z2-Environment - Feature #2098

Provide a Whitelisting Filter Handler in Jetty Configuration to limit access to Web Applications for non-localhost access

15.08.2021 13:11 - Henning Blohm

Status:	Resolved	**Start date:**	15.08.2021
Priority:	Normal	**Due date:**	
Assignee:	Henning Blohm	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:	2.9		
origin:			

Description

We implement a simple filtering for regex patterns that qualify for remote access while limiting access to Web applications to localhost by default.

This is a simple protective measure allowing to run /adm (web admin) and development related paths with no or little protection from localhost while requiring

a dedicating configuration for non-localhost access.

History

#1 - 15.08.2021 13:13 - Henning Blohm

- Status changed from In Progress to To Be Documented

#2 - 15.08.2021 20:09 - Henning Blohm

- Status changed from To Be Documented to Resolved

Documentation can be found in the manual and How to secure a Z2 installation:

How to Secure a Z2 Installation

This how to is looking into some very basic measures to implement to provide basic protection to a z2 installation. While we

are considering z2 here, these suggestions apply to pretty much any Web application system.

Run a Firewall to Block Port Access

Java applications typically have more ports open than you think. This could be JMX related, debugging support etc. The same is true for your operating system. So in general make sure that only those ports are accessible that are required to run your application. On Linux this may well be just one port for SSH access (i.e. 22 by default) and one for Web application access.

There are some variations on the latter. In most cases your application is not the actual entry point for Web access but instead there will be some request routing happening before to make sure maintenance scenarios (and outages) and load-balancing can be dealt with and most importantly for SSL termination.

Instead of protecting every single server node of your installation, you may consider setting up a Virtual LAN setup where you can concentrate all access limitations and rules to securing a single gateway node.

The z2 Environment hasa not particular means for managing port-based access and indeed the reason for this section is to make you aware of this fact.

Typically the following ports will be used by default with z2:

Port	Purpose	Configuration
8080	Web Container (Jetty)	environment.base/webServer/jetty-http.xml
5000	Java debug port for home process	$Z2_HOME/bin/launch.properties
5100+x	Java debug port for web worker process	environment.base/webWorker.properties
7800+x	JMX port for web worker process	environment.base/webWorker.properties

However depending on your configuration other ports may be opened. Also note How_to_Remote_Manage.

Use Whitelisting for Web Applications

In contrast to this, you may want to also limit access to specific Web applications. For example, you should make sure that access to development or management related Web applications should be tightly restricted to avoid any possibility of triggering unexpected changes on your production environment.

The simplest way of achieving that is by allowing access from any other host by localhost only for dedicated Web applications.

In short this means:

- Only outward facing Web application usage is available by default
- Any other access requires access from localhost.

The latter is a brilliant limitation as it means you can use lower level means such as SSH tunneling to secure limited access by the same policy you use to limit management access to execution environments.

Starting with Version 2.9 (#2098) the Jetty configuration of z2-base prohibits access to any built-in Web application from anywhere but locahost.

The configuration is part of environment/webServer (which links to environment.base/webServer).

See also z2-nonlocalhostwhitelist.xml.

Given a Web application that uses the context path /abc , you would enable access to it by adding the line

```
<Item>^/abc($|/.*)</Item>
```

Built-in Web Users Realm for Built-In Web Apps

The z2-base distribution comes with a number of built-in Web application that are (by default) started upon attaining the system state **environment/webWorker** that is the target state of the default Web worker process defined in environment/webWorker (note: You can change all that).

These Web applications require basic authentication with the simple user realm defined in environment/webUsers that is configured with the default Jetty Web server configuration.

Web Application Component	Context Path	Purpose	Role requirement
com.zfabrik.admin/web	/adm	Simple Web Administration	admin
com.zfabrik.dev.eclipsoid.srv/web	/eclipsoid	Server side of Eclipsoid development support	eclipsoid
com.zfabrik.dev.javadoc/web	/javadoc	Javadoc access	<none>
com.zfabrik.dev.z2jupiter/web	/z2jupiter	JUnit 5 In-System testing	tester
com.zfabrik.dev.z2unit/web	/z2unit	JUnit 4 In-System testing	tester

By default, the user realm defines the user “z*” with password “z” that has all required roles (and is used by default in JUnit clients – see How to Unit Test in Z2).