

Udostępnienie aplikacji przez Internet jako program CGI

1. CGI = *Common Gateway Interface*, "to znormalizowany interfejs, umożliwiający komunikację pomiędzy oprogramowaniem serwera WWW a innymi programami znajdującymi się na serwerze." Więcej informacji w wikipedii: <http://pl.wikipedia.org/wiki/CGI>
2. Aby móc skorzystać z CGI, oprócz aplikacji, którą chcemy udostępnić, potrzebujemy:
 - a) serwera www, np. Apache (<http://httpd.apache.org/>), który będzie prezentował wyniki naszej aplikacji (w postaci strony www) użytkownikom interpretera języka skryptowego, np. Perl (www.perl.org);
 - skrypt (czyli program w języku interpretowanym, a nie kompilowanym) będzie na podstawie parametrów przekazanych ze strony www wywoływał naszą aplikację, a następnie przygotowywał jej wyniki do prezentacji na stronie www.

Zaletą Apache'a jak i Perla jest to że są dostępne zarówno dla systemów windowsowych jak i linuxowych, choć jako wywodzące się z tych ostatnich mają raczej linuxowy aromat:-) Oczywiście są darmowe:-)

3. Być może najłatwiejsza i najlepsza opcja (szczególnie, jeśli nie chcesz kończyć zabawy z oprogramowaniem webowym na CGI) jest zainstalować pewnego rodzaju kombajnu: **A+M**(ew. PostgreSQL)+**P+P**, dzięki czemu odpada samodzielne zgrywanie ze sobą poszczególnych modułów. (Wyjaśnienie: MySQL to baza danych typu SQL, a PHP to język służący do prostego budowania całkiem złożonych dynamicznych stron www). Można spróbować programu XAMPP (<http://www.apachefriends.org/en/xampp.html>), którego zaletą jest m.in. dostępność wersji dla Windows (XAMPP) i Linux (LAMP). Instalacja jest bardzo prosta i jedyne o czym trzeba pamiętać to ustawienia bezpieczeństwa. Można do nich dotrzeć z poziomu strony www służącej do zarządzania XAMPPem, do której uzyskamy dostęp wpisując w przeglądarce "<http://localhost>" - jest to "adres" sieciowy (IPv4=<http://127.0.0.1>) serwera działającego na lokalnym komputerze.
4. Mając już Apache'a oraz Perl'a możemy przejść do uruchamiania aplikacji CGI:
 - Jeżeli trzeba będzie nieco zmienić konfigurację Apache'a - plik **httpd.conf** (można go znaleźć w katalogu **C:\xampp\apache\conf\httpd.conf** jeśli instalowany był XAMPP dla Windows) można skorzystać z HOW-TO na stronie Apache'a: <http://httpd.apache.org/docs/2.0/howto/cgi.html>.
 - W pliku httpd.conf należy sprawdzić katalogi: dokumentów, np. **DocumentRoot "C:\xampp\htdocs"** oraz CGI, np. **ScriptAlias /cgi-bin/ "C:\xampp/cgi-bin/"**.
 - w tym pierwszym (i jego podkatalogach) będą umieszczane strony html lub php (np. XAMPP umieścił tu już coś swojego...)w tym drugim umieszcza się skrypty cgi: ze względów bezpieczeństwa prawo zapisu do niego powinien mieć tylko administrator serwera/konta www. Zasadniczo lepiej aby z żadnego innego katalogu nie było możliwości uruchamiania skryptów CGI (nie ustawiać atrybutu ExecCGI dyrektywy Options w katalogu dokumentów htdocs)
 - Jeśli zainstalowany został XAMPP działanie CGI można przetestować wpisując <http://localhost/cgi-bin/printenv.pl> w przeglądarce. Skrypt ten drukuje zmienne środowiskowe dostępne Perlowi. Znowu - ze względu na bezpieczeństwo docelowo lepiej skrypt ten wyrzucić!
5. Przykład: CGI do wywołania obliczeń Octave (Matlab):
 - Instalujemy Octave w wersji dla Windows lub Linux (<http://www.gnu.org/software/octave/>); dalej zakładamy że po instalacji program wykonywalny jest w katalogu: **C:\Octave\3.2.2_gcc-4.3.0\bin\octave.exe**

wklejamy ponizszy kod Perl (na podstawie kodu **Mai Zhou**, <http://www.ms.uky.edu/~statweb/testmatlab.html>) do pliku, np. **gooctave.cgi**, który umieszczamy w katalogu serwera www dla cgi-bin:

```
#!/C:/xampp/perl/bin/perl.exe"
# Sciezka do interpretera Perla

# Octave-cgi main file...  written by witold.dyrka@pwr.wroc.pl
# Based on the code provided by mai@ms.uky.edu .
# The code is under GPL 2.0 copyright.
# For security reason, Octave's file/directory commands are disabled.

my($temp);          # my - zmienna dostepna tylko wewnatrz biezacych nawiasow klamrowych =
lokalna

$temp = "C:/xampp/htdocs/tmp";      # katalog tymczasowy...

use strict;         # uzywamy scislej skladni
use CGI;            # uzywamy CGI

my($results, $in);
my($q)= new CGI;    # tworzymy obiekt CGI - tak: Perl 5 jest jezykiem obiektowym:-)

# Kolejne parametry pola INPUT przekazanego skryptowi
# laczymy w liste rozdzielona \n (newline):
$in=join("\n", $q->param('INPUT'));

# A teraz tworzymy strone HTML na ktorej wyswietlimy wyniki.
# Strona ta bedzie miala pole INPUT typu TEXTAREA oraz klawisz SUBMIT:
print $q->header,
'
<HTML><HEAD><TITLE>Octave-cgi</TITLE></HEAD>
<B>Program Input</B><BR>
<FORM METHOD=POST ACTION="'. $q->url.'">
<TEXTAREA NAME="INPUT" ROWS=5 COLS=64>
'. $in. '</TEXTAREA><BR><INPUT TYPE=SUBMIT VALUE="Submit to Octave">
</FORM>';

if ($in) {          # jezeli otrzymalismy jakikolwiek parametr (=polecenia Octave):
# Filtrujemy nasza liste parametrow, na wypadek gdyby ktos chcial
# zrobic cos zlosliwego naszemu komputerowi...
# Uzywamy do tego wbudowanego w Perl narzedzia korzystajacego z wyrazen regularnych:
$in =~ s/\r//g;
$in =~ s/(dir|shell|system|rmdir|ls|mkdir|rename|unlink) [^\n]*//g;
$in =~ s/(umask|mkfifo|readdir|popen|fork|cd|chdir|getpw) [^\n]*//g;

chdir($temp);      # wchodzimy do katalogu tymczasowego
open(R, ">$temp/octave.in"); # otwieramy plik 'octave.in' - skrypt Octave

print R $in. "\n"; # bezpieczna liste polecen zapisujemy w pliku 'octave.in'
# konczac ja elegancko znakiem nowej linii...

print R "print -dgif \"$temp/octave.gif\";\n"; # zarzadzamy zeby ostatni rysunek w ciagu
# polecen Octave zapisac do pliku
octave.gif

print R "quit\n";  # polecenie wyjścia z Octave na koncu
skryptu

close R;           # zamykamy skrypt Octave

# Odpalamy Octave, jako wejście dajemy plik 'octave.in', jako wyjście 'octave.out':
my $x = system(`C:\\Octave\\3.2.2_gcc-4.3.0\\bin\\octave.exe --traditional -q -H -f <$temp/octave.in
>$temp/octave.out`);
# Ustawiamy prawa dostępu (wlasiciel odczyt/zapis/uruchamianie; inni: odczyt) do utworzonego pliku
graficznego...
my $x = system(`chmod 644 C:/xampp/htdocs/tmp/octave.gif`);

# Otwieramy plik 'octave.out' z rezultatami:
open(IN, "$temp/octave.out");
```

```

# Czyscimy go, aby ladnie wyswietlil sie na stronie www:
$results = join(" ", <IN>);
$results =~ s/&/&amp;/g;
$results =~ s/</&lt;/g;
$results =~ s/>/&gt;/g;
close(IN);          # i zamykamy!

# Drukujemy wynik:
print '
<HR><B>Program Output</B><PRE>'. $results. '</PRE>';

# Oraz obrazek:
print '<HR /><B>Graphical Output (if any) </B><BR/><img src= "/tmp/octave.gif"><BR>';

}

# I zakanczamy strone www...:
print '<HR> </BODY></HTML>';

```

- w przeglądarce wpisujemy: <http://localhost/cgi-bin/go octave.cgi>, w oknie podajemy polecenia Octave'a, proponuje aby między nimi było też narysowania jakiegoś wykresu. Wciskamy klawisz submit, czekamy chwilę i powinniśmy otrzymać wynik... jeśli coś nie działa możesz skontaktować się ze mną: witold.dyrka@pwr.wroc.pl

6. Przypominam, że jeśli mamy serwer, który chcielibyśmy pokazać światu, a nie mamy stałego ip, pewne (tymczasowe) rozwiązanie mogą stanowić usługi dynamicznego dns (no-ip). W skrócie działa to w ten sposób, że
 - tworzymy sobie konto i subdomene w serwisie no-ip (np. w www.no-ip.pl)
 - mały program uruchomiony na naszym komputerze przekazuje informacje o swoim aktualnym, dynamicznie przydzielonym adresie IP do serwisu no-ip
 - każdy kto wpisze w przeglądarce nazwę naszej subdomeny zostanie skierowany na aktualny adres naszego serwera.