

# Configuring an External PID Minter

To configure an external PID minter, create a Spring bean configuration file using the `HttpPidMinter` class, and use a system property to use it instead of the internal PID minter:

```
-Dfcrepo.minter.config=file:///path/to/minter.xml -Dfcrepo.minter.username=minterUser -Dfcrepo.minter.password=minterPass
```

Example `minter.xml`:

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:c="http://www.springframework.org/schema/c"
  xsi:schemaLocation="http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">

  <bean class="org.fcrepo.kernel.identifiers.HttpPidMinter"
    c:url="http://localhost/my/minter" c:method="POST"
    c:username="${fcrepo.minter.username:minterUsername}"
    c:password="${fcrepo.minter.password:minterPassword}"
    c:regex=" " c:xpath="/response/ids/value"/>

</beans>
```

The properties that can be set are:

- **url (required):** The URL of the external minter REST service.
- **method:** The HTTP method to use to generate a new identifier: GET, POST, or PUT. If not specified, POST is used.
- **username/password:** The username and password to use for connecting to the minter REST service. You may want to pass the username /password using System properties instead of hard-coding them in the config file (the example above shows how to use a System property with a fallback value if the property is not set). If not specified, no authentication is used.
- **regex:** If set, remove text matching the regular expression from the minter service response. For example, if the minter response is "id: foo /abc123", the regular expression ".\*/" would remove the "id: foo/" and result in the identifier "abc123".
- **xpath:** If set, parse the minter service response as XML and use XPath to get the identifier value to use.