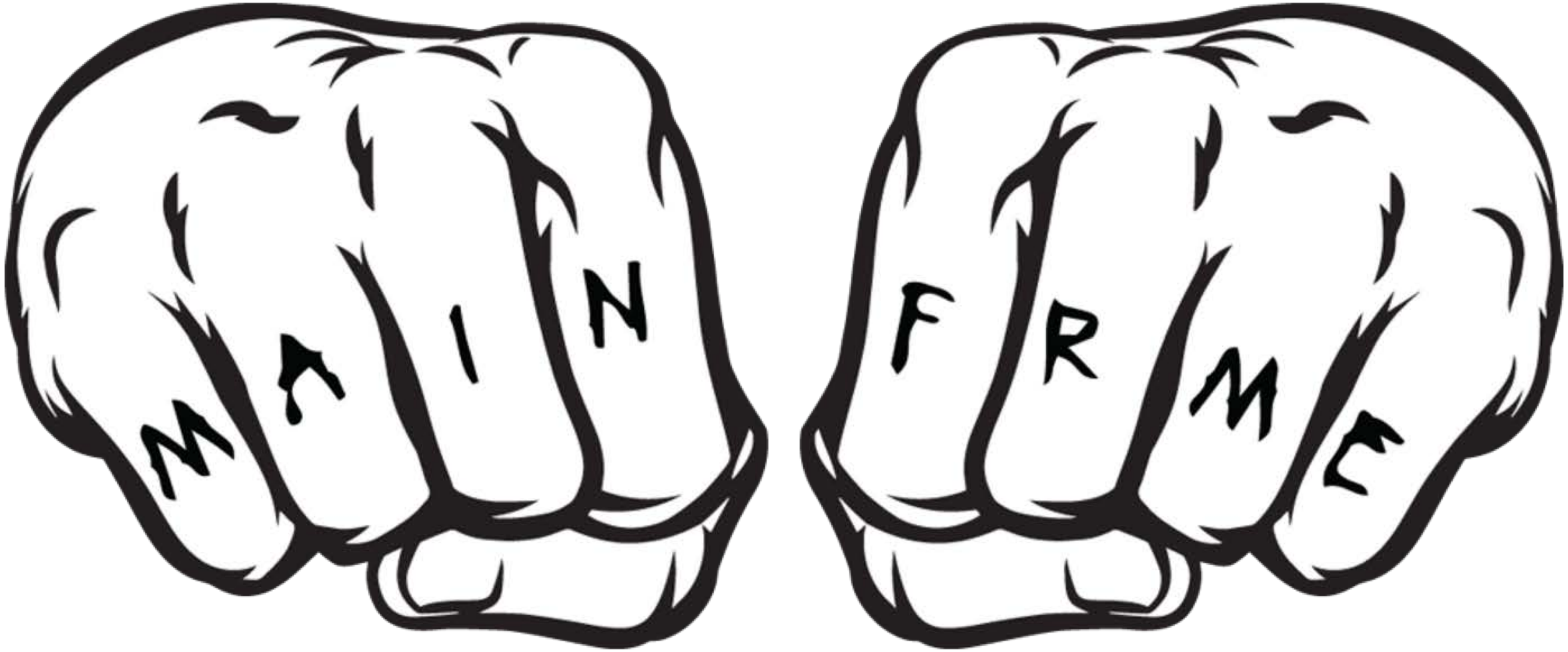
mainframe [z/OS] reverse engineering and exploit development



Chad Rikansrud
Director, North America
RSM Partners

about me

i used to

FASHION

but now i

K-9
CAUTION: CANINE ON BOARD
MID-STATE
CONSULTING
O.S.P. ENG. / O.S.P. INSPECTOR
SPORT
WRANGLER UNLIMITED
GOODYEAR

and teach mainframe hacking

EVIL MAINFRAME

#070C0000

so pretty much i

hack gibsons for a living

at mainframe security hq

RSM PARTNERS

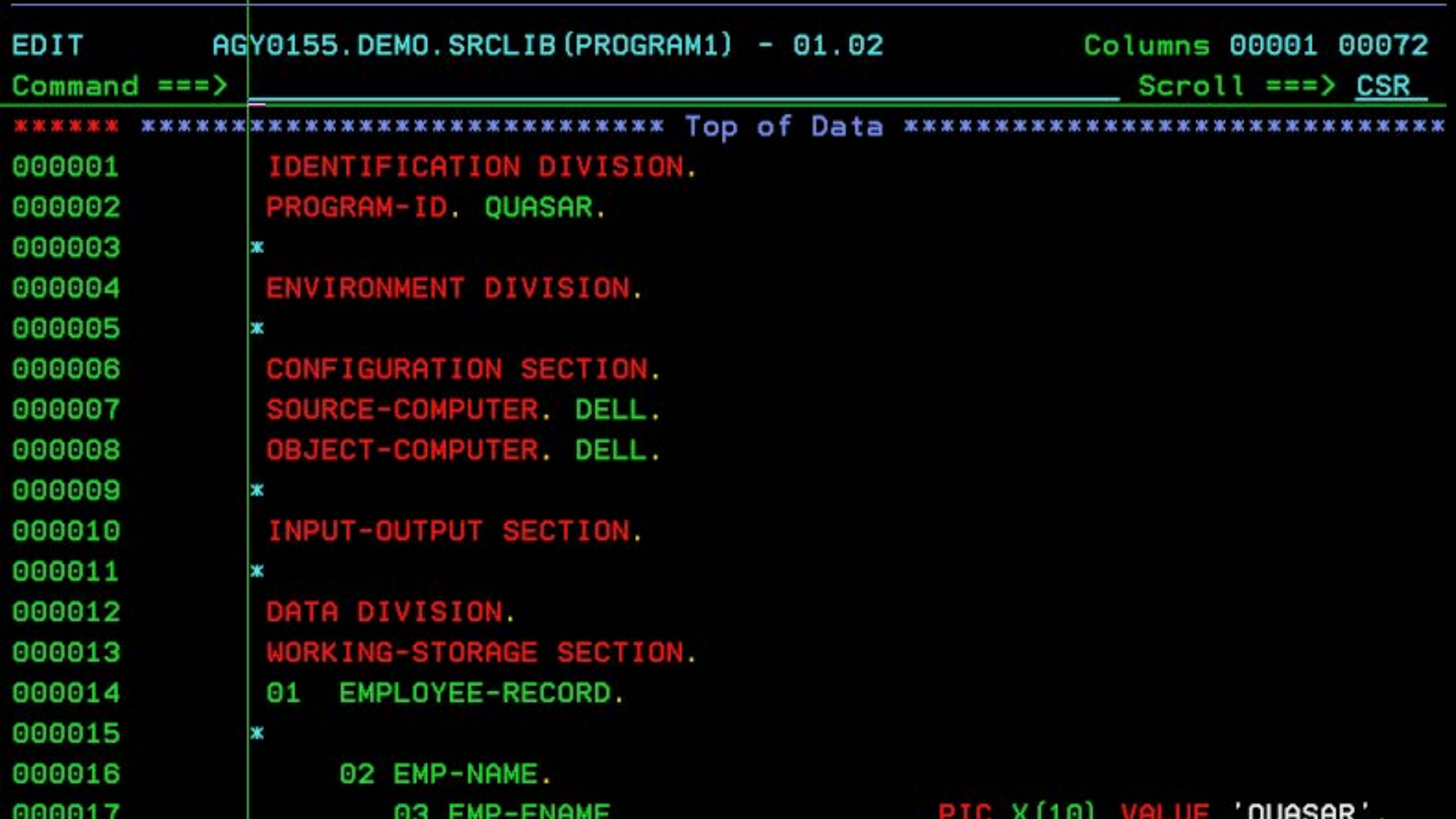
Mainframe Experts

- Pentesting
- Assessments
- Software
- Red Team Augmentation

the machine

architecture

what most people think



```
EDIT       AGY0155.DEMO.SRCLIB(PROGRAM1) - 01.02              Columns 00001 00072
Command ===>                                                   Scroll ===> CSR
****** ***************************** Top of Data ******************************
000001        IDENTIFICATION DIVISION.
000002        PROGRAM-ID. QUASAR.
000003       *
000004        ENVIRONMENT DIVISION.
000005       *
000006        CONFIGURATION SECTION.
000007        SOURCE-COMPUTER. DELL.
000008        OBJECT-COMPUTER. DELL.
000009       *
000010        INPUT-OUTPUT SECTION.
000011       *
000012        DATA DIVISION.
000013        WORKING-STORAGE SECTION.
000014        01  EMPLOYEE-RECORD.
000015       *
000016            02 EMP-NAME.
000017               03 EMP-FNAME                      PIC X(10) VALUE 'QUASAR'.
```

what media thinks

what it really is

IBM
Z

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

E-MAN-BAN

mainframe style

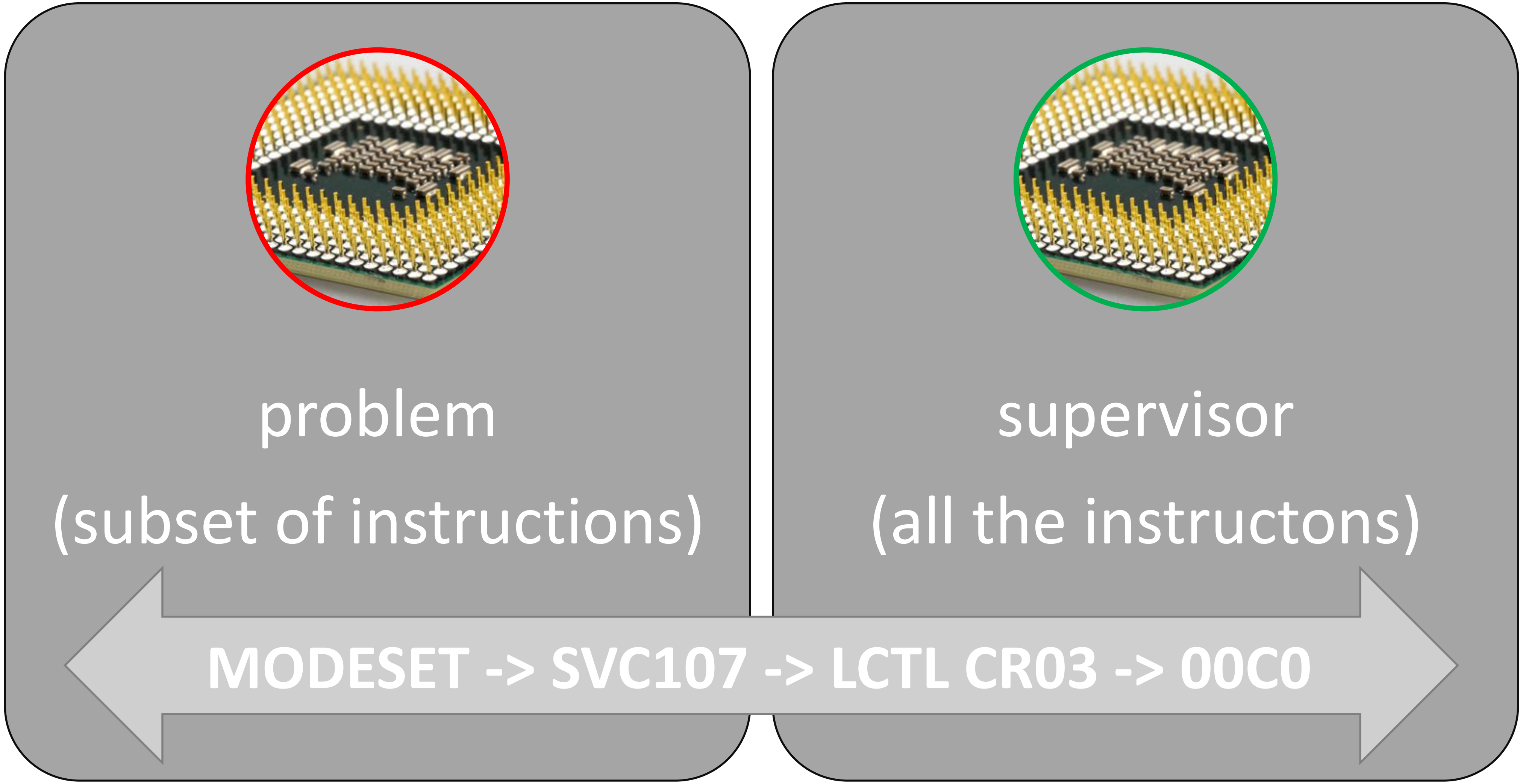
z/architecture and z/os terms

just the basics

not going into

- CICS
- TSO/e
- Datasets
- ESM (RACF, TSS, ACF/2)
- see loads of other talks, presenations and content by:
  - myself
  - @mainframed767
  - @ayoul3__

changing cpu state



problem

(subset of instructions)

supervisor

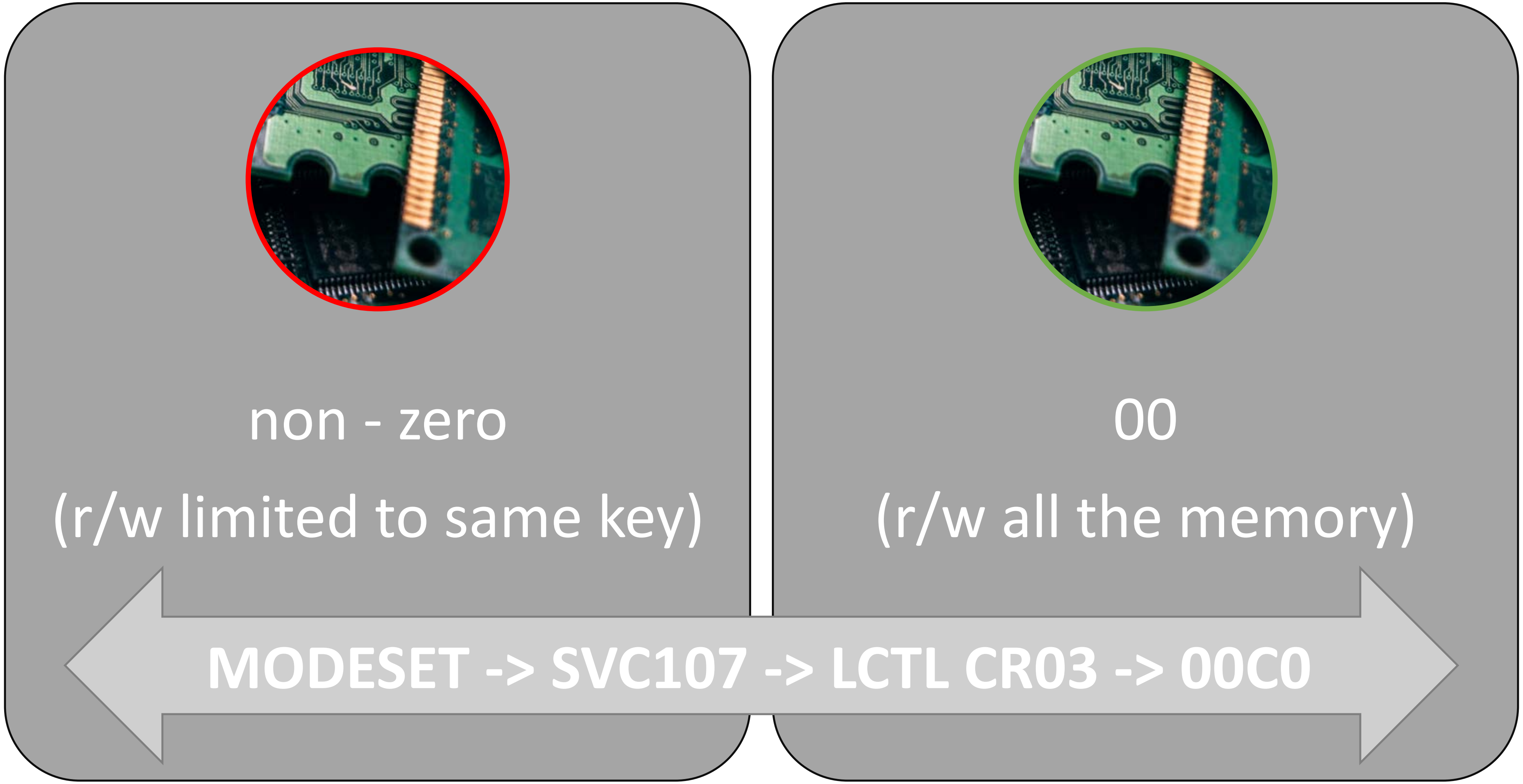
(all the instructons)

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

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

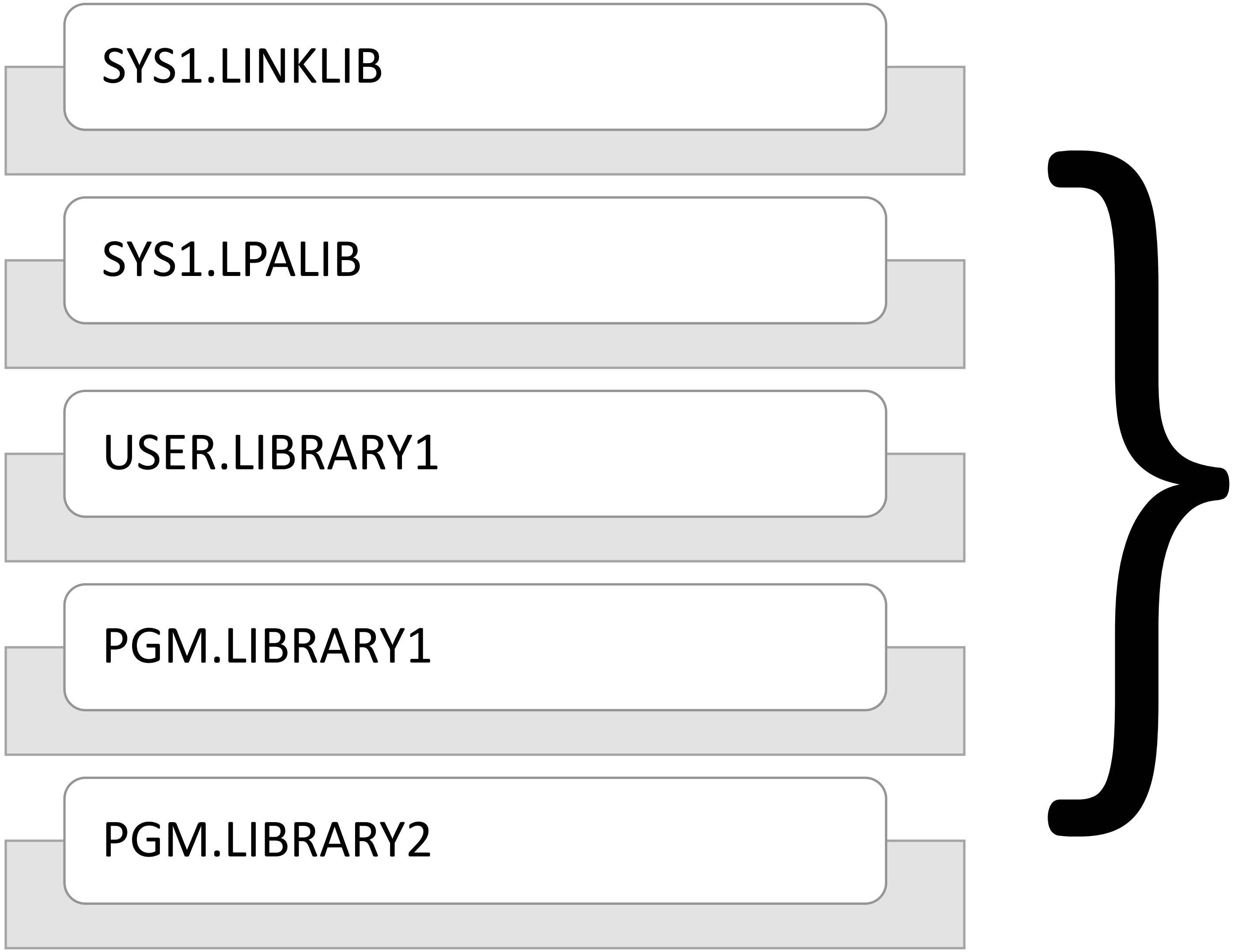
how it works in z/os

- 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

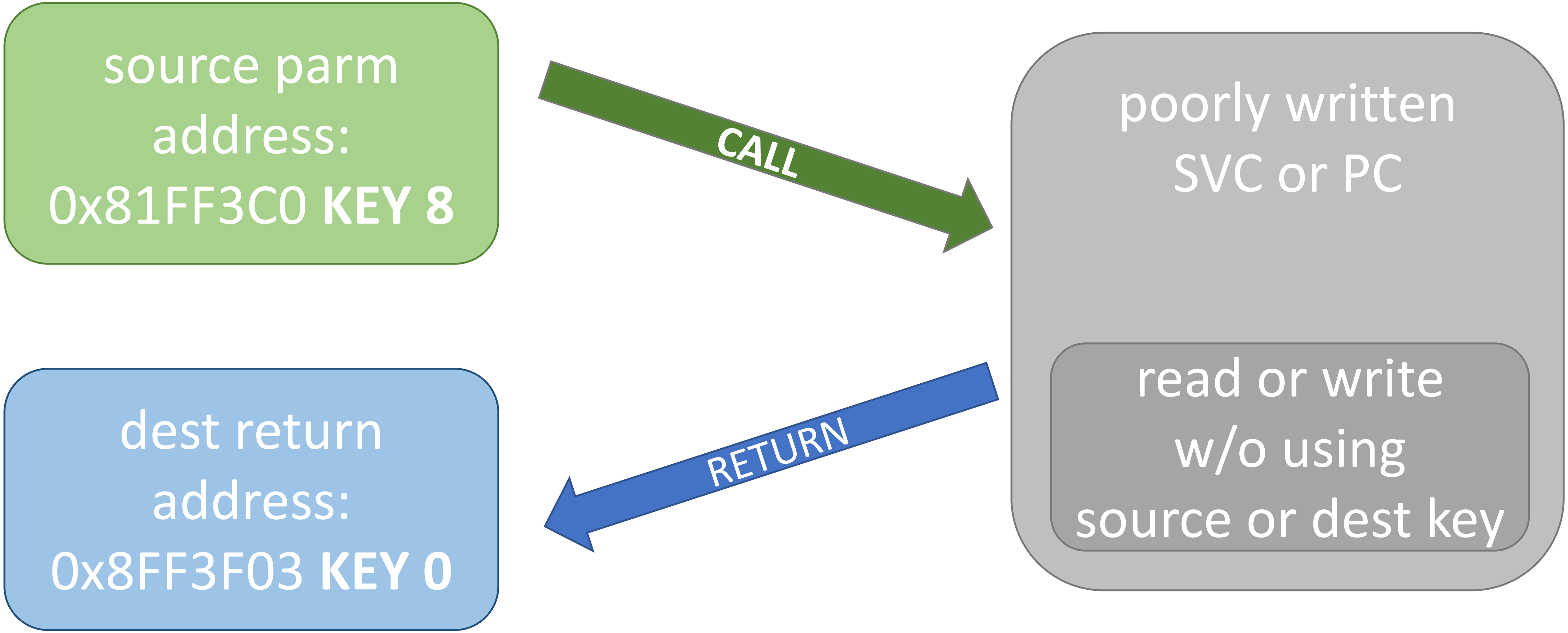
PGM.LIBRARY2

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

vulnerabilties

some unique, some familiar

untrusted parameters



source parm address: 0x81FF3C0 **KEY 8**

CALL

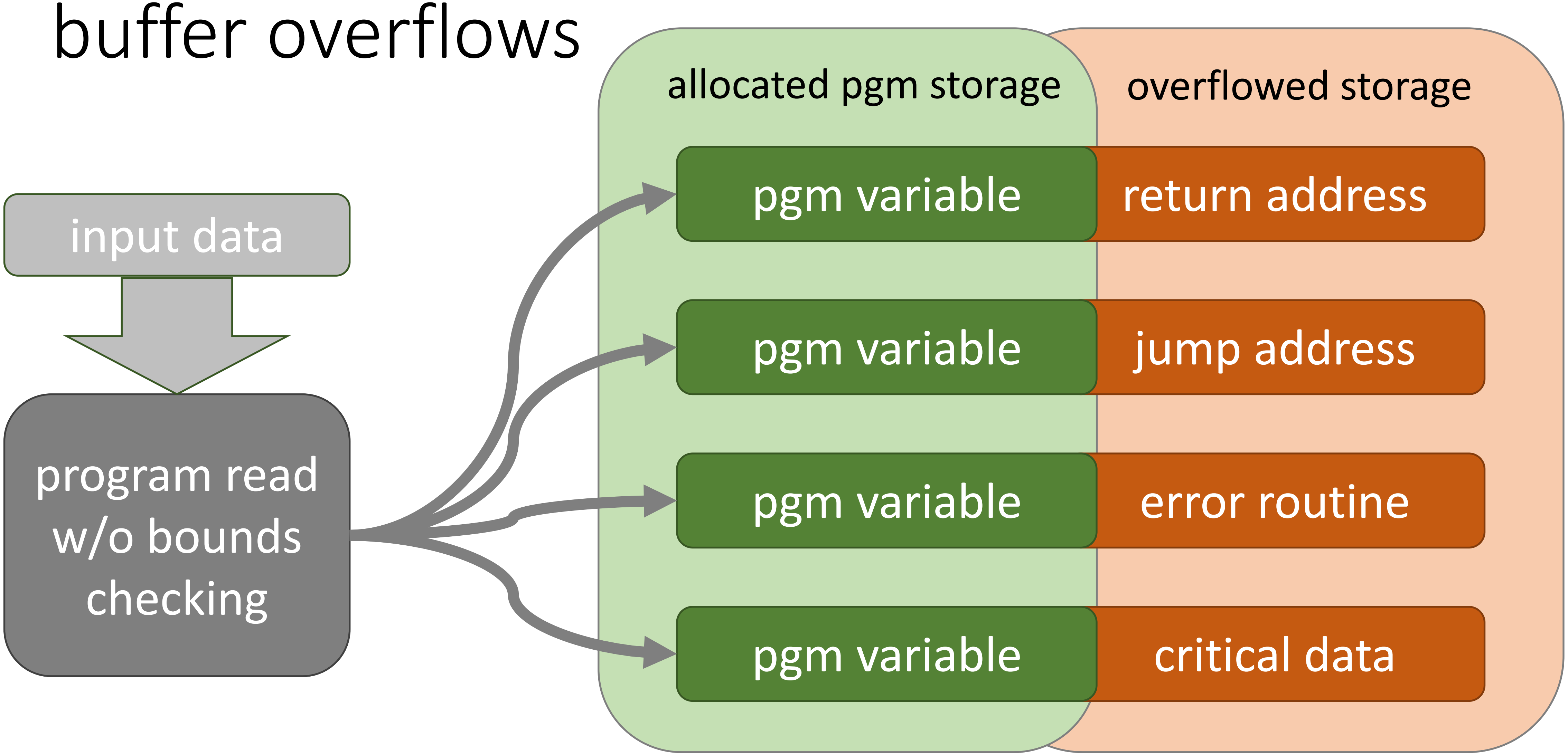
poorly written SVC or PC

read or write w/o using source or dest key

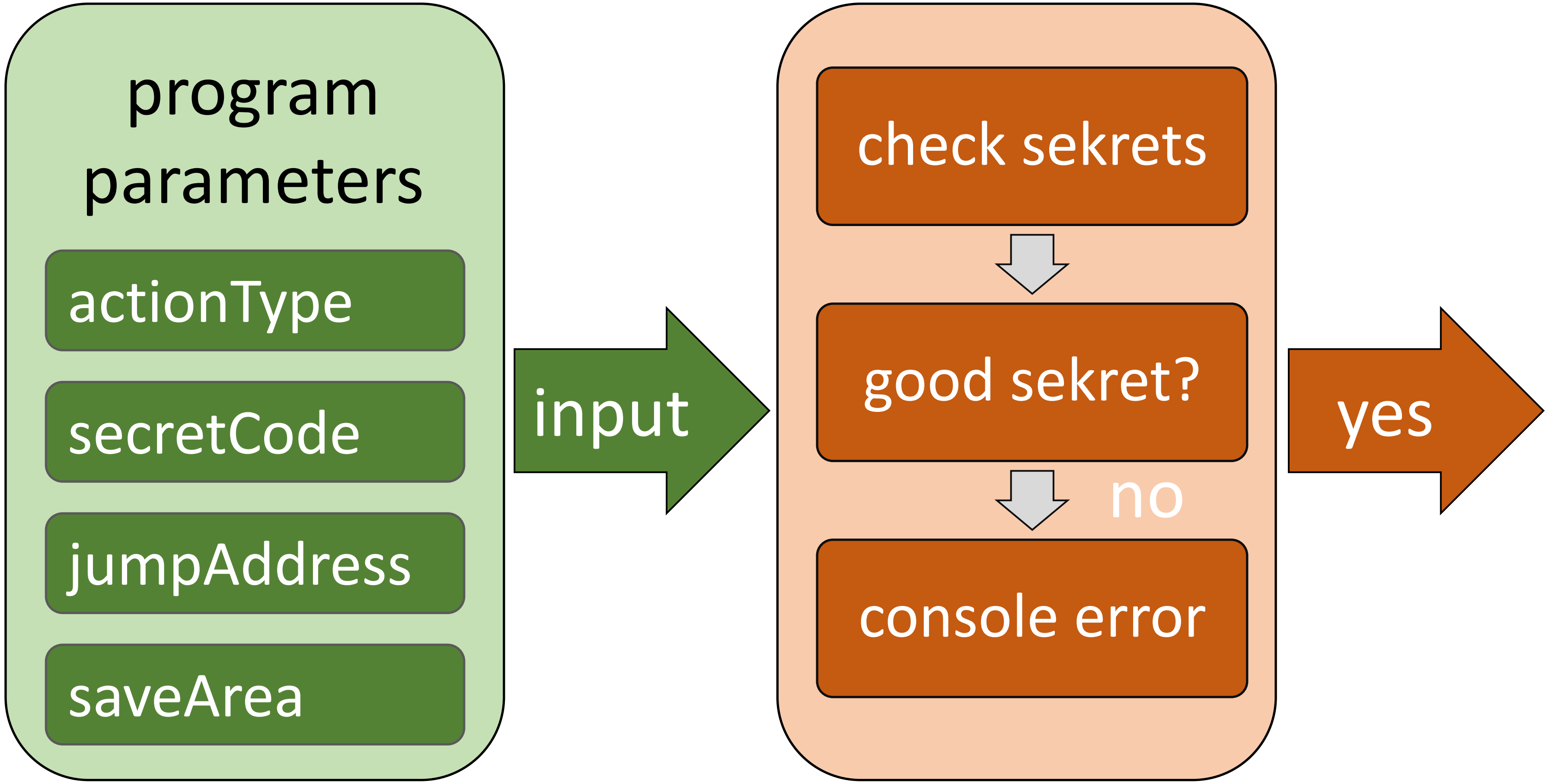
RETURN

dest return address: 0x8FF3F03 **KEY 0**

buffer overflows



intentional backdoors



program parameters

- actionType
- secretCode
- jumpAddress
- saveArea

input

check sekrets

good sekret?

no

console error

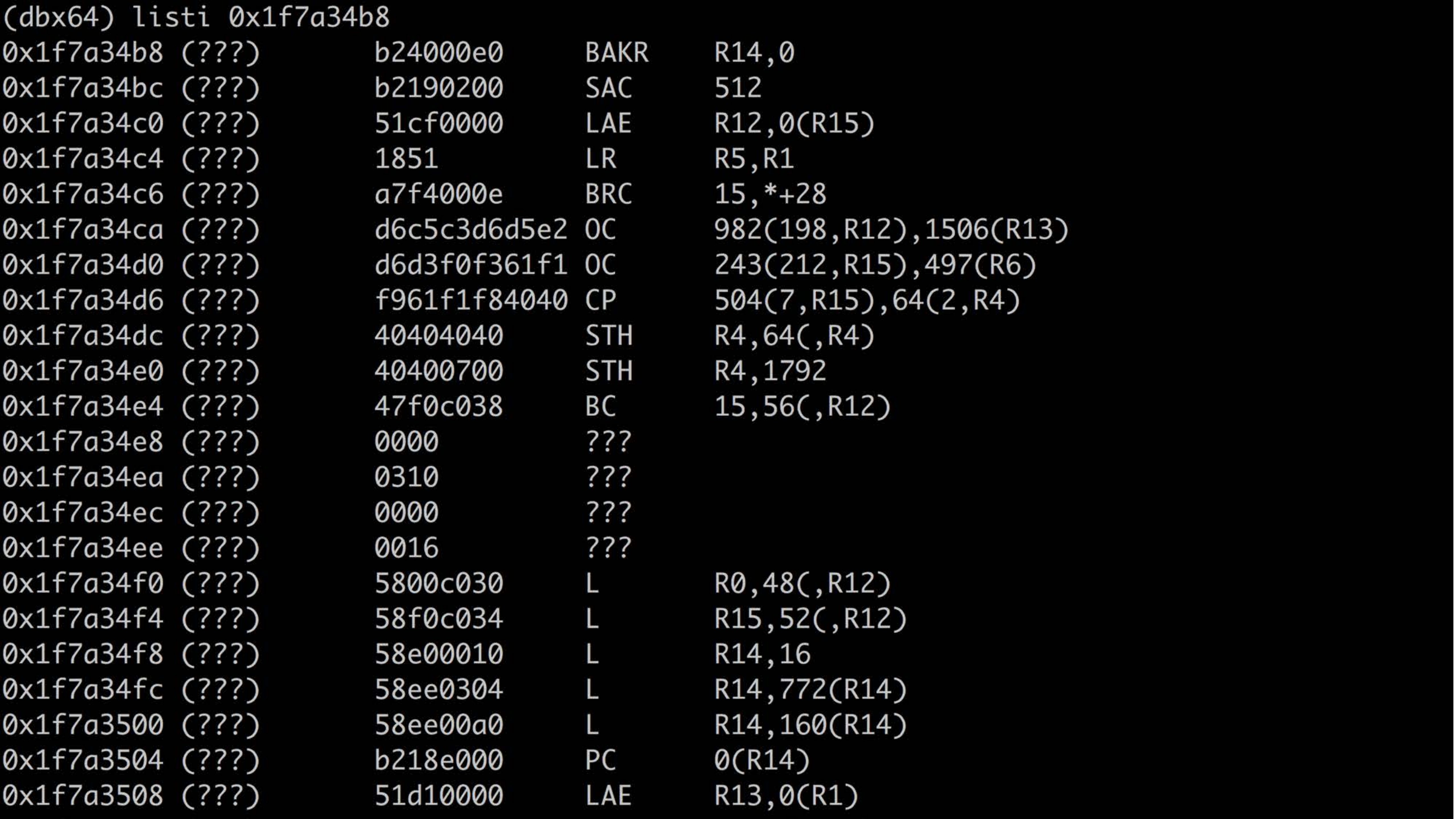
yes

the tools

bad, badder, baddest, really quite good

DBX

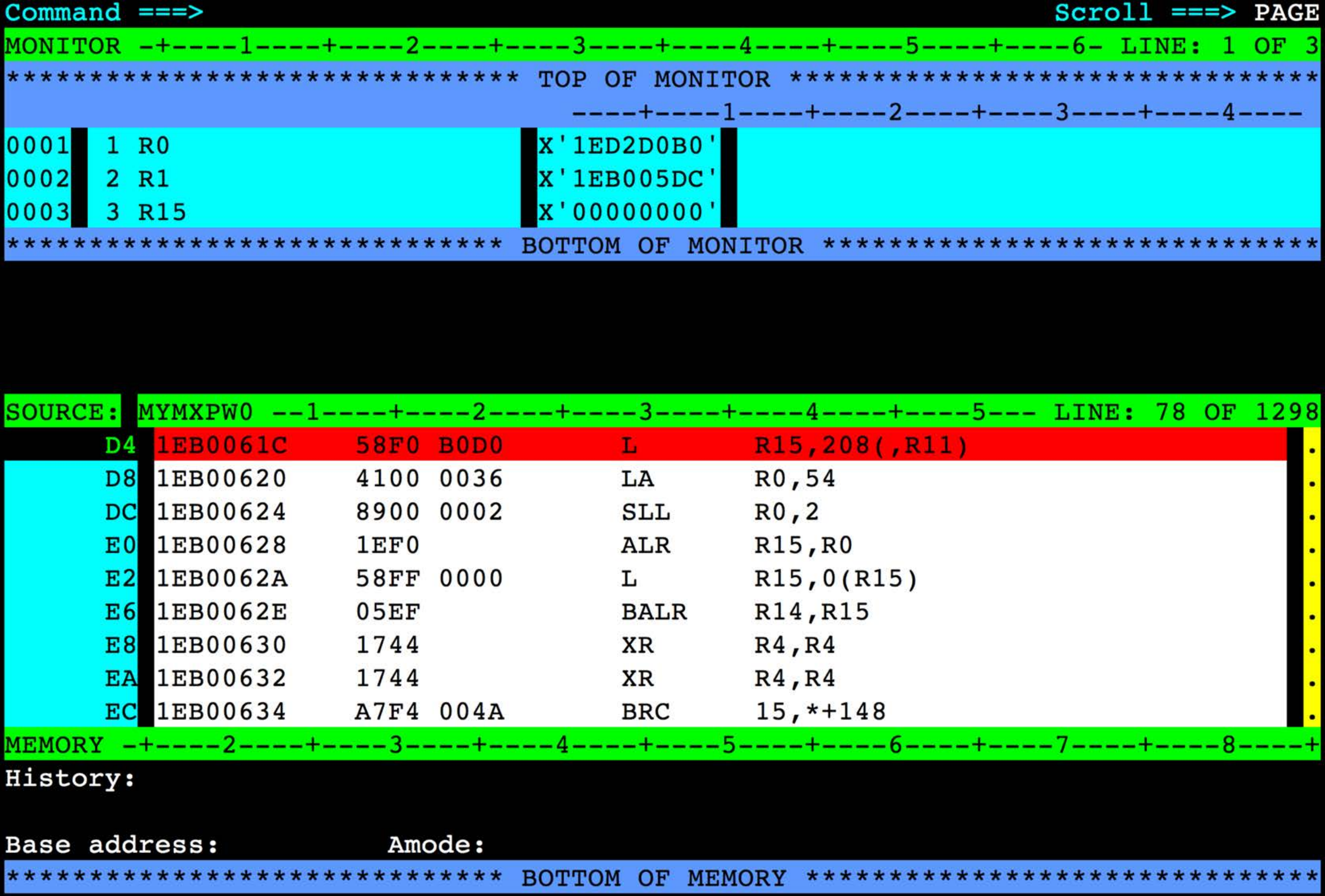
like GDB, but not nearly as fun



```
(dbx64) listi 0x1f7a34b8
0x1f7a34b8 (???)       b24000e0     BAKR    R14,0
0x1f7a34bc (???)       b2190200     SAC     512
0x1f7a34c0 (???)       51cf0000     LAE     R12,0(R15)
0x1f7a34c4 (???)       1851         LR      R5,R1
0x1f7a34c6 (???)       a7f4000e     BRC     15,*+28
0x1f7a34ca (???)       d6c5c3d6d5e2 OC      982(198,R12),1506(R13)
0x1f7a34d0 (???)       d6d3f0f361f1 OC      243(212,R15),497(R6)
0x1f7a34d6 (???)       f961f1f84040 CP      504(7,R15),64(2,R4)
0x1f7a34dc (???)       40404040     STH     R4,64(,R4)
0x1f7a34e0 (???)       40400700     STH     R4,1792
0x1f7a34e4 (???)       47f0c038     BC      15,56(,R12)
0x1f7a34e8 (???)       0000         ???
0x1f7a34ea (???)       0310         ???
0x1f7a34ec (???)       0000         ???
0x1f7a34ee (???)       0016         ???
0x1f7a34f0 (???)       5800c030     L       R0,48(,R12)
0x1f7a34f4 (???)       58f0c034     L       R15,52(,R12)
0x1f7a34f8 (???)       58e00010     L       R14,16
0x1f7a34fc (???)       58ee0304     L       R14,772(R14)
0x1f7a3500 (???)       58ee00a0     L       R14,160(R14)
0x1f7a3504 (???)       b218e000     PC      0(R14)
0x1f7a3508 (???)       51d10000     LAE     R13,0(R1)
```

debug tool

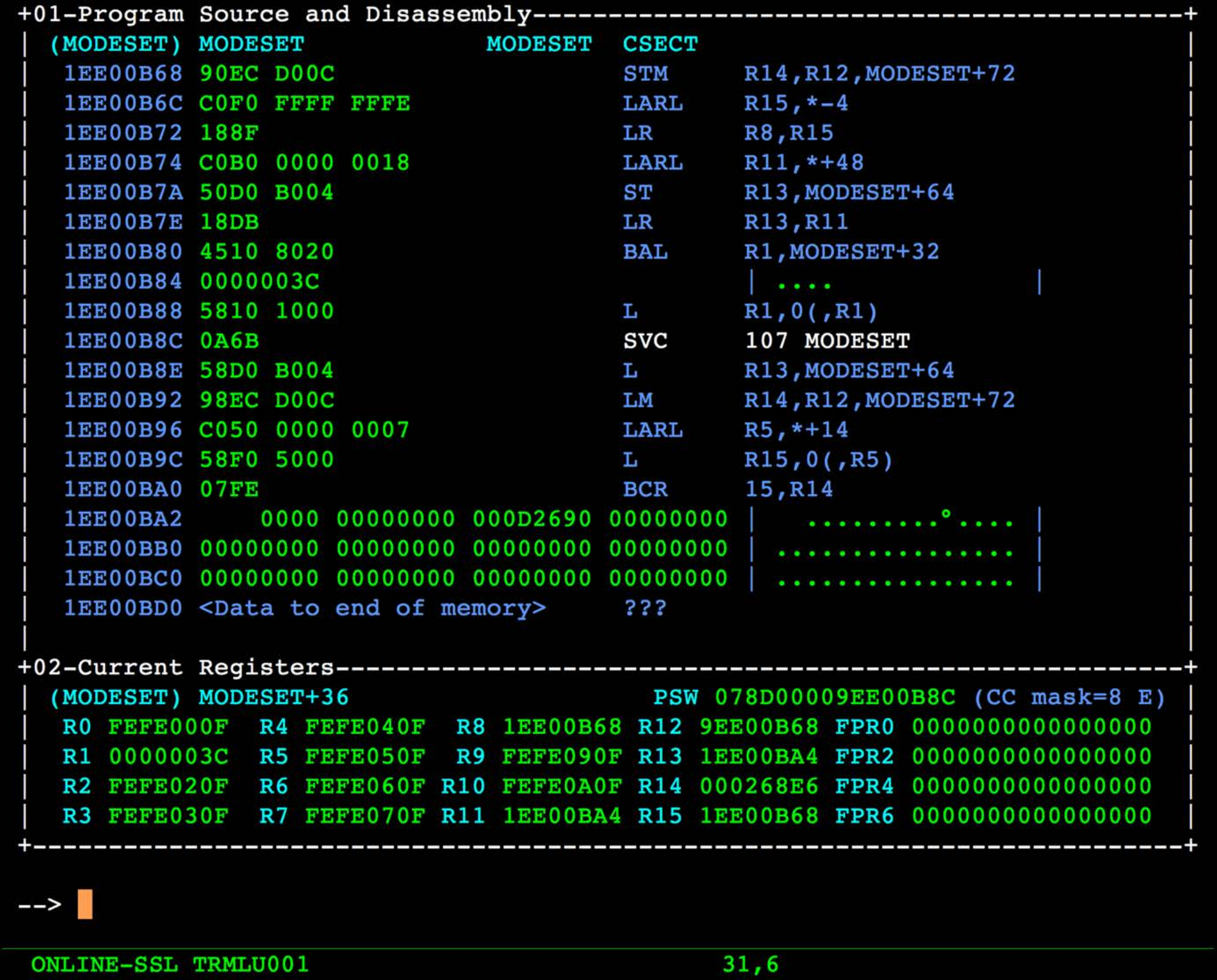
really just here for the colors



```
Command ===>                                                    Scroll ===> PAGE
MONITOR -+----1----+----2----+----3----+----4----+----5----+----6- LINE: 1 OF 3
****************************** TOP OF MONITOR ********************************
                                          ----+----1----+----2----+----3----+----4----
0001   1 R0                              X'1ED2D0B0'
0002   2 R1                              X'1EB005DC'
0003   3 R15                             X'00000000'
****************************** BOTTOM OF MONITOR *****************************



SOURCE: MYMXPW0 --1----+----2----+----3----+----4----+----5--- LINE: 78 OF 1298
     D4 1EB0061C    58F0 B0D0        L        R15,208(,R11)                      .
     D8 1EB00620    4100 0036        LA       R0,54                              .
     DC 1EB00624    8900 0002        SLL      R0,2                               .
     E0 1EB00628    1EF0             ALR      R15,R0                             .
     E2 1EB0062A    58FF 0000        L        R15,0(R15)                         .
     E6 1EB0062E    05EF             BALR     R14,R15                            .
     E8 1EB00630    1744             XR       R4,R4                              .
     EA 1EB00632    1744             XR       R4,R4                              .
     EC 1EB00634    A7F4 004A        BRC      15,*+148                           .
MEMORY -+----2----+----3----+----4----+----5----+----6----+----7----+----8----+
History:

Base address:                Amode:
****************************** BOTTOM OF MEMORY ******************************
```

ASMIDF

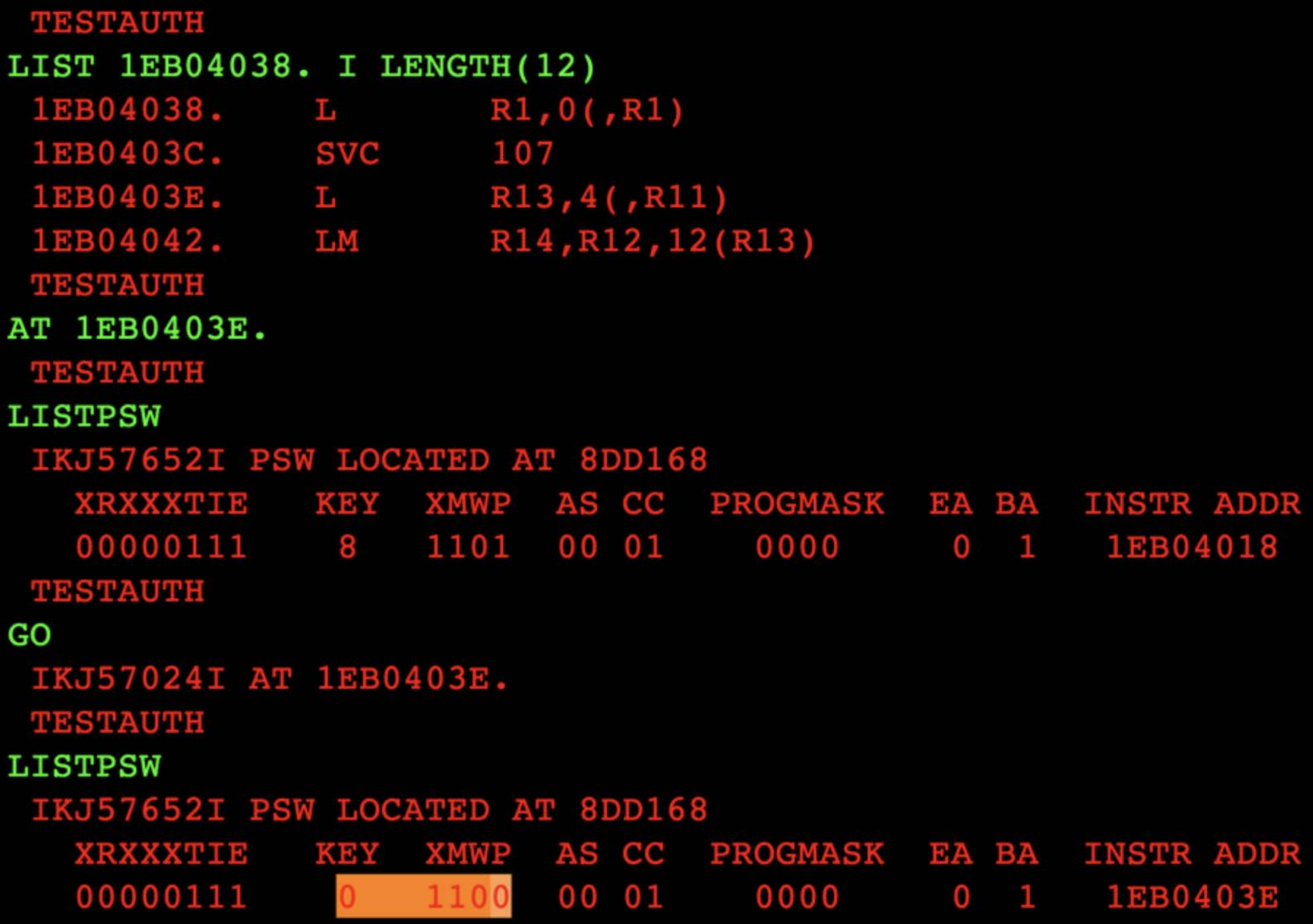
after hella modifications, can be somewhat useful



```
+01-Program Source and Disassembly-------------------------------------------+
| (MODESET) MODESET            MODESET  CSECT                                 |
|  1EE00B68 90EC D00C                   STM     R14,R12,MODESET+72            |
|  1EE00B6C C0F0 FFFF FFFE              LARL    R15,*-4                       |
|  1EE00B72 188F                        LR      R8,R15                        |
|  1EE00B74 C0B0 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 00000000 000D2690 00000000 |   .........°.... |          |
|  1EE00BB0 00000000 00000000 00000000 00000000 | ................ |          |
|  1EE00BC0 00000000 00000000 00000000 00000000 | ................ |          |
|  1EE00BD0 <Data to end of memory>     ???                                   |
|                                                                             |
+02-Current Registers--------------------------------------------------------+
| (MODESET) MODESET+36              PSW 078D00009EE00B8C (CC mask=8 E) |
|  R0 FEFE000F  R4 FEFE040F  R8 1EE00B68 R12 9EE00B68 FPR0 0000000000000000 |
|  R1 0000003C  R5 FEFE050F  R9 FEFE090F R13 1EE00BA4 FPR2 0000000000000000 |
|  R2 FEFE020F  R6 FEFE060F R10 FEFE0A0F R14 000268E6 FPR4 0000000000000000 |
|  R3 FEFE030F  R7 FEFE070F R11 1EE00BA4 R15 1EE00B68 FPR6 0000000000000000 |
+---------------------------------------------------------------------------+

-->

ONLINE-SSL TRMLU001                        31,6
```

TSO/e TEST

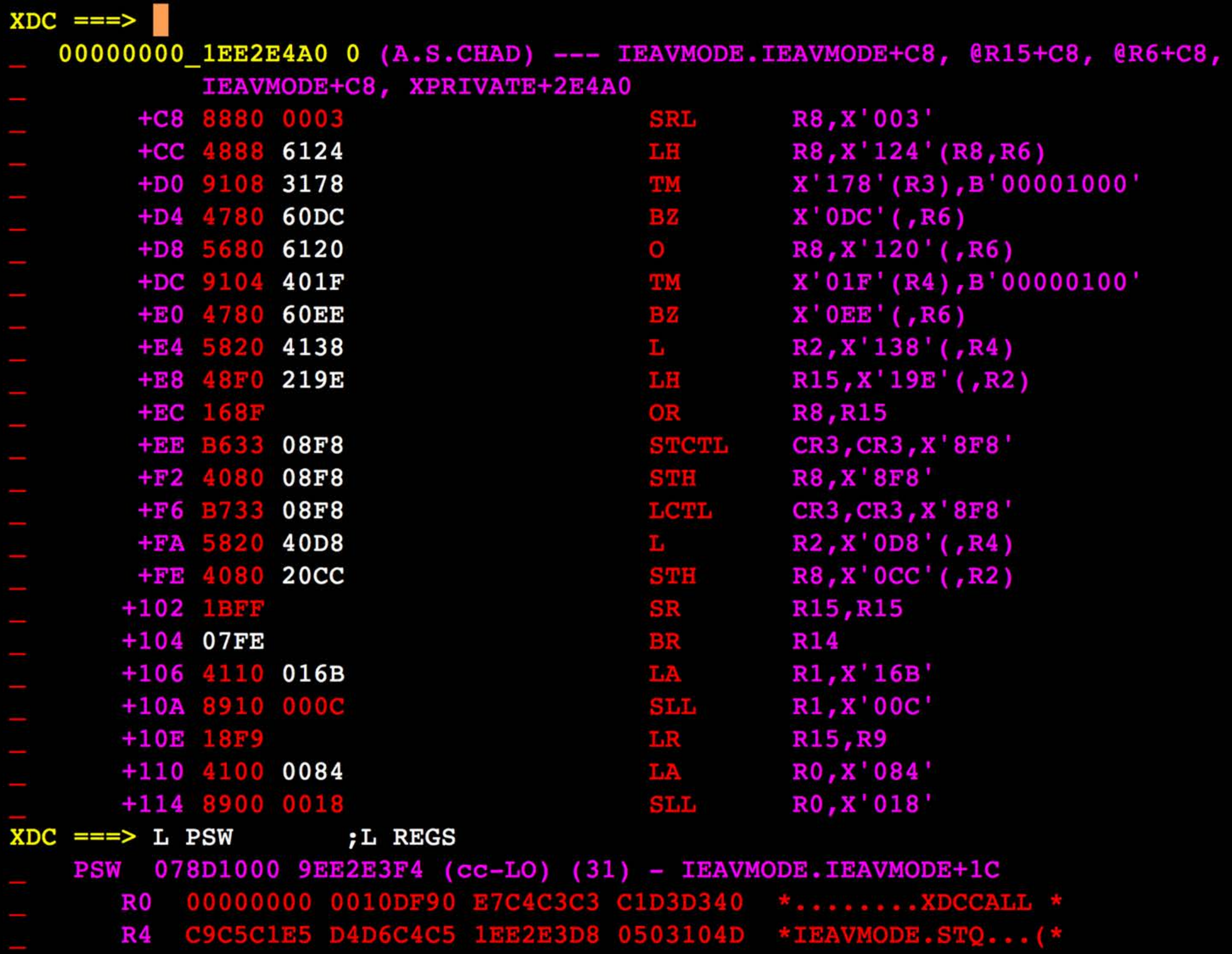
learn it for the same reason you learned ‘ed’



```
 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
   XRXXXTIE   KEY  XMWP  AS CC  PROGMASK  EA BA  INSTR ADDR
   00000111    8   1101  00 01    0000     0  1   1EB04018
 TESTAUTH
GO
 IKJ57024I AT 1EB0403E.
 TESTAUTH
LISTPSW
 IKJ57652I PSW LOCATED AT 8DD168
   XRXXXTIE   KEY  XMWP  AS CC  PROGMASK  EA BA  INSTR ADDR
   00000111    0   1100  00 01    0000     0  1   1EB0403E
```

z/XDC

the real contender (non-IBM)



```
XDC ===>
_  00000000_1EE2E4A0 0 (A.S.CHAD) --- IEAVMODE.IEAVMODE+C8, @R15+C8, @R6+C8,
_           IEAVMODE+C8, XPRIVATE+2E4A0
_       +C8 8880 0003                   SRL      R8,X'003'
_       +CC 4888 6124                   LH       R8,X'124'(R8,R6)
_       +D0 9108 3178                   TM       X'178'(R3),B'00001000'
_       +D4 4780 60DC                   BZ       X'0DC'(,R6)
_       +D8 5680 6120                   O        R8,X'120'(,R6)
_       +DC 9104 401F                   TM       X'01F'(R4),B'00000100'
_       +E0 4780 60EE                   BZ       X'0EE'(,R6)
_       +E4 5820 4138                   L        R2,X'138'(,R4)
_       +E8 48F0 219E                   LH       R15,X'19E'(,R2)
_       +EC 168F                        OR       R8,R15
_       +EE B633 08F8                   STCTL    CR3,CR3,X'8F8'
_       +F2 4080 08F8                   STH      R8,X'8F8'
_       +F6 B733 08F8                   LCTL     CR3,CR3,X'8F8'
_       +FA 5820 40D8                   L        R2,X'0D8'(,R4)
_       +FE 4080 20CC                   STH      R8,X'0CC'(,R2)
_      +102 1BFF                        SR       R15,R15
_      +104 07FE                        BR       R14
_      +106 4110 016B                   LA       R1,X'16B'
_      +10A 8910 000C                   SLL      R1,X'00C'
_      +10E 18F9                        LR       R15,R9
_      +110 4100 0084                   LA       R0,X'084'
_      +114 8900 0018                   SLL      R0,X'018'
XDC ===> L PSW       ;L REGS
_   PSW  078D1000 9EE2E3F4 (cc-LO) (31) - IEAVMODE.IEAVMODE+1C
_      R0  00000000 0010DF90 E7C4C3C3 C1D3D340  *........XDCCALL *
_      R4  C9C5C1E5 D4D6C4C5 1EE2E3D8 0503104D  *IEAVMODE.STQ...(*
```

reversing and exploiting

wonder what this vendor-provided svc does?

Untrusted parameters and registers

DEMO

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

reading - info

- 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

black hat USA 2018

AUGUST 4-9, 2018
MANDALAY BAY / LAS VEGAS

Contact Info:

Chad Rikansrud
Director, N.A. Operations
chadr@rsmpartners.com
@bigendiansmalls

RSM PARTNERS
Z SECURITY