

CommandBox 5.2.0-RC.1 Release Candidate ready for testing (<https://www.ortussolutions.com/blog/commandbox-520-release-candidate-ready-for-testing>)



Brad Wood

Oct 12, 2020

It's been 4 months since we've had a CommandBox release (<https://www.ortussolutions.com/blog/commandbox-511-released>), but we've actually been quite busy on a number of large improvements that took a while to settle down. Today we are pleased to announce a Release Candidate 5.2.0-RC.1 for you to help test. There's not usually a release candidate for "minor" CommandBox releases, but we've updated a lot of libraries and introduced some pretty big new features so we wanted to have a round of testing and feedback before we cut the final release. There are 46 completed tickets for the 5.2.0 release.



Docs and Downloads

Before we get into the new juicy features, everything has been documented already in the CommandBox docs here:

<https://commandbox.ortusbooks.com/> (<https://commandbox.ortusbooks.com/>)

And the RC.1 builds can be downloaded from our artifact repository here:

<https://downloads.ortussolutions.com/#/ortussolutions/commandbox/5.2.0-rc.1/> (<https://downloads.ortussolutions.com/#/ortussolutions/commandbox/5.2.0-rc.1/>)

New Stuff!

Ok, where to begin. There are a number of pretty exciting new features and a pile of bug fixes. And as usual, input from the community via Pull Requests. Huge thanks to **Pete Freitag**, **Kai Koenig**, **Matthew Clemente**, **Bobby Hartsfield**, **Scott Steinbeck**, **Daniel Mejia**, and **Miguel Mathus**! Here's an overview of the new stuff in 5.2.0

Library Updates

We know library updates are boring, but they are important and we want you to know we take them seriously. Keeping up-to-date ensure you have the latest fixes and security updates from all the third-party libs we bundle in CommandBox.

Here's an overview of what we updated in CommandBox 5.2.0.

- Upgraded **Runwar** from 4.1.2 to 4.3.8
- Upgraded **JBoss Undertow** from 2.0.27.Final to 2.2.0
- Upgraded **UrlRewritesFilter** to Ortus fork 5.0.1 with custom fixes
- Upgraded **jboss-logging** from 3.2.1.Final to 3.4.1.Final
- Upgraded **jboss-logging-annotations** from 2.1.0 to 2.2.1.Final
- Upgraded **JCabi Log** from 0.18 to 0.18.1
- Upgraded **Apache HttpClient** from 4.2.6 to 4.5.12
- Upgraded **Apache httpmime** from 4.2.6 to 4.5.12

- Removed unused **JOpt Simple** 5.0-beta-1
- Removed unused **Gson** 1.2.3

The Undertow bump is a minor update, but a pretty big deal. Ortus sent three pull requests to the core Undertow project fixing bugs and adding predicate logging. All of our pulls were accepted and merged into the core Undertow project and released in the 2.2 release.

Read more about library updates here:

<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1064> (
<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1064>)

Server Security Profiles

This is a feature that we expect to grow in the future. We've started it out simple, yet powerful but left a lot of room to build on it. The two main goals here are

- Make CommandBox secure-by-default so a server shoved in production comes nice and locked down
- Makes it very easy for you to toggle off all the security stuff for development

CommandBox now has profiles you can assign to a server when you start it to configure the default settings. This is to provide easy secure-by-default setups for your production servers, and to make it easier to switch between a development mode and production mode.

There are 3 currently supported profiles. Custom profiles will be added as a future feature.

- **Production** - Locked down for production hosting
- **Development** - Lax security for local development

- **None** - For backwards compat and custom setups. Doesn't apply any web server rules

In **production** mode, CommandBox will block access to your CF admin to all external traffic, will block all common config files such as **box.json** or **.env** and will block, the "TRACK" and "TRACE" HTTP verbs

You can set the profile for your server in your server.json

```
server set profile=production
```

Or you can specify it when starting the server like so:

```
server start profile=production
```

If a profile is not set, CommandBox looks for an environment variable called "environment" or it checks to see if the site is bound on localhost to try and guess the correct profile for you.

We've also added some new flags in your **server.json** to fully customize how your profile behaves.

```
server set web.blockCFAdmin=true
server set web.blockCFAdmin=false
server set web.blockCFAdmin=external

server set web.blockSensitivePaths=true
server set web.blockSensitivePaths=false

server set web.blockFlashRemoting=true
server set web.blockFlashRemoting=false
```

Read more about Server Profiles here:

<https://commandbox.ortusbooks.com/embedded-server/configuring-your-server/server-profiles> (<https://commandbox.ortusbooks.com/embedded-server/configuring-your-server/server-profiles>)

Server Rules

This is huge-- probably the biggest chunk of work, and it's actually what makes the server profiles above even possible! It's always been possible to perform basic lock downs with a custom rewrite file, but we've exposed an amazing built-in functionality of Undertow called the Predicate language. It allows you to create ad-hoc rules that apply to your server to provide any of the following:

- **Security** - Block paths, IPs, or users
- **URL rewrites** - Rewrite incoming URLs to something different
- **Modifying HTTP requests** on the fly - Set headers, cookies, or response codes

An example of a server rule using Undertow's predicate language to block access to any box.json files looks like this:

```
path-suffix(/box.json) -> set-error(404)
```

One of the best things about these rules, is they don't have to be in a single monolithic XML file. Instead they can come from

- An array of ad-hoc definitions in your **server.json** file or config server defaults
- one or more external JSON or text file specified in your **server.json** or config server defaults
- Built in CommandBox server profiles (see above)

- Custom 3rd party CommandBox modules that contribute rules on-the-fly (time to get creative!)

Here's some examples of what can be in your **server.json**

```
{
  "web" : {
    "rules" : [
      "path-suffix(/box.json) -> set-error(404)",
      "path-suffix(hidden.js) -> set-error(404)",
      "path-prefix(/admin/) -> ip-access-control(192.168.0.* allow)",
      "path(/sitemap.xml) -> rewrite(/sitemap.cfm)",
      "disallowed-methods(trace)"
    ],
    "rulesFile" : "../secure-rules.json"
    // Or...
    "rulesFile" : ["../security.json","../rewrites.txt","../app-headers.json"]
    // Or...
    "rulesFile" : "../rules/*.json"
  }
}
```

There are TON of built in predicates and handlers your rules can use. We've documented some of them here (<https://commandbox.ortusbooks.com/embedded-server/configuring-your-server/server-rules/rule-language>):

CommandBox also registers some custom rules in Undertow you can use for your CF apps:

```
// Block all CF admin access
cf-admin() -> set-error( 404 );

// Shortcut for the previous rule
block-cf-admin()

// Block external CF admin access
cf-admin() -> block-external()
```

There are lots of new docs on this. Read more about Server Rules [here](#):

<https://commandbox.ortusbooks.com/embedded-server/configuring-your-server/server-rules> (<https://commandbox.ortusbooks.com/embedded-server/configuring-your-server/server-rules>)

Task Runner Lifecycle events

The more we use Task Runners for builds, scheduled tasks, and utilities, we've seen the need to have lifecycle events in the same manner as the preHandler and postHandler sort of stuff in ColdBox MVC. Now if a task runner has methods of this name, they will be executed automatically.

- **preTask** - Before any target in the task
- **postTask** - After any target in the task
- **aroundTask** - Wraps execution of any target in the task
- **pre<targetName>** - Before a specific target
- **post<targetName>** - After a specific target
- **around<targetName>** - Wraps execution of a specific target
- **onComplete** - Fires regardless of exit status
- **onSuccess** - Fires when task runs without failing exit code or exception
- **onFail** - Fires if exit code is failing after the action is done (always fires along with onError, but does not receive an exception object). Use this to respond generally to failures of the job.
- **onError** - fires only if an unhandled exception is thrown and receives exception object. Use this to respond to errors in the task. Does not fire for interrupted exceptions
- **onCancel** - Fires when the task is interrupted with Ctrl-C

The lifecycle methods are very powerful and can be controlled via whitelist and blacklists to control what targets they execute for. "Around" events are very easy to use thanks to the use of closure. There's a lot more details in the docs.

Read more about Task Runner Lifecycle events here:

<https://commandbox.ortusbooks.com/task-runners/lifecycle-events> (
<https://commandbox.ortusbooks.com/task-runners/lifecycle-events>)

System Setting `${}` Namespaces

The default namespace when using the `$ {}` system setting expansion syntax is box environment variable, Java system properties, and OS environment variables.

It is also possible to leverage built-in namespaces to allow expansions that reference:

- **server.json** properties
- **box.json** properties
- arbitrary **JSON file** properties
- **Config settings** (like the config show command)
- **Server info properties** (like the server info property=name command)
- **Other properties** in the same JSON file

This gives you a lot more power now to be able to create dynamic configuration in your JSON files and even from the command line. Here are some examples:

```
// Reference box.json file in this directory
$ {}boxjson.slug
```



```
// Reference server.json file in this directory
$ serverjson.web.http.port:80}

// Reference local server details
$ {serverinfo.serverHomeDirectory}

// Reference arbitrary JSON file
$ {json.myProperty@file.json}

// Reference CLI's config settings
$ {configsetting.endpoints.forgebox.apitoken}

// Local reference to a JSON property in the same file
{
  "appFileGlobs" : "models/**/*.*cfc,tests/specs/**/*.*cfc",
  "scripts":{
    "format":"cformat run $ {@appFileGlobs} --overwrite",
    "format:check":"cformat check $ {@appFileGlobs} --verbose"
  }
}
```

And one of the coolest things is this implementation is driven by a new **onSystemSettingExpansion** interception point and completely extendable! That means you can write a module that powers something hypothetical like this:

```
$ {AWSecretStore.mySecretKey}
```

Read more about System Setting namespaces here:

<https://commandbox.ortusbooks.com/usage/system-setting-expansion-namespaces> (<https://commandbox.ortusbooks.com/usage/system-setting-expansion-namespaces>)

GZip Compression Control

CommandBox has had the ability to enable/disable GZip compression in Undertow for a while. Now you can fully control when it activates based on the type or size of file, etc. This feature utilizes the same Undertow predicate language that we introduced above.

```
server set web.gzipEnable=true
server set web.gzipPredicate="not path-prefix( admin ) and regex( '(.*).css' ) and request-size( 10000 )"
```

Read more about GZip Compression Control here:

<https://commandbox.ortusbooks.com/embedded-server/configuring-your-server/gzip-compression> (<https://commandbox.ortusbooks.com/embedded-server/configuring-your-server/gzip-compression>)

Generic Watch Command

This is a fun one. There are some specific commands that make use of the Watcher library in CommandBox such as **testbox watch** and **coldbox watch-reinit**. However, there is also now a generic **watch** command that will run any arbitrary command of your choosing when a path matching your file globbing pattern is added/updated/deleted. Use this to build custom watchers on-the-fly without needing to touch any CFML code to write a Task Runner.

```
watch *.json "echo 'config file updated!'"
```

That command will echo out "config files updated!" every time a JSON file gets changed in the current directory. Here's a more complex one:

```
set command = "echo 'You added \$ {item}!'"  
watch command="foreach '\$ {watcher_added}' \$ {command}" --verbose
```

That one will list every new file that's added in this directory and all sub directories.

Read more about the generic **watch** command here:

<https://commandbox.ortusbooks.com/usage/watch-command> (
<https://commandbox.ortusbooks.com/usage/watch-command>)

Control Default Browser

There are a handful of features in CommandBox that will open URLs for you in your default browser. We've had requests to allow the browser in use to be customized, so we've reworked all of that logic, consolidating it in some places and now you can control what browser CommandBox uses. To change the default browser for all URL opening functions use this:

```
// use Chrome  
config set preferredBrowser=chrome  
  
// use FireFox  
config set preferredBrowser=firefox  
  
// Just kidding, no one is going to use this!!  
config set preferredBrowser=ie
```

Supported browsers are:

- **firefox**
- **chrome**
- **opera**
- **edge** (Windows and Mac only)
- **ie** (Windows only)

- **safari** (Mac only)
- **konqueror** (Linux only)
- **epiphany** (Linux only)

And you can even dial in a browser on demand for the **browse** and **server open** commands.

```
server open browser=opera
```

Read more about setting the default browser here:

<https://commandbox.ortusbooks.com/config-settings/misc-settings#preferredbrowser> (<https://commandbox.ortusbooks.com/config-settings/misc-settings#preferredbrowser>)

Miscellaneous

Here are some honorable mentions.

.htaccess rewrite flags

The CommandBox Tuckey rewrites allow an .htaccess file that uses the mod_rewrite style syntax of rewrites. Previously, use of flags such as these didn't work:L

```
RewriteRule ^/login.cfm$ /condworks.html [R=301]
```

<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1221> (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1221>)

Server restart from tray icon

We've added a "**restart**" option to the tray icon that does exactly what you think it does.

Trick for "cd"ing up directories

There's a new trick supported in CommandBox's shell that we've borrowed that allows you to change directories and go "up" more than one directory with less typing:

```
// current directory
cd . -> ./
// back 1 directory
cd .. -> ../
// back 2 directories
cd ... -> cd ../../
// back 3 directories
cd .... -> cd ../../../
```

<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1209> (
<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1209>)

Pipe into standard input of native binaries

You can now pipe the output of a previous command in CommandBox directly to a native binary like so:

```
#createguid | !clip
or
#createguid | run clip
```

In this case, `clip` is a Windows binary that will read the standard input and place that text on the clipboard.

<https://commandbox.ortusbooks.com/usage/execution/os-binaries#piping-to-the-native-binarys-standard-input> (
<https://commandbox.ortusbooks.com/usage/execution/os-binaries#piping-to-the-native-binarys-standard-input>)

Breaking Changes

We work hard to make every CommandBox upgrade backwards compatible. There's a couple things that you may notice different in this release. They're both done to put security first and can be modified to get your original behavior back.

Since the CF Administrator is now blocked for traffic not coming from localhost when in production mode, you may need to explicitly open up the CF admin to make it accessible again if you needed it open to the public on a production server. Even with the **profile** set to **production**, you can activate just the CF admin like so:

```
server set web.blockCFAdmin=false
```

The web server built into CommandBox will now only serve static files if their extension is found in a whitelist of acceptable files. This is to prevent prying eyes from hitting files they shouldn't be able to access on your server. The current list of valid extensions is:

```
3gp,3gpp,7z,ai,aif,aiff,asf,asx,atom,au,avi,bin,bmp,btm,cco,crt,css,deb,der,dmg,doc,d
```

If you have a common static file you need to serve, you can add your own custom extensions to the list like so:

```
server set web.allowedExt=jar,exe,dll
```

And if you think we've missed an obvious one that deserves to be added to the default list, please let us know.

Release Notes

Here's the full list of tickets in the 5.2.0-RC.1 release.

Bug

- [COMMANDBOX-1138 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1138>)] - Tuckey UrlRewrite DTD version issues
- [COMMANDBOX-1141 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1141>)] - when installing a package which doesn't exist, commandbox claims forgebox is unreachable
- [COMMANDBOX-1199 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1199>)] - Remove mail-4.1.1.jar from runwar's lib dir
- [COMMANDBOX-1201 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1201>)] - "testbox run" output garbled on Windows (wrong encoding)
- [COMMANDBOX-1205 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1205>)] - UndertowOptions and XNIOOptions don't work for Long type
- [COMMANDBOX-1206 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1206>)] - HTTP endpoint leaves a zip file in the CommandBox temp folder
- [COMMANDBOX-1213 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1213>)] - url rewrite no longer works
- [COMMANDBOX-1218 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1218>)] - Piping a command into run does not execute interactively
- [COMMANDBOX-1221 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1221>)] - Support flags on .htaccess file for Tuckey rewrites

New Feature

- [COMMANDBOX-126 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-126>)] - Restart server via tray icon
- [COMMANDBOX-1012 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1012>)] - Stop expanding /WEB-INF paths in servlet init params
- [COMMANDBOX-1021 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1021>)] - Allow configuring default browser to use when opening a URL
- [COMMANDBOX-1084 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1084>)] - Add a preInstallAll and postInstallAll interception points when running an `install` command
- [COMMANDBOX-1167 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1167>)] - Add Task Runner lifecycle events
- [COMMANDBOX-1197 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1197>)] - Generic watch command
- [COMMANDBOX-1200 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1200>)] - When starting an already-started server, offer to open the existing one instead
- [COMMANDBOX-1209 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1209>)] - Add directory expansion command for going back multiple directories
- [COMMANDBOX-1214 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1214>)] - Add built-in predicates and handlers for undertow for easier lockdown

- [COMMANDBOX-1215 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1215>)] - Add "profile" setting to help default security settings
- [COMMANDBOX-1220 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1220>)] - Allow standard input to be piped to native binaries

Task

- [COMMANDBOX-1064 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1064>)] - review all the runwar dependencies and check for outdated ones

Improvement

- [COMMANDBOX-1044 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1044>)] - Block TRACE HTTP Verb by default
- [COMMANDBOX-1094 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1094>)] - Implement web server rules in Undertow
- [COMMANDBOX-1103 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1103>)] - Add an option to console log output without ANSI codes
- [COMMANDBOX-1109 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1109>)] - Migrate to AdoptOpenJDK API v3
- [COMMANDBOX-1131 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1131>)] - Move ANSI logging format from Runwar to CommandBox
- [COMMANDBOX-1157 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1157>)] - Automated flag negation hint is the same as the hint for the flag itself

- [COMMANDBOX-1182 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1182>)] - Default server menu actions working directory to web root
- [COMMANDBOX-1183 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1183>)] - Expand working directory when specified for a menu item
- [COMMANDBOX-1191 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1191>)] - Validate incoming version for bump command
- [COMMANDBOX-1192 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1192>)] - Programmatic skipping of package install via interceptor
- [COMMANDBOX-1193 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1193>)] - Adding support for installing lex files from file or unc paths
- [COMMANDBOX-1195 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1195>)] - Add File Filtering For GZIP Compression
- [COMMANDBOX-1196 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1196>)] - Allow default server java version to be cleared
- [COMMANDBOX-1198 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1198>)] - Add `setSystemSetting()` to `BaseCommand`
- [COMMANDBOX-1202 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1202>)] - "testbox run" command - show tag context for global bundle exceptions
- [COMMANDBOX-1203 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1203>)] - Add `--trayEnable` flag to server start

- [COMMANDBOX-1204 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1204>)] - Allow `${}` system setting expansions to have extendable namespaces
- [COMMANDBOX-1208 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1208>)] - Improve verbose output of JVM args if args contain " - " in them
- [COMMANDBOX-1217 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1217>)] - forgeboxstorage default ignores are over-aggressive
- [COMMANDBOX-1219 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1219>)] - If native command is piped into RUN, allow the output to be piped again

Sub-task

- [COMMANDBOX-1185 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1185>)] - Load predicates in Runwar
- [COMMANDBOX-1186 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1186>)] - Pass Predicates as part of Server Start in CommandBox
- [COMMANDBOX-1187 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1187>)] - Add default server rules/predicates in CommandBox for default lockdown
- [COMMANDBOX-1188 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1188>)] - Improve logging in Undertow for execution of predicate handlers
- [COMMANDBOX-1189 (<https://ortussolutions.atlassian.net/browse/COMMANDBOX-1189>)] - Track/address Undertow tickets

www.ortussolutions.com (<http://www.ortussolutions.com>)

© Copyright **Ortus Solutions, Corp.**