość menu *Start* za pomocą obiektu automatyzacji powłoki, wykonaj następujące polecenia:

```
Set Shell = CreateObject("Shell.Application")
Shell.RefreshMenu
```

Zawieszanie komputera

Większość komputerów przenośnych ma funkcję zawieszania komputera, powodująca jego przejście do trybu oszczędzania energii, w przypadku gdy urządzenie nie jest używane. Aby zawiesić komputer za pomocą obiektu automatyzacji powłoki, wykonaj następujące polecenia:

```
Set Shell = CreateObject("Shell.Application")
Shell.Suspend
```

Łączenie się z przestrzenią nazw folderu

W rozdziale 2. dowiedziałeś się, jak połączyć się z folderem za pomocą metody `GetFolderFileSystemObject`. Aby połączyć się z folderem przy użyciu obiektu automatyzacji powłoki, użyj metody `NameSpace` i wykonaj następujące polecenia:

```
Set Shell = CreateObject("Shell.Application")
Set Folder = Shell.NameSpace(katalog_macierzysty)
```

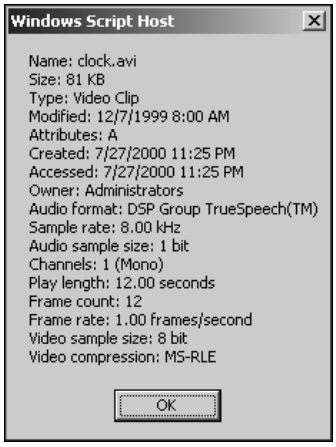
Uzyskiwanie szczegółów pliku lub folderu

Windows NT i 9x przechowują tylko podstawowe informacje o plikach i folderze, o wiele więcej informacji można odnaleźć w Windows 2000. Aby uzyskać informacje o podanym pliku lub folderze, możesz użyć metody obiektu folderu `GetDetailsOf` na wszystkich systemach operacyjnych. Aby połączyć się z folderem za pomocą obiektu automatyzacji powłoki, użyj metody `NameSpace` i wykonaj następujące polecenia:

```
Set Shell = CreateObject("Shell.Application")
Set Folder = Shell.NameSpace(katalog_macierzysty)
For Each Item in Folder.Items
  Summary = "Nazwa: " & Item.Name & vblf
  For Count = 1 to 37
    On Error Resume Next
    Detail = Folder.GetDetailsOf(Item.Count)
    If Detail <> "" Then
      Summary = Summary & Folder.GetDetailsOf(0,Count) & _ ": " &
Folder.GetDetailsOf(Item,Count) & vblf
    End If
  Next
  Wscript.Echo Summary
Next
```

katalog_macierzysty może być tutaj ścieżką katalogu lub folderem specjalnym. Efekt działania skryptu może być podobny do pokazanego na rysunku 4.2.

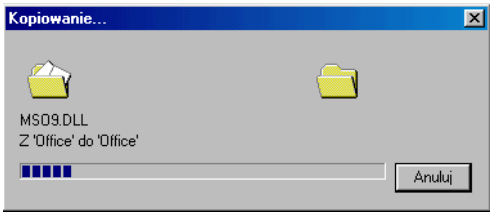
Rysunek 4.2. *Efekt działania metody GetDetailsOf dla pliku*



Kopiowanie i przenoszenie plików i folderów

Przy każdym kopiowaniu lub przenoszeniu pliku w Windows pojawiają się graficzne okna dialogowe, zawierające pasek postępu oraz potwierdzenie operacji (zobacz rysunek 4.3).

Rysunek 4.3. *Okno dialogowe kopiowania pliku w Windows*



Chociaż obiekt FileSystemObject może wykonać operacje zarządzania plikami, nie wyświetla żadnego z tych okien dialogowych. Aby użyć takich okien dialogowych w skryptach, należy skorzystać z obiektu automatyzacji powłoki. Aby skopiować lub przenieść pliki i foldery do innego folderu, wykonaj następujące polecenia:

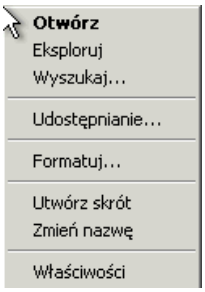
```
Set Shell = CreateObject("Shell.Application")
Set Folder = Shell.NameSpace(katalog_macierzysty)
Folder.Metoda "pliki", znaczniki
```

katalog_macierzysty może być tutaj ścieżką katalogu lub folderem specjalnym, *Metoda* to metoda folderu CopyHere lub MoveHere, *pliki* to pliki lub foldery, które należy skopiować lub przenieść, a *znaczniki* to opcjonalne parametry, które sterują operacją na plikach. Możliwe jest łączenie wielu parametrów za pomocą znaku +.

Uzyskiwanie dostępu do menu kontekstowego

Po każdym kliknięciu pliku prawym przyciskiem myszki (w przypadku myszki dla osoby praworęcznej) wywoływane jest menu kontekstowe. To menu pełne jest poleceń dodanych przez system, media i dowolne zainstalowane programy (zobacz rysunek 4.4).

Rysunek 4.4. *Menu kontekstowe Windows*



Dostęp do tych poleceń można uzyskać poprzez ich kliknięcie lub naciśnięcie kombinacji klawiszy skrótu (*Alt+podświetlona litera*). Dzięki automatyzacji powłoki możliwa jest aktywacja dowolnego polecenia:

```
Set Shell = CreateObject("Shell.Application")
Set Folder = Shell.NameSpace(katalog_macierzysty)
Set File = Folder.ParseName("plik")
File.InvokeVerb("polecenie")
```

katalog_macierzysty może być tutaj ścieżką katalogu lub folderem specjalnym, *plik* to dowolny w plik w *katalogu_macierzystym*, a *polecenie* to dowolne polecenie, znajdujące się w menu kontekstowym.

Istnieją dwie istotne kwestie, o których należy pamiętać używając polecena InvokeVerb *zadanie*. Po pierwsze, jeśli polecenie wykorzystuje klawisze skrótu, musisz poprzedzić odpowiednią literę znakiem &. Dla przykładu, aby uruchomić polecenie *Otwórz* z rysunku 4.4, należy wprowadzić &Open. Po drugie, jeśli dane polecenie wyświetla okno systemowe (takie jak okno właściwości), zostanie ono zamknięte natychmiast po zakończeniu pracy przez skrypt.

Automatyzacja aplikacji poprzez wysyłanie klawiszy

Niektóre aplikacje zostały specjalnie zaprojektowane tak, aby nie obsługiwać opcji wiersza poleceń lub modeli obiektów automatyzacji. Wobec braku możliwości użycia skryptów do wysyłania poleceń do takich programów, należy skorzystać z narzędzi do wysyłania klawiszy.

Tworzenie skryptów defragmentacji dysku przez Diskeepera Lite

Kiedy Diskeeper Lite wykryje jakąkolwiek próbę użycia skryptu (poprzez uruchomienie z pliku wsadowego lub bezpośrednie wywołanie ze skryptu), program natychmiast się wyłącza. Alternatywną metodą tworzenia skryptów dla Diskeepera Lite jest wysyłanie klawiszy. Dzięki metodzie, aplikacji wydaje się, że polecenia są wydawane przez użytkownika, a nie przez skrypt. Aby zautomatyzować defragmentację dysku przy użyciu Diskeepera Lite, wykonaj poniższe czynności:

1. Utwórz nowy katalog dla wszystkich plików, które są wykorzystywane w tym przykładzie.
2. Pobierz i zainstaluj Diskeepera Lite 1.1 (szczegóły znajdziesz w rozdziale 2.).
3. Z witryny *www.hiddensoft.com/autoit* pobierz i zainstaluj AutoIt.
4. Wybierz *Start*|*Uruchom* i wprowadź `autoit2 plik_skryptowy`.

autoit2 oznacza tutaj pełną ścieżkę i nazwę pliku wykonywalnego *autoit*, a *plik_skryptowy* to plik tekstowy, który zawiera poniższe polecenia:

```
SEND, {LWIN}R
SEND, "katalog_instalacji"{ENTER}
WINWAITACTIVE, Diskeeper Lite+Tree View+Fragmented Files
SEND, {ALTDOWN}D{ALTUP}D
WINWAITACTIVE, Select Drive To Defragment
SEND, {ALTDOWN}O
WINWAITACTIVE, Defragmentation Completed
SEND, {ALTDOWN}O
WINWAITACTIVE, Diskeeper Lite+Tree View+Fragmented Files
SEND, {ALTDOWN}F{ALTUP}X
```

katalog_instalacji oznacza tutaj katalog, w którym zainstalowano Diskeepera Lite.

Zauważ, że plik *DKLITE.EXE* nie jest uruchamiany bezpośrednio, ale poprzez polecenie Windows *URUCHOM*.

Tworzenie skryptów defragmentacji dysku w Windows 2000

Windows 2000 zawiera specjalną, okrojoną wersję Diskeepera firmy Executive Software, przygotowaną specjalnie dla Windows 2000. Podobnie jak Diskeeper Lite, narzędzie do defragmentacji w Windows 2000 nie umożliwia użycia funkcji skryptów i terminarzy, obecnych w pełnej wersji aplikacji. Aby utworzyć skrypt defragmentacji dysku w Windows 2000, wykonaj poniższe czynności:

1. Z witryny *www.hiddensoft.com/autoit* pobierz i zainstaluj AutoIt.
2. Wybierz *Start*|*Uruchom* i wprowadź `autoit2 plik_skryptowy`.

autoit2 oznacza tutaj pełną ścieżkę i nazwę pliku wykonywalnego *autoit*, a *plik_skryptowy* to plik tekstowy, który zawiera poniższe polecenia:

```
Run, defragmmc
Winwaitactive, Disk Defrag
Send, {ALTDOWN}A{ALTUP}D
Winwaitactive, Defragmentation Complete
Send, {TAB}{ENTER}
Winwaitactive, Disk Defrag
Send, {ALTDOWN}{F4}{ALTUP}
```

defragmmc oznacza tutaj pełną ścieżkę do pliku *DFRG.MMC*, który zwykle znajduje się w katalogu *Winnt\System32*.

Zmiana domyślnej strony początkowej Internet Explorera

Aby zmienić domyślną stronę początkową Internet Explorera, wykonaj poniższe czynności:

1. Z witryny *www.hiddensoft.com/autoit* pobierz i zainstaluj AutoIt.
2. Wybierz *Start*|*Uruchom* i wprowadź `autoit2 plik_skryptowy`.

autoit2 oznacza tutaj pełną ścieżkę i nazwę pliku wykonywalnego *autoit*, a *plik_skryptowy* to plik tekstowy, który zawiera poniższe polecenia:

```
Run, control.exe inetcpl.cpl
WinWaitActive, Internet Properties
Send, http://www.jesseweb.com{Enter}
```

Zmiana ustawień identyfikacji sieciowej (tylko w Windows 9x)

Aby zmienić ustawienia identyfikacji sieciowej w Windows 9x, wykonaj poniższe czynności:

1. Z witryny *www.hiddensoft.com/autoit* pobierz i zainstaluj AutoIt.
2. Wybierz *Start*|*Uruchom* i wprowadź `autoit2 plik_skryptowy`.

autoit2 oznacza tutaj pełną ścieżkę i nazwę pliku wykonywalnego *autoit*, a *plik_skryptowy* to plik tekstowy, który zawiera poniższe polecenia:

```
Run, control.exe netcpl.cpl
WinWaitActive, Network
Send, {Ctrldown}{Tab}{Ctrlup}
Send, Nowa_nazwa_komputera{Tab}
Send, Nowa_grupa_robocza{Tab}
Send, Nowy_opis{Enter}
```

Przeglądanie Internetu

Jeśli twój dostawca Internetu ciągle cię rozłącza albo używasz programu, który przerywa aktywne połączenie internetowe, powinieneś skorzystać z opisanego poniżej sposobu utrzymania aktywnego połączenia. Aby przeglądać witryny internetowe bez przerwy, wykonaj poniższe czynności:

1. Z witryny *www.hiddensoft.com/autoit* pobierz i zainstaluj AutoIt.
2. Wybierz *Start*|*Uruchom* i wprowadź `autoit2 plik_skryptowy`.

autoit2 oznacza tutaj pełną ścieżkę i nazwę pliku wykonywalnego *autoit*, a *plik_skryptowy* to plik tekstowy, który zawiera poniższe polecenia:

```
SetTitleMatchMode, 2
Run, C:\\Program Files\Internet Explorer\\Iexplore.exe
WinWaitActive, Microsoft Internet Explorer
Repeat
Send, {ALTDOWN}D{ALTUP}www.jesseweb.com{Enter}
Sleep, 1000
Send, {ALTDOWN}D{ALTUP}www.fightclub.com{Enter}
Sleep, 1000
Send, {ALTDOWN}D{ALTUP}www.tylerandjacks.com{Enter}
Sleep, 1000
Send, {ALTDOWN}D{ALTUP}www.napster.com{Enter}
Sleep, 1000
Send, {ALTDOWN}D{ALTUP}www.audiofind.com{Enter}
Sleep, 1000
EndRepeat
```

Oczyszczanie pamięci podręcznej Microsoft Internet Explorer

Internet Explorer umieszcza w pamięci podręcznej witryny internetowe, wprowadzane nazwy użytkowników, hasła i formularze. Aby usunąć te elementy za pomocą obiektu sterującego AutoIt ActiveX, wykonaj poniższe czynności:

1. Z witryny *www.hiddensoft.com/autoit* pobierz i zainstaluj AutoIt.
2. Wybierz *Start|Uruchom* i wprowadź `autoit2 plik_skryptowy`.

autoit2 oznacza tutaj pełną ścieżkę i nazwę pliku wykonywalnego *autoit*, a *plik_skryptowy* to plik tekstowy, który zawiera poniższe polecenia:

```
Set Shell = WScript.CreateObject("WScript.Shell")
Set AIT = WScript.CreateObject("AutoItX.Control")

Shell.Run "control.exe inetcpl.cpl", 1, FALSE
AIT .WinWaitActive "Internet Properties", ""
AIT .Send "{ALTDOWN}F{ALTUP}"
AIT .WinWaitActive "Delete Files", ""
AIT .Send "{TAB}{ENTER}"
AIT .WinWaitActive "Internet Properties", ""
AIT .WinClose "Internet Properties", ""

Shell.Run "control.exe inetcpl.cpl, ,2", 1,  FALSE
AIT .WinWaitActive "Internet Properties", ""
AIT .Send "{ALTDOWN}U{ALTUP}"
AIT .WinWaitActive "AutoComplete Settings", ""
AIT .Send "{ALTDOWN}C{ALTUP}"
AIT .WinWaitActive "Internet Options", ""
AIT .Send "{ENTER}"
AIT .WinWaitActive "AutoComplete Settings", ""
AIT .Send "{ALTDOWN}L{ALTUP}"
AIT .WinWaitActive "Internet Options", ""
AIT .Send "{ENTER}{ESC}"
AIT .WinWaitActive "Internet Properties", ""
AIT .Send "{ESC}"

WScript.Quit
```