Manual para la generación del instalador Windows de AutoFirma

Índice de contenidos

[1 Introducción 3](#_Toc447039017)

[2 Requisitos mínimos 4](#_Toc447039018)

[3 Proceso de generación 5](#_Toc447039019)

[3.1 Generación del instalador .exe 5](#_Toc447039020)

[3.2 Generación del instalador .msi 9](#_Toc447039021)

# Introducción

En el proceso de instalación de una aplicación Windows es necesaria la existencia de un ejecutable que contenga una serie de instrucciones para ejecutar el programa y un conjunto de recursos como librerías, imágenes u otros ejecutables que lo hagan funcionar correctamente.

En este caso la aplicación AutoFirma cuenta con dos tipos de ficheros que permiten llevar a cabo esta instalación:

* **Fichero .exe**: Su función es la de ejecutar una serie de instrucciones en un sistema Windows. Este tipo de archivos se utilizan para instalar, arrancar programas y ejecutar aplicaciones de forma manual.
* **Fichero .msi:** Permite automatizar una instalación en Windows, reduciendo al mínimo la intervención manual, ya que toda la información necesaria se encuentra en el propio fichero.

Este documento tiene como objetivo la explicación en detalle de los pasos necesarios para la generación de ambos ficheros.

# Requisitos mínimos

Sistema operativo:

* Windows XP SP3, Vista SP2, 7 SP1, 8 o superior, en 32 (x86) o 64 (x64) bits.

Adicionalmente para la construcción de los instaladores se hace uso de las siguientes herramientas:

* Java SE Runtime Environment 8 (<http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>)
* Launch4j (<https://sourceforge.net/projects/launch4j/files/launch4j-3/3.8/>)
* WiX (<http://wixtoolset.org/releases/>)
* NSIS (<http://nsis.sourceforge.net/Download>)
* NSIS NsProcess Plugin (<http://nsis.sourceforge.net/NsProcess_plugin>) incluyendo los ficheros .nsh y .dll según la siguiente jerarquía:

**NSIS 2.x:**

NSIS/  
├── Includes/  
│ └── nsProcess.nsh  
└── Plugins/  
    └── nsProcess.dll

**NSIS 3.x:**

NSIS/  
├── Includes/  
│ └── nsProcess.nsh  
└── Plugins/  
    ├── x86-ansi/  
    │ └── nsProcess.dll  
    └── x86-unicode/  
        └── nsProcess.dll

# Proceso de generación

## Generación del instalador .exe

En primer lugar se compila el core del proyecto ejecutando en el directorio principal del proyecto el siguiente comando:

mvn clean install -Dmaven.test.skip=true

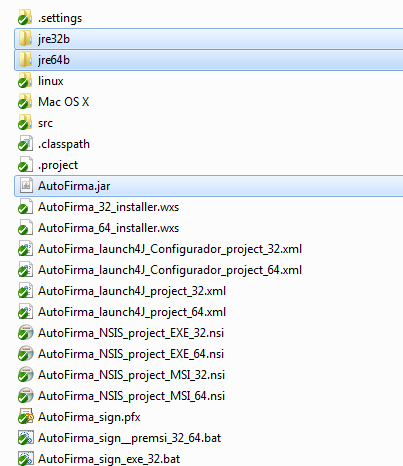
Una vez hecho esto se genera el .jar principal y el de configuración ejecutando el mismo comando en los siguientes proyectos:

* **afirma-simple:** Genera afirma-ui-simpleafirma-X.X.X-SNAPSHOT.jar
* **afirma-ui-simple-configurator:** Genera afirma-ui-simpleafirma-configurator-X.X.X-SNAPSHOT.jar

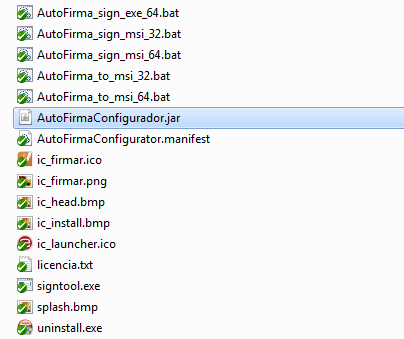
Entrando en la carpeta del proyecto **afirma-simple-installer** se encuentran el resto de recursos necesarios para construir el instalador. A ellos se deben añadir:

* jre32b: Directorio que contiene la jre de 32 bits de java
* jre64b: Directorio que contiene la jre de 64 bits de java

Adicionalmente se renombran los dos .jar generados a AutoFirma.jar y AutoFirmaConfigurador.jar respectivamente y son copiados en la carpeta.

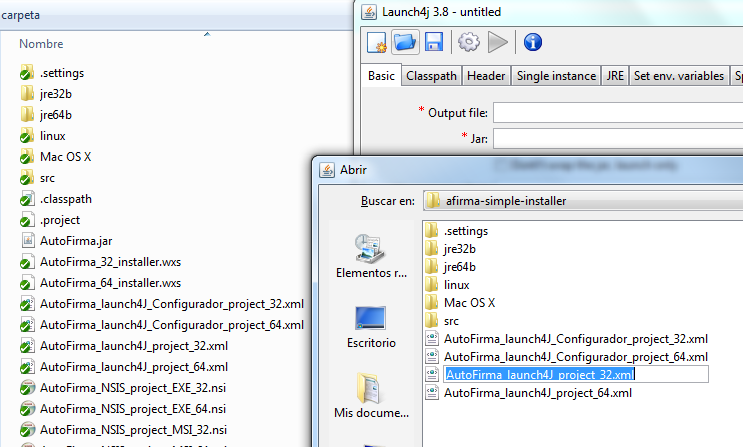


**Figura 1**

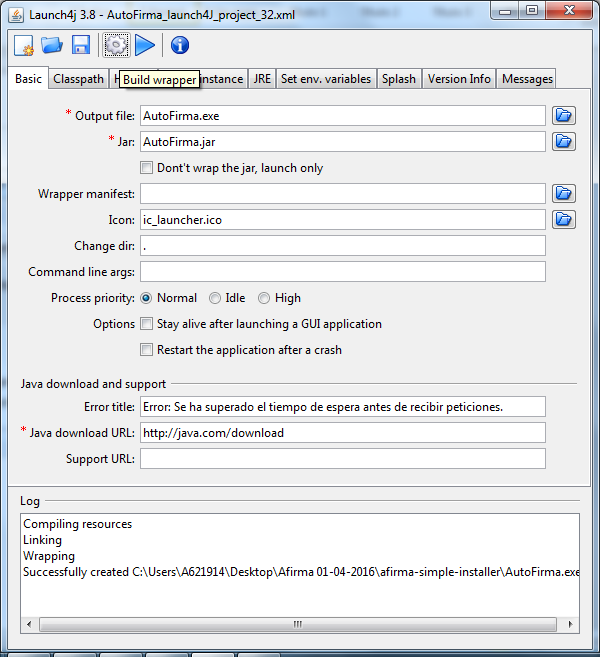


**Figura 2**

El siguiente paso hace uso de la aplicación Launch4j que convierte los .jar en .exe. Para ello se ejecuta la aplicación, se abre el AutoFirma\_launch4J\_project\_32.xml y se hace pincha en build wrapper para obtener el AutoFirma.exe (Figuras 3 y 4).

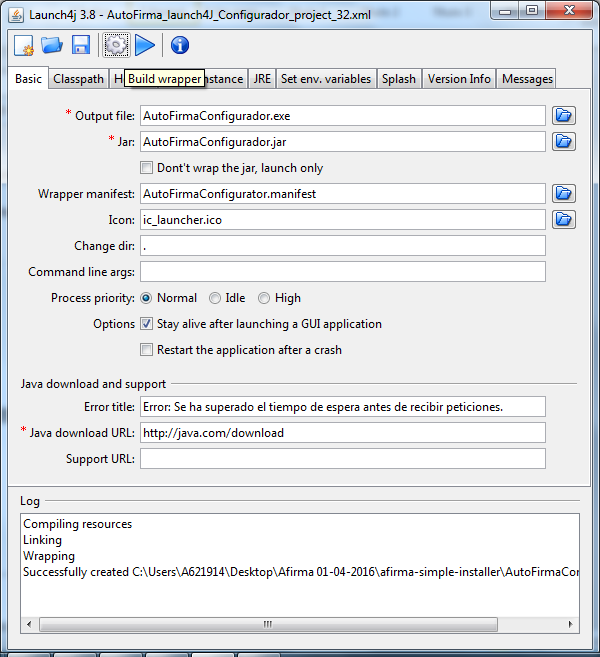
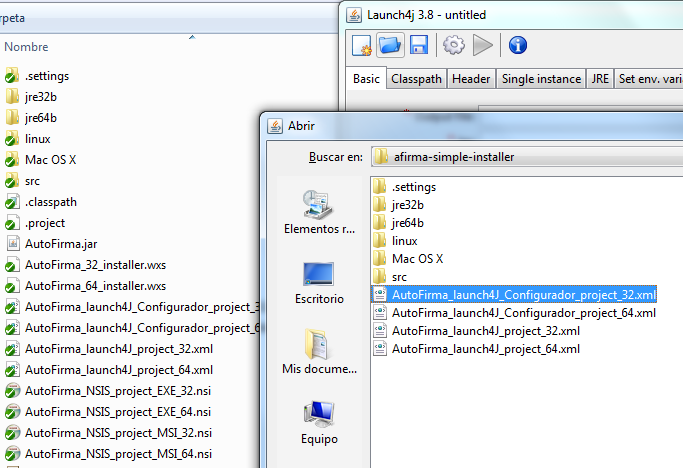


**Figura 3**



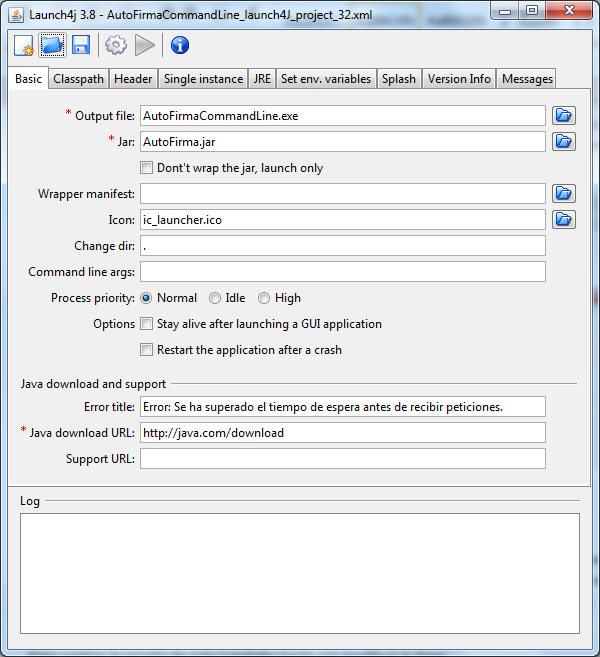
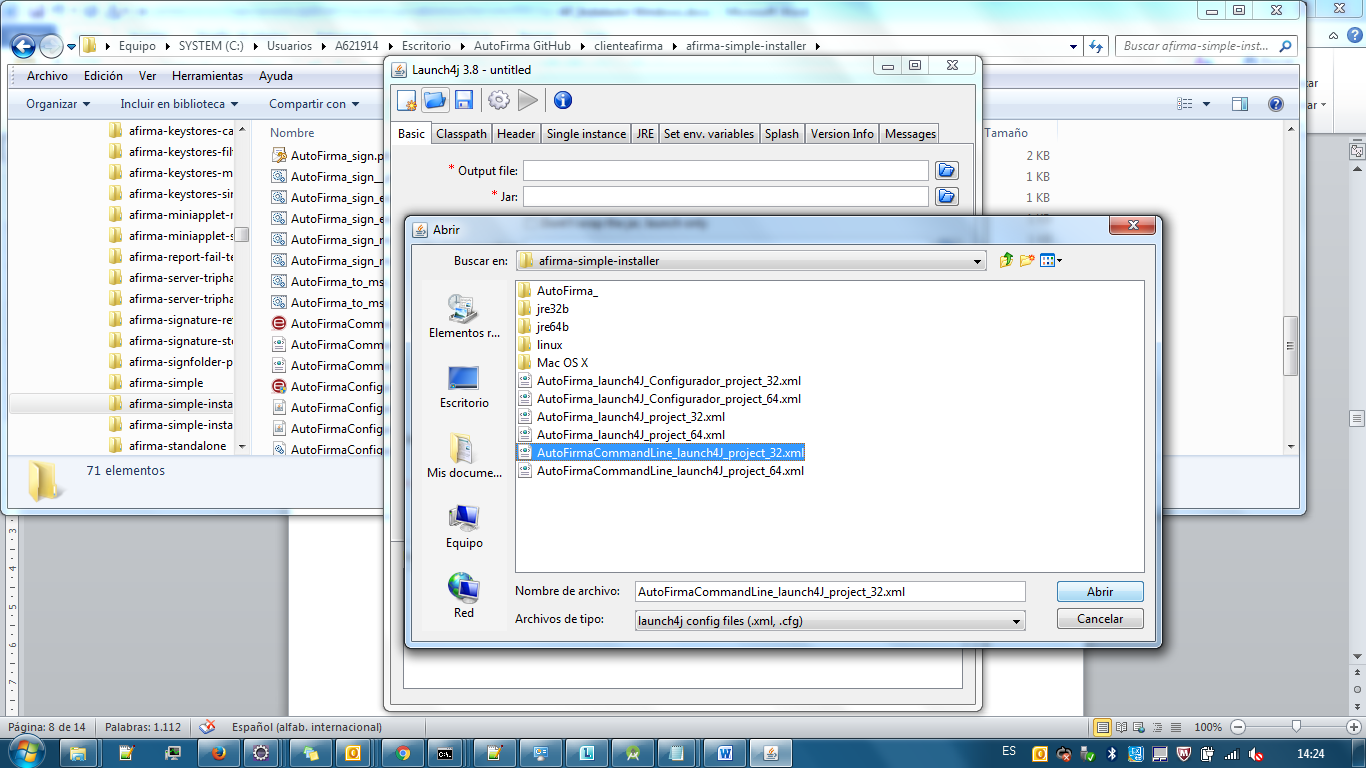
**Figura 4**

De la misma manera se transforma el AutoFirmaConfigurador.jar en AutoFirmaConfigurador.exe esta vez mediante el fichero AutoFirmaConfigurator\_launch4J\_project\_32.xml (Figura 5).



**Figura 5**

Por tercera vez se siguen los mismos pasos para generar el ejecutable mediante línea de comandos. La conversión de AutoFirma.jar en AutoFirmaCommandLine.exe se lleva a cabo utilizando el fichero AutoFirmaCommandLine\_launch4J\_project\_32.xml (Figura 6).



**Figura 6**

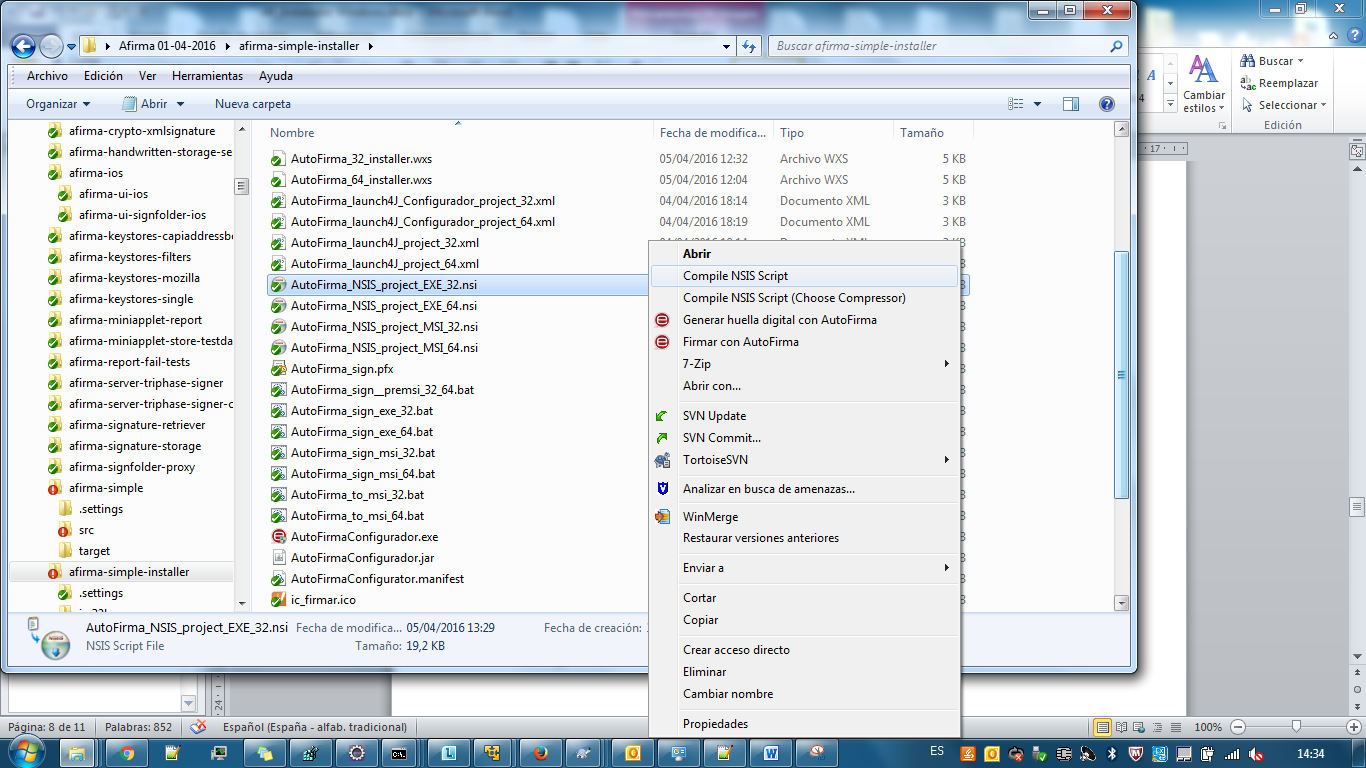
Para este último paso es necesario instalar la aplicación NSIS.

Con los dos .exe generados en el directorio y el resto de recursos ya presentes, se pincha con botón derecho en el fichero AutoFirma\_NSIS\_project\_EXE\_32 .nsi y se compila con NSIS para generar el instalador final AutoAfirma\_32\_installer.exe. Es necesario tener tanto NSIS como el NsProcess Plugin de NSIS en el sistema.

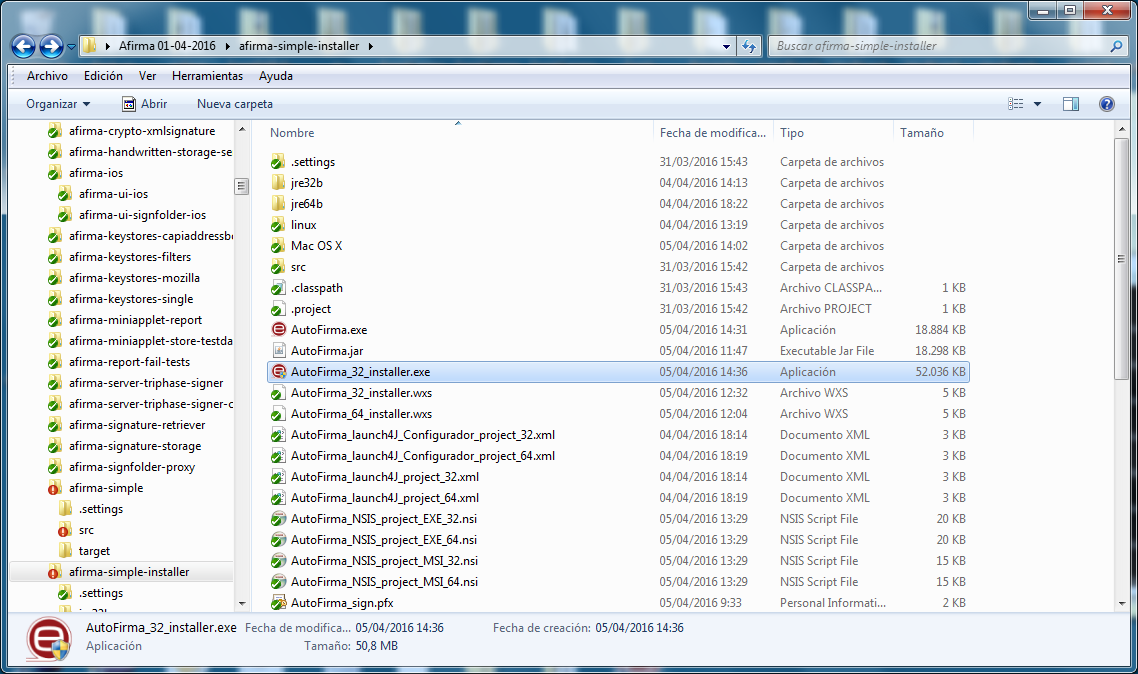
Para cambiar la versión de este instalador basta con modificar la línea

**!define VERSION "1.4.2"**

en el fichero AutoFirma\_NSIS\_project \_EXE\_32 .nsi

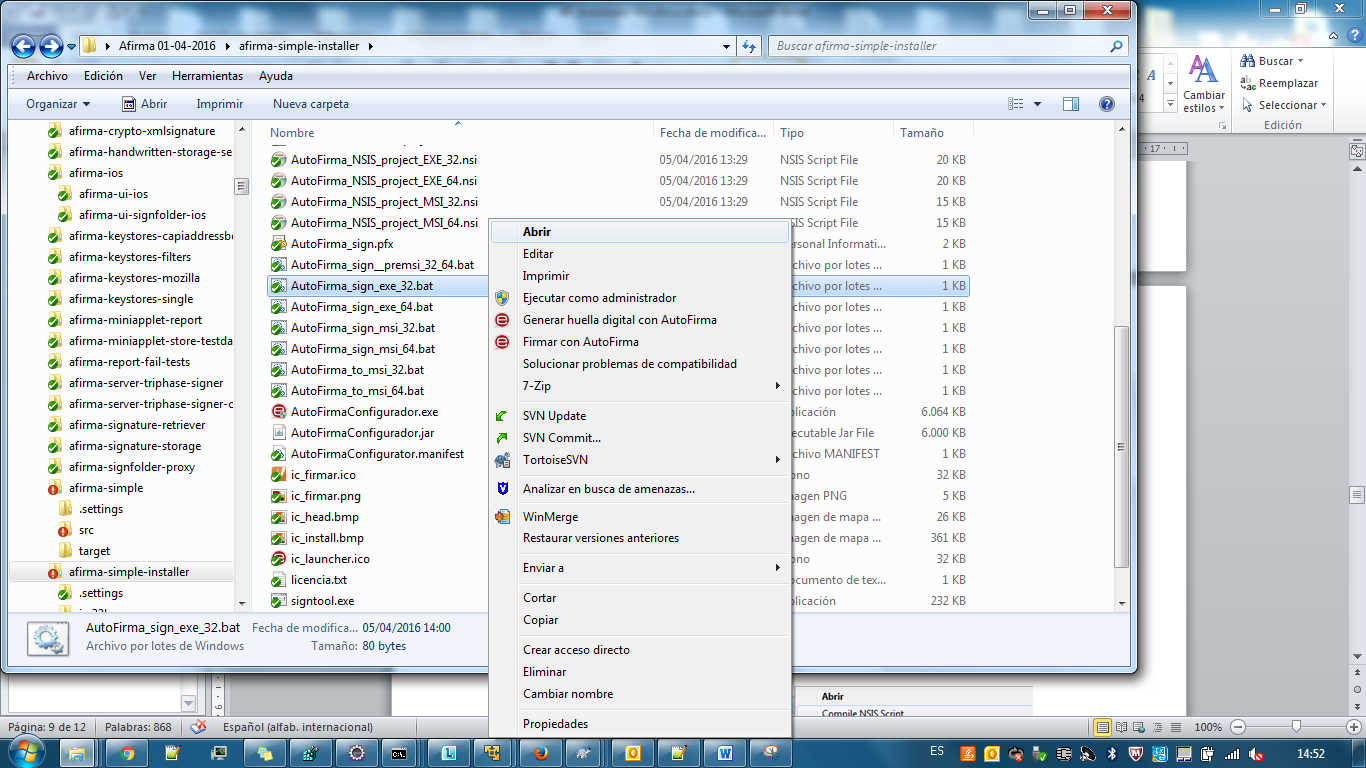


**Figura 7**



**Figura 8**

Por último la firma del .exe se realiza mediante la ejecución del ejecutable Afirma\_sign\_exe\_32.bat a partir del certificado AutoFirma\_sign.pfx.



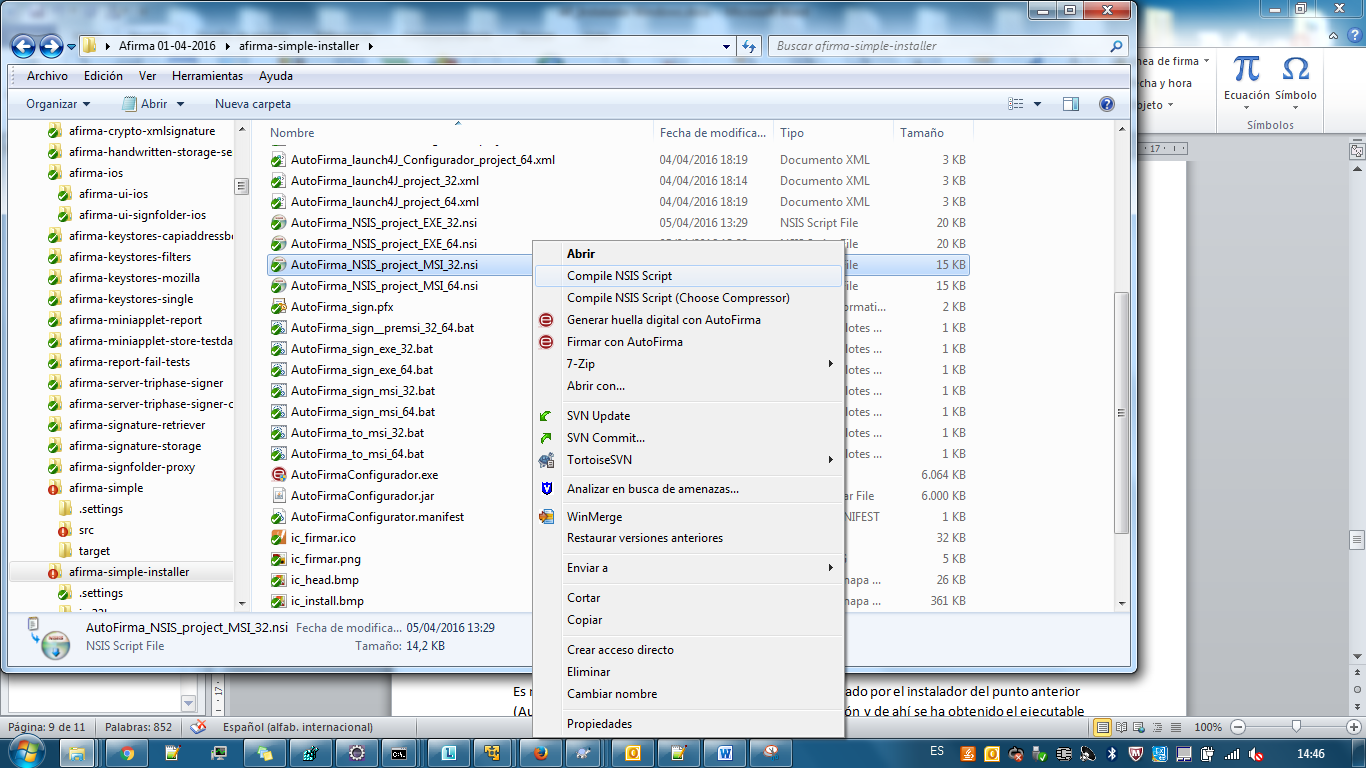
**Figura 9**

Este instalador generado contiene la versión de 32 bits de Java. Para generar uno de 64 bits que permita trabajar con ficheros de mayor tamaño es necesario que para cada fichero que contenga un 32 en el nombre se utilice su equivalente con un 64 en su lugar.

## Generación del instalador .msi

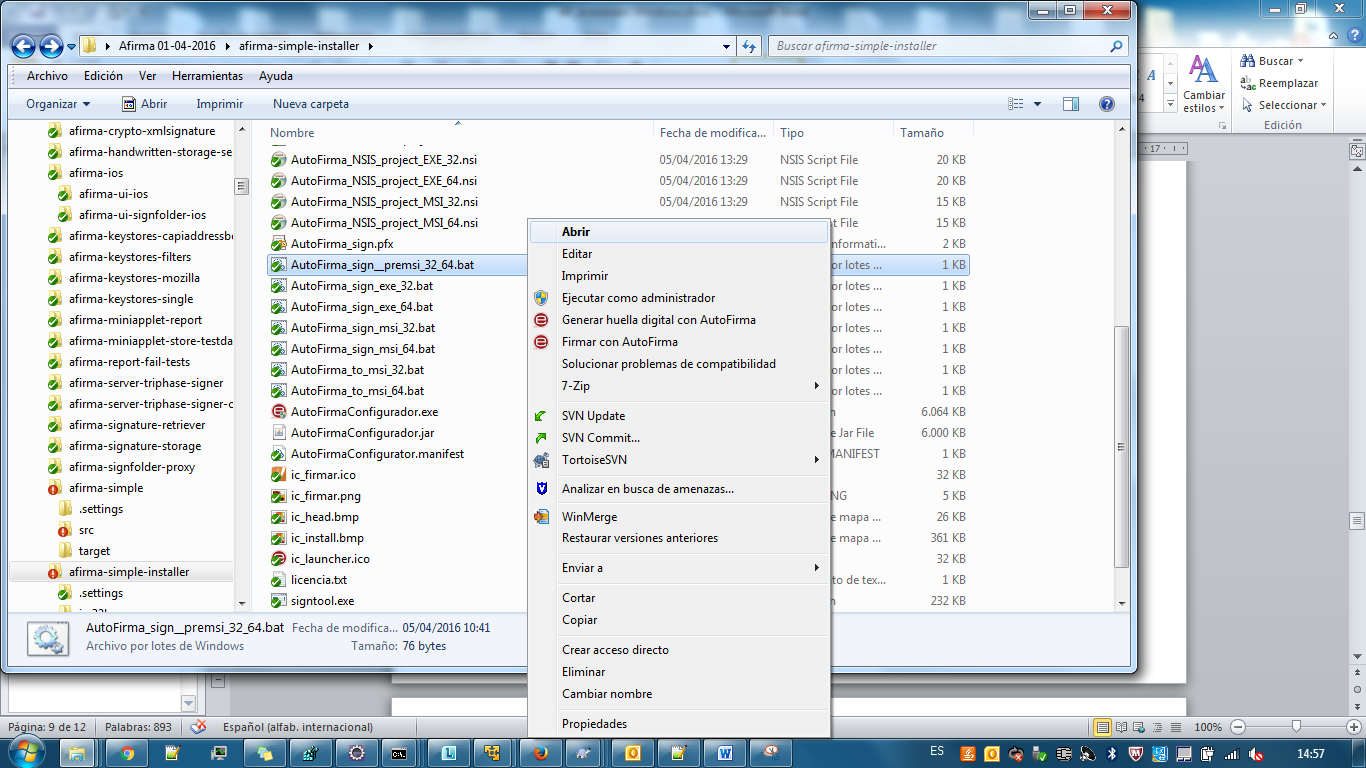
Para la generación del segundo instalador se hace uso de las herramientas de WiX que deben ser descargadas, instaladas y con la ruta C:\Program Files (x86)\WiX Toolset v3.11\bin añadida a la variable de entorno PATH.

En este caso, también con los ficheros AutoFirmaConfigurador.exe y AutoFirma.exe en el mismo directorio, se compilará el fichero AutoFirma\_NSIS\_project\_MSI\_32.nsi con NSIS para generar el “AutoFirmaGenerator.exe”.



**Figura 10**

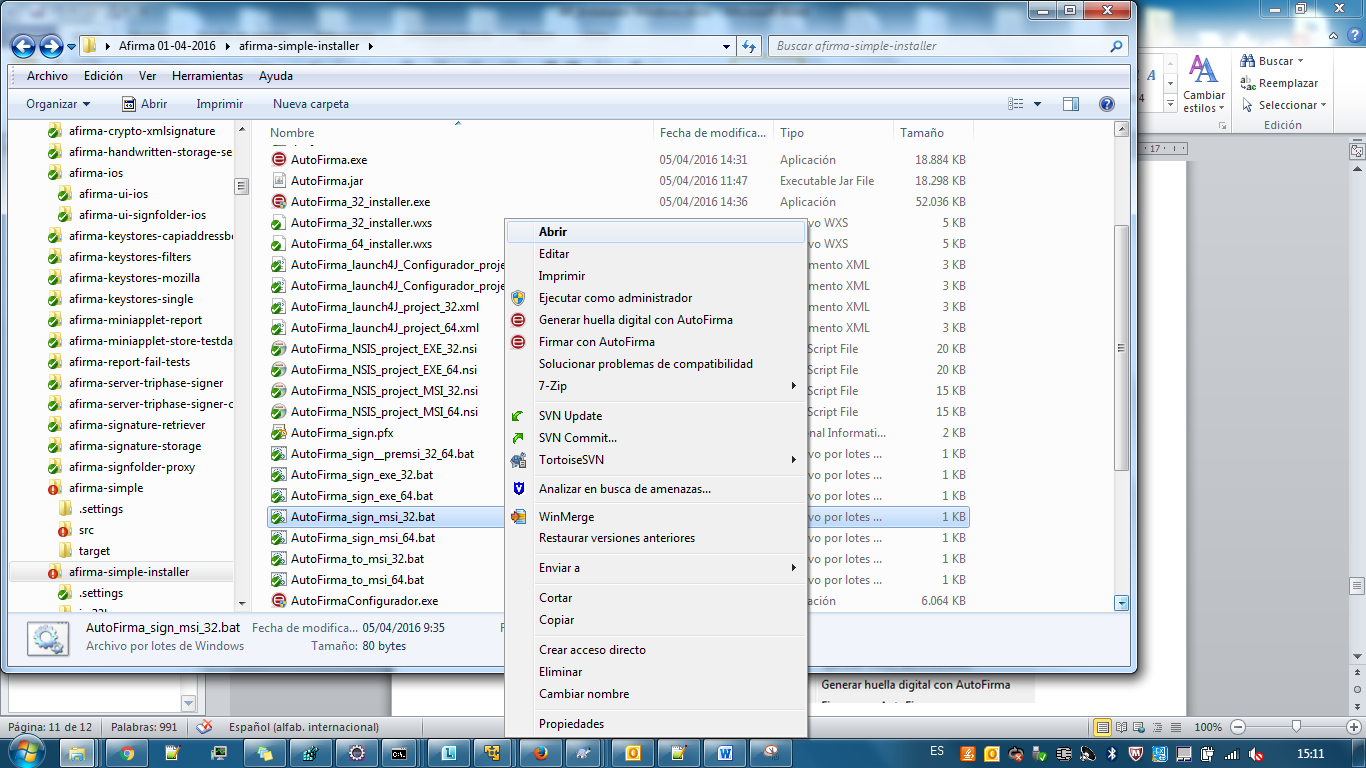
A diferencia del .exe generado en el punto anterior, este ejecutable no crea ni elimina información del registro, ni tampoco genera un ejecutable de desinstalación. Únicamente crea la estructura de directorios de la aplicación y genera e importa los certificados necesarios y debe ser firmado de la misma manera que en el caso anterior ejecutando el fichero AutoFirma\_sign\_\_premsi\_32\_64.bat (se ejecuta el mismo .bat para las versiones de 32 y 64 bits).



**Figura 11**

La funcionalidad relativa a registrar AutoFirma como aplicación del sistema se lleva a cabo en el msi obtenido al ejecutar el fichero AutoFirma\_to\_msi\_32.bat . En el mismo directorio deberán existir:

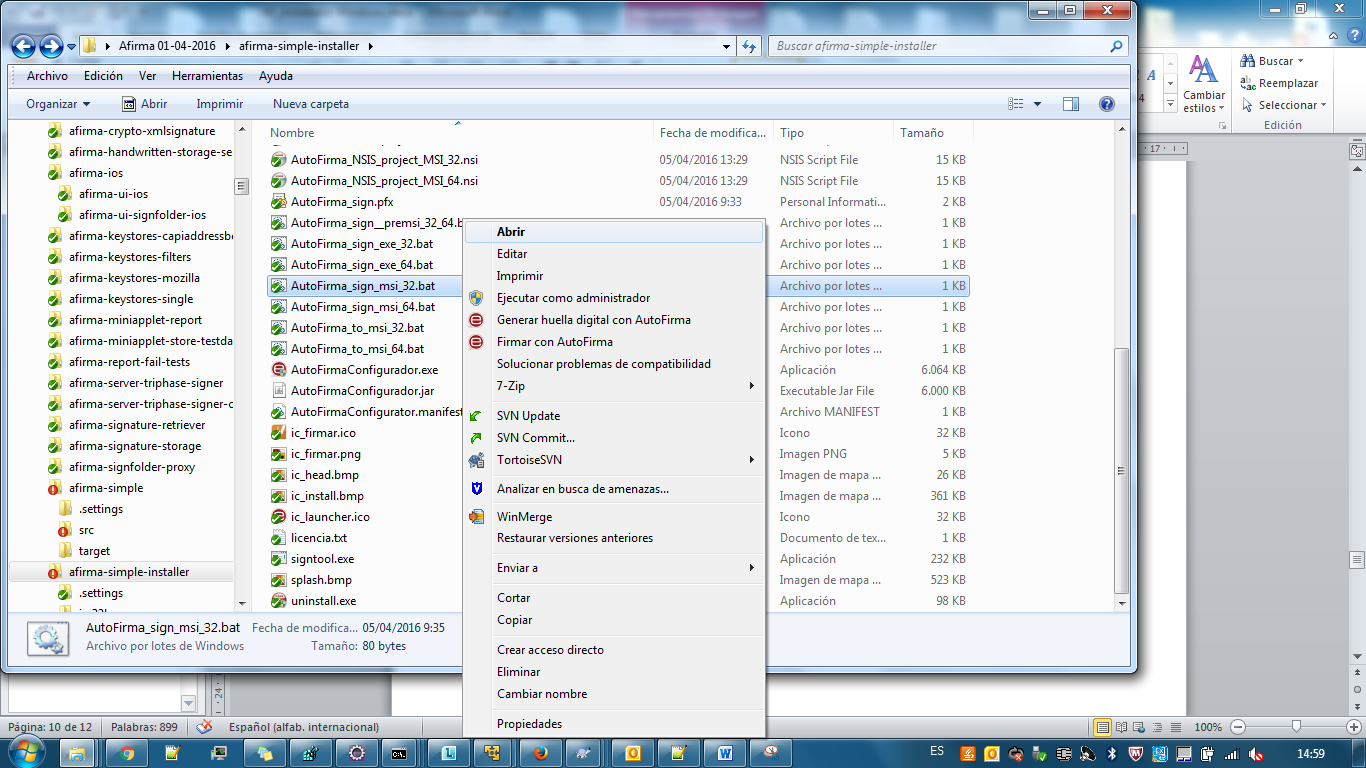
* AutoFirmaGenerator.exe (generado anteriormente)
* uninstall.exe (en el SVN)
* AutoFirma\_32\_installer.wxs (en el SVN)
* Iconos de la aplicación (en el SVN)



**Figura 12**

Es necesario destacar que el uninstall.exe ha sido generado por el instalador del punto anterior (AutoFirma\_32\_installer.exe). Se ha instalado la aplicación y de ahí se ha obtenido el ejecutable creado en C:/Archivos de Programa (x86)/AutoFirma/uninstall.exe. La librería WIX requiere que todos los ficheros de instalación y desinstalación estén disponibles antes de empezar a crear el .msi, por lo que el ejecutable uninstall.exe debe haber sido creado antes y **en caso de modificar la sección “uninstall” del “AutoFirma\_NSIS\_project\_EXE\_32.nsi”** **deberá subirse un nuevo uninstall.exe firmado con el que generar el instalador msi.**

Al ejecutar el .bat se crean los ficheros AutoFirma\_32\_installer .wixobj, AutoFirma\_32\_installer.wixpdb y **AutoFirma\_32\_installer.msi** que será el instalador .msi final que debe ser firmado con AutoFirma\_sign\_msi\_32.bat.



**Figura 13**

En caso de utilizar una versión o un nombre de proyecto diferente a “AutoFirma” y “1.4.2” se deben cambiar las primeras líneas del fichero AutoFirma\_32\_installer.wxs:

* <?define Version = "1.4.2" ?>
* <?define ProjectName = "AutoFirma" ?>

Por último la instalación del fichero .msi en una máquina puede llevarse a cabo abriendo directamente el fichero o mediante consola de manera silenciosa con la ejecución del siguiente comando:

msiexec /i "\_RUTA\_\AutoFirma.msi" /quiet

De la misma forma su desinstalación se llevaría a cabo de la siguiente manera:

msiexec /x "\_RUTA\_\AutoFirma.msi" /quiet

Al igual que el primer instalador es posible obtener una versión de 64 bits ejecutando los ficheros que contengan un “64” como nombre de fichero en vez de un “32”.

Adicionalmente, **la ejecución del .msi cerrará los navegadores abiertos del equipo para realizar la instalación y restaurará las pestañas al finalizarla**, por lo que es conveniente avisar al usuario en caso de ser un administrador el que instala las aplicaciones remotamente en otros sistemas.

[Creative Commons License](#Licencia_Creative_Commons)

Esta obra está bajo una licencia [Creative Commons Reconocimiento-NoComercial-CompartirIgual 3.0 Unported](#Licencia_Creative_Commons).