

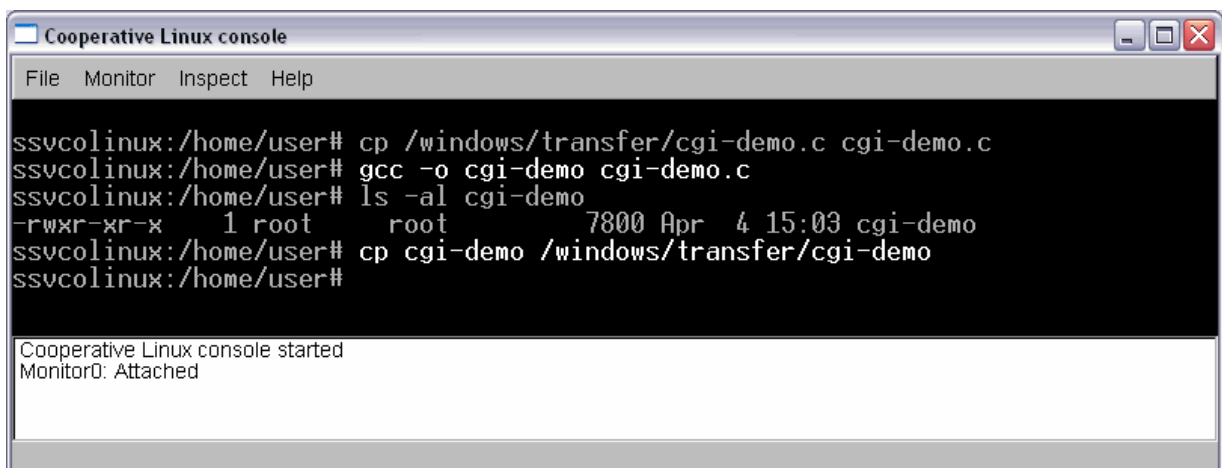
How to write a CGI for the Apache Web server in C

The Com/PC Embedded Gateway Linux (EGL/2) operating system comes with a pre-installed Apache Web server. Please see also *mHT-CPC1L-07.pdf: How to use the Apache Web server* for more information. This server supports also CGI (Common Gateway Interface).

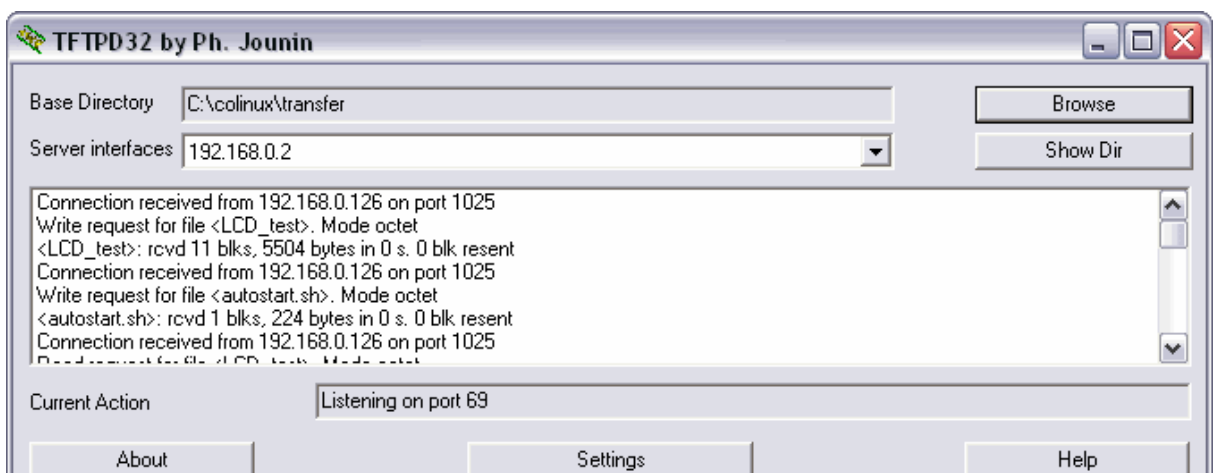
- **1. Step:** Download the file *cgi-demo.c* from the Com/PC website at www.ssv-comm.de. This file is the C source code for a simple CGI sample.
- **2. Step:** Run your SSV coLinux environment on your development PC. Translate *cgi-demo.c* to an executable. Use the following command line

```
gcc -o cgi-demo cgi-demo.c
```

within the SSV coLinux environment. This command line runs the GNU C compiler and creates the executable *cgi-demo*.

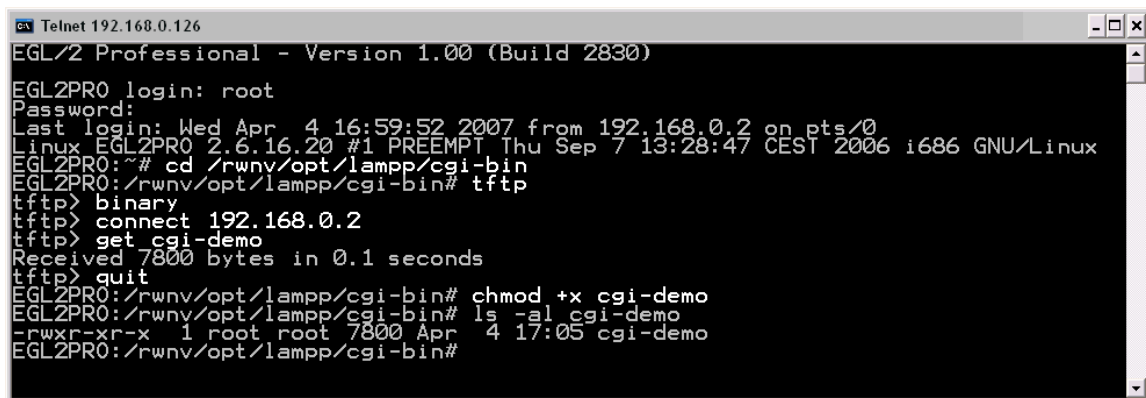


- **3. Step:** Transfer the executable *cgi-demo* from the coLinux directory to the Windows directory */windows/transfer*. Then run a TFTP server within the Windows environment.



- **4. Step:** Run a Telnet or SSH session. Then execute the following commands:

```
cd /rwnv/opt/lampp/cgi-bin
tftp
binary
connect 192.168.0.2
get cgi-demo
quit
chmod +x cgi-demo
```

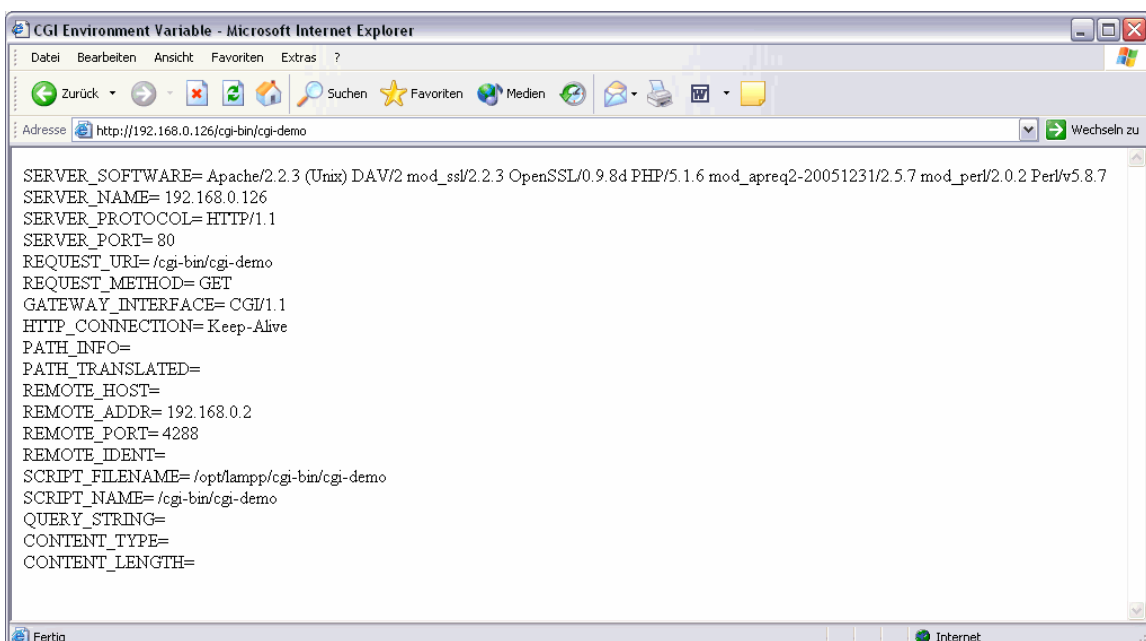


```
Telnet 192.168.0.126
EGL/2 Professional - Version 1.00 (Build 2830)
EGL2PR0 login: root
Password:
Last login: Wed Apr  4 16:59:52 2007 from 192.168.0.2 on pts/0
Linux EGL2PR0 2.6.16.20 #1 PREEMPT Thu Sep  7 13:28:47 CEST 2006 i686 GNU/Linux
EGL2PR0:~# cd /rwnv/opt/lampp/cgi-bin
EGL2PR0:/rwnv/opt/lampp/cgi-bin# tftp
tftp> binary
tftp> connect 192.168.0.2
tftp> get cgi-demo
Received 7800 bytes in 0.1 seconds
tftp> quit
EGL2PR0:/rwnv/opt/lampp/cgi-bin# chmod +x cgi-demo
EGL2PR0:/rwnv/opt/lampp/cgi-bin# ls -al cgi-demo
-rwxr-xr-x  1 root root 7800 Apr  4 17:05 cgi-demo
EGL2PR0:/rwnv/opt/lampp/cgi-bin#
```

Please note: This sample assumes that your development PC with Windows and coLinux is using the IP address *192.168.0.2*. Change this address if necessary.

- **5. Step:** Access the executable *cgi-demo* with your Web browser. The URL for this CGI on the Com/PC is:

<http://192.168.0.126/cgi-bin/cgi-demo>



```
CGI Environment Variable - Microsoft Internet Explorer
Datei Bearbeiten Ansicht Favoriten Extras ?
Zurück Suchen Favoriten Medien
Adresse http://192.168.0.126/cgi-bin/cgi-demo
Wechseln zu

SERVER_SOFTWARE= Apache/2.2.3 (Unix) DAV/2 mod_ssl/2.2.3 OpenSSL/0.9.8d PHP/5.1.6 mod_apreq2-20051231/2.5.7 mod_perl/2.0.2 Perl/v5.8.7
SERVER_NAME= 192.168.0.126
SERVER_PROTOCOL= HTTP/1.1
SERVER_PORT= 80
REQUEST_URI= /cgi-bin/cgi-demo
REQUEST_METHOD= GET
GATEWAY_INTERFACE= CGI/1.1
HTTP_CONNECTION= Keep-Alive
PATH_INFO=
PATH_TRANSLATED=
REMOTE_HOST=
REMOTE_ADDR= 192.168.0.2
REMOTE_PORT= 4288
REMOTE_IDENT=
SCRIPT_FILENAME= /opt/lampp/cgi-bin/cgi-demo
SCRIPT_NAME= /cgi-bin/cgi-demo
QUERY_STRING=
CONTENT_TYPE=
CONTENT_LENGTH=

Fertig Internet
```

This URL assumes that your Com/PC is using the IP address *192.168.0.126*. Change this address if necessary.

The *cgi-demo.c* source code:

```
// cgi-demo.c: Simple CGI demo program.
// Vers. 2.00 - 04.April 2007
// k.d.walter@t-online.de

// Includes

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main (int argc, char *argv[])
{
    char *pEnvPtr;

    printf ("Content-type: text/html\n");
    printf ("\n");
    printf ("<HTML>\n");
    printf ("<HEAD>\n");
    printf ("<TITLE>CGI Environment Variable</TITLE>\n");
    printf ("</HEAD>\n");
    printf ("<BODY>\n");

    // SERVER_SOFTWARE

    pEnvPtr= getenv ("SERVER_SOFTWARE");
    printf ("SERVER_SOFTWARE= ");
    if (!pEnvPtr)
        printf ("<NULL-POINTER>\n");
    else
        printf ("%s\n", pEnvPtr);
    printf ("<br>\n");

    // SERVER_NAME

    pEnvPtr= getenv ("SERVER_NAME");
    printf ("SERVER_NAME= ");
    if (!pEnvPtr)
        printf ("<NULL-POINTER>\n");
    else
        printf ("%s\n", pEnvPtr);
    printf ("<br>\n");

    // SERVER_PROTOCOL

    pEnvPtr= getenv ("SERVER_PROTOCOL");
    printf ("SERVER_PROTOCOL= ");
    if (!pEnvPtr)
        printf ("<NULL-POINTER>\n");
    else
        printf ("%s\n", pEnvPtr);
    printf ("<br>\n");
```

```
// SERVER_PORT

pEnvPtr= getenv ("SERVER_PORT");
printf ("SERVER_PORT= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// REQUEST_URI

pEnvPtr= getenv ("REQUEST_URI");
printf ("REQUEST_URI= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// REQUEST_METHOD

pEnvPtr= getenv ("REQUEST_METHOD");
printf ("REQUEST_METHOD= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// GATEWAY_INTERFACE

pEnvPtr= getenv ("GATEWAY_INTERFACE");
printf ("GATEWAY_INTERFACE= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// HTTP_CONNECTION

pEnvPtr= getenv ("HTTP_CONNECTION");
printf ("HTTP_CONNECTION= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// PATH_INFO

pEnvPtr= getenv ("PATH_INFO");
printf ("PATH_INFO= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// PATH_TRANSLATED
```

```
pEnvPtr= getenv ("PATH_TRANSLATED");
printf ("PATH_TRANSLATED= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// REMOTE_HOST

pEnvPtr= getenv ("REMOTE_HOST");
printf ("REMOTE_HOST= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// REMOTE_ADDR

pEnvPtr= getenv ("REMOTE_ADDR");
printf ("REMOTE_ADDR= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// REMOTE_PORT

pEnvPtr= getenv ("REMOTE_PORT");
printf ("REMOTE_PORT= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// REMOTE_IDENT

pEnvPtr= getenv ("REMOTE_IDENT");
printf ("REMOTE_IDENT= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// SCRIPT_FILENAME

pEnvPtr= getenv ("SCRIPT_FILENAME");
printf ("SCRIPT_FILENAME= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// SCRIPT_NAME
```

```
pEnvPtr= getenv ("SCRIPT_NAME");
printf ("SCRIPT_NAME= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// QUERY_STRING

pEnvPtr= getenv ("QUERY_STRING");
printf ("QUERY_STRING= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// CONTENT_TYPE

pEnvPtr= getenv ("CONTENT_TYPE");
printf ("CONTENT_TYPE= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

// CONTENT_LENGTH

pEnvPtr= getenv ("CONTENT_LENGTH");
printf ("CONTENT_LENGTH= ");
if (!pEnvPtr)
    printf ("<NULL-POINTER>\n");
else
    printf ("%s\n", pEnvPtr);
printf ("<br>\n");

printf ("</BODY>\n");
printf ("</HTML>\n");
return (EXIT_SUCCESS);
}
```

That is all.