

How Secure is Kali Out of the Box?

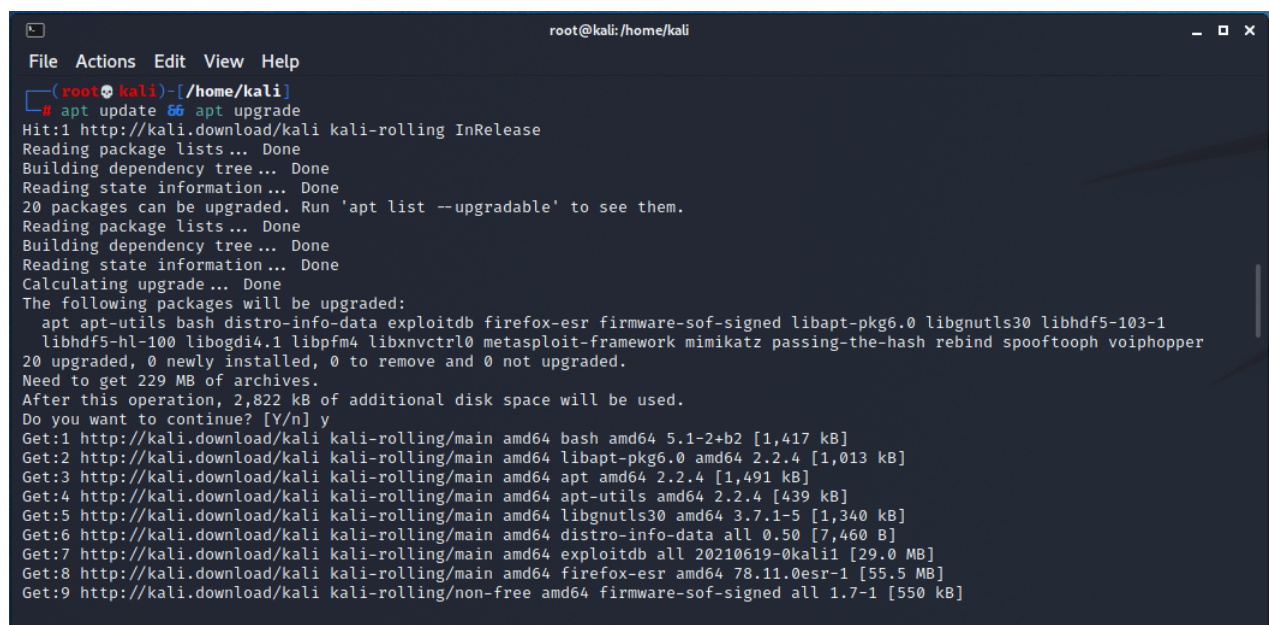
blog.spookysec.net/kali-ootb

26 Jun 2021

I often see people label Kali as dangerous to dual boot and that you shouldn't run Kali as a daily driver, and I often wonder how much truth there is to this. Personally, going into this, I don't think there is much truth. Today we're going to run an experiment to see just how safe Kali is OOTB.

Setup

For this test, we will be using Kali 2021.2 installed on a virtual machine. First we'll issue an `apt update` & `apt upgrade` to update all installed packages on the distribution; this is a command that I believe your typical pentester would run fairly often.



```
root@kali: /home/kali
File Actions Edit View Help
(root@kali)-[/home/kali]
# apt update && apt upgrade
Hit:1 http://kali.download/kali kali-rolling InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
20 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  apt apt-utils bash distro-info-data exploitable firefox-esr firmware-sof-signed libapt-pkg6.0 libgnutls30 libhdf5-103-1
  libhdf5-hl-100 libogdi4.1 libpbfm4 libxnvctrl0 metasploit-framework mimikatz passing-the-hash rebind spooftooph voiphopper
20 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 229 MB of archives.
After this operation, 2,822 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://kali.download/kali kali-rolling/main amd64 bash amd64 5.1-2+b2 [1,417 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 libapt-pkg6.0 amd64 2.2.4 [1,013 kB]
Get:3 http://kali.download/kali kali-rolling/main amd64 apt amd64 2.2.4 [1,491 kB]
Get:4 http://kali.download/kali kali-rolling/main amd64 apt-utils amd64 2.2.4 [439 kB]
Get:5 http://kali.download/kali kali-rolling/main amd64 libgnutls30 amd64 3.7.1-5 [1,340 kB]
Get:6 http://kali.download/kali kali-rolling/main amd64 distro-info-data all 0.50 [7,460 B]
Get:7 http://kali.download/kali kali-rolling/main amd64 exploitable all 20210619-0kali1 [29.0 MB]
Get:8 http://kali.download/kali kali-rolling/main amd64 firefox-esr amd64 78.11.0esr-1 [55.5 MB]
Get:9 http://kali.download/kali kali-rolling/non-free amd64 firmware-sof-signed all 1.7-1 [550 kB]
```

There was a total of 20 packages that needed to be upgraded, for specific analysis, here's the packages:

```
apt apt-utils bash distro-info-data exploitable firefox-esr firmware-sof-signed
libapt-pkg6.0 libgnutls30 libhdf5-103-1 libhdf5-hl-100 libogdi4.1 libpbfm4
libxnvctrl0 metasploit-framework mimikatz passing-the-hash rebind spooftooph
voiphopper
```

To me, it seems like most of these packages would likely be upgraded for security purposes. Let's move on. In terms of the specific Kernel/OS Version, let's take a look:

```
root@kali: /home/kali
File Actions Edit View Help

(kali@kali)-[~]
$ uname -a
Linux kali 5.10.0-kali8-amd64 #1 SMP Debian 5.10.40-1kali1 (2021-05-31) x86_64 GNU/Linux

(kali@kali)-[~]
$ cat /etc/os-release
PRETTY_NAME="Kali GNU/Linux Rolling"
NAME="Kali GNU/Linux"
ID=kali
VERSION="2021.2"
VERSION_ID="2021.2"
VERSION_CODENAME="kali-rolling"
ID_LIKE=debian
ANSI_COLOR="1;31"
HOME_URL="https://www.kali.org/"
SUPPORT_URL="https://forums.kali.org/"
BUG_REPORT_URL="https://bugs.kali.org/"
```

Kernel

The kernel running is 5.10.40-1kali1 which is dated by about 6 months, however, on the next major release, we should expect another minor kernel upgrade. Traditionally, Offensive Security updates the Kali kernel on their quarterly releases. See [here](#) for what Kali Version has which kernel.

In terms of vulnerabilities for version 5.10 of the Linux Kernel, there's about 7 or so, not all of which exist in the final version of the 5.10 of the Kernel.

https://www.cvedetails.com/vulnerability-list/vendor_id-33/product_id-47/version_id-646877/Linux-Linux-Kernel-5.10.html - Release Candidate 1

https://www.cvedetails.com/vulnerability-list/vendor_id-33/product_id-47/version_id-647836/Linux-Linux-Kernel-5.10.html - Release Candidate 2

https://www.cvedetails.com/vulnerability-list/vendor_id-33/product_id-47/version_id-658135/Linux-Linux-Kernel-5.10.html - Release Candidate 4

https://www.cvedetails.com/vulnerability-list/vendor_id-33/product_id-47/version_id-646876/Linux-Linux-Kernel-5.10.html - Full Release

None of these vulnerabilities pose a major risk that effect the Kali kernel, and to be honest, the only high severity vulnerability targeted for 5.10 is actually mislabeled and is meant for 5.9.3:

An issue was discovered in the Linux kernel before 5.9.3. `io_uring` takes a non-refcounted reference to the `files_struct` of the process that submitted a request, causing `execve()` to incorrectly optimize `unshare_fd()`, aka CID-0f2122045b94.

So in terms of Kernel exploits, very little exploits publicly exist.

Running Network Services

In terms of running network services, very little exist. The only recorded service running on my Kali device was DHCP which was communicating with my Router to prevent the DHCP lease from expiring.

```
kali@kali: ~  
File Actions Edit View Help  
❏ kali@kali:~  
❏ $ netstat -tulpn  
(Not all processes could be identified, non-owned process info  
will not be shown, you would have to be root to see it all.)  
Active Internet connections (only servers)  
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name  
  
❏ kali@kali:~  
❏ $ netstat -tunlp  
(Not all processes could be identified, non-owned process info  
will not be shown, you would have to be root to see it all.)  
Active Internet connections (servers and established)  
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name  
udp        0      0 0.0.0.0:53                0.0.0.0:*                 ESTABLISHED -
```

Next up is Cronjobs, a few interesting jobs exist, however, none of them have RWX permissions for any user other than root, so we're all good on that front.

```
kali@kali: ~  
File Actions Edit View Help  
kali@kali: ~ x kali@kali: ~ x  
Cron jobs  
https://book.hacktricks.xyz/linux-unix/privilege-escalation#scheduled-cron-jobs  
/usr/bin/crontab  
incrontab Not Found  
-rw-r--r-- 1 root root 1042 Feb 22 17:43 /etc/crontab  
  
/etc/cron.d:  
total 36  
drwxr-xr-x 2 root root 4096 Jun 26 15:27 .  
drwxr-xr-x 157 root root 12288 Jun 26 15:32 ..  
-rw-r--r-- 1 root root 201 Feb 28 21:24 e2scrub_all  
-rw-r--r-- 1 root root 607 Sep 13 2019 john  
-rw-r--r-- 1 root root 712 May 11 2020 php  
-rw-r--r-- 1 root root 102 Feb 22 17:43 .placeholder  
-rw-r--r-- 1 root root 396 Feb 2 17:46 sysstat  
  
/etc/cron.daily:  
total 60  
drwxr-xr-x 2 root root 4096 Jun 26 15:32 .  
drwxr-xr-x 157 root root 12288 Jun 26 15:32 ..  
-rw-r--r-- 1 root root 539 Aug 8 2020 apache2  
-rw-r--r-- 1 root root 1478 Apr 13 11:53 apt-compat  
-rw-r--r-- 1 root root 157 Dec 13 2017 debtags  
-rw-r--r-- 1 root root 1298 May 18 10:02 dpkg  
-rw-r--r-- 1 root root 377 Feb 28 11:37 logrotate  
-rw-r--r-- 1 root root 1123 Feb 19 05:14 man-db  
-rw-r--r-- 1 root root 628 Dec 2 2020 mlocate  
-rw-r--r-- 1 root root 1403 Sep 23 2020 ntp  
-rw-r--r-- 1 root root 102 Feb 22 17:43 .placeholder  
-rw-r--r-- 1 root root 383 May 6 15:01 samba  
-rw-r--r-- 1 root root 518 Feb 2 17:46 sysstat  
  
/etc/cron.hourly:  
total 20  
drwxr-xr-x 2 root root 4096 Jun 26 15:22 .  
drwxr-xr-x 157 root root 12288 Jun 26 15:32 ..  
-rw-r--r-- 1 root root 102 Feb 22 17:43 .placeholder  
  
/etc/cron.monthly:  
total 24  
drwxr-xr-x 2 root root 4096 Jun 26 15:27 .  
drwxr-xr-x 157 root root 12288 Jun 26 15:32 ..  
-rw-r--r-- 1 root root 102 Feb 22 17:43 .placeholder  
-rw-r--r-- 1 root root 144 Jun 5 2013 rwhod  
  
/etc/cron.weekly:  
total 24
```

Next we've got SUID binaries:

```
SUID - Check easy privesc, exploits and write perms  
https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-and-suid  
strace Not Found  
-rwsr-xr-x 1 root root 63K Feb 7 2020 /usr/bin/passwd → Apple_Mac_OSX(03-2006)/Solaris_8/9(12-2004)/SPARC_8/9/Sun_Solaris_2.3.to.2.5.1(02-1997)  
-rwsr-xr-x 1 root root 44K Feb 7 2020 /usr/bin/newgrp → HP-UX_10.20  
-rwsr-xr-x 1 root root 87K Feb 7 2020 /usr/bin/gpasswd  
-rwsr-xr-x 1 root root 52K Feb 7 2020 /usr/bin/chsh  
-rwsr-xr-x 1 root root 58K Feb 7 2020 /usr/bin/chfn → SuSE_9.3/10  
-rwsr-xr-x 1 root root 395K Jan 6 19:10 /usr/sbin/pppd → Apple_Mac_OSX_10.4.8(05-2007)  
-rwsr-xr-x 1 root root 35K Feb 7 09:38 /usr/bin/umount → BSD/Linux(08-1996)  
-rwsr-xr-x 1 root root 71K Feb 7 09:38 /usr/bin/su  
-rwsr-xr-x 1 root root 55K Feb 7 09:38 /usr/bin/mount → Apple_Mac_OSX(Lion)_Kernel_xnu-1699.32.7_except_xnu-1699.24.8  
-rwsr-xr-x 1 root messagebus 51K Feb 21 09:02 /usr/lib/dbus-1.0/dbus-daemon-launch-helper  
-rwsr-xr-x 1 root root 155K Feb 23 16:23 /usr/bin/nfs-3g → Debian9/8/7/Ubuntu/Gentoo/others/Ubuntu_Server_16.10_and_others(02-2017)  
-rwsr-xr-x 1 root root 15K Feb 25 11:49 /usr/bin/vmware-user-suid-wrapper  
-rwsr-xr-x 1 root root 179K Feb 27 03:28 /usr/bin/sudo → check_if_the_sudo_version_is_vulnerable  
-rwsr-xr-x 1 root root 113K Mar 9 11:17 /usr/sbin/mount.nfs  
-rwsr-xr-x 1 root root 471K Mar 13 04:59 /usr/lib/openssh/ssh-keysign  
-rwsr-xr-x 1 root root 19K Apr 7 22:47 /usr/libexec/polkit-agent-helper-1  
-rwsr-xr-x 1 root root 23K Apr 7 22:47 /usr/bin/okxoc → Linux_4.10_to_5.1.17(CVE-2019-13272)/rhel_6(CVE-2011-1485)  
-rwsr-xr-x 1 root root 15K Apr 13 12:07 /usr/lib/xorg/Xorg.wrap  
-rwsr-xr-x 1 root root 35K Apr 21 08:34 /usr/bin/fusermount3  
-rwsr-xr-x 1 root root 43K May 6 15:24 /usr/sbin/mount.cifs  
-rwsr-xr-x 1 root kismet 143K Jun 2 02:26 /usr/bin/kismet_cap_ubertooth_one  
-rwsr-xr-x 1 root kismet 143K Jun 2 02:26 /usr/bin/kismet_cap_ti_cc_2540  
-rwsr-xr-x 1 root kismet 143K Jun 2 02:26 /usr/bin/kismet_cap_ti_cc_2531  
-rwsr-xr-x 1 root kismet 143K Jun 2 02:26 /usr/bin/kismet_cap_rf_killerbee (Unknown SUID binary)  
-rwsr-xr-x 1 root kismet 143K Jun 2 02:26 /usr/bin/kismet_cap_nrf_kw4iz  
-rwsr-xr-x 1 root kismet 143K Jun 2 02:26 /usr/bin/kismet_cap_nrf_mousejack  
-rwsr-xr-x 1 root kismet 143K Jun 2 02:26 /usr/bin/kismet_cap_nrf_52840 (Unknown SUID binary)  
-rwsr-xr-x 1 root kismet 139K Jun 2 02:26 /usr/bin/kismet_cap_nrf_51822  
-rwsr-xr-x 1 root kismet 211K Jun 2 02:26 /usr/bin/kismet_cap_linux_wifi  
-rwsr-xr-x 1 root kismet 155K Jun 2 02:26 /usr/bin/kismet_cap_linux_bluetooth  
  
This check took 9 seconds
```

Sudo is the interesting one here as recently the Sudoedit Bufferoverflow exploit came out, however, Kali 2021.2 is **not** vulnerable:


```
kali@kali: ~ x  kali@kali: ~ x  root@kali: /etc/cron.daily x  kali@kali: ~ x
(kali@kali)-[~]
$ sudo -V
Sudo version 1.9.5p2
Sudoers policy plugin version 1.9.5p2
Sudoers file grammar version 48
Sudoers I/O plugin version 1.9.5p2
Sudoers audit plugin version 1.9.5p2
(kali@kali)-[~]
$
```

Versions 1.9.5.p1 and lower are vulnerable. The only other big thing that jumps out here is the Kismet binaries, which are advised to be configured as SUID binaries:

<https://www.kismetwireless.net/docs/readme/suid/>

So at first glance, there isn't any huge vulnerabilities that you may be able to leverage to gain root access. The only minor thing is the user you create will be in the Sudoers group, so you should choose a strong password. If you would like to review the LinPEAS output, you can do so [here](#)

Remote Scanning

Pivoting over to my Kali machine that I use for daily ops, lets start with a couple nmap scans, we'll do a Full TCP Connect, Syn Stealth, and a X-mas scan:

```
~ : bash — Konsole
[~]
# nmap -sT -p- 10.10.10.234
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-26 16:17 EDT
Nmap scan report for 10.10.10.234
Host is up (0.00064s latency).
All 65535 scanned ports on 10.10.10.234 are closed
MAC Address: 00:0C:29:7E:B1:9E (VMware)

Nmap done: 1 IP address (1 host up) scanned in 1.54 seconds
[~]
# nmap -sS -p- 10.10.10.234
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-26 16:17 EDT
Nmap scan report for 10.10.10.234
Host is up (0.0019s latency).
All 65535 scanned ports on 10.10.10.234 are closed
MAC Address: 00:0C:29:7E:B1:9E (VMware)

Nmap done: 1 IP address (1 host up) scanned in 1.35 seconds
[~]
# nmap -sX -p- 10.10.10.234
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-26 16:17 EDT
Nmap scan report for 10.10.10.234
Host is up (0.0015s latency).
All 65535 scanned ports on 10.10.10.234 are closed
MAC Address: 00:0C:29:7E:B1:9E (VMware)

Nmap done: 1 IP address (1 host up) scanned in 1.08 seconds
[~]
#
```

No scan seemed to matter here, moving onto UDP:

```
[root@pandorasbox]~#nmap -sU -p- 10.10.10.234
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-26 16:18 EDT
Stats: 2:30:22 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 12.98% done; ETC: 11:36 (16:47:41 remaining)
```

Edit: After I published this, the UDP scan was still running. If an attacker manages to find a vulnerability 2 hours into a UDP port scan, they can have it.

Nessus Essential

Next up, we're going to run a remote scan on our 2021.2 Kali box. The next sections going to contain a number of screenshots that displays the Nessus Scan Config so anyone reading can replicate it.

Scan Type: Advanced Scan

Basic -> General

New Scan / Advanced Scan
[Back to Scan Templates](#)

Settings Credentials Plugins

BASIC

- General
- Schedule
- Notifications

DISCOVERY

ASSESSMENT

REPORT

ADVANCED

Name: Kali-OOTB

Description:

Folder: My Scans

Targets: 10.10.10.234/32

Upload Targets [Add File](#)

Save Cancel

Discovery -> Host Discovery

New Scan / Advanced Scan
[Back to Scan Templates](#)

Settings Credentials Plugins

BASIC

DISCOVERY

- Host Discovery
- Port Scanning
- Service Discovery

ASSESSMENT

REPORT

ADVANCED

Remote Host Ping

Ping the remote host ☐

Fragile Devices

- ☐ Scan Network Printers
- ☐ Scan Novell Netware hosts
- ☐ Scan Operational Technology devices

Wake-on-LAN

List of MAC addresses [Add File](#)

Boot time wait (in minutes) 5

Save Cancel

Discovery -> Port Scanning

The screenshot shows the 'New Scan / Advanced Scan' configuration window in a web application. The left sidebar contains a navigation menu with categories: BASIC, DISCOVERY, ASSESSMENT, REPORT, and ADVANCED. Under DISCOVERY, 'Port Scanning' is selected. The main content area is titled 'Settings' and has tabs for 'Settings', 'Credentials', and 'Plugins'. The 'Settings' tab is active, showing options for 'Ports', 'Local Port Enumerators', and 'Network Port Scanners'. The 'Ports' section includes a checkbox for 'Consider unscanned ports as closed' and a 'Port scan range' dropdown set to 'all'. The 'Local Port Enumerators' section has several checked options: 'SSH (netstat)', 'WMI (netstat)', 'SNMP', 'Only run network port scanners if local port enumeration failed', and 'Verify open TCP ports found by local port enumerators'. The 'Network Port Scanners' section has 'SYN' checked, with options to 'Override automatic firewall detection' (set to 'Use soft detection'), and 'UDP' checked with a warning note. At the bottom, there are 'Save' and 'Cancel' buttons.

Scans Settings

New Scan / Advanced Scan

[Back to Scan Templates](#)

Settings Credentials Plugins

BASIC >

DISCOVERY >

Host Discovery

Port Scanning

Service Discovery

ASSESSMENT >

REPORT >

ADVANCED >

Ports

☐ Consider unscanned ports as closed

Port scan range: all

Local Port Enumerators

☒ SSH (netstat)

☒ WMI (netstat)

☒ SNMP

☒ Only run network port scanners if local port enumeration failed

☒ Verify open TCP ports found by local port enumerators

Network Port Scanners

☒ SYN

☐ Override automatic firewall detection

☒ Use soft detection

☐ Use aggressive detection

☐ Disable detection

☒ UDP

Due to the nature of the protocol, it is generally not possible for a port scanner to tell the difference between open and filtered UDP ports. Enabling the UDP port scanner may dramatically increase the scan time and produce unreliable results. Consider using the netstat or SNMP port enumeration options instead if possible.

Save Cancel

Discovery -> Service Discovery

The screenshot shows the 'New Scan / Advanced Scan' configuration window, specifically the 'Service Discovery' settings. The left sidebar is the same as the previous screenshot, with 'Service Discovery' selected under the DISCOVERY category. The main content area shows the 'General Settings' tab. It includes a checked option 'Probe all ports to find services' with a warning note. Below this is a toggle for 'Search for SSL/TLS/DTLS services' set to 'ON'. There are two dropdown menus: 'Search for SSL/TLS on' set to 'All TCP ports' and 'Search for DTLS on' set to 'None'. A text input field for 'Identify certificates expiring within x days' is set to '60'. There is a checked option 'Enumerate all SSL/TLS ciphers' with a warning note, and an unchecked option 'Enable CRL checking (connects to the Internet)'. At the bottom, there are 'Save' and 'Cancel' buttons.

New Scan / Advanced Scan

[Back to Scan Templates](#)

Settings Credentials Plugins

BASIC >

DISCOVERY >

Host Discovery

Port Scanning

Service Discovery

ASSESSMENT >

REPORT >

ADVANCED >

General Settings

☒ Probe all ports to find services

Attempts to map each open port with the service that is running on that port. Note that in some rare cases, this might disrupt some services and cause unforeseen side effects.

Search for SSL/TLS/DTLS services ON

Search for SSL/TLS on All TCP ports

Search for DTLS on None

Identify certificates expiring within x days 60

☒ Enumerate all SSL/TLS ciphers

When selected, Nessus ignores the list of ciphers advertised by SSL/TLS services, and enumerates them by attempting to establish connections using all possible ciphers.

☐ Enable CRL checking (connects to the Internet)

Save Cancel

Assessment -> General

New Scan / Advanced Scan
[Back to Scan Templates](#)

Settings Credentials Plugins

BASIC >
 DISCOVERY >
 ASSESSMENT >
 REPORT >
 ADVANCED >

General
 Brute Force
 Web Applications
 Windows
 Malware
 Databases

Accuracy

☒ Override normal accuracy

☐ Avoid potential false alarms

☒ Show potential false alarms

☒ Perform thorough tests (may disrupt your network or impact scan speed)

Antivirus

Antivirus definition grace period (in days):

SMTP

Third party domain:
This domain must be outside the range of the site being scanned or the site performing the scan. Otherwise, the test might be aborted by the SMTP server.

From address:

To address:

Report

New Scan / Advanced Scan
[Back to Scan Templates](#)

Settings Credentials Plugins

BASIC >
 DISCOVERY >
 ASSESSMENT >
 REPORT >
 ADVANCED >

Processing

☒ Override normal verbosity

☐ I have limited disk space. Report as little information as possible

☒ Report as much information as possible

☒ Show missing patches that have been superseded

☒ Hide results from plugins initiated as a dependency

Output

☒ Allow users to edit scan results

☐ Designate hosts by their DNS name

☐ Display hosts that respond to ping

☐ Display unreachable hosts

☐ Display Unicode characters
WARNING: This feature may cause issues with compliance checks and custom plugins that encounter ISO-8859-1 encoded output

Save Cancel

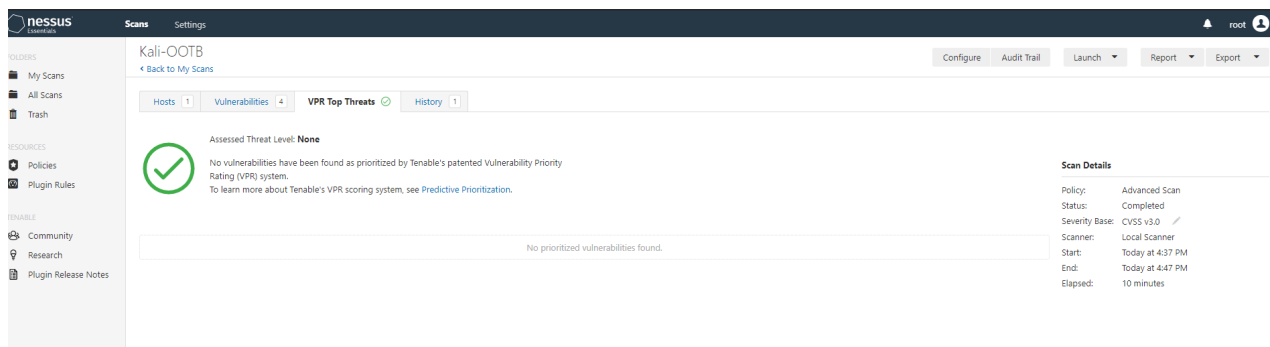
Now that the scan is setup, we can run the scan by pressing “Save” and clicking on the “Play” button.

My Scans Import New Folder New Scan

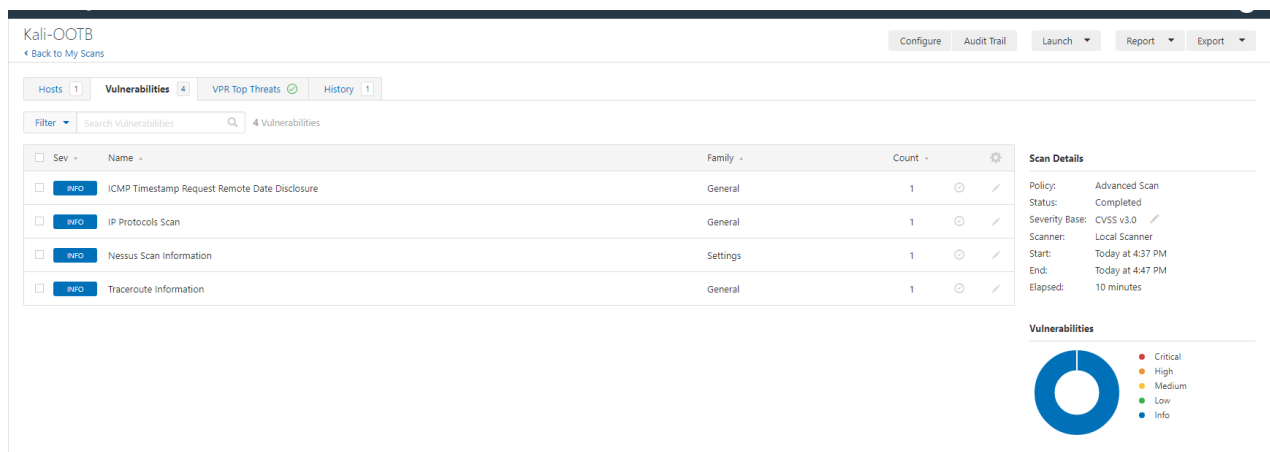
Search Scans 1 Scan

<input type="checkbox"/> Name	Schedule	Last Modified	
<input type="checkbox"/> Kali-OOTB	On Demand	Today at 4:38 PM	

In about 10 minutes, the scan will finish and you will likely receive something pretty similar to what I did – No major vulnerabilities that Tenable advises you patch:



A grand total of 4 Informational level “Vulnerabilities” were identified:



1. Traceroute Information:

For your information, here is the traceroute from 10.10.10.100 to 10.10.10.234 :

```
10.10.10.100
10.10.10.234
```

Hop Count: 1

2. Nessus Scan Information:

Information about this scan :

Nessus version : 8.15.0
Nessus build : 20271
Plugin feed version : 202106260143
Scanner edition used : Nessus Home
Scanner OS : WINDOWS
Scanner distribution : win-x86-64
Scan type : Normal
Scan name : Kali-00TB
Scan policy used : Advanced Scan
Scanner IP : 10.10.10.100
Port scanner(s) : nessus_syn_scanner
Port range : all
Ping RTT : Unavailable
Thorough tests : yes
Experimental tests : no
Paranoia level : 2
Report verbosity : 2
Safe checks : yes
Optimize the test : yes
Credentialed checks : no
Patch management checks : None
Display superseded patches : yes (supersedence plugin launched)
CGI scanning : disabled
Web application tests : disabled
Max hosts : 5
Max checks : 5
Recv timeout : 5
Backports : None
Allow post-scan editing: Yes
Scan Start Date : 2021/6/26 16:37 Eastern Standard Time
Scan duration : 566 sec

3. IP Protocol Scans

The following IP protocols are accepted on this host:

1	ICMP
2	IGMP
6	TCP
17	UDP
103	PIM
136	UDPLite

4. ICMP Timestamp Request Remote Date Disclosure

The difference between the local and remote clocks is -20 seconds.

I wouldn't quite qualify any of these as an actual vulnerability. Now lets try a low privileged internal scan by enabling SSH and giving Nessus a set of credentials. The user account will be named "banana" and will have the password of "banana"

In about 5 minutes, the scan should have finished and we're ready to review the data returned.

Kali-OOTB-Internal
[Back to My Scans](#) Configure Audit Trail Launch Report Export

Hosts 1 Vulnerabilities 30 VPR Top Threats History 1

Assessed Threat Level: High

The following vulnerabilities are ranked by Tenable's patented Vulnerability Priority Rating (VPR) system. The findings listed below detail the top ten vulnerabilities, providing a prioritized view to help guide remediation to effectively reduce risk. Click on each finding to show further details along with the impacted hosts. To learn more about Tenable's VPR scoring system, see [Predictive Prioritization](#).

VPR Severity	Name	Reasons	VPR Score	Hosts
High	Linux Kernel Detection of MDS vulnerabilities (MDSUM/RIDL) (MFBDS/RIDL/ZombieLoad) (MLPDS/RIDL) (MSBDS/Fallout)	No recorded events	7.9	1
Medium	OpenSSH < 8.5	No recorded events	6.7	1

Scan Details

Policy: Advanced Scan
Status: Completed
Severity Base: CVSS v3.0
Scanner: Local Scanner
Start: Today at 5:00 PM
End: Today at 5:06 PM
Elapsed: 5 minutes

There were a grand total of two major vulnerabilities found.

One of which is a misreport – According to [this article](#) (Referenced by Nessus) we can manually validate this finding as a privileged user by checking `/sys/devices/system/cpu/vulnerabilities/mds`

```
(root@kali)-[/home/kali]
# cat /sys/devices/system/cpu/vulnerabilities/mds
Not affected
```

And we can see that our device is not vulnerable. This leaves us with one major vulnerability left; an outdated version of OpenSSH. The reason Nessus flags this as a vulnerability is because there is a social engineering attack within OpenSSH called “Double Free” which requires a user to connect to a malicious SSH server where an attacker has root privileges; the attacker can then send specially crafted data to gain code execution. This would not be an easy vulnerability for an attacker to exploit as no public Proof of Concepts exist meaning that if an adversary would like to exploit this vulnerability, they’ll be building it from the ground up.

Src: <https://www.cybersecurity-help.cz/vulnerabilities/51444/>

28 Other Informational Level vulnerabilities were found, most of the information reported was related to valid credentials, SSH Configuration, OS versions, etc.

Note: I also ran a credentialed scan against the root user account. The MDS vulnerability was no longer present, all of the same issues found in the original credentialed scan were also present. The CSV is available for download [here](#)

Conclusion

To me, the phrase “Kali is insecure” is a very uninformed statement. It’s like saying “Ubuntu Server is insecure”. Sure, in the real world you might take actions to harden Ubuntu Server, just like you might choose to install and deploy UFW on Kali.

Based on the tests preformed today, as long as you keep your Kali distribution up to date and ensure the Kernel and public facing services are all up to date, I see no reason why Kali could not run on bare metal.

As long as you, the user, is educated enough in System Administration to know what starting a new service or hosting an application on your Kali machine might introduce (vulnerability wise), there is absolutely no reason you need to worry about your Kali machine.

My professional advise is this: Use a strong password (Not Kali:Kali), don't keep that password in any text files on your Kali machine, and don't store any sensitive data (long term) on your Kali machine and you will be perfectly fine.

Happy Hacking everyone!

Comments
