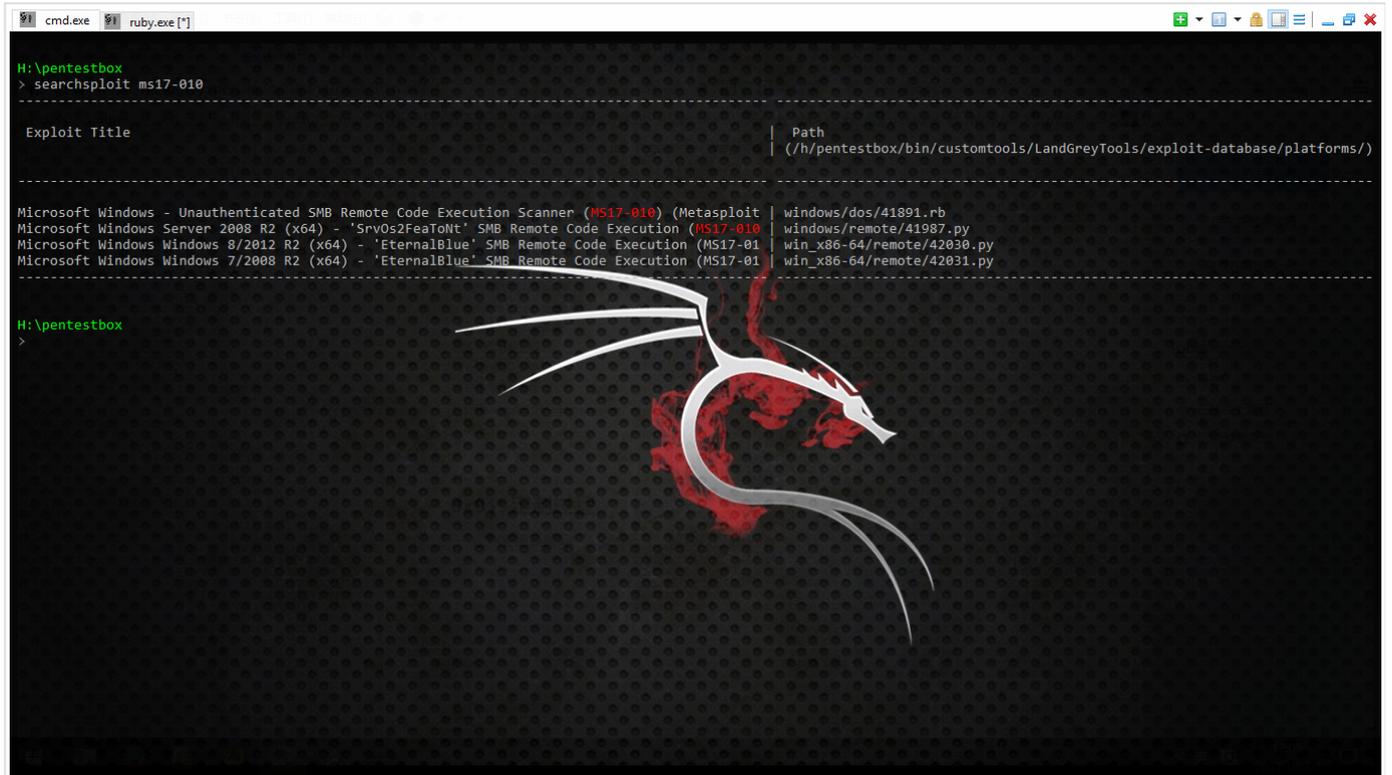


打造自己专属的PentestBox

作者: LANDGREY • 创建时间 2017年6月7日 11:25 • 更新时间 2017年6月27日 21:15

浏览: 1709 次 • 标签: #渗透测试

您的IP地址: 140.207.23.83



```
cmd.exe ruby.exe [*]
H:\pentestbox
> searchsploit ms17-010

-----
Exploit Title | Path
-----|-----
Microsoft Windows - Unauthenticated SMB Remote Code Execution Scanner (MS17-010) (Metasploit | windows/dos/41891.rb
Microsoft Windows Server 2008 R2 (x64) - 'SrvOs2FeaToNt' SMB Remote Code Execution (MS17-010 | windows/remote/41987.py
Microsoft Windows Windows 8/2012 R2 (x64) - 'EternalBlue' SMB Remote Code Execution (MS17-01 | win_x86-64/remote/42030.py
Microsoft Windows Windows 7/2008 R2 (x64) - 'EternalBlue' SMB Remote Code Execution (MS17-01 | win_x86-64/remote/42031.py
-----

H:\pentestbox
>
```

打造自己专属的PentestBox 正文

WS 0X00: 准备

一. 预备知识

请先大致浏览下以下内容:

官网

Windows渗透利器之Pentest BOX使用详解 (一)

Windows渗透利器之Pentest BOX使用详解 (二)

二. 原料

1. PentestBox (官网下载)

2. U盘 (Kingston DTR30G2 32GB)

安装在U盘中, 是为了最大化发挥PentestBox易携带的优点, 即插即用, 去依赖化;

推荐购买32G大小, 支持USB 3.0的优质U盘, 因为读写比较频繁, 质量差的U盘不能胜任;

另外选中 Kingston DTR30G2 32GB 的原因，主要是橡胶套，一定程度上防水、防尘、抗震，安全的特点。



3. 移动硬盘或备份U盘(可选)

主要是用作安全备份，定制中途错误恢复；

最低要求拷贝完整的一份PentestBox到主机硬盘中；

三. 定制思路

由表及里：

- 定制图标和界面
- 定制toolsmanager
- 定制python环境
- 定制默认工具
- 定制自定义命令
- 定制默认命令
- 定制exploit-database
- 其他

0x01: 定制过程

注：可先跳至 0x02: **需要注意的坑** 部分查看，防止踩坑。

一. 定制图标和界面

在U盘根目录下创建目录 pentestbox，将下载的 PentestBox-with-Metasploit-v2.2.exe 直接解压至pentestbox 目录；

接着在U盘根目录下autorun.info 文件，ico.ico文件，hack.bat文件，各个文件内容如下：

autorun.info (自动加载U盘图标)

```
[autorun]
open="" autoplay=true
ICON="ico.ico"
```

ico.ico (U盘图标文件)



hack.bat (双击运行pentestbox)

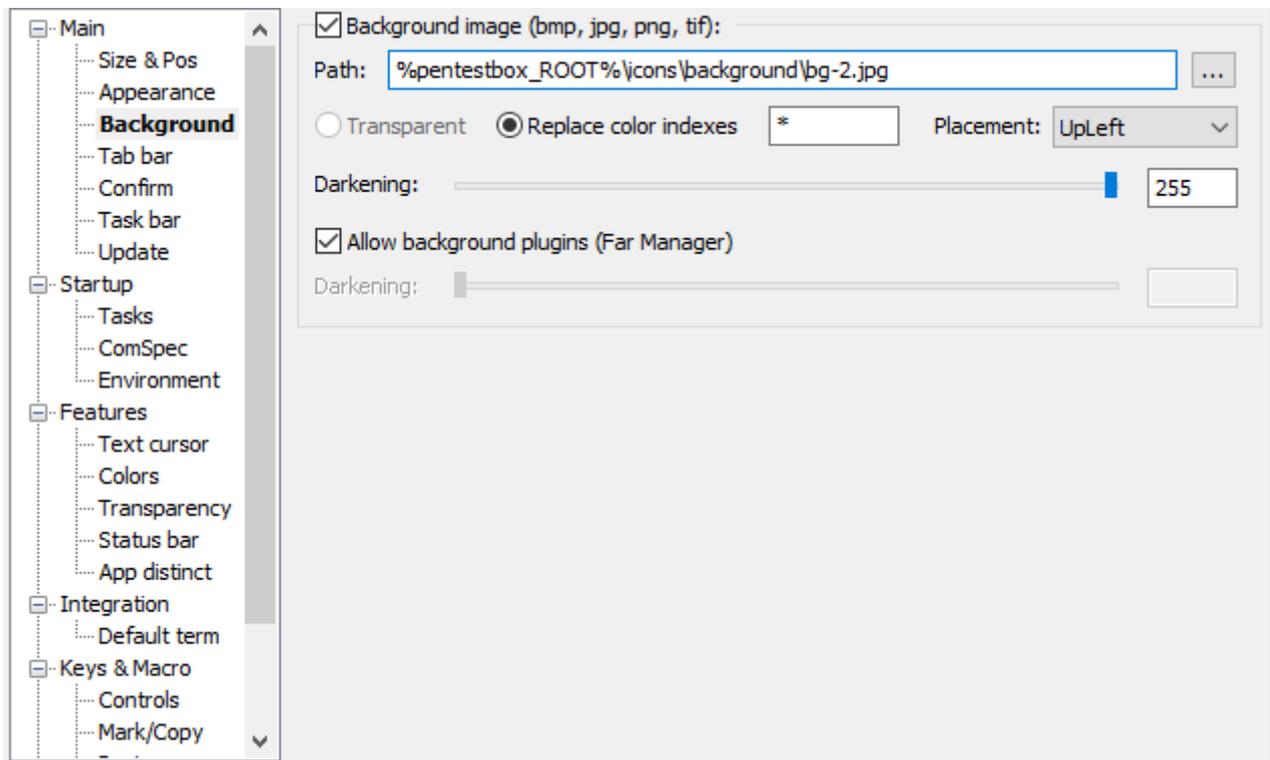
```
@echo off
cd pentestbox
SET pentestbox_ROOT=%CD%
start %CD%/base/conemu-maximus5/ConEmu.exe /Icon "%pentestbox_ROOT%/icons/PentestBox.ico" /LoadCfgFile "%pentestb
```

这里注意要有 \pentestbox\icons\PentestBox.ico 文件，这是启动的窗口的图标文件。

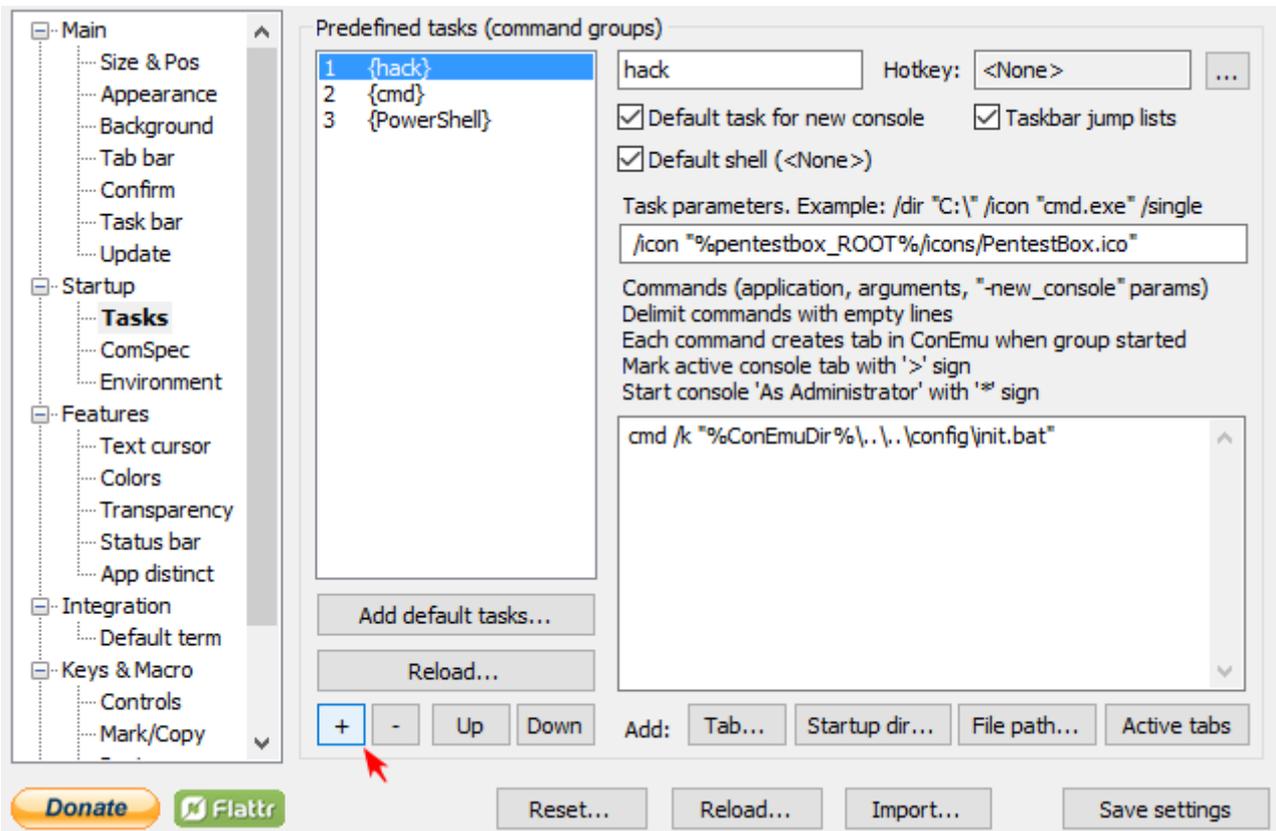
这个脚本主要作用是让ConEmu加载配置文件ConEmu.xml，初始化PentestBox运行的环境变量。

快捷键 Win+Alt+T，进入ConEmu 配置界面。全屏先点右上角"最大化符号"，再按快捷键Alt + Enter。

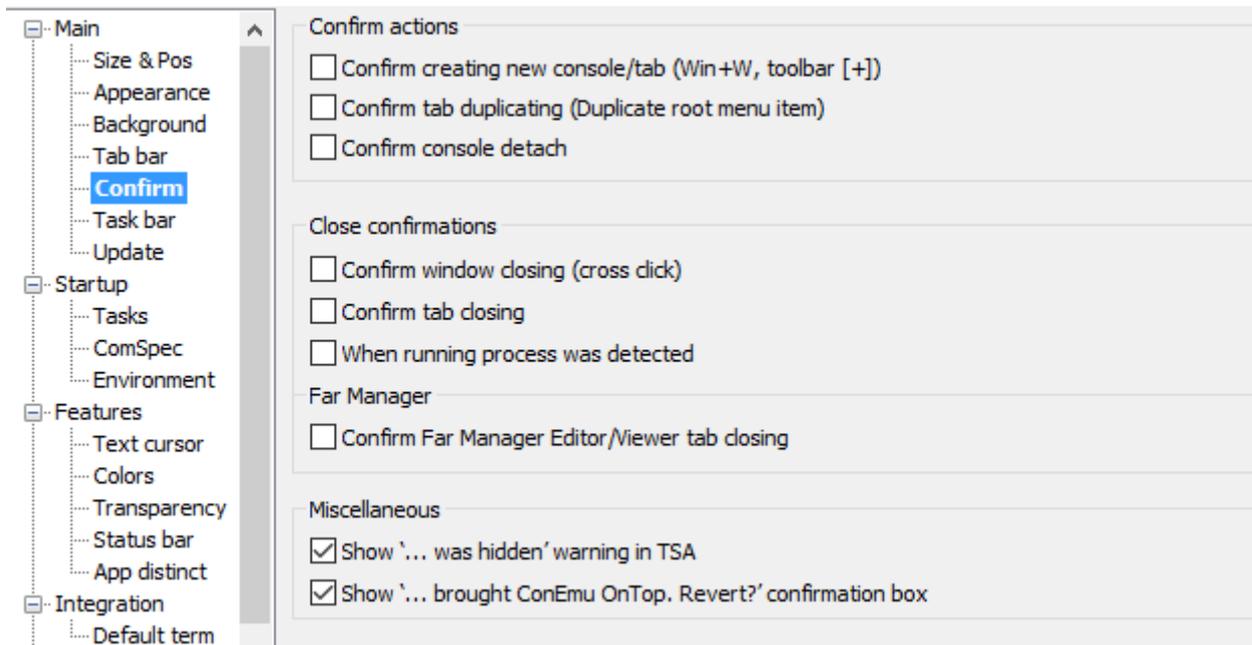
配置背景图片:



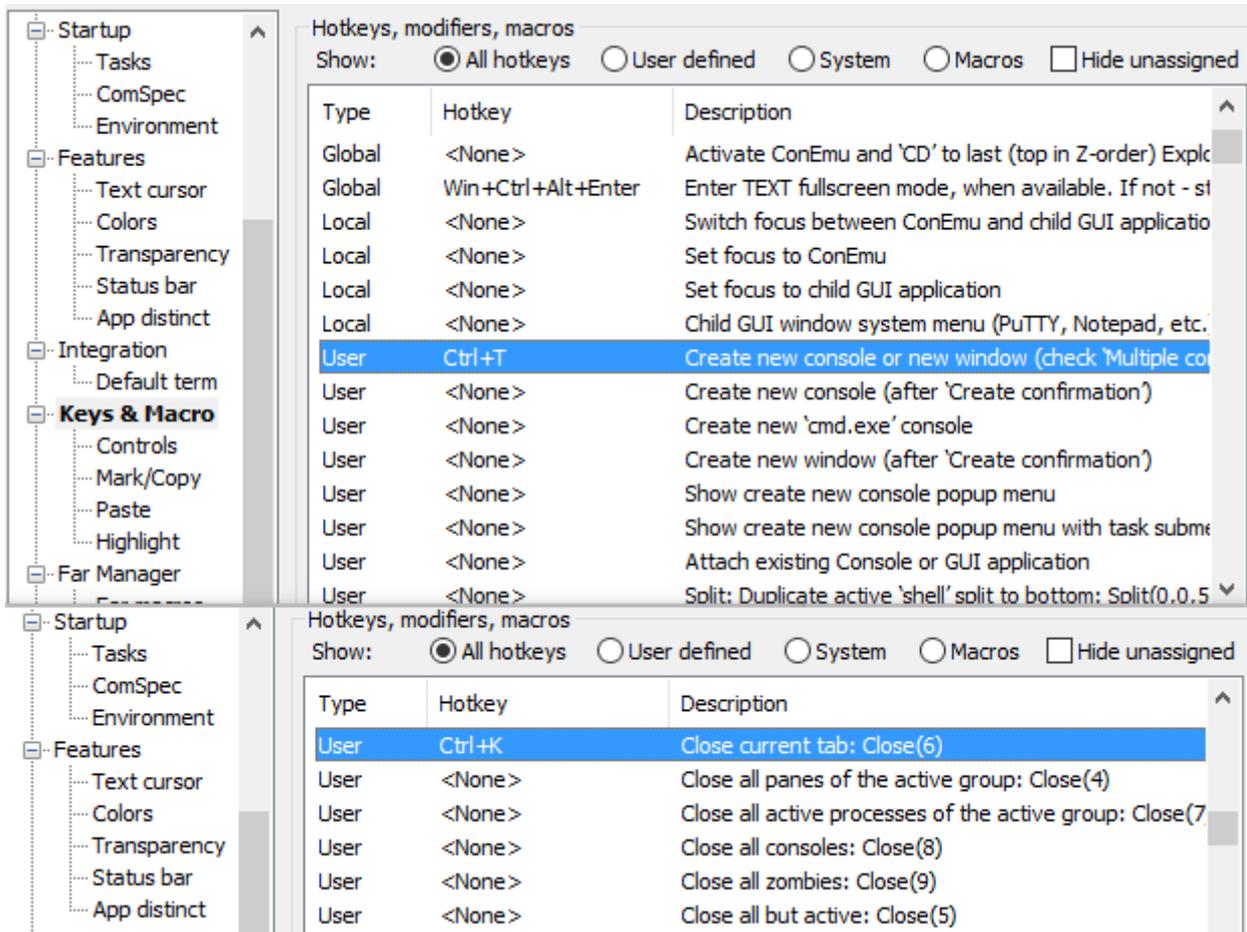
添加一个默认启动窗口:



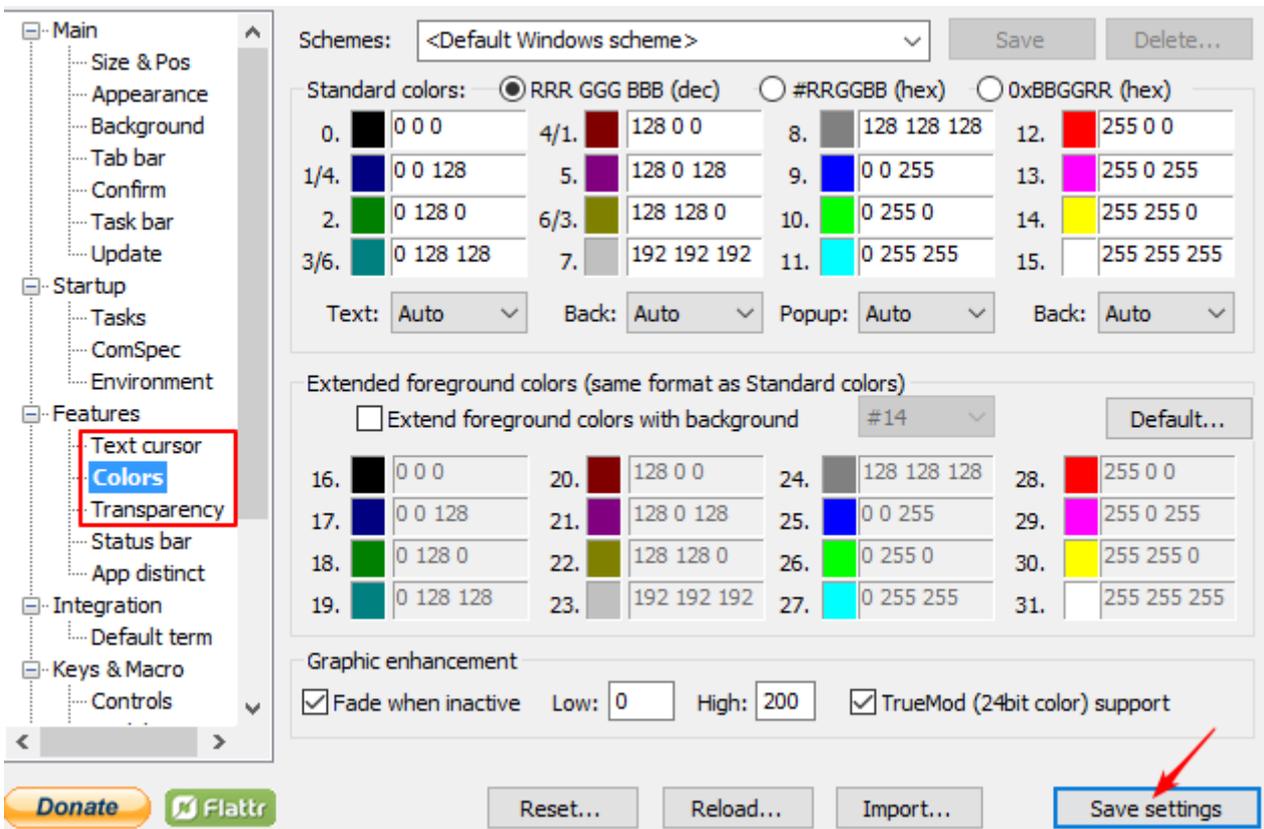
开关窗口时去除确认步骤:



添加创建新窗口、销毁当前窗口热键:



然后设置光标大小、闪烁、形状、scheme方案、背景透明度等，最后别忘了点击保存。



二. 定制toolsmanager

初始时，键入'toolsmanager'命令，每次都会自检有没有更新，然后才进入安装的选择界面，安装较多工具时，比较繁琐，可以在自检过一次后，修改 /pentestbox/bin/scripts/toolsmanager.py 倒数第二行，注销这条语句：

```
# updating_scripts()
```

然后，就可以方便的重复进入toolsmanager，快速多安装几个工具了。

三. 定制python环境

使用PentestBox常遇到的一个问题就是：**本机python 2.7.x环境和PentestBox内置python环境经常冲突，导包时可能既导入了本机Python包又导入了内置python包，导致一些程序运行出错，** 怎么避免这种错误呢？很遗憾，还没找到解决办法，只找到一种缓解方法：**同步两种python环境**

1. 备份内置的 pentestbox/base/python/Lib 目录
2. 删除 pentestbox/base/python
3. 复制本机 python 到 pentestbox/base/python 目录
4. 用第一步备份的文件覆盖掉pentestbox/base/python/Lib目录

如果以后pentestbox 有新安装的工具依赖和包误安装到本机python环境，可以单独拷贝到pentestbox的python环境中。

ps: 环境混乱可能对本机使用pentestbox无影响，但是在其它机器上使用时会出错。

四. 定制默认工具

所谓定制默认工具，其实是手动替换掉一些难更新的内置工具，比如nmap、wireshark、burpsuite。

定制很简单，就是找到 pentestbox/bin 目录下的默认工具目录，先备份一下，再全部删除，安装/复制新版的nmap、wireshark、burpsuite到各自目录，新的工具名和替换前的保持一致。

五. 定制自定义命令

在 pentestbox/bin/customtools 目录下，有一个 customaliases 文件，里面可以存放我们自定的命令。

在目录下创建个 Tools 目录，把自己的工具都放进去，依葫芦画瓢，写自定义的命令即可，修改后，要新建个窗口或重新打开pentestbox才能使用

```
charles="%pentestbox_ROOT%\bin\customtools\Tools\charles\Charles.exe"
cknife=java -jar "%pentestbox_ROOT%\bin\customtools\Tools\cknife\Cknife.jar"
pydictor=python "%pentestbox_ROOT%\bin\customtools\Tools\pydictor\pydictor.py" %*
```

六. 定制默认命令

默认的一些命令存储在 pentestbox/config/aliases 文件中，可以自己修改，比如：

```
beefproject    修改成    beef
burpsuite     修改成    bp
```

七. 定制exploit-database

Kali 下默认带有exploit-database，可以方便的搜索，但是pentestbox中没有，需要自己添加。创建目录

```
pentestbox\bin\customtools\Tools\exploit-database
```

你可以选择下载 **官方的存档漏洞库**，但是需要的只是其中一个脚本，用来下载其余所有漏洞和后期更新。**这里提供下载**，如果下载官方的，需要修改 searchsploit脚本，设置正确的目录

```
gitpath="$ (cd "$(dirname "$0")";pwd)"
csvpath="$ (cd "$(dirname "$0")";pwd)/files.csv"
```

之后将脚本改名如：searchsploit-keeper，不然更新时会被覆盖。将脚本放到创建的目录中，修改 pentestbox/bin/customtools/customaliases 文件，添加一行

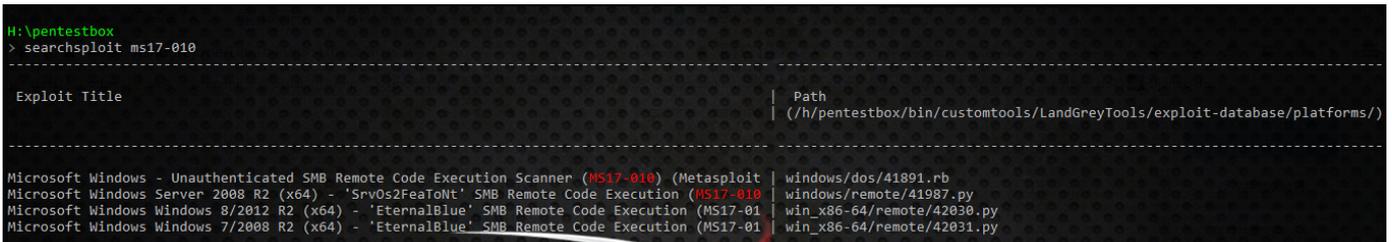
```
searchsploit=sh "%pentestbox_ROOT%\bin\customtools\Tools\exploit-database\searchsploit-keeper" $*
```

之后新建窗口，运行更新命令，首次会拉下Github上所有内容；如果以后有新的内容，就会像下图一样，同步更新

```
searchsploit --update
```



等更新完成后，使用自定义的 searchsploit 命令搜索漏洞



八. 其他

分享个自己常用的trick，例如 保存struts2.bat 脚本到 struts2 目录下：

```
@echo off
SET LandGrey_ROOT= %CD%\bin\customtools\Tools
start %LandGrey_ROOT%\struts
```

自定义命令写

```
struts2="%pentestbox_ROOT%\bin\customtools\Tools\struts2\struts2. bat"
```

就可以用struts2 命令打开 struts2 目录，然后自己再进一步挑选合适的工具。

类似的可以用于打开自己的 webshell 目录，思维导图 目录等。

另外推荐一个工具 GitHubFolderDownloader，可以下载单个Github项目的目录，部分更新时可能用的到。

0x02: 需要注意的坑

一. 备份、备份、备份

整个过程中至少要有两处完整的冗余备份，如本地电脑磁盘中、移动硬盘/其它U盘中；重要的配置文件在修改过多时也要先备份下；如果放心的话，还可以存云盘上一份；不然一旦出现意外情况，就够忙一阵子的了 -> -> 官方文档少，坑可不少。

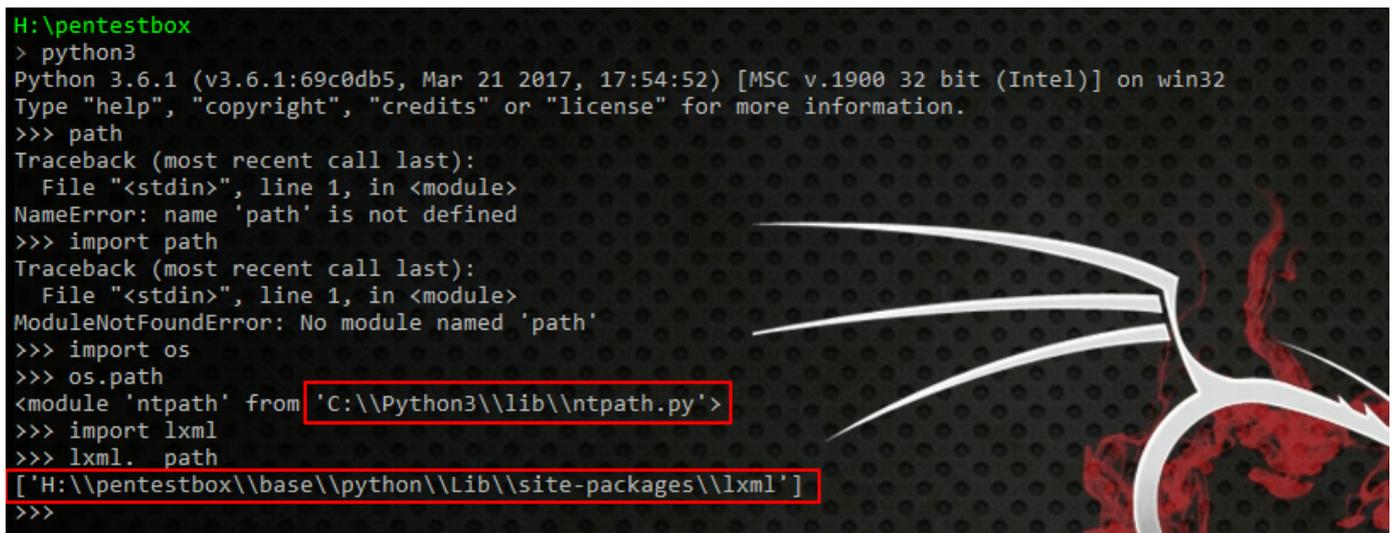
二. 关杀软、开代理

特别是安装带有Metasploit版本的pentestbox，需要关闭主机防护软件，关闭windows防火墙；

需要破墙代理，要不许多请求都会被墙，导致各种错误。

三. 关于自定义环境

比如把python3环境加到pentestbox的默认环境base目录中，建议最好**不要添加**。特别是python 2.x和python 3.x同时存在时，容易引错模块，各种错误，不是改个系统变量就能解决的。



```
H:\pentestbox
> python3
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 17:54:52) [MSC v.1900 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> path
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'path' is not defined
>>> import path
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'path'
>>> import os
>>> os.path
<module 'ntpath' from 'C:\Python3\lib\ntpath.py'>
>>> import lxml
>>> lxml. path
['H:\pentestbox\base\python\Lib\site-packages\lxml']
>>>
```

四. 关于Metasploit 更新

别想了

你可以更新gem源、bundler，但是更新Metasploit的瓶颈在ruby的版本上，而官方明确说明不支持pentestbox中ruby的更新，不然你可以试试爆出类似于下面的错误

```
Gem::InstallError: rex-text requires Ruby version >= 2.2.0.
An error occurred while installing rex-text (0.2.15), and Bundler
cannot continue.
Make sure that `gem install rex-text -v '0.2.15'` succeeds before bundling.
```

```
In Gemfile:
  metasploit-framework was resolved to 4.12.20, which depends on
    rex-arch was resolved to 0.1.8, which depends on
      rex-text
```

但是，如果 msfconsole 出现 "DL is deprecated, please use Fiddle"提示，可以修改 pentestbox/base/ruby/lib/ruby/2.1.0/dl.rb 文件，用"#"号注释掉。

五. 慎用 update

PentestBox 可以用update 命令（在这之前你需要先破墙，然后：）

```
gem sources -u
gem update --system
gem update bundler
```

接着，像用update webapplication命令更新一些工具，但是更新大量工具的话，就容易使原来的工具或本身出错，特别是update all命令，千万别用。

六. 无效的宏定义错误

这个错误比较奇怪，打开一个新窗口就会在第一行出现 "无效的宏定义。"，虽然不影响正常功能，但是强迫症表示很难受。百度"pentest 无效的宏定义。"、google "pentestbox Invalid macro definition"、"pentestbox undefined macro"，什么都没找到。在这之前，我刚用过 update all 命令和其它已经忘了的折腾手段，暂时无法解决，后续有进展会补充在博客上。

疑点一: 自己拿出先前在移动硬盘中备份的pentestbox运行，发现没有这个错误，就把出问题的一份的，所有与环境变量有关的脚本、配置、文件都删除后复制替换了，发现还是一样；

疑点二: 陆续找ConEmu、pentestbox/config 目录下的ConEmu.xml、init.bat、查看了ConEmu 配置的Keys&Macro选项的原因，都排除了；

疑点三: 最后用ConEmu 自带的 Debug log(GUI) 功能，导出debug错误信息，发现竟然只有一个"找不到"提示，原因一栏为空，没有任何有效信息。

感谢 @ Evilmass，在评论中指出了是因为 pentestbox\bin\customtools\customaliases\customaliases 文件或 pentestbox\config\aliases 文件中存在错误配置的命令导致的，认真排查自定义命令即可解决。

七. 更新wpscan数据库

使用wpscan -Update 更新漏洞库，报错

```
[!] Unable to get https://data.wpscan.org/local_vulnerable_files.xml.sha512 (Problem with the SSL CA cert (path?
```

使用curl命令测试

```
curl https://data.wpscan.org/local_vulnerable_files.xml.sha512
```

提示

```
curl: (77) error setting certificate verify locations:
CAfile: C:\PentestBox\base\curl\bin\ca-bundle.crt
CApath: none
```

然后原始的 base/curl/bin/ca-bundle.crt 文件拷贝到

```
C:\PentestBox\base\curl\bin\ca-bundle.crt
```

暂时解决, 最后一步 **国外代理破墙**, 否则可能被墙的莫名其妙: (

OX03:后记

pentestbox 的定制很灵活, 有需要的话可以结合上面提到的, 按需定制; 希望能让后面学习的人少踩些坑, 也希望有人能解决上述提到的错误。

最后提供下自己的 ConEmu.xml **配置文件**.

上述文章来源于原创文章 <https://landgrey.me/customized-awesome-pentestbox/>, 自发布之日起就拥有以下声明:

1. 任何以盈利、吸引流量为目的的企业、机构、公众号等法人, 在没有取得作者同意的情况下, 不得直接转载、抄袭本文, 否则保留追究其法律责任的权利;
2. 任何不以盈利为目的的个人, 转载或引用时, 必须注明作者和出处。

blog comments powered by Disqus

<