PRACTICAL TIPS FOR Z/OS VULNERABILITY SCANNING & PROACTIVE SECURITY MANAGEMENT

Key Resources, Inc.
ray.overby@kr-inc.com
(312) KRI-0007
www.kr-in.com

key resources, inc.
Ensuring System Integrity For z/Series

Ray Overby

- SKK - ACF2 Developer (1981-1988)
- Key Resources, Inc. incorporated 1988
  - Systems Programming
  - Security Audit and Reviews
  - Security Product Development
- Developed ESM Conversion and Merge products
- Consulting & Development for RACF add-on ISV
- Developed Automated Penetration Testing product
- z/OS Internals & Security expert

Overview

"Real" World Experiences:

1. A Terrorist Threat
2. An Unexpected Client Request
3. Recent RACF-L Conversations

First Example - Terrorist Threat

- August 1998, somewhere on the West Coast.....
- Large Municipal Government entity
- MVS System
- ACF2 was the ESM
- Threat phoned in to take down the MVS system
- Client requested assessment to determine vulnerability

Terrorist Threat

- **The following was reviewed:**
  - Hardware Setup
  - IPL and subsystem startup parameters
  - ESM Configuration
  - Policies and Procedures for maintaining and upgrading system's
  - System Exits
  - Nothing obvious turned up
  - Penetration testing was the next step

Terrorist Threat

Design penetration test cases:

- Focused on System Datasets
- Created a list:
  - LINLIST, LPALIB, APF list, …….
  - IPL parameter datasets
  - Subsystem startup datasets
- Created a list of low level users of the system.
- Ran ESM reports reviewing access to the list of files.
- Smoking gun not found.

Terrorist Threat

Implement Pen Tests:

- Test cases tried to updated the files in the list.
- Ran with the authority from the list of test users.
- These tests identified the ability to update certain system datasets.
- This caused re-examination of ESM global options.
- Finally located root cause of the error.
- Security for certain DASD volumes was done at volume level bypassing security at the dataset level.
- One of the volumes was the system residence volume.

Terrorist Threat

What did assessment uncover?

- Threat was credible.
- ESM configuration changes were recommended.
- Re-tested after the changes using penetration tests.
- Client considered assignment successfully completed.

Terrorist Threat

Tips:

- Make sure you understand the impact of the changes you make on the system.
- Error might have been identified in test phase or very early on in production if penetration testing had been a standard quality assurance process.
- No real tools exist so you have to be creative (sticky tape and string).
- You can not cover everything - start small and work your way up.
- Identify critical assets (ex - system vs. application datasets).

Second Example - System Integrity Exploitation

- 2005; Somewhere on the Eastern Seaboard
- z/OS 1.6
- Large Financial Corporation
- Assignment: Bypass z/OS Installation Controls
- Client never disclosed why they needed external review
- We were asked to focus on system integrity

System Integrity Exploitation

- 1st day on site (after fingerprinting and background checks), before TSO userids were available.
- Explored the system using sponsors TSO session.
- MXI CBT Shareware program was installed (files 409|410).
- Good practice to see what system exits were in place (SYSX|DYNX).
- Check out cool MXI command GQE.
- List allocated common storage (requires storage tracking) by SP and storage key.
- Large amount of CSA Key 8 storage allocated on system

System Integrity Exploitation

- MXI GQE output showed some Key 8 CSA might be a SMF exit.
- Verified Key 8 storage was actually a SMF IEFUJI exit.
- Turns out ISV had loaded SMF Exit IEFUJI exit into CSA Key 8.
- Informed client an exploitable integrity vulnerability found.
- Client's Senior Systems programmer did not believe it was possible to exploit a vulnerability in z/OS.
- Vulnerability exploit written in REXX.
- REXX exec dynamically elevated user authority – set RACF Privileged attribute for a TSO user – allowed access to most, if not all RACF protected resources.
- REXX exec could have crashed the system.
- The exploit written for this vulnerability could have been used to compromise all data on the system (i.e. – financial assets).

System Integrity Exploitation

- ESM is not capable of stopping, monitoring, or reporting on this type of vulnerability
- Compliance violation for every compliance regulation there is!
- To his credit client systems programmer accepted the evidence.
- Systems programmer admitted that this type of activity was beyond his level of expertise.
- He was a very experienced systems programmer.

System Integrity Exploitation

Comments:

- Installation had some sort of loss. No details ever shared.
  - “Conspiracy of Silence”
  - If we don't talk about it, it did not happen and it can't hurt us.
- Installations do not have the expertise to perform integrity assessments.
  - In general, sr. systems programmers do not have the same skillsets as those from 10 years ago.
- It is a matter of when, not if, Key 8 common storage will be compromised and used in a vulnerability exploit.
  - For those that don't think this is an issue, they are putting their companies at risk.
- Eliminating Key 8 common storage on your system removes an attack vector for a hacker.
  - Common criteria labs will treat any Key 8 common storage as an exploitable vulnerability.

System Integrity Exploitation

Comments:

- There is an IPL parameter in `DIAGxx (ALLOWUSERKEYCSA)` that controls the ability to allocate Key 8 common storage.
- This `ALLOWUSERKEYCSA` setting can be changed dynamically via `SET DIAG=xx` operator command.
- Code has been located in the "wild" to change the `ALLOWUSERKEYCSA` setting dynamically (if off turn on, perform the Key 8 CSA allocation, then turn back off).
- ISVs are aware of the Key 8 CSA issue and most have been moving to eliminate any common storage user key usage.
- IBM recommends that you not specify `ALLOWUSERKEYCSA(YES)` as user key CSA creates a security risk as any unauthorized program can modify it.

System Integrity Exploitation

Tips:

- Update your security policy to indicate that no Key 8 CSA memory usage is allowed.
- DIAGxx Parmlib setting at IPL should be ALLOWUSERKEYCSA(NO).
    - Migration of ISV or installation code to newer version may be required. Check with your ISV or your installation developers.
- Do not allow dynamic changing of the ALLOWUSERKEYCSA setting
    - ISV or installation written programs may need to be upgraded.
- You must still monitor Key 8 CSA storage allocation.
- Identify any abusers and remediate them.
- Any ISV that does not, should be replaced.

Third Example - A RACF L posting

We have a storage area that we obtain at the first CICS address space start up. The area is referenced by all CICS regions - but only a couple do any actual updating. The code we use for this is ----------

A RACF L posting

```
LA     R1,SVCSAVE   HOLD AREA FOR SVC 255
SVC    255          GET INTO SUP. STATE WITH KEY 0
STORAGE OBTAIN,LENGTH=20480,SP=241,KEY=9
ST     R1,MVSCSADR  STORE AREA ADDR. IN CSAEXT
IC     R11,=X'80'
SPKA   0(R11)       CHANGE TO KEY 8 CICS
MODESET MODE=PROB   SWITCH TO PROBLEM STATE
```


A RACF L posting

Vulnerabilities May Be Added

- By well meaning Systems Programmers:
  - Who need a specific function
  - Who did not understand the implications
  - Who have long since left or retired

- Removal will likely require re-designing or eliminating the function.

A RACF L posting

Tips:

- Remove “magic” SVCs or other authorization mechanisms that can compromise your system.
- Redesign the function
  - Function must be accomplished in a manner that does not compromise system integrity.

A RACF L posting - ESM Configuration Exploit

- RACF L (late Sept/early Oct 2012)
- Discussion labeled "Mysterious Dataset Access?“
- Access was allowed to a dataset and it should not have been
- Dataset contained sensitive information
- Breach was reported by company employee

A RACF L posting - ESM Configuration Exploit

```
ADDSD 'SYS9.RACF.*.**' UACC(NONE) OWNER(SYS9)
PERMIT 'SYS9.RACF.*.**' ID(SYSADM) ACCESS(ALTER)
PERMIT 'SYS9.RACF.*.**' ID(SYSPRG) ACCESS(ALTER)
ADDSD 'SYS9.*.**' UACC(READ) OWNER(SYS9)
PERMIT 'SYS9.*.**' ID(SYSADM) ACCESS(ALTER)
PERMIT 'SYS9.*.**' ID(SYSPRG) ACCESS(ALTER)
PERMIT ‘SYS9.*.**’ ID(APPLGRP) ACCESS(READ)


RALT GLOBAL DATASET ADDMEM('SYS9.RACF.*.**'/NONE)
RALT GLOBAL DATASET ADDMEM('SYS9.*.**'/READ)
SETR GLOBAL(DATASET) REFRESH
```


A RACF L posting - ESM Configuration Exploit

- An "undercutting" global entry `'SYS9.RACF.*.**'/NONE` should have caused RACF to look at the `'SYS9.RACF.*.**'` DATASET profile for access.
- `'SYS9.RACF.*.**'` DATASET profile would deny access.
- If undercutting Global entry removed, then READ access would have been allowed.
- If undercutting Global entry added but refresh not done, would still allow access.
- Based on posts on RACF L the previous two bullet items are the likely cause of the problem.

A RACF L posting - ESM Configuration Exploit

Comments:

- It is likely that these types of problems are reported from time to time.
- When reported, you react by performing root cause analysis and fixing the error.
- Depending upon the sensitivity of the protected asset, that may not be enough.
- Penetration testing could have identified this problem very early in its existence.
- Not performing penetration testing will put your company at risk.
- Compliance standards call for continuous monitoring.
- Penetration testing can be used to, at least partially, cover this requirement.

A RACF L posting - ESM Configuration Exploit

- You can use the following to test READ access to datasets
- Requires RACF SURROGATE to be implemented

```
Replace with Valid job card + USER=execution userid
//CHECKIT  EXEC PGM=IEBGENER
//SYSIN    DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD DISP=SHR,DSN=SYS9.RACF.REC102
//SYSUT2   DD DUMMY
//
```


A RACF L posting - ESM Configuration Exploit

Tips:

- Verify the changes you make perform what you are trying to do
  - *Create a set of test case's to verify changes work*
- Verify the changes you make don't have unintended consequences
  - *Create a set of test cases to verify no unintended consequences*

In Summary

- Penetration testing can be helpful in:
  - Verifying that ESM changes actually work
  - Do not create unintended consequences
- A robust set of penetration tests can help eliminate ESM based vulnerabilities (proactive instead of reactive).
- Penetration testing is obligatory by all compliance guidelines.

Questions?

Key Resources, Inc.
ray.overby@kr-inc.com
(312) KRI-0007
www.kr-inc.com