AEM Gems – Dispatcher Caching – New Features and Optimizations

Dominique Pfister | Senior Computer Scientist
Andrew Khoury | Senior Customer Satisfaction Engineer

Bē
Evolution Bureau

What’s Covered

- New Dispatcher Features (since Dispatcher 4.0.9 through 4.1.9)
  - New format in `/filter` rules
  - New format for DispatcherPassError
  - Using environment variables in dispatcher.any
  - ELB and `/always-resolve`
  - Automatic insertion of vanity URLs

/filter rules - Purpose

- The filter rules determine what requests are handled by the dispatcher
- It can be used to deny requests that should not reach AEM

/filter rules - Old style

- In the old style, rules are matched against the complete request URL, e.g.

```
GET /index.html HTTP/1.1
```

- It was easy to forget that the protocol is part of the request URL, e.g

```
  /filter {
   /0001 {
     /glob “GET /index.html”
     /type “allow”
   }
}
```

would not match the request above. Instead, one would need to specify:

```
/glob “GET /index.html *”
```

/filter rules - Old style (continued)

- Some rules could actually be circumvented by forging requests:

```
  /filter {
   { /glob “* /crx/de/* *” /type “deny” }
   { /glob “* *.css *” /type “allow” }
}
```

would still let through the following request:

```
GET /crx/de/index.jsp?foo=.css HTTP/1.0
```

- This was considered a security vulnerability (DISP-407)

/filter rules - New style

- New rule style is structured:

```
  /filter {
   { /type “deny” /url “/crx/de/*” }
   { /type “allow” /method “GET” /url “*.css” }
}
```

This is the correct approach to deny every access to CRXDE but still allow GET requests to CSS files.

- Other items allowed in new rules:
  - `/query`
  - `/protocol`
- Configuration samples in the distribution packages are all converted to new style

New format for DispatcherPassError

- DispatcherPassError is an Apache-specific configuration
- When set to “1”, it allowed error documents set with ErrorDocument directives to be processed.
- This proved to be too generic, so we changed it to allow status code ranges:

```
DispatcherPassError 400,402-404,500
```

Using environment variables in dispatcher.any

- String-valued entries in dispatcher.any can contain environment variables
- Assume our Apache is setup in some directory like this:

```
 $ ls -l
total 40
drwxr-xr-x   8 dpfister  staff    272 Oct 29 13:54 cache
lrwxr-xr-x   1 dpfister  staff     26 Oct 28 16:24 dispatcher.any
-rwxr-xr-x   1 dpfister  staff     57 Apr 19  2013 apachectl
-rw-r--r--   1 dpfister  staff  20755 Nov 26 15:27 httpd.conf
```

- We start Apache using the script apachectl from this directory. This allows us to set the cache root in dispatcher.any as follows:

```
 /cache {
  /docroot "${PWD}/cache"
}
}
```

ELB and /always-resolve

- Elastic load balancers have a permanent DNS name but can switch IP addresses
- The dispatcher initially resolved DNS name to IP addresses once, on startup
- In an ELB situation, it is therefore advisable to use:

```
/always-resolve “1”
```

- Moreover, for an ELB, a host name maps to multiple IP addresses
- The modern `getaddrinfo(3)` system call will always return the same sequence of IP addresses, while the old `gethostbyname(3)` will return them in random order
- To load balance better through the list of IP addresses, it is therefore advisable to use:

```
/ipv4 “1”
```

More info: http://daniel.haxx.se/blog/2012/01/03/getaddrinfo-with-round-robin-dns-and-happy-eyeballs/

Automatic insertion of vanity URLs

- With a (simplified) setup of filter rules as follows:

```
  /filter {
   { /type “deny”  /url “/*” }
   { /type “allow” /url “/content” }
}
```

builtin vanity URLs, e.g. redirecting `/geohome` to `/content/geometrixx/en.html` - will not work, but return a 404 (Not found).

- To this date, only workaround was to manually add exceptions for vanity URLs to the filter list:

```
{ /type “allow”  /url “/geohome” }
```

which is cumbersome.

Automatic insertion of vanity URLs (continued)

- Enter new vanity URL support (in 4.1.9):

```
 /vanity_urls {
  /file “/tmp/vanity_urls”
  /url “/libs/granite/dispatcher/content/vanityUrls.html”
  /delay “300”
}
```

- The dispatcher will periodically fetch the list of vanity URLs from the publish instance and store it locally.
- If some request matches a vanity URL, it will be passed to the backend - exactly, as if it was marked as an allowed url in the /filter section

Adobe