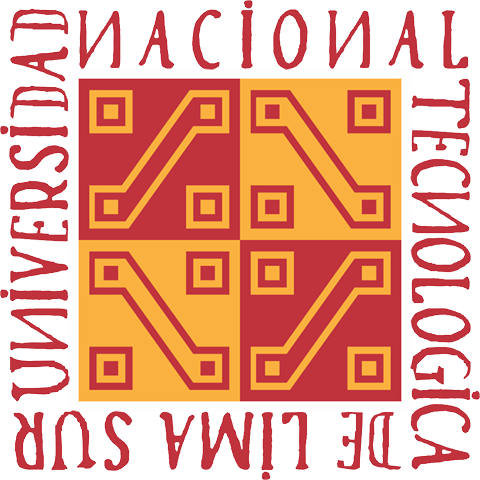
UNIVERSIDAD NACIONAL TECNOLÓGICA DE LIMA SUR

FACULTAD DE INGENIERÍA Y GESTIÓN

E.A.P. DEINGENIERÍA DE SISTEMAS



WINDOWS DESATENDIDO INTEGRADO

CON SOFTWARES LIBRES

Trabajo presentado en cumplimiento parcial

de la materia de Sistemas de Tiempo Real

Por:

Cumpa Capcha Grismer Elias

Ciudad universitaria, Agosto 2020

# **INDICE**

[**INDICE** ii](#_Toc47313487)

[**INTRODUCCION** v](#_Toc47313488)

[**PROGRAMAS REQUERIDOS** 1](#_Toc47313489)

[**DESCOMPRIMIR Y GUARDAR LA ISO WINDOWS 10** 3](#_Toc47313490)

[**CARGA Y MODIFICACION DE LA ISO** 5](#_Toc47313491)

[1) Cargar la carpeta de la iso 5](#_Toc47313492)

[2) Modificacion de la iso 6](#_Toc47313493)

[3) Generando la nueva iso 7](#_Toc47313494)

[**INTEGRAR LAS APLICACIONES A LA ISO** 9](#_Toc47313495)

[1. Softwares libres 9](#_Toc47313496)

[2. Integracion 10](#_Toc47313497)

[**EJECUCION EN MAQUINA VIRTUAL** 17](#_Toc47313498)

# **INDICE DE FIGURAS**

[**Figura 1** Logo software Ntlite 1](#_Toc47314640)

[Figura 2 Logo software VirtualBox 2](#_Toc47314641)

[Figura 3 Interfaz software USSF 2](#_Toc47314642)

[Figura 4 Pagina oficial de Microsoft 3](#_Toc47314643)

[Figura 5 Ruta carpeta de la iso descomprimida 3](#_Toc47314644)

[Figura 6 Carpetas necesarias creadas en el Escritorio 4](#_Toc47314645)

[Figura 7 Carga de la iso en NtLite 5](#_Toc47314646)

[Figura 8 Iso cargada completamente 6](#_Toc47314647)

[Figura 9 Lista de componentes de la iso en NtLite 6](#_Toc47314648)

[Figura 10 Procedimiento de eliminación de componentes de la iso 7](#_Toc47314649)

[Figura 11 Generando la nueva iso 7](#_Toc47314650)

[Figura 12 Carpeta de isos generadas 8](#_Toc47314651)

[Figura 13 Paginas oficiales de las aplicaciones libres a integrar 9](#_Toc47314652)

[Figura 14 Carpeta con los instaladores de las aplicaciones a integrar 10](#_Toc47314653)

[Figura 15 Ruta a la carpeta Source 11](#_Toc47314654)

[Figura 16 Ruta a la carpeta $OEM$ 11](#_Toc47314655)

[Figura 17 Ruta a la carpeta $$ 12](#_Toc47314656)

[Figura 18 Ruta a la carpeta Setup 12](#_Toc47314657)

[Figura 19 Ruta a la carpeta Scripts 13](#_Toc47314658)

[Figura 20 Comando de inicio para la integracion 13](#_Toc47314659)

[Figura 21 Selección de archivo para la generación del comando de instalación silenciosa con USSF 14](#_Toc47314660)

[Figura 22 Comando generado con USSF 14](#_Toc47314661)

[Figura 23 Archivo txt con los comandos necesarios para la integracion 15](#_Toc47314662)

[Figura 24 Selección del modo de guardado del archivo txt 15](#_Toc47314663)

[Figura 25 Cambio de archivo txt a cmd 16](#_Toc47314664)

[Figura 26 Archivo cmd generado en la carpeta Scripts 16](#_Toc47314665)

[Figura 27 Máquina virtual creada en VBox 17](#_Toc47314666)

[Figura 28 Selección de iso para el booteo 18](#_Toc47314667)

[Figura 29 Ejecución de la iso 18](#_Toc47314668)

[Figura 30 Pantalla de inicio de Windows desatendido 19](#_Toc47314669)

[Figura 31 Confirmacion de entrada a travez del ingreso del PIN 19](#_Toc47314670)

[Figura 32 Escritorio de Windows desatendido y ventana de VirtualBox 20](#_Toc47314671)

[Figura 33 Aplicaciones libres instaladas correctamente 21](#_Toc47314672)

# **INTRODUCCION**

Conforme avanzamos en la materia de Sistemas de Tiempo Real pudimos aprender sobre los diferentes sistemas tecnológicos de un computador tanto en torno al hardware como software, viendo los temas de periféricos, placas madres, controladores, estructura de un computador, funciones y utilidad de las unidades lógicas, etc. Con estos conocimientos se propuso a entender como estaba estructurada un sistema operativo, el cual nos lleva a la tarea propuesta de crear un sistema operativo Windows 10 sin aplicaciones innecesarias para una empresa y/o uso personal como por ejemplo Cortana, Xbox, tienda de Windows, etc.

Estos programas innecesarios consumirían más espacio de lo que se querría en un computador, así que creando esta nueva imagen del sistema operativo podríamos instalarlo y que este ya no tuviera dichos programas.

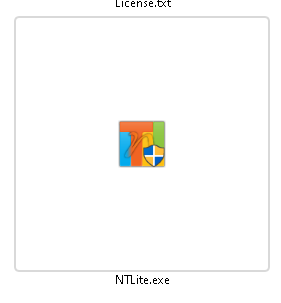
Luego se decidió añadir programas libres, sin paga que si podríamos necesitar en el día a día como por ejemplo el open office para poder leer documentos, vlc para la visualización de multimedia, etc. Buscando así una iso personalizada y con todo lo requerido sin programas innecesarios para la empresa en cuestión.

# **PROGRAMAS REQUERIDOS**

Lo que debemos tener para realizar la modificación y generación de nuestra iso desatendida son algunos softwares que nos ayudaran a estructurar y simular nuestras pruebas, estas son:

1. Ntlite

Este es el software con el cual podremos cargar y modificar nuestra iso para reducir su peso al eliminar ciertos componentes que nos resulte innecesarios en la iso actual. Al mismo tiempo es la herramienta por la cual generaremos la nueva iso.



**Figura 1** Logo software Ntlite

1. VirtualBox

Software utilizado para crear maquinas virtuales para diferentes Sistemas Operativos como Mac, linux, Windows, etc.

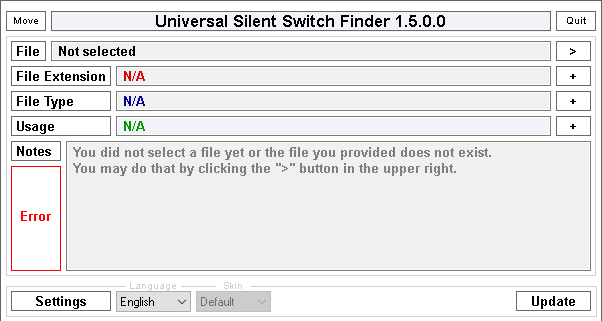
Lo usaremos para simular las pruebas de las isos generadas luego de modificarlas, para saber si se cometio o no un error.



**Figura 2** Logo software VirtualBox

1. Universal Silent Switch Finder

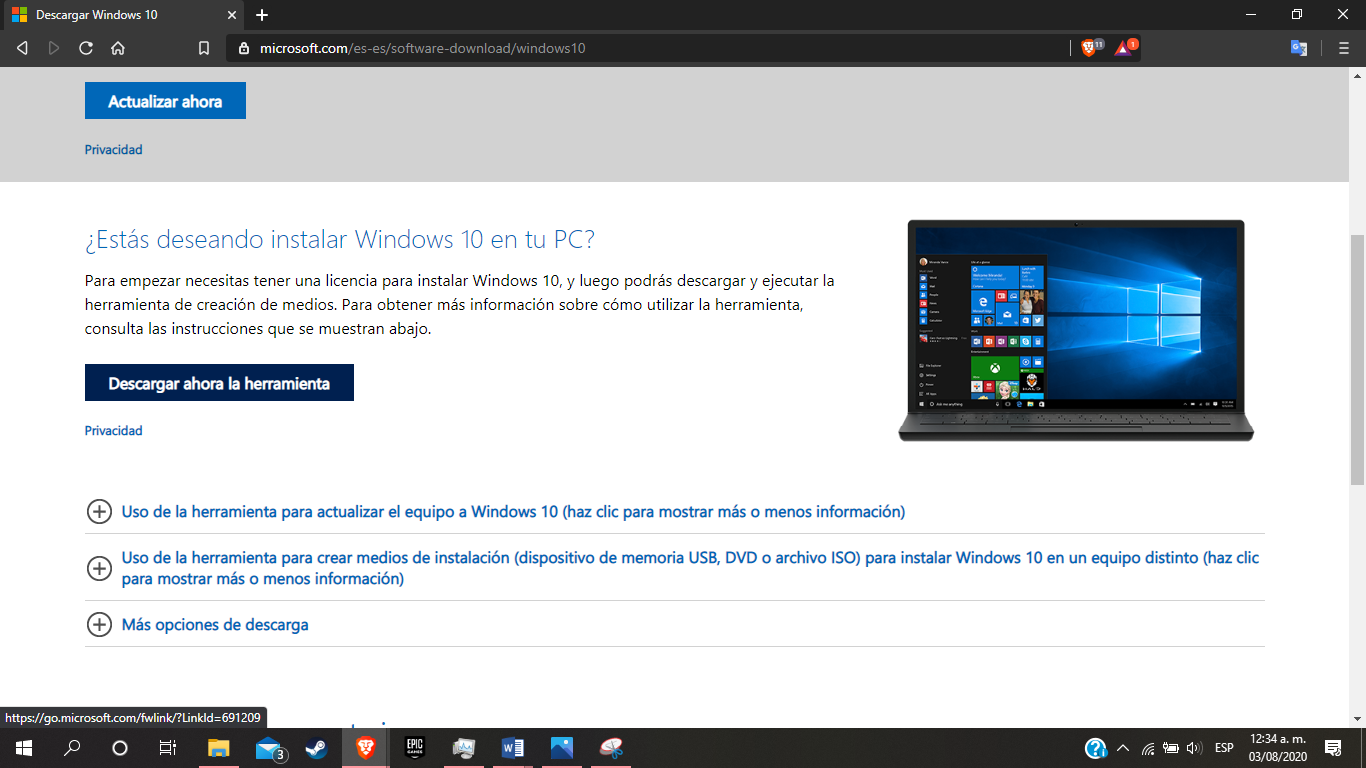
Esta herramienta portable nos ayudara a conocer los comandos para ejecutar en la consola integrada en las isos, para asi poder mandar a que el sistema operativo instale las aplicaciones que integremos automáticamente en medio del proceso de booteo.



**Figura 3** Interfaz software USSF

# **DESCOMPRIMIR Y GUARDAR LA ISO WINDOWS 10**

Lo primero que debemos hacer es obtener la iso del sistema operativo el cual será Windows 10 para este trabajo, lo obtendremos desde la pagina oficial de Microsoft.

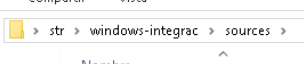


**Figura 4** Pagina oficial de Microsoft

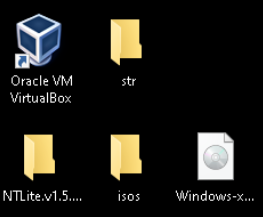
Lo que se hara es descargar la herramienta, ejecutarla y seguir las instrucciones que te indiquen, seleccionando el idioma y que versión de Windows quieres que en el presente trabajo será el Windows 10 de 64 bits.

Una vez obtenido se recomienda guardarlo en el escritorio y crear una carpeta para guardar las isos generadas en un futuro, en este caso yo lo llame “isos” a la carpeta.

Luego, obtenido el archivo iso lo pasaremos a descomprimir en una carpeta, yo lo descomprimi en una carpeta creada la cual llame “windows-integrac” creándola anteriormente dentro una carpeta nombrada “str” el cual cree en el escritorio.



**Figura 5** Ruta carpeta de la iso descomprimida



**Figura 6** Carpetas necesarias creadas en el Escritorio

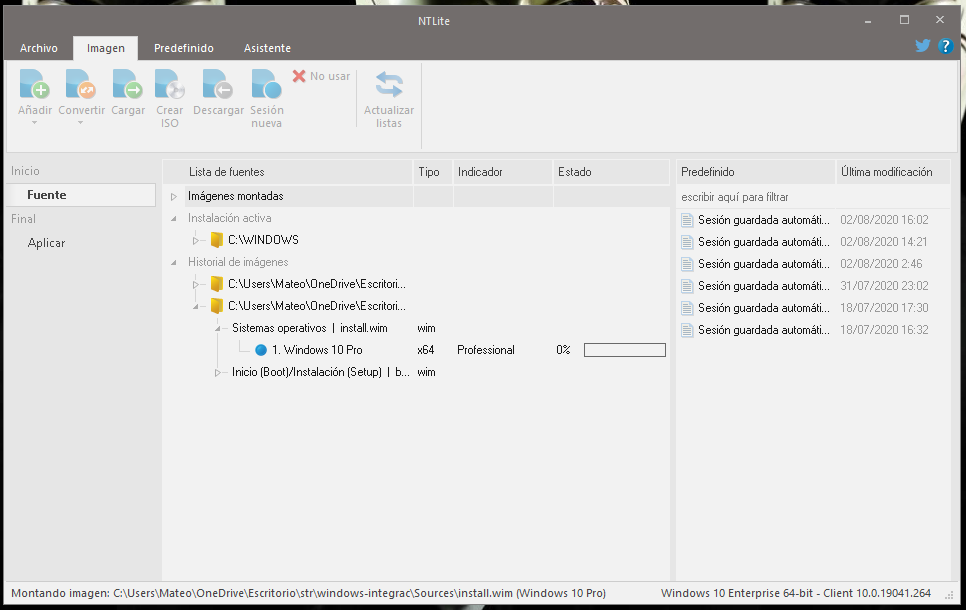
# **CARGA Y MODIFICACION DE LA ISO**

Para este siguiente procedimiento usaremos el software Ntlite el cual comos e menciono antes nos ayudara en la mayoría del proceso, ya que será el que cargue, modifique y cree la iso desatendida.

## Cargar la carpeta de la iso

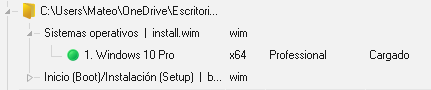
Teniendo la carpeta donde se descomprimio la iso procederemos a añadirla a la herramienta Ntlite haciendo click en Añadir, Seleccioanr carpeta de iso y eligiendo la carpeta respectiva.

Luego visualizaremos la ruta y los sistemas operativos que estén disponibles para cargar, seleccionaremos la versión Pro y daremos click derecho y Cargar.



**Figura 7** Carga de la iso en NtLite

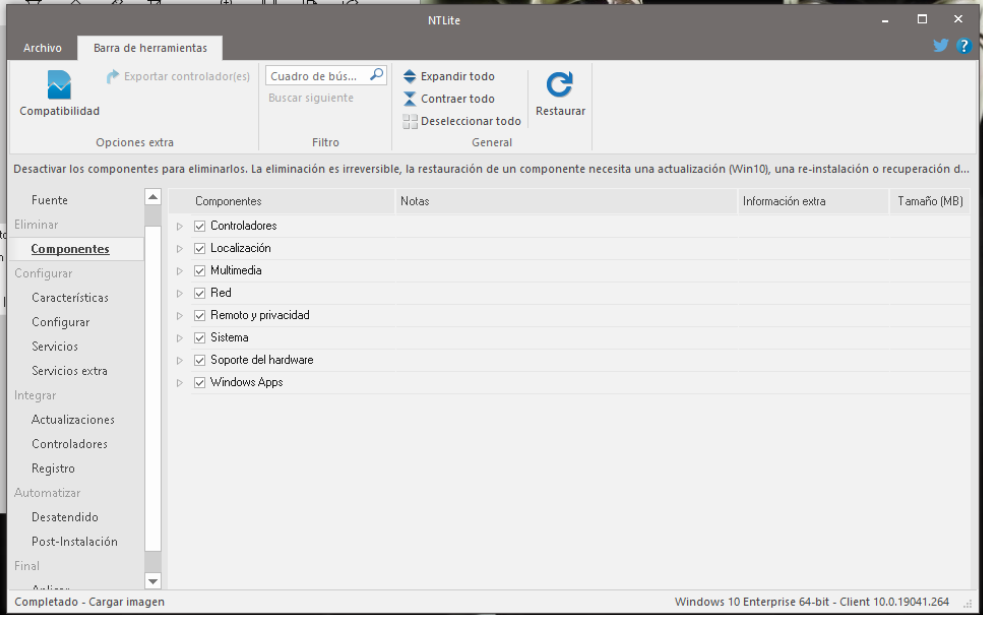
Cuando finalice la carga el circulo azul que se pudo ver pasara a estar verde y se podra ver un menú en la parte izquierda con las opciones que tenemos para aplicar.



**Figura 8** Iso cargada completamente

## Modificacion de la iso

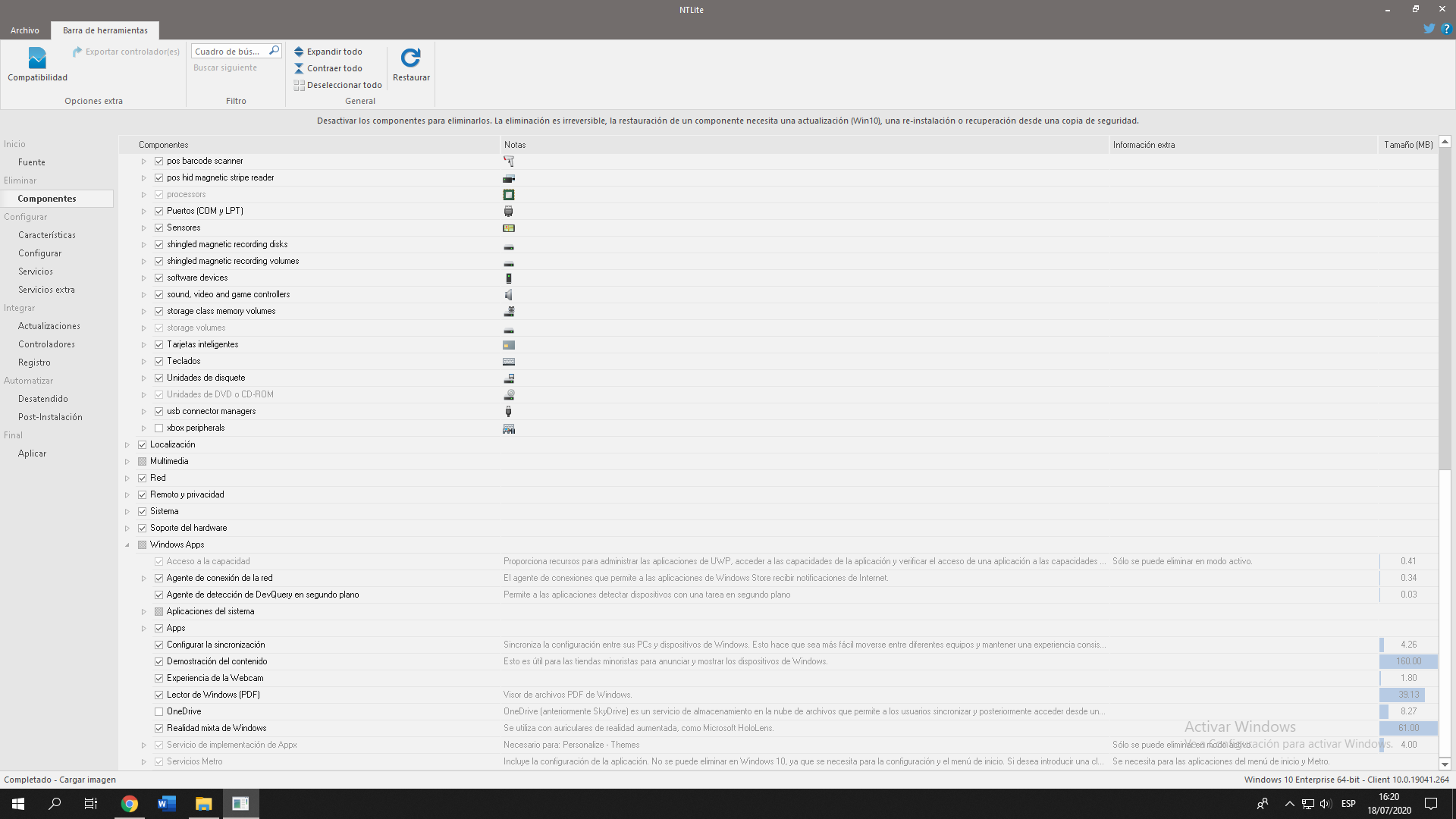
Ahora teniendo la iso cargada y con el menú a la izquierda para la modificación iremos a la opción componentes para poder eliminar las apps y servicios que no se requieran en la iso.



**Figura 9** Lista de componentes de la iso en NtLite

Podemos visualizar todas los archivos que tiene nuestra iso desde controladores hasta las aplicaciones funcionales.

Procederemos a quitar lo que no necesitemos (quitando el check de los cuadros) como los controladores del servicio Xbox, las fuentes e idiomas excepto por el nuestro y opcional dejar el inglés, aplicaciones de multimedia, control parental, etc.

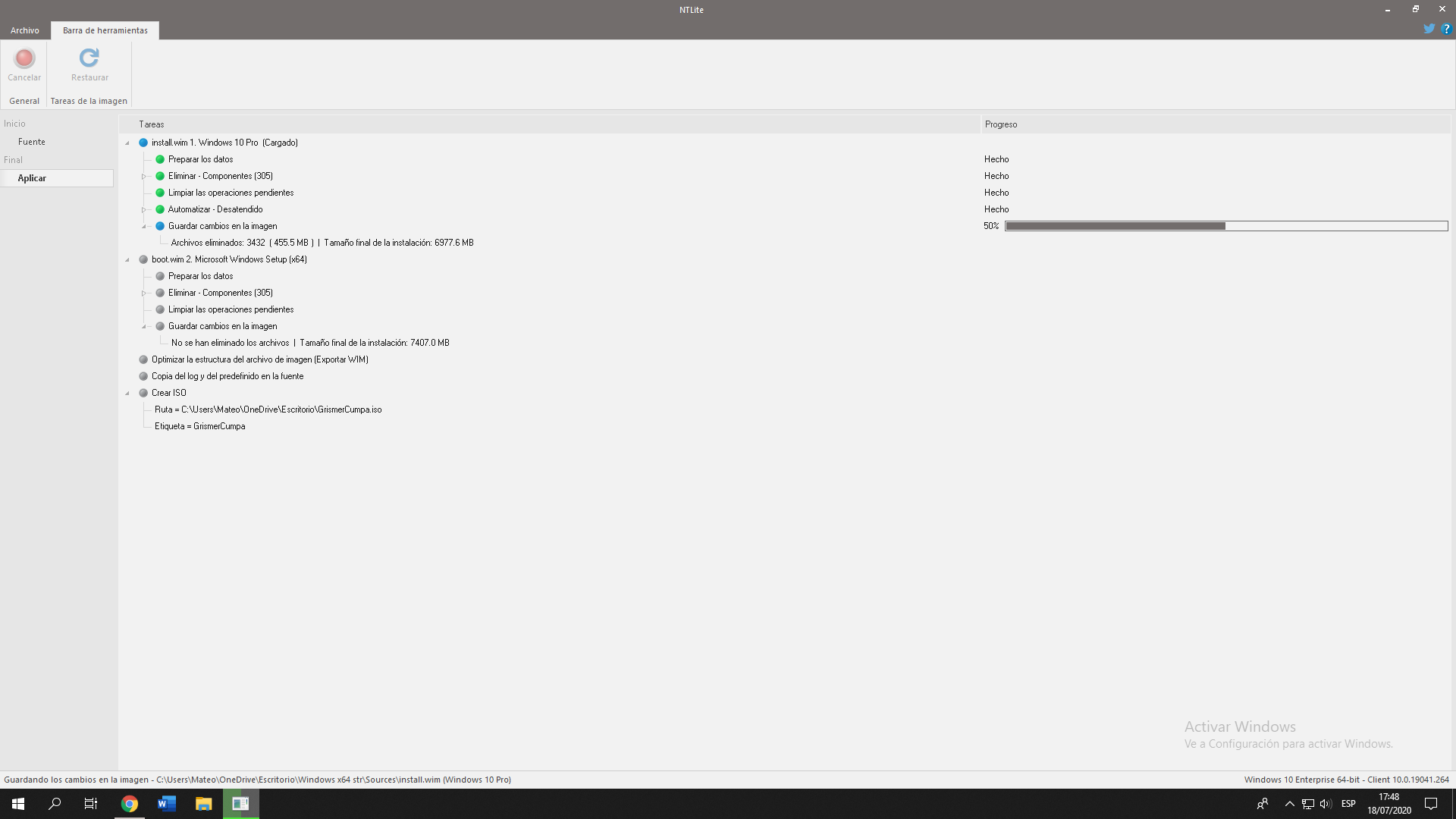


**Figura 10** Procedimiento de eliminación de componentes de la iso

## Generando la nueva iso

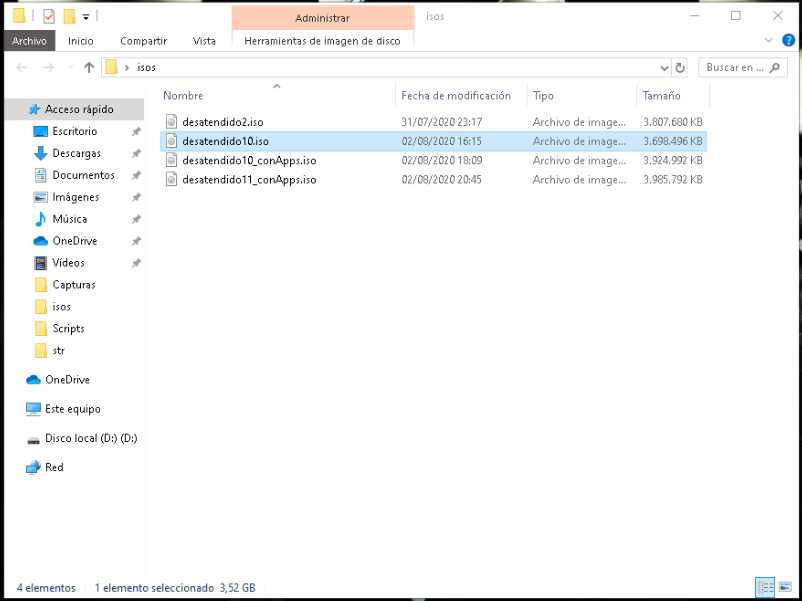
Una vez terminado iremos a la opción aplicar en el menú izquierdo y veremos los procesos que podemos hacer luego de aplicar estos cambios.

Le daremos check a crear ISO, elegiremos donde guardarlo (En este caso en la carpeta “isos” que se creo al inicio) y daremos a procesar que se encuentra en la parte superior izquierda de la interfaz de Ntlite.



**Figura 11** Generando la nueva iso

Una vez se complete los procesos y toda la lista este con el punto verde, se podrá ver la iso generada en la carpeta que elegimos como destino.



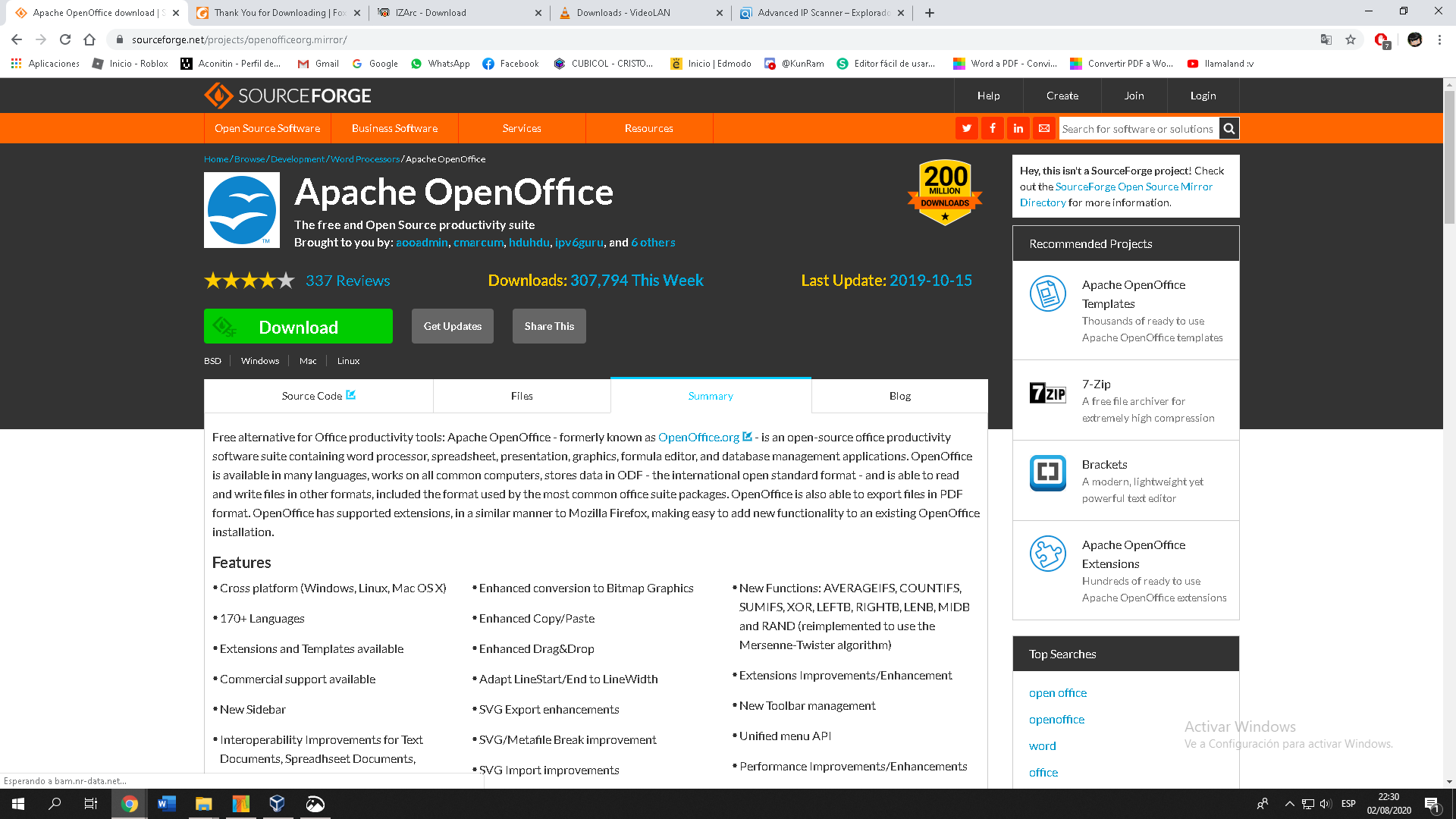
**Figura 12** Carpeta de isos generadas

# **INTEGRAR LAS APLICACIONES A LA ISO**

Teniendo ya la iso reducida pasaremos a integrar las aplicaciones libres.

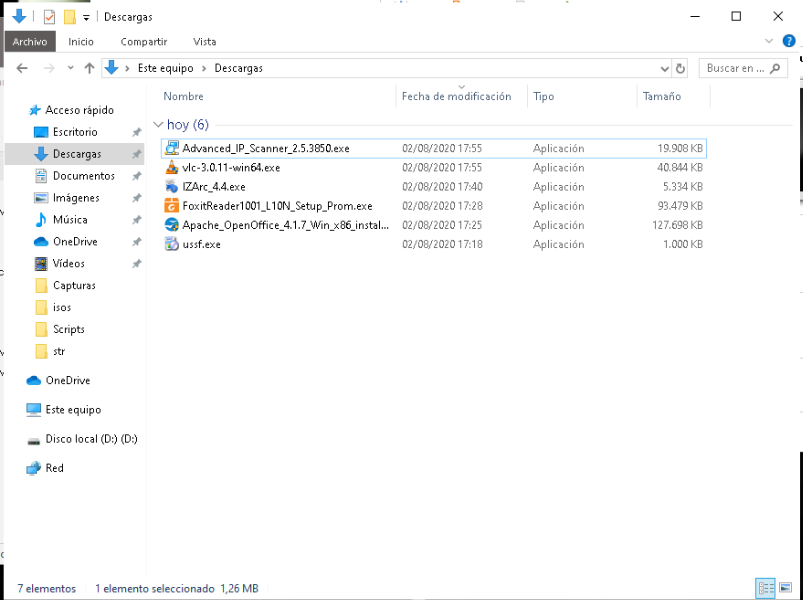
## Softwares libres

Para poder integrar estas aplicaciones tan solo necesitamos sus instaladores que podremos conseguirlos en las páginas oficiales de cada programa.



**Figura 13** Paginas oficiales de las aplicaciones libres a integrar

Y las guardaremos en una carpeta o en todo caso lo podemos dejar en la carpeta Descargas donde se guardaran por defecto al descargar.

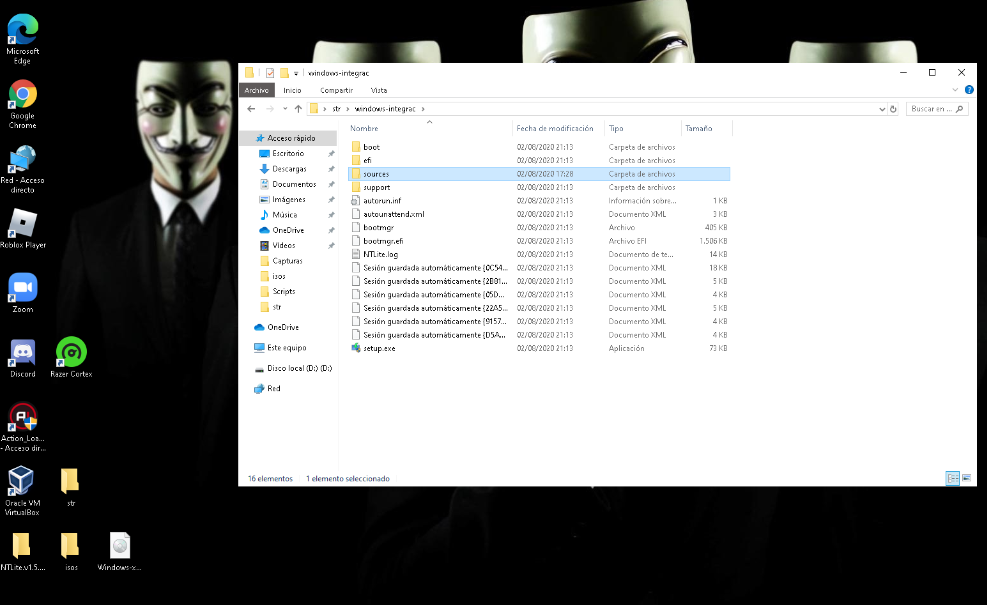


**Figura 14** Carpeta con los instaladores de las aplicaciones a integrar

## Integracion

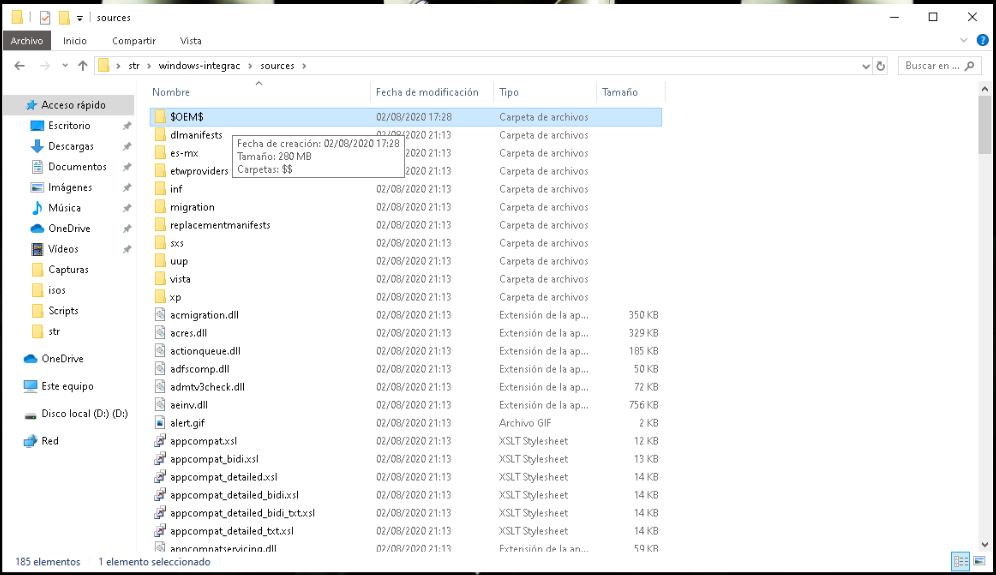
Teniendo listo los instaladores podremos empezar con la integración.

* Para esto lo primero que haremos será ir a la carpeta donde tenemos la iso descomprimida (de la ultima iso generada) y entraremos a la carpeta source.



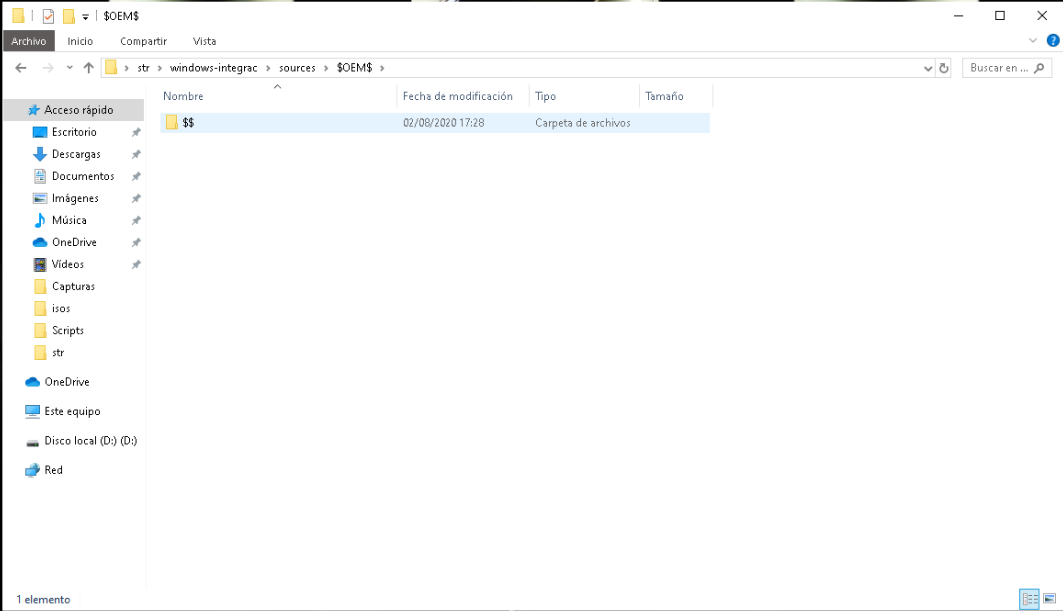
**Figura 15** Ruta a la carpeta Source

* Dentro de esta crearemos una carpeta con el nombre “$OEM$”.



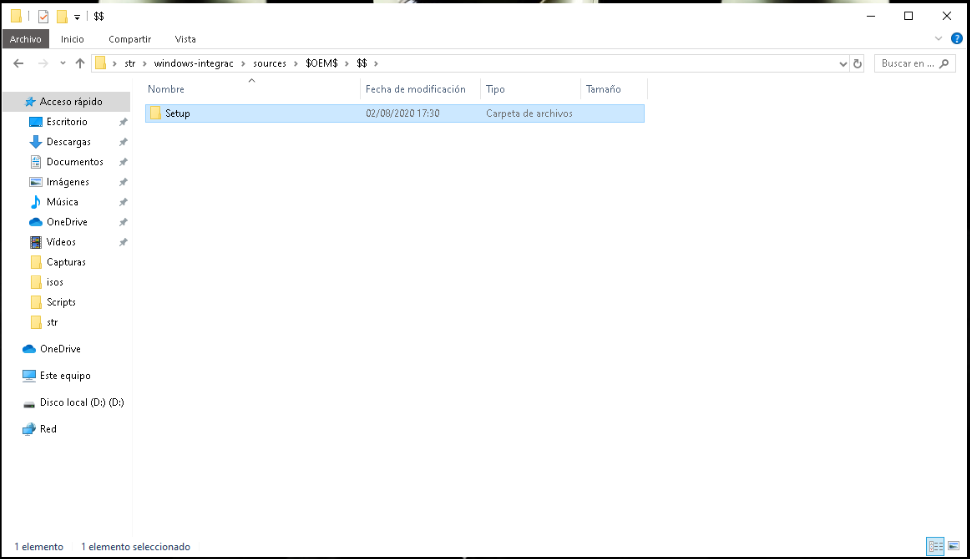
**Figura 16** Ruta a la carpeta $OEM$

* Ahora dentro de esa carpeta crearemos una nueva carpeta llamada “$$”



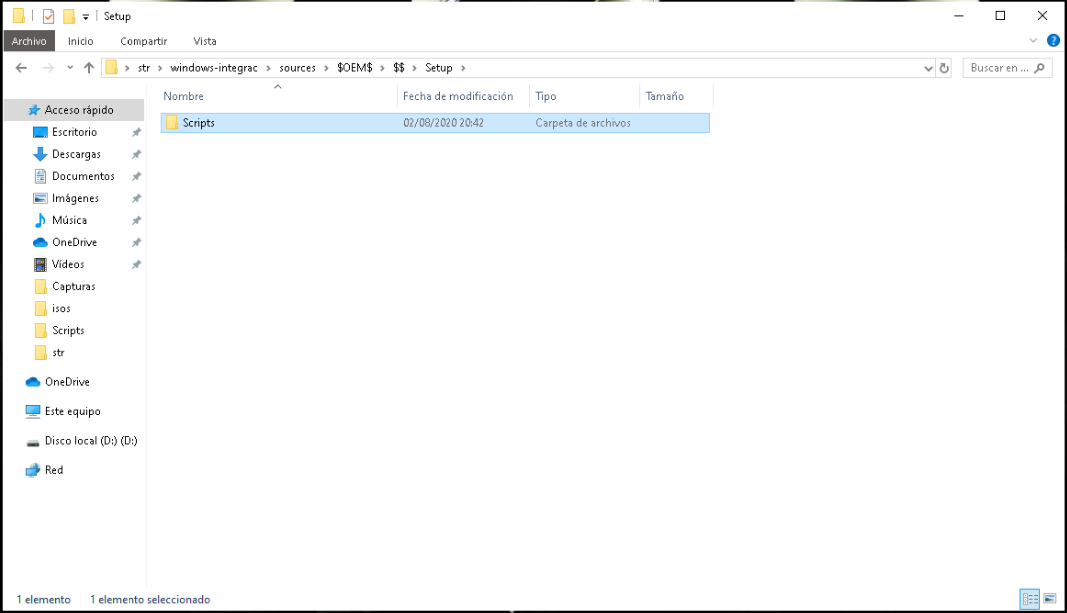
**Figura 17** Ruta a la carpeta$$

* Dentro de esta ahora crearemos una nueva carpeta llamada “Setup”



**Figura 18** Ruta a la carpeta Setup

* Crearemos una nueva carpeta dentro de Setup llamada “Scripts”



**Figura 19** Ruta a la carpeta Scripts

* Y dentro de esta carpeta Scripts copiaremos los instaladores de los programas que queremos integrar, y a continuación crearemos un txt llamado “setupcomplete”.

Luego dentro del txt escribiremos lo siguiente

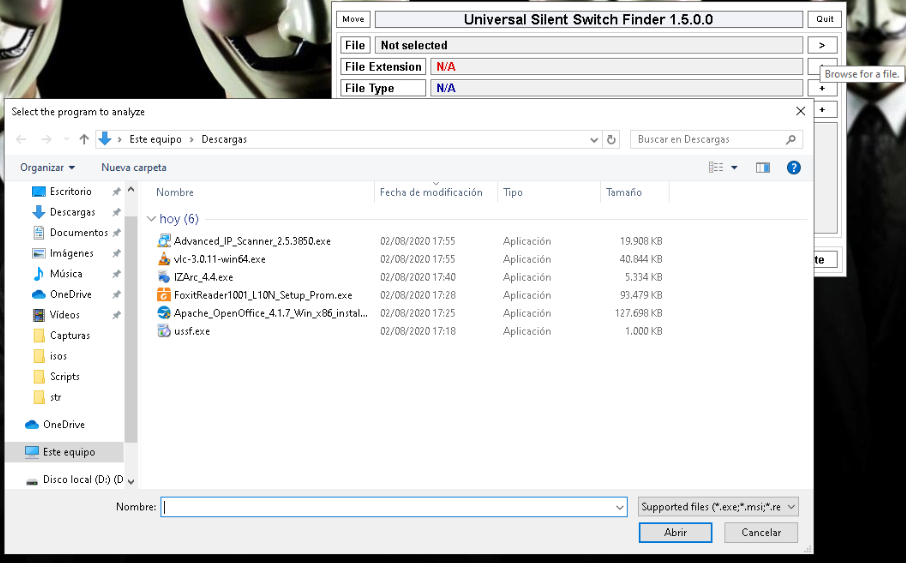


**Figura 20** Comando de inicio para la integracion

Y luego escribiremos los comandos:

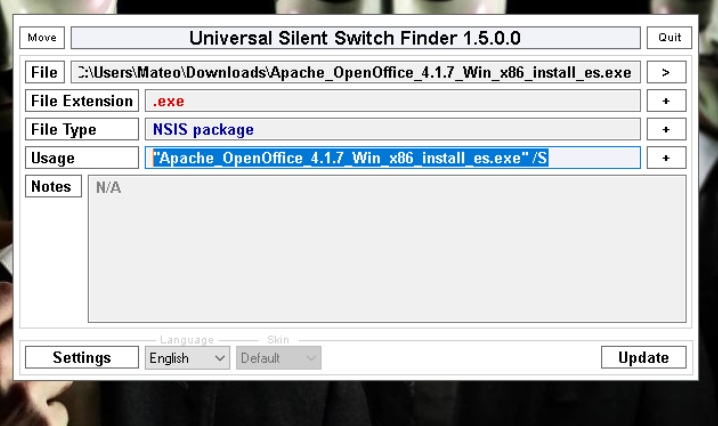
* start /wait %windir%\Setup\Scripts\”nombre del instalador” /(commando para instalacion silenciosa)
* Para obtener los comandos de instalación silenciosa usaremos la herramienta Universal Silent Switch Finder.

Ejecutamos la herramienta y elegimos el archivo .exe del instalador que queremos dándole click al icono “>” a la derecha de File.



**Figura 21** Selección de archivo para la generación del comando de instalación silenciosa con USSF

* Luego de eso la herramienta nos generara el comando adecuado para ponerlo en el txt y para que asi la iso pueda ejecutarlo en la instalación.



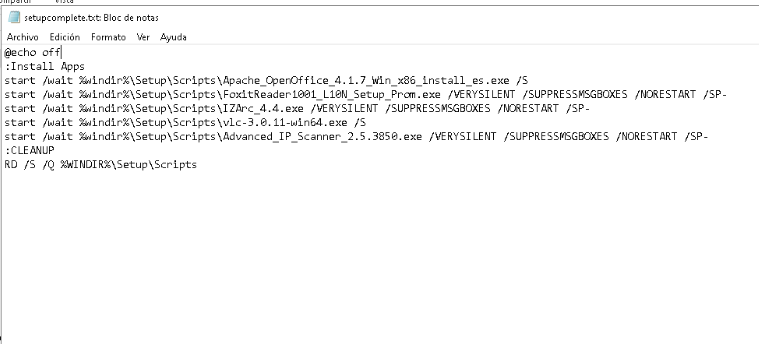
**Figura 22** Comando generado con USSF

* Copiamos el comando en el txt y les quitamos las comillas en caso el nombre del instalador no tenga espacios(si tiene espacios dejar las comillas escritas).

Ahora pondremos el comando para borrar archivos residuales de la instlacion una vez se ejecuten en el sistema operativo, el cual es:

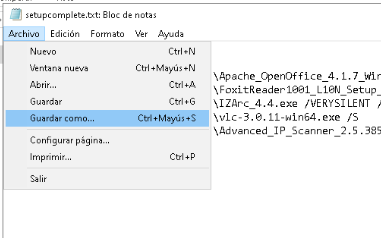
* :CLEANUP:

RD /S /Q %windir%\Setup\Scripts



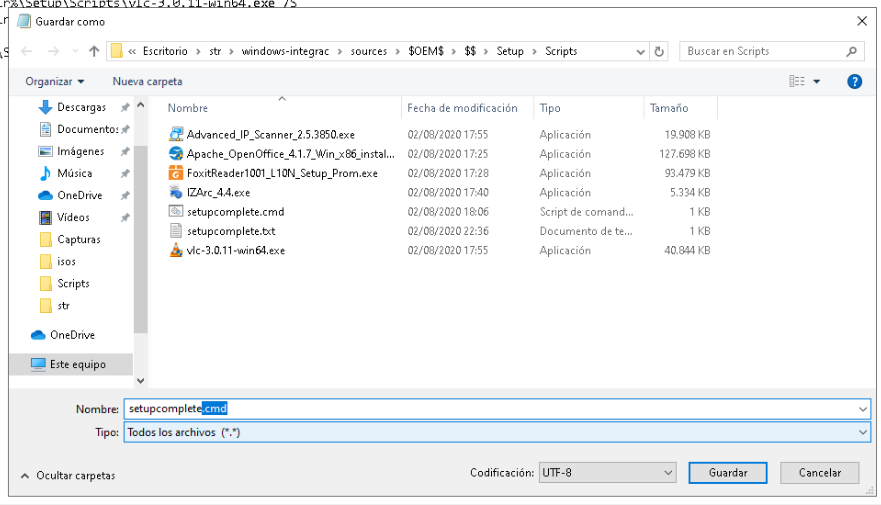
**Figura 23** Archivo txt con los comandos necesarios para la integración

* Una vez terminado de colocar los comandos para todas las aplicaciones guardaremos el txt pero ahora como archivo “.cmd” para que este sea ejecutado por la iso, lo haremos dando a guardar como.



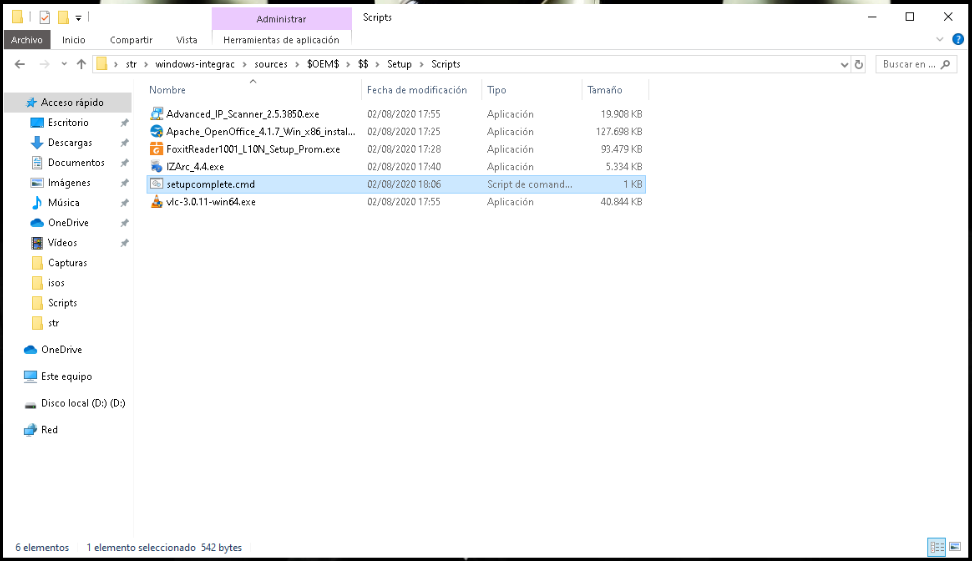
**Figura 24** Selección del modo de guardado del archivo txt

* Luego al momento de guardarlo cambiaremos el .txt por el .cmd y daremos a guardar.



**Figura 25** Cambio de archivo txt a cmd

* Con eso acabaríamos la integración de las aplicaciones en los archivos de la iso, podemos eliminar el archivo txt en caso se guarde en la carpeta.



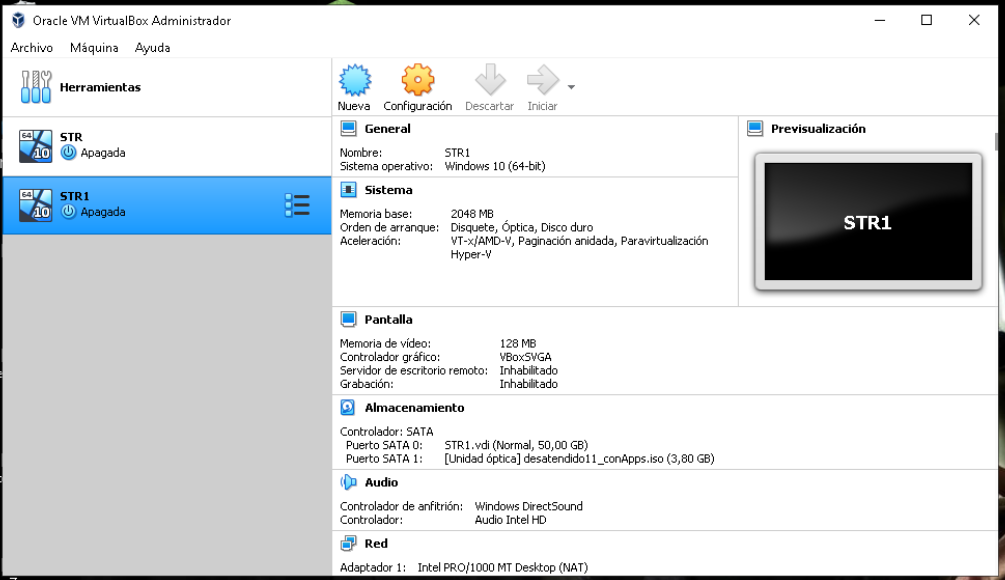
**Figura 26** Archivo cmd generado en la carpeta Scripts

* Para finalizar esta parte usaremos el software Ntlite para crear denuevo la iso pero con estos archivos que hemos agregado, para eso añadiremos la carpeta de la iso denuevo y daremos click derecho a la ruta que hemos añadido y Crear ISO, elegiremos el destino donde guardar la iso y listo.

# **EJECUCION EN MAQUINA VIRTUAL**

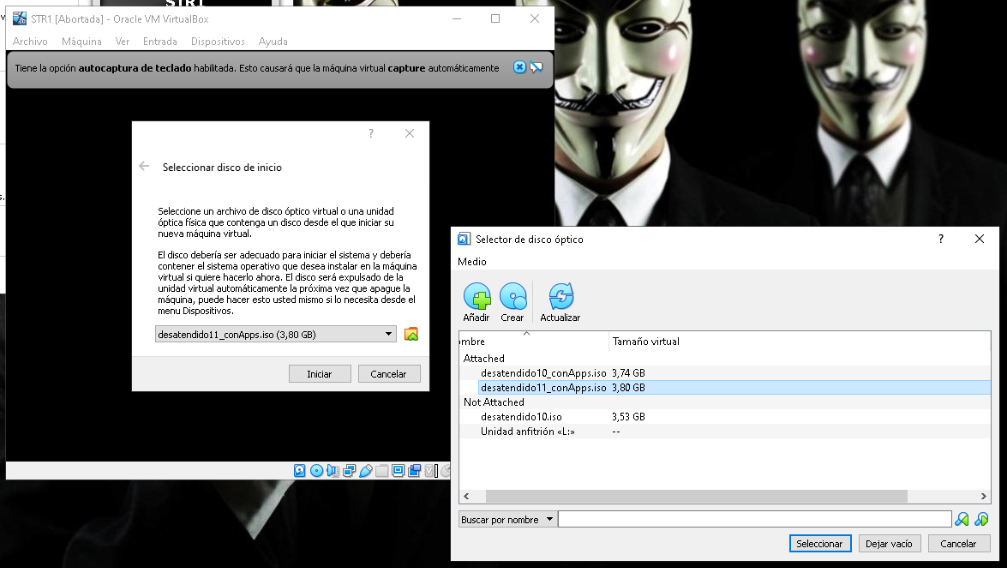
Para visualizar nuestra iso funcionando usaremos el software VirtualBox para que mediante una maquina virtual podamos bootear nuestra iso.

* Primero abrimos el programa y creamos una maquina virtual para windows 10 de 64 bits, dándole a el icono verde Nueva y siguiendo las instrucciones.



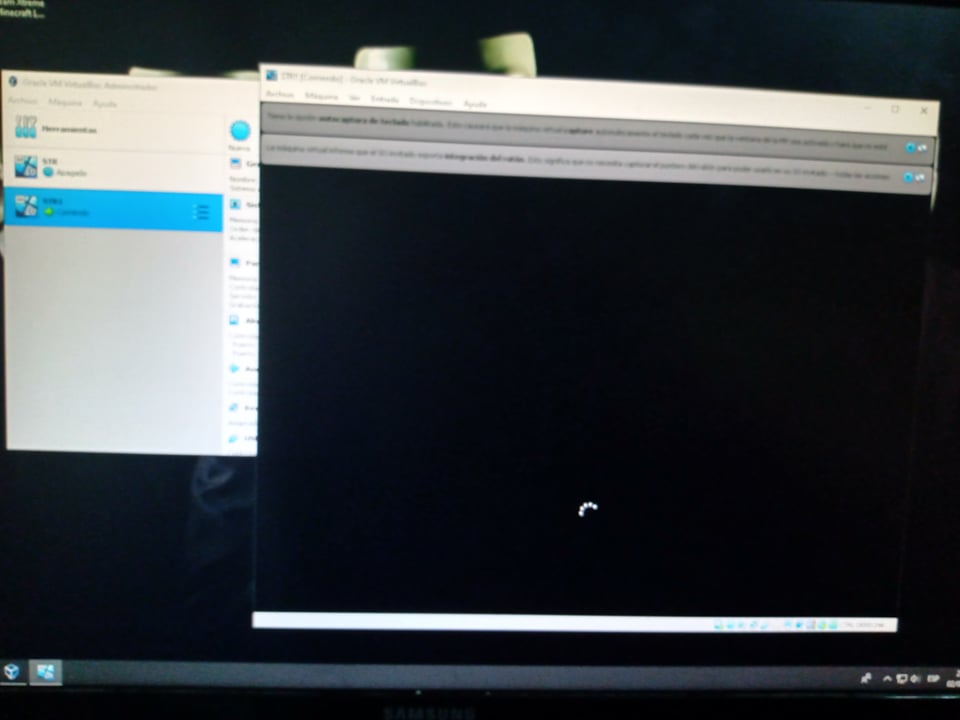
**Figura 27** Máquina virtual creada en VBox

* Una vez creada la maquina virtual le daremos a Iniciar y nos pedirá el archivo iso del sistema operativo. Elegiremos la iso dándole click al icono de una carpetita y eligiendo el iso que generemos. (En este caso mi iso se llama desatendido11\_conApps ya que esta fue la ultima luego de varias modificaciones e integraciones).



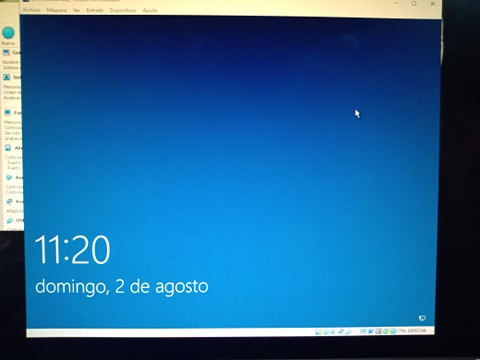
**Figura 28** Selección de iso para el booteo

* Le daremos a Iniciar y se procederá a instalar la iso



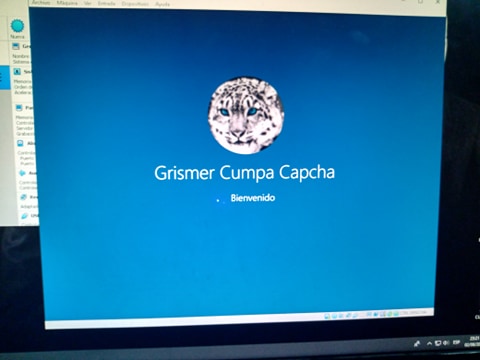
**Figura 29** Ejecución de la iso

* Luego se procede a instalar el sistema operativo como usualmente se hace siguiendo los pasos mostrados y eligiendo nuestras preferencias. Para al final llegar a la pantalla de inicio de windows.



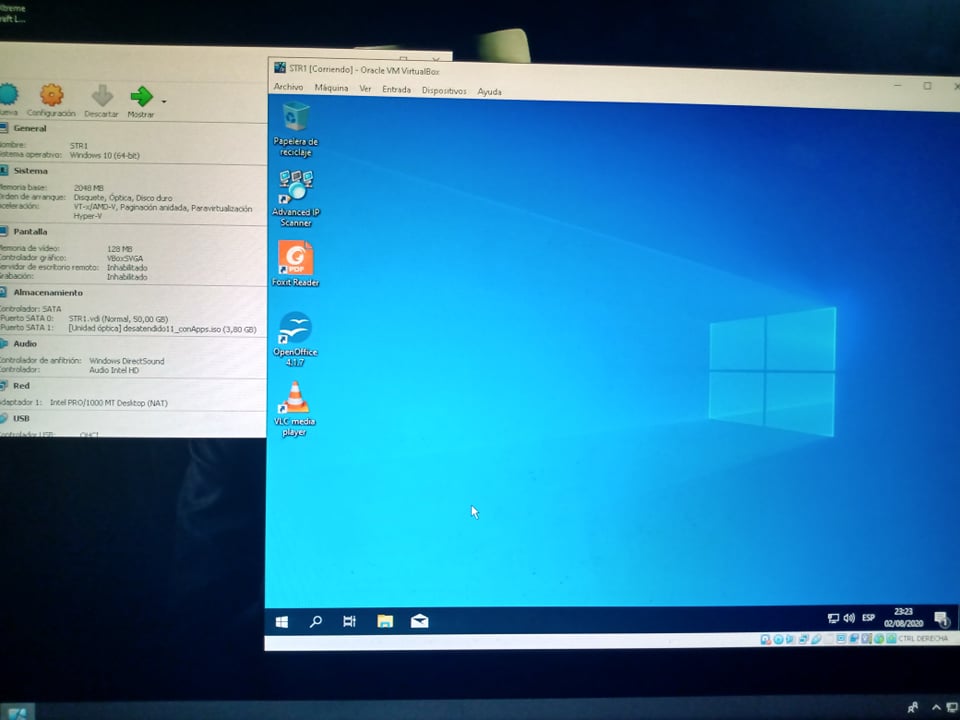
**Figura 30** Pantalla de inicio de Windows desatendido

* Ingresamos con el PIN que creamos.

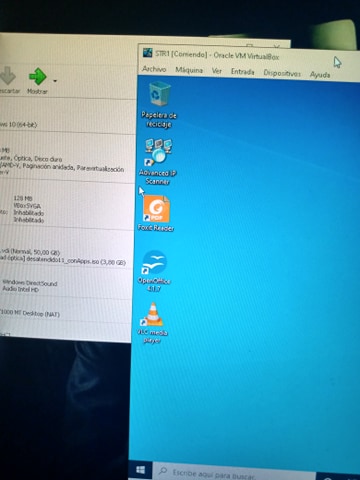


**Figura 31** Confirmacion de entrada a travez del ingreso del PIN

* Y podremos visualizar el escritorio de Windows con los accesos directos de los programas que se integraron.



**Figura 32** Escritorio de Windows desatendido y ventana de VirtualBox



**Figura 33** Aplicaciones libres instaladas correctamente

Con esto acabaríamos con el procedimiento para generar una iso de windows 10 de 64 bits desatendido y con aplicaciones libres integradas.