KEY RESOURCES
The Key to zSystems Integrity

ARTICLE

z/OS Operating System Vulnerabilities (2013-2017)

Cynthia Overby

March 2, 2018

Mainframe Integrity Vulnerabilities

Key Resources, Inc. (KRI) began tracking system integrity vulnerabilities in 2013 at the inception of our Integrity Assessment Review practice.

With respect to z/OS System Integrity, the attack surface we are addressing is the interface between problem state programs and supervisor state programs. Supervisor state programs include Program Call (PC) routines, Supervisor State Programs (SVC's), and Authorized Programs (programs residing in APF libraries). Privileged instructions are only executed in "supervisor state". Problem state programs (user programs) can only access and modify data by requesting a function from a supervisor state program.

When one of these supervisor state programs is coded incorrectly and a user program is allowed to bypass the controls used to obtain supervisor state it has broken through the attack surface and is able to circumvent the integrity of the system. This is because a supervisor state program can modify any area of memory as well as potentially assume credentials of other users including administrators or system personnel.

Categories of Vulnerabilities

KRI recognizes the following categories of vulnerabilities. Common to all of the listed categories is that the vulnerability can be triggered or exploited by a non-authorized user with no special privileges.

Trap Door vulnerabilities provide the non-authorized user the ability to call a user provided routine in an authorized state. (Either system key (0-7) or supervisor state.) This enables a hacker to directly make any changes to the active environment, including:

- Direct invocation of any authorized z/OS interface
- Changing user's state (authorized)
- Making changes to the operating system
- Making changes to system or application data
- Impersonating other users
- Disabling logging auditing (SMF)
- Disrupt z/OS operation (crash or failure)

Note: Exploitation of Trap Door vulnerabilities requires passing the address of a non-authorized routine to the vulnerable code. The vulnerable code then invokes the routine in an authorized state. A Trap Door exploitation is the most severe vulnerability for z/OS integrity.

Storage Alteration vulnerabilities provide the non-authorized user the ability to alter memory. While this is not as direct as a Trap Door vulnerability, it allows the manipulation of almost all virtual memory. By making changes to the their environment the hacker can:

- Changing user's state to be authorized (PSW Supervisor State)
- Making changes to the operating system
- Making changes to system or application data
- Impersonating other users
- Disabling logging auditing (SMF)
- Disrupt z/OS operation (crash or failure)

Note: Storage Alteration vulnerability exploitation is simpler than the Trap Door vulnerability, as it does not directly grant total control. However, as any storage may be altered, the exploiter can use storage alteration to obtain total control, as well as corruption of data, or causing an outage.

System instability occurs when the authorized program is invoked other than as designed and the program does not protect itself against improper invocation. This improper invocation can cause z/OS service issues including crashing the system.

Storage Reference vulnerabilities allow the non-authorized user the ability to pass fetch1 protected storage to an authorized service (SVC or PC). This enables the non-authorized user to use data or parameters for which they are not authorized.

Identify Spoofing vulnerabilities allow the non-authorized user to create alternate security credentials.

Note: Another type of vulnerability KRI began tracking in 2018 is the usage of User Key Common Storage. Allocating, obtaining, or changing common areas of virtual storage, such that the storage is in user key (8-15), will not be supported after z/OS V2R3. Any applications utilizing such storage areas will have to be repaired OR removed.

KRI categorizes these vulnerabilities as ALTER level and READ level. An ALTER level integrity vulnerability is defined as a vulnerability that when exploited will completely compromise all data on a z/OS system, as well as the system itself. Why? Because the exploiter can change their authority and allow them to alter any security parameter and gain access to all data on the system. ALTER level vulnerabilities would typically score at least 8.4 or higher using the CVSS V3 calculator (they are NOT posted in the National Vulnerability Database as they are considered proprietary information by the vendors). Fixes are provided as part of vendor maintenance contracts.

A READ level vulnerability is defined as a vulnerability that when exploited, will completely comprise some or all memory on your system. It is common practice to place sensitive data into fetch protected memory. Data placed into fetch protected memory includes clear text passwords, encryption keys, and other similar sensitive data. This sensitive data could also include installation defined sensitive data. READ level vulnerabilities would typically score at least 5.0 or higher using the CVSS V3 calculator (they are NOT posted in the National Vulnerability Database as they are considered proprietary information by the vendors). Fixes are provided as part of vendor maintenance contracts.

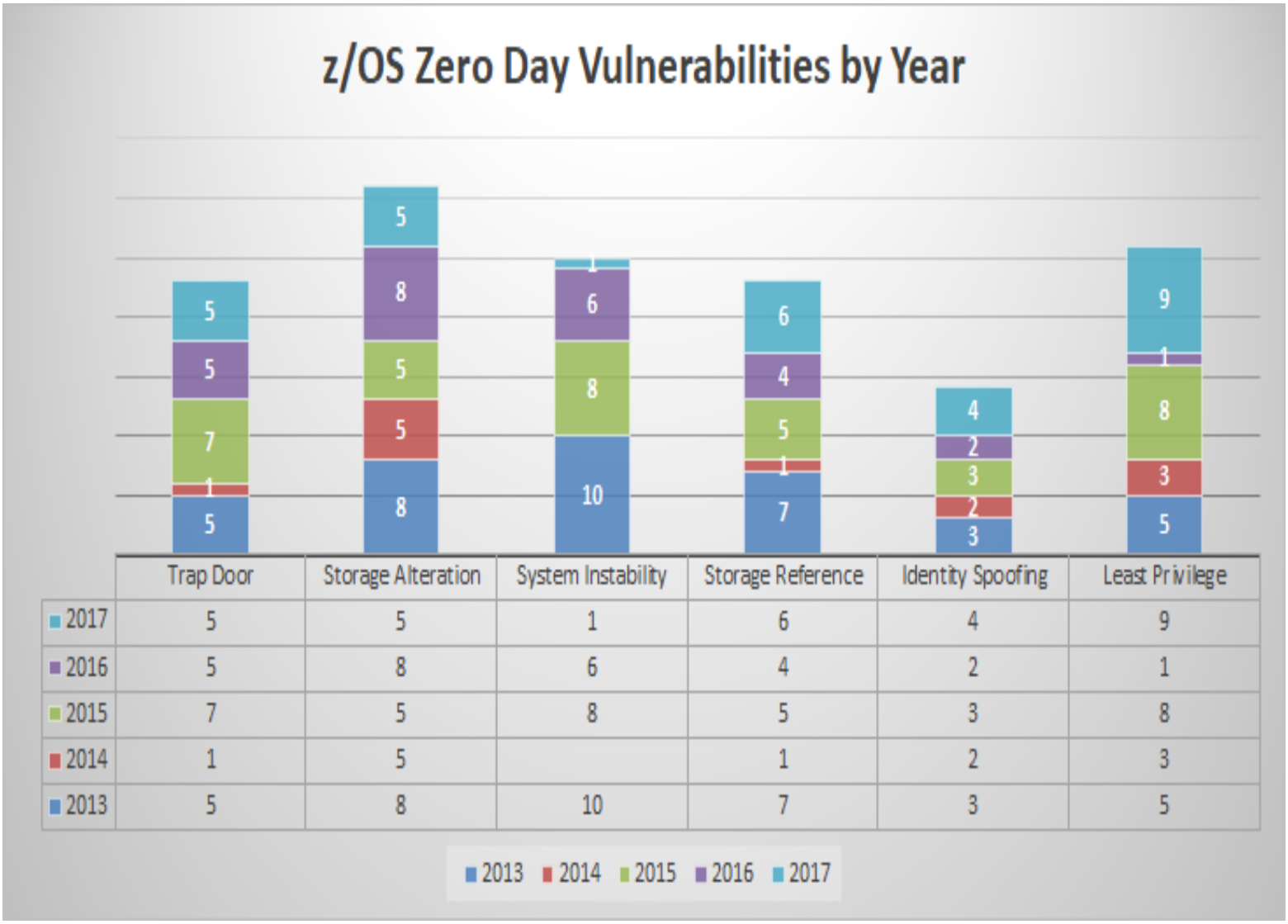
Explanation of Chart

The following chart is a synopsis of vulnerabilities KRI has found using the z/Assure® Vulnerability Analysis Program (VAP). This is proprietary software that uses interactive, binary scanning to find integrity vulnerabilities in the IBM z/OS® operating system environment. These are zero-day vulnerabilities and have been found on production mainframes running in corporate environments in the U.S. and Europe. The operating system level ranges from 1.13 to 2.2. Two vulnerabilities that were found and fixed in 1.13 in 2013 were rediscovered in newer releases in 2016 in the same product.

Authorized code that these vulnerabilities were found in comes from:

- IBM
- Independent Software Vendors (ISV's)
- Installation added programs (shareware or added system exits).

Note: Ensuring system integrity is outside the scope of the current External Security Managers. None of the big three security packages are capable of enforcing the defined security policy when integrity vulnerabilities allow users to gain unauthorized access through an exploit to circumvent z/OS system integrity and bypass the security controls.



	Trap Door	Storage Alteration	System Instability	Storage Reference	Identity Spoofing	Least Privilege
2017	5	5	1	6	4	9
2016	5	8	6	4	2	1
2015	7	5	8	5	3	8
2014	1	5		1	2	3
2013	5	8	10	7	3	5

Key Resources Inc. has decades of expertise providing software, services and consulting to enterprises running critical apps on IBM® z/OS. We help CIOs, CISOs and programmers take control of mainframe security so they can protect their data, avoid costly breaches and maintain regulatory compliance.

Contact Key Resources

Call 1-800-574-1339 ext 701

Visit www.krisecurity.com/contact