mainframe [z/OS] reverse engineering and exploit development

Chad Rikansrud
Director, North America
RSM Partners
about me
i used to
but now i
and teach mainframe hacking
so pretty much i
hack gibsons for a living
at mainframe security hq
Mainframe Experts
• Pentesting
• Assessments
• Software
• Red Team Augmentation
the machine

architecture
what most people think
IDENTIFICATION DIVISION.

PROGRAM-ID. QUASAR.

ENVIRONMENT DIVISION.

CONFIGURATION SECTION.

SOURCE-COMPUTER. DELL.

OBJECT-COMPUTER. DELL.

INPUT-OUTPUT SECTION.

DATA DIVISION.

WORKING-STORAGE SECTION.

01 EMPLOYEE-RECORD.

02 EMP-NAME.

03 EMP-FNAME PICT X (10) VALUE 'QUASAR'.
what media thinks
what it really is
it’s important
how important?

- $8$ Trillion (4 commas) GDP: U.K. + France + India + Brazil
- $919$ ATM transactions/second - $158$/second
- $7,610$ Passenger flights/minute
- $347,222$ Total transactions/second – $8.5x > Google$
- It’s important
an analogy
today is full stack / devops
mainframe style
z/architecture and z/os terms

just the basics
not going into

• CICS
• TSO/e
• Datasets
• ESM (RACF, TSS, ACF/2)
• see loads of other talks, presentations and content by:
  • myself
  • @mainframed767
  • @ayou13__
changing cpu state

MODESET -> SVC107 -> LCTL CR03 -> 00C0

problem (subset of instructions)
supervisor (all the instructions)
PSW mode and storage key protection

• supervisor vs problem state
  • PSW – program status word (summary of system flags, settings, EIP)
  • basically - some vs all CPU instructions
changing access storage key

non-zero
(r/w limited to same key)

00
(r/w all the memory)

MODESET -> SVC107 -> LCTL CR03 -> 00C0
PSW mode and storage key protection

• supervisor vs problem state
  • PSW – program status word
  • basically - some vs all CPU instructions

• storage (memory) key
  • 0-15 – PSW current storage key
  • PSW key must match (or be 0) storage key
how it works in z/os

• system startup processes (IPL)
  • supervisor by design

• SVC / PC (privileged system calls)
  • SVC – supervisor call
  • PC – program call

• APF authorized library list
  • static and dynamic list of libraries (folders)
authorized program facility list (apf)

- SYS1.LINKLIB
- SYS1.LPALIB
- USER.LIBRARY1
- PGM.LIBRARY1
- PGM.LIBRARY2

if you can edit this list, or update one of these libraries:

   game over
vulnerabilities

some unique, some familiar
untrusted parameters

source parm address: 0x81FF3C0 KEY 8

dest return address: 0x8FF3F03 KEY 0

poorly written SVC or PC

read or write w/o using source or dest key
buffer overflows

input data

program read w/o bounds checking

allocated pgm storage

return address

jump address

error routine

critical data

overflowed storage
intentional backdoors

program parameters
- actionType
- secretCode
- jumpAddress
- saveArea

input
- check sekret
- good sekret?
- no
- console error

yes
the tools

bad, badder, baddest, really quite good
DBX

like GDB, but not nearly as fun
debug tool
really just here for the colors
<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>1 R0</td>
<td>X'1ED2D0B0'</td>
</tr>
<tr>
<td>0002</td>
<td>2 R1</td>
<td>X'1EB005DC'</td>
</tr>
<tr>
<td>0003</td>
<td>3 R15</td>
<td>X'00000000'</td>
</tr>
</tbody>
</table>

**Source:** MYMXPW0

---

**History:**
ASMIDF

after hella modifications, can be somewhat useful
Program Source and Disassembly

(MODESET) MODESET MODESET CSECT
1EE00B68 90EC D00C STM R14,R12,MODESET+72
1EE00B6C 0F00 FFFF FFFE LARL R15,*-4
1EE00B72 188F LR R8,R15
1EE00B74 0F00 0000 0018 LARL R11,*+48
1EE00B7A 50D0 B004 ST R13,MODESET+64
1EE00B7E 18DB LR R13,R11
1EE00B80 4510 8020 BAL R1,MODESET+32
1EE00B84 0000003C | .... | .... |
1EE00B88 5810 1000 L R1,0,(R1)
1EE00B8C 0A6B SVC 107 MODESET
1EE00B8E 58D0 B004 L R13,MODESET+64
1EE00B92 98EC D00C LM R14,R12,MODESET+72
1EE00B96 C050 0000 0007 LARL R5,*+14
1EE00B9C 58F0 5000 L R15,0,(R5)
1EE00BA0 07FE BCR 15,R14
1EE00BA2 0000 000000 000D2690 00000000 | ......... | ......... |
1EE00BB0 00000000 00000000 00000000 00000000 | ......... | ......... |
1EE00BC0 00000000 00000000 00000000 00000000 | ......... | ......... |
1EE00BD0 <Data to end of memory> ???

Current Registers

(MODESET) MODESET+36 PSW 078D00009EE00B8C (CC mask=8 E)
R0 FF0E000F R4 FF0E040F R8 1EE00B68 R12 9EE00B68 FPR0 0000000000000000
R1 0000003C R5 FF0E050F R9 FF0E090F R13 1EE00B4 A FPR2 0000000000000000
R2 FF0E020F R6 FF0E060F R10 FF0E0A0F R14 00268E6 FPR4 0000000000000000
R3 FF0E030F R7 FF0E070F R11 1EE00B4 A R15 1EE00B68 FPR6 0000000000000000

-->

ONLINE-SSL TRMLU001 31,6
TSO/e TEST

learn it for the same reason you learned ‘ed’
TESTAUTH
LIST 1EB04038. I LENGTH(12)
  1EB04038. L R1,0(,R1)
  1EB0403C. SVC 107
  1EB0403E. L R13,4(,R11)
  1EB04042. LM R14,R12,12(R13)
TESTAUTH
AT 1EB0403E.
TESTAUTH
LISTPSW
IKJ57652I PSW LOCATED AT 8DD168
  XRRXXXTIE KEY XMWP AS CC PROGMASK EA BA BA INSTR ADDR
  000000111  8  1101 00 01 0000 0 1 1EB04018
TESTAUTH
GO
IKJ57024I AT 1EB0403E.
TESTAUTH
LISTPSW
IKJ57652I PSW LOCATED AT 8DD168
  XRRXXXTIE KEY XMWP AS CC PROGMASK EA BA BA INSTR ADDR
  000000111  0  1100 00 01 0000 0 1 1EB0403E
z/XDC
the real contender (non-IBM)
<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000000</td>
<td>EEE2E4A0</td>
<td>0</td>
<td>(A.S.CHAD)</td>
</tr>
<tr>
<td>+C8</td>
<td>8880</td>
<td>SRL</td>
<td>R8, X'003'</td>
</tr>
<tr>
<td>+CC</td>
<td>4888</td>
<td>LH</td>
<td>R8, X'124'(R8,R6)</td>
</tr>
<tr>
<td>+D0</td>
<td>9108</td>
<td>TM</td>
<td>X'178'(R3), B'00001000'</td>
</tr>
<tr>
<td>+D4</td>
<td>4780</td>
<td>BZ</td>
<td>X'0DC'(,R6)</td>
</tr>
<tr>
<td>+D8</td>
<td>5680</td>
<td>O</td>
<td>R8, X'120'(,R6)</td>
</tr>
<tr>
<td>+DC</td>
<td>9104</td>
<td>TM</td>
<td>X'01F'(R4), B'00001000'</td>
</tr>
<tr>
<td>+E0</td>
<td>4780</td>
<td>BZ</td>
<td>X'0EE'(,R6)</td>
</tr>
<tr>
<td>+E4</td>
<td>5820</td>
<td>L</td>
<td>R2, X'138'(,R4)</td>
</tr>
<tr>
<td>+E8</td>
<td>48F0</td>
<td>LH</td>
<td>R15, X'19E'(,R2)</td>
</tr>
<tr>
<td>+EC</td>
<td>168F</td>
<td>OR</td>
<td>R8, R15</td>
</tr>
<tr>
<td>+EE</td>
<td>B633</td>
<td>STC</td>
<td>CR3, CR3, X'8F8'</td>
</tr>
<tr>
<td>+F2</td>
<td>4080</td>
<td>STH</td>
<td>R8, X'8F8'</td>
</tr>
<tr>
<td>+F6</td>
<td>B733</td>
<td>LCTRL</td>
<td>CR3, CR3, X'8F8'</td>
</tr>
<tr>
<td>+FA</td>
<td>5820</td>
<td>L</td>
<td>R2, X'0D8'(,R4)</td>
</tr>
<tr>
<td>+FE</td>
<td>4080</td>
<td>STH</td>
<td>R8, X'0CC'(,R2)</td>
</tr>
<tr>
<td>+102</td>
<td>1BFF</td>
<td>SR</td>
<td>R15, R15</td>
</tr>
<tr>
<td>+104</td>
<td>07FE</td>
<td>BR</td>
<td>R14</td>
</tr>
<tr>
<td>+106</td>
<td>4110</td>
<td>LA</td>
<td>R1, X'16B'</td>
</tr>
<tr>
<td>+10A</td>
<td>8910</td>
<td>SLL</td>
<td>R1, X'00C'</td>
</tr>
<tr>
<td>+10E</td>
<td>18F9</td>
<td>LR</td>
<td>R15, R9</td>
</tr>
<tr>
<td>+110</td>
<td>4100</td>
<td>LA</td>
<td>R0, X'084'</td>
</tr>
<tr>
<td>+114</td>
<td>8900</td>
<td>SLL</td>
<td>R0, X'018'</td>
</tr>
</tbody>
</table>

XDC ==> L PSW ; L REGS

- PSW 078D1000 EEE2E3F4 (cc-LO) (31) - IEAVMODE.IEAVMODE+1C
- R0 00000000 0010DF90 E7C4C3C3 C1D3D340 *.....XDC CALL *
- R4 C9C5C1E5 D4D6C4C5 1EE2E3D8 0503104D *IEAVMODE.STQ...(*
reversing and exploiting

wonder what this vendor-provided svc does?
Untrusted parameters and registers
Just a backdoor

DEMO
putting it all together

DEMO
further research

where to go from here?
black hat sound bytes

• mainframe is just another computer
• it isn’t COBOL
• it pretty much runs the financial infrastructure of the planet
• oh, and also the airlines, government and healthcare
• the security posture could be good, but isn’t yet
• most vulnerabilities work here, with some variation
• get a pentest, assessment at least annually
• **Vulnerability patterns on z/OS**
  (http://events.share.org/Summer2017/Public/SessionDetails.aspx?FromPage=Speakers.aspx&SessionID=3401&nav=true&Role=U%27)

• **z/Architecture Principles of Operations**
  (https://www-01.ibm.com/support/docview.wss?uid=isg2b9de5f05a9d57819852571c500428f9a)

• **z/XDC Debugger**
  (http://colesoft.com/zxdc/)
thank you
Contact Info:
Chad Rikansrud
Director, N.A. Operations
chadr@rsmpartners.com
@bigendiansmalls