Securing Your Job with JESJOBS

Joel Tilton
RACF Engineer
Mainframe Evangelist
May 2018

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) ← Because it's cool!

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

- Why Secure Jobnames?
- Where to Start ?
- Profile Syntax
- GLOBAL Class
- JESJOBS Default Return Code is 8
- RACFVARS
- Securing One Job Prefix
- JESJOBS vs. SURROGAT
- PROPCNTL
- Summary

Why Secure Jobnames ?

- Because we secure "stuff"?
  - No! Security should be based on a practical need
- Close an attack vector
  - Jobnames are a sensitive resource
  - Especially if you do not secure your ports!
    - See SERVAUTH EZB.PORTACCESS profiles
- Think about how many jobs that run which are:
  - Sensitive, use very confidential data
- TSO submit exit is not really security
- Yes you could use a JES exit
  - Do you really want another exit?

Where to Start ?

- At the beginning …
  - The journey of 1,000 miles begins with a single step
- What jobs run in your shop that are sensitive?
- Security team
- Data storage management
- CICS or IMS Regions
  - Prevent UserID propagation

JESJOBS Profile Syntax

SUBMIT.*nodename.jobname.execution_userid*

Qualifier	Description	Recommendation
nodename	JES node where the job runs	• Use * unless security per JES node necessary
jobname	Jobname from the jobcard	• Use * to group together similar jobs
Execution _UserID	The UserID that is actually running the job *NOT* the UserID that might have used IEBGENER to copy the job to the internal reader	• Code specific UserID to ensure proper security for job Scheduler UserIDs • Group by RACF variable where possible • Each jobname mask should be tied to a specific execution UserID for a set of jobs

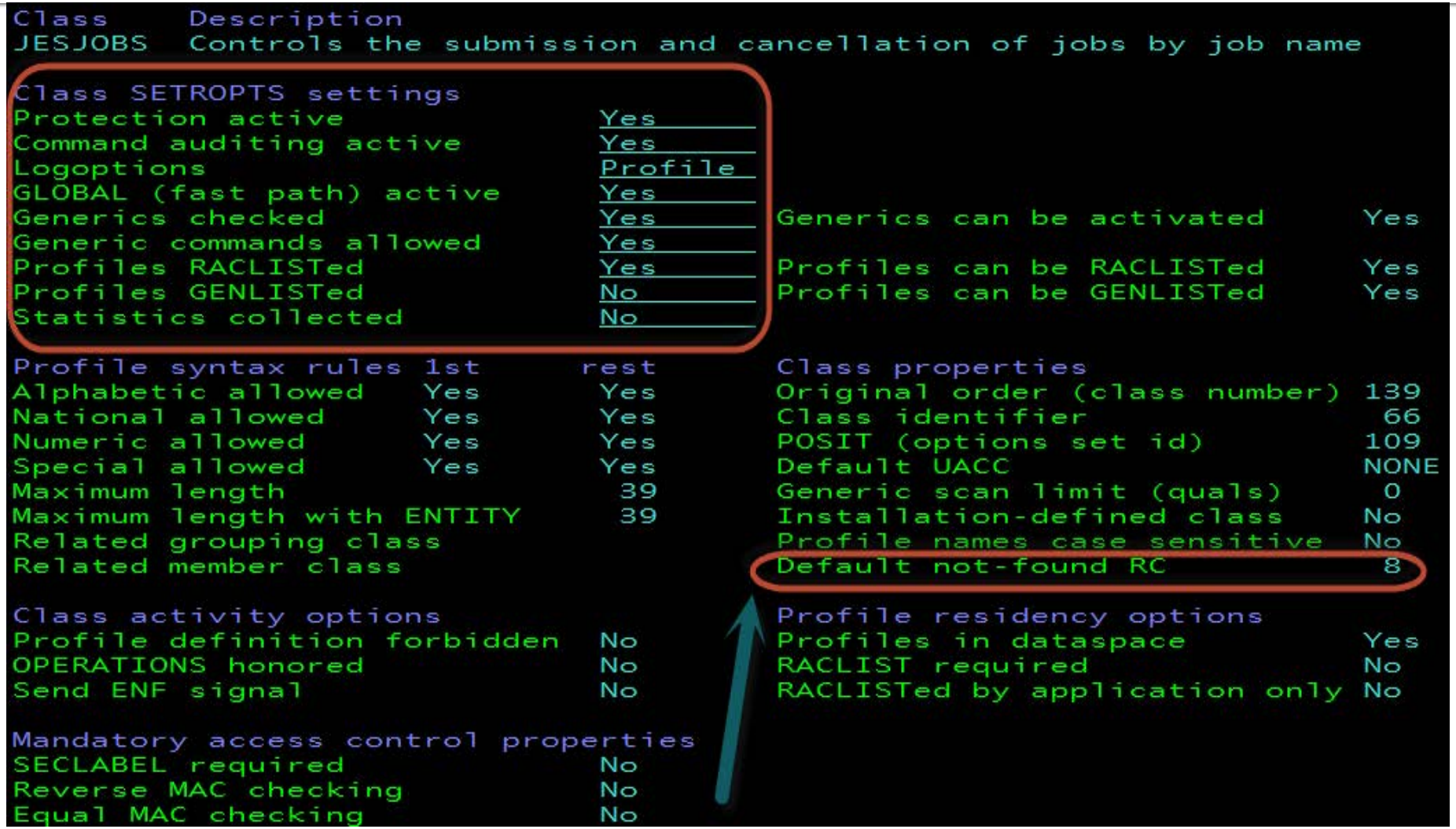
GLOBAL CANCEL.*.&RACUID.*/ALTER

- rdefine GLOBAL JESJOBS owner(#RA#RACF) audit(failures(READ)) uacc(NONE)
- ralter GLOBAL JESJOBS addmem(CANCEL.*.&RACUID.*/ALTER)
- setropts global(JESJOBS)
- setropts refresh global(JESJOBS)

Class Activation

- SETR CLASSACT(jesjobs) AUDIT(jesjobs) GENERIC(jesjobs) RACLIST(jesjobs)
  - GENERIC() means GENCMD too
  - However NOGENERIC does not include NOGENCMD
- SETR CLASSACT(jesjobs) AUDIT(jesjobs) GENERIC(jesjobs) GENCMD(jesjobs) RACLIST(jesjobs)

JESJOBS Default RC is 8



SUBMIT.** UACC(READ)

- RDEFINE SUBMIT.** **UACC(READ)** AUDIT(**ALL**(READ))
- SETR RACLIST(JESJOBS) REFRESH
- –OR–
- RDEFINE SUBMIT.** **UACC(READ)** AUDIT(**FAILURES**(READ))
  - If you have a tool to collect data from RACF Exit points
- SUBMIT.** recommended by JES development
  - **DO NOT define ** UACC(READ)!**
- Consider we really are only concerned about the SUBMIT resource for JESJOBS profiles

Recommended RACFVARS

- &JJOBSCH
  - Scheduling UserIDs
- &JJOBSTC
  - Started Tasks
  - RACF, ICSF, z/OS PKI
- &JJOBCPS
  - Jobs prefixes submitted by change man
- &JJOBZBK
  - Job prefixes that will be blocked from all other users
  - Think Backstop
- No RACF does not support system symbols
  - However I do not see how that would help with JESJOBS

JESJOBS Profile Syntax

SUBMIT.*nodename.jobname.execution_userid*

Qualifier	Description	Recommendation
nodename	JES node where the job runs	• Use * unless security per JES node necessary
jobname	Jobname from the jobcard	• Use * to group together similar jobs
Execution _UserID	The UserID that is actually running the job *NOT* the UserID that might have used IEBGENER to copy the job to the internal reader	• Code specific UserID to ensure proper security for job Scheduler UserIDs • Group by RACF variable where possible • Each jobname mask should be tied to a specific execution UserID for a set of jobs

Securing One Job Prefix

- SUBMIT.*.CKR*.*UserID*
  - Only Scheduler UserID has READ
  - Or any individuals you intend to submit jobs that start with jobname CKR*something* running with that scheduler UserID as the execution UserID.
- SUBMIT.*.CKR*
  - Protects all other jobnames starting with CKR
  - Access list depends upon how jobname is used
- Once these profiles are in place:
  - Any submission of a job starting with CKR will end with ICH408I unless you have access to the above profiles
  - IEBGENER to the SPOOL as well
  - JESJOBS sees all

Not Authorized to Submit Job

```
ICH408I USER(execution_UserID)
GROUP() NAME()
SUBMITTER(execution_UserID)

LOGON/JOB INITIATION - NOT
AUTHORIZED TO SUBMIT JOB CKRabcde
```

- In the type 80 you will find all of this information INCLUDING the **submitting** jobname
  - Which is critical
  - How do you know which job tried to run this job?
  - Think CICS or IEBGENER straight to the spool

Submitter is Not Authorized

```
ICH408I USER(execution_UserID)
GROUP() NAME()
SUBMITTER(execution_UserID)
LOGON/JOB INITIATION - SUBMITTER
IS NOT AUTHORIZED BY USER
```

- Typical ICH408I for SURROGAT class

Securing the Execution UserID

- `JESJOBS SUBMIT.*.CKR*.`*`UserID`*
  - Secure all jobs starting with CKR for Execution UserID
- `JESJOBS SUBMIT.*.CKR*`
  - Secure all jobs starting with CKR
- `JESJOBS SUBMIT.*.*.&JJOBSCH*`
  - Secure **all other** jobnames for the UserIDs in variable &JJOBSCH
- `JESJOBS SUBMIT.*.*.&JJOBSTC*`
  - Secure **all other** jobnames for the UserIDs in variable &JJOBSTC
- Code the generic in the profile!
  - NOT I repeat *NOT* in the RACFVAR

JESJOBS Profile Syntax

SUBMIT.*nodename.jobname.execution_userid*

Qualifier	Description	Recommendation
nodename	JES node where the job runs	• Use * unless security per JES node necessary
jobname	Jobname from the jobcard	• Use * to group together similar jobs
Execution _UserID	The UserID that is actually running the job *NOT* the UserID that might have used IEBGENER to copy the job to the internal reader	• Code specific UserID to ensure proper security for job Scheduler UserIDs • Group by RACF variable where possible • Each jobname mask should be tied to a specific execution UserID for a set of jobs

Putting it all Together

- **SUBMIT.*nodename.jobname.execution_userid***
- SUBMIT.*.CKR*.*UserID*
- SUBMIT.*.CKR*
- SUBMIT.*.*.&JJOBSCH*
- SUBMIT.*.*.&JJOBSTC*
- Remember RACF evaluates profiles to find the "best matching" profile for a resource

Changeman

- **SUBMIT.*nodename.jobname.execution_userid***
- SUBMIT.*.&JJOBCPS%%%%%%.*UserID*
- SUBMIT.*.&JJOB**Z**BK*
- SUBMIT.*.*.&JJOBSCH*
- SUBMIT.*.*.&JJOBSTC*
- &&JOBZBK Variables mission in life is a backstop for the job prefixes also used by Changeman

Wouldn't it be nice if …

- We could use **TWO** RACFVARS in the **SAME** RACF profile
- SUBMIT.*.&JJOBS*.&JJOBSCH*
  - Then I could cover all of my standard job prefixes and scheduling UserIDs with just **ONE** RACF profile!
- Open Discussion
- RFE ?
  - Remember to vote for RFEs!

JESJOBS Does Not Replace SURROGAT

- In the CKR example
- In order to be able to submit as the scheduler UserID of the JESJOBS SUBMIT.*.CKR*.*UserID* profile
- Access to the SURROGAT class *UserID.SUBMIT* profile is ***STILL REQUIRED***

PROPCNTL

- Should we still be using PROPCNTL after deploying JESJOBS?
  - Open Discussion
- RDEF PROPCNTL *UserID*
- For a profile in the PROPCNTL class, RACF checks only for the presence or absence of a profile in this class. If a profile exists for a particular user ID, user ID propagation does not occur for that user ID.
- RACF performs no logging and issues no messages for profiles in the PROPCNTL class.
- https://www.ibm.com/support/knowledgecenter/en/SSLTBW_2.1.0/com.ibm.zos.v2r1.icha700/prpctl.htm

Summary

- Try not. Do…or do not. There is no try!
- JESJOBS is not complicated just a bit time consuming
  - How many jobs do you wish to secure?
- Start with critical jobs
  - Security team, data storage
  - Payroll, sensitive data
- Plan your implementation
- Define new profiles wisely
- Rinse, recycle, repeat

Questions?