Python Pentest Cheat Sheet

Simple GET request for html source

```
import httplib
connection =
httplib.HTTPConnection("xyz_website.com")
connection.request("GET", "/index.html")
response = connection.getresponse()
relevant_payload = response.read()
print(relevant_payload)
```

2Get http response headers

```
import urllib2
url = 'http://xyz_website.com'
headers = { 'User-Agent' : 'Mozilla/5.0 (Windows NT
6.3; WOW64)' }
request = urllib2.Request(url, None, headers)
response = urllib2.urlopen(request)
headers = response.headers
print(headers)
```

Capture cookies generation/possibly session IDs

```
from anonBrowser import *
#import anonBrowser

ab = anonBrowser(proxies=[],\
     user_agents=[('User-agent','My browser')])

for attempt in range(1, 10 ):
    ab.anonymize()
    print '[*] Fetching page'
    response = ab.open('http://google.com')
    for cookie in ab.cookie_jar:
        print cookieprint(headers)
```

Testing for anonymous FTP login

```
import ftplib
def testAnonymousLogin(hostname):
    try:
        ftp = ftplib.FTP(hostname)
        ftp.login('anonymous', 'xyz@gmail.com')
        print '\n[*] ' + str(hostname) +\
          ' FTP Anonymous Logon Succeeded.'
        ftp.quit()
        return True
    except Exception, e:
        print '\n[-] ' + str(hostname) +\
          ' FTP Anonymous Logon Failed!'
        return False
host = 'xyz_website.com'
testAnonymousLogin(host)
```

Nmap scan using python

```
import nmap
import optparse
def nmapScan(tgtHost,tgtPort):
    nmScan = nmap.PortScanner()
    nmScan.scan(tgtHost,tgtPort)
    state=nmScan[tgtHost]['tcp'][int(tgtPort)]['state']
    print "[*] " + tgtHost + " tcp/"+tgtPort +" "+state
def main():
    parser = optparse.OptionParser('usage %prog '+\
                                   '-H <target
host> -p <target port>')
    parser.add_option('-H', dest='tgtHost',
type='string',\
                      help='specify target host')
    parser.add_option('-p', dest='tgtPort',
type='string',\
                      help='specify target port[s]
separated by comma')
        (options, args) = parser.parse_args()
        tgtHost = options.tgtHost
    tgtPorts = str(options.tgtPort).split(',')
    if (tgtHost == None) | (tgtPorts[0] == None):
        print parser.usage
        exit(0)
    for tgtPort in tgtPorts:
        nmapScan(tgtHost, tgtPort)
if __name__ == '__main__':
    main()
```

Site recon – scraping links from target

```
from anonBrowser import *
from BeautifulSoup import BeautifulSoup
import os
import optparse
import re
def printLinks(url):
    ab = anonBrowser()
    ab.anonymize()
    page = ab.open(url)
    html = page.read()
    try:
        print '[+] Printing Links From  Regex.'
        link_finder = re.compile('href="(.*?)"')
        links = link_finder.findall(html)
        for link in links:
            print link
    except:
        pass
    try:
        print '\n[+] Printing Links From BeautifulSoup.'
        soup = BeautifulSoup(html)
        links = soup.findAll(name='a')
        for link in links:
            if link.has_key('href'):
                print link['href']
    except:
        pas
def main():
    parser = optparse.OptionParser('usage %prog ' +\
      '-u <target url including protocol>')
    parser.add_option('-u', dest='tgtURL',
type='string',\
      help='specify target url')
    (options, args) = parser.parse_args()
    url = options.tgtURL
    if url == None:
        print parser.usage
        exit(0)
    else:
        printLinks(url)
if __name__ == '__main__':
    main()
```

Common Ports Reference

7	Echo	902	Vmware Server	5500	VNC Server
19	Chargen	989-990	FTP over SSL	5554	Sasser
20-21	FTP	993	IMAP4 over SSL	5631-5632	pcAnywhere
22	SSH/SCP	995	POP3 over SSL	5800	VNC over HTTP
23	Telnet	1025	Microsoft RPC	5900+	VNC Server
25	SMTP	1026-1029	Windows Messenger	6000-6001	X11
42	WINS Replication	1080	SOCKS Proxy	6112	Battle.net
43	WHOIS	1080	MyDoom	6129	DameWare
49	TACACS	1194	OpenVPN	6257	WinMX
53	DNS	1214	Kazaa	6346-6347	Gnutella
67-68	DHCP/BOOTP	1241	Nessus	6500	GameSpy Arcade
69	TFTP	1311	Dell OpenManage	6566	SANE
70	Gopher	1337	WASTE	6588	AnalogX
79	Finger	1433-1434	Microsoft SQL	6665-6669	IRC
80	HTTP	1512	WINS	6679/6697	IRC over SSL
88	Kerberos	1589	Cisco VQP	6699	Napster
102	MS Exchange	1701	L2TP	6881-6999	BitTorrent
110	POP3	1723	MS PPTP	6891-6901	Windows Live
113	Ident	1725	Steam	6970	Quicktime
119	NNTP (Usenet)	1741	CiscoWorks 2000	7212	GhostSurf
123	NTP	1755	MS Media Server	7648-7649	CU-SeeMe
135	Microsoft RPC	1812-1813	RADIUS	8000	Internet Radio
137-139	NetBIOS	1863	MSN	8080	HTTP Proxy
143	IMAP4	1985	Cisco HSRP	8086-8087	Kaspersky AV
161-162	SNMP	2000	Cisco SCCP	8118	Privoxy
177	XDMCP	2002	Cisco ACS	8200	Vmware Server
179	BGP	2049	NFS	8500	Adobe ColdFusion
201	AppleTalk	2082-2083	cPanel	8767	TeamSpeak
264	BGMP	2100	Oracle XDB	8866	Bagle.B
318	TSP	2222	DirectAdmin	9100	HP JetDirect
381-383	HP Openview	2302	Halo	9101-9103	Bacula
389	LDAP	2483-2484	Oracle DB	9119	MXit
411-412	Direct Connect	2745	Bagle.H	9800	WebDAV
443	HTTP over SSL	2967	Symantec AV	9898	Dabber
445	Microsoft DS	3050	Interbase DB	9988	Rbot/Spybot
464	Kerberos	3074	XBOX Live	9999	Urchin
465	SMTP over SSL	3124	HTTP Proxy	10000	Webmin
497	Retrospect	3127	MyDoom	10000	BackupExec
500	ISAKMP	3128	HTTP	10113-10116	NetIQ
512	rexec	3222	GLBP	11371	OpenPGP
513	rlogin	3260	iSCSI Target	12035-12036	Second Life
514	syslog	3306	MySQL	12345	NetBus
515	LPD/LPR	3389	Terminal Server	13720-13721	NetBackup
520	RIP	3689	iTunes	14567	Battlefield
521	RIPng (IPv6)	3690	Subversion	15118	Dipnet/Oddbob
540	UUCP	3724	World of Warcraft	19226	AdminSecure
554	RTSP	3784-3785	Ventrilo	19638	Ensim
546-547	DHCPv6	4333	mSQL	20000	Usermin
560	rmonitor	4444	Blaster	24800	Synergy
563	NNTP over SSL	4664	Google Destop	25999	Xfire
587	SMTP	4672	eMule	27015	Half-Life
591	FileMaker	4899	Radmin	27374	Sub7
593	Microsoft DCOM	5000	UPnP	28960	Call of Duty
631	Internet Printing	5001	Slingbox	31337	Back Orifice
636	LDAP over SSL	5001	iperf	33434+	traceroute
639	MSDP (PIM)	5004-5005	RTP	5432	PostgreSQL
646	LDP (MPLS)	5050	Yahoo! Messenger	873	rsync
691	MS Exchange	5060	SIP	5222-5223	XMPP/Jabber
860	iSCSI	5190	AIM/ICQ		

PDF version is available at **LIFARS**.com

LIFARS is a digital forensics and cybersecurity intelligence firm based in New York City. Our incident response and penetration testing teams consist of the top experts in the field. As a testament to our excellence, LIFARS was ranked the #2 cybersecurity company in New York Metro area on the Cybersecurity 500 list of the hottest and most innovative cyber security companies.

WARNING: Only scan hosts and networks that you own or have permission to scan! Don't be evil. LIFARS LLC is not responsible for misuse of information provided in this document.

LIFARS
your digital world, **secured**