Wikiprint Book

Title: Masowe skanowanie przy pomocy FS_Monit

Subject: eDokumenty - elektroniczny system obiegu dokumentów, workflow i CRM - DeployerGuide/Customization/MassScaning

Version: 16

Date: 01/20/26 19:37:19

Table of Contents

Masowe skanowanie przy pomocy FS_Monit *3*
Menu *3*
Wprowadzenie *3*
Konfiguracja usługi masowego skanowania *3*
Przykład: Tworzenie różnych plików w eDokumentach *4*
Podsumowanie *5*

Masowe skanowanie przy pomocy FS_Monit

Menu

1. Wprowadzenie
2. Instalacja programu FS_Monit
3. Przykład: Tworzenie różnych plików w eDokumentach
4. Podsumowanie

Wprowadzenie

Domyślnie system eDokumenty do skanowania wykorzystuje program BSCanServer, który jest instalowany na stanowisko pracownika obsługującego skanowanie. Jednakże istnieją sytuacje, w których narzędzie to może nie spełnić oczekiwań. Są to m.in.:

- w przypadku skanerów sieciowych, których program BScanServer nie jest w stanie wykryć,
- polityka firmy nie zezwala na "wiązanie stanowisk" ze skanowaniem ,
- liczba osób skanujących jest bardzo duża,
- oraz inne...

W takich przypadkach sugeruje się wykorzystanie programu FS_Monit. Jest to usługa, której działanie polega na nasłuchiwaniu zdefiniowanych w konfiguracji folderów w poszukiwaniu zmiany w strukturze plików, a następnie wykonaniu na tych plikach skryptu. Poniżej znajduje się opis konfiguracji tego narzędzia.

Przejdź do Menu

Konfiguracja usługi masowego skanowania

Konfigurację należy rozpocząć od pobrania i zainstalowania programu FS_Monit.exe :-). Instalacja jest bardzo prosta i wymaga przechodzenia pomiędzy ekranami przyciskiem *Następny*. Domyślnie program instaluje się w katalogu: Dla systemów 32-bitowych:

```
C:\Program Files\BetaSoft\FS_Monit
```

Natomiast dla 64-bitowych

```
C:\Program Files (x86)\BetaSoft\FS_Monit
```

1. W pliku wdnotify.ini (w katalogu aplikacji) ustawiamy jaki skrypt ma zostać wykonany po wykryciu zmiany w systemie plików we wskazanym katalogu np.:

```
[options]
# ms
pause_between_files=500

[path.1]
path="C:\SKANY\"
recursive=1
cmd=C:\Program Files\BetaSoft\FS_Monit\php\php.exe -c "C:\Program Files\BetaSoft\FS_Monit\php" -q "C:\Program Files\BetaSo
```

Podana konfiguracja oznacza że folder SKANY na dysku C będzie obserwowany przez FS_Monit i po wykryciu nowego pliku uruchomi skrypt doit.php. (nieco niżej będzie więcej o samym skrypcie). UWAGA!!! Skrypt działa w momencie kiedy plik trafia do katalogu !!!

Następnie uruchamiamy usługę poprzez wykonanie pliku start.bat. UWAGA! Ścieżka C:\SKANY musi już istnieć!!

1. Następnie ustawiamy opcje w pliku konfiguracyjnym C:\Program Files\BetaSoft\FS_Monit\scripts\sc\config.inc - odpowiadają one za podstawowe dane dostarczane do wykonywanego później skryptu który przez protokół SOAP będzie wywoływał API systemu eDokumenty:

```
<?php
// Adres systemu
```

```
define('LOCATION',  'http://support.edokumenty.eu/vdemo/eDokumentyApi.php');
// Login i hasło oraz oznaczenie jednostki do API - umieszczone w config.inc na serwerze
define('EDOK_API_LOGIN', 'edokumenty_api_vdemo');
define('EDOK_API_PASSWORD', '100%apiedokumenty');
define('DEFAULT_ENTITY_SYMBOL', 'ps');

// przyklad: LOGOS_Jan Nowak_Paper_Router_20110222.pdf
define('NAME_REGEX', '/^(?<client>\d+)_(?<name>[^_]+)_(?<dctptp>[^_]+)_(?<type>[^_]+)_(?<adddat>\d{8})\.[a-zA-Z]+/');

// id jednostki organizacyjnej do której trafi utworzony dokument
define('TARGET_ORUNID', 1);

define('TARGET_USRNAM', '');
?>
```

Użytkownik zdefiniowany w danej:

```
define('EDOK_API_LOGIN', 'edokumenty_api_user');
```

oraz hasło

```
define('EDOK_API_PASSWORD', '100%apiedokumenty');
```

Musi być analogicznie zdefiniowane w pliku **config.inc** systemu eDokumenty:

```
// Nazwa użytkownika wymaganego do zalogowania się do API
define('EDOK_API_LOGIN', 'edokumenty_api_user');

// Hasło dla użytkownika EDOK_API_LOGIN
define('EDOK_API_PASSWORD', '100%apiedokumenty');

// Dodatkowa stała która umożliwia pominięcie autentykacji (FALSE - nie sprawdza danych EDOK_API_LOGIN i EDOK_API_PASSWORD
// domyślnie ustawiona na TRUE, można poprzez GET oraz WSS
define('EDOK_API_AUTH_MODE', TRUE);
```

1. Pozostaje jeszcze określić co będzie robił skrypt doit.php.

Załączony skrypt na przykład tworzy dokument w systemie używając danych z powyżej opisanej konfiguracji (TARGET_ORUNID).

Przejdź do Menu

Przykład: Tworzenie różnych plików w eDokumentach

Niniejszy przykład będzie opisywał automatyczne generowanie różnego rodzaju pism na podstawie różnych katalogów, do których trafiają skany. Dokładniej, to na serwerze na dysku D jest katalog Skany, a w nim udostępnione poprzez sieć dwa katalogi:

- Pisma
- Faktury

Dodatkowo w każdym z nich utworzono katalog sent, do którego trafiają zaimportowane dokumenty. Czyli ścieżki wyglądają następująco:

```
D:\Skany\Pisma
D:\Skany\Faktury
D:\Skany\Pisma\sent
D:\Skany\Faktury\sent
```

Do każdego katalogu wpada odpowiedni dokument. Wobec tego konfigurację należy rozpocząć od odpowiedniego zdefiniowania pliku: **wdnotify.ini**. Przykładową taką konfigurację przedstawiono poniżej:

```
[options]

# ms
pause_between_files=500

# Pisma
[path.1]
path="D:\Skany\Pisma\"
recursive=0
cmd=C:\Program Files (x86)\BetaSoft\FS_Monit\php\php.exe -q -c "C:\Program Files (x86)\BetaSoft\FS_Monit\php" "C:\Program

#Faktury
[path.2]
path="D:\Skany\Pisma\"
recursive=0
cmd=C:\Program Files (x86)\BetaSoft\FS_Monit\php\php.exe -q -c "C:\Program Files (x86)\BetaSoft\FS_Monit\php" "C:\Program
```

Należy pamiętać, aby po każdej zmianie w pliku **wdnotify.ini** zrestartować usługę **Wykrywanie zmian w systemie plików** (usługa rejestrowana przez FS_Monit przy instalacji). Jeśli jest problem z zatrzymaniem to należy zabić proces wdnotify.ini.

Następnie należy utworzyć w katalogu:

```
C:\Program Files (x86)\BetaSoft\FS_Monit\scripts\
```

podkatalog np. *paper* i umieścić w nim skrypt **create_doc.php** oraz plik konfiguracyjny **config.inc**

Skrypt **create_doc.php** służy do generowania dokumentu na podstawie parametrów wywołania oraz parametrów z pliku konfiguracyjnego. Plik konfiguracyjny config.inc ma postać:

```
<?php
final class Config {

   public static $EDOK_API_LOGIN = 'edok_api_user';
   public static $EDOK_API_PASSWORD = 'aqq123';
   public static $DEFAULT_ENTITY_SYMBOL = 'def';
   public static $DEFAULT_DCTPID = 2;
   public static $EDOK_API_LOCATION = 'http://localhost/eDokumentyApi.php';
   public static $TARGET_ORUNID = 3;
   public static $SENT_DIR = 'D:/Skany/Faktury';
   public static $NAME_REGEX = '/^(?<name>[^_]+)/';
   public static $args = NULL;
}

if (isset($_SERVER['argv'])) {
   $args = $_SERVER['argv'];
   array_shift($args);

   Config::$args = $args;
}
?>
```

Przejdź do Menu

Podsumowanie

W powyższym przykładzie przedstawiono sposób realizacji ładowania określonego typu plików (w tym przypadku faktury) na podstawie lokalizacji załadowanego pliku.

Można tworzyć pliki w inny sposób np. parametry przekazywać poprzez lokalizację katalogu i wykrywanie jego nazwy. Można również determinować typ dokumenty na podstawie nazwy pliku.

Często spotykaną konfiguracją jest rozpoznawanie nazwy klienta na podstawie nazwy katalogu (musi być zgodna z symbolem kontrahenta) i umieszczenie automatyczne na tej podstawie dokumentu w kartotece klienta (skrypt doit.php).

W pliku config.inc (najlepiej umieścić w osobym katalogu razem z doit.php) należy zmienić NAME_REGEX:

```
public static $NAME_REGEX = '/^(?<client>\w+)\\\(?<name>[^_]+)/';
```