
Vulnerabilities in
TN3270
Applications
by @singe

Dominic White

CTO @sensepost

We

Hack | Build | Train | Scan

Stuff

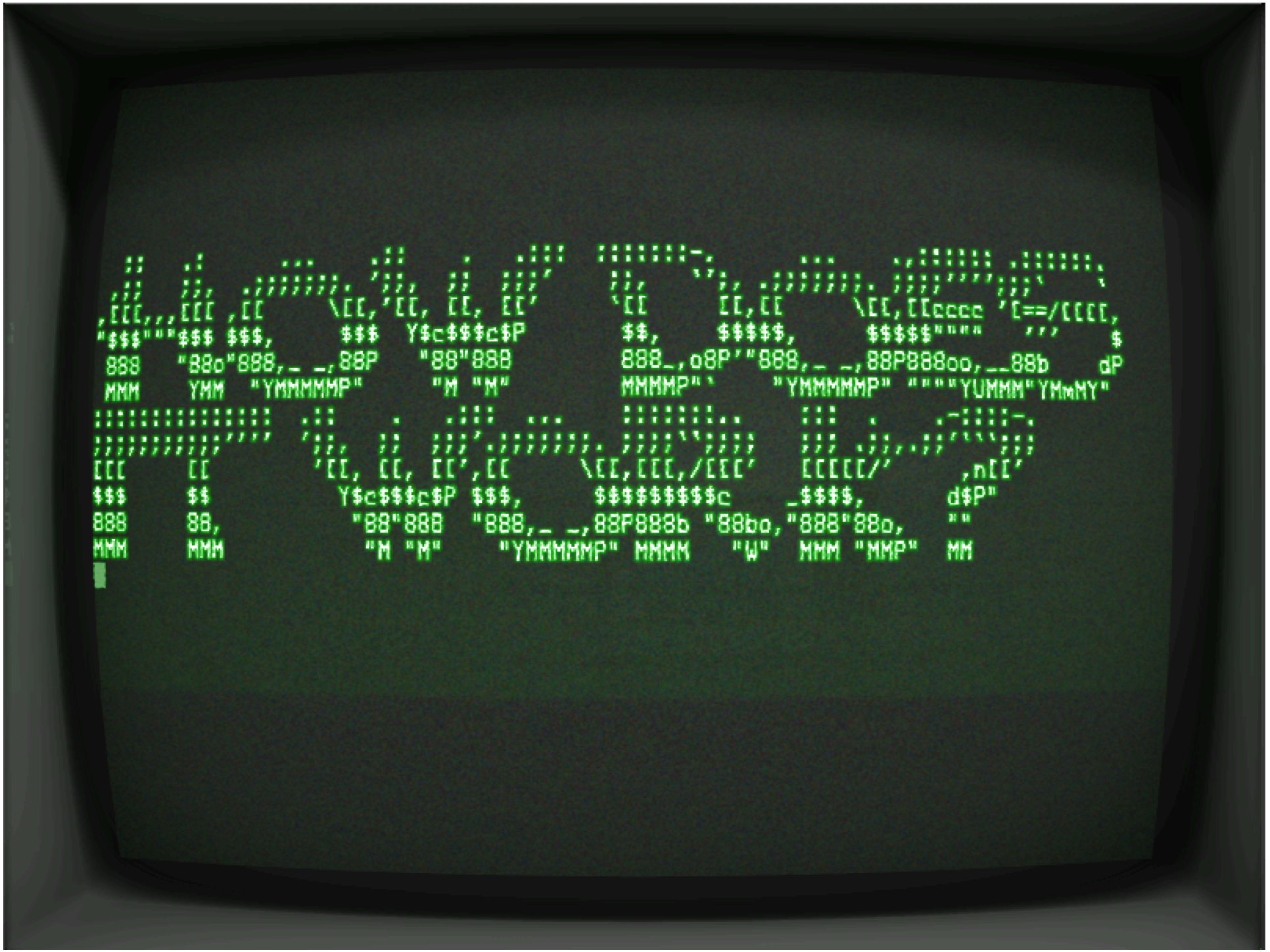
@singe

1. Justification
2. Structure & Reconnaissance
3. Applications
    1. Application Regions
    2. TN3270 protocol-level vulns
4. Disclosure Notes
5. Bootstrapping

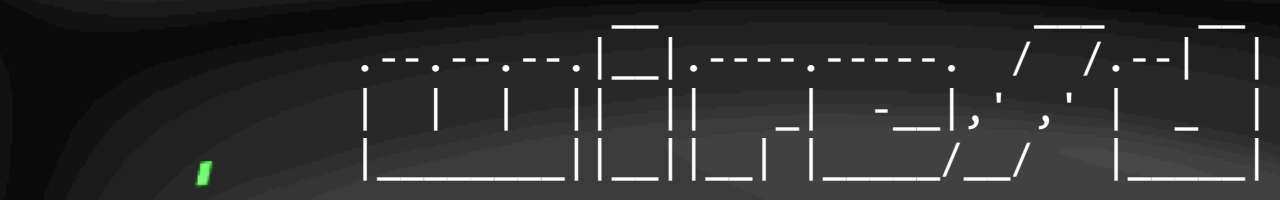
- Context: IBM z/OS systems

- Still in heavy production use
- Often underpin critical business processes
- Actively maintained

- They won't be around much longer!
- They're very secure!
- Concerns about downtime & failures
- It's hard to learn & practice
- IBM's legal frameworks
- More forgotten than you'll learn

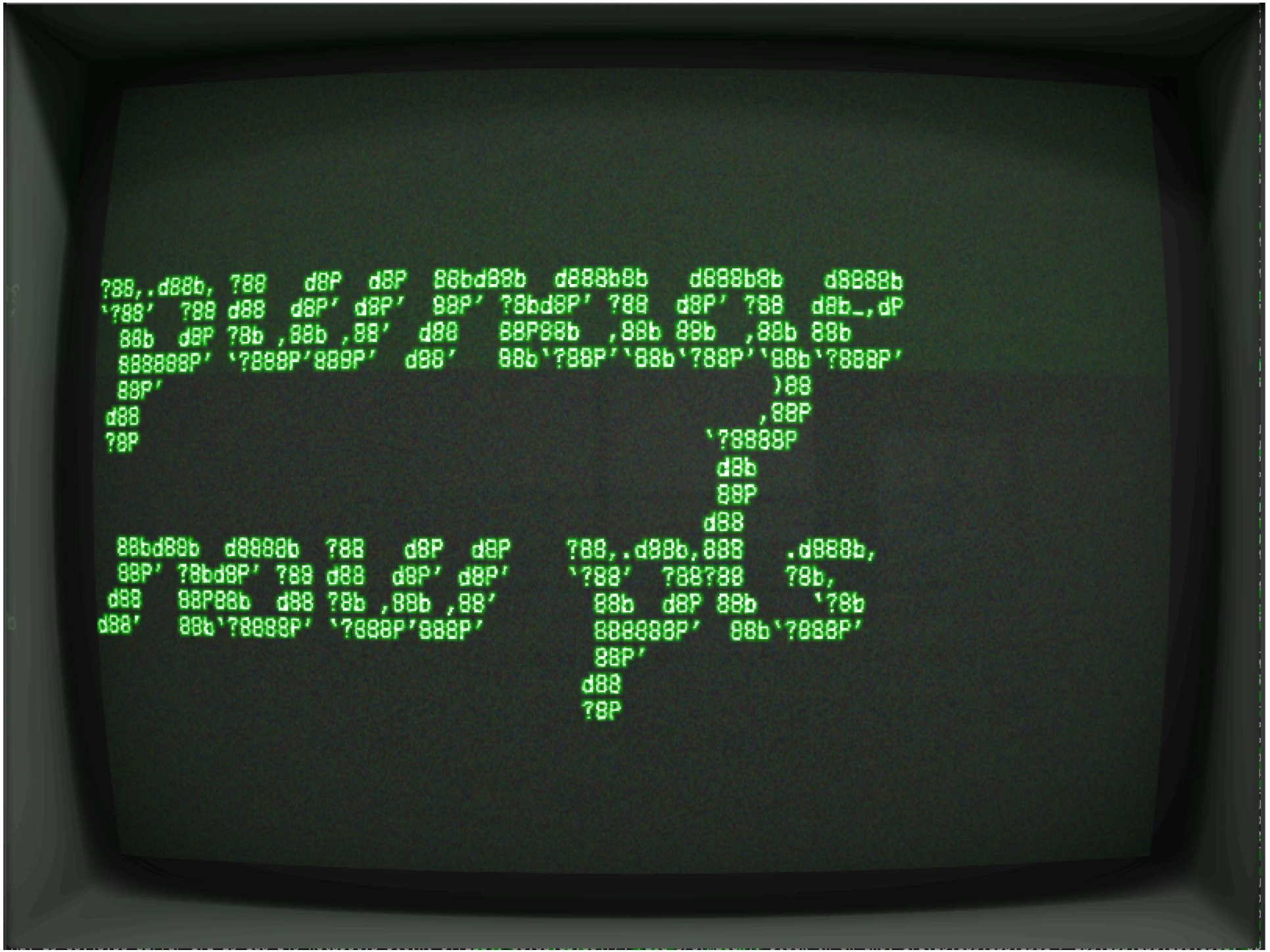


- Mainframe/Host == Big hulking pieces of **hardware** running **z/OS** (or Z/VM)
- Partitioned into lots of VMs called **LPARs** (logical partition)
  - Each LPAR can run different stuff e.g. IBM z/OS (mainframe) or with z/VM Linux (AIX or RedHat)
  - 1972 hypervisors baby!
- Lots of LPARs (across hardware too) is a **Sysplex**



- *SNA* – Systems Network Architecture
  - Inter-mainframe or peripheral comms
- TN3270/E
  - 3270 terminal emulation over Telnet
- *VTAM* – Virtual Telecommunications Access Method
  - Subsystem that implements SNA
  - Often the first thing you connect to on a mainframe
- LU / PU – Logical/Physical Unit
  - Connections to VTAM (wired vs multiplexed)
  - TN3270 to mainframe usually gives you a LU

- *TSO* – z/OS CLI
  - "traditional" process accounting
  - CLIST/REXX/JCL scripting
  - *OMVS* / USS – Unix
  - *ISPF* – Menu Screens (GUI)
- Transaction Managers
  - *CICS* – Modern bindings
  - *IMS* – MQ style
  - Efficient high-volume processing
- Applications run within these
  - COBOL / FORTRAN
- Lots of other stuff e.g.
  - Databases: DB2 & IMS
  - Unix: FTP, HTTP, WebSphere
  - MQ
  - Etc.
- Subsystems
  - RACF
  - ACF-2


?88,,d88b, ?88 d8P d8P 88bd88b d888b8b d888b8b d8888b
`?88' ?88 d88 d8P' d8P' 88P' ?8bd8P' ?88 d8P' ?88 d8b_,dP
88b d8P ?8b ,88b ,88' d88 88P88b ,88b 88b ,88b 88b
888888P' `?888P'888P' d88' 88b`?88P'`88b`?88P'`88b`?888P'
88P')88
d88 ,88P
?8P `?8888P
d8b
88P
d88
88bd88b d8888b ?88 d8P d8P ?88,,.d88b,888 .d888b,
88P' ?8bd8P' ?88 d88 d8P' d8P' `?88' ?88?88 ?8b,
d88 88P88b d88 ?8b ,88b ,88' 88b d8P 88b `?8b
d8a' 88b`?8888P' `?888P'888P' 888888P' 88b`?888P'
88P'
d88
?8P

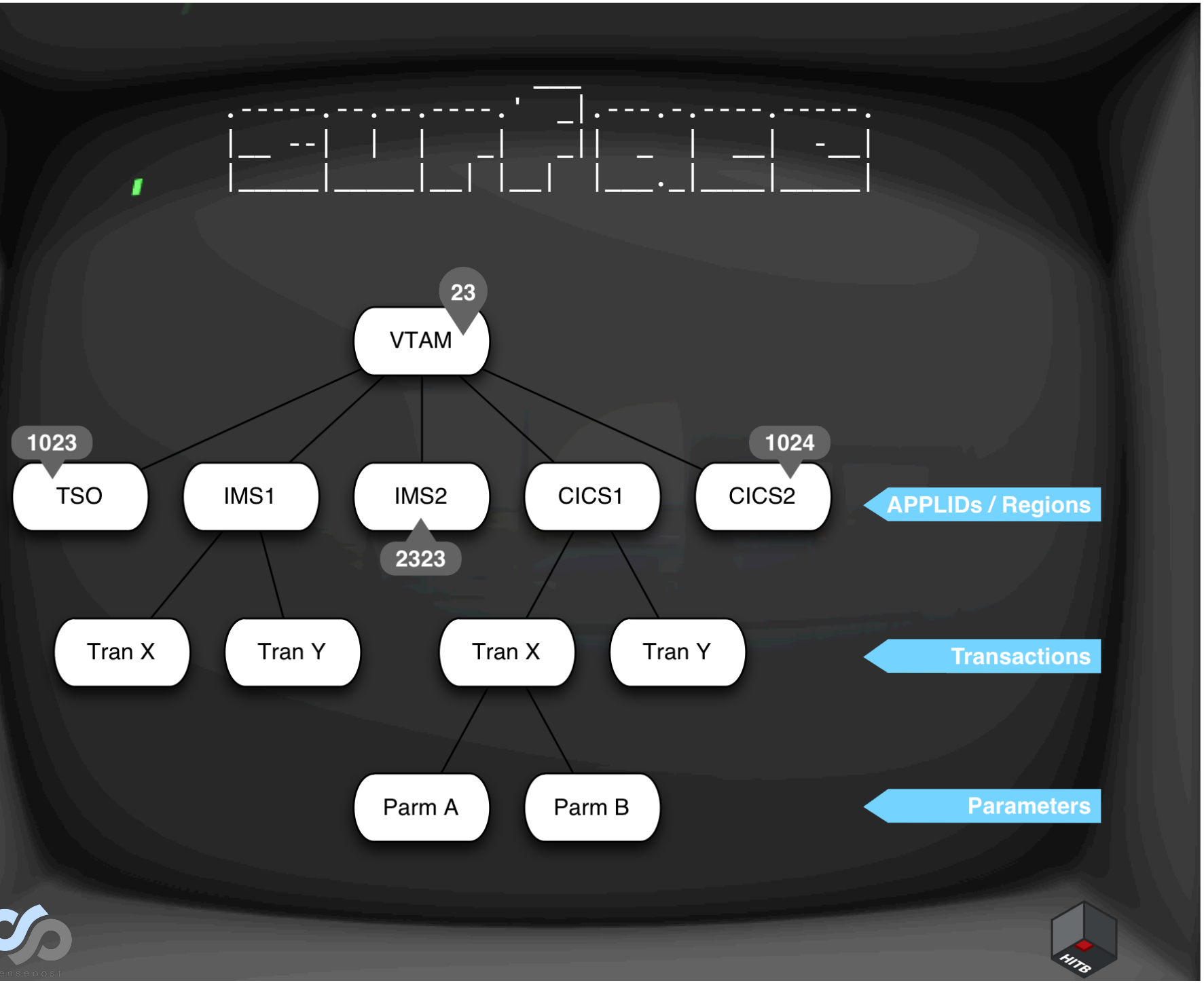
- Application ports == TN3270
  - 23 – default, often VTAM
  - 992 – default SSL enabled
  - 1023-x0xx – application environments (direct to CICS/IMS regions)
  - 2323, x023, x992 – other ports to check
  - Ignore NMAP's OS/390 SNA bit
- FTP
  - Provides access to both worlds (TSO & OMVS)
  - Respects wildcards (*.RACF*.*)
- Other
  - DB2 (5023) & MQ (1415)
  - HP/BMC/Tivoli monitoring
  - WebSphere
- One host can have lots of IPs : Order of 10-20

- Not much you can do without creds
- Legacy password policies
  - 8 char length restrictions
  - No special characters
- FTP
  - Fantastic “traditional” brute-point
- TSO
  - User enumeration flaw
  - TSO-Brute / psiotik (mainframed)
- App creds
  - User enumeration flaws common
  - Sometimes weaker password policies

- Ports can connect directly to a app or region
- There can be multiple instances of an app across regions
- e.g. 1023 → TSO, 1024 → CICS prod, 1025 → CICS dev, 1026 → IMS head office, 1027 → IMS Branch X
- Screenshotting tools
  - 3270_screen_grab.nse (mainframed)
  - screenshotter.py

- Don’t stop at ports only!
- VTAM is a multiplexer of sorts
  - Lets you connect to different application
  - Can connect you to other LPARs & sysplex’s
  - Uses APPLIDs or “macros”
    - LOGON APPLID(TSO) vs TSO
- APPLID bruting
  - mainframe_bruter.py
- https://github.com/singe/mainframe_brute
  - Poor man parallelisation: xargs –P

- Don’t take initial screens at face value
  - “spider” link with webapps
- IMS
  - PA24 to “leave screen”
  - Alternate transaction invocation
    - /display tran & /display psb
  - “Fuzz” parameters and flow
- CICS
  - Look for (brute) transaction codes (URL paths)
    - mainframe_bruter.py again
  - Fuzz parameters
  - Screenshotter useful for mapping output


23
VTAM
1023
TSO
IMS1
IMS2
2323
CICS1
1024
CICS2
APPLIDs / Regions
Tran X
Tran Y
Tran X
Tran Y
Transactions
Parm A
Parm B
Parameters

- Telnet-like protocol introduced in 1971
- Allowed “green screen” terminals to go over network TCP/IP rather than hardwire
- Transmits “screens” made up of fields
- Response submits modified screen & fields
- Synchronous & Stateful
- All apps presented in same way
  - i.e. TSO, CICS, IMS, REXX etc. all use it

- A screen is:
  - n x <Field Marker><Field>
- A field marker can be (bit mask):
  - PRINTABLE = 0xc0 #these make the character "printable"
  - PROTECT = 0x20 #unprotected (0) / protected (1)
  - NUMERIC = 0x10 #alphanumeric (0) /numeric (1) Skip?
  - HIDDEN = 0x0c #display/selector pen detectable:
  - INT_NORM_NSEL = 0x00 # 00 normal, non-detect
  - INT_NORM_SEL = 0x04 # 01 normal, detectable
  - INT_HIGH_SEL = 0x08 # 10 intensified, detectable
  - INT_ZERO_NSEL = 0x0c # 11 nondisplay, non-detect, same as hidden
  - RESERVED = 0x02 #must be 0
  - MODIFY = 0x01 #modified (1)
- There's also a display bit mask for colours

Hack me Bank

USERNAME _____

PASSWORD _____

Hidden & Protected Field

•P1•Hack me Bank

NULL Bytes

NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL

Field Marker

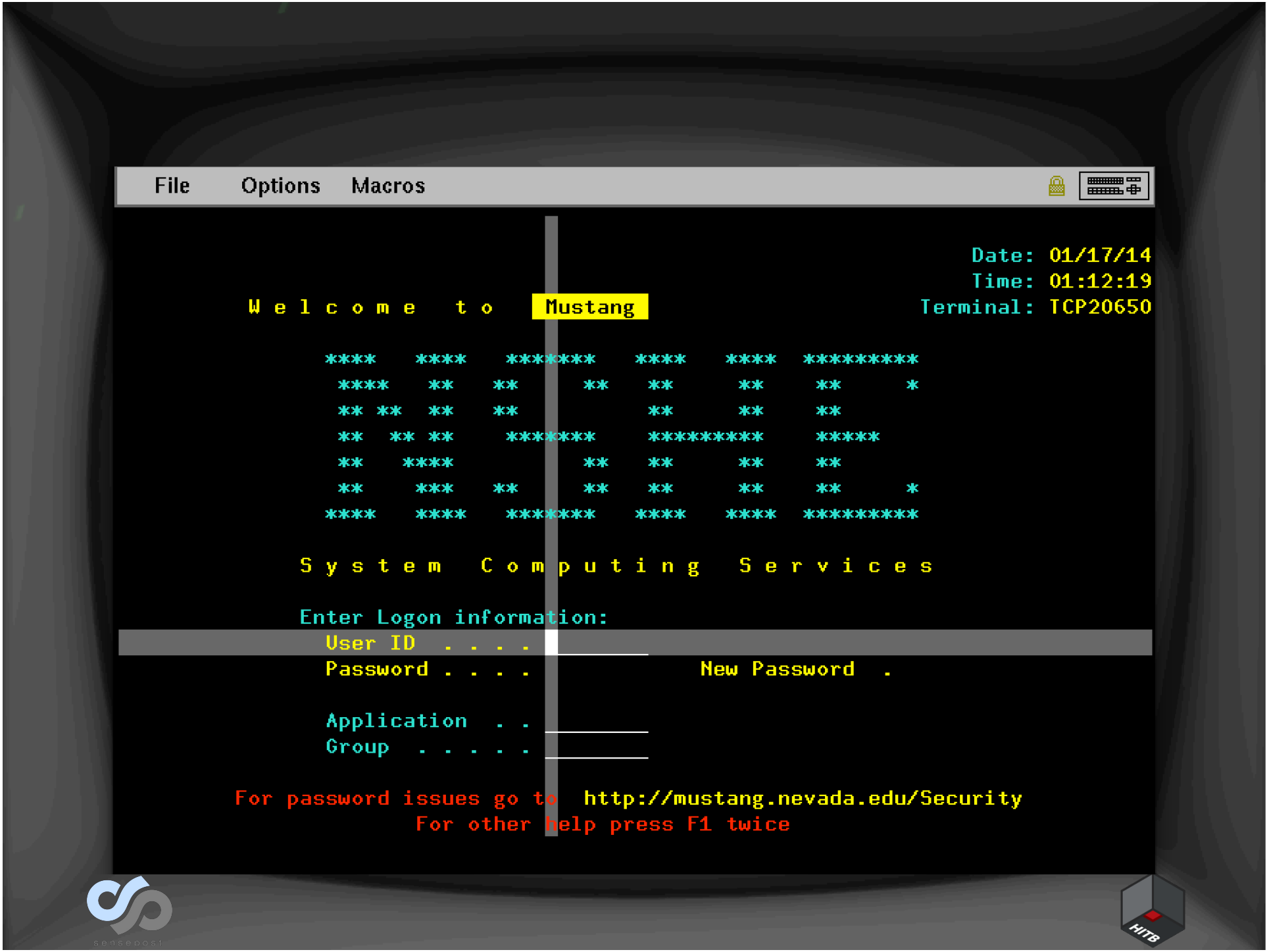
•USERNAME• ____________ •

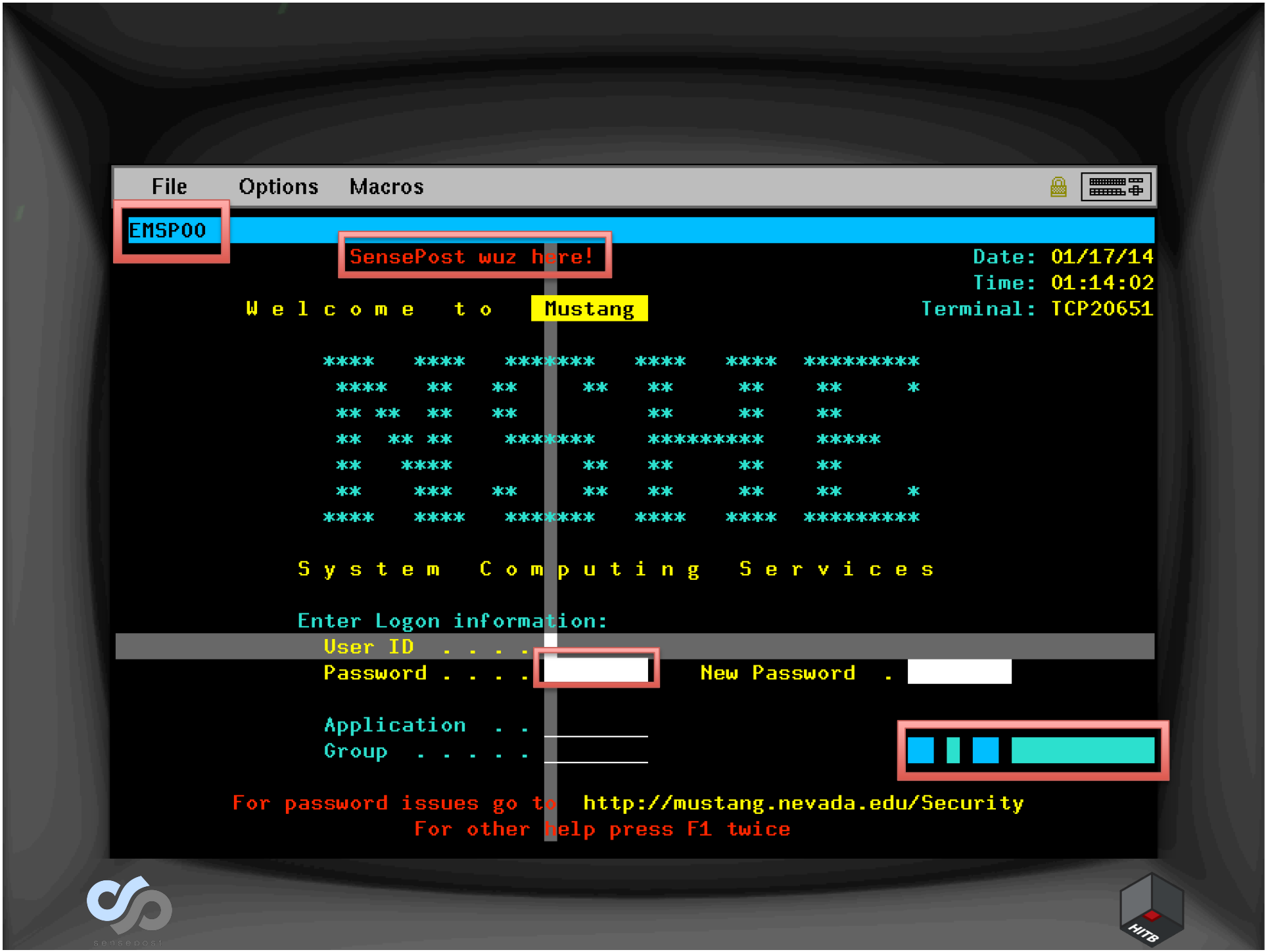
Un-Protected Field

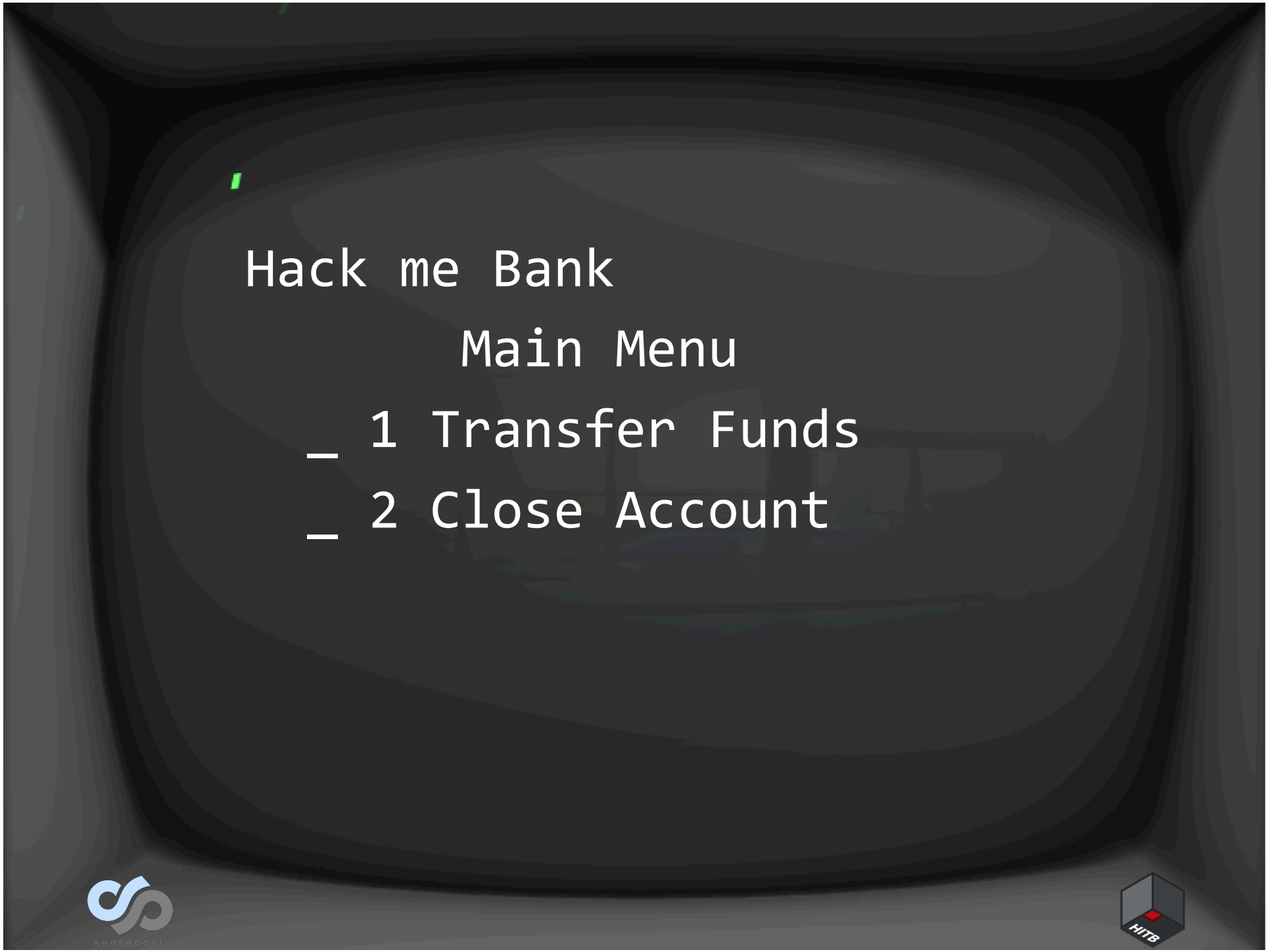
•PASSWORD• ____________ •

Un-Protected & Hidden

- This is all managed by the client
- So, hack our TN3270 emulator to:
  - Allow editing of PROTECTED fields
  - Show HIDDEN or “non-display” fields
- Works gangbusters!
- Three patches for x3270
  - Allow editing of protected fields
  - Show hidden fields
  - Combined patch


File Options Macros
Date: 01/17/14
Time: 01:12:19
Terminal: TCP20650
Welcome to Mustang
System Computing Services
Enter Logon information:
User ID . . . .
Password . . . .
New Password .
Application . .
Group . . . . .
For password issues go to http://mustang.nevada.edu/Security
For other help press F1 twice


File
Options
Macros
EMSP00
SensePost wuz here!
Date: 01/17/14
Time: 01:14:02
Terminal: TCP20651
Welcome to Mustang
System Computing Services
Enter Logon information:
User ID . . . .
Password . . . .
New Password .
Application . .
Group . . . . .
For password issues go to http://mustang.nevada.edu/Security
For other help press F1 twice


Hack me Bank
Main Menu
_ 1 Transfer Funds
_ 2 Close Account

Auth Bypass

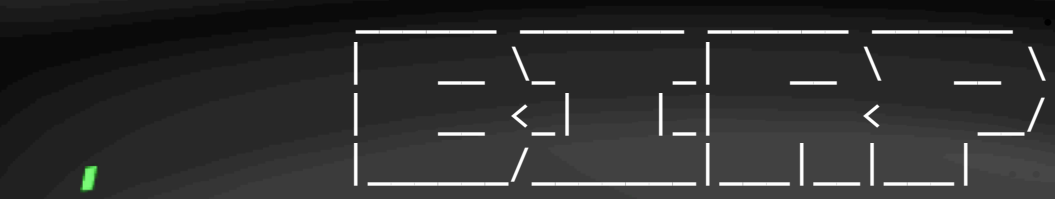
- P2 Hack me Bank
  - Main Menu
    - _ 1 Transfer Funds
    - _ 2 Close Account
    - 3 Free Money

Invoke restricted function

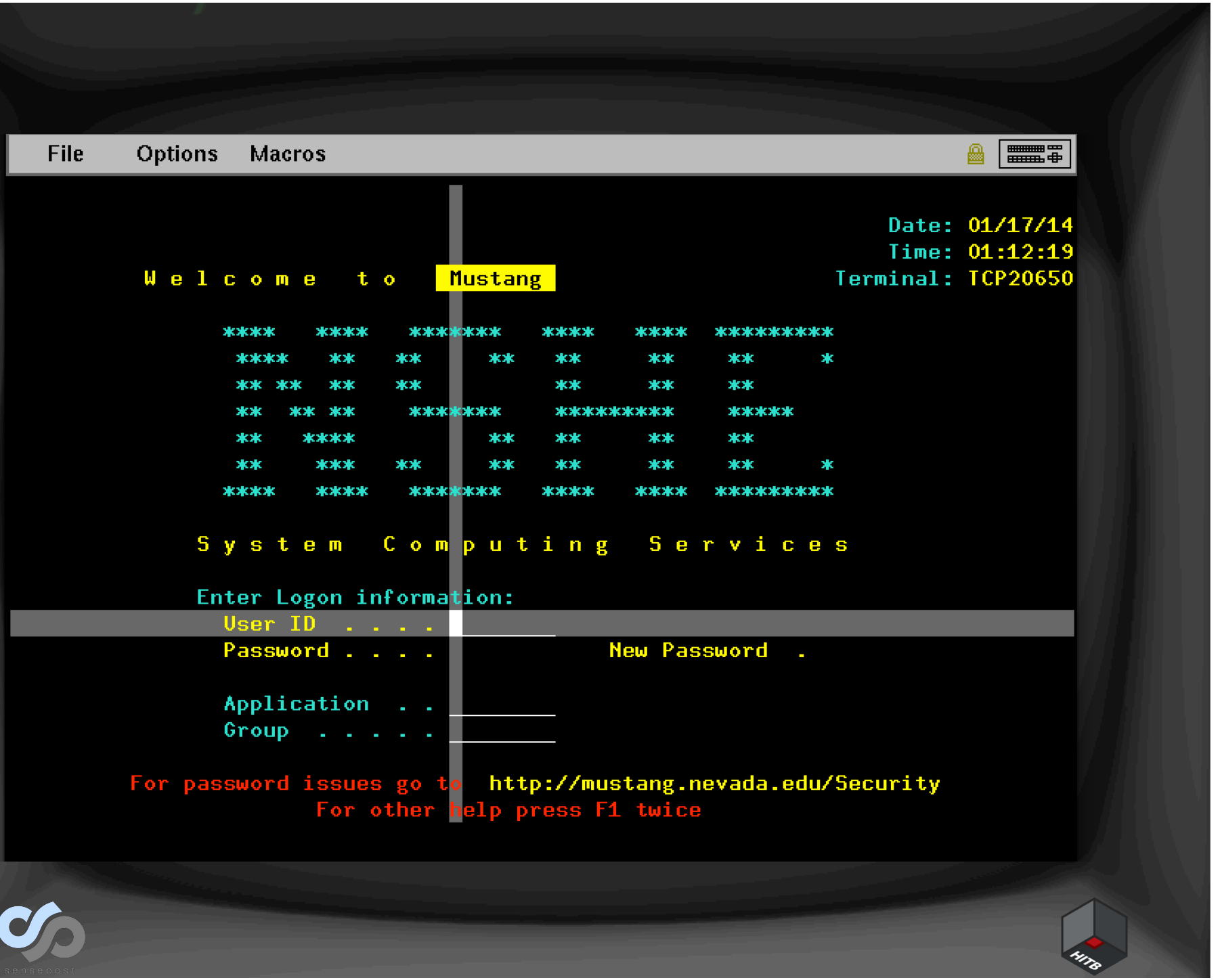
- LUSER

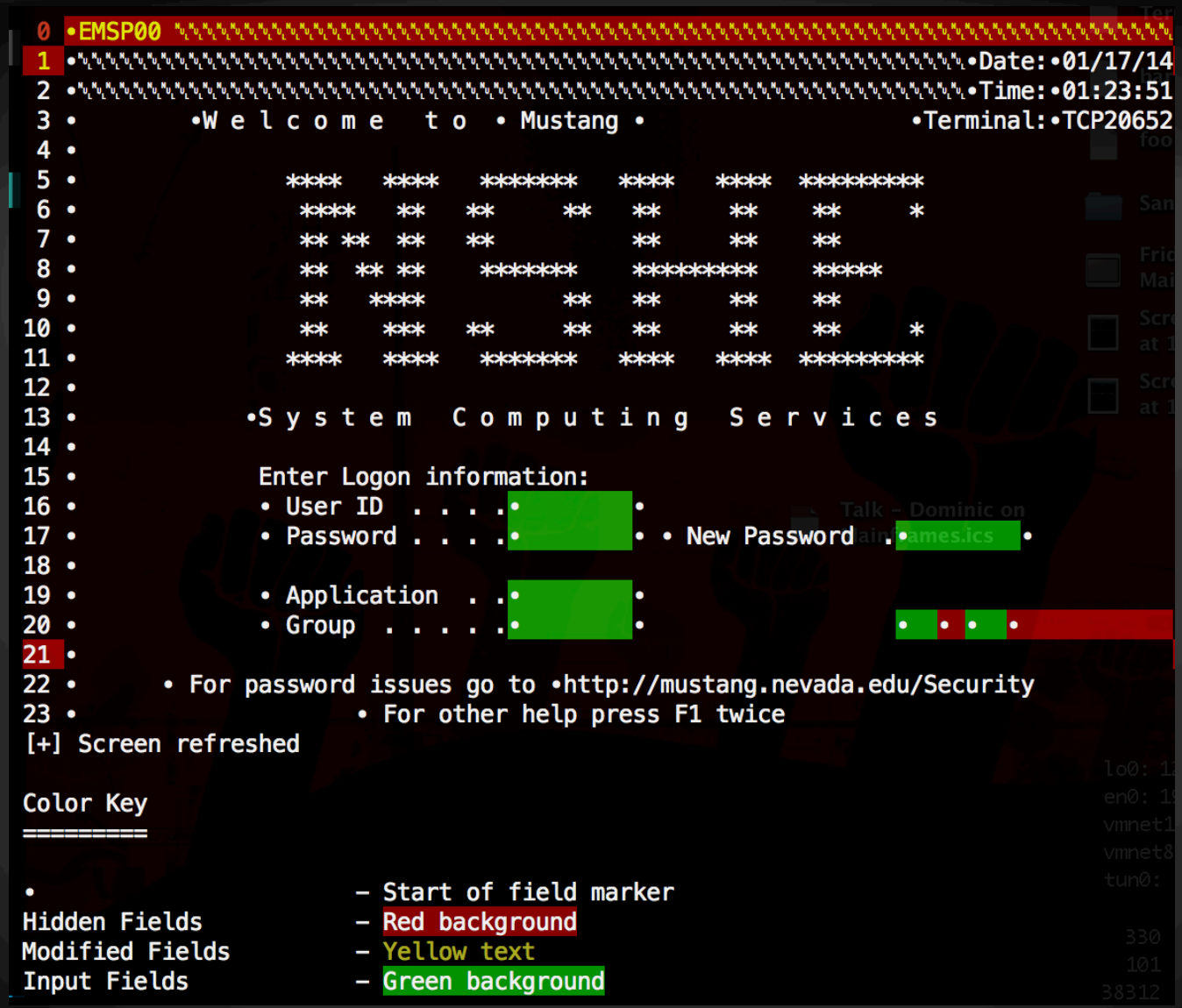
Access other accounts

- Eerily similar to web application
  - Hidden fields
  - Modifying un-modifiable inputs
  - But, markup & protocol not separated here
- Bypass develop assumptions
- Vuln exists everywhere developer made bad assumption
- New family of mainframe application vulnerabilities?



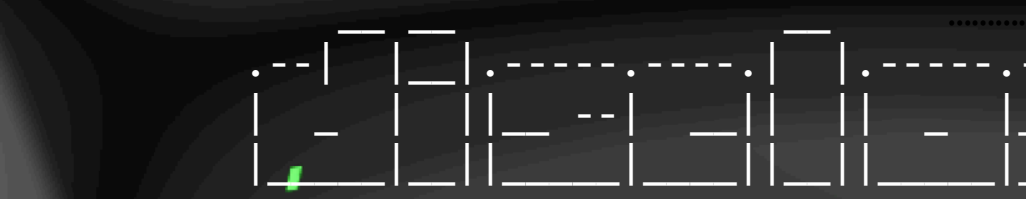
- Vulns are similar to webapps, perhaps the tool should be too
- Introducing Big Iron Recon & Pwnage (BIRP)
  - Thanks for the name Andreas (@addelindh)!
- Provides a "companion/hacker view" to your emulator
- Records transactions


File
Options
Macros
Date: 01/17/14
Time: 01:12:19
Terminal: TCP20650
W e l c o m e t o Mustang
S y s t e m C o m p u t i n g S e r v i c e s
Enter Logon information:
User ID . . . .
Password . . . .
New Password .
Application . .
Group . . . . .
For password issues go to http://mustang.nevada.edu/Security
For other help press F1 twice


•EMSP00
•Date:•01/17/14
•Time:•01:23:51
•Terminal:•TCP20652
•W e l c o m e t o • Mustang •
**** **** ******* **** **** *********
**** ** ** ** ** ** ** *
** ** ** ** ** ** **
** ** ** ******* ********* *****
** **** ** ** ** **
** *** ** ** ** ** ** *
**** **** ******* **** **** *********
•S y s t e m C o m p u t i n g S e r v i c e s
Enter Logon information:
• User ID . . . .•
• Password . . . .•
• New Password .•
• Application . .•
• Group . . . . .•
• For password issues go to •http://mustang.nevada.edu/Security
• For other help press F1 twice
[+] Screen refreshed
Color Key
=========
• - Start of field marker
Hidden Fields - Red background
Modified Fields - Yellow text
Input Fields - Green background

- Interactive mode
- Transaction recording
- Transaction analysis
- Save/Load history
- Playback
- Finding transactions
- Cool python objects

- IBM contacted me in Jan after seeing the HITB abstract (proactive)
- Detailed info about vulnerability in NVAS given
- Patched/Fixed (?) in February
- No public disclosure or acknowledgment by IBM
- Not even sure if it's disclosed privately or fix only
- If you're a customer, check your NVAS APARs on Security Zone or open a PMR
- As for wider issue …



- Obscurity discourages people looking, but doesn't prevent the finding
  - Handful of days and as a tourist I found a critical vuln across all apps
- IBM chooses not to disclose (need to know) to prevent attacks from being made public
  - Some merit in this (publish == inc in attack)
  - World has gone the other way – bug bounties
- I found this in a handful of days, initial hack was 45mins of work, as a n00b
  - Highly likely others have discovered this in 30+ years, not disclosing != no exploits

- Run your own mainframe
  - http://pastebin.com/PHiT8jmE
  - Piracy ☹
- Online DeZhi instance
  - http://zos.efglobe.com
- Soldier of Fortran/Mainframed/Phil Young
  - http://mainframed767.tumblr.com
  - @mainframed767
  - YouTube
- IBM “Master the Mainframe” course
- BIRP
  - https://github.com/sensepost/birp

- Phil Young++
- Unnamed customers
- IBM
- HITB
- Andreas Lindh