

5. Tips

ご注意

- この資料に含まれる情報は可能な限り正確を期しておりますが、日本アイ・ビー・エム株式会社ならびに日本アイ・ビー・エム システムズ・エンジニアリング株式会社の正式なレビューを受けておらず、当資料に記載された内容に関して日本アイ・ビー・エム株式会社ならびに日本アイ・ビー・エム システムズ・エンジニアリング株式会社は何ら保証するものではありません。
- 従って、この情報の利用またはこれらの技法の実施はひとえに使用者の責任において為されるものであり、資料の内容によって受けたいかなる被害に関しても一切の補償をするものではありません。
- 当資料をコピー等で複製することは、日本アイ・ビー・エム株式会社、日本アイ・ビー・エム システムズ・エンジニアリング株式会社および執筆者の承諾なしではできません。また、当資料に記載された製品名または会社名はそれぞれの各社の商標または登録商標です。

Agenda

- 1. IHSプロセス複数起動
- 2. セキュリティー / ハードニング
- 3. Sorry Server / エラー画面
- 参考文献

1. IHSプロセス複数起動

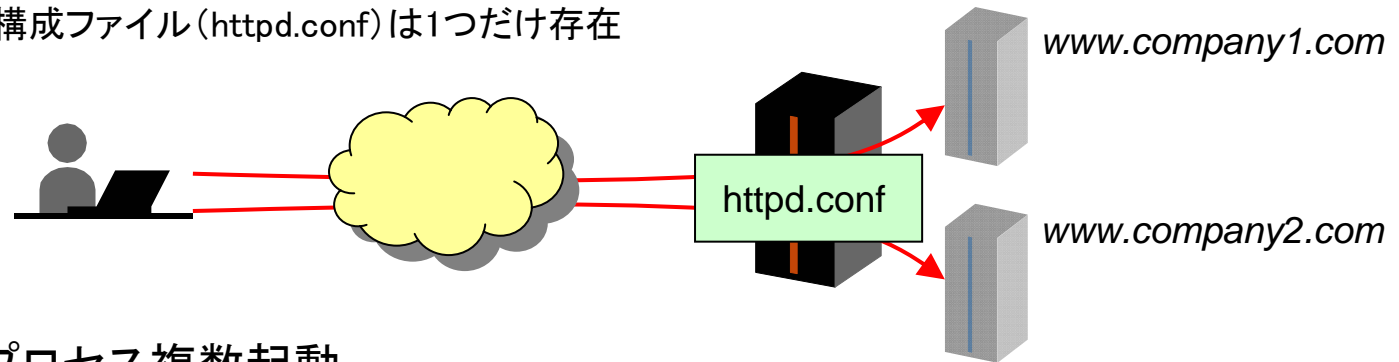
単一筐体における複数のWebサーバー構成

- 一台の筐体内で複数の論理的なWebサーバー構成する要件/ケース
 - ◆ 業務の種類別に設定や管理(起動/停止)を行いたい
 - ◆ クライアント別にURL(ドメイン)を変えてリクエストを受け付けたい
...など

- 複数の論理的なWebサーバーを構成する方法

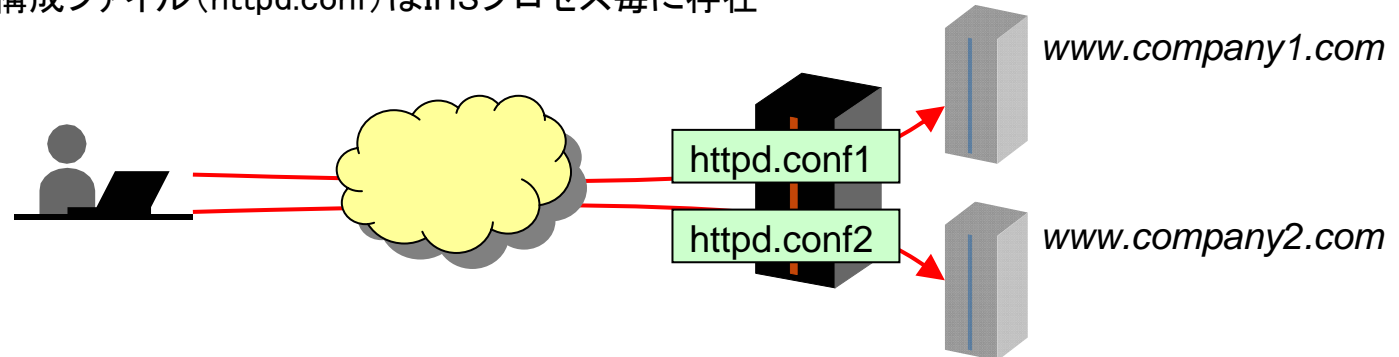
- ◆ VirtualHost

- ▶ 構成ファイル(httpd.conf)は1つだけ存在



- ◆ IHSプロセス複数起動

- ▶ 構成ファイル(httpd.conf)はIHSプロセス毎に存在



VirtualHostとIHSプロセス複数起動の比較

■ VirtualHostとIHSプロセス複数起動の比較

	VirtualHost	IHSプロセス複数起動
構成ファイル/構成の管理	○:管理対象は1つのみ	×:管理対象がIHS毎に複数存在
設定変更の柔軟性	△:設定できないパラメータあり (※次項目参照)	○:IHS毎に設定および再起動が可能
起動/停止の柔軟性	×:IHS全体で同時実施	○:IHS毎に実施可能
システムリソースの活用方法	IHSプロセスに割り当てられたリソースを全てのVirtualHostが共用する	各IHSプロセス単位でリソースが割り当てられてそれぞれ使用する

■ VirtualHostの単位で設定できないディレクティブ (詳細はApacheのマニュアル参照)

- ◆ プロセス数やスレッド数などリクエスト処理数を制御するディレクティブ
 - MaxClients、StartServers、ThreadLimit、ThreadsPerChild、ServerLimit、MaxRequestsPerChild、MaxSpareThreads、MinSpareThreads、ListenBackLog、など
- ◆ IHSの1親プロセス毎に1つの値が割り当てられるディレクティブ
 - ServerRoot、Listen、PidFile、など
- ◆ その他
 - Timeout、SendBufferSize、User、Group、ServerTokens、LimitRequestFields、LimitRequestFieldSize、LimitRequestLine、など

■ 選択基準

- ◆ VirtualHostで設定できるパラメータだけを分けたい場合 ⇒ VirtualHost
- ◆ 設定や運用を完全に分離させたい場合 ⇒ IHSプロセス複数起動

IHSプロセス複数起動の方法

■ 設定方法(AIX)

◆ 起動したい数分の構成ファイル(httpd.conf)を用意

- 例1 : 別IPアドレスのIHSプロセス用の構成ファイル「httpd1.conf」

```

.....
Listen <IPアドレス>:80
BindAddress <IPアドレス>
PidFile /usr/HTTPServer/logs/httpd1.pid
.....

```

- 例2 : 別ポート番号(81)のIHSプロセス用の構成ファイル「httpd2.conf」

```

.....
Listen <IPアドレス>:81
BindAddress <IPアドレス>
PidFile /usr/HTTPServer/logs/httpd2.pid
.....

```

■ 起動/停止方法

◆ 構成ファイルを指定してIHSプロセスを起動/停止

```

[/usr/IHS7/bin]# ./apachectl -k start -f ./conf/httpd1.conf
[/usr/IHS7/bin]# ./apachectl -k stop -f ./conf/httpd1.conf
[/usr/IHS7/bin]# ./apachectl -k start -f ./conf/httpd2.conf
[/usr/IHS7/bin]# ./apachectl -k stop -f ./conf/httpd2.conf

```

IHS複数起動した場合のプロセス状況

■ IHSプロセス単一起動(デフォルト)

- ◆ 1つの親プロセスがrootで起動して親プロセスから1つ/複数の子プロセスが生成
- ◆ 子プロセスはhttpd.conf内で指定したユーザー(デフォルト:nobody)で起動

```

[/usr/IHS7/bin]ps -ef | grep httpd
親プロセス ← root 17594 1 2 15:50:23 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -k start
nobody 17716 17594 0 15:50:24 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -k start
nobody 18332 17594 0 15:50:24 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -k start
nobody 19434 17594 0 15:50:24 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -k start
    
```

■ IHSプロセス複数起動

- ◆ 複数の親プロセスがrootで起動して各親プロセスから1つ/複数の子プロセスが生成
- ◆ 子プロセスはそれぞれの親プロセスのhttpd.conf内で指定したユーザーで起動

```

[/usr/IHS7/bin]ps -ef | grep httpd
httpd1.confの親プロセス ← root 13956 37 16:17:46 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -f conf/httpd1.conf
nobody 16984 13956 0 16:17:47 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -f conf/httpd1.conf
nobody 17172 13956 0 16:17:47 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -f conf/httpd1.conf
nobody 17436 19316 0 16:17:07 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -f conf/httpd2.conf
nobody 17688 19316 0 16:17:07 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -f conf/httpd2.conf
nobody 18334 19316 0 16:17:07 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -f conf/httpd2.conf
nobody 18436 13956 0 16:17:47 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -f conf/httpd1.conf
httpd2.confの親プロセス ← root 19316 0 16:17:06 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -f conf/httpd2.conf
    
```

2. セキュリティー / ハードニング

非rootユーザー稼動

■ 目的

- ◆ システムを運用する上で、できるだけrootの使用を避けたい、といった運用管理要件を実現するためのセキュリティ対策

■ 設定方法

- ◆ 1. IHSのインストール・ディレクトリーのオーナーを非rootユーザーに設定
- ◆ 2. Listenポートを1024番以降にするために構成ファイル httpd.conf で以下のように設定

Listen 8080

1024番以降を指定

■ 起動/停止方法

- ◆ IHSのインストール・ディレクトリーのオーナーに設定した非rootユーザーで起動/停止を実行

```
# su - ihsadmin
$ /usr/IBM/HTTPServer/bin/apachectl -kstart
$ /usr/IBM/HTTPServer/bin/apachectl -k stop
```

■ 考慮事項

- ◆ IHSを非rootユーザーで稼動させるには1024番以降のポート番号にする必要あり
 - AIX V6.1の新機能である拡張RBAC(Role Based Access Control)機能を使用すると80番ポートにて非root稼動させることが可能 (詳細はスピーカーノートのリンク先を参照)
- ◆ IHSをWASとは異なる非rootユーザーで稼動させる場合、IHSをWASの管理コンソールから管理する場合は、ノードエージェント経由ではなくIHS管理サーバ経由にする必要あり

<上記の設定になっている場合>

```
[/usr/IHS7/bin]ps -ef | grep httpd
ihsadmin 540884 1 0 17:21:20 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -k start
ihsadmin 676010 540884 0 17:21:22 - 0:00 /usr/IHS7/bin/httpd -d /usr/IHS7 -k start
```

非root ←

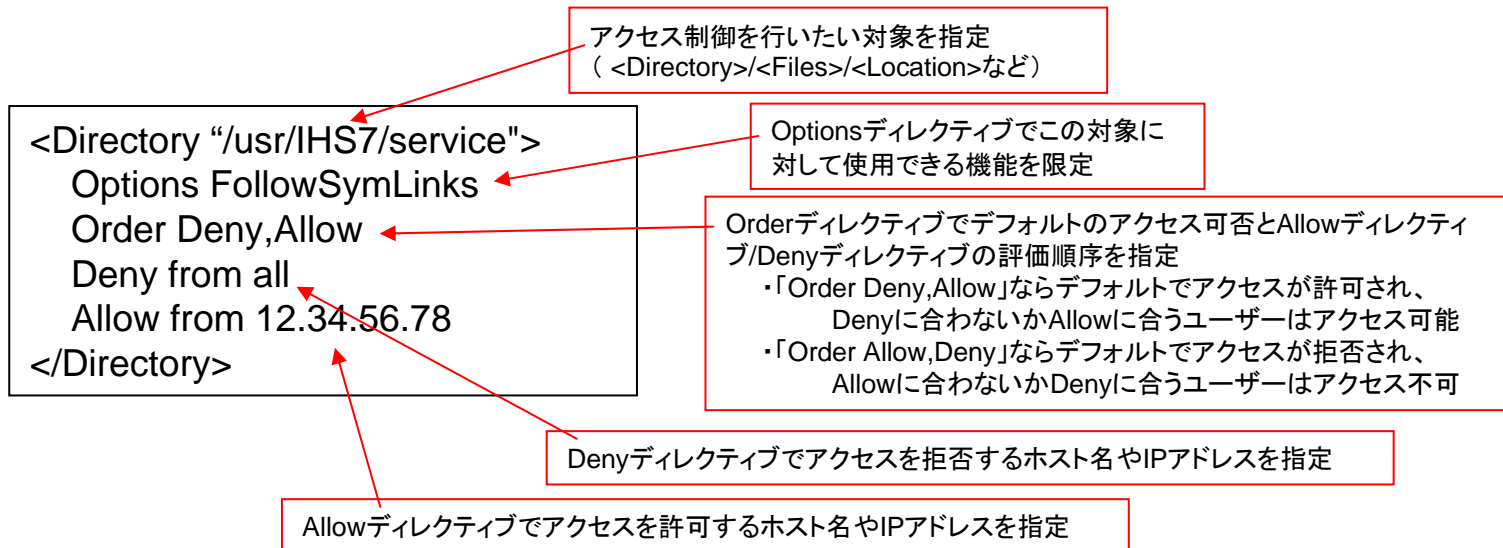
アクセス制御

■ 目的

- ◆ 不特定多数のユーザーが多数のコンテンツやアプリケーションを参照/使用するシステムにおいては、ユーザーに対して想定外あるいは参照させるべきでないコンテンツを参照されてしまう可能性があるため、そのためのセキュリティ対策

■ 設定方法

- ◆ 構成ファイル httpd.conf で以下のように設定



■ 考慮事項

- ◆ 更にHTTPメソッド (GET/PUT/POSTなど) 毎にアクセス制御を行いたい場合は LimitExceptディレクティブを使用

バージョン情報の隠蔽

■ 目的

- ◆ HTTPレスポンスヘッダーやエラー画面のフッターにIHSのバージョン情報が表示されると、悪意のあるユーザーからセキュリティホールを狙われてしまう可能性があるため、そのためのセキュリティ対策

■ 設定方法

- ◆ 構成ファイル httpd.conf で以下のように設定

```
ServerTokens Prod(uctOnly)
ServerSignature Off
```

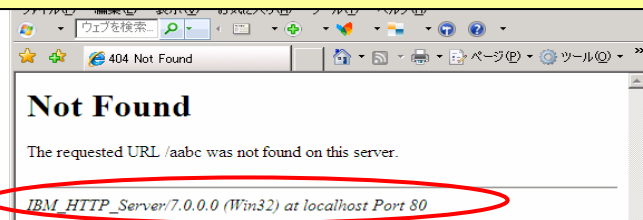
HTTPレスポンスヘッダーに関する設定で「Prod」でも「ProductOnly」でもOK

サーバーが生成するドキュメント（エラー画面など）のフッターに関する設定

■ バージョン差異による考慮事項

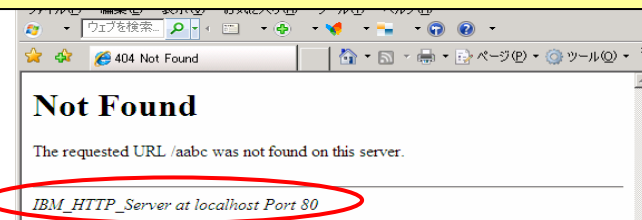
- ◆ IHSV6.0以降（Apache2.0.44以降）では、ServerTokensが「Prod」に設定されていればServerSignatureがOnであってもバージョン情報は非表示
- ◆ それより前のバージョンの場合はServerTokensとServerSignatureを共に設定する必要あり

<上記の設定になっていない場合>



```
satsuki:~ # telnet 123.456.789.123 80
Connected to 123.456.789.123.
.....
<address>IBM_HTTP_Server/7.0.0.0 (Win32) at
hogehoge.japan.ibm.com Port 80</address>
</body></html>
Connection closed by foreign host.
```

<上記の設定になっている場合>



```
satsuki:~ # telnet 123.456.789.123 80
Connected to 123.456.789.123.
.....
<address>IBM_HTTP_Server at
hogehoge.japan.ibm.com Port 80</address>
</body></html>
Connection closed by foreign host.
```

TRACEメソッドの無効化

■ 目的

- ◆ クライアントが送信したリクエストメッセージをそのまま返す機能であるTRACEメソッドが有効になっていると、悪意のあるユーザーが別ユーザーのブラウザがこのTRACEメソッドを発行するように仕向けてそのレスポンスの中のパスワードを奪うという Cross Site Tracing と呼ばれる攻撃を受ける可能性があるため、そのためのセキュリティ対策

■ 設定方法

- ◆ 構成ファイル httpd.conf で以下のように設定

```
TraceEnable Off
```

この1行を追加

■ バージョン差異による考慮事項

- ◆ IHSV7.0(Apache2.0.55以降)より前のバージョンの場合は以下のように設定する必要あり

```
LoadModule rewrite_module modules/mod_rewrite.so
RewriteEngine On
RewriteCond %{REQUEST_METHOD} ^TRACE
RewriteRule .* - [F]
```

<上記の設定になっていない場合>

```
satsuki:~ # telnet 123.456.789.123 80
Connected to 123.456.789.123.
OPTIONS / HTTP/1.0
.....
Server: IBM_HTTP_Server
Allow: GET,HEAD,POST,OPTIONS,TRACE
.....
Connection closed by foreign host.
```

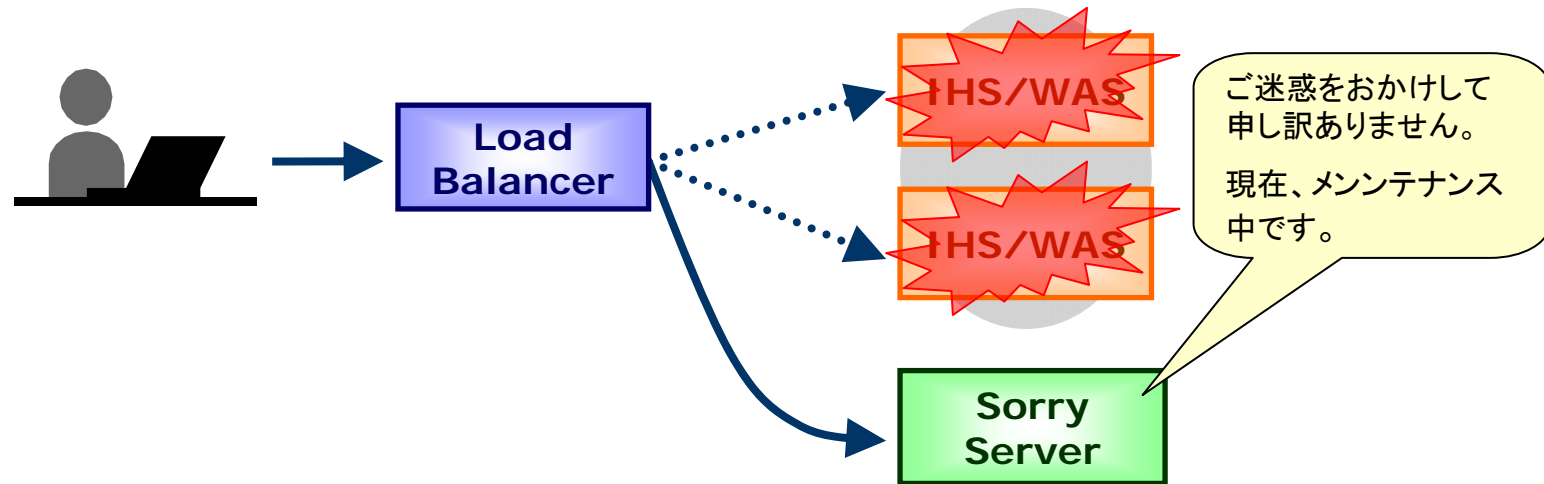
<上記の設定になっている場合>

```
satsuki:~ # telnet 123.456.789.123 80
Connected to 123.456.789.123.
OPTIONS / HTTP/1.0
.....
Server: IBM_HTTP_Server
Allow: GET,HEAD,POST,OPTIONS
.....
Connection closed by foreign host.
```

3. Sorry Server / エラー画面

Sorry Server とは

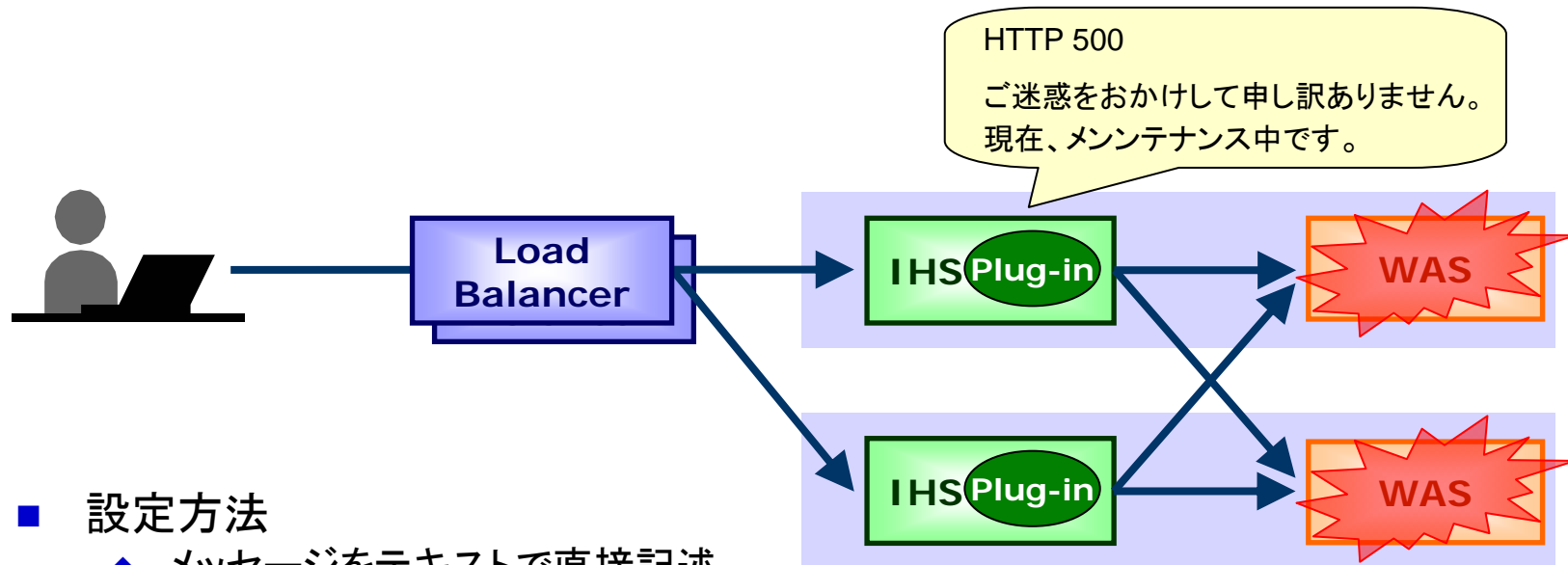
- リクエストが処理できない場合に代理で応答を返すサーバー
 - ◆ Webサーバーのダウン、過負荷状態、サービス時間外、メンテナンスなど



- Sorryサーバーに必要な機能
 - ◆ あらゆるリクエストに対してSorryコンテンツを返す
 - 負荷分散装置から、あらゆるコンテンツへのリクエストがそのまま転送されてくる
 - ◆ コンテンツをプロキシやクライアントにキャッシュさせない
 - キャッシュされてしまうと、サーバーが復旧しているのにクライアントにはSorryコンテンツが見えてしまう
 - ◆ クライアントとの接続を必要以上に維持しない
 - サイトダウンや過負荷時には、Sorryサーバーに接続が集中する

ErrorDocumentディレクティブ

- ErrorDocumentディレクティブを使って、指定のエラーコードに対するレスポンスをカスタマイズする事が可能
 - ◆ Sorryサーバーがない環境でも、ErrorDocumentをうまく使うことで事足りる場合もある



- 設定方法
 - ◆ メッセージをテキストで直接記述
 - ◆ ローカルファイルへのリライト
 - ◆ 外部サイトへのリダイレクト

```
ErrorDocument 500 "The server made a boo boo."
ErrorDocument 404 /missing.html
ErrorDocument 402 http://www.example.com/subscription_info.html
```

Sorry Server の設定 (1/3)

■ 基本的な設定方法

◆ AliasMatchディレクティブを使ってあらゆるリクエストを処理

- Sorryページに付随する画像やCSSへのリクエストがAliasMatchされないように注意が必要
- AliasMatchディレクティブよりも先に処理されるようにAlias設定する

```
Alias /sorry.gif /usr/IHS/htdocs/ja_JP/sorry.gif
AliasMatch ^/* /usr/IHS/htdocs/ja_JP/sorry.html
```

◆ 同様の設定をmod_rewriteで行う例

```
LoadModule rewrite_module modules/mod_rewrite.so
<IfModule mod_rewrite.c>
  RewriteEngine On
  RewriteCond %{REQUEST_URI} !=/sorry.gif
  RewriteRule ^.*$ /sorry.html [L]
</IfModule>
```

■ 考慮点

- ◆ レスポンスのステータスコードが200となるため、検索エンジンがSorryページであることを判別せずにキャッシュする可能性がある
 - インターネットサイトでは503を返す方法がベター

Sorry Server の設定 (2/3)

- 503 (Service Unavailable)でエラーページとして設定する方法
 - ◆ RedirectMatchディレクティブを使ってリクエストをエラーページへリダイレクト
 - 実際にはリダイレクト処理はされずにリライト処理されるようにする
 - エラーページ自身がRedirectMatchによって処理されないように注意

```

ErrorDocument 503 /sorry.html
RedirectMatch 503 ^/(?!sorry¥.html$)
    
```

- エラーページに画像が含まれる場合、サイト構成や正規表現に工夫が必要となる

```

ErrorDocument 503 /sorry.html
RedirectMatch 503 ^/(?!sorry¥.(html|gif)$)
    
```

- ◆ IHS 7.0 (Apache 2.2)では同様の設定をmod_rewriteで行うことが可能
 - IHS 6.1 (Apache 2.0)以前ではリダイレクトに500番台のステータスコードが指定できないため不可

```

ErrorDocument 503 /sorry.html

LoadModule rewrite_module modules/mod_rewrite.so
<IfModule mod_rewrite.c>
  RewriteEngine On
  RewriteCond %{REQUEST_URI} !=/sorry.html
  RewriteCond %{REQUEST_URI} !=/sorry.gif
  RewriteRule ^.*$ - [R=503,L]
</IfModule>
    
```

Sorry Server の設定 (3/3)

- クライアントとの接続を必要以上に維持しない(本番サーバー回復後にすぐに割り振り再開させる)ために、KeepAliveのチューニングが必要

<デフォルト値>

```
KeepAlive On  
MaxKeepAliveRequests 100  
KeepAliveTimeout 10
```

- ◆ 基本的にはKeepAliveはOffにする
- ◆ Sorryページに画像やCSSなど付随するファイルがある場合、MaxKeepAliveRequestsをファイル数に合わせて設定し、KeepAliveTimeoutを1秒など短く設定するのがベター

Sorryコンテンツ

- Sorryコンテンツに必要な設定
 - ◆ ブラウザにキャッシュさせないためのmetaタグを設定する
 - ◆ ファイルへのリンクはサイトルート相対URLで記述する

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<title>test</title>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Expires" content="0">
<meta http-equiv="Pragma" content="No-cache">
<meta http-equiv="Cache-Control" content="no-cache">
</head>
<body>
<h1>ただいまメンテナンス中です。</h1>

</body>
</html>

```

ブラウザにキャッシュ
させないための設定

ファイルへのリンクは
サイトルート相対URLまたは
絶対URLで記述する

(絶対URL)・・・http://www.ibm.com/index.html
(サイトルート相対URL)・・・/index.html
(相対URL)・・・./index.html

参考文献

参考文献

- Apache HTTP サーバドキュメンテーション
 - ◆ Apache HTTP サーババージョン 2.2 ドキュメント
 - <http://httpd.apache.org/docs/2.2/>
- Information Center
 - ◆ IBM HTTP Server for WebSphere Application Server バージョン 7.0
 - http://publib.boulder.ibm.com/infocenter/wasinfo/v7r0/index.jsp?topic=/com.ibm.websphere.ihs.doc/info/welcome_ihs.html
- ワークショップ資料
 - ◆ WAS V6 Edge Components & IHSインプリメンテーション・ワークショップ資料
 - http://www.ibm.com/developerworks/jp/websphere/library/was/edge6_ihs6/
 - ◆ IBM HTTP Server 1.3.19 解説
 - https://www.ibm.com/developerworks/jp/websphere/library/was/ihs13_guide/5.html
- その他
 - ◆ 【ガイド】IHSを80番ポートで非rootユーザー起動する方法(AIX V6.1のみ)
 - <http://www-06.ibm.com/jp/domino01/mkt/cnpages1.nsf/page/default-00129046>