



This site is dedicated to sharing information about the practice, ideas, concepts and patterns regarding computer security.

How to securely isolate and execute recon-ng from Kali Linux

Version 0.1, Last Updated: Aug 19

Table of Contents

1. Introduction	1
1.1. recon-ng features	1
1.2. recon-ng uses	1
2. Requirements	3
2.1. Writing Conventions	3
2.2. VirtualBox	3
2.2.1. Clean VirtualBox Networking	3
2.2.2. Add VirtualBox Networking	5
2.3. Vagrant	5
2.4. Kali Linux and Damn Vulnerable Web Application (DVWA)	6
2.4.1. Vagrantfile	6
2.4.2. bootstrap.sh	8
3. Running recon-ng	14
3.1. Launch recon-ng from a terminal	15
3.1.1. Workspaces	16
3.1.2. Help DB	17
3.1.3. Help Marketplace	17
3.1.4. Help Modules	17
3.1.5. Help script	17
3.1.6. Help options	17
3.1.7. Example Enumeration against of DVWA	18
3.1.8. getting more interesting with Profiler	22
4. Conclusion	24
5. Appendix	25

Chapter 1. Introduction

The motivation behind this paper is to explore using the tool `recon-ng` that comes with Kali Linux.

"Recon-ng is a full-featured Web Reconnaissance framework written in Python. Complete with independent modules, database interaction, built in convenience functions, interactive help, and command completion, Recon-ng provides a powerful environment in which open source web-based reconnaissance can be conducted quickly and thoroughly." - Kali

"Recon-ng has a look and feel similar to the Metasploit Framework, reducing the learning curve for leveraging the framework. However, it is quite different. Recon-ng is not intended to compete with existing frameworks, as it is designed exclusively for web-based open source reconnaissance. If you want to exploit, use the Metasploit Framework." - Kali

1.1. recon-ng features

- Recon-ng is free and open source tool this means you can download and use it at free of cost.
- Recon-ng can target a single domain and can find all the subdomains of that domain which makes tasks easy for pentesters.
- Recon-ng can easily find loopholes in the code of web applications and websites.
- Recon-ng can use the Shodan.io search engine to scan IOT devices.
- Recon-ng interface is very similar to metasploitable 1 and metasploitable 2 that makes it easy to use.
- Recon-ng is a complete package of information gathering modules.
- Recon-ng is one of the easiest and useful tool for performing reconnaissance.
- Recon-ng is used for information gathering and vulnerability assessment of web applications.
- Recon-ng works and acts as a web application/website scanner.
- Recon-ng's interactive console provides a number of helpful features.
- Recon-ng has many modules, including:
 - Banner grabbing
 - DNS lookup
 - Geoip lookup
 - port scanning

1.2. recon-ng uses

- Recon-ng is a complete package of Information gathering tools.
- Recon-ng port scanner modules find open ports which can be used to access to target servers.
- Recon-ng can be used:
 - find IP Addresses of target(s)

- find sensitive files such as robots.txt
- look for error based SQL injections
- Recon-ng can be used to find information about:
 - Banner grabbing
 - DNS lookup
 - Geo-IP lookup
 - port scanning
 - reverse IP using WHOIS lookup
 - sub-domain information
- Recon-ng can be used to:
 - Banner grabbing
 - detects Content Management Systems (CMS) in use of a target web application
 - DNS lookup
 - Geo-IP lookup
 - InfoSploit can be used for WHOIS data collection
 - MX records lookup
 - port scanning
 - reverse IP
 - sub-domain information
- Recon-ng subdomain finder modules is used to find subdomains of a primary domain.

Chapter 2. Requirements

2.1. Writing Conventions

If you see the following \$ symbol on a command line to execute, what that means is that the command is executed as a regular user; meaning an account that does not have administrative privileges. Ignore the leading \$ and execute the rest of the command.

```
$ command to execute as a regular user
```

If you see a command line lead with the # symbol, then that means that the command is executed as the root user. This implies you need to elevate to the root user before running the command, e.g. with: `sudo su - root`.

```
# command to execute as the root user
```

2.2. VirtualBox

Go to: <https://www.virtualbox.org/wiki/Downloads> and download VirtualBox.

The author is running on Ubuntu 18.04, so following to this URL: https://www.virtualbox.org/wiki/Linux_Downloads

For Ubuntu, double click on the .deb file, i.e. `virtualbox-5.2_5.2.0-118431-Ubuntu-zesty_amd64.deb`, and install VirtualBox on your local workstation.

2.2.1. Clean VirtualBox Networking

This section is here in case you already had virtualbox installed from before. The intent is to clean up the previous networking. If you do not need to do this, skip to [Add VirtualBox Networking](#)

Run these two commands from a Terminal:

```
$ VBoxManage list natnetworks  
$ VBoxManage list dhcpservers
```

Output (example):

```
NetworkName:    192.168.139-NAT  
IP:             192.168.139.1  
Network:       192.168.139.0/24  
IPv6 Enabled:  No  
IPv6 Prefix:   fd17:625c:f037:2::/64  
DHCP Enabled:  Yes
```

```
Enabled:      Yes
loopback mappings (ipv4)
  127.0.0.1=2
```

```
NetworkName:  192.168.139-NAT
Dhcpd IP:     192.168.139.3
LowerIPAddress: 192.168.139.101
UpperIPAddress: 192.168.139.254
NetworkMask:  255.255.255.0
```

```
Enabled:      Yes
Global Configuration:
  minLeaseTime:  default
  defaultLeaseTime: default
  maxLeaseTime:  default
  Forced options:  None
  Suppressed opts.: None
  1/legacy: 255.255.255.0
```

```
Groups:      None
Individual Configs:  None
```

```
NetworkName:  HostInterfaceNetworking-vboxnet0
Dhcpd IP:     172.20.0.3
LowerIPAddress: 172.20.0.101
UpperIPAddress: 172.20.0.254
NetworkMask:  255.255.255.0
```

```
Enabled:      Yes
Global Configuration:
  minLeaseTime:  default
  defaultLeaseTime: default
  maxLeaseTime:  default
  Forced options:  None
  Suppressed opts.: None
  1/legacy: 255.255.255.0
```

```
Groups:      None
Individual Configs:  None
```

Now, delete ALL of the pre-installed VirtualBox networks (one at a time following the syntax below):

```
VBoxManage natnetwork remove --netname <NetworkName_from_above>
VBoxManage natnetwork remove --netname 192.168.139-NAT
```

Repeat as many times as necessary to delete all of them.

Now, delete ALL of the pre-installed DHCP services:

```
VBoxManage dhcpserver remove --netname <DHCP_Server_NetworkName_from_above>
```

```
VBoxManage dhcpserver remove --netname 192.168.139-NAT
```

Repeat as many times as necessary to delete all of them.

2.2.2. Add VirtualBox Networking

Now, add the new VirtualBox networks so the Kali Linux guides work.

```
VBoxManage natnetwork add \  
  --netname 192.168.139-NAT \  
  --network "192.168.139.0/24" \  
  --enable --dhcp on  
  
VBoxManage dhcpserver add \  
  --netname 192.168.139-NAT \  
  --ip 192.168.139.3 \  
  --lowerip 192.168.139.101 \  
  --upperip 192.168.139.254 \  
  --netmask 255.255.255.0 \  
  --enable  
  
VBoxManage hostonlyif create  
  
VBoxManage hostonlyif ipconfig vboxnet0 \  
  --ip 172.20.0.1 \  
  --netmask 255.255.255.0  
  
VBoxManage dhcpserver add \  
  --ifname vboxnet0 \  
  --ip 172.20.0.3 \  
  --lowerip 172.20.0.101 \  
  --upperip 172.20.0.254 \  
  --netmask 255.255.255.0  
  
VBoxManage dhcpserver modify \  
  --ifname vboxnet0 \  
  --enable
```

VirtualBox install complete.

2.3. Vagrant

Go to: <https://www.vagrantup.com/downloads.html>, follow the appropriate link to your OS and 32 or 64 bit version representing your local workstation. Download.

For Ubuntu, double click on the .deb file, i.e. vagrant_2.0.1_x86_64.deb, and install Vagrant on your local system.

NOTE | Update vagrant vm: [vagrant box update](#)

2.4. Kali Linux and Damn Vulnerable Web Application (DVWA)

The author highly recommends to create a directory structure that is easy to navigate and find your code. As an example, you could use something similar to:

```
`${HOME}/Source_Code/Education/vagrant-machines/kali-linux-vm/
```

Go ahead and make this structure with the following command (inside a Terminal):

```
$ mkdir -p `${HOME}/Source_Code/Education/vagrant-machines/kali-linux-vm/
```

From a Terminal, change directory to:

```
$ cd `${HOME}/Source_Code/Education/vagrant-machines/kali-linux-vm/
```

2.4.1. Vagrantfile

Inside of the kali-linux-vm directory, populate a new file with the exact name, “Vagrantfile”. Case matters, uppercase the “V”. This file will contain both virtual machines for Kali Linux as well as setting up the DVWA virtual machine. Aggregating both virtual machines into one file has saved the author a lot of time. The coolness here is setting up the variables at the top of the Vagrantfile mimicing shell scripting inside of a virtual machine (passed in with provision: shell). I tested using: `apt-get update && apt-get upgrade -y`, but opted to take it out since it took over 45 minutes on my slower (old) hardware. See comment about downloading this file immediately preceding the code block.

```
# -*- mode: ruby -*-
# vi: set ft=ruby :

$os_update = <<SCRIPT
apt-get update
SCRIPT

VAGRANTFILE_API_VERSION = "2"

Vagrant.configure(VAGRANTFILE_API_VERSION) do |config|
  config.vm.define "kali-linux-vagrant" do |conf|
    conf.vm.box = "kalilinux/rolling"

    # For Linux systems with the Wireless network, uncomment the line:
```



```

conf.vm.network "public_network", bridge: "wlo1", auto_config: true

# For macbook/OSx systems, uncomment the line and comment out the Linux
Wireless network:
#conf.vm.network "public_network", bridge: "en0: Wi-Fi (AirPort)",
auto_config: true

conf.vm.hostname = "kali-linux-vagrant"
conf.vm.provider "virtualbox" do |vb|
  vb.gui = true
  vb.memory = "4096"
  vb.cpus = "2"
  vb.customize ["modifyvm", :id, "--vram", "32"]
  vb.customize ["modifyvm", :id, "--accelerate3d", "off"]
  vb.customize ["modifyvm", :id, "--ostype", "Debian_64"]
  vb.customize ["modifyvm", :id, "--boot1", "dvd"]
  vb.customize ["modifyvm", :id, "--boot2", "disk"]
  vb.customize ["modifyvm", :id, "--audio", "none"]
  vb.customize ["modifyvm", :id, "--clipboard", "hosttoguest"]
  vb.customize ["modifyvm", :id, "--draganddrop", "hosttoguest"]
  vb.customize ["modifyvm", :id, "--paravirtprovider", "kvm"]
end
conf.vm.provision "shell", inline: $os_update
end

config.vm.define "dvwa-vagrant" do |conf|

  conf.vm.box = "ubuntu/xenial64"

  conf.vm.hostname = "dvwa-vagrant"

  # For Linux systems with the Wireless network, uncomment the line:
  conf.vm.network "public_network", bridge: "wlo1", auto_config: true

  # For macbook/OSx systems, uncomment the line and comment out the Linux
Wireless network:
#conf.vm.network "public_network", bridge: "en0: Wi-Fi (AirPort)",
auto_config: true

  config.vm.network "forwarded_port", guest: 80, host: 8080, auto_correct: true
  config.vm.network "forwarded_port", guest: 3306, host: 3306, auto_correct:
true

  conf.vm.provider "virtualbox" do |vb|
    vb.memory = "1024"
    vb.cpus = "2"
    vb.gui = false
    vb.customize ["modifyvm", :id, "--vram", "32"]
    vb.customize ["modifyvm", :id, "--accelerate3d", "off"]
    vb.customize ["modifyvm", :id, "--ostype", "Ubuntu_64"]
    vb.customize ["modifyvm", :id, "--boot1", "dvd"]
  end
end

```

```

        vb.customize ["modifyvm", :id, "--boot2", "disk"]
        vb.customize ["modifyvm", :id, "--audio", "none"]
        vb.customize ["modifyvm", :id, "--clipboard", "hosttoguest"]
        vb.customize ["modifyvm", :id, "--draganddrop", "hosttoguest"]
        vb.customize ["modifyvm", :id, "--paravirtprovider", "kvm"]
    end
    conf.vm.provision "shell", inline: $os_update
    conf.vm.provision :shell, path: "bootstrap.sh"
end
end
end

```

Save and write this file.

You can also download from:

```
$ curl -o Vagrantfile http://securityhardening.com/files/Vagrantfile_20200928.txt
```

2.4.2. bootstrap.sh

Inside of the kali-linux-vm directory, populate a new file with the exact name, `bootstrap.sh`. Case matters, all lowercase. See comment about downloading this file immediately preceding the code block. `bootstrap.sh` (include the shebang in your file: the first line with `#!/usr/bin/env bash`):

```

#!/usr/bin/env bash
PHP_FPM_PATH_INI='/etc/php/7.0/fpm/php.ini'
PHP_FPM_POOL_CONF='/etc/php/7.0/fpm/pool.d/www.conf'
MYSQL_ROOT_PW='Assword12345'
MYSQL_dwva_user='dwva'
MYSQL_dwva_password='sunshine'
DVWA_admin_password='admin'
recaptcha_public_key='u8392ihj32k18hujalkshuil32'
recaptcha_private_key='89ry8932873832lih32ilj32'

install_base() {
    add-apt-repository -y ppa:nginx/stable
    sudo apt-get update
    sudo apt-get dist-upgrade -y
    sudo apt-get install -y \
        nginx \
        mariadb-server \
        mariadb-client \
        php \
        php-common \
        php-cgi \
        php-fpm \
        php-gd \
        php-cli \

```

```

    php-pear \
    php-mcrypt \
    php-mysql \
    php-gd \
    git \
    vim
}

config_mysql(){
    mysqladmin -u root password "${MYSQL_ROOT_PW}"
    ## Config the mysql config file for root so it doesn't prompt for password.
    ## Also sets pw in plain text for easy access.
    ## Don't forget to change the password here!!

    cat <<EOF > /root/.my.cnf
    [client]
    user="root"
    password="${MYSQL_ROOT_PW}"
    EOF

    mysql -BNe "drop database if exists dvwa;"
    mysql -BNe "CREATE DATABASE dvwa;"
    mysql -BNe "GRANT ALL ON *.* TO '${MYSQL_dvwa_user}'@'localhost' IDENTIFIED BY
    '${MYSQL_dvwa_password}';"

    systemctl enable mysql
    systemctl restart mysql
    sleep 2
}

config_php(){
    ## Config PHP FPM INI to disable some security settings:

    sed -i 's/^;cgi.fix_pathinfo.*$/cgi.fix_pathinfo = 0/g' ${PHP_FPM_PATH_INI}
    sed -i 's/allow_url_include = Off/allow_url_include = On/g' ${PHP_FPM_PATH_INI}
    sed -i 's/allow_url_fopen = Off/allow_url_fopen = On/g' ${PHP_FPM_PATH_INI}
    sed -i 's/safe_mode = On/safe_mode = Off/g' ${PHP_FPM_PATH_INI}
    echo "magic_quotes_gpc = Off" >> ${PHP_FPM_PATH_INI}
    sed -i 's/display_errors = Off/display_errors = On/g' ${PHP_FPM_PATH_INI}

    ## explicitly set pool options
    ## (these are defaults in ubuntu 16.04 so i'm commenting them out.
    ## If they are not defaults for you try uncommenting these)
    #sed -i 's/^;security.limit_extensions.*$/security.limit_extensions = \
    #.php .php3 .php4 .php5 .php7/g' /etc/php/7.0/fpm/pool.d/www.conf
    #sed -i 's/^listen.owner.*$/listen.owner = www-data/g'
    /etc/php/7.0/fpm/pool.d/www.conf
    #sed -i 's/^listen.group.*$/listen.group = www-data/g'
    /etc/php/7.0/fpm/pool.d/www.conf
    #sed -i 's/^listen.mode.*$/listen.mode = 0660/g' /etc/php/7.0/fpm/pool.d/www.conf

```

```

    systemctl restart php7.0-fpm
}

config_nginx(){

cat << 'EOF' > /etc/nginx/sites-enabled/default
server
{
    listen 80;
    root /var/www/html;
    index index.php index.html index.htm;
    #server_name localhost
    location "/"
    {
        index index.php index.html index.htm;
        #try_files $uri $uri/ =404;
    }

    location ~ \.php$
    {
        include /etc/nginx/fastcgi_params;
        fastcgi_pass unix:/var/run/php/php7.0-fpm.sock;
        fastcgi_index index.php;
        fastcgi_param SCRIPT_FILENAME $request_filename;
    }
}
EOF

    systemctl restart nginx
}

install_dvwa(){

    if [[ ! -d "/var/www/html" ]];
    then
        mkdir -p /var/www;
        ln -s /usr/share/nginx/html /var/www/html;
        chown -R www-data. /var/www/html;
    fi

    cd /var/www/html
    rm -rf /var/www/html/.[!.*]
    rm -rf /var/www/html/*
    git clone https://github.com/ethicalhack3r/DVWA.git ./
    chown -R www-data. ./
    cp config/config.inc.php.dist config/config.inc.php

    ### chmod uploads and log file to be writable by nobody

```

```

chmod 777 ./hackable/uploads/
chmod 777 ./external/phpids/0.6/lib/IDS/tmp/phpids_log.txt

## change the values in the config to match our setup (these are what you need to
update!
sed -i '/db_user/ s/root/'${MYSQL_dvwa_user}'/'
/var/www/html/config/config.inc.php
sed -i '/db_password/ s/p@ssw0rd/'${MYSQL_dvwa_password}'/'
/var/www/html/config/config.inc.php
sed -i "/recaptcha_public_key/ s/'/'"${recaptcha_public_key}"'/"
/var/www/html/config/config.inc.php
sed -i "/recaptcha_private_key/ s/'/'"${recaptcha_private_key}"'/"
/var/www/html/config/config.inc.php

}

update_mysql_user_pws(){
## The mysql passwords are set via /usr/share/nginx/html/dvwa/includes/DBMS/MySQL.php.
# If you edit this every time they are reset it will reset to those.
# Otherwise you can do a sql update statement to update them all (they are just md5's
of the string.
# The issue is the users table doesn't get created until you click that button T_T to
init.

#mysql -Bne "UPDATE dvwa.users SET password = md5('YOUR_MYSQL_PW_HERE') WHERE user =
'admin';"
#mysql -Bne "UPDATE dvwa.users SET password = md5('YOUR_MYSQL_PW_HERE') WHERE user =
'gordonb';"
#mysql -Bne "UPDATE dvwa.users SET password = md5('YOUR_MYSQL_PW_HERE') WHERE user =
'1337';"
#mysql -Bne "UPDATE dvwa.users SET password = md5('YOUR_MYSQL_PW_HERE') WHERE user =
'pablo';"
#mysql -Bne "UPDATE dvwa.users SET password = md5('YOUR_MYSQL_PW_HERE') WHERE user =
'smithy';"

sed -i '/admin/ s/password/'${DVWA_admin_password}'/g'
/var/www/html/dvwa/includes/DBMS/MySQL.php
sed -i '/gordonb/ s/abc123/'${DVWA_admin_password}'/g'
/var/www/html/dvwa/includes/DBMS/MySQL.php
sed -i '/1337/ s/charley/'${DVWA_admin_password}'/g'
/var/www/html/dvwa/includes/DBMS/MySQL.php
sed -i '/pablo/ s/letmein/'${DVWA_admin_password}'/g'
/var/www/html/dvwa/includes/DBMS/MySQL.php
sed -i '/smithy/ s/password/'${DVWA_admin_password}'/g'
/var/www/html/dvwa/includes/DBMS/MySQL.php
}

install_base
config_mysql

```

```
install_dvwa
update_mysql_user_pws
config_php
config_nginx
```

Save and write this file.

If you have issues with copying and pasting the above file because code blocks in PDFs always copy correctly [NOT!], you could use curl, i.e. Make sure the bootstrap.sh file ends up in the same directory as the Vagrantfile.

```
$ curl -o bootstrap.sh http://securityhardening.com/files/bootstrap_sh_20200928.txt
```

From a Terminal, change directory to:

```
$ cd ${HOME}/Source_Code/Education/vagrant-machines/kali-linux-vm/
```

Then run (inside the directory kali-linux-vm):

```
$ vagrant up
```

This will download the appropriate images and start the virtual machines. Once running, through the VirtuaBox GUI, login as root. Password is “toor”, root backwards. Edit the following file: [/etc/ssh/sshd_config](#)

And change the line: `#PermitRootLogin prohibit-password` To: `PermitRootLogin yes` Meaning strip the comment out on the beginning of the line and alter `prohibit-password` to `yes`.

Then restart the ssh daemon:

```
# kill -HUP $(pgrep sshd)
```

Notice, you are on a Bridged adapter, this will open the instance to allow root to ssh in with the most unsecure password in the world. Only make this change (allowing root to login via SSH) if you require root SSH access. You can change the root user’s password, which is highly recommended.

For the DVWA instance, I would first run ‘vagrant status’ to capture the name that vagrant is using for the running instance.

```
# vagrant status
```

Choose

```
Current machine states:
```

```
kali-linux-vagrant running (virtualbox)
dvwa-vagrant running (virtualbox)
```

This environment represents multiple VMs. The VMs are all listed above with their current state. For more information about a specific VM, run `vagrant status NAME`.

From there, log into the DVWA instance with:

```
$ vagrant ssh dvwa-vagrant
```

And then get the current IP address.

```
$ ip a
```

Choose the second network adapter, it should look like:

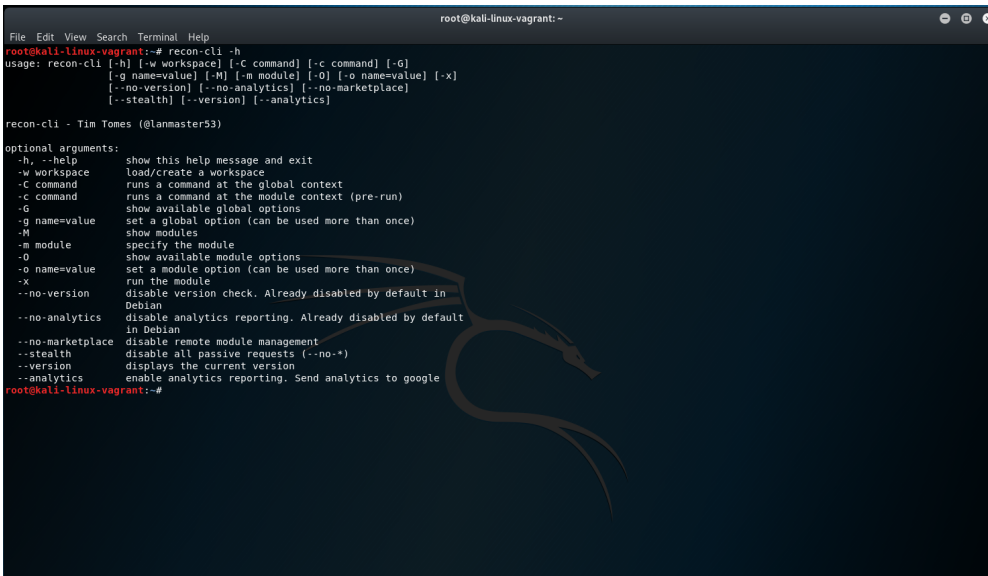
```
ubuntu@dvwa:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen
1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group
default qlen 1000
    link/ether 02:53:17:3c:de:80 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::53:17ff:fe3c:de80/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group
default qlen 1000
    link/ether 08:00:27:f0:77:2d brd ff:ff:ff:ff:ff:ff
    inet 172.20.156.76/24 brd 172.20.156.255 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fef0:772d/64 scope link
        valid_lft forever preferred_lft forever
```

The test network used for this paper uses 172.20.156.0/24 as the network range [shown here in section 3]. Therefore, the adapter, enp0s8 is what he is looking for. The IP to use as a target is 172.20.156.76. Write down your value.

Chapter 3. Running recon-ng

recon-cli

Tool to use recon-ng from the command line



```
root@kali-linux-vagrant:~# recon-cli -h
usage: recon-cli [-h] [-w workspace] [-C command] [-c command] [-G]
               [-g name=value] [-M] [-m module] [-O] [-o name=value] [-x]
               [--no-version] [--no-analytics] [--no-marketplace]
               [--stealth] [--analytics]

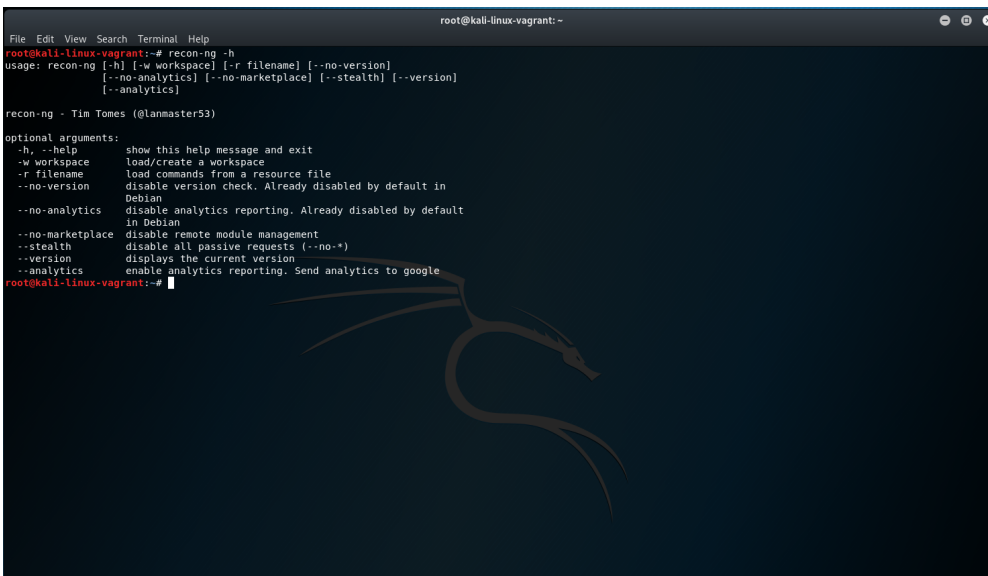
recon-cli - Tim Tomes (@lanmaster53)

optional arguments:
  -h, --help            show this help message and exit
  -w workspace          load/create a workspace
  -C command            runs a command at the global context
  -c command            runs a command at the module context (pre-run)
  -G                   show available global options
  -g name=value        set a global option (can be used more than once)
  -M                   show modules
  -m module            specify the module
  -O                   show available module options
  -o name=value        set a module option (can be used more than once)
  -x                   run the module
  --no-version         disable version check. Already disabled by default in
                      Debian
  --no-analytics      disable analytics reporting. Already disabled by default
                      in Debian
  --no-marketplace    disable remote module management
  --stealth           disable all passive requests (--no-*)
  --version           displays the current version
  --analytics        enable analytics reporting. Send analytics to google

root@kali-linux-vagrant:~#
```

recon-ng

The Web Reconnaissance framework



```
root@kali-linux-vagrant:~# recon-ng -h
usage: recon-ng [-h] [-w workspace] [-r filename] [--no-version]
               [--no-analytics] [--no-marketplace] [--stealth] [--version]
               [--analytics]

recon-ng - Tim Tomes (@lanmaster53)

optional arguments:
  -h, --help            show this help message and exit
  -w workspace          load/create a workspace
  -r filename          load commands from a resource file
  --no-version         disable version check. Already disabled by default in
                      Debian
  --no-analytics      disable analytics reporting. Already disabled by default
                      in Debian
  --no-marketplace    disable remote module management
  --stealth           disable all passive requests (--no-*)
  --version           displays the current version
  --analytics        enable analytics reporting. Send analytics to google

root@kali-linux-vagrant:~#
```

recon-web

The Web-based user interface for Recon-ng


```
File Edit View Search Terminal Help
root@kali:linux-vagrant~# recon-ng -h
usage: recon-ng [-h] [-w workspace] [-r filename] [--no-version]
               [--no-analytics] [--no-marketplace] [--stealth] [--version]
               [--analytics]
recon-ng - Tim Tomes (@lanmaster53)

optional arguments:
  -h, --help            show this help message and exit
  -w workspace          load/create a workspace
  -r filename           load commands from a resource file
  --no-version          disable version check. Already disabled by default in
                       Debian
  --no-analytics       disable analytics reporting. Already disabled by default
                       in Debian
  --no-marketplace     disable remote module management
  --stealth            disable all passive requests (--no-*)
  --version            displays the current version
  --analytics          enable analytics reporting. Send analytics to google
root@kali:linux-vagrant:~#
```

help menu

This is probably the most important menu to know in order to navigate recon-ng.

3.1.1. Workspaces

Add Workspaces

inside of the command line with recon-ng, run:

```
workspaces create dvwa-workspace
workspaces list
workspaces load default
```

Where these workspaces live in your directory structure to be persistable [meaning being able to close recon-ng and re-open and still have your previous work stored in a local structure to be used again].

inside of a different terminal, run:

```
cd ${HOME}/.recon-ng/workspaces
ls -l
```

Delete Workspaces

Inside of the command line with recon-ng, run:

```
workspaces remove dvwa-workspace
workspaces list
```

3.1.2. Help DB

```
[recon-ng][default] > help db
Interfaces with the workspaces database

Usage: db <delete|insert|query|schema> [...]
```

3.1.3. Help Marketplace

```
[recon-ng][default] > help marketplace
Interfaces with the module marketplace

Usage: marketplace <info|install|refresh|remove|search> [...]
```

NOTE | the Marketplace is **remote** stuff

You will most likely want to run a **marketplace search** after reading through the help context. This will help you identify modules that you want to import for penetration testing.

3.1.4. Help Modules

```
[recon-ng][default] > help modules
Interfaces with installed modules

Usage: modules <load|reload|search> [...]
```

NOTE | the Modules is **local** stuff

3.1.5. Help script

```
[recon-ng][default] > help script
Records and executes command scripts

Usage: script <execute|record|status|stop> [...]
```

3.1.6. Help options

```
[recon-ng][default] > help options
Manages the current context options

Usage: options <list|set|unset> [...]
```

example usage:

```
options set VERBOSITY 1
```

3.1.7. Example Enumeration against of DVWA

Knowing that our DVWA is on ip address **172.20.156.146** we will run the following attack sequence

```
marketplace install hackertarget
modules load hackertarget
options set SOURCE 172.20.156.146
input
run
show hosts
```

```
root@kali:~
File Edit View Search Terminal Help
recon/locations-locations/reverse_geocode | 1.0 | not installed | 2019-06-24 | * |
recon/locations-pushpins/flickr | 1.0 | not installed | 2019-06-24 | * |
recon/locations-pushpins/shodan | 1.1 | not installed | 2020-07-07 | * |
recon/locations-pushpins/twitter | 1.1 | not installed | 2019-10-17 | * |
recon/locations-pushpins/youtube | 1.2 | not installed | 2020-09-02 | * |
recon/netblocks-companies/censys_netblock_company | 2.0 | not installed | 2021-05-11 | * |
recon/netblocks-companies/whois_orgs | 1.0 | not installed | 2019-06-24 | * |
recon/netblocks-hosts/censys_netblock | 2.0 | not installed | 2021-05-10 | * |
recon/netblocks-hosts/reverse_resolve | 1.0 | not installed | 2019-06-24 | * |
recon/netblocks-hosts/shodan_net | 1.2 | not installed | 2020-07-21 | * |
recon/netblocks-hosts/virustotal | 1.0 | not installed | 2019-06-24 | * |
recon/netblocks-ports/census_2012 | 1.0 | not installed | 2019-06-24 | * |
recon/netblocks-ports/censysio | 1.0 | not installed | 2019-06-24 | * |
recon/ports-hosts/migrate_ports | 1.0 | not installed | 2019-06-24 | * |
recon/ports-hosts/ssl_scan | 1.1 | not installed | 2021-08-24 | * |
recon/profiles-contacts/bing_linkedin_contacts | 1.2 | not installed | 2021-08-24 | * |
recon/profiles-contacts/dev_diver | 1.1 | not installed | 2020-05-15 | * |
recon/profiles-contacts/github_users | 1.0 | not installed | 2019-06-24 | * |
recon/profiles-profiles/namechk | 1.0 | not installed | 2019-06-24 | * |
recon/profiles-profiles/profiler | 1.0 | not installed | 2019-06-24 | * |
recon/profiles-profiles/twitter_mentioned | 1.0 | not installed | 2019-06-24 | * |
recon/profiles-profiles/twitter_mentions | 1.0 | not installed | 2019-06-24 | * |
recon/profiles-repositories/github_repos | 1.1 | not installed | 2020-05-15 | * |
recon/repositories-profiles/github_commits | 1.0 | not installed | 2019-06-24 | * |
recon/repositories-vulnerabilities/gists_search | 1.0 | not installed | 2019-06-24 | * |
recon/repositories-vulnerabilities/github_dorks | 1.0 | not installed | 2019-06-24 | * |
reporting/csv | 1.0 | not installed | 2019-06-24 | * |
reporting/html | 1.0 | not installed | 2019-06-24 | * |
reporting/json | 1.0 | not installed | 2019-06-24 | * |
reporting/list | 1.0 | not installed | 2019-06-24 | * |
reporting/proxifier | 1.0 | not installed | 2019-06-24 | * |
reporting/pushpin | 1.0 | not installed | 2019-06-24 | * |
reporting/xlsx | 1.0 | not installed | 2019-06-24 | * |
reporting/xml | 1.1 | not installed | 2019-06-24 | * |
-----
D = Has dependencies. See info for details.
K = Requires keys. See info for details.
[recon-ng][default] >
```

Searching the MarketPlace

```
root@kali:~
File Edit View Search Terminal Help
name PushPin Report Generator
author Tim Tomes (@lanmaster53)
version 1.0
last updated 2019-06-24
description Creates HTML media and map reports for all of the PushPins stored in the database.
required_keys ['google_api']
dependencies []
files ['template_media.html', 'template_map.html']
status not installed
-----
path reporting/xlsx
name XLSX File Creator
author Tim Tomes (@lanmaster53)
version 1.0
last updated 2019-06-24
description Creates an Excel compatible XLSX file containing the entire data set.
required_keys []
dependencies []
files []
status not installed
-----
path reporting/xml
name XML Report Generator
author Eric Humphries (@e2fsck) and Tim Tomes (@lanmaster53)
version 1.1
last updated 2019-06-24
description Creates an XML report.
required_keys []
dependencies []
files []
status not installed
-----
[recon-ng][default] >
```

Showing more information about individual modules on the MarketPlace

```
root@kali: ~  
File Edit View Search Terminal Help  
| last updated | 2019-06-24  
| description  | Creates HTML media and map reports for all of the PushPins stored in the database.  
| required_keys | ['google_api']  
| dependencies | []  
| files       | ['template_media.html', 'template_map.html']  
| status      | not installed  
+-----+  
| path        | reporting/xlsx  
| name        | XLSX File Creator  
| author      | Tim Tomes (@lanmaster53)  
| version     | 1.0  
| last updated | 2019-06-24  
| description  | Creates an Excel compatible XLSX file containing the entire data set.  
| required_keys | []  
| dependencies | []  
| files       | []  
| status      | not installed  
+-----+  
| path        | reporting/xml  
| name        | XML Report Generator  
| author      | Eric Humphries (@e2fscck) and Tim Tomes (@lanmaster53)  
| version     | 1.1  
| last updated | 2019-06-24  
| description  | Creates an XML report.  
| required_keys | []  
| dependencies | []  
| files       | []  
| status      | not installed  
+-----+  
[recon-ng][default] > marketplace install hackertarget  
[*] Module installed: recon/domains-hosts/hackertarget  
[*] Reloading modules...  
[recon-ng][default] >
```

Install the modudule **hackertarget** from the MarketPlace

NOTE to install all modules, run: **marketplace install all**

```
root@kali: ~  
File Edit View Search Terminal Help  
| description  | Creates HTML media and map reports for all of the PushPins stored in the database.  
| required_keys | ['google_api']  
| dependencies | []  
| files       | ['template_media.html', 'template_map.html']  
| status      | not installed  
+-----+  
| path        | reporting/xlsx  
| name        | XLSX File Creator  
| author      | Tim Tomes (@lanmaster53)  
| version     | 1.0  
| last updated | 2019-06-24  
| description  | Creates an Excel compatible XLSX file containing the entire data set.  
| required_keys | []  
| dependencies | []  
| files       | []  
| status      | not installed  
+-----+  
| path        | reporting/xml  
| name        | XML Report Generator  
| author      | Eric Humphries (@e2fscck) and Tim Tomes (@lanmaster53)  
| version     | 1.1  
| last updated | 2019-06-24  
| description  | Creates an XML report.  
| required_keys | []  
| dependencies | []  
| files       | []  
| status      | not installed  
+-----+  
[recon-ng][default] > marketplace install hackertarget  
[*] Module installed: recon/domains-hosts/hackertarget  
[*] Reloading modules...  
[recon-ng][default] > modules load hackertarget  
[recon-ng][default][hackertarget] >
```

Load the module **hackertarget**

```
root@kali: ~
File Edit View Search Terminal Help
+-----+
| path      | reporting/xlsx
| name      | XLSX File Creator
| author    | Tim Tomes (@lanmaster53)
| version   | 1.0
| last updated | 2019-06-24
| description | Creates an Excel compatible XLSX file containing the entire data set.
| required keys | []
| dependencies | []
| files     | []
| status    | not installed
+-----+

+-----+
| path      | reporting/xml
| name      | XML Report Generator
| author    | Eric Humphries (@e2fscck) and Tim Tomes (@lanmaster53)
| version   | 1.1
| last updated | 2019-06-24
| description | Creates an XML report.
| required keys | []
| dependencies | []
| files     | []
| status    | not installed
+-----+

[recon-ng][default] > marketplace install hackertarget
[*] Module installed: recon/domains-hosts/hackertarget
[*] Reloading modules...
[recon-ng][default] > modules load hackertarget
[recon-ng][default][hackertarget] > show options
Shows various framework items

Usage: show <companies|contacts|credentials|domains|hosts|leaks|locations|netblocks|ports|profiles|pushpins|repositories|vulnerabilities>

[recon-ng][default][hackertarget] >
```

Show Options available

```
root@kali: ~
File Edit View Search Terminal Help
+-----+
| path      | reporting/xlsx
| name      | XLSX File Creator
| author    | Tim Tomes (@lanmaster53)
| version   | 1.0
| last updated | 2019-06-24
| description | Creates an Excel compatible XLSX file containing the entire data set.
| required keys | []
| dependencies | []
| files     | []
| status    | not installed
+-----+

+-----+
| path      | reporting/xml
| name      | XML Report Generator
| author    | Eric Humphries (@e2fscck) and Tim Tomes (@lanmaster53)
| version   | 1.1
| last updated | 2019-06-24
| description | Creates an XML report.
| required keys | []
| dependencies | []
| files     | []
| status    | not installed
+-----+

[recon-ng][default] > marketplace install hackertarget
[*] Module installed: recon/domains-hosts/hackertarget
[*] Reloading modules...
[recon-ng][default] > modules load hackertarget
[recon-ng][default][hackertarget] > show options
Shows various framework items

Usage: show <companies|contacts|credentials|domains|hosts|leaks|locations|netblocks|ports|profiles|pushpins|repositories|vulnerabilities>

[recon-ng][default][hackertarget] > options set SOURCE 172.20.156.146
SOURCE => 172.20.156.146
[recon-ng][default][hackertarget] >
```

Set options for attack

```
root@kali: ~
File Edit View Search Terminal Help
| last_updated | 2019-06-24
| description | Creates an XML report.
| required keys | []
| dependencies | []
| files | []
| status | not installed
+-----+
[recon-ng][default] > marketplace install hackertarget
[*] Module installed: recon/domains-hosts/hackertarget
[*] Reloading modules...
[recon-ng][default] > modules load hackertarget
[recon-ng][default][hackertarget] > show options
Shows various framework items

Usage: show <companies|contacts|credentials|domains|hosts|leaks|locations|netblocks|ports|profiles|pushpins|repositories|vulnerabilities>

[recon-ng][default][hackertarget] > options set SOURCE 172.20.156.146
SOURCE => 172.20.156.146
[recon-ng][default][hackertarget] > info

Name: HackerTarget Lookup
Author: Michael Henriksen (@michenriksen)
Version: 1.1

Description:
Uses the HackerTarget.com API to find host names. Updates the 'hosts' table with the results.

Options:
-----
Name      Current Value  Required  Description
-----
SOURCE    172.20.156.146  yes       source of input (see 'show info' for details)

Source Options:
default   SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string>  string representing a single input
<path>    path to a file containing a list of inputs
query <sql> database query returning one column of inputs

[recon-ng][default][hackertarget] >
```

module info

```
root@kali: ~
File Edit View Search Terminal Help
[recon-ng][default] > marketplace install hackertarget
[*] Module installed: recon/domains-hosts/hackertarget
[*] Reloading modules...
[recon-ng][default] > modules load hackertarget
[recon-ng][default][hackertarget] > show options
Shows various framework items

Usage: show <companies|contacts|credentials|domains|hosts|leaks|locations|netblocks|ports|profiles|pushpins|repositories|vulnerabilities>

[recon-ng][default][hackertarget] > options set SOURCE 172.20.156.146
SOURCE => 172.20.156.146
[recon-ng][default][hackertarget] > info

Name: HackerTarget Lookup
Author: Michael Henriksen (@michenriksen)
Version: 1.1

Description:
Uses the HackerTarget.com API to find host names. Updates the 'hosts' table with the results.

Options:
-----
Name      Current Value  Required  Description
-----
SOURCE    172.20.156.146  yes       source of input (see 'show info' for details)

Source Options:
default   SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string>  string representing a single input
<path>    path to a file containing a list of inputs
query <sql> database query returning one column of inputs

[recon-ng][default][hackertarget] > input

+-----+
| Module Inputs |
+-----+
| 172.20.156.146 |
+-----+

[recon-ng][default][hackertarget] >
```

input the module

```
root@kali: ~
File Edit View Search Terminal Help
[recon-ng][default][hackertarget] > info

Name: HackerTarget Lookup
Author: Michael Henriksen (@michenriksen)
Version: 1.1

Description:
Uses the HackerTarget.com API to find host names. Updates the 'hosts' table with the results.

Options:
Name      Current Value  Required  Description
-----
SOURCE    172.20.156.146 yes        source of input (see 'show info' for details)

Source Options:
default      SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string>    string representing a single input
<path>      path to a file containing a list of inputs
query <sql>  database query returning one column of inputs

[recon-ng][default][hackertarget] > input
+-----+
| Module Inputs |
+-----+
| 172.20.156.146 |
+-----+

[recon-ng][default][hackertarget] > run
-----
172.20.156.146
-----
[*] [host] 172.20.156.146 (172.20.156.146)
-----
SUMMARY
-----
[*] 1 total (0 new) hosts found.
[recon-ng][default][hackertarget] >
```

run the module

```
root@kali: ~
File Edit View Search Terminal Help
Options:
Name      Current Value  Required  Description
-----
SOURCE    172.20.156.146 yes        source of input (see 'show info' for details)

Source Options:
default      SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string>    string representing a single input
<path>      path to a file containing a list of inputs
query <sql>  database query returning one column of inputs

[recon-ng][default][hackertarget] > input
+-----+
| Module Inputs |
+-----+
| 172.20.156.146 |
+-----+

[recon-ng][default][hackertarget] > run
-----
172.20.156.146
-----
[*] [host] 172.20.156.146 (172.20.156.146)
-----
SUMMARY
-----
[*] 1 total (0 new) hosts found.
[recon-ng][default][hackertarget] > show hosts
+-----+
| rowid | host      | ip_address | region | country | latitude | longitude | module |
+-----+
| 1      | 172.20.156.146 | 172.20.156.146 |      |      |      |      | hackertarget |
+-----+

[*] 1 rows returned
[recon-ng][default][hackertarget] >
```

finally, show hosts

3.1.8. getting more interesting with Profiler

The above to the author is boring.

So the author did a search for **web** with:

Chapter 4. Conclusion

Having run recon-ng against DVWA, we can see the results of the attack. Again, the tool is a framework in which to leverage as many modules as necessary to gain a better understanding of the remote target. The author highly recommends the audience to play with this tool in an isolated sandbox to explore the Marketplace and individual Modules and how they interact with DVWA. From Kali linux, it would interesting to run a tcpdump and record the attack sequence for further analysis. Something like: `tcpdump -i eth1 -w /tmp/attack_run_001.dmp -s 1524 'ip and host 172.20.156.146'` to capture all traffic to and from the host `172.20.156.146`. And then replay with: `tcpdump -r /tmp/attack_run_001.dmp -nvvX | less`.

Chapter 5. Appendix

References

<https://www.kali.org/tools/recon-ng/>

<https://techyrick.com/recon-ng/>

<https://github.com/lanmaster53/recon-ng/wiki/Troubleshooting>