Event log

Takmer každá aplikácia potrebuje z času na čas informovať užívateľa, prípadne administrátora o nejakej špeciálnej udalosti. V prípade serverových aplikácií však nie je možné priamo komunikovať s užívateľom. V takomto prípade je vhodné prípadnú udalosť zaznamenať vo forme logu. Tu sa však vynára otázka, akým spôsobom logovať. Vo Windows existuje spôsob zjednocujúci logovanie a uľahčujúci prácu administrátorom, ktorí nemusia ovládať spôsob logovania každej nainštalovanej aplikácie. Reč je o tzv. event logu, štandardnom mechanizme pre zaznamenávanie udalostí. Event log ponúka štandardný formát logov a administrátorovi stačí jednoduchá aplikácia na ich prechádzanie.

Čo je event log?

Z pohľadu systémového administrátora je event log niečo viac, než len zoznam správ vytvorený systémom alebo nejakou aplikáciou. Zoznam správ je organizovaný do logických skupín zvaných *log files* (logy). Súbor logov sa označuje ako *event log*. Administrátor môže prezerať event log pomocou Event Viewer snap-inu v Microsoft Management Console (MMC).

Implicitne event log systému obsahuje 3 logy: *Application, System* a *Security*. Logy *System* a *Application* sú pre použitie operačného systému, prípadne Windows aplikácií. Do logu *Security* môže priamo zapisovať iba služba *Local Security Authority Subsystem Service (lsass.exe)*. Pre logovanie vašej aplikácie môžete použiť log *Application*, alebo si vytvoriť vlastný log.

V každom logu sa nachádzajú jednotlivé udalosti. Udalosť je samostatný záznam v logu, ktorý pozostáva z nasledujúcich položiek: typ udalosti, dátum a čas vygenerovania, dátum a čas zápisu, zdroj udalosti, kategória udalosti, ID udalosti, užívateľ a systém. Okrem toho ešte môže obsahovať podrobný textový popis a môžu k nemu byť asociované binárne data.

Väčšina z týchto položiek je samovysvetľujúca, ale položky zdroj udalosti, ID udalosti, kategória udalosti a typ udalosti si zaslúžia lepšie vysvetlenie.

Zdroj udalosti reprezentuje aplikáciu, službu alebo komponent systému, ktorý udalosť nahlásil. Typicky sa medzi nahlasujúcim agentom a zdrojom udalosti jedná o vzťah jeden na jedného. Zdroj udalosti je však volený kódom vašej aplikácie, takže jedna aplikácia môže hlásiť udalosti aj niekoľkých zdrojov a naopak, jeden zdroj môže byť použitý pre viac aplikácií.

ID udalosti je hodnota definovaná zdrojom udalosti, ktorá identifikuje určitý typ udalosti. Akákoľvek udalosť môže byť identifikovaná pomocou zloženia zdroja a ID udalosti.

Kategória udalosti je voliteľná kategória udalosti definovaná zdrojom udalosti. Užitočná je pri aplikáciách s veľkým počtom udalostí rôznych typov, ktoré sú týmto spôsobom rozdelené do logických kategórií. Či sú kategórie nevyhnutné, prípadne užitočné pre vašu aplikáciu, sa musíte rozhodnúť sami.

Typ udalosti je jeden z nasledujúcich možných:

- `EVENTLOG_INFORMATION_TYPE` – udalosti tohto typu oznamujú situáciu alebo operáciu, ktorá nie je problematická pre aplikáciu, prípadne systém. Napríklad štart a ukončenie služby.

- `EVENTLOG_WARNING_TYPE` – udalosti tohto typu naznačujú potenciálnu problémovú situáciu. Napríklad relatívne málo voľnej pamäte na disku.

- `EVENTLOG_ERROR_TYPE` – udalosti typu *error* sú zalogované, keď zlyhá funkčnosť časti aplikácie, prípadne systému. Napríklad neschopnosť zapísať dáta na disk vedúca ku strate dát.

- `EVENTLOG_AUDIT_SUCCESS` – udalosti typu audit success sú logované po úspešnom prevedení auditu.

- `EVENTLOG_AUDIT_FAILURE` – udalosti typu audit failure sú logované ak pokus o audit zlyhá.

Hlásenie udalostí

Skôr, než začnete logovať udalosti do event logu, by ste si mali dobre premyslieť, ktoré udalosti logovať. Pri udalostiach typu *error* a *warning* je to viacmenej jasné. Horšie je to s udalosťami typu *information*. V tomto prípade si musíte dávať pozor, aby ste nezahltili event log prílišným množstvom udalostí. Typicky je vhodné logovať tieto udalosti, ak nie sú v aplikácií bežné, prípadne pri zmene stavu pri službách,... Pri rozhodovaní, ktoré udalosti logovať, pomáha aj pozrieť sa na vec očami administrátora.

Ako hlásiť udalosti

Skôr než váš proces začne hlásiť správy do event logu, musí zaregistrovať zdroj udalostí, a to volaním funkcie *`RegisterEventSource`*:

```
HANDLE RegisterEventSource(
   PCTSTR machineName,
   PCTSTR sourceName
   );
```

Parameter *`machineName`* identifikuje systém, ktorý obsahuje logy, ku ktorým chcete pripojiť záznamy o udalostiach. Ak tento parameter zvolíte `NULL`, je otvorený log súbor na lokálnom počítači. Málokedy sú správy hlásené do logov vzdialenému počítaču.

Parameter *`sourceName`* je pomenovanie zdroja udalosti. Väčšinou je ním meno vašej aplikácie, ale nie je to podmienkou.

Ak je funkcia *`RegisterEventSource`* úspešná, vráti handle, ktorú vaša aplikácia použije pri hlásení udalostí. Ak túto *handle* už nepotrebujete, musíte ju zatvoriť, a to volaním funkcie *`DeregisterEvent-Source`*:

```
BOOL DeregisterEventSource(HANDLE hEventLog);
```

Samotné hlásenie správ je realizované volaním funkcie *`ReportEvent`*:

```
BOOL ReportEvent(
   HANDLE hEventLog,
   WORD eventType,
   WORD eventCategory,
   DWORD eventID,
   PSID userSid,
   WORD numStrings,
   DWORD dataSize,
   PCTSTR* strings,
   PVOID rawData
   );
```

Dokumentáciu k tejto funkcii nájdete tu.

Spomínané funkcie sú všetko, čo potrebujete vedieť k tomu, aby ste hlásili udalosti do event logu. Technicky je to pravda, avšak pár kľúčových bodov je vynechaných. Keď porovnáte údaje, ktoré odovzdáte funkcii `ReportEvent` a údaje, ktoré zobrazuje Event Viewer snap-in, zistíte, že niektoré dáta chýbajú. Takisto ani jedna zo spomínaných funkcií nešpecifikuje, do ktorého logu chceme udalosti hlásiť (*Application, System* alebo *Security*). Funkcii `ReportEvent` takisto chýba parameter obsahujúci detailný popis udalosti.

Spomínaný prístup imlicitne umiestňuje udalosti do *Application* logu. Log obsahuje všeobecný text pre detailný popis udalosti a na záver tohto popisu pripojí vaše dáta.

Message file súbory

Kvôli jazykovej nezávislosti sú človeku zrozumiteľné časti popisu udalosti – kategória a detailný popis – abstrahované z aplikácie do tzv. *message file* súboru. *Message file* súbory sú implementované ako DLL alebo EXE súbory (ďalej v texte referované iba ako *message* DLL súbory), ktoré obsahujú binárny *resource*, ktorý obsahuje text pre vaše správy.

S daným zdrojom udalostí môžu byť asociované 3 typy *message file* súborov: *event message file, category message file* a *parameter message file*. Jeden *message* DLL súbor môže reprezentovať ľubovoľnú kombináciu týchto *message file* súborov.

Message DLL súbor je so zdrojom udalostí asociovaný cez hodnoty registra v podkľúči s rovnakým názvom, ako obsahuje `sourceName` parameter odovzdaný funkcii `RegisterEventSource`. Na základe tohto podkľúča je aj rozhodnuté, do ktorého logu sa majú udalosti hlásiť. Typicky sa náležité zmeny v registri robia pri inštalácií, ale môže ich spraviť aj samotná aplikácia. Relevantné kľúče v registri majú potom nasledujúcu podobu:

```
HKEY_LOCAL_MACHINE
   SYSTEM
       CurrentControlSet
           Services
               EventLog
                   Application
                       Event Source
                           ...
                   Security
                       Event Source
                           ...
                   System
                       Event Source
                           ...
                   CustomLog
                       Event Source
                           ...
```

Implicitne má kľúč `EventLog` len 3 podkľúče – `Application, Security` a `System`. Ďalší podkľúč môžete vytvoriť pre vašu aplikáciu. Keď potom voláte funkciu `RegisterEventSource`, systém prehľadáva podkľúče kľúča `EventLog` v abecednom poradí, až kým nenatrafí na kľúč zhodný so `sourceName` parametrom.

Pre asociáciu vašich *message file* súborov musíte vytvoriť pod kľúčom vášho zdroja udalostí náležité hodnoty. Podporované hodnoty sú nasledovné:

Hodnota registra	Typ	Popis
TypesSupported	REG_DWORD	Množina príznakov popísujúca typ udalostí, ktoré podporuje váš message DLL súbor.
EventMessageFile	REG_EXPAND_SZ	Cesta k *event message file* súborom. Event Viewer hľadá túto hodnotu pri konvertovaní ID udalosti na zrozu-miteľný text.
CategoryMessageFile	REG_EXPAND_SZ	Cesta ku *category message file* súborom. Event Viewer hľadá túto hodnotu pri konvertovaní ID kategórie na zrozumietľný text.
ParameterMessageFile	REG_EXPAND_SZ	Cesta k *parameter message file* súborom. Event Viewer hľadá túto hodnotu pri konvertovaní ID zameniteľného parametru na zrozumiteľný text.
CategoryCount	REG_DWORD	Udáva počet kategórií podporovaných zdrojom udalosti.

Okrem týchto hodnôt sú tu ešte dve užitočné hodnoty – `MaxSize` a `Retention`. Hondota `MaxSize` je typu `REG_DWORD` a udáva maximálnu veľkosť, ktorú môže daný log nadobudnúť. Hodnota `Retention` je takisto typu `REG_DWORD` a určuje, maximálne aká stará môže byť udalosť (v sekundách), kým dôjde k jej zmazaniu. Pokiaľ tieto hodnoty neurčíte, ich implicitná hodnota bude pre `MaxSize` 512 KB a pre `Retention` 7 dní. Tieto dve hodnoty sa najčastejšie menia pomocou Event Viewer snap-inu.

Vytvorenie *message* DLL a EXE súborov

Prvým krokom k vytvoreniu *message* DLL súboru je vytvorenie textového súboru pre zdroj vašich správ. Tento tzv. *message source* súbor je všeobecne označovaný ako „MC“ súbor. Jeho názov môže byť ľubovoľný, podstatná je prípona .mc, ktorú musí mať.

MC súbor je štruktúrovaný ako séria ľubovoľne usporiadaných správ. Popis syntaxe *message file* súborov nájdete tu. Názorný príklad MC súboru sa nachádza vo vzorovom príklade k tejto téme.

Pri hlásení udalosti s použitím funkcie *`ReportEvent`*, *`eventID`* parameter určuje ID udalosti. Event Viewer snap-in prehľadá *event message file* súbor a skonvertuje dané ID na zrozumiteľný textový popis. Obdobne to funguje s parametrom *`eventType`*, kedy sa prehľadáva *category message file* súbor.

Môže sa zdať, že pre každé ID udalosti je jednoducho nájdený zodpovedajúci nemenný textový reťazec. Niekedy je však užitočné použiť časť tohto textového reťazca prispôsobenú danej situácií – napr. pri nemožnosti otvoriť nejaký súbor, sa vypíše cesta k danému súboru. V tomto okamihu je využitý parameter *`strings`* funkcie *`ReportEvent`*.

Vytvorená správa môže obsahovať špeciálnu následnosť znakov, ktoré udávajú, kam sa má umiestniť reťazec špecifický pre danú udalosť. Napríklad nasledujúca správa k udalosti obsahuje 2 zameniteľné reťazce:

```
“The file %1 was replaced with the file %2”
```

Ak teraz ako *`strings`* parameter funkcie *`ReportEvent`* odovzdáte pole dvoch reťazcov, Event Viewer snap-in nahradí “%1” za prvý reťazec a “%2” za druhý reťazec. Tieto reťazce by mali byť nezávislé na jazyku – čísla, názvy súborov, názvy iných zdrojov…

Podobne fungujú aj nahraditeľné parametre. Tie sú v správach označené ako “%%” nasledované číslom, udávajúcim ID parametru:

```
“This is an example %%237”
```

Event Viewer snap-in v tomto prípade v *parameter message file* hľadá parameter s ID 237.

Po vytvorení vášho MC súboru je nutné ho skompilovať. Kompilátor pre MC súbory je obsiahnutý priamo vo Visual Studiu. Po skompilovaní MC súboru sú vytvorené nasledujúce súbory:

- **MSG00001.bin** – Tento súbor obsahuje všetky reťazce správ v binárnej forme. Taktiež obsahuje informácie pre spájanie ID správy a textu správy.

- **MyMsgs.rc** – Tento resource script súbor obsahuje odkaz na binárne informácie obsiahnuté v MSG00001.bin súbore.

- **MyMsgs.h** – Jedná sa o hlavičkový súbor obsahujúci *#define* pre každé meno, ktoré sa objavuje v MC súbore. Tento súbor by ste mali priložiť ku každému súboru so zdrojovým kódom, ktorý volá funkciu *ReportEvent*.

Čítanie event logu

Aby ste mohli čítať z event logu, prvé, čo potrebujete je *handle* event logu. Tú získate volaním funkcie *OpenEventLog*. Nakoľko sa jedná o *handle*, po skončení práce s ňou by ste ju mali zatvoriť volaním funkcie *CloseEventLog*. Po získaní *handle* z event logu čítate pomocou funkcie *ReadEventLog*. Je vhodné túto funkciu najskôr volať pre získanie veľkosti buffera, ktorý bude potrebný na získanie dát a až následne volať funkciu pre získanie dát. Po úspešnom volaní funkcie *ReadEventLog* váš buffer obsahuje jednu, alebo viac štruktúr EVENTLOGRECORD.

Štruktúra EVENTLOGRECORD okrem iného obsahuje čas vygenerovania a zapísania udalosti. Tieto časy udávajú počet sekúnd od 1.1.1970 0:00:00.

K ťažkostiam dochádza v okamihu, keď potrebujete zistiť text ku kategórii udalosti a detailný popis udalosti. Na základe toho, ako event log funguje, je jediná cesta, ako získať požadované informácie, je vyhľadať zdroj udalostí v registri, načítať príslušný message DLL súbor a extrahovať z neho detaily kategórie a popisu. Samotné extrahovanie textu zabezpečí funkcia *FormatMessage*.

Ako už bolo v tomto texte uvedené, cesty k message DLL súborom sú uvedené v hodnotách registru pod názvom EventMessageFile, ParameterMessageFile a CategoryMessageFile, v závislosti akú správu chcete nájsť.

Vaša aplikácia je zodpovedná za nájdenie správnej hodnoty registra a jej následné spracovanie – nakoľko sa jedná o hodnotu REG_EXPAND_SZ, je nutné pred jej použitím zavolať funkciu *ExpandEnvironmentStrings*. Tým získate zoznam *message DLL* súborov, ktoré teraz musíte načítať do adresového priestoru procesu a to volaním funkcie *LoadLibraryEx*. Po tom, čo ste získali *instance handle* z funkcie *LoadLibraryEx*, odovzdáte ju spolu s ID správy funkcii *FormatMessage*.

Ak funkcia uspeje, správa bola lokalizovaná, získate hľadaný text a môžete uvoľniť načítanú knižnicu. Ak funkcia neuspeje, musíte uvoľniť načítanú knižnicu a načítať ďalší message DLL súbor.

Ak potrebujete, aby vaša aplikácia bola upozorňovaná na zmeny v event logu, zavolajte funkciu *NotifyChangeEventLog*:

```
BOOL NotifyChangeEventLog(
   HANDLE hEventLog,
   HANDLE hEvent
   );
```

Keď systém zaznamená zmenu, je signalizovaný *hEvent*. Nakoľko sa táto signalizácia deje pomocou funkcie *PulseEvent*, nie je nutné hEvent resetovať.

Použitá literatúra:

RICHTER, Jeffrey. CLARK, Jason D. *Programming Server-Side Applications for Microsoft Windows 2000*. 1. vyd. 2000. ISBN 0-7356-0753-2

Microsoft Developer Network, domovská www stránka, dostupná na URL http://msdn.microsoft.com