4.10.1

* Docs  
  [User Manual](http://docs.google.com/userguide/userguide.html)  
  [Guides and Tutorials](https://guides.gradle.org)  
  [DSL Reference](http://docs.google.com/dsl/)  
  [Javadoc](http://docs.google.com/javadoc/)  
  [Release Notes](http://docs.google.com/release-notes.html)
* [Forums](https://discuss.gradle.org/)
* [Training](https://gradle.org/training/)
* [Try Gradle Enterprise](https://gradle.com/enterprise)
* [User Manual Home](http://docs.google.com/userguide/userguide.html)
* [DSL Reference Home](http://docs.google.com/dsl/)
* [Release Notes](http://docs.google.com/release-notes.html)
  + [Properties](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#N121D6)
  + [Methods](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#N121FD)

### Build script blocks

* [allprojects { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:allprojects(groovy.lang.Closure))
* [artifacts { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:artifacts(groovy.lang.Closure))
* [buildscript { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:buildscript(groovy.lang.Closure))
* [configurations { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:configurations(groovy.lang.Closure))
* [dependencies { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:dependencies(groovy.lang.Closure))
* [repositories { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:repositories(groovy.lang.Closure))
* [sourceSets { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:sourceSets(groovy.lang.Closure))
* [subprojects { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:subprojects(groovy.lang.Closure))
* [publishing { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:publishing(groovy.lang.Closure))

### Core types

* [Project](http://docs.google.com/org.gradle.api.Project.html)
* [Task](http://docs.google.com/org.gradle.api.Task.html)
* [Gradle](http://docs.google.com/org.gradle.api.invocation.Gradle.html)
* [Settings](http://docs.google.com/org.gradle.api.initialization.Settings.html)
* [IncludedBuild](http://docs.google.com/org.gradle.api.initialization.IncludedBuild.html)
* [Script](http://docs.google.com/org.gradle.api.Script.html)
* [JavaToolChain](http://docs.google.com/org.gradle.jvm.toolchain.JavaToolChain.html)
* [SourceSet](http://docs.google.com/org.gradle.api.tasks.SourceSet.html)
* [SourceSetOutput](http://docs.google.com/org.gradle.api.tasks.SourceSetOutput.html)
* [SourceDirectorySet](http://docs.google.com/org.gradle.api.file.SourceDirectorySet.html)
* [IncrementalTaskInputs](http://docs.google.com/org.gradle.api.tasks.incremental.IncrementalTaskInputs.html)
* [Configuration](http://docs.google.com/org.gradle.api.artifacts.Configuration.html)
* [ResolutionStrategy](http://docs.google.com/org.gradle.api.artifacts.ResolutionStrategy.html)
* [ArtifactResolutionQuery](http://docs.google.com/org.gradle.api.artifacts.query.ArtifactResolutionQuery.html)
* [ComponentSelection](http://docs.google.com/org.gradle.api.artifacts.ComponentSelection.html)
* [ComponentSelectionRules](http://docs.google.com/org.gradle.api.artifacts.ComponentSelectionRules.html)
* [ExtensionAware](http://docs.google.com/org.gradle.api.plugins.ExtensionAware.html)
* [ExtraPropertiesExtension](http://docs.google.com/org.gradle.api.plugins.ExtraPropertiesExtension.html)
* [PluginDependenciesSpec](http://docs.google.com/org.gradle.plugin.use.PluginDependenciesSpec.html)
* [PluginDependencySpec](http://docs.google.com/org.gradle.plugin.use.PluginDependencySpec.html)
* [PluginManagementSpec](http://docs.google.com/org.gradle.plugin.management.PluginManagementSpec.html)
* [ResourceHandler](http://docs.google.com/org.gradle.api.resources.ResourceHandler.html)
* [TextResourceFactory](http://docs.google.com/org.gradle.api.resources.TextResourceFactory.html)

### Publishing types

* [PublishingExtension](http://docs.google.com/org.gradle.api.publish.PublishingExtension.html)
* [IvyPublication](http://docs.google.com/org.gradle.api.publish.ivy.IvyPublication.html)
* [IvyArtifact](http://docs.google.com/org.gradle.api.publish.ivy.IvyArtifact.html)
* [IvyArtifactSet](http://docs.google.com/org.gradle.api.publish.ivy.IvyArtifactSet.html)
* [IvyModuleDescriptorSpec](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorSpec.html)
* [IvyModuleDescriptorAuthor](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorAuthor.html)
* [IvyModuleDescriptorLicense](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorLicense.html)
* [IvyModuleDescriptorDescription](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorDescription.html)
* [MavenPublication](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html)
* [MavenArtifact](http://docs.google.com/org.gradle.api.publish.maven.MavenArtifact.html)
* [MavenArtifactSet](http://docs.google.com/org.gradle.api.publish.maven.MavenArtifactSet.html)
* [MavenPom](http://docs.google.com/org.gradle.api.publish.maven.MavenPom.html)
* [MavenPomCiManagement](http://docs.google.com/org.gradle.api.publish.maven.MavenPomCiManagement.html)
* [MavenPomContributor](http://docs.google.com/org.gradle.api.publish.maven.MavenPomContributor.html)
* [MavenPomContributorSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomContributorSpec.html)
* [MavenPomDeveloper](http://docs.google.com/org.gradle.api.publish.maven.MavenPomDeveloper.html)
* [MavenPomDeveloperSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomDeveloperSpec.html)
* [MavenPomDistributionManagement](http://docs.google.com/org.gradle.api.publish.maven.MavenPomDistributionManagement.html)
* [MavenPomIssueManagement](http://docs.google.com/org.gradle.api.publish.maven.MavenPomIssueManagement.html)
* [MavenPomLicense](http://docs.google.com/org.gradle.api.publish.maven.MavenPomLicense.html)
* [MavenPomLicenseSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomLicenseSpec.html)
* [MavenPomMailingList](http://docs.google.com/org.gradle.api.publish.maven.MavenPomMailingList.html)
* [MavenPomMailingListSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomMailingListSpec.html)
* [MavenPomOrganization](http://docs.google.com/org.gradle.api.publish.maven.MavenPomOrganization.html)
* [MavenPomRelocation](http://docs.google.com/org.gradle.api.publish.maven.MavenPomRelocation.html)
* [MavenPomScm](http://docs.google.com/org.gradle.api.publish.maven.MavenPomScm.html)

### Container types

* [TaskContainer](http://docs.google.com/org.gradle.api.tasks.TaskContainer.html)
* [ConfigurationContainer](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html)
* [RepositoryHandler](http://docs.google.com/org.gradle.api.artifacts.dsl.RepositoryHandler.html)
* [DependencyHandler](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html)
* [ArtifactHandler](http://docs.google.com/org.gradle.api.artifacts.dsl.ArtifactHandler.html)

### Build Cache types

* [BuildCacheConfiguration](http://docs.google.com/org.gradle.caching.configuration.BuildCacheConfiguration.html)
* [DirectoryBuildCache](http://docs.google.com/org.gradle.caching.local.DirectoryBuildCache.html)
* [HttpBuildCache](http://docs.google.com/org.gradle.caching.http.HttpBuildCache.html)

### Input Normalization types

* [InputNormalizationHandler](http://docs.google.com/org.gradle.normalization.InputNormalizationHandler.html)
* [InputNormalization](http://docs.google.com/org.gradle.normalization.InputNormalization.html)
* [RuntimeClasspathNormalization](http://docs.google.com/org.gradle.normalization.RuntimeClasspathNormalization.html)

### Help Task types

* [TaskReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.TaskReportTask.html)
* [ProjectReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.ProjectReportTask.html)
* [DependencyReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.DependencyReportTask.html)
* [DependencyInsightReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.DependencyInsightReportTask.html)
* [PropertyReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.PropertyReportTask.html)
* [ComponentReport](http://docs.google.com/org.gradle.api.reporting.components.ComponentReport.html)
* [DependentComponentsReport](http://docs.google.com/org.gradle.api.reporting.dependents.DependentComponentsReport.html)
* [ModelReport](http://docs.google.com/org.gradle.api.reporting.model.ModelReport.html)

### Task types

* [AntlrTask](http://docs.google.com/org.gradle.api.plugins.antlr.AntlrTask.html)
* [BuildEnvironmentReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.BuildEnvironmentReportTask.html)
* [Checkstyle](http://docs.google.com/org.gradle.api.plugins.quality.Checkstyle.html)
* [CodeNarc](http://docs.google.com/org.gradle.api.plugins.quality.CodeNarc.html)
* [CompareGradleBuilds](http://docs.google.com/org.gradle.api.plugins.buildcomparison.gradle.CompareGradleBuilds.html)
* [Copy](http://docs.google.com/org.gradle.api.tasks.Copy.html)
* [CreateStartScripts](http://docs.google.com/org.gradle.jvm.application.tasks.CreateStartScripts.html)
* [Delete](http://docs.google.com/org.gradle.api.tasks.Delete.html)
* [Ear](http://docs.google.com/org.gradle.plugins.ear.Ear.html)
* [Exec](http://docs.google.com/org.gradle.api.tasks.Exec.html)
* [FindBugs](http://docs.google.com/org.gradle.api.plugins.quality.FindBugs.html)
* [GenerateIvyDescriptor](http://docs.google.com/org.gradle.api.publish.ivy.tasks.GenerateIvyDescriptor.html)
* [GenerateMavenPom](http://docs.google.com/org.gradle.api.publish.maven.tasks.GenerateMavenPom.html)
* [GenerateBuildDashboard](http://docs.google.com/org.gradle.api.reporting.GenerateBuildDashboard.html)
* [GradleBuild](http://docs.google.com/org.gradle.api.tasks.GradleBuild.html)
* [GroovyCompile](http://docs.google.com/org.gradle.api.tasks.compile.GroovyCompile.html)
* [Groovydoc](http://docs.google.com/org.gradle.api.tasks.javadoc.Groovydoc.html)
* [HtmlDependencyReportTask](http://docs.google.com/org.gradle.api.reporting.dependencies.HtmlDependencyReportTask.html)
* [JacocoReport](http://docs.google.com/org.gradle.testing.jacoco.tasks.JacocoReport.html)
* [JacocoMerge](http://docs.google.com/org.gradle.testing.jacoco.tasks.JacocoMerge.html)
* [JacocoCoverageVerification](http://docs.google.com/org.gradle.testing.jacoco.tasks.JacocoCoverageVerification.html)
* [Jar](http://docs.google.com/org.gradle.api.tasks.bundling.Jar.html)
* [JavaCompile](http://docs.google.com/org.gradle.api.tasks.compile.JavaCompile.html)
* [Javadoc](http://docs.google.com/org.gradle.api.tasks.javadoc.Javadoc.html)
* [JavaExec](http://docs.google.com/org.gradle.api.tasks.JavaExec.html)
* [JDepend](http://docs.google.com/org.gradle.api.plugins.quality.JDepend.html)
* [Pmd](http://docs.google.com/org.gradle.api.plugins.quality.Pmd.html)
* [PublishToIvyRepository](http://docs.google.com/org.gradle.api.publish.ivy.tasks.PublishToIvyRepository.html)
* [PublishToMavenRepository](http://docs.google.com/org.gradle.api.publish.maven.tasks.PublishToMavenRepository.html)
* [ScalaCompile](http://docs.google.com/org.gradle.api.tasks.scala.ScalaCompile.html)
* [ScalaDoc](http://docs.google.com/org.gradle.api.tasks.scala.ScalaDoc.html)
* [InitBuild](http://docs.google.com/org.gradle.buildinit.tasks.InitBuild.html)
* [Sign](http://docs.google.com/org.gradle.plugins.signing.Sign.html)
* [Sync](http://docs.google.com/org.gradle.api.tasks.Sync.html)
* [Tar](http://docs.google.com/org.gradle.api.tasks.bundling.Tar.html)
* [AbstractTestTask](http://docs.google.com/org.gradle.api.tasks.testing.AbstractTestTask.html)
* [Test](http://docs.google.com/org.gradle.api.tasks.testing.Test.html)
* [TestReport](http://docs.google.com/org.gradle.api.tasks.testing.TestReport.html)
* [Upload](http://docs.google.com/org.gradle.api.tasks.Upload.html)
* [War](http://docs.google.com/org.gradle.api.tasks.bundling.War.html)
* [Wrapper](http://docs.google.com/org.gradle.api.tasks.wrapper.Wrapper.html)
* [WriteProperties](http://docs.google.com/org.gradle.api.tasks.WriteProperties.html)
* [Zip](http://docs.google.com/org.gradle.api.tasks.bundling.Zip.html)

### Reporting types

* [CustomizableHtmlReport](http://docs.google.com/org.gradle.api.reporting.CustomizableHtmlReport.html)
* [SingleFileReport](http://docs.google.com/org.gradle.api.reporting.SingleFileReport.html)
* [DirectoryReport](http://docs.google.com/org.gradle.api.reporting.DirectoryReport.html)
* [FindBugsXmlReport](http://docs.google.com/org.gradle.api.plugins.quality.FindBugsXmlReport.html)
* [Report](http://docs.google.com/org.gradle.api.reporting.Report.html)
* [Reporting](http://docs.google.com/org.gradle.api.reporting.Reporting.html)
* [ReportContainer](http://docs.google.com/org.gradle.api.reporting.ReportContainer.html)
* [ReportingExtension](http://docs.google.com/org.gradle.api.reporting.ReportingExtension.html)

### Eclipse/IDEA model types

* [EclipseModel](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseModel.html)
* [EclipseProject](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)
* [EclipseClasspath](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseClasspath.html)
* [EclipseJdt](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseJdt.html)
* [EclipseWtp](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseWtp.html)
* [EclipseWtpComponent](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseWtpComponent.html)
* [EclipseWtpFacet](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseWtpFacet.html)
* [IdeaModel](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaModel.html)
* [IdeaProject](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaProject.html)
* [IdeaModule](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaModule.html)
* [IdeaWorkspace](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaWorkspace.html)
* [XmlFileContentMerger](http://docs.google.com/org.gradle.plugins.ide.api.XmlFileContentMerger.html)
* [FileContentMerger](http://docs.google.com/org.gradle.plugins.ide.api.FileContentMerger.html)

### Eclipse/IDEA task types

* [GenerateEclipseProject](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseProject.html)
* [GenerateEclipseClasspath](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseClasspath.html)
* [GenerateEclipseJdt](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseJdt.html)
* [GenerateEclipseWtpComponent](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseWtpComponent.html)
* [GenerateEclipseWtpFacet](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseWtpFacet.html)
* [GenerateIdeaModule](http://docs.google.com/org.gradle.plugins.ide.idea.GenerateIdeaModule.html)
* [GenerateIdeaProject](http://docs.google.com/org.gradle.plugins.ide.idea.GenerateIdeaProject.html)
* [GenerateIdeaWorkspace](http://docs.google.com/org.gradle.plugins.ide.idea.GenerateIdeaWorkspace.html)

### Native software types

* [PrebuiltLibrary](http://docs.google.com/org.gradle.nativeplatform.PrebuiltLibrary.html)
* [PrebuiltSharedLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.PrebuiltSharedLibraryBinary.html)
* [PrebuiltStaticLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.PrebuiltStaticLibraryBinary.html)
* [NativeComponentSpec](http://docs.google.com/org.gradle.nativeplatform.NativeComponentSpec.html)
* [NativeExecutableSpec](http://docs.google.com/org.gradle.nativeplatform.NativeExecutableSpec.html)
* [NativeLibrarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeLibrarySpec.html)
* [NativeTestSuiteSpec](http://docs.google.com/org.gradle.nativeplatform.test.NativeTestSuiteSpec.html)
* [CUnitTestSuiteSpec](http://docs.google.com/org.gradle.nativeplatform.test.cunit.CUnitTestSuiteSpec.html)
* [GoogleTestTestSuiteSpec](http://docs.google.com/org.gradle.nativeplatform.test.googletest.GoogleTestTestSuiteSpec.html)
* [NativeBinarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeBinarySpec.html)
* [NativeExecutableBinarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeExecutableBinarySpec.html)
* [NativeLibraryBinarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeLibraryBinarySpec.html)
* [SharedLibraryBinarySpec](http://docs.google.com/org.gradle.nativeplatform.SharedLibraryBinarySpec.html)
* [StaticLibraryBinarySpec](http://docs.google.com/org.gradle.nativeplatform.StaticLibraryBinarySpec.html)
* [NativeTestSuiteBinarySpec](http://docs.google.com/org.gradle.nativeplatform.test.NativeTestSuiteBinarySpec.html)
* [CUnitTestSuiteBinarySpec](http://docs.google.com/org.gradle.nativeplatform.test.cunit.CUnitTestSuiteBinarySpec.html)
* [GoogleTestTestSuiteBinarySpec](http://docs.google.com/org.gradle.nativeplatform.test.googletest.GoogleTestTestSuiteBinarySpec.html)
* [NativePlatform](http://docs.google.com/org.gradle.nativeplatform.platform.NativePlatform.html)
* [BuildType](http://docs.google.com/org.gradle.nativeplatform.BuildType.html)
* [Flavor](http://docs.google.com/org.gradle.nativeplatform.Flavor.html)
* [Gcc](http://docs.google.com/org.gradle.nativeplatform.toolchain.Gcc.html)
* [Clang](http://docs.google.com/org.gradle.nativeplatform.toolchain.Clang.html)
* [VisualCpp](http://docs.google.com/org.gradle.nativeplatform.toolchain.VisualCpp.html)
* [AssemblerSourceSet](http://docs.google.com/org.gradle.language.assembler.AssemblerSourceSet.html)
* [CSourceSet](http://docs.google.com/org.gradle.language.c.CSourceSet.html)
* [CppSourceSet](http://docs.google.com/org.gradle.language.cpp.CppSourceSet.html)
* [ObjectiveCSourceSet](http://docs.google.com/org.gradle.language.objectivec.ObjectiveCSourceSet.html)
* [ObjectiveCppSourceSet](http://docs.google.com/org.gradle.language.objectivecpp.ObjectiveCppSourceSet.html)
* [WindowsResourceSet](http://docs.google.com/org.gradle.language.rc.WindowsResourceSet.html)
* [VisualStudioProject](http://docs.google.com/org.gradle.ide.visualstudio.VisualStudioProject.html)
* [VisualStudioSolution](http://docs.google.com/org.gradle.ide.visualstudio.VisualStudioSolution.html)
* [NativeExecutable](http://docs.google.com/org.gradle.nativeplatform.NativeExecutable.html)
* [NativeLibrary](http://docs.google.com/org.gradle.nativeplatform.NativeLibrary.html)
* [NativeBinary](http://docs.google.com/org.gradle.nativeplatform.NativeBinary.html)
* [NativeExecutableBinary](http://docs.google.com/org.gradle.nativeplatform.NativeExecutableBinary.html)
* [SharedLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.SharedLibraryBinary.html)
* [StaticLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.StaticLibraryBinary.html)

### Native component task types

* [CppCompile](http://docs.google.com/org.gradle.language.cpp.tasks.CppCompile.html)
* [CCompile](http://docs.google.com/org.gradle.language.c.tasks.CCompile.html)
* [Assemble](http://docs.google.com/org.gradle.language.assembler.tasks.Assemble.html)
* [ObjectiveCCompile](http://docs.google.com/org.gradle.language.objectivec.tasks.ObjectiveCCompile.html)
* [ObjectiveCppCompile](http://docs.google.com/org.gradle.language.objectivecpp.tasks.ObjectiveCppCompile.html)
* [WindowsResourceCompile](http://docs.google.com/org.gradle.language.rc.tasks.WindowsResourceCompile.html)
* [LinkExecutable](http://docs.google.com/org.gradle.nativeplatform.tasks.LinkExecutable.html)
* [LinkSharedLibrary](http://docs.google.com/org.gradle.nativeplatform.tasks.LinkSharedLibrary.html)
* [CreateStaticLibrary](http://docs.google.com/org.gradle.nativeplatform.tasks.CreateStaticLibrary.html)
* [InstallExecutable](http://docs.google.com/org.gradle.nativeplatform.tasks.InstallExecutable.html)
* [RunTestExecutable](http://docs.google.com/org.gradle.nativeplatform.test.tasks.RunTestExecutable.html)

SigningExtension

**Table of Contents**

[Properties](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#N121D6)[Methods](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#N121FD)[Script blocks](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#N1227B)[Property details](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#N12280)[Method details](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#N122B3)

| API Documentation: | [SigningExtension](http://docs.google.com/javadoc/org/gradle/plugins/signing/SigningExtension.html) |
| --- | --- |

The global signing configuration for a project.

Properties

| Property | Description |
| --- | --- |
| [required](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:required) | Whether or not this task should fail if no signatory or signature type are configured at generation time. |
| [signatory](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:signatory) | The signatory that will be used for signing when an explicit signatory has not been specified. |
| [signatureType](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:signatureType) | The signature type that will be used for signing files when an explicit signature type has not been specified. |

Methods

| Method | Description |
| --- | --- |
| [sign](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:sign(groovy.lang.Closure))(closure) | Creates a new [SignOperation](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) using the given closure to configure it before executing it. |
| [sign](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:sign(java.io.File%5B%5D))(files) | Digitally signs the files, generating signature files alongside them. |
| [sign](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:sign(java.lang.String,%20java.io.File%5B%5D))(classifier, files) | Digitally signs the files, generating signature files alongside them. |
| [sign](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:sign(org.gradle.api.DomainObjectCollection))(publications) | Incubating  Creates signing tasks that sign all publishable artifacts of the given publication collection. |
| [sign](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:sign(org.gradle.api.Task%5B%5D))(tasks) | Creates signing tasks that depend on and sign the "archive" produced by the given tasks. |
| [sign](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:sign(org.gradle.api.artifacts.Configuration%5B%5D))(configurations) | Creates signing tasks that sign [Configuration.getAllArtifacts()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:allArtifacts) of the given configurations. |
| [sign](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:sign(org.gradle.api.artifacts.PublishArtifact%5B%5D))(publishArtifacts) | Digitally signs the publish artifacts, generating signature files alongside them. |
| [sign](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:sign(org.gradle.api.publish.Publication%5B%5D))(publications) | Incubating  Creates signing tasks that sign all publishable artifacts of the given publications. |
| [signPom](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:signPom(org.gradle.api.artifacts.maven.MavenDeployment))(mavenDeployment) | Signs the POM artifact for the given Maven deployment. |
| [signPom](http://docs.google.com/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:signPom(org.gradle.api.artifacts.maven.MavenDeployment,%20groovy.lang.Closure))(mavenDeployment, closure) | Signs the POM artifact for the given Maven deployment. |

Script blocks

No script blocks

Property details

boolean required

Whether or not this task should fail if no signatory or signature type are configured at generation time.

Defaults to true.

[**Signatory**](http://docs.google.com/javadoc/org/gradle/plugins/signing/signatory/Signatory.html) signatory (read-only)

The signatory that will be used for signing when an explicit signatory has not been specified.

Delegates to the signatory provider's default signatory.

[**SignatureType**](http://docs.google.com/javadoc/org/gradle/plugins/signing/type/SignatureType.html) signatureType (read-only)

The signature type that will be used for signing files when an explicit signature type has not been specified.

Delegates to the signature type provider's default type.

Method details

[**SignOperation**](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) sign([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) closure)

Creates a new [SignOperation](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) using the given closure to configure it before executing it.

The project's default signatory and default signature type from the [SigningExtension](http://docs.google.com/dsl/org.gradle.plugins.signing.SigningExtension.html) will be used to generate the signature. The returned [SignOperation](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) gives access to the created signature files.

If there is no configured default signatory available (and one is not explicitly specified in this operation's configuration), the sign operation will fail.

[**SignOperation**](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) sign([**File**](http://download.oracle.com/javase/8/docs/api/java/io/File.html)... files)

Digitally signs the files, generating signature files alongside them.

The project's default signatory and default signature type from the [SigningExtension](http://docs.google.com/dsl/org.gradle.plugins.signing.SigningExtension.html) will be used to generate the signature. The returned [SignOperation](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) gives access to the created signature files.

If there is no configured default signatory available, the sign operation will fail.

[**SignOperation**](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) sign([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) classifier, [**File**](http://download.oracle.com/javase/8/docs/api/java/io/File.html)... files)

Digitally signs the files, generating signature files alongside them.

The project's default signatory and default signature type from the [SigningExtension](http://docs.google.com/dsl/org.gradle.plugins.signing.SigningExtension.html) will be used to generate the signature. The returned [SignOperation](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) gives access to the created signature files.

If there is no configured default signatory available, the sign operation will fail.

[**List**](http://download.oracle.com/javase/8/docs/api/java/util/List.html)<[**Sign**](http://docs.google.com/dsl/org.gradle.plugins.signing.Sign.html)> sign([**DomainObjectCollection**](http://docs.google.com/javadoc/org/gradle/api/DomainObjectCollection.html)<[**Publication**](http://docs.google.com/javadoc/org/gradle/api/publish/Publication.html)> publications)

Note: This method is [incubating](http://docs.google.com/userguide/feature_lifecycle.html) and may change in a future version of Gradle.

Creates signing tasks that sign all publishable artifacts of the given publication collection.

The created tasks will be named "sign*<publication name capitalized>*Publication". That is, given a publication with the name "mavenJava" the created task will be named "signMavenJavaPublication". The signature artifacts for the created tasks are added to the publishable artifacts of the given publications.

[**List**](http://download.oracle.com/javase/8/docs/api/java/util/List.html)<[**Sign**](http://docs.google.com/dsl/org.gradle.plugins.signing.Sign.html)> sign([**Task**](http://docs.google.com/dsl/org.gradle.api.Task.html)... tasks)

Creates signing tasks that depend on and sign the "archive" produced by the given tasks.

The created tasks will be named "sign*<input task name capitalized>*". That is, given a task with the name "jar" the created task will be named "signJar".

If the task is not an [AbstractArchiveTask](http://docs.google.com/dsl/org.gradle.api.tasks.bundling.AbstractArchiveTask.html), an [InvalidUserDataException](http://docs.google.com/javadoc/org/gradle/api/InvalidUserDataException.html) will be thrown.

The signature artifact for the created task is added to the [SigningExtension.getConfiguration()](http://docs.google.com/javadoc/org/gradle/plugins/signing/SigningExtension.html#getConfiguration--).

[**List**](http://download.oracle.com/javase/8/docs/api/java/util/List.html)<[**Sign**](http://docs.google.com/dsl/org.gradle.plugins.signing.Sign.html)> sign([**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html)... configurations)

Creates signing tasks that sign [Configuration.getAllArtifacts()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:allArtifacts) of the given configurations.

The created tasks will be named "sign*<configuration name capitalized>*". That is, given a configuration with the name "archives" the created task will be named "signArchives". The signature artifacts for the created tasks are added to the [SigningExtension.getConfiguration()](http://docs.google.com/javadoc/org/gradle/plugins/signing/SigningExtension.html#getConfiguration--) for this settings object.

[**SignOperation**](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) sign([**PublishArtifact**](http://docs.google.com/javadoc/org/gradle/api/artifacts/PublishArtifact.html)... publishArtifacts)

Digitally signs the publish artifacts, generating signature files alongside them.

The project's default signatory and default signature type from the [SigningExtension](http://docs.google.com/dsl/org.gradle.plugins.signing.SigningExtension.html) will be used to generate the signature. The returned [SignOperation](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) gives access to the created signature files.

If there is no configured default signatory available, the sign operation will fail.

[**List**](http://download.oracle.com/javase/8/docs/api/java/util/List.html)<[**Sign**](http://docs.google.com/dsl/org.gradle.plugins.signing.Sign.html)> sign([**Publication**](http://docs.google.com/javadoc/org/gradle/api/publish/Publication.html)... publications)

Note: This method is [incubating](http://docs.google.com/userguide/feature_lifecycle.html) and may change in a future version of Gradle.

Creates signing tasks that sign all publishable artifacts of the given publications.

The created tasks will be named "sign*<publication name capitalized>*Publication". That is, given a publication with the name "mavenJava" the created task will be named "signMavenJavaPublication". The signature artifacts for the created tasks are added to the publishable artifacts of the given publications.

[**Signature**](http://docs.google.com/javadoc/org/gradle/plugins/signing/Signature.html) signPom([**MavenDeployment**](http://docs.google.com/javadoc/org/gradle/api/artifacts/maven/MavenDeployment.html) mavenDeployment)

Signs the POM artifact for the given Maven deployment.

You can use this method to sign the generated POM when publishing to a Maven repository with the Maven plugin.

uploadArchives {  
 repositories {  
 mavenDeployer {  
 beforeDeployment { MavenDeployment deployment ->  
 signing.signPom(deployment)  
 }  
 }  
 }  
}

You can optionally provide a configuration closure to fine tune the [SignOperation](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) for the POM.

If [SigningExtension.isRequired()](http://docs.google.com/dsl/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:required) is false and the signature cannot be generated (e.g. no configured signatory), this method will silently do nothing. That is, a signature for the POM file will not be uploaded.

*Note:* Signing the generated POM file generated by the Maven Publishing plugin is currently not supported. Future versions of Gradle might add this functionality.

[**Signature**](http://docs.google.com/javadoc/org/gradle/plugins/signing/Signature.html) signPom([**MavenDeployment**](http://docs.google.com/javadoc/org/gradle/api/artifacts/maven/MavenDeployment.html) mavenDeployment, [**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) closure)

Signs the POM artifact for the given Maven deployment.

You can use this method to sign the generated POM when publishing to a Maven repository with the Maven plugin.

uploadArchives {  
 repositories {  
 mavenDeployer {  
 beforeDeployment { MavenDeployment deployment ->  
 signing.signPom(deployment)  
 }  
 }  
 }  
}

You can optionally provide a configuration closure to fine tune the [SignOperation](http://docs.google.com/javadoc/org/gradle/plugins/signing/SignOperation.html) for the POM.

If [SigningExtension.isRequired()](http://docs.google.com/dsl/org.gradle.plugins.signing.SigningExtension.html#org.gradle.plugins.signing.SigningExtension:required) is false and the signature cannot be generated (e.g. no configured signatory), this method will silently do nothing. That is, a signature for the POM file will not be uploaded.

*Note:* Signing the generated POM file generated by the Maven Publishing plugin is currently not supported. Future versions of Gradle might add this functionality.

**Docs**

* [User Manual](http://docs.google.com/userguide/userguide.html)
* [DSL Reference](http://docs.google.com/dsl/)
* [Release Notes](http://docs.google.com/release-notes.html)
* [Javadoc](http://docs.google.com/javadoc/)

**News**

* [Blog](https://blog.gradle.org/)
* [Newsletter](https://newsletter.gradle.com/)
* [Twitter](https://twitter.com/gradle)

**Products**

* [Build Scans](https://gradle.com/build-scans)
* [Build Cache](https://gradle.com/build-cache)
* [Enterprise Docs](https://gradle.com/enterprise/resources)

**Get Help**

* [Forums](https://discuss.gradle.org/c/help-discuss)
* [GitHub](https://github.com/gradle/)
* [Training](https://gradle.org/training/)
* [Services](https://gradle.org/services/)

Subscribe for important Gradle updates and news

Subscribe

By entering your email, you agree to our [Terms](https://gradle.org/terms/) and [Privacy Policy](https://gradle.org/privacy/), including receipt of emails. You can unsubscribe at any time.

© [Gradle Inc.](https://gradle.com)2018 All rights reserved.

[Careers](https://gradle.com/careers) | [Privacy](https://gradle.org/privacy) | [Terms of Service](https://gradle.org/terms) | [Contact](https://gradle.org/contact/)