Notas:

- EyeWitness ya no tiene la función de --activescan, con lo que la opción de --active de domained queda deshabilitada. Con la opción --quick no funciona ya que no lanza Sublist3r.
- Todos los outputs se guardan en la carpeta /domained/output/.
- Recomiendo hacer toda la instalación con usuario root.

Yo realice la instalación sin ningún problema en una carpeta creada en /root/tools/

1. Antes de la instalación.

Instalar varias dependencias (añado las que ya se indican):
apt-get install libldns-dev -y libxml2-dev libxslt-dev python-dev

Instalamos Golang1.14.6 siguiendo estas indicaciones:
https://golang.org/doc/install?download=go1.14.6.linux-amd64.tar.gz

Escribir estas líneas al final del archivo ~/.bashrc tanto para root como para tu usuario actual:
export GOPATH=(Directorio de "trabajo" de Go deseado)

export GOROOT=/usr/local/go (Directorio de instalación de Go)

export PATH=$PATH:$GOROOT/bin:$GOPATH/bin

Instalamos python2 y python3 junto a pip (Normalmente las versiones de Kali ya lo tienen):
apt-get install python2 python3 python-pip

Cambios en fichero domained.py:

Linea 156 -> massdnsCMD = "python {} {} {} | {} -r resolvers.txt -t A -o S -w {}_massdns.txt".format(

Linea 157 -> os.path.join(script_path, "bin/massdns/scripts/subbrute.py"),

Linea 251 -> EWHTTPScriptIPS = "python3 {} -f {} --no-prompt --web -d */domained/{}-{}-EW".format(

*** = En la opción -d indicar vuestro directorio de domained.**

Linea 252 -> os.path.join(script_path, "bin/EyeWitness/Python/EyeWitness.py"),

Eliminar linea 254 (Ya no existe esa opción en EW)

Cambios en fichero installer.py:

Linea 46 -> eyeInstallReq = "bash bin/EyeWitness/Python/setup/setup.sh"

2. Pasamos a la instalación.

Primero instalamos los requirements como indican:
pip install -r ./ext/requirements.txt

Si os da algún fallo de scandir en los modulos de Python yo lo solucione ejecutando:
pip install --upgrade pip

Ya podemos lanzar el script de instalación (debemos hacerlo con usuario root):
python3 domained.py --install
Durante la instalación te pregunta si quieres instalar Golang, yo en todo momento dije que no.

3. Programas.

- Sublist3r funciona desde el principio.

- Enumall: Para reparar enumall tuve que descargar otra versión de Recon-ng. Simplemente debemos sustituir la carpeta de /usr/share/recon-ng por la versión descargada de este repositorio: https://github.com/methos2016/recon-ng

E instalar los modulos de recon-ng por si faltara alguno de la versión anterior:
pip install -r /usr/share/recon-ng/REQUIREMENTS

He cambiado otro par de cosas en el script de enumall.py debido a que guarda los archivos en domained/ y luego no los encuentra para crear el archivo único, yo los quería guardar en la carpeta de domained/output/.
Abrimos enumall.py:

Linea 64 -> outFile = "FILENAME "+os.getcwd()+"/output/"+domains[0]+"_enumall"

- Knock funciona desde el principio.

- Subbrute funciona desde el principio con massdns y con Sublist3r

- **Massdns:** Con massdns debemos sustituir directamente la carpeta en domained/bin/ la cual baje de este repositorio: https://github.com/blechschmidt/massdns
El caso es que en ese repositorio no existe la carpeta bin, lo que hice fue coger la carpeta bin de la instalación que realiza NahamSec en su programa de LazyRecon, dispongo de la carpeta ya modificada con el archivo correcto con lo que la dejare subida en mi repositorio: https://github.com/Inferrrno4tmk/Fix-domained

Sustituyendo la carpeta entera de massdns en domained/bin/ con la “creada” debería funcionar.

- **Recon-ng:** Lo único que hice es sustituir la versión 5.1.1 por la 4.7.1

- **Amass** funciona desde el principio.

- **Subfinder:** Con subfinder he tenido varios casos, si no te funciona tras instalar todo normal, prueba a instalar manualmente subfinder desde este repositorio, siguiendo las indicaciones de instalación “From Github” se instala perfectamente:
https://github.com/projectdiscovery/subfinder

- **EyeWitness:** Con EyeWitness antes teníamos el problema de que no funcionaba con Python3, pero ahora si por ello he hecho los cambios en domained.py

4. Adicional:

He añadido que se ejecute AltDNS cuando se ejecute Enumall, hace que el proceso sea un poco mas lento pero a mi me gusta esa herramienta, su instalación es simple:
pip install py-altdns

El primer cambio que debemos hacer es crear el fichero words.txt en:
/usr/local/lib/python2.7/dist-packages/altdns/

El cual podemos copiar el contenido del repositorio de altdns:
https://github.com/infosec-au/altdns/blob/master/words.txt

Tras tener esto tenemos que actualizar el Path de altdns en el script de enumall.py y en config.py como hicimos con recon-ng:
altDnsPath = "/usr/local/lib/python2.7/dist-packages/altdns/"

Si queremos que AltDNS funcione con domained cada vez que se ejecute Enumall tenemos que volver a editar el fichero **domained.py** y añadir la opción al comando de enumall:
Linea 142 -> enumallCMD = "python {} {} -a".format(

El ultimo cambio es debido a que hemos cambiado la ruta del output de Enumall, con lo que debemos indicarle a AltDNS donde buscar dicho archivo. Vamos al fichero de **enumall.py**:
Linea 103 -> subdomains = os.path.join(os.getcwd()+"/output/", workspace+"_enumall.lst")

Y cambiamos otra linea para que domained no lo elimine al terminar:
Linea 105 -> output = os.path.join(os.getcwd()+"/altdns_"+workspace+"_output.txt")

Con esto ya estaríamos usando AltDNS cuando ejecutamos domained.