Is Your z/OS System Secure?

Ray Overby
Key Resources, Inc.
Info@kr-inc.com
(312) KRI-0007

A complete z/OS audit will:

- Evaluate your z/OS system
- Identify vulnerabilities
- Generate exploits if necessary
- Require installation remediation
- Require vendor remediation that installation will apply

What is a Vulnerability?

- A weakness in the system
- Allows an attacker to circumvent installation controls
- Could be related to hardware configuration
- Could be related to system configuration parameters
- Could be related to security system configuration
- Could be related to system integrity

What's an Exploit?

- An exploit is a way of taking advantage of a vulnerability
- Exploits are based upon one or more vulnerabilities
- An exploit contains a check list
  - Follow the directions
  - Perform each step

What is an exploit?

- With an exploit, you can bypass the installation controls
  - And gain unauthorized access to data
  - Without proper permissions
  - Without proper logging (SMF)
- Exploits can be created in your environment
- Exploits can be imported from outside sources
- Exploits have requirements to implement

Exploit requirements

- May require certain authorities such as:
  - Ability to submit batch jobs or TSO access
  - Access to assembler and/or linkage editor
  - Access to Rexx
  - Ability to create new or modify existing data sets
- Will not require:
  - Extra-ordinary security authority

Exploit requirements

- May require certain software installed
- May require certain software levels
- May require certain features implemented

Vulnerabilities Can be Caused By

- Poor hardware configuration
- Poor system configuration parameters
- Poor Security System Controls
- System Integrity Violations

Vulnerabilities Can

- Be exploited by knowledgeable insiders (high level of technical expertise)
- Script kiddies (low or lower level of technical expertise)
- Cause Compliance violations
- Cause loss or modification of confidential information without generating SMF or other log records

Vulnerabilities can be caused by:

- Poor Hardware Configuration
  - Shared DASD, but non-shared security database
  - May allow test system access to production data
- Poor System Controls
  - Failure to properly secure APF libraries
- Poor System Configuration
  - Program Properties Table entries

System Integrity Violations

- What are they?
- Why should you care?

System Integrity vs. System Security

- Barry Schrager formed the SHARE Security Project in 1972
- To develop security requirements for future IBM Operating Systems
- Problem that could not be overcome was that any security rules could be bypassed if the defined Operating System Interfaces could be circumvented
- The Security Project conclusion was –

  There can be no System Security without Operating System Integrity

Also in 1972

- Eldon Worley (original author of IBM's RACF) and Barry Schrager (original author of CA's ACF2) gave a SHARE presentation on their concepts for data security
- Several IBMers watched for their customers' reaction to data security in IBM Operating Systems
- It must have been good … 👍
- In 1973 IBM announced its Statement of Integrity for OS/VS2

IBM's Commitment to z/OS Integrity

IBM's commitment includes design and development practices intended to prevent unauthorized application programs, subsystems, and users from bypassing z/OS security – that is, to prevent them from gaining access, circumventing, disabling, altering, or obtaining control of key z/OS system processes and resources unless allowed by the installation.

Specifically, z/OS "System Integrity" is defined as the inability of any program not authorized by a mechanism under the installation's control to circumvent or disable store or fetch protection, access a resource protected by the z/OS Security Server (RACF®), or obtain control in an authorized state; that is, in supervisor state, with a protection key less than eight (8), or Authorized Program Facility (APF) authorized.

In the event that an IBM System Integrity problem is reported, IBM will always take action to resolve it.

Three Security Systems

- RACF developed by IBM and introduced in 1976
- ACF2 developed by SKK and introduced in 1978 (now owned by CA)
- Top Secret developed by CGA Allen and introduced in 1981 (now owned by CA)
- ***ALL DEPEND ON PROPER z/OS CONFIGURATION, SECURITY SYSTEM CONTROLS AND SYSTEM INTEGRITY***

What does it mean?

- If you have a secure hardware configuration
- And if you have secure system configuration parameters
- And if you have secure installation security controls
- But you don't have integrity – you are NOT secure!

System Integrity Vulnerabilities

- Are violations of the z/OS statement of integrity
- Cannot be addressed by hardware or system configuration changes
- Cannot be address by security system controls changes
- Cannot be remediated by the installation

System Integrity Vulnerabilities

- Independent of the Security System – ACF2, RACF or Top Secret
- Must be remediated by the Code Owner
- You are dependent on the Code Owner to address them!

Who is the Code Owner?

- IBM in the case of the z/OS Operating System
- Vendors for Program Products - could be IBM or other Independent Software Vendors
- Installation Staff for locally developed Operating System SVCs, System Exits, APF authorized programs, etc.
- ? for code obtained from other sites

Lets recap what we have learned

- Vulnerabilities don't do anything – they just are there
- Vulnerabilities are weaknesses in the system
- Vulnerabilities could be located in:
  - Hardware configuration
  - System configuration parameters
  - Security implementation
  - Authorized programs (Integrity)

Lets recap what we have learned

- Exploits do something
- Exploits allow a user access to something outside of installation control
- Exploits are based upon one or more vulnerabilities
- Exploits don't require any extra-ordinary security authority

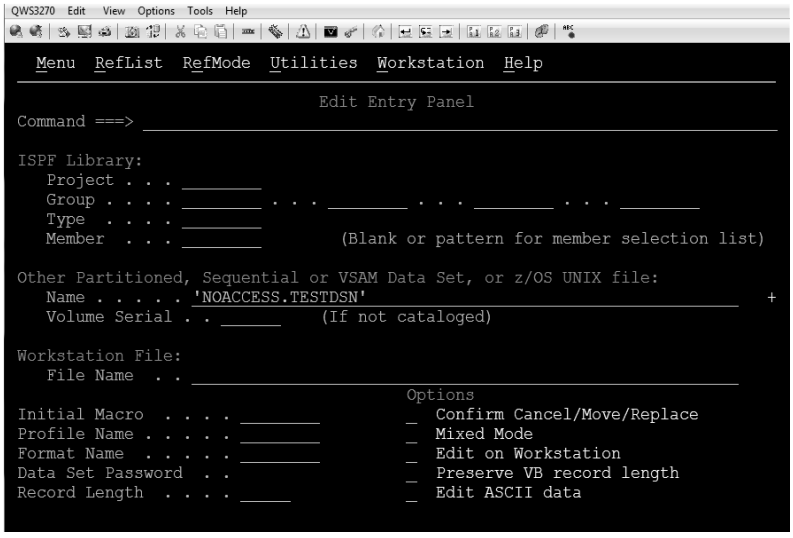
Lets recap what we have learned

- Exploits can lead to compliance violations
- Exploits, and the activity they hide, may not leave any audit trail
- Exploits based on poor z/OS configuration parameters can be remediated by you
- Exploits based on poor security implementation can be remediated by you
- Exploits based upon system integrity exposures **CANNOT** be remediated by you

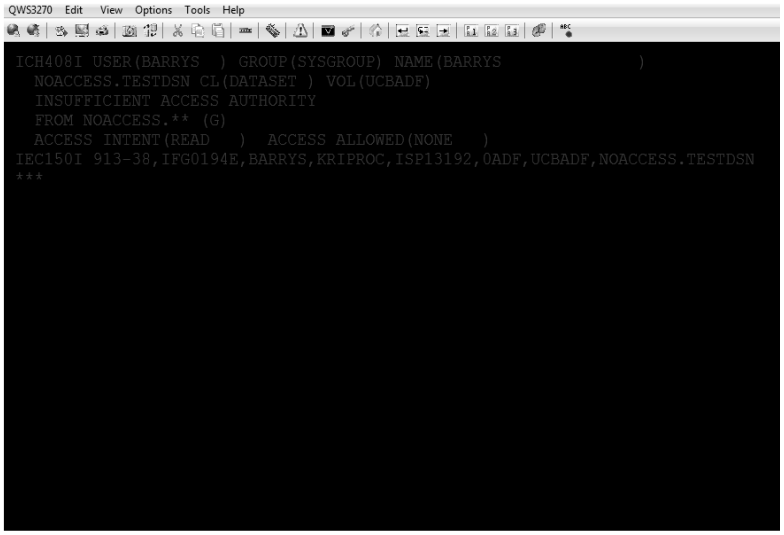
Lets recap what we have learned

- System integrity exposures are the same as not protecting your APF authorized libraries
- This means that if not properly protecting your APF libraries is a compliance violation then integrity exposures are also……..
- You have integrity exposures on your systems today!

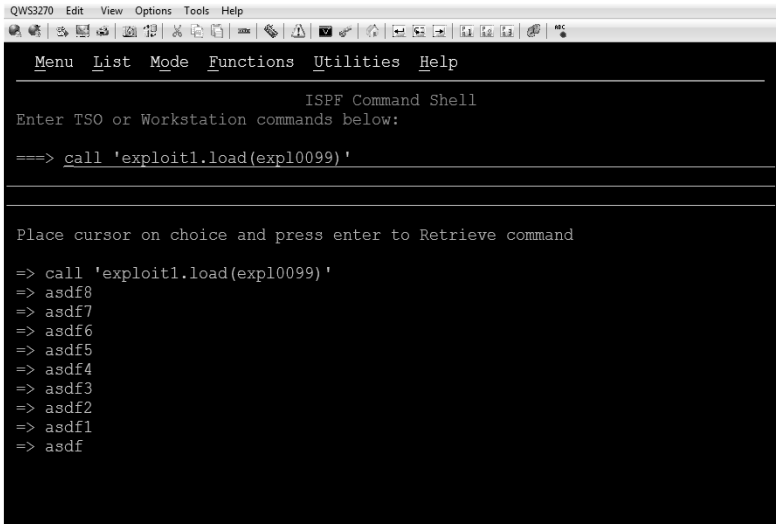
Access a Dataset



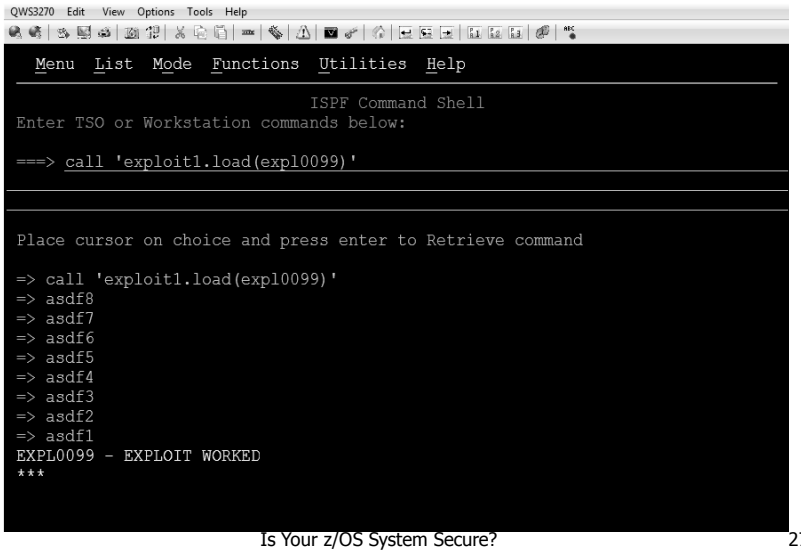
Denied by RACF – 913 ABEND!!



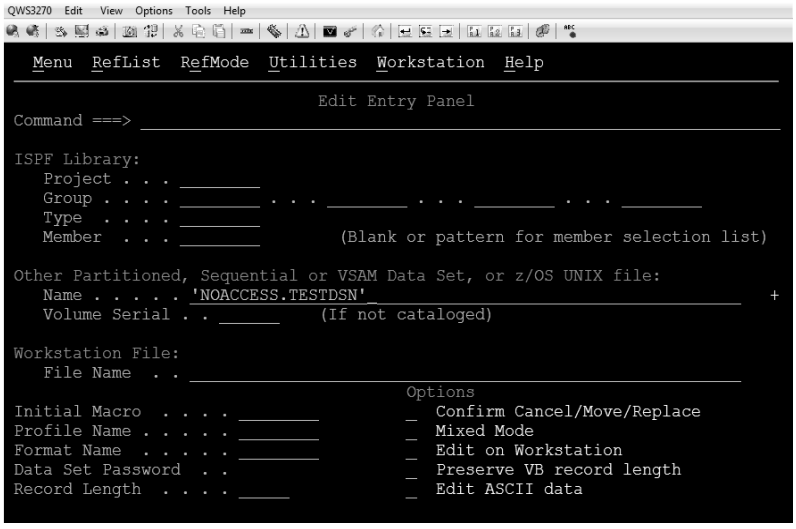
Run an Exploit



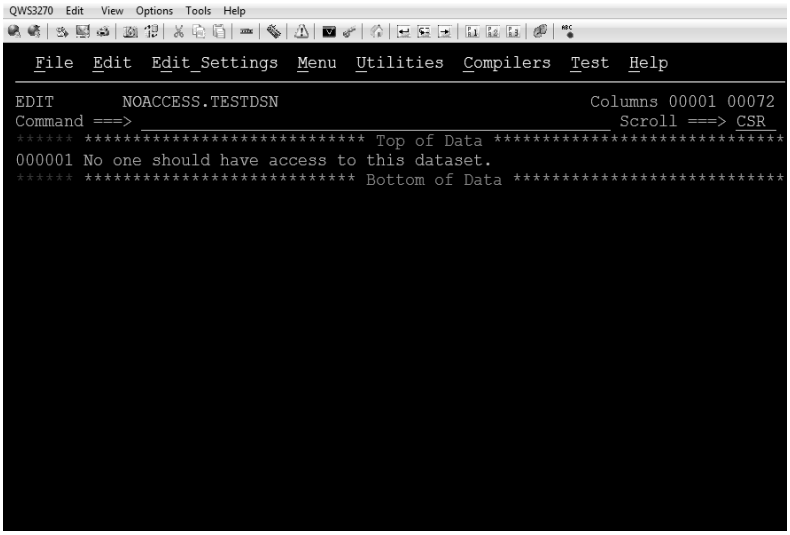
Now in RACF PRIVILEGED!!



Access the Dataset Again



Now Have Access!!



System integrity vulnerability example

- Similar to poor security system example
- Write a program to dynamically modify your security credentials
- System integrity vulnerability does NOT require APF authorized library – any load module library will do!
- You can remediate the poor security system example
- You cannot remediate the system integrity vulnerability

Why a Vulnerability Testing Process?

- Helps safeguard your organization
- Prevents financial loss through fraud
- Stops hackers, spies and disgruntled employees
- Provides due diligence
- Provides compliance to industry standards (PCI Requirement 11.3), government standards (NIST 800-53), and International Standards (ISO 27001)

2008 Strategic Counsel Survey

- Commissioned by CA Technologies
- Internal Breaches are Rising
  - 2003 – 15% of breaches
  - 2006 – 42% of breaches
  - 2008 – 44% of breaches
- The biggest security threats are from the inside!
- And, they are increasing!

2010 PacketMotion Survey

US Government Agencies surveyed

- 59% said employees are the biggest threat
- 18% said outsiders, including contractors, were biggest threat
- 9% said Hackers & Criminals were biggest threat
- 77% said foreign government spies could be planted in their agencies

Compliance Requirements

- Penetration Testing
- Vulnerability Scans

- *We always thought these were for networks and non-mainframe servers because mainframes were inherently secure*
- *But, although mainframes are **more** secure, they still have vulnerabilities*

PCI Requirement 11.3 Guidance

Before applications, network devices, and systems are released into production, they should be hardened and secured using security best practices (per Requirement 2.2).

Vulnerability scans and penetration tests will expose any remaining vulnerabilities that could later be found and exploited by an attacker.

NIST 800-53 – Control CA-2 Security Assessments

- The organization includes, as part of a security control assessment, malicious user testing and penetration testing

ISO/IEC 27001

15.2.2 Technical compliance checking

Information systems should be regularly checked for compliance with security implementation

Standards

Compliance checking also covers, for example, penetration testing and vulnerability assessments, which might be carried out by independent experts specifically contracted for this purpose

z/OS Audit Tools

- First was SKK's Examine/MVS, now known as CA-Audit
- Then Vanguard's Analyzer
- Then Consul's Audit (now part of zSecure)
- Vanguard's Configuration Manager (DISA STIGs for RACF)
- Key Resources' Vulnerability Analysis Tool for z/OS Vulnerability Assessments

DISA STIGs

- Compare your z/OS Configuration and Security System Controls against a standard
- Defense Information Systems Agency (DISA) created the Security Technical Installation Guidelines (STIGs) for many platforms
- Including z/OS Systems with RACF, ACF2 and Top Secret
- http://iase.disa.mil/stigs/stig/index.html

Automated DISA STIG Analysis

- If you are RACF …

  Vanguard's Configuration Manager automatically compares your system's configuration and security controls against the DISA STIG standards
- Or, DISA supplies a massive REXX Exec that uses CA-Examine to examine your configuration parameters and controls

But System Integrity Exposures …

- Everything depends on System Integrity
- It's like locking the front door, but not checking to see if the side windows are open
- System Integrity exposures can put your system at risk to insiders
- And according to the surveys, employees are becoming the greatest risk

z/OS Integrity Exposures

- Authorized programs, SVCs, PCs, whose sole purpose is to place the caller in an authorized state
- Errors in the implementation of:
  - Independent Software Vendor Products
  - Homegrown programs
  - Operating System Exits
  - Operating System extensions
  - z/OS

Types of Integrity Exposures

- Integrity exposures are in authorized programs, SVCs & PCs
- Store into caller supplied addresses
- Expose fetch protected storage
- Branch to caller supplied addresses
- APF authorized programs that don't require any special authority but do "authorized" things

Installations Create Their Own Vulnerabilities

- Systems Programmers add vulnerabilities for convenience
- Or they are added without performing true research and knowledge of the impact of what they are installing on your systems

A True 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 --

Do you think they have a Vulnerability?

```
 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
```

Vulnerabilities exist in shared code

- On the www.cbttape.org website
- Started as a tape by Connecticut Bank & Trust
- Contains MVS "shareware"
- One great enhancement to ISPF
- Submitted by a very reputable institution
- Contains a "get me in supervisor state" SVC

Vulnerabilities may have been added

- By well meaning Systems Programmers
- Who wanted a function
- Who did not think of the implications
- Who have long since left or retired

- But they STILL EXIST ON YOUR SYSTEMS!!!

Vendors Introduce Vulnerabilities

- One, now fixed, from an ISV was exploitable by an **11 line REXX Exec!** Which placed the caller in RACF Privileged State!
- Sometimes Vendors fix vulnerabilities but you may not have applied the maintenance
- There are integrity vulnerabilities on your systems today!

What do you do?

- Apply all IBM Integrity fixes
  - Integrity PTFs may be marked integrity, but not critical
  - Many installations apply critical PTFs but not integrity PTFs
  - Pay attention!
- Require all your ISVs to provide a commitment to System Integrity similar to what IBM has provided for the z/OS system itself

What do you do?

- Review all installation developed authorized code for integrity exposures
  - Retain outside experts if necessary
- Review all authorized code obtained from outside the company
  - Retain outside experts if necessary

What do you do?

- Periodically review the state of your z/OS system to:
  - Assure that older system integrity issues have been resolved
  - And no new ones have been introduced

Questions?

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