

# CEHTTP User's manual

---

更新日 2004年1月3日

<http://www001.upp.so-net.ne.jp/ishi/index.html>



# 1. はじめに

## 1. 何ができるのか

CEHTTP は WindowsCE(ver2.11 以上)で動作する WEB サーバーです。ネットワークを通して広く HTML などの文章を公開できます。

Perl などの一般的な CGI は、まだ動作しません。その代わりに C++で拡張できる手段を用意しています。

基本認証をサポートしています。クライアントに対してユーザー名とパスワードを要求することが可能です。

## 2. Apache や IIS と何がちがうのか

小さく、早いを重視しています。

CEHTTP は WindowsCE で動作するように作成された専用 Web サーバーです。全て C++で記述されています。マルチスレッドで動作し、常にストレステストを行いながら実装をしています。若干、実装の美しさよりも効率を重視しています。

その代わりに、あまり大きなデータを取り扱うのは苦手かもしれません。

Apache や IIS はサーバースペックの PC で効率よく動作するように設計されています。WindowsCE には若干重たい気がしています。

またウイルスに感染しにくい特徴があります。世に出回っている多くのサーバーウイルスは、攻撃対象を Intel の Pentium と想定しています。NIMDA のようなウイルスにはまず感染しないでしょうし、またウイルス製作者も CEHTTP は相手にしていないでしょう。

## 3. 何につかうのか

何につかっても問題ありませんが、

私は ADSL 等の常時接続環境で、古い WindowsCE 機が復活し、お役に立てればよいと考えています。また、ちょっとイベント会場などで HTML を配信したいときに利用するのも良いと思います。

WindowsCE 機ですから気軽に持ち運べて、しかもすぐにはリセットができます。



## 2. インストールの方法

### 1. ダウンロード

<http://www001.upp.so-net.ne.jp/ishi/>

から最新版の CEHTTP をパソコン上にダウンロードしてください。

### 2. 解凍する

ダウンロードしたファイルを解凍ツール(たとえば lhaca 等)で解凍します。

### 3. WindowsCE 機にコピーする

自分の持っている WindowsCE 機の CPU にあわせて、ファイルを WindowsCE 機にコピーします。

ARM 用 cehttp2.ARM.CAB  
SH3 用 cehttp2.SH3.CAB  
SH4 用 cehttp2.SH4.CAB  
MIPS 用 cehttp2.MIPS.CAB

自分の WindowsCE 機がどの CPU かは、コントロールパネルで「システムのプロパティ」を開き、「デバイス」タグを見れば判別できます。インターネット上で検索しても OK です。

WindowsCE 機にコピーできたら、CE 機上で CAB ファイルを実行してください。自動的にインストーラーが起動し、CEHTTP がインストールされます。

### 4. インストールされるファイル

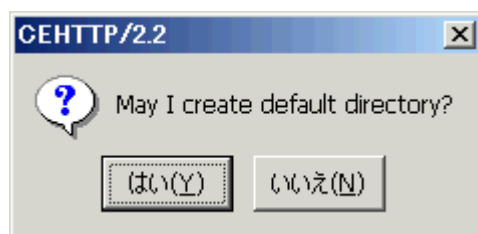
インストールされるファイルとディレクトリは下記のようになります。

「¥program files¥cehttp¥cehttp2.exe」

### 5. 実行する

インストールが完了したら、まずは実行してみましょう。

ここで最低限行っておかなくてはならないことは「WindowsCE 機をネットワークに接続しておく」事です。Web サーバーですからネットワークに接続されていることが必須条件です。



もし、初めて CEHTTP を実行したのならば、上記のようなダイアログが現れます。これは CEHTTP が「ドキュメントルートがないので、作成しても良いか?」と聞いています。

ここでは「はい」をクリックします。

そうすると自動的に「¥wwwroot」フォルダと、その中に「index.html」ファイルが自動的に生成されます。



上記のような画面になります。(HPC/Pro/2000 の場合)

IP Address は現在 CEHTTP が認識している IP アドレスです。

もしネットワークに接続されていない場合、ここには「127.0.0.1」と出ているかもしれません。このような場合は他の PC から CEHTTP にアクセスできません。まず WindowsCE 機をネットワークに接続して、CEHTTP を再度起動してください。

アイコンをタップすると、CEHTTP の画面が非表示になります。再度アイコンをタップすると、CEHTTP が表示されます。

Status は、現在の CEHTTP の状況が表示されています。

「Listen...」と表示されていると、CEHTTP はうまく動作しています。

「Listen failed.」と表示されている場合は、なんらかの理由で起動が失敗している状態です。CEHTTP を二つ立ち上げたりすると、CEHTTP の中でポート番号を取り合って、失敗しています。

ここで他の PC からアクセスしてみましょう。上記の例だと IP アドレスが「192.168.2.31」となっていますから、

<http://192.168.2.31/>

とブラウザのアドレスに打ち込めば良いことになります。

「Can you see me?」とブラウザに表示されたら OK です。インストールは完了です。

## 6. その他のファイル

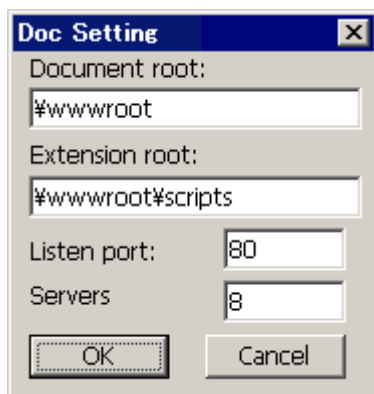
必要であれば、ログファイルを取る拡張 DLL などと同様な方法で追加インストールします。



## 3. 設定

### 1. 設定ダイアログ

メインダイアログの「Setting...」をクリックすることによって、設定ダイアログが現れます。



### 2. ドキュメントルートの設定

ドキュメントは、Web アクセスの際にルートとなる場所のことです。デフォルトでは「%wwwroot」となっています。

もし、「%wwwroot%index.html」というファイルがあれば、「http://<IP アドレス>/index.html」という URL でアクセスできることになります。

「%wwwroot%cehttp%index.html」というファイルは「http://<IP アドレス>/cehttp/index.html」という URL でアクセスできます。

### 3. 拡張ルートの設定

サーバー拡張を動作させるための特別なフォルダを指定します。デフォルトでは「%wwwroot%scripts」となっています。

拡張ルートは、必ず「ドキュメントルート」の下に設定されていなければなりません。

このディレクトリは拡張子が「.CHX」と「.DLL」のファイルにアクセスしたときだけ、CEHTTP がリクエストを処理する特別なフォルダです。普通の HTML ファイル等はここから読み出せません。

- 1.CHX と DLL ファイルがダウンロードされることを防ぎます。
- 2.CHX と DLL ファイルが実行することを許可します。

### 4. ポート番号の設定

Web サーバーが聞き取り(listen)を行うポート番号を指定します。通常 Web サーバーは 80 番で聞き取りを行います。

たとえばポート番号 8081 番で聞き取りを行った場合、ブラウザは下記のようにしてアクセスできるようになります。

<http://<IP アドレス>:8081/index.html>

### 5. サーバスレッド数の設定

Web サーバー数(同時アクセス)を設定します。初期値は 8 スレッドです。

数が大きくなると、同時アクセス数が増えて、高負荷時の応答性がよくなりますが、メモリをその分消費します。



## 4. 基本認証

### 1. 基本認証とは

Web サーバーにアクセスをする際に、クライアントにユーザー名とパスワードを要求することができます。これを認証といいます。

CEHTTP では「基本認証(Basic authorization)」をサポートしています。

基本認証が掛かったページをブラウザから参照しようとする



上記のようなダイアログが出てきます。

ユーザーはブラウザからユーザー名とパスワードを入力し、Web サーバー側で認証されると、はじめてページを参照することができるようになります。

BBS の管理用ページなどは基本認証でアクセス制限する必要があるでしょう。また、ログファイルに認証したユーザーが残ります。誰がアクセスしたのか厳密に管理したい場合などに使ってください。

### 2. 基本認証ファイルの書式

認証ファイルはデフォルトでは「access.cha」というファイル名になります。

下記の書式でテキスト形式で記述します。

```
<username>:<password>
```

たとえば、ユーザー名「taro」、パスワードが「yamada」を設定したい場合、

```
taro:yamada
```

という風に記述します。ユーザー名とパスワードは連続して何人でも記述できます。

```
taro:yamada<改行>
```

```
ichiro:suzuki<改行>
```

...

SJIS 形式で保存してください。WindowsCE 上の Word などではデフォルトでは unicode で保存しますので、PC 上で作成し、後ほど WindowsCE 機に転送すると良いでしょう。

### 3. 基本認証ファイルの置き場所

基本認証ファイルを配置する場所は、認証を掛けたいフォルダです。

基本認証を配置した場合、そのフォルダ配下全てのフォルダに認証の効力は及びます。複数の認証ファイルがある場合、もっとも近い場所にある認証ファイルを利用します。

例;

```
[¥wwwroot]
+ access.cha -A
|
+ [¥Japan]
| |
| + access.cha -B
| + [¥Tokyo]
```

上記のような配置であったとします。認証ファイルは  
¥wwwroot¥access.cha(A)  
¥wwwroot¥japan¥access.cha(B)  
があります。

このとき、URL

「<http://<サーバー>/>」は access.cha(A)に影響されます。  
「<http://<サーバー>/Japan/>」は access.cha(B)に影響されます。  
「<http://<サーバー>/Japan/Tokyo/>」は access.cha(B)に影響されます。

access.cha(A)だけ配置すると、全てのアクセスが access.cha による認証となります。



## 5. レジストリで設定できる値

CEHTTP はいくつかの設定値をレジストリで操作することができます。レジストリツールについては、フリーソフトが沢山ありますので、それらを利用してください。これらの値は、それほど変更する理由はありません。レジストリ保管場所は [HKEY\_LOCAL\_MACHINE¥SOFTWARE¥ISHISOFT¥CEHTTP] です。

キー名	型	デフォルト値	解説
Port	DWORD	80	Listen するポート番号
Timeout	DWORD	3000	タイムアウト値 現在は利用されていません。タイムアウト値参考： リクエスト受付時間は合計90秒です。 データ送信時間は無通信時間が5秒までです。
KeepAlive	DWORD	1	キープアライブ キープアライブとは、一度接続した場合、連続してリクエスト処理を行う処理のことを言います。HTTP/1.0 以前の実装では、一回のリクエストを行うたびに、一回のソケット接続を行っていました。しかしながら昨今の HTML は GIF などの画像が多く利用されており、一枚のページを表示するのに、3～10回ほどサーバーにリクエストが発生します。ソケット接続はコストのかかる処理であり、この負荷を軽減するために HTTP/1.1 では、キープアライブの実装が求められています。
StartServers	DWORD	8	サーバー数 CEHTTP はマルチスレッドで動作します。 このスレッドの数を指定できます。 サーバー数を大きくすると、高負荷時に応答が良くなりますが、メモリを多く消費します。 サーバー数を小さくすると、メモリ使用量が減ります。
DocRoot	STRING	¥wwwroot	ドキュメントルートディレクトリ 設定ダイアログで設定できます。
CgiRoot	STRING	¥wwwroot¥scripts	.dll.chx が動作可能なディレクトリ 必ずドキュメントルート配下になるようにしてください。
DefaultDoc	STRING	index.html	デフォルトドキュメント ファイル名を指定されなかった場合に利用されるファイル名です。
BasicAuthFile	STRING	access.cha	基本認証ファイル名です。 「access.cha」というファイルは、通常 CEHTTP は特別なファイル名として扱い、アクセスできません。 どうしても access.cha というファイルが利用したい場合はここで変更してください。
LogFile	STRING	¥access_log.txt	ログファイル名 アクセスログを取得している場合、ログの保管場所となります。  たとえばシグマリオン II が内蔵している「カスタム メリ」にログファイルを保存したい場合は 「¥カスタム メリ¥access_log.txt」とします。
LogSize	DWORD	102400	ログファイルサイズ ログファイルが無限に増えていくのは困りますので、このサイズで制限を掛けます。  ファイルサイズが 102400(=100KB)を超えると、バックアップを取ってログファイルを新たに作成します。  つまりログファイルは、バックアップをあわせて最大(LogSize)*2 となります。
DNSLookup	DWORD	0	DNS ルックアップ アクセスを受けたときに DNSlookup を行う場合は 0 以外を指定します。 これによって、dll.chx 呼び出しの際には「REMOTE_HOST」がセットされます。 またログファイルにドメインネームが書き出されるようになります。 ただし応答性は悪くなります。





## 6 . MIME タイプ設定

### 1. MIME タイプとは

MIME タイプとは、ファイルの拡張子とファイルの中身を結びつける拡張子です。たとえば、index.html の拡張子「.html」は、HTML であるという意味を持ちます。

たとえば、あるファイルが下記のように記述されているとします。

```
<html><body><b>こんにちは</b></body></html>
```

このファイル名が「test.html」だと、ブラウザは

こんにちは

と表示しますが、「test.txt」だと

```
<html><body><b>こんにちは</b></body></html>
```

と表示します。

前者は「Content-type: text/html」、後者は「Content-type: text/plain」とサーバーが指定してブラウザに送信します。これを受けてブラウザ側で、テキストとして表示を行います。

### 2. なぜ必要なのか

拡張子が「.html」なのだから「HTML 文章でいいのでは？」と誤ってしまいますが、以外とコンテンツタイプは重要です。

ブラウザが「http://hogehoge/cgi.pl」というファイルをリクエストしたとします。このとき拡張子は「.pl」が提供するファイルは一体「Content-type: text/html」なのでしょうか。それとも「Content-type: image/gif」なのでしょうか。

当然 Perl ですから、どちらのデータを返す可能性があります。

Perl で CGI を記述したことがある人は、先頭行に「Content-type: text/html」を記述したことがあると思います。ブラウザは拡張子だけでは、送信されてきたデータが何者であるのかを決定できません。そのため Web サーバーはブラウザに「Content-type:」を送信するのです。

MIME タイプに登録しておけば、拡張子にあわせて適切な「Content-type:」を送信してくれます。Perl などの CGI はそれを自分で記述するのです。

### 3. デフォルトの MIME タイプ

CEHTTP では、あらかじめいくつかの MIME タイプが設定されています。

```
.html text/html
.htm text/html
.gif image/gif
.png image/png
.jpg image/jpeg
.jpeg image/jpeg
.txt text/plain
.css text/css
.bmp image/bitmap
.tif image/tiff
.tiff image/tiff
.mpg video/mpeg
```

.mpeg video/mpeg  
.pdf application/pdf  
.ls application/x-javascript  
.js application/x-javascript  
.mp3 audio/mp3  
.m3u audio/x-mpegurl  
.wav audio/x-wav  
.mid audio/mid  
.midi audio/mid  
.swf application/x-shockwave-flash

## 4. MIME タイプの追加

上記 MIME タイプの他に追加したい MIME タイプがある場合、MIME タイプを追加します。

MIME タイプ追加ファイルは「AddType.txt」というファイル名になります。

下記の書式で、テキスト形式で記述します。

**<拡張子>:<対応するコンテンツタイプ>**

たとえば、拡張子「.html」、コンテンツタイプ「text/x-html; charset=Shift\_JIS」を設定したい場合、

```
.html text/x-html; charset=Shift_JIS
```

と記述します。この記述をしないと、Web サーバーは MIME タイプを決定できず、単なるバイナリとして送信します。CEHTTP では認識できない MIME タイプを全て「Content-type: application/octet-stream」としてブラウザに送信します。

MIME タイプはいくつでも記述することができます。au の EZWeb に対応するコンテンツを記述する場合は、

```
.html text/x-html; charset=Shift_JIS<改行>  
.wml text/vnd.wap.wml; charset=Shift_JIS<改行>
```

と記述します。

## 5. MIME タイプファイルの置き場所

MIME タイプファイルを配置する場所は、CEHTTP がインストールされた場所です。CEHTTP は起動時に「AddType.txt」というファイルを探して読み込みを行います。



## 7. ログファイル

### 1. ログファイルとは

CEHTTP は拡張 DLL によってログを取ることができるようになります。これによって、アクセスがあったのかを記録に残すことができます。

cehttplog.dll を CEHTTP がインストールされたディレクトリと同じ場所に配置することによって自動的にログファイルが取得されます。

各種設定は「レジストリで設定できる値」を参照してください。

### 2. ログファイルの書式

Apache 標準形式を参考にしています。

```
192.168.2.31 - user [11/Feb/2003:14:58:25 ] "GET / HTTP/1.1" 304 - "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
```

上記のようなログが取れます。詳細は下記のようになります。

192.168.2.31	IP アドレス
-	ident - 未使用
user	基本認証で認証したユーザー名
[11/Feb/2003:14:58:25 +900]	アクセス日時
"GET / HTTP/1.1"	リクエストの内容
304	リクエストの結果
-	送信したバイト数
"Mozilla/4.0"	クライアントが利用したブラウザ名

### 3. ログファイルの置き場所

デフォルトでは「%access\_log.txt」になります。この値はレジストリ操作によって変更できます。

### 4. 自動バックアップ

ログファイルはある一定の大きさになると、自動的に一部バックアップを作成し、新たにファイルを作成します。バックアップファイル名は末尾がアンダースコアに置き換えられたものです。

「access\_log.txt」->「access\_log.tx\_」

### 5. 注意点

ログファイルを取るのにはコストが掛かります。高負荷時にはパフォーマンスの足かせとなります。もし、ログファイルが必要ないのならば、「cehttplog.dll」の名前を「\_cehttplog.dll」のように変更しておくか、削除するとログは取得されません。



## 8 . 拡張ファイル

### 1. 拡張ファイルとは

CEHTTP は DLL により拡張が可能となっています。  
拡張ファイルには CHX 形式、DLL 形式、MOD 形式のものがあります。

拡張ファイルについては、各拡張ファイル同梱のヘルプ等を参照ください。

また作成方法は「CEHTTP テクニカル マニュアル」を参照してください。

### 2. CHX/DLL 拡張ファイル

拡張子「.chx」/「.dll」のファイルは、CGI ディレクトリ配下で動作します。

ブラウザより呼び出されたときに、CEHTTP がダイナミックにロードし、実行します。

### 3. MOD 拡張ファイル

拡張子「.mod」のファイルは、CEHTTP がインストールされた場所に配置してください。

CEHTTP は起動時に mod ファイルを検索します。たとえば「ns.mod」という MOD 拡張ファイルがあれば、MIME タイプリストに動的に「.ns」を追加します。「.ns」拡張子を持つファイルがリクエストされた場合、登録された MOD 拡張ファイルが呼び出され処理されます。

### 4. 注意点

不完全な拡張ファイルは CEHTTP を不安定にします。



## 9 . エラーページファイル

---

### 1. エラーページファイルとは

HTTP レスポンスにあわせた独自のエラーページを送信する事ができます。

たとえば、存在しないページをリクエストされた時に、あらかじめ作成しておいた HTML ファイルを送信したり、拡張ファイルを呼び出したりする事が可能です。

「404 Page not found.」を表示する際に、トップページへのリンクを表示するなどを利用すると良いでしょう。

### 2. エラーページ設定ファイルの書式

「ErrorDocument.txt」というファイルを作成し、CEHTTP がインストールされた場所に配置してください。

下記の書式で、テキスト形式で記述します。

<エラー番号><空白><エラーページ>

たとえば 404 エラーが発生した時に /error/404.html を表示したい場合は次のようにします。

404 /error/404.html