SERGIO DEL

How to use Spring Security Core to Secure you Grails 3 App?

25 / 02

11 Replies

Spring Security Core plugin is probably the most famous security plugin in the Grails ecosystem.

Lets create a simple app to publish product announcements:

```
| Grails Version: 3.1.1
| Groovy Version: 2.4.5
| JVM Version: 1.8.0_45
$ grails create-app myapp
| Application created at /Users/shoptimix/Documents/tests/springsecurityexample/myapp
```

I create a domain class to store the product announcements.

```
$ cd myapp
$ grails
grails> create-domain-class ProductAnnouncement
| Created grails-app/domain/myapp/ProductAnnouncement.groovy
| Created src/test/groovy/myapp/ProductAnnouncementSpec.groovy
```

I add a message field to store the announcement and a *dateCreated* field to use Grails autoTimestamp capabilities.

```
package myapp

class ProductAnnouncement {
    String message
    Date dateCreated

    static constraints = {
    }
}
```

Add some ProductAnnouncement messages when the application starts. To do that modify *grails-app/init/BootStrap.groovy*

```
import myapp.*

class BootStrap {

    def init = { servletContext ->
    new ProductAnnouncement(message: 'Launch day').save()
    new ProductAnnouncement(message: 'We keep adding features').save()
    }

    def destroy = {
    }
}
```

Create a controller to show the last message:

```
$ cd myapp
$ grails
grails> create-controller ProductAnnouncement
| Created grails-app/controllers/myapp/ProductAnnouncementController.groovy
| Created src/test/groovy/myapp/ProductAnnouncementControllerSpec.groovy
```

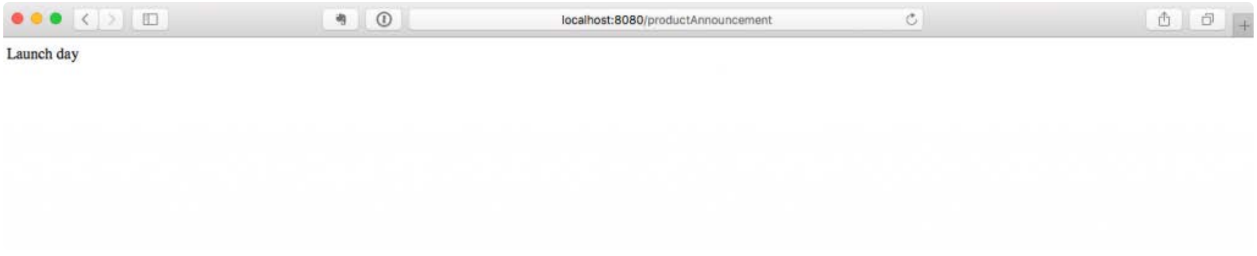
This is the controller content:

```
package myapp

class ProductAnnouncementController {

    def index() {

        def announcements = ProductAnnouncement.createCriteria().list {
            order("dateCreated", "desc")
            maxResults(1)
        }
        render announcements.first()?.message
    }
}
```

And if you run the app:

```
$ grails run-app
```

You can go to *http://localhost:8080/productAnnouncement*



Launch day

Now lets secure the app. Add the spring-security-core plugin to the *build.gradle* file.

```
buildscript {
    ext {
        grailsVersion = project.grailsVersion
    }
    repositories {
        mavenLocal()
        maven { url "https://repo.grails.org/grails/core" }
    }
    dependencies {
        classpath "org.grails:grails-gradle-plugin:$grailsVersion"
        classpath "com.bertramlabs.plugins:asset-pipeline-gradle:2.5.0"
        classpath "org.grails.plugins:hibernate4:5.0.0"
    }
}

version "0.1"
group "myapp"

apply plugin:"eclipse"
apply plugin:"idea"
apply plugin:"war"
apply plugin:"org.grails.grails-web"
apply plugin:"org.grails.grails-gsp"
apply plugin:"asset-pipeline"

ext {
    grailsVersion = project.grailsVersion
    gradleWrapperVersion = project.gradleWrapperVersion
}

repositories {
    mavenLocal()
    maven { url "https://repo.grails.org/grails/core" }
}

dependencyManagement {
    imports {
        mavenBom "org.grails:grails-bom:$grailsVersion"
    }
    applyMavenExclusions false
}

dependencies {
    compile "org.springframework.boot:spring-boot-starter-logging"
    compile "org.springframework.boot:spring-boot-autoconfigure"
    compile "org.grails:grails-core"
    compile "org.springframework.boot:spring-boot-starter-actuator"
    compile "org.springframework.boot:spring-boot-starter-tomcat"
    compile "org.grails:grails-dependencies"
    compile "org.grails:grails-web-boot"
    compile "org.grails.plugins:cache"
    compile "org.grails.plugins:scaffolding"
    compile "org.grails.plugins:hibernate4"
    compile "org.hibernate:hibernate-ehcache"
    console "org.grails:grails-console"
    profile "org.grails.profiles:web:3.1.1"
    runtime "org.grails.plugins:asset-pipeline"
    runtime "com.h2database:h2"
    testCompile "org.grails:grails-plugin-testing"
    testCompile "org.grails.plugins:geb"
    testRuntime "org.seleniumhq.selenium:selenium-htmlunit-driver:2.47.1"
    testRuntime "net.sourceforge.htmlunit:htmlunit:2.18"
    compile 'org.grails.plugins:spring-security-core:3.0.3'
}
```

```

task wrapper(type: Wrapper) {
    gradleVersion = gradleWrapperVersion
}

assets {
    minifyJs = true
    minifyCss = true
}
```

One of the things the plugin offers is several Grails commands. How do you access all the available Grails command for an app running *grails help*

```
grails help

Usage (optionals marked with *):'
grails [environment]* [target] [arguments]*'

| Examples:
$ grails dev run-app
$ grails create-app books

| Available Commands (type grails help 'command-name' for more info):
| Command Name                          Command Description
-------------------------------------------------------------------------------------------
---------
bug-report                              Creates a zip file that can be attached to issue
reports for the current project
clean                                   Cleans a Grails application's compiled sources
compile                                 Compiles a Grails application
console                                 Runs the Grails interactive console
create-controller                       Creates a controller
create-domain-class                     Creates a Domain Class
create-functional-test                  Creates a Geb Functional Test
create-integration-test                 Creates an integration test
create-interceptor                      Creates an interceptor
create-scaffold-controller              Creates a scaffolded controller
create-script                           Creates a Grails script
create-service                          Creates a Service
create-taglib                           Creates a Tag Library
create-unit-test                        Creates a unit test
dependency-report                       Prints out the Grails application's dependencies
generate-async-controller               Generates an asynchronous controller that performs
CRUD operations
gradle                                  Allows running of Gradle tasks
help                                    Prints help information for a specific command
idea-list-injected-traits               Prints which traits are injected to which artifacts
install                                 Installs a Grails application or plugin into the
local Maven cache
install-templates                       Installs scaffolding templates that use f:all to
render properties
list-plugins                            Lists available plugins from the Plugin Repository
open                                    Opens a file in the project
package                                 Packages a Grails application
plugin-info                             Prints information about the given plugin
run-app                                 Runs a Grails application
s2-create-persistent-token              Creates a persistent token domain class for the
Spring Security Core plugin
s2-create-role-hierarchy-entry          Creates a domain class for a persistent role
hierarchy for the Spring Security Core plugin
s2-quickstart                           Creates domain classes and updates config settings
for the Spring Security plugin
```

```
schema-export                           Creates a DDL file of the database schema
shell                                   Runs the Grails interactive shell
stats                                   Prints statistics about the project
stop-app                                Stops the running Grails application
test-app                                Runs the applications tests
url-mappings-report                     Prints out a report of the project's URL mappings
war                                     Creates a WAR file for deployment to a container
(like Tomcat)

| Detailed usage with help [command]
```

Lets run the s2-quickstart command:

```
grails s2-quickstart
| Error Usage:
   grails s2-quickstart <domain-class-package> <user-class-name> <role-class-name>
[requestmap-class-name] [--groupClassName=group-class-name]
or grails s2-quickstart --uiOnly

Example: grails s2-quickstart com.yourapp User Role
Example: grails s2-quickstart com.yourapp User Role --groupClassName=RoleGroup
Example: grails s2-quickstart com.yourapp Person Authority Requestmap
Example: grails s2-quickstart --uiOnly
```

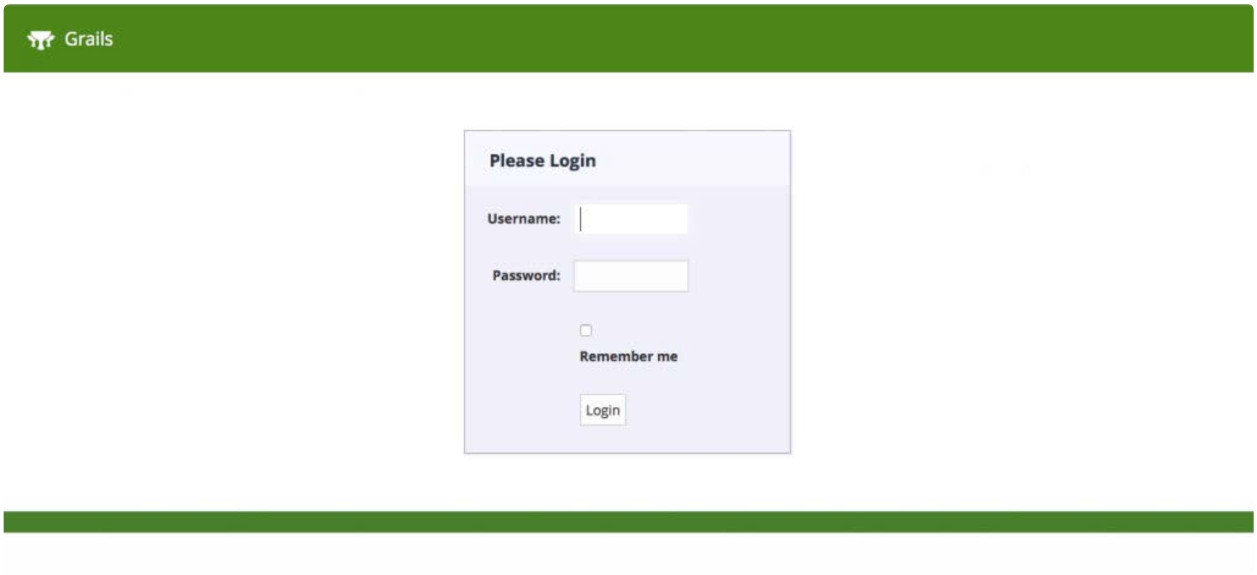
Lets create the security-related domain classes:

```
grails s2-quickstart myapp User Role
| Creating User class 'User' and Role class 'Role' in package 'myapp'
| Rendered template Person.groovy.template to destination grails-
app/domain/myapp/User.groovy
| Rendered template Authority.groovy.template to destination grails-
app/domain/myapp/Role.groovy
| Rendered template PersonAuthority.groovy.template to destination grails-
app/domain/myapp/UserRole.groovy
|
************************************************************
* Created security-related domain classes. Your           *
* grails-app/conf/application.groovy has been updated with *
* the class names of the configured domain classes;       *
* please verify that the values are correct.              *
************************************************************
```

Run the app again and try to go now to *http://localhost:8080/productAnnouncement*

http://localhost:8080/productAnnouncement

Now everything is secured by default. Because of that we are forwarded to the default Spring Security Core login page:

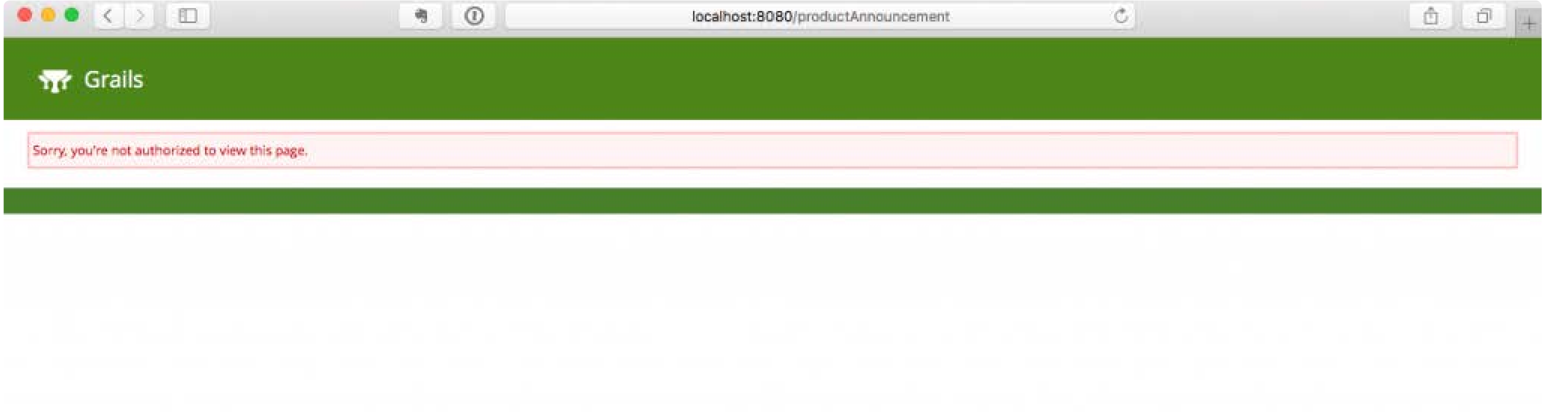


We cannot login since we do not have a user yet. Lets create it:

Edit *grails-app/init/BootStrap.groovy*

```
import myapp.*

class BootStrap {

    def springSecurityService

    def init = { servletContext ->
    new ProductAnnouncement(message: 'Launch day').save()

        def userRole = new Role('ROLE_USER').save()

        def me = new User('me@sergiodelamo.com', 'groovycalamari').save()

        UserRole.create me, userRole

        UserRole.withSession {
            it.flush()
            it.clear()
        }

    }

    def destroy = {
    }

}
```

If I try to go *http://localhost:8080/productAnnouncement* I am prompted to log in. I can now log in but after a successfull login I get this page:



Sorry, you're not authorized to view this page.

We need to annotate the method as illustrated below:

```
package myapp

import grails.plugin.springsecurity.annotation.Secured

class ProductAnnouncementController {

    @Secured('ROLE_USER')
    def index() {

        def announcements = ProductAnnouncement.createCriteria().list {
            order("dateCreated", "desc")
            maxResults(1)
        }
        render announcements.first()?.message
    }
}
```

Now, after login, you will be able to access that endpoint.

← How to log SQL statements in a Grails 3 app

Grails Programmer : How to secure your Grails 3 API with Spring Security REST for Grails? →

11 thoughts on "How to use Spring Security Core to Secure you Grails 3 App?"

sayyedbagher
2 March, 2016 at 14:56

just tnx
so usefull

Reply ↓

Yorick
17 October, 2016 at 6:01

really easy to understand and helps me a lot. thank you.

Reply ↓

Puwanat Sretavevuth
26 November, 2016 at 16:02

Thanks you

Reply ↓

Pete
14 January, 2017 at 7:16

in BootStrap.groovy I had to use
new Role(authority: 'ROLE_USER').save instead of new Role('ROLE_USER').save()
▯ Grails Version: 3.2.4
▯ Groovy Version: 2.4.7
▯ JVM Version: 1.8.0_101

Anyway it was very useful, thank you

Reply ↓

---

sdelamo Post author
17 January, 2017 at 18:22

Weird Pete, I believe the s2-quickstart autogenerated Authority class has a constructor with a single string parameter

Reply ↓

---

daniel

7 February, 2017 at 0:43

really helpful. thanks

Reply ↓

Ravi Ailani

17 February, 2017 at 16:51

how do i remove default security from pages?

Reply ↓

Rajat Malhotra

9 May, 2017 at 6:46

Really helpful. Thanks..

Reply ↓

David Young

27 July, 2017 at 18:01

Wow. I greatly appreciate how wonderfully step-wise, methodical and comprehensive your how-to is.
Thank you so much for taking the time to build this document.
Wish you could do "the rest of the programmable Internet" but I'm happy with this and look forward to seeing your other efforts.
Regards,
David

Reply ↓

---

njayaaa MBODJI

19 September, 2017 at 11:28

intéressant, j'utilise la version 3.3.0 de grails et au lieu de
def userRole = new Role('ROLE_USER').save()
Je fais
def userRole = new Role(authority:'ROLE_USER').save()
pour que cela marche (au cas où quelqu'un rencontre le même problème)

Merci encore à l'auteur, ça aide

Reply ↓

sdelamo Post author

19 September, 2017 at 16:50

yes, those constructors were removed in the latest versions of the plugin

Reply ↓

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

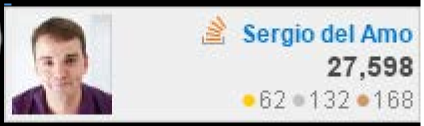
Website

Post Comment

□ Notify me of follow-up comments by email.

□ Notify me of new posts by email.

Search



Sergio del Amo

27,598

●62 ●132 ●168

Drag to outliner or Upload
Close