参考資料

Yellowfin Web サーバー連携ガイド

Version 9

August 2020

目次

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 と Tomcat を 1 対 1 で単純に接続します。

server.xml の変更

AJP Connector を有効にします。同時に AJP Connector の属性も設定します。

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

https://tomcat.apache.org/tomcat-9.0-doc/config/ajp.html

Yellowfin サーバーの Yellowfin インストールディレクトリ/appserver/conf/server.xml の下記の記述を探します。

```
<!-- Define an AJP 1.3 Connector on port 8009
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
-->
```

上記の記述の下に下記のように設定を追記します。

```
<!-- Define an AJP 1.3 Connector on port 8009
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
-->
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443"
           maxThreads="500" minSpareThreads="25"
           enableLookups="false" acceptCount="100"
           connectionTimeout="20000"
           URIEncoding="UTF-8"
           maxPostSize="10485760" maxParameterCount="50000"
           address="TOMCAT_IP_ADDRESS"
           secretRequired="true" secret="SECRET_KEY"
/>
```

属性の補足説明です。

各属性の詳細については下記の URL を参照してください。

https://tomcat.apache.org/tomcat-9.0-doc/config/ajp.htm

- TOMCAT_IP_ADDRESS

  リッスンする IP アドレスです。

  通常は Tomcat が稼働しているマシンの IP アドレスを指定します。

  接続する Web サーバーが同一マシンで稼働しており、Tomcat を localhost で参照する場合は address 属性を削除するか、”127.0.0.1”を指定します。

- SECRET_KEY

  この AJP に接続する際に指定する秘密鍵（パスワード）を設定します。

  ※secretRequired 属性に”false”を設定することで secret 属性を省略（無効）することができますが AJP Connector のセキュリティレベルが低下しますのでお勧めいたしません。

proxy-ajp.conf の作成

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

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://TOMCAT_IP_ADDRESS:8009/ secret=SECRET_KEY timeout=1860
ProxyPassReverse / ajp://TOMCAT_IP_ADDRESS:8009/ secret=SECRET_KEY
```

各指定値の補足説明です。

- TOMCAT_IP_ADDRESS

  server.xml で指定した TOMCAT_IP_ADDRESS を指定します。

  通常は Tomcat が稼働しているマシンの IP アドレスを指定します。

  server.xml の address 属性を省略した場合や”127.0.0.1”に設定している場合は proxy-ajp.conf では localhost あるいは”127.0.0.1”を指定します。

- SECRET_KEY

  server.xml で設定した SECRET_KEY を指定します。AJP に接続する際に指定する秘密鍵（パスワード）を設定します。

  ※server.xml で secretRequired 属性に”false”を設定している場合は secret の指定を省略できます。

- Timeout

  Httdp 側のタイムアウト値（秒）です。

  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。

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 と Tomcat を 1 対 n で接続し、ロードバランシングを行います。

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は Yellowfin クラスタリングガイドを参照してください。

server.xml の変更

Yellowfin の各ノードの AJP Connector を有効にします。また、ロードバランシングの際のノード識別で必要となる jvmRoute の設定を行います。

Yellowfin の各ノードの Yellowfin インストールディレクトリ/appserver/conf/server.xml を「Apache httpd と Yellowfin を単純に連携する場合＞server.xml の変更」を参照して編集し、AJP Connector を有効にします。

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="JVM_ROUTE_NAME">
```

属性の補足説明です。

- JVM_ROUTE_NAME

  ロードバランシングする際にノードを識別するために必要です。

  ロードバランシングするメンバー内でユニークである必要があります。

proxy-ajp.conf の作成

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

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:// TOMCAT_NODE1_IP_ADDRESS:8009/ route=jvmRoute1
secret=SECRET_KEY1 loadfactor=10 keepalive=On connectiontimeout=600000
        BalancerMember ajp:// TOMCAT_NODE2_IP_ADDRESS:8009/ route= jvmRoute2
secret=SECRET_KEY2 loadfactor=10 keepalive=On connectiontimeout=600000
</Proxy>
```

各指定値の補足説明です。

- TOMCAT_NODE(n)_IP_ADDRESS

  接続するノードの server.xml で指定した TOMCAT_IP_ADDRESS を指定します。

  通常は Tomcat が稼働しているマシンの IP アドレスを指定します。

  server.xml の address 属性を省略した場合や”127.0.0.1”に設定している場合は proxy-ajp.conf では localhost あるいは”127.0.0.1”を指定します。

- SECRET_KEY(n)

  接続するノードの server.xml で設定した SECRET_KEY を指定します。AJP に接続する際に指定する秘密鍵（パスワード）を設定します。

※接続するノードの server.xml で secretRequired 属性に”false”を設定している場合は secret の指定を省略できます。

- jvmRoute(n)

  接続するノードの server.xml で設定した JVM_ROUTE_NAME を指定します。

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 からアクセスします。