



参考資料

Yellowfin Web サーバー連携ガイド

Version 7.4

October 2018

目次

Yellowfin Web サーバー連携 概要	3
Apache httpd の拡張モジュールの準備.....	4
httpd.conf の変更.....	4
00-proxy.conf の変更	4
Apache httpd と Yellowfin を単純に連携する場合	5
proxy-ajp.conf の作成	5
server.xml の変更	5
Apache httpd でロードバランシングを行い Yellowfin と連携する場合	6
proxy-ajp.conf の作成	6
server.xml の変更	7

Yellowfin Web サーバー連携 概要

Yellowfin のデフォルト構成ではバンドルされている Tomcat の Web サーバー機能を使用します。負荷分散や大量の Http アクセス処理などが必要な場合に他の Web サーバーと連携して動作することも可能です。この資料では代表的な Web サーバーである Apache httpd と Yellowfin を連携する設定について概説します。

Yellowfin はバンドルされている Tomcat のインスタンスとして動作します。したがって Apache httpd と Yellowfin を連携させる手順は Apache httpd と Tomcat を連携させる手順に準拠します。

代表的な二つの構成を例に挙げます：

- Apache httpd と Yellowfin を単純に連携する場合
- Apache httpd でロードバランシングを行い Yellowfin と連携する場合

※Apache httpd は Linux プラットフォームのバージョン 2.4 の設定を例とします。

Apache httpd の拡張モジュールの準備

Apache httpd と Tomcat の連携には Apache httpd に同行されている Proxy 系の一連の拡張モジュールを使用します。拡張モジュールを使用するには Apache httpd の起動時に読み込むように設定します。

httpd.conf の変更

Web サーバーの `/etc/httpd/conf/httpd.conf` を開き、オプションモジュール読み込みを有効にします。下記の記述の行が存在するか確認し、存在しない場合は追記、コメント化されている場合はコメント化を解除（先頭の `#` を削除）します。

※通常は全て有効になっています。

```
IncludeOptional conf.d/*.conf
Include conf.modules.d/*.conf
```

00-proxy.conf の変更

Web サーバーの `/etc/httpd/conf.modules.d/00-proxy.conf` を開き、proxy 系の一連のモジュールがロードされるように下記の記述の行のコメント化を解除（先頭の `#` を削除）します。

※通常は全て有効になっています。

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
LoadModule proxy_ftp_module modules/mod_proxy_ftp.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule proxy_ajp_module modules/mod_proxy_ajp.so
LoadModule proxy_connect_module modules/mod_proxy_connect.so
```

Apache httpd と Yellowfin を単純に連携する場合

Apache httpd の拡張モジュールの一つである mod_proxy_ajp を使用して Tomcat と連携します。

mod_proxy_ajp の詳細については下記の URL を参照してください。

https://httpd.apache.org/docs/2.4/ja/mod/mod_proxy_ajp.html

proxy-ajp.conf の作成

Web サーバーの /etc/httpd/conf.d に proxy-ajp.conf という名前のファイルを作成し、下記のように記述します。

```
ProxyRequests Off  
  
ProxyPass / ajp://192.168.1.10:8009/ timeout=1860  
  
ProxyPassReverse / ajp://192.168.1.10:8009/
```

上記の赤太字の部分は連携先の Yellowfin(Tomcat)サーバーの実際の IP アドレスを記述します。
timeout のデフォルト値は 300(秒)なので、必要に応じて時間を伸ばすようにしてください。

server.xml の変更

Yellowfin サーバーの Yellowfin インストールディレクトリ/appserver/conf/server.xml に対して下記の記述の行のコメント化を解除（対象の行を囲っている<!-- -->を削除）します。

```
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
```

文字化けを回避するために同じ行に URIEncoding 属性を下記のように追加します。

```
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" URIEncoding="UTF-8" />
```

Apache httpd でロードバランシングを行い Yellowfin と連携する場合

Apache httpd の拡張モジュールの一つである mod_proxy_balancer を使用して Tomcat と連携します。mod_proxy_balancer の詳細については下記の URL を参照してください。

https://httpd.apache.org/docs/2.4/ja/mod/mod_proxy_balancer.html

Yellowfin が複数ある場合はクラスタリング構成が必須です。Yellowfin のクラスタリング構成については「Yellowfin クラスタリングガイド」を参照してください。

proxy-ajp.conf の作成

Web サーバーの /etc/httpd/conf.d に proxy-ajp.conf という名前のファイルを作成し、下記のように記述します。

```
ProxyRequests Off

ProxyPass / balancer://yf-balancer/ stickysession=JSESSIONID nofailover=Off
ProxyPassReverse / balancer://yf-balancer/ stickysession=JSESSIONID

<Proxy balancer://yf-balancer>
    BalancerMember ajp://192.168.1.10:8009/ route=yf01 loadfactor=10 keepalive=On
connectiontimeout=600000
    BalancerMember ajp://192.168.1.20:8009/ route=yf02 loadfactor=10 keepalive=On
connectiontimeout=600000
</Proxy>
```

上記の赤太字の部分は連携先の Yellowfin(Tomcat)サーバーの実際の IP アドレスを記述します。
上記の青太字は後述の server.xml 内の jvmRoute 属性で指定した値と一致させる必要があります。

balancer-manager を設定することでセッションの振り分け状態を Web 画面で参照することができます。必要な場合はさらに下記の記述を追加してください。

```
ProxyPass /balancer-manager !

<Location /balancer-manager>
    SetHandler balancer-manager
    Order Deny,Allow
    Deny from all
    Allow from 192.168.1.0/24
</Location>
```

上記の赤太字の部分は balancer-manager にアクセスできる IP アドレスを記述します。

balancer-manager へは [http://\[web-servername\]/balancer-manager](http://[web-servername]/balancer-manager) からアクセスします。

server.xml の変更

各ノードの、Yellowfin サーバーの Yellowfin インストールディレクトリ/appserver/conf/server.xml に対して下記の記述の行のコメント化を解除（対象の行を囲っている<!-- -->を削除）します。

```
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
```

文字化けを回避するために同じ行に URLEncoder 属性を下記のように追加します。

```
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" URLEncoder="UTF-8" />
```

さらに Engine 要素に jvmRoute 属性を下記のように追加します。

```
<Engine name="Catalina" defaultHost="localhost">
```



```
<Engine name="Catalina" defaultHost="localhost" jvmRoute="yf01">
```

上記の赤太字の部分は先述の proxy-ajp.conf で指定した route パラメーターの値と一致させる必要があります。また、パラメーターの値は各ノードでユニークにする必要があります。