

How to access CGI Environment Variables

If you running a CGI (Common Gateway Interface) program over the DNP/1110 *thttpd* web server, there are many environment variables for your CGI program.

- **1. Step:** Download the sample source code **cgidm2-1.c** from **www.dilnetpc.com** (i.e. enter the URL **www.dilnetpc.com/cgidm2-1.c** to your browser). The following listing shows this sample code.

```
// cgidm2-1.c: Simple CGI demo program. Use gcc ...
// Vers. 1.00 - 09.August 2002
// 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-Umgebungsvariable</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 ("  
\n");

// SERVER_PROTOCOL

pEnvPtr= getenv ("SERVER_PROTOCOL");
printf ("SERVER_PROTOCOL= ");
if (!pEnvPtr)
    printf ("  
\n");
else
    printf ("%s\n", pEnvPtr);
printf ("  
\n");

// SERVER_PORT

pEnvPtr= getenv ("SERVER_PORT");
printf ("SERVER_PORT= ");
if (!pEnvPtr)
    printf ("  
\n");
else
    printf ("%s\n", pEnvPtr);
printf ("  
\n");

// REQUEST_URI

pEnvPtr= getenv ("REQUEST_URI");
printf ("REQUEST_URI= ");
if (!pEnvPtr)
    printf ("  
\n");
else
    printf ("%s\n", pEnvPtr);
printf ("  
\n");

// REQUEST_METHOD

pEnvPtr= getenv ("REQUEST_METHOD");
printf ("REQUEST_METHOD= ");
if (!pEnvPtr)
    printf ("  
\n");
else
    printf ("%s\n", pEnvPtr);
printf ("  
\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");

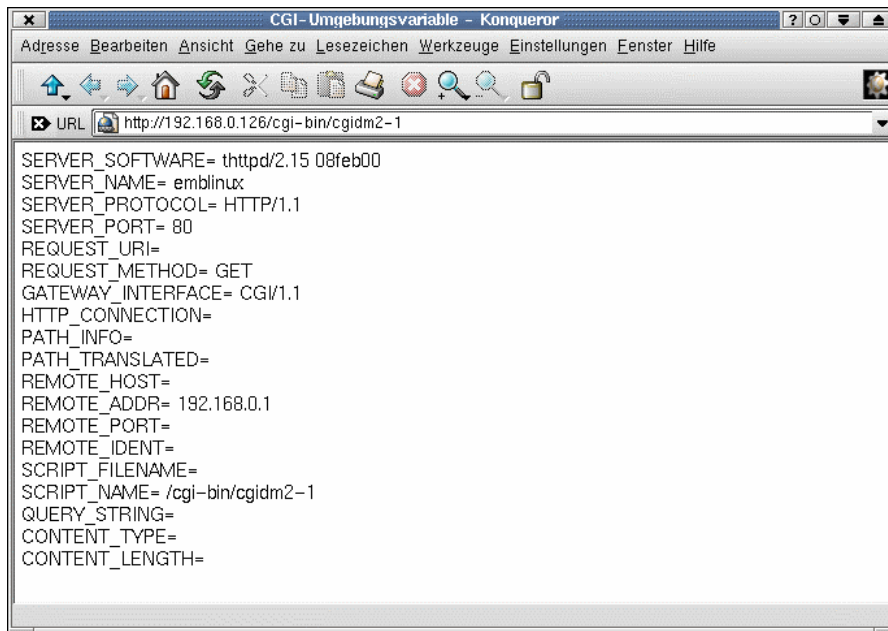
printf ("</BODY>\n");
printf ("</HTML>\n");
return (EXIT_SUCCESS);
}

```

- **2. Step:** Compile the C source code with the gcc (GNU C Compiler) direct with a command line or with the help of a makefile.
- **3. Step:** Transfer the executable form your development system to the DIL/NetPC (directory */home/gast*). Use a FTP session for this file transfer.
- **4. Step:** Copy/move the file with the executable from */home/gast* to */usr/local/www/cgi-bin*. You need superuser rights for the write access to */usr/local/www/cgi-bin*.
- **5. Step:** Run a Web browser on your development system. Point the browser to the CGI program on the DIL/NetPC. Use the following (sample) URL:

`http://192.168.0.126/cgi-bin/cgidm2-1`

Please note: `cgidm2-1` is the name of the executable CGI program. If your program is using a other name, please modify the URL to your needs.



- **6. Step:** For transferring parameters to a CGI program please use the following (sample) URL and watch **QUERY_STRING**:

`http://192.168.0.126/cgi-bin/cgidm2-1/?parameter`

