TechU

IBM

My adventures with TCP/IP port security and RACF on z/OS – Advance
Topics: A user experience

Joel Tilton, CISSP
RACF Engineer
Mainframe Evangelist

2019 IBM Systems Technical University
2 May Atlanta GA

About Joel Tilton, CISSP

- Joel Tilton is a former employee of IBM, where he got his start with mainframes, who continues to champion mainframe security issues and solutions.
- Over 20+ years technical IT experience, the majority of which was gained in hands-on technical roles, performing a variety of duties in diverse and complex environments.
- The majority of Joel's experience is focused on IBM mainframe systems, where he performs as a Technician and Project Manager. Joel's specialist subject is IT Security, in particular z/OS and associated subsystems (CICS, DB2, MQ, zSecure, etc.) security with RACF.
- Joel is also an active member of the Tampa Bay RUG (RACF User Group) which meets jointly with the NY RUG. Joel has a true passion for security and the mainframe. Long live the mainframe!
- https://www.linkedin.com/in/joeltilton
- RACFEngineer@gmail.com
- 702-483-RACF (Google Voice)

Disclaimers

- All products, trademarks, and information mentioned are the property of the respective vendors.
- Mention of a product does not imply a recommendation.
- Always test new profiles on a non-production system.
- Only you can prevent IPLs…
- The views expressed are his own personal views, and are not endorsed or supported by, and do not necessarily express or reflect, the views, positions or strategies of his employer

Agenda

- So what is a port anyway?
- Why Port Security with RACF
- UDP Unreserved Ports Are Easy with PTF UI9430 for z/OS 2.1
- TCP Unreserved Ports
  - WAS, Omegamon
- Port 20 & Program Control
  - Are there other ways to secure Port 20?
- z/OS FTP Client
  - Virtual Keyrings
  - Parm Search Order
  - Interesting parms including AT TLS
- TCP Unreserved Ports – WHENBIND
- Implementation Strategy
- A Simple CARLa
- zSecure Alert & WTOs
- How to simplify RACF Troubleshooting
- Some Stats
- Summary

What is a Port?

- An IP address is used to route the message to your computer. Once it arrives there, TCP uses the port number to know which program like ftp or email to hand it to
- From a SERVAUTH perspective…
  - Any mainframe program binding to
  - and/or listening on a TCPIP Port
  - SYS1.TCPIP.PROFILE

- Why?
  - Ensure ports can not be abused
  - Software can only bind and listen on assigned ports

Why Port Security with RACF?

NATIVE TCPIP

- Reservation by Jobname
- Can be spoofed
  - Unless JESJOBS profiles protecting jobnames
- Violations not well logged
- Unreserved ports not easily controlled
- Low Ports possibly protected with
- RESTRICTLOWPORTS
  - PORT JOBNAME reservation takes precedence
  - Did I mention JESJOBS?!

RACF

- Reservation by SAFNAME
- Cannot be spoofed
  - RACF profile FINAL answer
- Successes or Violations logged to SMF (type 80)
- Unreserved ports easily controlled
- Low Ports ALWAYS protected with RESTRICTLOWPORTS
  - EZB.PORTACCESS profiles take precedence

Unreserved Ports Syntax

- **`PORT UNRSV TCP * SAF UNRSVTCP`**
  - Prevent TCP port listeners → **TCP default**
- **`PORT UNRSV UDP * SAF UNRSVUDP`**
  - Prevent UDP port listeners & binds → **UDP default**

- Stop Unauthorized Port Use
  - Goal: Empty ACLs
  - **`AUDIT(ALL(READ)) UACC(NONE)`**

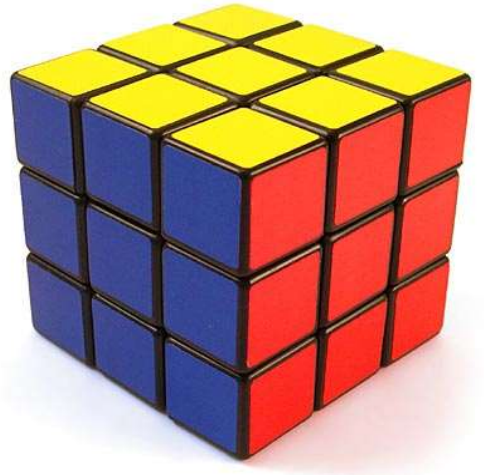
- Consideration: Ephemeral UDP ports
  - PTF UI9430 for z/OS 2.1 → Built into z/OS 2.2

Unreserved Ports RACF Profile Syntax

- Build SERVAUTH Profiles
  - **RDEFINE EZB.PORTACCESS.*.*.UNRSVTCP OWNER(…) UACC(NONE) WARNING AUDIT(ALL(READ))**
    - Permit all STCs in the Beginning
  - **RDEFINE EZB.PORTACCESS.*.*.UNRSVUDP OWNER(…) UACC(NONE) WARNING AUDIT(ALL(READ))**
- Read SMF, Read SMF, Read SMF
  - Logstring Your New BFF
- Reserve Port in SYS1.TCPIP.PROFILE
- Adjust Software Parms, If Necessary
- Obey / IPL
- Many Months later when nothing is accessing these profiles remove WARNING

UDP Unreserved Ports – SOLVED

- Applications Need Ephemeral UDP Ports
  - Bind to port 0, stack assigns port
- STC wants to send E-Mail
- STC opens UDP Ephemeral port on demand
  - SMTP
- Triggers SAF call unless…
  - PTF UI9430 for z/OS 2.1
  - Built in for z/OS 2.2

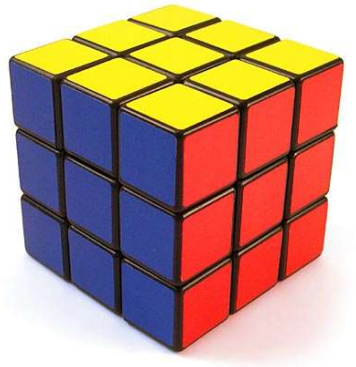


TCP Unreserved Ports – WAS Port Scans

- PM96838
  - Available for WAS v7.0, v8.0 and v8.5
  - Optionally disable port activity checking when a server is created
  - **com.ibm.ws.management.suppressPortScan=true**
    - JVM argument is added to suppress port check
  - Note that when this is in effect, ports in use by other applications will not be detected and could lead to port conflicts.
- PI40568
  - Available in Fix Pack 8.5.5.8
  - com.ibm.ws.management.suppressPortScan set to true is not stopping port scanning done in getUniquePort() in EndpointConfigHelper.java

TCP Unreserved Ports – Omegamon

- Binds ports to loopback (127.0.0.1)
  - Latest PTF UA68940 for v6.23.03
  - Next PTF for v6.30 fp6 level will include fix
  - New Parm KDE_LOOPBACK_POOL
  - IBM Recommends 1,000 ports… but wait there's more!



- Multiples of 4096 + a base number
- 1918 + (n * 4096) = Agent Port Number
  - Where N = The startup agent number.
- 1918 – Base Port, always assigned to hub or remote TEMS (Omegcms)
- 1920 – IBM Tivoli Monitoring Service Console (assigned to first agent to start up, OMEG*)
- 6014 – MVS Agent
- 10110 – CICS Agent
- 14206 – Network Agent

TCP Unreserved Ports – Omegamon

```
 6014 TCP * SAF OMEGAMON ; OMEGAMON
10110 TCP * SAF OMEGAMON ; OMEGAMON
14206 TCP * SAF OMEGAMON ; OMEGAMON
18302 TCP * SAF OMEGAMON ; OMEGAMON
22398 TCP * SAF OMEGAMON ; OMEGAMON
26494 TCP * SAF OMEGAMON ; OMEGAMON
30590 TCP * SAF OMEGAMON ; OMEGAMON
34686 TCP * SAF OMEGAMON ; OMEGAMON
38782 TCP * SAF OMEGAMON ; OMEGAMON
42878 TCP * SAF OMEGAMON ; OMEGAMON
46974 TCP * SAF OMEGAMON ; OMEGAMON
51070 TCP * SAF OMEGAMON ; OMEGAMON
55166 TCP * SAF OMEGAMON ; OMEGAMON
59262 TCP * SAF OMEGAMON ; OMEGAMON
63358 TCP * SAF OMEGAMON ; OMEGAMON
PORTRANGE  1850        101 TCP *        SAF OMEGAMON
PORTRANGE  1850        101 UDP *        SAF OMEGAMON
PORTRANGE  19000       101 TCP *        SAF OMEGAMON
PORTRANGE  19000       101 UDP *        SAF OMEGAMON
```

Secure 20 & 21 WHEN(PROGRAM)

- Use conditional access list to ensure only the **exact** programs invoked can use Port 20 & 21
- For the Data Port TCP 20
  - `permit EZB.PORTACCESS.*.*.FTPDATA class(SERVAUTH) id(xxxxxxxx) access(READ) when(PROGRAM(FTPDNS))`
- For the Control Port TCP 21
  - `permit EZB.PORTACCESS.*.*.FTP class(SERVAUTH) id(FTP_STC_User) access(READ) when(PROGRAM(FTPD))`

Further Tighten Port Security using Conditional Access

- Use of conditional access would be the only way to ensure ports aren't abused by the software to which they're assigned
  - Today type 80 records do **not** record program name
- Difficult to find the program you're looking for…
  - Trial & Error by adding a "bogus" program name to a conditional access list entry
    - Then play a "fun" game called "cycle the STC during the maintenance window"
  - `RDEFINE FACILITY IRR.PGMSECURITY APPLDATA('ENHWARN')`
  - `SETR RACLIST(FACILITY) REFRESH`
  - `SETR WHEN(PROGRAM) REFRESH`
  - `SETRLIST` ➔ `WHEN(PROGRAM -- ENHANCED WARNING)`
- **ICH432I CONDITIONAL ACCESS LIST FOR *class-name resource-name* DID NOT GRANT AUTHORITY TO PROGRAM(S): *program-name program-name2 program-name3***

Port 20 – Making the World Safer

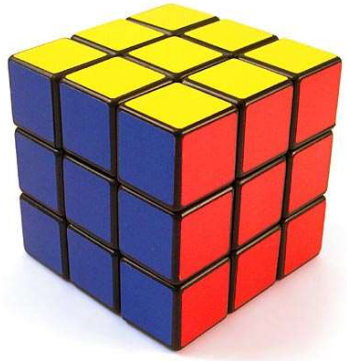
- SERVAUTH does not support RACF-DELEGATED…yet ☺
  - Request For Enhancement (RFE)
  - https://www.ibm.com/developerworks/rfe/execute?use_case=viewRfe&CR_ID=75166
  - Please VOTE! Your Vote Matters!

TCP Unreserved Ports – z/OS FTP Client PTF

- If Passive FTP fails, z/OS FTP Client attempts Active connection by default
  - Active connection Binds & Listens on…
  - Random TCP port! ICH408I!

```
ICH408I USER() GROUP() NAME()
EZB.PORTACCESS.SYSTEM.TCPIP.UNRSVTCP CL(SERVAUTH)
INSUFFICIENT ACCESS AUTHORITY
FROM EZB.PORTACCESS.*.*.UNRSVTCP (G)
ACCESS INTENT(READ    )  ACCESS ALLOWED(NONE    )
```

- APAR PI36683 PTF UI27396 for z/OS 2.1
  - http://www-01.ibm.com/support/docview.wss?uid=isg1PI36683
- New FTP.DATA Parm:
  - **PassiveOnly TRUE**
- Add Existing Parm too:
  - **FWFRIENDLY TRUE**



Use Virtual Keyring with FTPS

- Reduced keyring maintenance for FTPS users
- More Easily Widen use of FTPS
- **KEYRING *AUTH*/***
  - Use any CA (certificate authority) certificate in trust status
- **RDEFINE RDATALIB CERTIFAUTH.IRR_VIRTUAL_KEYRING.LST UACC(NONE) AUDIT(FAILURES(READ))**
  - Secures access to the Virtual Keyring
  - RACF Callable Services Manual

FTP Client Parm Search Order

- https://www.ibm.com/support/knowledgecenter/SSLTBW_2.3.0/com.ibm.zos.v2r3.halz001/ftpconfigstatementsinftpdata.htm

TSO shell	UNIX System Services shell
1. -f	1. -f
2. SYSFTPD DD statement	2. $HOME/ftp.data
3. tso_prefix.FTP.DATA	3. userid.FTP.DATA
4. userid.FTP.DATA	4. /etc/ftp.data
5. /etc/ftp.data	5. SYS1.TCPPARMS(FTPDATA) data set
6. SYS1.TCPPARMS(FTPDATA) data set	6. tcpip_hlq.FTP.DATA file
7. tcpip_hlq.FTP.DATA file	

Recommend z/OS FTP Client Parms

- **CLINTERRORCODES EXTENDED**
  - Request FTP client detailed return codes
  - Easy to cross reference in manuals
- **CLIENTEXIT TRUE**
  - Always exit with the true return code
- **LOGCLIENTERR TRUE**
  - Adds message EZZ9830I to top of batch job detailing which FTP command failed
  - **EZZ9830I ABCD1234 FTP failed - Cmd = 26(pass) Reply = 501 EX CEE RC = 1126**

Still More z/OS FTP Client Parms

- ISPFSTATS TRUE
- **KEYRING *AUTH*/***
- LOGCILENTERR TRUE
- **PASSIVEONLY TRUE**
- **PASSIVEINGOREADDR TRUE**
- SECURE_DATACONN PRIVATE
- SECURE_CTRLCONN PRIVATE
- SECURE_FTP REQUIRED
- SECURE_MECHANISM TLS

AT TLS z/OS FTP Client Parms

- SEQNUMSUPPORT TRUE
- **TLSMECHANISM ATTLS**
- **TLSRFCLEVEL RFC4217**
- No need for KEYRING statement
- z/OS Communications Server: IP Configuration Reference
- https://www.ibm.com/support/knowledgecenter/SSLTBW_2.3.0/com.ibm.zos.v2r3.halz001/filetransportprotocol.htm

Debugging z/OS FTP Client Parms

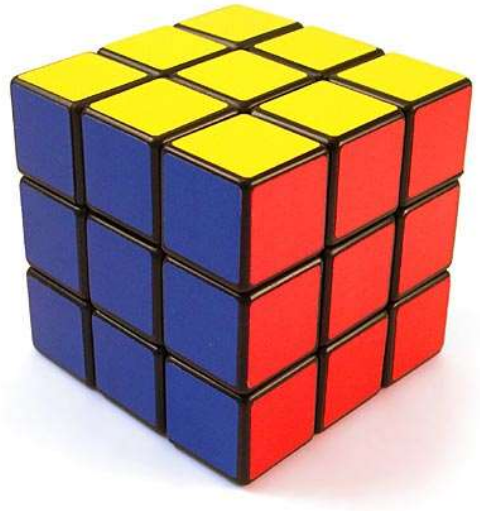
- **`CHKCONFIDENCE True`**
  - Check & Report on Confidence level of transfer
  - EZA2108I Confidence=High for PUT of …
- **`DEBUG SEC`**
  - Activate security tracing
  - Shows entire processing of TLS handshake
  - Very helpful for debugging
  - Only one trace can be active at a time

TCP Client Bind Security – WHENBIND

- Purpose: Call SAF for all TCP client binds
  - Client needs to request an ephemeral port by binding to port **0**
- Why?
  - Do you really trust a TCP client to be in control of what port it can use?
- **PORT UNRSV TCP * SAF UNRSVTCP WHENBIND**
  - WARNING or UACC(READ) **RECOMMENDED**
  - **RDEFINE SERVAUTH EZB.PORTACCESS.*SYSNAME*.*.UNRSVTCP WARNING UACC(NONE) AUDIT(ALL(READ))**
    - i.e. One system at a time
  - Read SMF…Parse LOGSTRING for Port
- Plan of Attack: One Software product at a time

TCP WHENBIND – SAS

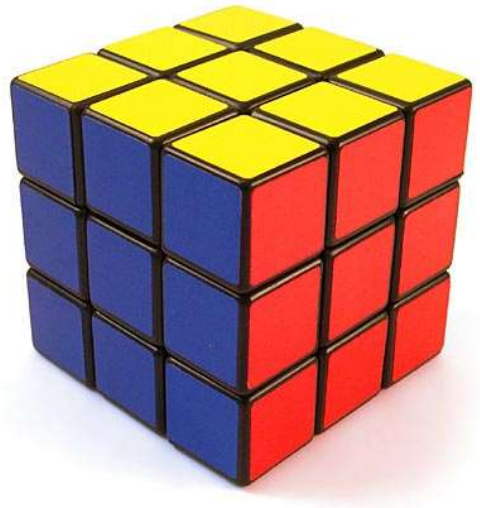
- SAS pre v9.4
  - E-mail engine uses ports
  - TCP_EPH_MAP_ENABLED=**0** ➔ zero
  - TKMVSENV DD
    - hlq.TKMVSENV(TKMVSENV)



- SAS v9.4
  - TCP_EPH_MAP_ENABLED=1
  - TCP_EPH_MAP_FIRST=xxxx
  - TCP_EPH_MAP_LAST=yyyy
  - permit EZB.PORTACCESS.*ABCD*.*.SAS class(SERVAUTH) id(xxxxxxxx) access(READ) when(PROGRAM(SAS))

TCP WHENBIND – Challenges

- VPS
  - Remove parm → TCPHOSTS
  - Remove parm → TCPPORT
  - How is VPS used?
- FTPS
  - Must reserve data port



Implementation Strategy

- Apply required PTFs
- Activate SERVAUTH class (RACGLIST too!)
- Known TCP & UDP Ports – Phase 1
  - Profiles in WARNING as appropriate
- Constant monitoring of SMF to catch Rare Port Listeners – Phase 2
- Secure Unreserved UDP ports – Phase 3
- Secure Unreserved TCP ports – Phase 4
  - Goal: Empty ACLs for Unreserved ports
- Secure Unreserved TCP client binds – Phase 5
  - Because we shouldn't trust TCP software to be in control of ports

Validate Ports Reserved with SAF

- Detect Port Reservations without SAF
- Single Batch checks all systems using CKNSERVE

```
alloc type=CKFREEZE          active zsecsys=*

NEWLIST type=ip_port name=IPPORTCK,
TT='WARNING!  Port Reserved without SAF
keyword!',
Emptylist=SHOW

EXCLUDE exists(resname)

SORTLIST system end_port protocol count jobname
resname bind options
SUMMARY begin_port count(nd)
```

Use zSecure Alert to Drive WTO

- Key information for SERVAUTH like IP and Port only recorded in logstring
- Many teams that now get ICH408I messages need to see that information
- Custom zSecure Alert Drives WTO with logstring

CARLa to Drive WTO

```
)CM Pass one query
)SEL &C2PEPASS = Y
)ENDSEL
)CM Alert condition
)SEL &C2PEPASS = N
)IM C2PSGNEW
 S  likelist=recent event=access(failure),
    class=($C4RVFY,APPCPORT,APPL,FACILITY,JESSPOOL,OPERCMDS,
    SERVAUTH,VTAMAPPL,XFACILIT)
)CM Action command
)IM C2PSACTX
)IM C2PSACTS
)CM WTO sortlist
)SEL &C2PERCTP = WTO
sortlist recno(nd),
 'C2P&c2pemem.&c2peflag' 'Date: ' | date(0) | ' Time: ' | time(12) /,
'.  User(' | user | ') Group(' | user:dfltgrp |,
') Name(' | name | ')' /,
'.  ' | resource(0) 'CL(' | class | ')' /,
'.  Insufficient Access Authority' /,
'.  From ' | profile(0) | /,
'.  Access Intent(' | intent | ') Access Allowed(' | access | ')' /,
'.  CreateDate:'(15,ne) | :creadate(0) /,
'.   Prof Type:'(15,ne) | :proftype(0) /,
'.   Logstring:'(15,ne) | logstr(0,hor,wordwrap) /,
'.   Job Name-:'(15,ne) | jobname /,
'.   Job ID---:'(15,ne) | jobid   /,
'.   Terminal-:'(15,ne) | terminal
)ENDSEL
)ENDSEL
```

Configure zSecure Alert

```
Command ===>

Description  . . WTO: Failed Access on SERVAUTH Class
Member prefix    CKR
Alert id . . . . 4099  Severity . . . . I  (D, I, W, E or S)
Data source  . . SMF
Parameters . . .
Panel name . . .           (Panel for additional customization)

Specify SMF records to be collected for this alert
Type Sub    Type Sub    Type Sub    Type Sub    Type Sub
80

Specify WTO filters for this alert
Prefix    Prefix    Prefix    Prefix    Prefix

Allowable destination types      E-mail     Cellphone     SNMP   /   WTO
                                 Unix Syslog              Action command
Specify action  . . . . . . . N  (Y/N)
Extended Monitoring alert . . N  (Y/N)
View/edit the alert skeleton  _  ISPF Skeleton CKRS4099
```

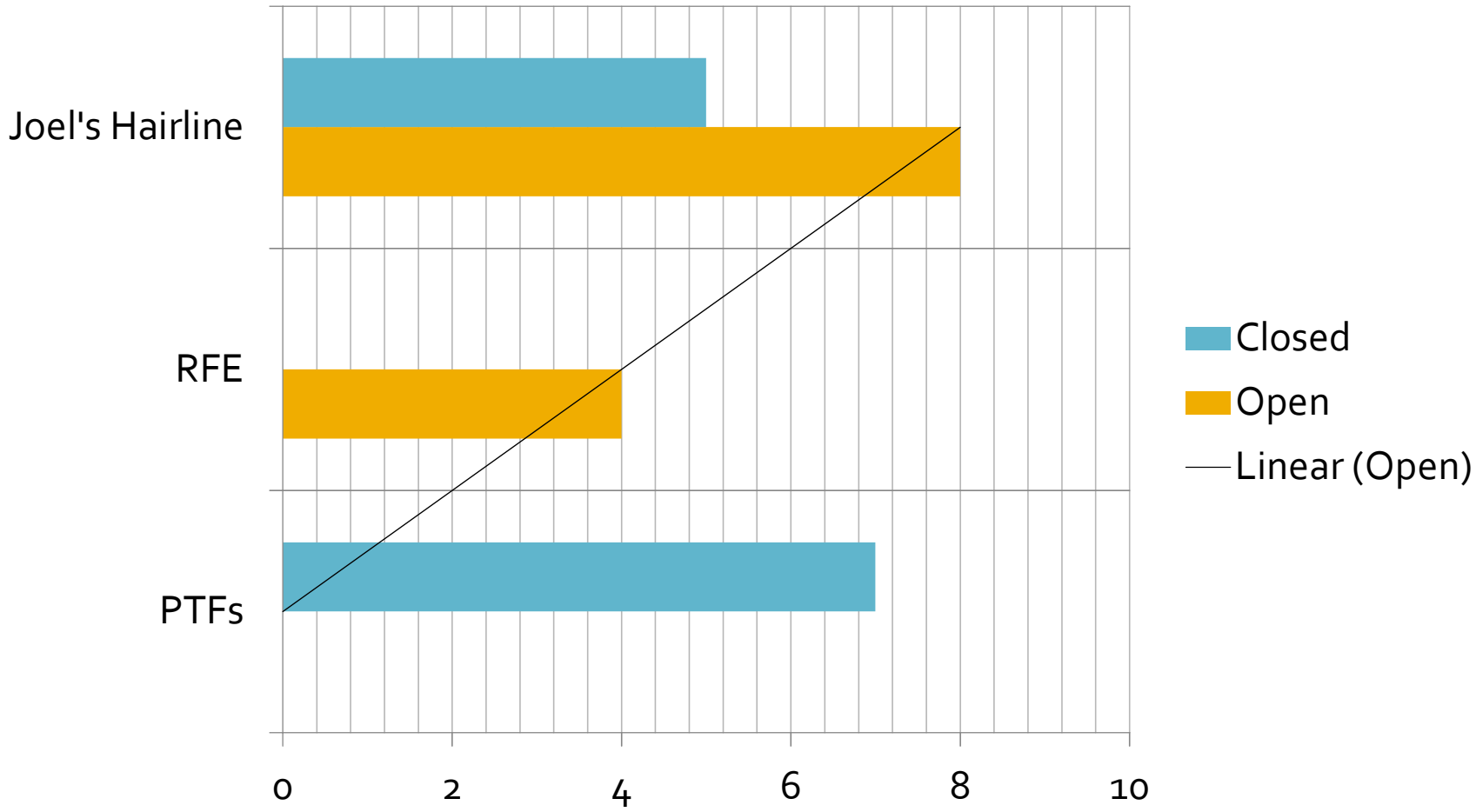
Simplify RACF Troubleshooting

- Some RACF checks are cross-memory and will **only** show in the STC log
  - ➔SERVAUTH
- Use automation software to pick up WTOs
- Message IDs
  - C2P4001 – 4099
  - Suppress in MPFLSTxx ➔ C2P4*,SUP(ALL)
- Store in DD card in automations software address space
- Centralized reporting of ICH408I & zSecure Alert WTO Enhanced messages
- ICH408Is
- DSNL030I
  - Contains IP address!

Additional Resources

- Techdocs Library – Using SERVAUTH to Protect TCP Port Usage
  - http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP100673
- Techdocs – Undesired PortAccess Violations
  - http://www-01.ibm.com/support/docview.wss?rs=852&uid=swg21237916
- Port Access Control Chapter
  - z/OS Communications Server: IP Configuration Guide
  - https://www.ibm.com/support/knowledgecenter/SSLTBW_2.3.0/com.ibm.zos.v2r3.halz002/security_tcpip_resrcs_ports.htm
- SERVAUTH Class profiles used by TCP/IP
  - EZB.PORTACCESS syntax
  - https://www.ibm.com/support/knowledgecenter/SSLTBW_2.3.0/com.ibm.zos.v2r3.halz002/security_tcpip_resrcs_saf.htm

Some Statistics – 2013 to 2015



Summary

- The journal of a thousand miles beings with a single step
- Protecting Unreserved Ports is of Paramount Im**PORT**ance
  - Securing with RACF
    - prevents spoofing
    - logs port usage (success & failures) to SMF
- Requires Proper Planning
- Close partnership with Network Engineer
- Coordinate TCPIP Profile & RACF Changes
- IPL during maintenance windows
- Fix ICH408Is and:
  - Recycle STC or possibly IPL
- Port Security Engaged!

My Thanks To…

- Bob Hansel
- Stu Henderson
- Adam Klinger
- Ray Kohring
- Christopher Meyer
- Carolyn Miller
- James Ovincy
- Howie Odishoo
- Todd Valler
- William Vender
- Bruce Wells
- Daniel Zentner

- IBM Omegamon Level 2 & Level 3
- IBM z/OS Comm Server/TCPIP Level 2 & Level 3
- IBM zSecure Level 2 & Level 3
- IBM WAS Level 2 & Level 3

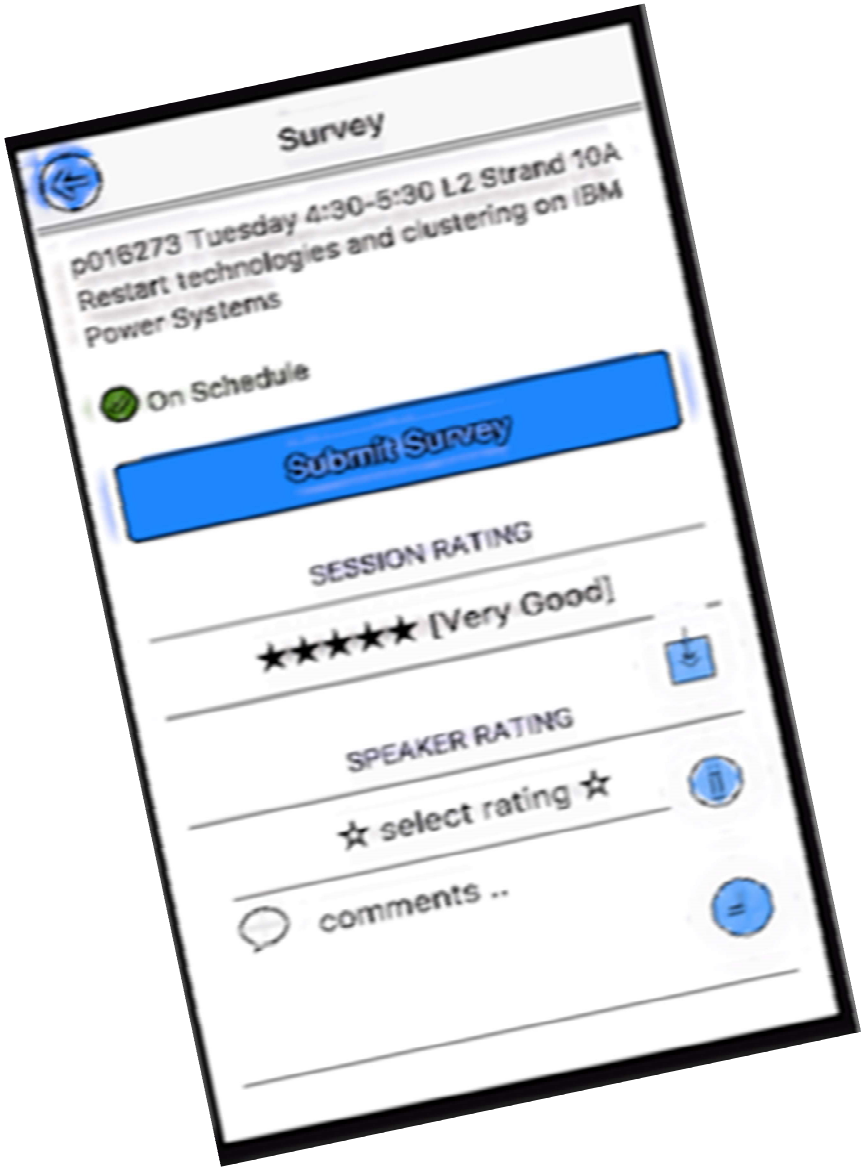
- And the Adventure Continues to Boldly Go Where No Port Has Gone Before …

- **DISCLAIMER:** No ports were harmed in the making of this presentation…perhaps shaken & stirred but they were not permanently damaged. ☺

Thank you!

Joel Tilton
Infrastructure Services Architect

RACFEngineer@gmail.com
+1-702-483-RACF



Please complete the Session Evaluation!

Questions?

Even more Useful Resources

- IBM z/OS V2R1 Communications Server TCP/IP Implementation: Volume 4 Security and Policy-Based Networking
  - http://www.redbooks.ibm.com/redbooks/pdfs/sg248099.pdf
- RESTRICTLOWPORTS parameter
  - https://www.ibm.com/support/knowledgecenter/SSLTBW_2.3.0/com.ibm.zos.v2r3.halz002/security_tcpip_resrcs_unresvd_ports_low.htm
- TCPIP PROFILE Port Assignments
  - https://www.ibm.com/support/knowledgecenter/SSLTBW_2.3.0/com.ibm.zos.v2r3.halz001/profiletcpipportassignments.htm
- IBM z/OS V2R2 Communications Server TCP/IP Implementation: Volume 4 Security and Policy-Based Networking
  - http://www.redbooks.ibm.com/abstracts/sg248363.html?Open

More z/OS FTP Client Parms

- AUTOTAPEMOUNT TRUE
- **BUFNO 35**
- CONDDISP DELETE
- **DATAKEEPALIVE 60**
- DSWATITIME 1
- EPSV4 TRUE
- **FTPKEEPALIVE 60**

Debugging z/OS FTP Client Parms

- **TRACE**
  - `DEBUG CMD`
    - Each command and parsing of the parms
  - `DEBUG INT`
    - Initialization & termination of FTP session.
  - `DEBUG FSC`
    - Processing file services server commands APPE, STOR, STOU, RETR, DELE, RNFR and RNTO.
    - For the client, shows details for subcommands GET, PUT APPEND, DELETE and RENAME
  - `DEBUG SOC`
    - Details of the processing during the setup of the interface between the FTP application and the network as well as how much actual data is processed.